Red Hat Enterprise Linux 8.2

8.2 Release Notes

Release Notes for Red Hat Enterprise Linux 8.2

Last Updated: 2020-10-13

Red Hat Enterprise Linux 8.2 8.2 Release Notes

Release Notes for Red Hat Enterprise Linux 8.2

Legal Notice

Copyright © 2020 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at
http://creativecommons.org/licenses/by-sa/3.0/
. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

The Release Notes provide high-level coverage of the improvements and additions that have been implemented in Red Hat Enterprise Linux 8.2 and document known problems in this release, as well as notable bug fixes, Technology Previews, deprecated functionality, and other details.

Table of Contents

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

Installer and image creation

In RHEL 8.2, you can register your system, attach RHEL subscriptions, and install from the Red Hat Content Delivery Network (CDN) before package installation. You can also register your system to Red Hat Insights during installation. Interactive GUI installations, as well as automated Kickstart installations, support these new features.

For more information, see Section 5.1, “Installer and image creation” .

Infrastructure services

The **Tuned** system tuning tool has been rebased to version 2.13, which adds support for architecture-dependent tuning and multiple include directives.

For more information, see Section 5.4, “Infrastructure services” .

Security

System-wide cryptographic policies now support **customization**. The administrator can now define a complete policy or modify only certain values.

RHEL 8.2 includes the **setools-gui** and **setools-console-analyses** packages that provide tools for SELinux-policy analysis and data-flow inspections.

SCAP Security Guide now provides a profile compliant with the Australian Cyber Security Centre (ACSC) **Essential Eight** Maturity Model.

See Section 5.5, “Security” for more information.

Dynamic programming languages, web and database servers

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

- **Python 3.8**
- **Maven 3.6**

See Section 5.10, “Dynamic programming languages, web and database servers” for details.

Compiler toolsets

The following compiler toolsets have been updated in RHEL 8.2:

- **GCC Toolset 9**
- **Clang and LLVM Toolset 9.0.1**
- **Rust Toolset 1.41**
- **Go Toolset 1.13**

See Section 5.11, “Compilers and development tools” for more information.

Identity Management

Identity Management introduces a new command-line tool: **Healthcheck**. **Healthcheck** helps users find problems that might impact the fitness of their IdM environments.

Identity Management now supports Ansible roles and modules for installation and management. This update makes installation and configuration of IdM-based solutions easier.

See Section 5.12, "Identity Management" for more information.

The web console

The web console has been redesigned to use the PatternFly 4 user interface system design.

A session timeout has been added to the web console to improve security.

See Section 5.15, "The web console" for more information.

Desktop

Workspace switcher in the GNOME Classic environment has been modified. The switcher is now located in the right part of the bottom bar, and it is designed as a horizontal strip of thumbnails. Switching between workspaces is possible by clicking on the required thumbnail.

The **Direct Rendering Manager** (DRM) kernel graphics subsystem has been rebased to upstream Linux kernel version 5.3. This version provides a number of enhancements over the previous version, including support for new GPUs and APUs, and various driver updates.

In-place upgrade

In-place upgrade from RHEL 7 to RHEL 8

The supported in-place upgrade path is:

- From RHEL 7.9 to RHEL 8.2 on the 64-bit Intel, IBM POWER 8 (little endian), and IBM Z architectures
- From RHEL 7.6 to RHEL 8.2 on architectures that require kernel version 4.14: 64-bit ARM, IBM POWER 9 (little endian), and IBM Z (Structure A). Note that these architectures remain fully supported in RHEL 7 but no longer receive minor release updates since RHEL 7.7.

For more information, see Supported in-place upgrade paths for Red Hat Enterprise Linux . For instructions on performing an in-place upgrade, see Upgrading from RHEL 7 to RHEL 8 .

Notable enhancements include:

- You can now use additional custom repositories for an in-place upgrade from RHEL 7 to RHEL 8. It is also possible to upgrade without Red Hat Subscription Manager.
- You can create your own actors to migrate your custom or third-party applications using the Leapp utility.

For details, see Customizing your Red Hat Enterprise Linux in-place upgrade .

If you are using CentOS 7 or Oracle Linux 7, you can convert your operating system to RHEL 7 using the **convert2rhel** utility prior to upgrading to RHEL 8. For instructions, see How to convert from CentOS or Oracle Linux to RHEL.

In-place upgrade from RHEL 6 to RHEL 8

To upgrade from RHEL 6.10 to RHEL 8.2, follow instructions in Upgrading from RHEL 6 to RHEL 8 .

If you are using CentOS 6 or Oracle Linux 6, you can convert your operating system to RHEL 6 using the **convert2rhel** utility prior to upgrading to RHEL 8. For instructions, see How to convert from CentOS or Oracle Linux to RHEL.

Additional resources

- **Capabilities and limits** of Red Hat Enterprise Linux 8 as compared to other versions of the system are available in the Knowledgebase article Red Hat Enterprise Linux technology capabilities and limits.
- Information regarding the Red Hat Enterprise Linux **life cycle** is provided in the Red Hat Enterprise Linux Life Cycle document.
- The Package manifest document provides a **package listing** for RHEL 8.
- Major **differences between RHEL 7 and RHEL 8** are documented in Considerations in adopting RHEL 8.
- Instructions on how to perform an **in-place upgrade from RHEL 7 to RHEL 8** are provided by the document Upgrading to RHEL 8 .
- The **Red Hat Insights** service, which enables you to proactively identify, examine, and resolve known technical issues, is now available with all RHEL subscriptions. For instructions on how to install the Red Hat Insights client and register your system to the service, see the Red Hat Insights Get Started page.

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 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

CHAPTER 2. ARCHITECTURES

Red Hat Enterprise Linux 8.2 is distributed with the kernel version 4.18.0-193, which provides support for the following architectures:

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

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

CHAPTER 3. DISTRIBUTION OF CONTENT IN RHEL 8

3.1. INSTALLATION

Red Hat Enterprise Linux 8 is installed using ISO images. Two types of ISO image are available for the AMD64, Intel 64-bit, 64-bit ARM, IBM Power Systems, and IBM Z architectures:

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

NOTE

The Binary DVD ISO image is larger than 4.7 GB, and as a result, it might not fit on a single-layer DVD. A dual-layer DVD or USB key is recommended when using the Binary DVD ISO image to create bootable installation media. You can also use the Image Builder tool to create customized RHEL images. For more information about Image Builder, see the *Composing a customized RHEL system image* document.

- Boot ISO: A minimal boot ISO image that is used to boot into the installation program. This option requires access to the BaseOS and AppStream repositories to install software packages. The repositories are part of the Binary DVD ISO image.

See the Performing a standard RHEL 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 8 is distributed through two main repositories:

- BaseOS
- AppStream

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

Content in the BaseOS repository is intended to provide the core set of the underlying OS functionality that provides the foundation for all installations. This content is available in the RPM format and is subject to support terms similar to those in previous releases of RHEL. For a list of packages distributed through BaseOS, see the Package manifest.

Content in the Application Stream repository includes additional user space applications, runtime languages, and databases in support of the varied workloads and use cases. Content in AppStream is available in one of two formats - the familiar RPM format and an extension to the RPM format called *modules*. For a list of packages available in AppStream, see the Package manifest.

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

For more information about RHEL 8 repositories, see the Package manifest.

3.3. APPLICATION STREAMS

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

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

Modules are collections of packages representing a logical unit: an application, a language stack, a database, or a set of tools. These packages are built, tested, and released together.

Module streams represent versions of the Application Stream components. For example, two streams (versions) of the PostgreSQL database server are available in the postgresql module: PostgreSQL 10 (the default stream) and PostgreSQL 9.6. Only one module stream can be installed on the system. Different versions can be used in separate containers.

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

CHAPTER 4. RHEL 8.2.1 RELEASE

Red Hat makes Red Hat Enterprise Linux 8 content available quarterly, in between minor releases (8.Y). The quarterly releases are numbered using the third digit (8.Y.1). The new features in the RHEL 8.2.1 release are described below.

4.1. NEW FEATURES

JDK Mission Control rebased to version 7.1.1

The JDK Mission Control (JMC) profiler for HotSpot JVMs, provided by the **jmc:rhel8** module stream, has been upgraded to version 7.1.1 with the RHEL 8.2.1 release.

This update includes numerous bug fixes and enhancements, including:

- Multiple rule optimizations
- A new JOverflow view based on Standard Widget Toolkit (SWT)
- A new flame graph view
- A new way of latency visualization using the High Dynamic Range (HDR) Histogram

The **jmc:rhel8** module stream has two profiles:

- The **common** profile, which installs the entire JMC application
- The **core** profile, which installs only the core Java libraries (**jmc-core**)

To install the **common** profile of the **jmc:rhel8** module stream, use:

```
# yum module install jmc:rhel8/common
```

Change the profile name to **core** to install only the **jmc-core** package.

(BZ#1792519)

Rust Toolset rebased to version 1.43

Rust Toolset has been updated to version 1.43. Notable changes include:

- Useful line numbers are now included in **Option** and **Result** panic messages where they were invoked.
- Expanded support for matching on subslice patterns.
- The **matches!** macro provides pattern matching that returns a boolean value.
- **item** fragments can be interpolated into traits, impls, and extern blocks.
- Improved type inference around primitives.
- Associated constants for floats and integers.

To install the Rust Toolset module, run the following command as **root**:

```
# yum module install rust-toolset
```

For usage information, see the Using Rust Toolset documentation.

(BZ#1811997)

Containers registries now support the skopeo sync command

With this enhancement, users can use **skopeo sync** command to synchronize container registries and local registries. The **skopeo sync** command is useful to synchronize a local container registry mirror, and to populate registries running inside of air-gapped environments.

The **skopeo sync** command requires both source (**--src**) and destination (**--dst**) transports to be specified separately. Available source and destination transports are **docker** (repository hosted on a container registry) and **dir** (directory in a local directory path). The source transports also include **yaml** (local YAML file path). For information on the usage of **skopeo sync**, see the **skopeo-sync** man page.

(BZ#1811779)

Configuration file container.conf is now available

With this enhancement, users and administrators can specify default configuration options and command-line flags for container engines. Container engines read the **/usr/share/containers/containers.conf** and **/etc/containers/containers.conf** files if they exist. In the rootless mode, container engines read the **$HOME/.config/containers/containers.conf** files.

Fields specified in the **containers.conf** file override the default options, as well as options in previously read **containers.conf** files. The **container.conf** file is shared between Podman and Buildah and replaces the **libpod.conf** file.

(BZ#11826486)

You can now log into and out from a registry server

With this enhancement, you can log into and logout from a specified registry server using the **skopeo login** and **skopeo logout** commands. The **skopeo login** command reads in the username and password from standard input. The username and password can also be set using the **--username** (or **-u**) and **--password** (or **-p**) options.

You can specify the path of the authentication file by setting the **--authfile** flag. The default path is **${XDG_RUNTIME_DIR}/containers/auth.json**. For information on the usage of **skopeo login** and **skopeo logout**, see the **skopeo-login** and **skopeo-logout** man pages, respectively.

(JIRA:RHELPLAN-47311)

You can now reset the podman storage

With this enhancement, users can use the **podman system reset** command to reset **podman** storage back to initial state. The **podman system reset** command removes all pods, containers, images and volumes. For more information, see the **podman-system-reset** man page.

(JIRA:RHELPLAN-48941)

CHAPTER 5. NEW FEATURES

This part describes new features and major enhancements introduced in Red Hat Enterprise Linux 8.2.

5.1. INSTALLER AND IMAGE CREATION

Ability to register your system, attach RHEL subscriptions, and install from the Red Hat CDN

In RHEL 8.2, you can register your system, attach RHEL subscriptions, and install from the Red Hat Content Delivery Network (CDN) before package installation. Interactive GUI installations, as well as automated Kickstart installations, support this feature. Benefits include:

- The use of the smaller Boot ISO image file removes the need to download the larger Binary DVD ISO image file.
- The CDN uses the latest packages that result in a fully subscribed and up-to-date system immediately after installation. There is no requirement to install package updates after installation.
- Registration is performed before package installation, resulting in a shorter and more streamlined installation process.
- Integrated support for Red Hat Insights is available.

(BZ#1748281)

Ability to register your system to Red Hat Insights during installation

In RHEL 8.2, you can register your system to Red Hat Insights during installation. Interactive GUI installations, as well as automated Kickstart installations, support this feature.

Benefits include:

- Easier to identify, prioritize, and resolve issues before business operations are affected.
- Proactively identify and remediate threats to security, performance, availability, and stability with predictive analytics.
- Avoid problems and unplanned downtime in your environment.

(BZ#1746391)

Image Builder now offers cloud-init support for creating Azure images

With this enhancement, cloud-init support is available for Azure images created by Image Builder. As a result, the creation of on-premise images with fast-provisioning and the ability to add custom data is available to customers.

(BZ#1754711)

5.2. SOFTWARE MANAGEMENT

User-Agent header string now includes information read from the **/etc/os-release** file

With this enhancement, the **User-Agent** header string, which is normally included with the HTTP requests made by DNF, has been extended with information read from the **/etc/os-release** file.

To obtain more information, see **user_agent** in the **dnf.conf(5)** man page.

(BZ#1676891)

All dnf-automatic.timer timer units now use the real-time clock by default

Previously, the **dnf-automatic.timer** timer units used the monotonic clock, which resulted in unpredictable activation time after the system boot. With this update, the timer units run between 6 a.m. and 7 a.m. If the system is off during that time, the timer units are activated within one hour after the system boot.

(BZ#1754609)

The createrepo_c utility now skips packages whose metadata contains the disallowed control characters

To ensure a valid XML, the package metadata must not contain any control characters, with the exception of:

- the horizontal tab
- the newline character
- the carriage return character

With this update, the **createrepo_c** utility does not include packages with metadata containing disallowed control characters in a newly created repository, and returns the following error message:

```
C_CREATEREPOLIB: Critical: Cannot dump XML for PACKAGE_NAME (PACKAGE_SUM):
Forbidden control chars found (ASCII values <32 except 9, 10 and 13)
```

(BZ#1743186)

5.3. SHELLS AND COMMAND-LINE TOOLS

opencv rebased to version 3.4.6

The **opencv** packages have been upgraded to upstream version 3.4.6. Notable changes include:

- Support for new Open CL parameters, such as **OPENCV_OPENCL_BUILD_EXTRA_OPTIONS** and **OPENCV_OPENCL_DEVICE_MAX_WORK_GROUP_SIZE**.
- The **objdetect** module now supports QR code detection algorithm.
- Multiple new methods, such as **MatSize::dims** or **VideoCapture::getBackendName**.
- Multiple new functions, such as **drawFrameAxes** or **getVersionMajor**.
- Various performance improvements, including improvements of the GaussianBlur function, **v_load_deinterleave** and **v_store_interleave** intrinsics when using SSSE3 instructions.

(BZ#1694647)

5.4. INFRASTRUCTURE SERVICES

graphviz-python3 is now distributed in the CRB repository

This update adds the **graphviz-python3** package to RHEL 8. The package provides bindings required for usage of the Graphviz graph visualization software from Python.

Note that the **graphviz-python3** package is distributed in the unsupported CodeReady Linux Builder repository (CRB).

(BZ#1704875)

tuned rebased to version 2.13.0

The **tuned** packages have been upgraded to upstream version 2.13.0. Notable enhancements include:

- Architecture-dependant tuning framework has been added.
- Support for multiple include directives has been added.
- Tuning in the **sap-hana**, **latency-performance**, and **realtime** profiles has been updated.

(BZ#1738250)

powertop rebased to version 2.11

The **powertop** package has been upgraded to version 2.11, which provides a following notable change:

- Support for the EHL, TGL, ICL/ICX platforms

(BZ#1716721)

BIND now supports .GeoIP2 instead of GeoLite Legacy GeoIP

The GeoLite Legacy GeoIP library is no longer supported in BIND. With this update, GeoLite Legacy GeoP has been replaced with GeoIP2, which is provided in the **libmaxminddb** data format.

Note that the new format may require some configuration changes, and the format also does not support following legacy GeoIP access control list (ACL) settings:

- geoip netspeed
- geoip org
- ISO 3166 Alpha-3 country codes

(BZ#1564443)

stale-answer now provides old cached records in case of DDoS attack

Previously, the Distributed Denial of Service (DDoS) attack caused the authoritative servers to fail with the SERVFAIL error. With this update, the **stale-answer** functionality provides the expired records until a fresh response is obtained.

To enable or disable the **serve-stale** feature, use either of these:

- Configuration file
- Remote control channel (rndc)

(BZ#1664863)

BIND rebased to version 9.11.13

The **bind** packages have been upgraded to version 9.11.13. Notable changes include:

- The **tcp-highwater** statistics variable has been added. This variable shows maximum concurrent TCP clients recorded during a run.
- The **SipHash-2-4**-based DNS Cookies (RFC 7873) algorithm has been added.
- Glue addresses for rooting priming queries are returned regardless of how the **minimal-responses** configuration option is set.
- The **named-checkconf** command now ensures the validity of the **DNS64** network prefixes.
- Automatic rollover per RFC 5011 no longer fails when the **trusted-keys** and **managed-keys** statements are both configured for the same name. Instead, a warning message is logged.
- Internationalized Domain Name (IDN) processing in the **dig** and **nslookup** utilities is now disabled by default when they are not run on terminal (for example, in a script). IDN processing in **dig** can be switched on by using the **+idnin** and **+idnout** options.

(BZ#1704328)

5.5. SECURITY

RHEL 8 now contains the DISA STIG profile

Security Technical Implementation Guides (STIG) are a set of baseline recommendations published by the Defense Information Systems Agency (DISA) to harden the security of information systems and software that might otherwise be vulnerable. This release includes the profile and Kickstart file for this security policy. With this enhancement, users can check systems for compliance, remediate systems to be compliant, and install systems compliant with DISA STIG for Red Hat Enterprise Linux 8.

(BZ#1755447)

crypto-policies can now be customized

With this update, you can adjust certain algorithms or protocols of any policy level or set a new complete policy file as the current system-wide cryptographic policy. This enables administrators to customize the system-wide cryptographic policy as required by different scenarios.

RPM packages should store policies provided by them in the **/usr/share/crypto-policies/policies** directory. The **/etc/crypto-policies/policies** directory contains local custom policies.

For more information, see the **Custom Policies** section in the **update-crypto-policies(8)** man page and the **Crypto Policy Definition Format** section in the **update-crypto-policies(8)** man page.

(BZ#1690565)

SCAP Security Guide now supports ACSC Essential Eight

The **scap-security-guide** packages now provide the Australian Cyber Security Centre (ACSC) Essential Eight compliance profile and a corresponding Kickstart file. With this enhancement, users can install a system that conforms with this security baseline. Furthermore, you can use the **OpenSCAP** suite for checking security compliance and remediation using this specification of minimum security controls defined by ACSC.

(BZ#1755194)

oscap-podman for security and compliance scanning of containers is now available

This update of the **openscap** packages introduces a new utility for security and compliance scanning of containers. The **oscap-podman** tool provides an equivalent of the **oscap-docker** utility that serves for scanning container and container images in RHEL 7.

(BZ#1642373)

setroubleshoot can now analyze and react to execmem access denials

This update introduces a new **setroubleshoot** plugin. The plugin can analyze **execmem** access denials (AVCs) and provide relevant advice. As a result, **setroubleshoot** can now suggest a possibility to switch a boolean if it allows access, or report the issue when no boolean can allow access.

(BZ#1649842)

New packages: setools-gui and setools-console-analyses

The **setools-gui** package, which has been part of RHEL 7, is now being introduced to RHEL 8. Graphical tools help inspect relations and data flows especially in multi-level systems with highly specialized SELinux policies. With the **apol** graphical tool from the **setools-gui** package, you can inspect and analyze aspects of an SELinux policy. Tools from the **setools-console-analyses** package enable you to analyze domain transitions and SELinux policy information flows.

(BZ#1731519)

Confined users in SELinux can now manage user session services

Previously, confined users were not able to manage user session services. As a result, they could not execute **systemctl --user** or **busctl --user** commands or work in the RHEL web console. With this update, confined users can manage user sessions.

(BZ#1727887)

The lvmdbusd service is now confined by SELinux

The **lvmdbusd** service provides a D-Bus API to the logical volume manager (LVM). Previously, the **lvmdbusd** daemon could not transition to the **lvm_t** context even though the SELinux policy for **lvm_t** was defined. As a consequence, the **lvmdbusd** daemon was executed in the **unconfined_service_t** domain and SELinux labeled **lvmdbusd** as unconfined. With this update, the **lvmdbusd** executable file has the **lvm_exec_t** context defined and **lvmdbusd** can now be used correctly with SELinux in enforcing mode.

(BZ#1726166)

semanage now supports listing and modifying SCTP and DCCP ports.

Previously, **semanage port** allowed listing and modifying of only TCP and UDP ports. This update adds SCTP and DCCP protocol support to **semanage port**. As a result, administrators can now check if two machines can communicate via SCTP and fully enable SCTP features to successfully deploy SCTP-based applications.

(BZ#1563742)

semanage export now shows customizations related to permissive domains

With this update, the **semanage** utility, which is part of the **policycoreutils** package for SELinux, is able to display customizations related to permissive domains. System administrators can now transfer permissive local modifications between machines using the **semanage export** command.

(BZ#1417455)

udica can add new allow rules generated from SELinux denials to existing container policy

When a container that is running under a policy generated by the **udica** utility triggers an SELinux denial, **udica** is now able to update the policy. The new parameter **-a** or **--append-rules** can be used to append rules from an AVC file.

(BZ#1732704)

New SELinux types enable services to run confined

This update introduces new SELinux types that enable the following services to run as confined services in SELinux enforcing mode instead of running in the **unconfined_service_t** domain:

- **lldpd** now runs as **lldpad_t**
- **rrdcached** now runs as **rrdcached_t**
- **stratisd** now runs as **stratisd_t**
- **timedatex** now runs as **timedatex_t**

(BZ#1726246, BZ#1726255, BZ#1726259, BZ#1730204)

Clevis is able to list policies in place for a given LUKS device

With this update, the **clevis luks list** command lists PBD policies in place for a given LUKS device. This makes it easier to find information on Clevis pins in use and pin configuration, for example, Tang server addresses, details on **tpm2** policies, and SSS thresholds.

(BZ#1766526)

Clevis provides new commands for reporting key status and rebinding expired keys

The **clevis luks report** command now provides a simple way to report whether keys for a particular binding require rotation. Regular key rotations in a Tang server improve the security of Network-Bound Disk Encryption (NBDE) deployments, and therefore the client should provide detection of expired keys. If the key is expired, Clevis suggests using the **clevis luks regen** command which rebinds the expired key slot with a current key. This significantly simplifies the process of key rotation.

(BZ#1564559, BZ#1564566)

Clevis can now extract the passphrase used for binding a particular slot in a LUKS device

With this update to the Clevis policy-based decryption framework, you can now extract the passphrase used for binding a particular slot in a LUKS device. Previously, if the LUKS installation passphrase was erased, Clevis could not perform LUKS administrative tasks, such as re-encryption, enabling a new key slot with a user passphrase, and re-binding Clevis when the administrator needs to change the **sss** threshold. This update introduces the **clevis luks pass** command that shows the passphrase used for binding a particular slot.

(BZ#1436780)

Clevis now provides improved support for decrypting multiple LUKS devices on boot

The **clevis** packages have been updated to provide better support for decrypting multiple LUKS-encrypted devices on boot. Prior to this improvement, the administrator had to perform complicated changes to the system configuration to enable the proper decryption of multiple devices by Clevis on

boot. With this release, you can set up the decryption by using the **clevis luks bind** command and updating the initramfs through the **dracut -fv --regenerate-all** command.

For more details, see the Configuring automated unlocking of encrypted volumes using policy-based decryption section.

(BZ#1784524)

openssl-pkcs11 rebased to 0.4.10

The **openssl-pkcs11** package has been upgraded to upstream version 0.4.10, which provides many bug fixes and enhancements over the previous version. The **openssl-pkcs11** package provides access to PKCS #11 modules through the engine interface. The major changes introduced by the new version are:

- If a public key object corresponding to the private key is not available when loading an ECDSA private key, the engine loads the public key from a matching certificate, if present.
- You can use generic PKCS #11 URI (for example **pkcs11:type=public**) because the **openssl-pkcs11** engine searches all tokens that match a given PKCS #11 URI.
- The system attempts to log in with a PIN only if a single device matches the URI search. This prevents authentication failures due to providing the PIN to all matching tokens.
- When accessing a device, the **openssl-pkcs11** engine now marks the RSA methods structure with the **RSA_FLAG_FIPS_METHOD** flag. In FIPS mode, OpenSSL requires the flag to be set in the RSA methods structure. Note that the engine cannot detect whether a device is FIPS-certified.

(BZ#1745082)

rsyslog rebased to 8.1911.0

The **rsyslog** utility has been upgraded to upstream version 8.1911.0, which provides a number of bug fixes and enhancements over the previous version. The following list includes notable enhancements:

- New **omhttp** module allows you to send messages over the HTTP REST interface.
- The file input module is enhanced to improve stability, error reporting, and truncation detection.
- New **action.resumeIntervalMax** parameter that can be used with any action allows capping retry interval growth at a specified value.
- New **StreamDriver.PermitExpiredCerts** option for TLS permits connections even if a certificate has expired.
- You can now suspend and resume output based on configured external file content. This is useful in cases where the other end always accepts messages and silently drops them when it is not able to process them all.
- Error reporting for the file output module is improved and now contains real file names and more information on causes of errors.
- Disk queues now run multi-threaded, which improves performance.
- You can set stricter TLS operation modes: checking of the **extendedKeyUsage** certificate field and stricter checking of the **CN/SAN** certificate fields.

(BZ#1740683)

rsyslog now provides the omhttp plugin for communication through an HTTP REST interface

With this update of the **rsyslog** packages, you can use the new **omhttp** plugin for producing an output compatible with services using a Representational State Transfer (REST) API, such as the Ceph storage platform, Amazon Simple Storage Service (Amazon S3), and Grafana Loki. This new HTTP output module provides a configurable REST path and message format, support for several batching formats, compression, and TLS encryption.

For more details, see the **/usr/share/doc/rsyslog/html/configuration/modules/omhttp.html** file installed on your system with the **rsyslog-doc** package.

(BZ#1676559)

omelasticsearch in rsyslog now supports rebindinterval

This update of the **rsyslog** packages introduces support for setting the time of periodical reconnection in the **omelasticsearch** module. You can improve performance when sending records to a cluster of Elasticsearch nodes by setting this parameter according to your scenario. The value of the **rebindinterval** parameter indicates the number of operations submitted to a node after which **rsyslog** closes the connection and establishes a new one. The default value **-1** means that **rsyslog** does not re-establish the connection.

(BZ#1692073)

rsyslog mmkubernetes now provides metadata cache expiration

With this update of the **rsyslog** packages, you can use two new parameters for the **mmkubernetes** module for setting metadata cache expiration. This ensures that deleted Kubernetes objects are removed from the **mmkubernetes** static cache. The value of the **cacheentryttl** parameter indicates the maximum age of cache entries in seconds. The **cacheexpireinterval** parameter has the following values:

- **-1** for disabling cache-expiration checks
- **0** for enabling cache-expiration checks
- greater than 0 for regular cache-expiration checks in seconds

(BZ#1692072)

audit rebased to version 3.0-0.14

The **audit** packages have been upgraded to upstream version 3.0-0.14, which provides many bug fixes and enhancements over the previous version, most notably:

- Added an option to interpret fields in the syslog plugin
- Divided the **30-ospp-v42.rules** file into more granular files
- Moved example rules to the **/usr/share/audit/sample-rules/** directory
- Fixed Audit KRB5 transport mode for remote logging

(BZ#1757986)

Audit now contains many improvements from the kernel v5.5-rc1

This addition to the Linux kernel contains the majority of enhancements, bug fixes, and cleanups related to the Audit subsystem and introduced between the version 4.18 and 5.5-rc1. The following list highlights important changes:

- Wider use of the **exe** field for filtering
- Support for v3 namespaced capabilities
- Improvements for filtering on remote file systems
- Fix of the **gid** filter rule
- Fixes of a use-after-free memory corruption and memory leaks
- Improvements of event-record association
- Cleanups of the **fanoticy** interface, Audit configuration options, and the syscall interface
- Fix of the Extended Verification Module (EVM) return value
- Fixes and cleanups of several record formats
- Simplifications and fixes of Virtual File System (VFS) auditing

(BZ#1716002)

fapolicyd rebased to 0.9.1-2

The **fapolicyd** packages that provide RHEL application whitelisting have been upgraded to upstream version 0.9.1-2. Notable bug fixes and enhancements include:

- Process identification is fixed.
- The subject part and the object part are now positioned strictly in the rule. Both parts are separated by a colon, and they contain the required permission (execute, open, any).
- The subject and object attributes are consolidated.
- The new rule format is the following:

```
DECISION PERMISSION SUBJECT : OBJECT
```

For example:

```
allow perm=open exe=/usr/bin/rpm : all
```

(BZ#1759895)

sudo rebased to 1.8.29-3.el8

sudo packages have been upgraded to upstream version 1.8.29-3, which provides a number of bug fixes and enhancements over the previous version. The major changes introduced by the new version are:

- **sudo** now writes Pluggable Authentication Module (PAM) messages to the user's terminal, if available, instead of the standard output or standard error output. This prevents possible confusion of PAM output and command output sent to files and pipes.
- The **notBefore** and **notAfter** options from LDAP and SSSD now work and display correctly with the **sudo -l** command.

- The **cvtsudoers** command now rejects non-LDAP Data Interchange Format (LDIF) input when converting from LDIF to **sudoers** and JSON formats.
- With the new **log_allowed** and **log_denied** settings for **sudoers**, you can disable logging and auditing of allowed and denied commands.
- You can now use **sudo** with the **-g** option to specify a group that matches any of the target user's groups even if no groups are present in the **runas_spec** specification. Previously, you could only do so if the group matched the target user's primary group.
- Fixed a bug that prevented **sudo** from matching the host name to the value of **ipa_hostname** from **sssd.conf**, if specified.
- A vulnerability that allowed a **sudo** user to run a command as root when the **Runas** specification disallowed **root** access with the **ALL** keyword is now fixed (CVE-2019-14287).
- The use of unknown user and group IDs for permissive **sudoers** entries, for example using the ALL keyword, is now disabled. You can enable it with the **allow_unknown_runas_id** setting (CVE-2019-19232).

(BZ#1733961)

The pam_namespace module now allows specifying additional mount options for tmpfs

The **nosuid**, **noexec**, and **nodev** mount options can now be used in the **/etc/security/namespace.conf** configuration file to respectively disable setuid bit effect, disable running executables, and to prevent files from being interpreted as character or block devices on the mounted **tmpfs** filesystem.

Additional mount options are specified in the **tmpfs(5)** man page.

(BZ#1252859)

pam_faillock can now read settings from faillock.conf configuration file

The **pam_faillock** module, a part of pluggable authentication modules (PAM), can now read settings from the configuration file located at **/etc/security/faillock.conf**. This makes it easier to set up an account lockout on authentication failures, provide user profiles for this functionality, and handle different PAM configurations by simply editing the **faillock.conf** file.

(BZ#1537242)

5.6. NETWORKING

User-space applications can now retrieve the netns id selected by the kernel

User-space applications can request the kernel to select a new **netns** ID and assign it to a network name space. With this enhancement, users can specify the **NLM_F_ECHO** flag when sending an **RTM_NETNSID netlink** message to the kernel. The kernel then sends the **netlink** message back to the user. This message includes the **netns** ID set to the value the kernel selected. As a result, user-space applications now have a reliable option to identify the **netlink** ID the kernel selected.

(BZ#1763661)

firewalld rebased to version 0.8

The **firewalld** packages have been updated to version 0.8. Notable changes include:

- This version of **firewalld** includes all bug fixes since version 0.7.0.

- **firewalld** now uses the **libnftables** JSON interface to the **nftables** subsystem. This improves performance and reliability of rule application.
- In service definitions, the new **helper** element replaces **module**.
- This version allows custom helpers to use standard helper modules.

(BZ#1740670)

ndptool can now specify a destination address in IPv6 header

With this update, the **ndptool** utility can send a Neighbor Solicitation (NS) or a Neighbor Advertisement (NA) message to a specific destination by specifying the address in the IPv6 header. As a result, a message can be sent to addresses other than just the link-local address.

(BZ#1697595)

nftables now supports multi-dimensional IP set types

With this enhancement, the **nftables** packet-filtering framework supports set types with concatenations and intervals. As a result, administrators no longer require workarounds to create multi-dimensional IP set types.

(BZ#1593711)

nftables rebased to version 0.9.3

The *nftables* packages have been upgraded to upstream version 0.9.3, which provides a number of bug fixes and enhancements over the previous version:

- A JSON API has been added to the **libnftables** library. This library provides a high-level interface to manage *nftables* rule sets from third-party applications. To use the new API in Python, install the **python3-nftables** package.
- Statements support IP prefixes and ranges, such as **192.0.2.0/24** and **192.0.2.0-192.0.2.30**.
- Support for operating system fingerprints has been added to mark packets based on the guessed operating system. For further details, see the **osf expression** section in the **nft(8)** man page.
- Transparent proxy support has been added to redirect packets to a local socket without changing the packet header in any way. For details, see the **tproxy statement** section in the **nft(8)** man page.
- The security mark support has been added.
- The support for dynamic sets updates has been improved to set updates from the packet path.
- The support for transport header port matching has been added.

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

- https://lore.kernel.org/netfilter-devel/20190624164910.defehs5giqziqnir@salvia/
- https://lore.kernel.org/netfilter-devel/20190819115807.myv6owxzblj2bthd@salvia/
- https://lore.kernel.org/netfilter-devel/20191202211737.xvmd6e6xxj4xvvjl@salvia/

(BZ#1643192)

Rules for the firewalld service can now use connection tracking helpers for services running on a non-standard port

User-defined helpers in the **firewalld** service can now use standard kernel helper modules. This enables administrators to create **firewalld** rules to use connection tracking helpers for services running on a non-standard port.

(BZ#1733066)

The whois package is now available

With this enhancement, the **whois** package is now available in RHEL 8.2.0. As a result, retrieving information about a specific domain name or IP address is now possible.

(BZ#1734183)

eBPF for tc is now fully supported

The Traffic Control (tc) kernel subsystem and the **tc** tool can attach extended Berkeley Packet Filtering (eBPF) programs as packet classifiers and actions for both ingress and egress queueing disciplines. This enables programmable packet processing inside the kernel network data path. eBPF for **tc**, previously available as a technology preview, is now fully supported in RHEL 8.2.

(BZ#1755347)

5.7. KERNEL

Kernel version in RHEL 8.2

Red Hat Enterprise Linux 8.2 is distributed with the kernel version 4.18.0-193.

See also Important Changes to External Kernel Parameters and Device Drivers.

(BZ#1797671)

Extended Berkeley Packet Filter for RHEL 8.2

The **Extended Berkeley Packet Filter (eBPF)** is an in-kernel virtual machine that allows code execution in the kernel space, in the restricted sandbox environment with access to a limited set of functions. The virtual machine executes a special assembly-like code. The **eBPF** bytecode first loads to the kernel, followed by its verification, code translation to the native machine code with just-in-time compilation, and then the virtual machine executes the code.

Red Hat ships numerous components that utilize the **eBPF** virtual machine. Each component is in a different development phase, and thus not all components are currently fully supported. In RHEL 8.2, the following **eBPF** components are supported:

- The **BPF Compiler Collection (BCC)** tools package, which is a userspace collection of dynamic kernel tracing utilities that use the **eBPF** virtual machine for creating efficient kernel tracing and manipulation programs. The **BCC** provides tools for I/O analysis, networking, and monitoring of Linux operating systems using **eBPF**.
- The **BCC** library which allows the development of tools similar to those provided in the **BCC** tools package.
- The **eBPF for Traffic Control (tc)** feature, which enables programmable packet processing inside the kernel network data path.

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

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

- The **bpftrace** tracing language
- The **eXpress Data Path (XDP)** feature

For more information regarding the Technology Preview components, see Technology Previews.

(BZ#1780124)

Control Group v2 is now fully supported in RHEL 8

Control Group v2 mechanism is a unified hierarchy control group. **Control Group v2** organizes processes hierarchically and distributes system resources along the hierarchy in a controlled and configurable manner.

Unlike the previous version, **Control Group v2** has only a single hierarchy. This single hierarchy enables the Linux kernel to:

- Categorize processes based on the role of their owner.
- Eliminate issues with conflicting policies of multiple hierarchies.

Control Group v2 supports numerous controllers. Some of the examples are:

- CPU controller regulates the distribution of CPU cycles. This controller implements:
  - Weight and absolute bandwidth limit models for normal scheduling policy.
  - Absolute bandwidth allocation model for real-time scheduling policy.
- Cpuset controller confines processor and/or memory placement of processes to only those of the mentioned resources that are specified in the **cpuset** interface files.
- Memory controller regulates the memory distribution. Currently, the following types of memory usages are tracked:
  - Userland memory - page cache and anonymous memory.
  - Kernel data structures such as dentries and inodes.
  - TCP socket buffers.
- I/O controller regulates the distribution of I/O resources.
- Writeback controller interacts with both Memory and I/O controllers and is **Control Group v2** specific.

The information above was based on Control Group v2 upstream documentation. You can refer to the same link to obtain more information about particular **Control Group v2** controllers.

Be warned that not all features mentioned in the upstream document are implemented yet in RHEL 8.

(BZ#1401552)

Randomizing free lists: Improved performance and utilization of direct-mapped memory-side-cache

With this enhancement, you can enable page allocator to randomize free lists and improve the average utilization of a direct-mapped memory-side-cache. The kernel command-line option **page_alloc.shuffle**, enables the page allocator to randomize the free lists and sets the boolean flag to **True**. The **sysfs** file, which is located at **/sys/module/page_alloc/parameters/shuffle** reads the flag status, shuffles the free lists, such that the Dynamic Random Access Memory (DRAM) is cached, and the latency band between the DRAM and persistent memory is reduced. As a result, persistent memory with a higher capacity and lower bandwidth is available on general purpose server platforms.

(BZ#1620349)

The TPM userspace tool has been updated to the last version

The **tpm2-tools** userspace tool has been updated to version 3.2.1. This update provides several bug fixes, in particular relating to Platform Configuration Register code and manual page clean ups.

(BZ#1725714)

The C620-series PCH chipset now supports the Intel Trace Hub feature

This update adds hardware support for Intel Trace Hub (TH) in C620-series Platform Controller Hub (PCH), also known as Lewisburg PCH. Users with C620-series PCH can now use Intel TH.

(BZ#1714486)

The perf tool now supports per die events aggregation for CLX-AP and CPX processors

With this update, the **perf** tool now provides support for per-die event counts aggregation for some Intel CPUs with multiple dies. To enable this mode, add the **--per-die** option in addition to the **-a** option for Xeon Cascade Lake-AP (CLX-AP) and Cooper Lake (CPX) system processors. As a result, this update detects any imbalance between the dies. The **perf stat** command captures the event counts and displays the output as:

```
# perf stat -e cycles --per-die -a -- sleep 1
 Performance counter stats for 'system wide':
S0-D0        8      21,029,877      cycles
S0-D1        8      19,192,372      cycles
```

(BZ#1660368)

The threshold of crashkernel=auto is decreased on IBM Z

The lower threshold of the **crashkernel=auto** kernel command-line parameter is now decreased from 4G to 1G on IBM Z systems. This implementation allows the IBM Z to align with the threshold of the AMD64 and Intel 64 systems to share the same reservation policy on the lower threshold of **crashkernel=auto**. As a result, the crash kernel is able to automatically reserve memory for **kdump** on systems with less than 4GB RAM.

(BZ#1780432)

The numactl manual entry clarifies the memory usage output

With this release of RHEL 8, the manual page for **numactl** explicitly mentions that the memory usage information reflects only the resident pages on the system. The reason for this addition is to eliminate potential confusion for users whether the memory usage information relates to resident pages or virtual memory.

(BZ#1730738)

The kexec-tools document is now updated to include Kdump FCoE target support

In this release, the **/usr/share/doc/kexec-tools/supported-kdump-targets.txt** file has been updated to include Kdump Fibre Channel over Ethernet (FCoE) target support. As a result, users can now have better understanding of the status and details of the **kdump** crash dumping mechanism on a FCoE target support.

(BZ#1690729)

Firmware-assisted dump now supports PowerNV

Firmware-assisted dump (**fadump**) mechanism is now supported on the PowerNV platform. The feature is supported with the IBM POWER9 FW941 firmware version and later. At the time of system failure, **fadump**, along with the **vmcore** file, also exports the **opalcore** file. The **opalcore** file contains information about the state of OpenPOWER Abstraction Layer (OPAL) memory at the time of breakdown. The **opalcore** file is helpful in debugging crashes of OPAL-based systems.

(BZ#1524687)

kernel-rt source tree now matches the latest RHEL 8 tree

The **kernel-rt** sources have been updated to use the latest RHEL kernel source tree. The realtime patch set has also been updated to the latest upstream v5.2.21-rt13 version. Both of these updates provide a number of bug fixes and enhancements.

(BZ#1680161)

rngd is now able to run with non-root privileges

The random number generator daemon (**rngd**) checks whether data supplied by the source of randomness is sufficiently random and then stores the data in the kernel's random-number entropy pool. With this update, **rngd** is able to run with non-root user privileges to enhance system security.

(BZ#1692435)

Virtual Persistent Memory now supported for RHEL 8.2 and later on POWER 9

When running a RHEL 8.2 or later host with a PowerVM hypervisor on IBM POWER9 hardware, the host can now use the Virtual Persistent Memory (vPMEM) feature. With vPMEM, data persists across application and partition restarts until the physical server is turned off. As a result, restarting workloads that use vPMEM is significantly faster.

The following requirements must be met for your system to be able to use vPMEM:

- Hardware Management Console (HMC) V9R1 M940 or later
- Firmware level FW940 or later
- E980 system firmware FW940 or later
- L922 system firmware FW940 or later
- PowerVM level V3.1.1

Note that several known issues currently occur in RHEL 8 with vPMEM. For details, see the following Knowledgebase articles:

- Hot plug/unplug of pmem memory can cause kernel panic on POWER9
- Booting of the capture kernel takes a very long time using vPMEM namespaces as a dump target for kdump/fadump

(BZ#1859262)

5.8. FILE SYSTEMS AND STORAGE

LVM now supports the dm-writecache caching method

LVM cache volumes now provide the **dm-writecache** caching method in addition to the existing **dm-cache** method.

dm-cache

This method speeds up access to frequently used data by caching it on the faster volume. The method caches both read and write operations.

dm-writecache

This method caches only write operations. The faster volume, usually an SSD or a persistent memory (PMEM) disk, stores the write operations first and then migrates them to the slower disk in the background.

To configure the caching method, use the **--type cache** or **--type writecache** option with the **lvconvert** utility.

For more information, see Enabling caching to improve logical volume performance .

(BZ#1600174)

VDO async policy is now ACID compliant

With this release, the VDO **async** write mode is now compliant with Atomicity, Consistency, Isolation, Durability (ACID). If the system unexpectedly halts while VDO is writing data in **async** mode, the recovered data is now always consistent.

Due to the ACID compliance, the performance of **async** is now lower compared to the previous release. To restore the original performance, you can change the write mode on your VDO volume to **async-unsafe** mode, which is not ACID compliant.

For more information, see Selecting a VDO write mode.

(BZ#1657301)

You can now import VDO volumes

The **vdo** utility now enables you to import existing VDO volumes that are currently not registered on your system. To import a VDO volume, use the **vdo import** command.

Additionally, you can modify the Universally Unique Identifier (UUID) of a VDO volume using the **vdo import** command.

(BZ#1713749)

New per-op error counter is now available in the output of the mountstats and nfsiostat

A minor supportability feature is available for the NFS client systems: the output of the **mountstats** and **nfsiostat** commands in **nfs-utils** have a **per-op** error count. This enhancement allows these tools to

display **per-op** error counts and percentages that can assist in narrowing down problems on specific NFS mount points on an NFS client machine. Note that these new statistics depend on kernel changes that are inside the Red Hat Enterprise Linux 8.2 kernel.

(BZ#1719983)

Writeback IOs with cgroup awareness is now available in XFS

With this release, XFS supports writeback IOs with **cgroup** awareness. In general, **cgroup** writeback requires explicit support from the underlying file system. Until now, writeback IOs on XFS was the attribute for the root **cgroup** only.

(BZ#1274406)

The FUSE file systems now implement copy_file_range()

The **copy_file_range()** system call provides a way for file systems to implement efficient data copy mechanism. With this update, GlusterFS, which is using the Filesystem in Userspace (FUSE) framework takes advantage of this mechanism. Since read/write functionality of FUSE file systems involves multiple copies of data, using **copy_file_range()** can significantly improve performance.

(BZ#1650518)

Support for per-op statistics is now available for the mountstats and nfsiostat commands

A support feature is now available for the NFS client systems: the **/proc/self/mountstats** file has the **per-op** error counter. With this update, under each **per-op** statistics row, the ninth number indicates the number of the operations that have been completed with a status value less then zero. This status value indicates an error. For more information, see the updates to the **mountstats** and **nfsiostat** programs in the **nfs-utils** that displays these new error counts.

(BZ#1636572)

New mount stats lease_time and lease_expired are available in /proc/self/mountstats file

A support feature is available for NFSv4.x client systems. The **/proc/self/mountstats** file has the **lease_time** and the **lease_expired** fields at the end of the line starting with **nfsv4:**. The **lease_time** field indicates the number of seconds in the NFSv4 lease time. The **lease_expired** field indicates the number of seconds since the lease has expired, or 0 if the lease has not expired.

(BZ#1727369)

5.9. HIGH AVAILABILITY AND CLUSTERS

New command options to disable a resource only if this would not affect other resources

It is sometimes necessary to disable resources only if this would not have an effect on other resources. Ensuring that this would be the case can be impossible to do by hand when complex resource relations are set up. To address this need, the **pcs resource disable** command now supports the following options:

- **pcs resource disable --simulate**: show effects of disabling specified resource(s) while not changing the cluster configuration
- **pcs resource disable --safe**: disable specified resource(s) only if no other resources would be affected in any way, such as being migrated from one node to another

- **pcs resource disable --safe --no-strict**: disable specified resource(s) only if no other resources would be stopped or demoted

In addition, the **pcs resource safe-disable** command has been introduced as an alias for **pcs resource disable --safe**.

(BZ#1631519)

New command to show relations between resources

The new **pcs resource relations** command allows you to display the relations between cluster resources in a tree structure.

(BZ#1631514)

New command to display the status of both a primary site and recovery site cluster

If you have configured a cluster to use as a recovery site, you can now configure that cluster as a recovery site cluster with the **pcs dr** command. You can then use the **pcs dr** command to display the status of both your primary site cluster and your recovery site cluster from a single node.

(BZ#1676431)

Expired resource constraints are now hidden by default when listing constraints

Listing resource constraints no longer by default displays expired constraints. To include expired constaints, use the **--all** option of the **pcs constraint** command. This will list expired constraints, noting the constraints and their associated rules as **(expired)** in the display.

(BZ#1442116)

Pacemaker support for configuring resources to remain stopped on clean node shutdown

When a cluster node shuts down, Pacemaker's default response is to stop all resources running on that node and recover them elsewhere. Some users prefer to have high availability only for failures, and to treat clean shutdowns as scheduled outages. To address this, Pacemaker now supports the **shutdown-lock** and **shutdown-lock-limit** cluster properties to specify that resources active on a node when it shuts down should remain stopped until the node next rejoins. Users can now use clean shutdowns as scheduled outages without any manual intervention. For information on configuring resources to remain stopped on a clean node shutdown, see link: Configuring resources to remain stopped on clean node shutdown.

(BZ#1712584)

Support for running the cluster environment in a single node

A cluster with only one member configured is now able to start and run resources in a cluster environment. This allows a user to configure a separate disaster recovery site for a multi-node cluster that uses a single node for backup. Note that a cluster with only one node is not in itself fault tolerant.

(BZ#1700104)

5.10. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

A new module: python38

RHEL 8.2 introduces Python 3.8, provided by the new module **python38** and the **ubi8/python-38** container image.

Notable enhancements compared to Python 3.6 include:

- New Python modules, for example, **contextvars**, **dataclasses**, or **importlib.resources**
- New language features, such as assignment expressions (the so-called walrus operator, **:=**) or positional-only parameters
- Improved developer experience with the **breakpoint()** built-in function, the **=** format string specification, and compatibility between debug and non-debug builds of Python and extension modules
- Performance improvements
- Improved support for optional static type hints
- An addition of the **=** specifier to formatted string literals (f-strings) for easier debugging
- Updated versions of packages, such as **pip**, **requests**, or **Cython**

Python 3.8 and packages built for it can be installed in parallel with Python 3.6 on the same system.

Note that the **python38** module does not include the same binary bindings to system tools (RPM, DNF, SELinux, and others) that are provided for the **python36** module.

To install packages from the **python38** module, use, for example:

```
# yum install python38
# yum install python38-Cython
```

The **python38:3.8** module stream will be enabled automatically.

To run the interpreter, use, for example:

```
$ python3.8
$ python3.8 -m cython --help
```

See Using Python for more information.

Note that Red Hat will continue to provide support for Python 3.6 until the end of life of RHEL 8. Python 3.8 will have a shorter life cycle, see RHEL 8 Application Streams Life Cycle .

(BZ#1747329)

Changes in mod_wsgi installation

Previously, when the user tried to install the **mod_wsgi** module using the **yum install mod_wsgi** command, the **python3-mod_wsgi** package was always installed. RHEL 8.2 introduces Python 3.8 as an addition to Python 3.6. With this update, you need to specify which version of **mod_wsgi** you want to install, otherwise an error message is returned.

To install the Python 3.6 version of **mod_wsgi**:

```
# yum install python3-mod_wsgi
```

To install the Python 3.8 version of **mod_wsgi**:

```
# yum install python38-mod_wsgi
```

Note that the **python3-mod_wsgi** and **python38-mod_wsgi** packages conflict with each other, and only one **mod_wsgi** module can be installed on a system due to a limitation of the Apache HTTP Server.

This change introduced a dependency known issue described in BZ#1829692.

(BZ#1779705)

Support for hardware-accelerated deflate in zlib on IBM Z

This update adds support for a hardware-accelerated deflate algorithm to the **zlib** library in the IBM Z mainframes. As a result, performance of compression and decompression on IBM Z vector machines has been improved.

(BZ#1659433)

Performance improved when decompressing gzip on IBM Power Systems, little endian

This update adds optimization for the 32-bit Cyclic Redundancy Check (CRC32) to the **zlib** library on IBM Power Systems, little endian. As a result, performance of decompressing **gzip** files has been improved.

(BZ#1666798)

A new module stream: maven:3.6

RHEL 8.2 introduces a new module stream, **maven:3.6**. This version of the Maven software project management and comprehension tool provides numerous bug fixes and various enhancements over the **maven:3.5** stream distributed with RHEL 8.0.

To install the **maven:3.6** stream, use:

```
# yum module install maven:3.6
```

If you want to upgrade from the **maven:3.5** stream, see Switching to a later stream .

(BZ#1783926)

mod_md now supports the ACMEv2 protocol

The **mod_md** module has been updated to version 2.0.8. This update adds a number of features, notably support for version 2 of the Automatic Certificate Management Environment (ACME) certificate issuance and management protocol, which is the Internet Engineering Task Force (IETF) standard (RFC 8555). The original ACMEv1 protocol remains supported but is deprecated by popular service providers.

(BZ#1747923)

New extensions for PHP 7.3

The **php:7.3** module stream has been updated to provide two new PHP extensions: **rrd** and **Xdebug**.

The **rrd** extension provides bindings to the **RRDtool** C library. **RRDtool** is a high performance data logging and graphing system for time series data.

The **Xdebug** extension is included to assist you with debugging and development. Note that the extension is provided only for development purposes and should not be used in production environments.

For information about installing and using PHP in RHEL 8, see Using the PHP scripting language.

(BZ#1769857, BZ#1764738)

New packages: perl-LDAP and perl-Convert-ASN1

This update adds the **perl-LDAP** and **Perl-Convert-ASN1** packages to RHEL 8. The **perl-LDAP** package provides an LDAP client for the Perl language. **perl-LDAP** requires the **perl-Convert-ASN1** package, which encodes and decodes Abstract Syntax Notation One (ASN.1) data structures using Basic Encoding Rules (BER) and Distinguished Encoding Rules (DER).

(BZ#1663063, BZ#1746898)

sscg now supports generating private key files protected by a password

The **sscg** utility is now able to generate private key files protected by a password. This adds another level of protection for private keys, and it is required by some services, such as FreeRADIUS.

(BZ#1717880)

5.11. COMPILERS AND DEVELOPMENT TOOLS

grafana rebased to version 6.3.6

The **grafana** package has been upgraded to version 6.3.6, which provides multiple bug fixes and enhancements. Notable changes include:

- Database: Rewrites system statistics query for better performance.
- Explore:
  - Fixes query field layout in split view for the Safari browsers.
  - Adds Live option for the supported data sources, adds the **orgId** to URL for sharing purposes.
  - Adds support for the new **loki start** and **end** parameters for labels endpoint.
  - Adds support for toggling raw query mode in the Explore, allow switching between metrics and logs.
  - Displays log lines context, does not parse log levels if provided by field or label.
  - Supports new **LogQL** filtering syntax.
  - Uses new **TimePicker** from Grafana/UI.
  - Handles newlines in the **LogRow** Highlighter.
  - Fixes browsing back to the dashboard panel.
  - Fixes filter by series level in logs graph.
  - Fix issues when loading and graph/table are collapsed.

  - Fixes the selection/copy of log lines.

- Dashboard: Fixes dashboards **init** failed loading error for dashboards with panel links that had missing properties, and fixes timezone dashboard setting while exporting to the comma-separated values (CSV) Data links.

- Editor: Fixes issue where only entire lines were being copied.

- LDAP: Integration of the **multi ldap** and **ldap** authentication components.

- Profile/UserAdmin: Fixes user agent parser crashing the **grafana-server** on 32-bit builds.

- Prometheus:

  - Prevents panel editor crash while switching to the **Prometheus** data source, changes **brace-insertion** behaviour to be less annoying.

  - Fixes queries with the **label_replace** and removes the $1 match when loading the query editor.

  - Consistently allows multi-line queries in the editor, taking timezone into account for the step alignment.

  - Uses the overridden panel range for **$__range** instead of the dashboard range.

  - Adds time range filter to series labels query, escapes | literals in the interpolated **PromQL** variables.

  - Fixes while adding labels for metrics which contain colons in the Explore.

- Auth: Allows expiration of the API keys, returns device, os and browser while listing user auth tokens in HTTP API, supports list and revoke of user auth tokens in UI.

- DataLinks: Correctly applies scoped variables to the data links, follows timezone while displaying datapoint timestamp in the graph context menu, uses datapoint timestamp correctly when interpolating the variables, fixes the incorrect interpolation of the **${__series_name}**.

- Graph: Fixes legend issue clicking on series line icon and issue with horizontal scrollbar being visible on windows, adds new fill gradient option.

- Graphite: Avoids the glob of single-value array variables, fixes issues with alias function being moved last, fixes issue with the **seriesByTag** & function with variable parameter, uses **POST** for /**metrics/find** requests.

- TimeSeries: Assumes values are all numbers.

- Gauge/BarGauge: Fixes issue with lost thresholds and an issue loading Gauge with the **avg** stat.

- PanelLinks: Fixes crash issue with Gauge & Bar Gauge panels with panel links (drill down links), fixes render issue while there is no panel description.

- OAuth: Fixes the **missing saved state** OAuth login failure due to SameSite cookie policy, fixes for wrong user token updated on the **OAuth** refresh in DS proxy.

- Auth Proxy: Includes additional headers as a part of the cache key.

- **cli**: Fix for recognizing when in dev mode, fixes the issue of **encrypt-datasource-passwords** failing with the sql error.

- Permissions: Show plugins in the navigation for non admin users but hides plugin configuration.
- TimePicker: Increases max height of quick range dropdown and fixes style issue for custom range popover.
- Loki: Displays live tailed logs in correct order in the Explore.
- Timerange: Fixes a bug where custom time ranges were not following the Universal Time Coordinated (UTC).
- **remote_cache**: Fixes the **redis connstr** parsing.
- Alerting: Add tags to alert rules, attempts to send email notifications to all the given email addresses, improves alert rule testing, support for configuring the content field for the **Discord** alert notifier.
- Alertmanager: Replaces illegal characters with underscore in the label names.
- AzureMonitor: Changes clashing built-in Grafana variables or macro names for the Azure Logs.
- CloudWatch: Made region visible for Amazon Web Services (AWS) Cloudwatch Expressions, adds the AWS **DocDB** metrics.
- GraphPanel: Do not sort series when legend table and sort column is not visible.
- InfluxDB: Supports visualizing logs in the Explore.
- MySQL/Postgres/MSSQL: Adds parsing for day, weeks, and year intervals in macros, adds support for periodically reloading client certs.
- Plugins: Replaces the **dataFormats** list with the **skipDataQuery** flag in the **plugin.json** file.
- Refresh picker: Handles empty intervals.
- Singlestat: Add **y** min/max configuration to the singlestat sparklines.
- Templating: Correctly displays the **__text** in the multi-value variable after page reloads, supports selecting all the filtered values of a multi-value variable.
- Frontend: Fixes Json tree component not working issue.
- InfluxDB: Fixes issues with single quotes not escaped in the label value filters.
- Config: Fixes the **connectionstring** option for the **remote_cache** in the **defaults.ini** file.
- Elasticsearch: Fixes the empty query (via template variable) should be sent as wildcard, fixes the default max concurrent shard requests, supports visualizing logs in the Explore.
- TablePanel: Fixes the annotations display.
- Grafana-CLI: Fixes receiving flags via command line, wrapper for the **grafana-cli** within the **RPM**/**DEB** packages and **config**/**homepath** are now global flags.
- HTTPServer: Fixes the **X-XSS-Protection** header formatting, options for returning new headers **X-Content-Type-Options**, **X-XSS-Protection** and **Strict-Transport-Security**, fixes the **Strict-Transport-Security** header, serves Grafana with a custom URL path prefix.

(BZ#1725278)

pcp rebased to version 5.0.2

The **pcp** package has been upgraded to version 5.0.2, which provides multiple bug fixes and enhancements. Notable changes include:

- The **pcp-webapp-*** packages are now replaced by the **grafana-pcp** package and **pmproxy**.
- The **pcp-collectl** tool is now replaced by the **pmrep** configurations.
- New and improved performance metric domain agents (PMDAs):
  - **pmdamssql**: New PMDA for Microsoft SQL Server implementation.
  - **pmdanetcheck**: New PMDA to perform network checks.
  - **pmdaopenmetrics**: Renames **prometheus** agent to **openmetrics**.
  - **pmdanfsclient**: Adds the **per-op** and **per-mount rpc** error metrics.
  - **pmdalmsensors**: Improvements in the name parsing and error handling.
  - **pmdaperfevent**: Supports **hv_24x7** nest events on the multi-node system.
  - **pmdalinux**:
    - Correctly handles sparse or discontinuous numa nodes.
    - Uses cpu **instname** and not the **instid** for **per-cpu** numa stats.
    - Adds an active and total slabs to **slabinfo** v2 parsing
    - Fixes several unix socket, **icmp6** metrics, hugepage metric value. calculations, **segfault** in interrupts code with large CPU counts
    - Fetches more network metrics in the **--container** namespace.
  - **pmdabcc**: Fixes the tracepoints module for the **bcc** 0.10.0 and higher versions
  - **pmdabpftrace**: New PMDA for metrics from the **bpftrace** scripts
  - **pmdaproc**:
    - Fixes memory leak in the **pidlist** refresh.
    - Avoids excessive stat calls in **cgroups_scan**.
    - Retains **cgroup** paths and only un-escape instance names.
  - **pmdaroot**: Improves handling of cached or inactive the **cgroup** behaviour and refreshes the container **indom** on **cgroup** fs change as well.
- Fixes to collector (server) tools:
  - **pmproxy**: Openmetrics support via the /**metrics** endpoint, consolidates the **pmseries**/**grafana** REST API, and adds new async **PMWEBAPI(3)** REST API implementation.
  - **selinux**: Numerous pcp policy updates.

    - python **pmdas**: Enables authentication support, new **set_comm_flags** method to set the communication flags.
    - **python api**: Exports the **pmdaGetContext()** and adds debugging wrapper.
    - **perl api**: Ensures context set up for PMDA store as with python wrapper.
    - **systemd**: Adds 120s timeout in all the services and fixes failure to start the **pmlogger** service.
- Fixes to analysis (client) tools:
    - **pmchart**: Fixes chart auto-scaling under fetch error conditions.
    - **pmrep**: Fixes the **wait.formula** for **collectl-dm-sD** and **collectl-sD**.
    - **pmseries**: Provides support for the delta keyword and better timestamps.
    - **pcp-atop**: Fixes the write mode (**-w**) to handle the **proc** vs **hotproc** metrics.
    - **pcp-atopsar**: Fixes the mishandling of a few command line arguments.
    - **pcp-dstat**: Fixes misaligned headers in CSV output and handling of the **--bits** command line option.
    - **libpcp**: Fixes the **cockpit-pcp segv** with local context and multi-archive replay error handling for the corrupted archive(s).

(BZ#1723598)

grafana-pcp is now available in RHEL 8.2

The **grafana-pcp** package provides new **grafana** data sources and application plugins connecting **PCP** with **grafana**. With the **grafana-pcp** package, you can analyze historical **PCP** metrics and real-time **PCP** metrics using the **pmseries** query language and **pmwebapi** live services respectively. For more information, see Performance Co-Pilot Grafana Plugin.

(BZ#1685315)

Updated GCC Toolset 9

GCC Toolset 9 is a compiler toolset that provides recent versions of development tools. It is available as an Application Stream in the form of a Software Collection in the **AppStream** repository.

Notable changes introduced with RHEL 8.2 include:

- The GCC compiler has been updated to version 9.2.1, which provides many bug fixes and enhancements that are available in upstream GCC.
- The GCC Toolset 9 components are now available in the two container images:
    - **rhel8/gcc-toolset-9-toolchain**, which includes the GCC compiler, the GDB debugger, and the **make** automation tool.
    - **rhel8/gcc-toolset-9-perftools**, which includes the performance monitoring tools, such as SystemTap and Valgrind.
      To pull a container image, run the following command as root:

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

The following tools and versions are provided by GCC Toolset 9:

Tool	Version
GCC	9.2.1
GDB	8.3
Valgrind	3.15.0
SystemTap	4.1
Dyninst	10.1.0
binutils	2.32
elfutils	0.176
dwz	0.12
make	4.2.1
strace	5.1
ltrace	0.7.91
annobin	9.08

To install GCC Toolset 9, run the following command as root:

```
# yum install gcc-toolset-9
```

To run a tool from GCC Toolset 9:

```
$ scl enable gcc-toolset-9 tool
```

To run a shell session where tool versions from GCC Toolset 9 take precedence over system versions of these tools:

```
$ scl enable gcc-toolset-9 bash
```

For more information, see Using GCC Toolset.

(BZ#1789401)

GCC Toolset 9 now supports NVIDIA PTX target offloading

The GCC compiler in GCC Toolset 9 now supports OpenMP target offloading for NVIDIA PTX.

(BZ#1698607)

The updated GCC compiler is now available for RHEL 8.2

The system GCC compiler, version 8.3.1, has been updated to include numerous bug fixes and enhancements available in the upstream GCC.

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

For usage information, see Developing C and C++ applications in RHEL 8 .

(BZ#1747157)

A new tunable for changing the maximum fastbin size in glibc

The **malloc** function uses a series of fastbins that hold reusable memory chunks up to a specific size. The default maximum chunk size is 80 bytes on 32-bit systems and 160 bytes on 64-bit systems. This enhancement introduces a new **glibc.malloc.mxfast** tunable to **glibc** that enables you to change the maximum fastbin size.

(BZ#1764218)

Vectorized math library is now enabled for GNU Fortran in GCC Toolset 9

With this enhancement, GNU Fortran from GCC Toolset can now use routines from the vectorized math library **libmvec**. Previously, the Fortran compiler in GCC Toolset needed a Fortran header file before it could use routines from **libmvec** provided by the GNU C Library **glibc**.

(BZ#1764238)

The glibc.malloc.tcache tunable has been enhanced

The **glibc.malloc.tcache_count** tunable allows to set the maximum number of memory chunks of each size that can be stored in the per-thread cache (tcache). With this update, the upper limit of the **glibc.malloc.tcache_count** tunable has been increased from 127 to 65535.

(BZ#1746933)

The glibc dynamic loader is enhanced to provide a non-inheriting library preloading mechanism

With this enhancement, the loader can now be invoked to load a user program with a **--preload** option followed by a colon-separated list of libraries to preload. This feature allows users to invoke their programs directly through the loader with a non-inheriting library preload list.

Previously, users had to use the LD_PRELOAD environment variable which was inherited by all child processes through their environment.

(BZ#1747453)

GDB now supports the ARCH(13) extension on the IBM Z architecture

With this enhancement, the GNU Debugger (GDB) now supports the new instructions implemented by the ARCH(13) extension on the IBM Z architecture.

(BZ#1768593)

elfutils rebased to version 0.178

The **elfutils** package has been upgraded to version 0.178, which provides multiple bug fixes and enhancements. Notable changes include:

- **elfclassify**: a new tool to analyze ELF objects.
- **debuginfod**: a new server, client tool, and library to index and automatically fetch ELF, DWARF, and source from files and RPM archives through HTTP.
- **libebl** is now directly compiled into **libdw.so**.
- **eu-readelf** has multiple new flags for notes, section numbering, and symbol tables.
- **libdw** has improved multithreading support.
- **libdw** supports additional GNU DWARF extensions.

(BZ#1744992)

SystemTap rebased to version 4.2

The SystemTap instrumentation tool has been updated to version 4.2. Notable enhancements include:

- Backtraces can now include source file names and line numbers.
- Numerous Berkeley Packet Filter (BPF) back-end extensions are now available, for example, for looping, timing, and other processes.
- A new service for managing SystemTap scripts is available. This service sends metrics to a Prometheus-compatible monitoring system.
- SystemTap has inherited functionality of a new HTTP file server for **elfutils** called **debuginfod**. This server automatically sends debugging resources to SystemTap.

(BZ#1744989)

Enhancements to IBM Z series performance counters

IBM Z series type 0x8561, 0x8562, and 0x3907 (z14 ZR1) machines are now recognized by **libpfm**. Performance events for monitoring elliptic-curve cryptography (ECC) operations on IBM Z series are now available. This allows monitoring of additional subsystems on IBM Z series machines.

(BZ#1731019)

Rust Toolset rebased to version 1.41

Rust Toolset has been updated to version 1.41. Notable changes include:

- Implementing new traits is now easier because the orphan rule is less strict.
- You can now attach the **#[non_exhaustive]** attribute to a **struct**, an **enum**, or **enum** variants.
- Using **Box<T>** in the Foreign Function Interface (FFI) has more guarantees now. **Box<T>** will have the same Application Binary Interface (ABI) as a **T*** pointer in the FFI.
- Rust is supposed to detect memory-safety bugs at compile time, but the previous borrow checker had limitations and allowed undefined behaviour and memory unsafety. The new non-lexical lifetimes (NLL) borrow checker can report memory unsafety problems as hard errors. It

now applies to the Rust 2015 and Rust 2018 editions. Previously, in Rust 2015 the NLL borrow checker only raised warnings about such problems.

To install the **rust-toolset** module, run the following command as root:

```
# yum module install rust-toolset
```

For usage information, see Using Rust Toolset.

(BZ#1776847)

LLVM Toolset rebased to version 9.0.1

LLVM Toolset has been upgraded to version 9.0.1. With this update, the **asm goto** statements are now supported. This change allows to compile the Linux kernel on the AMD64 and Intel 64 architectures.

To install the **llvm-toolset** module, run the following command as root:

```
# yum module install llvm-toolset
```

For more information, see Using LLVM Toolset.

(BZ#1747139)

Go Toolset rebased to version 1.13

Go Toolset has been upgraded to version 1.13. Notable enhancements include:

- Go can now use a FIPS-certified cryptographic module when the RHEL system is booted in the FIPS mode. Users can enable this mode manually using the **GOLANG_FIPS=1** environment variable.
- The Delve debugger, version 1.3.2, is now available for Go. It is a source-level debugger for the Go (**golang**) programming language.

To install the **go-toolset** module, run the following command as root:

```
# yum module install go-toolset
```

To install the Delve debugger, run the following command as root:

```
# yum install delve
```

To debug a **helloworld.go** program using Delve, run the following command:

```
$ dlv debug helloworld.go
```

For more information on Go Toolset, see Using Go Toolset.

For more information on Delve, see the upstream Delve documentation.

(BZ#1747150)

OpenJDK now supports also secp256k1

Previously, Open Java Development Kit (OpenJDK) could use only curves from the NSS library.

Consequently, OpenJDK provided only the secp256r1, secp384r1, and secp521r1 curves for elliptic curve cryptography (ECC). With this update, OpenJDK uses the internal ECC implementation and supports also the secp256k1 curve.

(BZ#1746875, BZ#1746879)

5.12. IDENTITY MANAGEMENT

IdM now supports new Ansible management modules

This update introduces several **ansible-freeipa** modules for automating common Identity Management (IdM) tasks using Ansible playbooks:

- The **ipauser** module automates adding and removing users.
- The **ipagroup** module automates adding and removing users and user groups to and from user groups.
- The **ipahost** module automates adding and removing hosts.
- The **ipahostgroup** module automates adding and removing hosts and host groups to and from host groups.
- The **ipasudorule** module automates the management of **sudo** command and **sudo** rule.
- The **ipapwpolicy** module automates the configuration of password policies in IdM.
- The **ipahbacrule** module automates the management of host-based access control in IdM.

Note that you can combine two or more **ipauser** calls into one with the **users** variable or, alternatively, use a JSON file containing the users. Similarly, you can combine two or more **ipahost** calls into one with the **hosts** variable or, alternatively, use a JSON file containing the hosts. The **ipahost** module can also ensure the presence or absence of several IPv4 and IPv6 addresses for a host.

(JIRA:RHELPLAN-37713)

IdM Healthcheck now supports screening DNS records

This update introduces a standalone manual test of DNS records on an Identity Management (IdM) server.

The test uses the **Healthcheck** tool and performs a DNS query using the local resolver in the **etc/resolv.conf** file. The test ensures that the expected DNS records required for autodiscovery are resolvable.

(JIRA:RHELPLAN-37777)

Direct integration of RHEL into AD using SSSD now supports FIPS

With this enhancement, the System Services Security Daemon (SSSD) now integrates with Active Directory (AD) deployments whose authentication mechanisms use encryption types that were approved by the Federal Information Processing Standard (FIPS). The enhancement enables you to directly integrate RHEL systems into AD in environments that must meet the FIPS criteria.

(BZ#1841170)

The SMB1 protocol has been disabled in the Samba server and client utilities by default

In Samba 4.11, the default values of the **server min protocol** and **client min protocol** parameters have been changed from **NT1** to **SMB2_02** because the server message block version 1 (SMB1) protocol is deprecated. If you have not set these parameters in the **/etc/samba/smb.conf** file:

- Clients that only support SMB1 are no longer able to connect to the Samba server.
- Samba client utilities, such as **smbclient**, and the **libsmbclient** library fail to connect to servers that only support SMB1.

Red Hat recommends to not use the SMB1 protocol. However, if your environment requires SMB1, you can manually re-enable the protocol.

To re-enable SMB1 on a Samba server:

- Add the following setting to the **/etc/samba/smb.conf** file:

```
server min protocol = NT1
```

- Restart the **smb** service:

```
# systemctl restart smb
```

To re-enable SMB1 for Samba client utilities and the **libsmbclient** library:

- Add the following setting to the **/etc/samba/smb.conf** file:

```
client min protocol = NT1
```

- Restart the **smb** service:

```
# systemctl restart smb
```

Note that the SMB1 protocol will be removed in a future Samba release.

(BZ#1785248)

samba rebased to version 4.11.2

The *samba* packages have been upgraded to upstream version 4.11.2, which provides a number of bug fixes and enhancements over the previous version. Notable changes include:

- By default, the server message block version 1 (SMB1) protocol is now disabled in the Samba server, client utilities, and the **libsmbclient** library. However, you can still set the **server min protocol** and **client min protocol** parameters manually to **NT1** to re-enable SMB1. Red Hat does not recommend to re-enabling the SMB1 protocol.
- The **lanman auth** and **encrypt passwords** parameters are deprecated. These parameters enable insecure authentication and are only available in the deprecated SMB1 protocol.
- The **-o** parameter has been removed from the **onode** clustered trivial database (CTDB) utility.
- Samba now uses the GnuTLS library for encryption. As a result, if the FIPS mode in RHEL is enabled, Samba is compliant with the FIPS standard.
- The **ctdbd** service now logs when it uses more than 90% of a CPU thread.

- The deprecated Python 2 support has been removed.

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

For further information about notable changes, read the upstream release notes before updating: https://www.samba.org/samba/history/samba-4.11.0.html

(BZ#1754409)

Directory Server rebased to version 1.4.2.4

The *389-ds-base* packages have been upgraded to upstream version 1.4.2.4, which provides a number of bug fixes and enhancements over the previous version. For a complete list of notable changes, read the upstream release notes before updating:

- https://directory.fedoraproject.org/docs/389ds/releases/release-1-4-2-4.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-1-4-2-3.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-1-4-2-2.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-1-4-2-1.html

(BZ#1748994)

Certain legacy scripts have been replaced in Directory Server

This enhancement provides replacements for the unsupported **dbverify**, **validate-syntax.pl**, **cl-dump.pl**, **fixup-memberuid.pl**, and **repl-monitor.pl** legacy scripts in Directory Server. These scripts have been replaced with the following commands:

- **dbverify**: **dsctl *instance_name* dbverify**
- **validate-syntax.pl**: **dsconf schema validate-syntax**
- **cl-dump.pl**: **dsconf replication dump-changelog**
- **fixup-memberuid.pl**: **dsconf plugin posix-winsync fixup**
- **repl-monitor.pl**: **dsconf replication monitor**

For a list of all legacy scripts and their replacements, see Command-line utilities replaced in Red Hat Directory Server 11.

(BZ#1739718)

Setting up IdM as a hidden replica is now fully supported

Identity Management (IdM) in RHEL 8.2 fully supports setting up IdM servers as hidden replicas. A hidden replica is an IdM server that has all services running and available. However, it is not advertised to other clients or masters because no **SRV** records exist for the services in DNS, and LDAP server roles are not enabled. Therefore, clients cannot use service discovery to detect hidden replicas.

Hidden replicas are primarily designed for dedicated services that can otherwise disrupt clients. For example, a full backup of IdM requires to shut down all IdM services on the master or replica. Since no clients use a hidden replica, administrators can temporarily shut down the services on this host without

affecting any clients. Other use cases include high-load operations on the IdM API or the LDAP server, such as a mass import or extensive queries.

To install a new hidden replica, use the **ipa-replica-install --hidden-replica** command. To change the state of an existing replica, use the **ipa server-state** command.

For further details, see Installing an IdM hidden replica .

(BZ#1719767)

Kerberos ticket policy now supports authentication indicators

Authentication indicators are attached to Kerberos tickets based on which pre-authentication mechanism has been used to acquire the ticket:

- **otp** for two-factor authentication (password + OTP)
- **radius** for RADIUS authentication
- **pkinit** for PKINIT, smart card or certificate authentication
- **hardened** for hardened passwords (SPAKE or FAST)

The Kerberos Distribution Center (KDC) can enforce policies such as service access control, maximum ticket lifetime, and maximum renewable age, on the service ticket requests which are based on the authentication indicators.

With this enhancement, administrators can achieve finer control over service ticket issuance by requiring specific authentication indicators from a user's tickets.

(BZ#1777564)

The krb5 package is now FIPS-compliant

With this enhancement, non-compliant cryptography is prohibited. As a result, administrators can use Kerberos in FIPS-regulated environments.

(BZ#1754690)

Directory Server sets the sslVersionMin parameter based on the system-wide crypto policy

By default, Directory Server now sets the value of the **sslVersionMin** parameter based on the system-wide crypto policy. If you set the crypto policy profile in the **/etc/crypto-policies/config** file to:

- **DEFAULT**, **FUTURE**, or **FIPS**, Directory Server sets **sslVersionMin** to **TLS1.2**
- **LEGACY**, Directory Server sets **sslVersionMin** to **TLS1.0**

Alternatively, you can manually set **sslVersionMin** to higher value than the one defined in the crypto policy:

```
# dsconf -D "cn=Directory Manager" __ldap://server.example.com__ security set --tls-protocol-min
TLS1.3
```

(BZ#1828727)

5.13. DESKTOP

Wayland is now enabled on dual-GPU systems

Previously, the GNOME environment defaulted to the **X11** session on laptops and other systems that have two graphical processing units (GPUs). With this release, GNOME now defaults to the **Wayland** session on dual-GPU systems, which is the same behavior as on single-GPU systems.

(BZ#1749960)

5.14. GRAPHICS INFRASTRUCTURES

Support for new graphics cards

The following graphics cards are now supported:

- Intel HD Graphics 610, 620, and 630, which are found with the Intel Comet Lake H and U processors
- Intel Ice Lake UHD Graphics 910 and Iris Plus Graphics 930, 940, and 950.
  You no longer need to set the **alpha_support** kernel option to enable support for Intel Ice Lake graphics.
- The AMD Navi 10 family, which includes the following models:
  - Radeon RX 5600
  - Radeon RX 5600 XT
  - Radeon RX 5700
  - Radeon RX 5700 XT
  - Radeon Pro W5700
- The Nvidia Turing TU116 family, which includes the following models.
  Note that the **nouveau** graphics driver does not yet support 3D acceleration with the Nvidia Turing TU116 family.
  - GeForce GTX 1650 Super
  - GeForce GTX 1660
  - GeForce GTX 1660 Super
  - GeForce GTX 1660 Ti
  - GeForce GTX 1660 Ti Max-Q

Additionally, the following graphics drivers have been updated:

- The Matrox **mgag2000** driver
- The Aspeed **ast** driver
- The Intel **i915** driver

(JIRA:RHELPLAN-41384)

5.15. THE WEB CONSOLE

Administrators can now use client certificates to authenticate to the RHEL 8 web console

With this web console enhancement, a system administrator can use client certificates to access a RHEL 8 system locally or remotely using a browser with certificate authentication built in. No additional client software is required. These certificates are commonly provided by a smart card or Yubikey, or can be imported into the browser.

When logging in with a certificate, the user cannot currently perform administrative actions in the web console. But the user can perform them on the Terminal page with the **sudo** command after authenticating with a password.

(JIRA:RHELPLAN-2507)

Option to log in to the web console with a TLS client certificate

With this update, it is possible to configure the web console to log in with a TLS client certificate that is provided by a browser or a device such as a smart card or a YubiKey.

(BZ#1678465)

Changes to web console login

RHEL web console has been updated with the following changes:

- The web console will automatically log you out of your current session after 15 minutes of inactivity. You can configure the timeout in minutes in the **/etc/cockpit/cockpit.conf** file.
- Similarly to SSH, the web console can now optionally show the content of banner files on the login screen. Users need to configure the functionality in the **/etc/cockpit/cockpit.conf** file.

See the **cockpit.conf(5)** manual page for more information.

(BZ#1754163)

The RHEL web console has been redesigned to use the PatternFly 4 user interface design system

The new design provides better accessibility and matches the design of OpenShift 4. Updates include:

- The Overview page has been completely redesigned. For example, information is grouped into easier-to-understand panels, health information is more prominent, resource graphs have been moved to their own page, and the hardware information page is now easier to find.
- Users can use the new Search field in the Navigation menu to easily find specific pages that are based on keywords.

For more information about PatternFly, see the PatternFly project page.

(BZ#1784455)

Virtual Machines page updates

The web console's **Virtual Machines** page got several storage improvements:

- Storage volume creation now works for all libvirt-supported types.
- Storage pools can be created on LVM or iSCSI.

Additionally, the **Virtual Machines** page now supports the creation and removal of virtual network interfaces.

(BZ#1676506, BZ#1672753)

Web console Storage page updates

Usability testing showed that the *default mount point* concept on the RHEL web console **Storage** page was hard to grasp, and led to a lot of confusion. With this update, the web console no longer offers a *Default* choice when mounting a file system. Creating a new file system now always requires a specified mount point.

Additionally, the web console now hides the distinction between the configuration (/**etc**/**fstab**) and the run-time state (/**proc**/**mounts**). Changes made in the web console always apply to both the configuration and the run-time state. When the configuration and the run-time state differ from each other, the web console shows a warning, and enable users to easily bring them back in sync.

(BZ#1784456)

5.16. VIRTUALIZATION

Attempting to create a RHEL virtual machine from an install tree now returns a more helpful error message.

RHEL 7 and RHEL 8 virtual machines created using the **virt-install** utility with the **--location** option in some cases fail to boot. This update adds a virt-install error message that provides instructions on how to work around this problem.

(BZ#1677019)

Intel Xeon Platinum 9200 series processors supported on KVM guests

Support for Intel Xeon Platinum 9200 series processors (previously known as **Cascade Lake**) has now been added to the KVM hypervisor and kernel code, and to the libvirt API. This enables KVM virtual machines to use Intel Xeon Platinum 9200 series processors.

(JIRA:RHELPLAN-13995)

EDK2 rebased to version stable201908

The *EDK2* package has been upgraded to version **stable201908**, which provides multiple enhancements. Notably:

- *EDK2* now includes support for OpenSSL-1.1.1.
- To comply with the upstream project's licensing requirements, the *EDK2* package license has been changed from **BSD and OpenSSL and MIT** to **BSD-2-Clause-Patent and OpenSSL and MIT**.

(BZ#1748180)

Creating nested virtual machines

With this update, nested virtualization is fully supported for KVM virtual machines (VMs) running on an Intel 64 host with RHEL 8. With this feature, a RHEL 7 or RHEL 8 VM that runs on a physical RHEL 8 host can act as a hypervisor, and host its own VMs.

Note that on AMD64 systems, nested KVM virtualization remains a Technology Preview.

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

5.17. CONTAINERS

The default registries search list in /etc/containers/registries.conf has been updated

The default **registries.search** list in **/etc/containers/registries.conf** has been updated to only include trusted registries that provide container images curated, patched, and maintained by Red Hat and its partners.

Red Hat recommends always using fully qualified image names including:

- The registry server (full DNS name)
- Namespace
- Image name
- Tag (for example **registry.redhat.io/ubi8/ubu:latest**)

When using short names, there is always an inherent risk of spoofing For example, a user wants to pull an image named **foobar** from a registry and expects it to come from **myregistry.com**. If **myregistry.com** is not first in the search list, an attacker could place a different **foobar** image at a registry earlier in the search list. The user would accidentally pull and run the attacker image and code rather than the intended content. Red Hat recommends only adding registries which are trusted, that is registries which do not allow unknown or anonymous users to create accounts with arbitrary names. This prevents an image from being spoofed, squatted or otherwise made insecure.

(BZ#1810053)

Podman no longer depends on oci-systemd-hook

Podman does not need or depend on the **oci-systemd-hook** package which has been removed from the **container-tools:rhel8** and **container-tools:2.0** module streams.

(BZ#1645280)

CHAPTER 6. IMPORTANT CHANGES TO EXTERNAL KERNEL PARAMETERS

This chapter provides system administrators with a summary of significant changes in the kernel distributed with Red Hat Enterprise Linux 8.2. These changes include added or updated **proc** entries, **sysctl**, and **sysfs** default values, boot parameters, kernel configuration options, or any noticeable behavior changes.

6.1. NEW KERNEL PARAMETERS

cpuidle.governor = [CPU_IDLE]

Name of the **cpuidle** governor to use.

deferred_probe_timeout = [KNL]

This is a debugging parameter for setting a timeout in seconds for the deferred probe to give up waiting on dependencies to probe.
Only specific dependencies (subsystems or drivers) that have opted in will be ignored. A timeout of 0 will timeout at the end of **initcalls**. This parameter will also dump out devices still on the deferred probe list after retrying.

kvm.nx_huge_pages = [KVM]

This parameter controls the software workaround for the **X86_BUG_ITLB_MULTIHIT** bug.
The options are:

- **force** - Always deploy workaround.
- **off** - Never deploy workaround.
- **auto** (default) - Deploy workaround based on the presence of **X86_BUG_ITLB_MULTIHIT**.

If the software workaround is enabled for the host, guests do not need to enable it for nested guests.

kvm.nx_huge_pages_recovery_ratio = [KVM]

This parameter controls how many 4KiB pages are periodically zapped back to huge pages. 0 disables the recovery, otherwise if the value is N, Kernel-based Virtual Machine (KVM) will zap 1/Nth of the 4KiB pages every minute. The default is 60.

page_alloc.shuffle = [KNL]

Boolean flag to control whether the page allocator should randomize its free lists.
The randomization may be automatically enabled if the kernel detects it is running on a platform with a direct-mapped memory-side cache. This parameter can be used to override/disable that behavior.

The state of the flag can be read from the **sysfs** pseudo filesystem from the **/sys/module/page_alloc/parameters/shuffle** file.

panic_print =

Bitmask for printing system info when panic happens.
The user can chose combination of the following bits:

- bit 0: print all tasks info
- bit 1: print system memory info

- bit 2: print timer info
- bit 3: print locks info if the **CONFIG_LOCKDEP** kernel configuration is on
- bit 4: print the **ftrace** buffer
- bit 5: print all **printk** messages in buffer

rcutree.sysrq_rcu = [KNL]

Commandeer a **sysrq** key to dump out Tree RCU's **rcu_node** tree with an eye towards determining why a new grace period has not yet started.

rcutorture.fwd_progress = [KNL]

Enable Read-copy update (RCU) grace-period forward-progress testing for the types of RCU supporting this notion.

rcutorture.fwd_progress_div = [KNL]

Specify the fraction of a CPU-stall-warning period to do tight-loop forward-progress testing.

rcutorture.fwd_progress_holdoff = [KNL]

Number of seconds to wait between successive forward-progress tests.

rcutorture.fwd_progress_need_resched = [KNL]

Enclose **cond_resched()** calls within checks for **need_resched()** during tight-loop forward-progress testing.

tsx = [X86]

This parameter controls the Transactional Synchronization Extensions (TSX) feature in Intel processors that support TSX control.
The options are:

- **on** - Enable TSX on the system. Although there are mitigations for all known security vulnerabilities, TSX accelerated several previous speculation-related CVEs. As a result, there may be unknown security risks associated with leaving it enabled.
- **off** - Disable TSX on the system. This option takes effect only on newer CPUs which are not vulnerable to Microarchitectural Data Sampling (MDS). In other words they have **MSR_IA32_ARCH_CAPABILITIES.MDS_NO=1** and get the new **IA32_TSX_CTRL** Model-specific register (MSR) through a microcode update. This new MSR allows for a reliable deactivation of the TSX functionality.
- **auto** - Disable TSX if **X86_BUG_TAA** is present, otherwise enable TSX on the system.

Not specifying this parameter is equivalent to **tsx=off**.

For details see the upstream kernel documentation.

tsx_async_abort = [X86,INTEL]

This parameter controls mitigation for the TSX Async Abort (TAA) vulnerability.
Similar to Micro-architectural Data Sampling (MDS), certain CPUs that support Transactional Synchronization Extensions (TSX) are vulnerable to an exploit against CPU internal buffers. The exploit is able to forward information to a disclosure gadget under certain conditions.

In vulnerable processors, the speculatively forwarded data can be used in a cache side channel attack, to access data to which the attacker does not have direct access.

The options are:

- **full** - Enable TAA mitigation on vulnerable CPUs if TSX is enabled.
- **full,nosmt** - Enable TAA mitigation and disable Simultaneous Multi Threading (SMT) on vulnerable CPUs. If TSX is disabled, SMT is not disabled because CPU is not vulnerable to cross-thread TAA attacks.
- **off** - Unconditionally disable TAA mitigation.
  On MDS-affected machines, the **tsx_async_abort=off** parameter can be prevented by an active MDS mitigation as both vulnerabilities are mitigated with the same mechanism. Therefore, to disable this mitigation, you need to specify the **mds=off** parameter as well.

  Not specifying this option is equivalent to **tsx_async_abort=full**. On CPUs which are MDS affected and deploy MDS mitigation, TAA mitigation is not required and does not provide any additional mitigation.

For details see the upstream kernel documentation.

6.2. UPDATED KERNEL PARAMETERS

intel_iommu = [DMAR]

Intel IOMMU driver Direct Memory Access Remapping (DMAR).
The options are:

- **sm_on** [Default Off] - By default, scalable mode will be disabled even if the hardware advertises that it has support for the scalable mode translation. With this option set, scalable mode will be used on hardware which claims to support it.

isolcpus = [KNL,SMP,ISOL]

This parameter isolates a given set of CPUs from disturbance.

- **managed_irq** - A sub-parameter, which prevents the isolated CPUs from being targeted by managed interrupts, which have an interrupt mask containing isolated CPUs. The affinity of managed interrupts is handled by the kernel and cannot be changed via the **/proc/irq/*** interfaces.
  This isolation is the best effort and is only effective if the automatically assigned interrupt mask of a device queue contains isolated and housekeeping CPUs. If the housekeeping CPUs are online then such interrupts are directed to the housekeeping CPU so that I/O submitted on the housekeeping CPU cannot disturb the isolated CPU.

  If the queue's affinity mask contains only isolated CPUs then this parameter has no effect on the interrupt routing decision. However the interrupts are only delivered when the tasks running on those isolated CPUs submit I/O. I/O submitted on the housekeeping CPUs has no influence on those queues.

mds = [X86,INTEL]

The changes to options:

- **off** - On TSX Async Abort (TAA)-affected machines, **mds=off** can be prevented by an active TAA mitigation as both vulnerabilities are mitigated with the same mechanism. So in order to disable this mitigation, you need to specify the **tsx_async_abort=off** kernel parameter too.

Not specifying this parameter is equivalent to **mds=full**.

For details see the upstream kernel documentation.

mem_encrypt = [X86-64]

AMD Secure Memory Encryption (SME) control
...

For details on when the memory encryption can be activated, see the upstream kernel documentation.

mitigations =

The changes to options:

- **off** - Disable all optional CPU mitigations. This improves system performance, but it may also expose users to several CPU vulnerabilities.
  Equivalent to:
  - **nopti [X86,PPC]**
  - **kpti=0 [ARM64]**
  - **nospectre_v1 [X86,PPC]**
  - **nobp=0 [S390]**
  - **nospectre_v2 [X86,PPC,S390,ARM64]**
  - **spectre_v2_user=off [X86]**
  - **spec_store_bypass_disable=off [X86,PPC]**
  - **ssbd=force-off [ARM64]**
  - **l1tf=off [X86]**
  - **mds=off [X86]**
  - **tsx_async_abort=off [X86]**
  - **kvm.nx_huge_pages=off [X86]**
    Exceptions:

    This does not have any effect on **kvm.nx_huge_pages** when **kvm.nx_huge_pages=force**.
- **auto,nosmt** - Mitigate all CPU vulnerabilities, disabling Simultaneous Multi Threading (SMT) if needed. This option is for users who always want to be fully mitigated, even if it means losing SMT.
  Equivalent to:
  - **l1tf=flush,nosmt [X86]**
  - **mds=full,nosmt [X86]**
  - **tsx_async_abort=full,nosmt [X86]**

rcutree.jiffies_till_sched_qs = [KNL]

This parameter sets the required age in jiffies for a given grace period before Read-copy update (RCU) starts soliciting quiescent-state help from the **rcu_note_context_switch()** and **cond_resched()** functions. If not specified, the kernel will calculate a value based on the most recent settings of the **rcutree.jiffies_till_first_fqs** and **rcutree.jiffies_till_next_fqs** kernel parameters. This calculated value may be viewed in the **rcutree.jiffies_to_sched_qs** kernel parameter. Any attempt to set **rcutree.jiffies_to_sched_qs** will be overwritten.

tsc =

This parameter disables clocksource stability checks for Time Stamp Counter (TSC).
Format: <string>

The options are:

- **reliable** [x86] - Marks the TSC clocksource as reliable. This option disables the clocksource verification at runtime, as well as the stability checks done at bootup. The option also enables the high-resolution timer mode on older hardware, and in virtualized environment.
- **noirqtime** [x86] - Do not use TSC to do Interrupt Request (IRQ) accounting. Used to run time disable **IRQ_TIME_ACCOUNTING** on any platforms where Read Time-Stamp Counter (RDTSC) is slow and this accounting can add overhead.
- **unstable** [x86] - Marks the TSC clocksource as unstable. This option marks the TSC unconditionally unstable at bootup and avoids any further wobbles once the TSC watchdog notices.
- **nowatchdog** [x86] - Disables the clocksource watchdog. The option is used in situations with strict latency requirements where interruptions from the clocksource watchdog are not acceptable.

6.3. NEW /PROC/SYS/KERNEL PARAMETERS

panic_print

Bitmask for printing the system info when panic occurs.
The user can chose the combination of the following bits:

- bit 0: print all tasks info
- bit 1: print system memory info
- bit 2: print timer info
- bit 3: print locks info if the **CONFIG_LOCKDEP** kernel configuration item is on
- bit 4: print **ftrace** buffer
  For example, to print tasks and memory info on panic, execute:

```
# echo 3 > /proc/sys/kernel/panic_print
```

sched_energy_aware

This parameter enables or disables Energy Aware Scheduling (EAS).

EAS starts automatically on platforms with asymmetric CPU topologies which have an Energy Model available.

If your platform meets the requirements for EAS but you do not want to use it, change this value to 0.

6.4. UPDATED /PROC/SYS/KERNEL PARAMETERS

threads-max

This parameter controls the maximum number of threads the **fork()** function can create. During initialization, the kernel sets this value in such a way that even if the maximum number of threads is created, the thread structures occupy only a part (1/8th) of the available RAM pages.

The minimum value that can be written to **threads-max** is 1. The maximum value is given by the constant **FUTEX_TID_MASK (0x3fffffff)**.

If a value outside of this range is written to **threads-max**, an error **EINVAL** occurs.

6.5. UPDATED /PROC/SYS/NET PARAMETERS

bpf_jit_enable

This parameter enables the **Berkeley Packet Filter Just-in-Time (BPF JIT)** compiler. **BPF** is a flexible and efficient infrastructure allowing to execute bytecode at various hook points. It is used in a number of Linux kernel subsystems such as networking (for example **XDP**, **tc**), tracing (for example **kprobes**, **uprobes**, **tracepoints**) and security (for example **seccomp**).

LLVM has a **BPF** back-end that can compile restricted C into a sequence of **BPF** instructions. After program load through the **bpf()** system call and passing a verifier in the kernel, **JIT** will then translate these **BPF** proglets into native CPU instructions.

There are two flavors of **JIT**, the newer **eBPF JIT** is currently supported on the following CPU architectures:

- **x86_64**
- **arm64**
- **ppc64** (both little and big endians)
- **s390x**

CHAPTER 7. DEVICE DRIVERS

This chapter provides a comprehensive listing of all device drivers that are new or have been updated in Red Hat Enterprise Linux 8.2.

7.1. NEW DRIVERS

Network drivers

- gVNIC Driver (gve.ko.xz)
- Broadcom UniMAC MDIO bus controller (mdio-bcm-unimac.ko.xz)
- Software iWARP Driver (siw.ko.xz)

Graphics drivers and miscellaneous drivers

- DRM VRAM memory-management helpers (drm_vram_helper.ko.xz)
- cpuidle driver for haltpoll governor (cpuidle-haltpoll.ko.xz)
- stm_ftrace driver (stm_ftrace.ko.xz)
- stm_console driver (stm_console.ko.xz)
- System Trace Module device class (stm_core.ko.xz)
- dummy_stm device (dummy_stm.ko.xz)
- stm_heartbeat driver (stm_heartbeat.ko.xz)
- Intel® Trace Hub Global Trace Hub driver (intel_th_gth.ko.xz)
- Intel® Trace Hub PTI/LPP output driver (intel_th_pti.ko.xz)
- Intel® Trace Hub controller driver (intel_th.ko.xz)
- Intel® Trace Hub Memory Storage Unit driver (intel_th_msu.ko.xz)
- Intel® Trace Hub Software Trace Hub driver (intel_th_sth.ko.xz)
- Intel® Trace Hub Memory Storage Unit software sink (intel_th_msu_sink.ko.xz)
- Intel® Trace Hub PCI controller driver (intel_th_pci.ko.xz)
- Intel® Trace Hub ACPI controller driver (intel_th_acpi.ko.xz)
- MC Driver for Intel 10nm server processors (i10nm_edac.ko.xz)
- Device DAX: direct access mapping device (dax_pmem_core.ko.xz)
- PMEM DAX: direct access to persistent memory (dax_pmem.ko.xz)
- PMEM DAX: support the deprecated /sys/class/dax interface (dax_pmem_compat.ko.xz)
- Intel PMC Core platform init (intel_pmc_core_pltdrv.ko.xz)

- Intel RAPL (Running Average Power Limit) control via MSR interface (intel_rapl_msr.ko.xz)
- Intel Runtime Average Power Limit (RAPL) common code (intel_rapl_common.ko.xz)

Storage drivers

- Clustering support for MD (md-cluster.ko.xz)

7.2. UPDATED DRIVERS

Network driver updates

- VMware vmxnet3 virtual NIC driver (vmxnet3.ko.xz) has been updated to version 1.4.17.0-k.
- Intel® 10 Gigabit Virtual Function Network Driver (ixgbevf.ko.xz) has been updated to version 4.1.0-k-rh8.2.0.
- Intel® 10 Gigabit PCI Express Network Driver (ixgbe.ko.xz) has been updated to version 5.1.0-k-rh8.2.0.
- Intel® Ethernet Connection E800 Series Linux Driver (ice.ko.xz) has been updated to version 0.8.1-k.
- The Netronome Flow Processor (NFP) driver (nfp.ko.xz) has been updated to version 4.18.0-185.el8.x86_64.
- Elastic Network Adapter (ENA) (ena.ko.xz) has been updated to version 2.1.0K.

Graphics and miscellaneous driver updates

- HPE watchdog driver (hpwdt.ko.xz) has been updated to version 2.0.3.
- Intel I/OAT DMA Linux driver (ioatdma.ko.xz) has been updated to version 5.00.

Storage driver updates

- Driver for HPE Smart Array Controller (hpsa.ko.xz) has been updated to version 3.4.20-170-RH4.
- LSI MPT Fusion SAS 3.0 Device Driver (mpt3sas.ko.xz) has been updated to version 32.100.00.00.
- QLogic FCoE Driver (bnx2fc.ko.xz) has been updated to version 2.12.10.
- Emulex LightPulse Fibre Channel SCSI driver (lpfc.ko.xz) has been updated to version 0:12.6.0.2.
- QLogic FastLinQ 4xxxx FCoE Module (qedf.ko.xz) has been updated to version 8.42.3.0.
- QLogic Fibre Channel HBA Driver (qla2xxx.ko.xz) has been updated to version 10.01.00.21.08.2-k.
- Driver for Microsemi Smart Family Controller version (smartpqi.ko.xz) has been updated to version 1.2.10-025.
- QLogic FastLinQ 4xxxx iSCSI Module (qedi.ko.xz) has been updated to version 8.37.0.20.

- Broadcom MegaRAID SAS Driver (megaraid_sas.ko.xz) has been updated to version 07.710.50.00-rc1.

CHAPTER 8. BUG FIXES

This part describes bugs fixed in Red Hat Enterprise Linux 8.2 that have a significant impact on users.

8.1. INSTALLER AND IMAGE CREATION

Using the version or inst.version kernel boot parameters no longer stops the installation program

Previously, booting the installation program from the kernel command line using the **version** or **inst.version** boot parameters printed the version, for example **anaconda 30.25.6**, and stopped the installation program.

With this update, the **version** and **inst.version** parameters are ignored when the installation program is booted from the kernel command line, and as a result, the installation program is not stopped.

(BZ#1637472)

Support secure boot for s390x in the installer

Previously, RHEL 8.1 provided support for preparing boot disks for use in IBM Z environments that enforced the use of secure boot. The capabilities of the server and hypervisor used during installation determined if the resulting on-disk format contained secure boot support. There was no way to influence the on-disk format during installation. Consequently, if you installed RHEL 8.1 in an environment that supported secure boot, the system was unable to boot when moved to an environment that lacked secure boot support, as is done in some failover scenarios.

With this update, you can now configure the secure boot option of the **zipl** tool. To do so, you can use either:

- The Kickstart **zipl** command and one of its options, for example: **--secure-boot**, **--no-secure-boot**, and **--force-secure-boot**.
- From the **Installation Summary** window in the GUI, you can select the **System > Installation Destination > Full disk summary and boot loader** link and set the boot device. As a result, the installation can now be booted in environments that lack secure boot support.

(BZ#1659400)

The secure boot feature is now available

Previously, the default value for the **secure=** boot option was not set to **auto**, and as a result, the secure boot feature was not available. With this update, unless previously configured, the default value is set to **auto**, and the secure boot feature is now available.

(BZ#1750326)

The /etc/sysconfig/kernel file no longer references the new-kernel-pkg script

Previously, the **/etc/sysconfig/kernel** file referenced the **new-kernel-pkg** script. However, the **new-kernel-pkg** script is not included in a RHEL 8 system. With this update, the reference to the **new-kernel-pkg** script has been removed from the **/etc/sysconfig/kernel** file.

(BZ#1747382)

The installation does not set more than the maximum number of allowed devices in the boot-device NVRAM variable

Previously, the RHEL 8 installation program set more than the maximum number of allowed devices in the **boot-device NVRAM** variable. As a result, the installation failed on systems that had more than the maximum number of devices. With this update, the RHEL 8 installation program now checks the maximum device setting and only adds the permitted number of devices.

(BZ#1748756)

Installations work for an image location that uses a URL command in a Kickstart file located in a non-network location

Previously, the installation failed early in the process when network activation triggered by the image remote location was specified by a URL command in a Kickstart file located in a non-network location. This update fixes the issue, and installations that provide the image location by using a URL command in a Kickstart file that is located in a non-network location, for example, a CD-ROM or local block device, now work as expected.

(BZ#1649359)

The RHEL 8 installation program only checks ECKD DASD for unformatted devices

Previously, when checking for unformatted devices, the installation program checked all DASD devices. However, the installation program should only have checked ECKD DASD devices. As a consequence, the installation failed with a traceback when an FBA DASD device with SWAPGEN was used. With this update, the installation program does not check FBA DASD devices, and the installation completes successfully.

(BZ#1715303)

8.2. SOFTWARE MANAGEMENT

yum repolist no longer ends on first unavailable repository

Previously, the repository configuration option **skip_if_unavailable** was by default set as follows:

```
skip_if_unavailable=false
```

This setting forced the **yum repolist** command to end on first unavailable repository with an error and exit status 1. Consequently, **yum repolist** did not continue listing available repositories.

With this update, **yum repolist** has been fixed to no longer require any downloads. As a result, **yum repolist** does not provide any output requiring metadata, and the command now continues listing available repositories as expected.

Note that the number of available packages is only returned by **yum repolist --verbose** or **yum repoinfo** that still require available metadata. Therefore these commands will end on the first unavailable repository.

(BZ#1697472)

8.3. SHELLS AND COMMAND-LINE TOOLS

ReaR updates

RHEL 8.2 introduces a number of updates to the Relax-and-Recover (**ReaR**) utility.

The build directory handling has been changed. Previously, the build directory was kept in a temporary location in case **ReaR** encountered a failure. With this update, the build directory is deleted by default in non-interactive runs to prevent consuming disk space.

The semantics of the **KEEP_BUILD_DIR** configuration variable has been enhanced to include a new **errors** value. You can set the **KEEP_BUILD_DIR** variable to the following values:

- **errors** to preserve the build directory on errors for debugging (the previous behavior)
- **y** (**true**) to always preserve the build directory
- **n** (**false**) to never preserve the build directory

The default value is an empty string with the meaning of **errors** when **ReaR** is being executed interactively (in a terminal) and **false** if **ReaR** is being executed non-interactively. Note that **KEEP_BUILD_DIR** is automatically set to **true** in debug mode (**-d**) and in debugscript mode (**-D**); this behavior has not been changed.

Notable bug fixes include:

- Support for NetBackup 8.0 has been fixed.
- **ReaR** no longer aborts with a bash error similar to **xrealloc: cannot allocate** on systems with a large number of users, groups, and users per group.
- The **bconsole** command now shows its prompt, which enables you to perform a restore operation when using the Bacula integration.
- **ReaR** now correctly backs up files also in situations when the **docker** service is running but no **docker** root directory has been defined, or when it is impossible to determine the status of the **docker** service.
- Recovery no longer fails when using thin pools or recovering a system in Migration Mode.
- Extremely slow rebuild of **initramfs** during the recovery process with LVM has been fixed.
- **ReaR** now creates a working bootable ISO image on the AMD and Intel 64-bit architectures when using the UEFI bootloader. Booting a rescue image in this setup no longer aborts in Grub with the error message **Unknown command 'configfile' (...) Entering rescue mode...**. Support for GRUB_RESCUE in this setup, which previously could fail due to missing XFS filesystem support, has also been fixed.

(BZ#1729501)

mlocate-updatedb.timer is now enabled during the mlocate package installation

Previously, reindexing of the file database was not performed automatically, because the **mlocate-updatedb.timer** timer was disabled after the **mlocate** package installation. With this update, the **mlocate-updatedb.timer** timer is now a part of the **90-default.preset** file and is enabled by default after the **mlocate** package installation. As a result, the file database is updated automatically.

(BZ#1817591)

8.4. INFRASTRUCTURE SERVICES

dnsmasq now correctly handles the non-recursive DNS queries

Previously, **dnsmasq** forwarded all the non-recursive queries to an upstream server, which led to different responses. With this update, the non-recursive queries to local known names, such as DHCP host lease names or hosts read from the **/etc/hosts** file, are handled by **dnsmasq** and are not forwarded to an upstream server. As a result, the same response as to recursive queries to known names is returned.

(BZ#1700916)

dhclient no longer fails to renew the IP address after system time changes

Previously, if the system time changed, the system could lose the IP address assigned due to the removal by the kernel. With this update, **dhclient** uses monotonic timer to detect backward time jumps and issues the **DHCPREQUEST** message for lease extension in case of discontinuous jump in the system time. As a result, the system no longer loses the IP address in the described scenario.

(BZ#1729211)

ipcalc now returns the correct broadcast address for the /31 networks

This update fixes the **ipcalc** utility to follow the RFC 3021 standard properly. As a result, **ipcalc** returns the correct broadcast address when the **/31** prefix is used on an interface.

(BZ#1638834)

/etc/services now contains proper NRPE port definition

This update adds the proper Nagios Remote Plug-in Executor (NRPE) service port definition to the **/etc/services** file.

(BZ#1730396)

The postfix DNS resolver code now uses res_search instead of res_query

Following its previous update in **postfix**, the DNS resolver code used the **res_query** function instead of the **res_search** function. As a consequence, the DNS resolver did not search host names in the current and parent domains with the following **postfix** configuration:

```
# postconf -e "smtp_host_lookup = dns"
# postconf -e "smtp_dns_resolver_options = res_defnames, res_dnsrch"
```

For example, for:

```
# postconf -e "relayhost = [smtp]"
```

and the domain name in the *example.com* format, the DNS resolver did not use the smtp. *example.com* SMTP server for relaying.

With this update, the DNS resolver code has been changed to use **res_search** instead of **res_query**, and it now searches the host names in the current and parent domains correctly.

(BZ#1723950)

PCRE, CDB, and SQLite can now be used with Postfix

In RHEL 8, the **postfix** package has been split into multiple subpackages, each subpackage providing a plug-in for a specific database. Previously, RPM packages containing the **postfix-pcre**, **postfix-cdb**, and **postfix-sqlite** plug-ins were not distributed. Consequently, databases with these plug-ins could not be

used with Postfix. This update adds RPM packages containing the PCRE, CDB, and SQLite plug-ins to the AppStream repository. As a result, these plug-ins can be used after the appropriate RPM package is installed.

(BZ#1745321)

8.5. SECURITY

openssl-pkcs11 no longer locks devices by attempting to log in to multiple devices

Previously, the **openssl-pkcs11** engine attempted to log in to the first result of a search using the provided PKCS #11 URI and used the provided PIN even if the first result was not the intended device and the PIN matched another device. These failed authentication attempts locked the device.

openssl-pkcs11 now attempts to log in to a device only if the provided PKCS #11 URI matches only a single device. The engine now intentionally fails in case the PKCS #11 search finds more than one device. For this reason, you must provide a PKCS #11 URI that matches only a single device when using **openssl-pkcs11** to log in to the device.

(BZ#1705505)

OpenSCAP offline scans using rpmverifyfile now work properly

Prior to this update, the **OpenSCAP** scanner did not correctly change the current working directory in offline mode, and the **fchdir** function was not called with the correct arguments in the **OpenSCAP** **rpmverifyfile** probe. The **OpenSCAP** scanner has been fixed to correctly change the current working directory in offline mode, and the **fchdir** function has been fixed to use correct arguments in **rpmverifyfile**. As a result, SCAP content that contains OVAL **rpmverifyfile** can be used by OpenSCAP to scan arbitrary file systems.

(BZ#1636431)

httpd now starts correctly if using an ECDSA private key without matching public key stored in a PKCS #11 device

Unlike RSA keys, ECDSA private keys do not necessarily contain public-key information. In this case, you cannot obtain the public key from an ECDSA private key. For this reason, a PKCS #11 device stores public-key information in a separate object whether it is a public-key object or a certificate object. OpenSSL expected the **EVP_PKEY** structure provided by an engine for a private key to contain the public-key information. When filling the **EVP_PKEY** structure to be provided to OpenSSL, the engine in the **openssl-pkcs11** package tried to fetch the public-key information only from matching public-key objects and ignored the present certificate objects.

When OpenSSL requested an ECDSA private key from the engine, the provided **EVP_PKEY** structure did not contain the public-key information if the public key was not present in the PKCS #11 device, even when a matching certificate that contained the public key was available. As a consequence, since the Apache **httpd** web server called the **X509_check_private_key()** function, which requires the public key, in its start-up process, **httpd** failed to start in this scenario. This problem has been solved by loading the EC public key from the certificate if the public-key object is not available. As a result, **httpd** now starts correctly when ECDSA keys are stored in a PKCS #11 device.

(BZ#1664807)

scap-security-guide PCI-DSS remediations of Audit rules now work properly

Previously, the **scap-security-guide** package contained a combination of remediation and a check that could result in one of the following scenarios:

- incorrect remediation of Audit rules
- scan evaluation containing false positives where passed rules were marked as failed

Consequently, during the RHEL installation process, scanning of the installed system reported some Audit rules as either failed or errored.

With this update, the remediations have been fixed, and scanning of the system installed with the PCI-DSS security policy no longer reports false positives for Audit rules.

(BZ#1754919)

OpenSCAP now provides offline scanning of virtual machines and containers

Previously, refactoring of the **OpenSCAP** codebase caused certain RPM probes to fail to scan VM and containers file systems in offline mode. Consequently, the following tools could not be included in the **openscap-utils** package: **oscap-vm** and **oscap-chroot**. Furthermore, the **openscap-containers** package was completely removed from RHEL 8. With this update, the problems in the probes have been fixed.

As a result, RHEL 8 now contains the **oscap-podman**, **oscap-vm**, and **oscap-chroot** tools in the **openscap-utils** package.

(BZ#1618489)

OpenSCAP rpmverifypackage now works correctly

Previously, the **chdir** and **chroot** system calls were called twice by the **rpmverifypackage** probe. Consequently, an error occurred when the probe was utilized during an **OpenSCAP** scan with custom Open Vulnerability and Assessment Language (OVAL) content. The **rpmverifypackage** probe has been fixed to properly utilize the **chdir** and **chroot** system calls. As a result, **rpmverifypackage** now works correctly.

(BZ#1646197)

8.6. NETWORKING

Locking in the qdisc_run function now does not cause kernel crash

Previously, a race condition when the **pfifo_fast** queue discipline resets while dequeuing traffic was leading to packet transmission after they were freed. As a consequence, sometimes kernel was getting terminated unexpectedly. With this update, locking in the **qdisc_run** function has been improved. As a result, kernel no longer crashes in the described scenario.

(BZ#1744397)

The DBus APIs in org.fedoraproject.FirewallD1.config.service work as expected

Previously, the DBus API **getIncludes**, **setIncludes**, and **queryIncludes** functions in **org.fedoraproject.FirewallD1** returned an error message: **org.fedoraproject.FirewallD1.Exception: list index out of range** due to bad indexing. With this update, the DBus API **getIncludes**, **setIncludes**, and **queryIncludes** functions work as expected.

(BZ#1737045)

RHEL no longer logs a kernel warning when unloading the ipvs module

Previously, the IP virtual server (**ipvs**) module used an incorrect reference counting, which caused a race condition when unloading the module. Consequently, RHEL logged a kernel warning. This update fixes the race condition. As a result, the kernel no longer logs the warning when you unload the **ipvs** module.

(BZ#1687094)

The nft utility no longer interprets arguments as command-line options after the first non-option argument

Previously, the **nft** utility accepted options anywhere in an **nft** command. For example, admins could use options between or after non-option arguments. As a consequence, due to the leading dash, **nft** interpreted negative priority values as options, and the command failed. The **nft** utility's command-line parser has been updated to not interpret arguments that are starting with a dash after the first non-option argument has been read. As a result, admins no longer require workarounds to pass negative priority values to **nft**.

Note that due to this change, you must now pass all command-options to **nft** before the first non-option argument. Before you update, verify your nftables scripts to match this new criteria to ensure that the script works as expected after you installed this update.

(BZ#1778883)

The /etc/hosts.allow and /etc/hosts.deny files no longer contain outdated references to removed tcp_wrappers

Previously, the **/etc/hosts.allow** and **/etc/hosts.deny** files contained outdated information about the **tcp_wrappers** package. The files are removed in RHEL 8 as they are no longer needed for **tcp_wrappers** which is removed.

(BZ#1663556)

A configuration parameter has been added to firewalld to disable zone drifting

Previously, the **firewalld** service contained an undocumented behavior known as "zone drifting". RHEL 8.0 removed this behavior because it could have a negative security impact. As a consequence, on hosts that used this behavior to configure a catch-all or fallback zone, **firewalld** denied connections that were previously allowed. This update re-adds the zone drifting behavior, but as a configurable feature. As a result, users can now decide to use zone drifting or disable the behavior for a more secure firewall setup.

By default, in RHEL 8.2, the new **AllowZoneDrifting** parameter in the **/etc/firewalld/firewalld.conf** file is set to **yes**. Note that, if the parameter is enabled, **firewalld** logs:

> WARNING: AllowZoneDrifting is enabled. This is considered an insecure configuration option. It will be removed in a future release. Please consider disabling it now.

(BZ#1772208)

8.7. KERNEL

Subsection memory hotplug is now fully supported

Previously, some platforms aligned physical memory regions such as Dual In-Line Modules (DIMMs) and interleave sets to 64MiB memory boundary. However, as the Linux hotplug subsystem uses a memory size of 128MiB, hot-plugging new devices caused multiple memory regions to overlap in a single hotplug memory window. Consequently, this caused failure in listing the available persistent memory namespaces with the following or a similar call trace:

```
WARNING: CPU: 38 PID: 928 at arch/x86/mm/init_64.c:850
add_pages+0x5c/0x60
   [..]
   RIP: 0010:add_pages+0x5c/0x60
   [..]
   Call Trace:
    devm_memremap_pages+0x460/0x6e0
    pmem_attach_disk+0x29e/0x680 [nd_pmem]
    ? nd_dax_probe+0xfc/0x120 [libnvdimm]
    nvdimm_bus_probe+0x66/0x160 [libnvdimm]
```

This update fixes the problem and supports Linux hotplug subsystem to enable multiple memory regions to share a single hotplug memory window.

(BZ#1724969)

Data corruption now triggers a BUG instead of a WARN message

With this enhancement, the list corruptions at **lib/list_debug.c** now triggers a BUG, which generates a report with a **vmcore**. Previously, when encountering a data corruption, a simple WARN was generated, which was likely to go unnoticed. With **set CONFIG_BUG_ON_DATA_CORRUPTION**, the kernel now creates a crash and triggers a BUG in response to data corruption. This prevents further damage and reduces the security risk. The **kdump** now generates a **vmcore**, which improves the data corruption bug reporting.

(BZ#1714330)

Support for Intel Carlsville card is available but not verified in RHEL 8.2

The **Intel Carlsville** card support is available but not tested on Red Hat Enterprise Linux 8.2.

(BZ#1720227)

RPS and XPS no longer place jobs on isolated CPUs

Previously, the Receive Packet Steering (RPS) software-queue mechanism and the Transmit Packet Steering (XPS) transmit queue selection mechanism allocated jobs on all CPU sets, including isolated CPUs. Consequently, this could cause an unexpected latency spike in a real-time environment when a latency-sensitive workload was using the same CPU where RPS or XPS jobs were running. With this update, the **store_rps_map()** function does not include any isolated CPUs for the purpose of RPS configuration. Similarly, the kernel drivers used for XPS configuration are respecting CPU isolation. As a result, RPS and XPS no longer place jobs on isolated CPUs in the described scenario. If you configure an isolated CPU in the **/sys/devices/pci*/net/dev/queues/rx-*/rps_cpus** file, the following error appears:

```
Error: "-bash: echo:write error: Invalid argument"
```

However, manually configuring an isolated CPU in the **/sys/devices/pci*/net/dev/queues/tx-*/xps_cpus** file successfully allocates XPS jobs on the isolated CPU.

Note that a networking workload in an environment with isolated CPUs is likely to experience some performance variation.

(BZ#1867174)

8.8. FILE SYSTEMS AND STORAGE

SCSI drivers no longer use an excessive amount of memory

Previously, certain SCSI drivers used a larger amount of memory than in RHEL 7. In certain cases, such as vPort creation on a Fibre Channel host bus adapter (HBA), the memory usage was excessive, depending upon the system configuration.

The increased memory usage was caused by memory preallocation in the block layer. Both the multiqueue block device scheduling (BLK-MQ) and the multiqueue SCSI stack (SCSI-MQ) preallocated memory for each I/O request, leading to the increased memory usage.

With this update, the block layer limits the amount of memory preallocation, and as a result, the SCSI drivers no longer use an excessive amount of memory.

(BZ#1698297)

VDO can now suspend before UDS has finished rebuilding

Previously, the **dmsetup suspend** command became unresponsive if you attempted to suspend a VDO volume while the UDS index was rebuilding. The command finished only after the rebuild.

With this update, the problem has been fixed. The **dmsetup suspend** command can finish before the UDS rebuild is done without becoming unresponsive.

(BZ#1737639)

8.9. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

Problems in mod_cgid logging have been fixed

Prior to this update, if the **mod_cgid** Apache **httpd** module was used under a threaded multi-processing module (MPM), the following logging problems occurred:

- The **stderr** output of the CGI script was not prefixed with standard timestamp information.
- The **stderr** output of the CGI script was not correctly redirected to a log file specific to the **VirtualHost**, if configured.

This update fixes the problems, and **mod_cgid** logging now works as expected.

(BZ#1633224)

8.10. COMPILERS AND DEVELOPMENT TOOLS

Unrelocated and uninitialized shared objects no longer result in failures if dlopen fails

Previously, if the **dlopen** call failed, the **glibc** dynamic linker did not remove shared objects with the **NODELETE** mark before reporting the error. Consequently, the unrelocated and uninitialized shared objects remained in the process image, eventually resulting in assertion failures or crashes. With this update, the dynamic loader uses a pending **NODELETE** state to remove shared objects upon **dlopen** failure, before marking them as **NODELETE** permanently. As a result, the process does not leave any unrelocated objects behind. Also, lazy binding failures while ELF constructors and destructors run now terminate the process.

(BZ#1410154)

Advanced SIMD functions on the 64-bit ARM architecture no longer miscompile when lazily resolved

Previously, the new vector Procedure Call Standard (PCS) for Advanced SIMD did not properly save and restore certain callee-saved registers when lazily resolving Advanced SIMD functions. As a consequence, binaries could misbehave at runtime. With this update, the Advanced SIMD and SVE vector functions in the symbol table are marked with **.variant_pcs** and, as a result, the dynamic linker will bind such functions early.

(BZ#1726641)

The sudo wrapper script now parses options

Previously, the **/opt/redhat/devtoolset*/root/usr/bin/sudo** wrapper script did not correctly parse **sudo** options. As a consequence, some **sudo** options (for example, **sudo -i**) could not be executed. With this update, more **sudo** options are correctly parsed and, as a result, the **sudo** wrapper script works more like **/usr/bin/sudo**.

(BZ#1774118)

Alignment of TLS variables in glibc has been fixed

Previously, aligned thread-local storage (TLS) data could, under certain conditions, become instantiated without the expected alignment. With this update, the POSIX Thread Library **libpthread** has been enhanced to ensure correct alignment under any conditions. As a result, aligned TLS data is now correctly instantiated for all threads with the correct alignment.

(BZ#1764214)

Repeated pututxline calls following EINTR or EAGAIN error no longer corrupt the utmp file

When the **pututxline** function tries to acquire a lock and does not succeed in time, the function returns with **EINTR** or **EAGAIN** error code. Previously in this situation, if **pututxline** was called immediately again and managed to obtain the lock, it did not use an already-allocated matching slot in the **utmp** file, but added another entry instead. As a consequence, these unused entries increased the size of the **utmp** file substantially. This update fixes the issue, and the entries are added to the **utmp** file correctly now.

(BZ#1749439)

mtrace no longer hangs when internal failures occur

Previously, a defect in the **mtrace** tool implementation could cause memory tracing to hang. To fix this issue, the **mtrace** memory tracing implementation has been made more robust to avoid the hang even in the face of internal failures. As a result, users can now call **mtrace** and it no longer hangs, completing in bounded time.

(BZ#1764235)

The fork function avoids certain deadlocks related to use of pthread_atfork

Previously, if a program registered an **atfork** handler and invoked **fork** from an asynchronous-signal handler, a defect in the internal implementation-dependent lock could cause the program to freeze. With this update, the implementation of **fork** and its **atfork** handlers is adjusted to avoid the deadlock in single-threaded programs.

(BZ#1746928)

strstr no longer returns incorrect matches for a truncated pattern

On certain IBM Z platforms (z15, previously known as arch13), the **strstr** function did not correctly update a CPU register when handling search patterns that cross a page boundary. As a consequence, **strstr** returned incorrect matches. This update fixes the problem, and as a result, **strstr** works as expected in the mentioned scenario.

(BZ#1777241)

C.UTF-8 locale source ellipsis expressions in glibc are fixed

Previously, a defect in the C.UTF-8 source locale resulted in all Unicode code points above U+10000 lacking collation weights. As a consequence, all code points above U+10000 did not collate as expected. The C.UTF-8 source locale has been corrected, and the newly compiled binary locale now has collation weights for all Unicode code points. The compiled C.UTF-8 locale is 5.3MiB larger as a result of this fix.

(BZ#1361965)

glibc no longer fails when getpwent() is called without calling setpwent()

If your **/etc/nsswitch.conf** file pointed to the Berkeley DB (**db**) password provider, you could request data using the **getpwent()** function without first calling **setpwent()** only once. When you called the **endpwent()** function, further calls to **getpwent()** without first calling **setpwent()** caused **glibc** to fail because **endpwent()** could not reset the internals to allow a new query. This update fixes the problem. As a result, after you end one query with **endpwent()**, further calls to **getpwent()** will start a new query even if you do not call **setpwent()**.

(BZ#1747502)

ltrace can now trace system calls in hardened binaries

Previously, **ltrace** did not produce any results on certain hardened binaries, such as system binaries, on the AMD and Intel 64-bit architectures. With this update, **ltrace** can now trace system calls in hardened binaries.

(BZ#1655368)

Intel's JCC flaw no longer causes significant performance loss in the GCC compiler

Certain Intel CPUs are affected by the Jump Conditional Code (JCC) bug causing machine instructions to be executed incorrectly. Consequently, the affected CPUs might not execute programs properly. The full fix involves updating the microcode of vulnerable CPUs, which can cause a performance degradation. This update enables a workaround in the assembler that helps to reduce the performance loss. The workaround is **not** enabled by default.

To apply the workaround, recompile a program using GCC with the **-Wa,-mbranches-within-32B-boundaries** command-line option. A program recompiled with this command-line option will not be affected by the JCC flaw, but the microcode update is still necessary to fully protect a system.

Note that applying the workaround will increase the size of the program and can still cause a slight performance decrease, although it should be less than it would have been without the recompilation.

(BZ#1777002)

make no longer slows down when using parallel builds

Previously, while running parallel builds, **make** sub-processes could become temporarily unresponsive when waiting for their turn to run. As a consequence, builds with high **-j** values slowed down or ran at lower effective **-j** values. With this update, the job control logic of **make** is now non-blocking. As a result, builds with high **-j** values run at full **-j** speed.

(BZ#1774790)

The ltrace tool now reports function calls correctly

Because of improvements to binary hardening applied to all RHEL components, the **ltrace** tool previously could not detect function calls in binary files coming from RHEL components. As a consequence, **ltrace** output was empty because it did not report any detected calls when used on such binary files. This update fixes the way **ltrace** handles function calls, which prevents the described problem from occurring.

(BZ#1618748)

8.11. IDENTITY MANAGEMENT

The dsctl utility no longer fails to manage instances with a hyphen in their name

Previously, the **dsctl** utility did not correctly parse hyphens in the Directory Server instance names. As a consequence, administrators could not use **dsctl** to manage instances with a hyphen in their name. This update fixes the problem, and **dsctl** now works as expected in the mentioned scenario.

(BZ#1715406)

Directory Server instance names can now have up to 103 characters

When an LDAP client establishes a connection to Directory Server, the server stores information related to the client address in a local buffer. Previously, the size of this buffer was too small to store an LDAPI path name longer than 46 characters. For example, this is the case if name of the Directory Server instance is too long. As a consequence, the server terminated unexpectedly due to an buffer overflow. This update increases the buffer size to the maximum size the Netscape Portable Runtime (NSPR) library supports for the path name. As a result, Directory Server no longer crashes in the mentioned scenario.

Note that due to the limitation in the NSPR library, an instance name can be maximum 103 characters.

(BZ#1748016)

The pkidestroy utility now picks the correct instance

Previously, the **pkidestroy --force** command executed on a half-removed instance picked the **pki-tomcat** instance by default, regardless of the instance name specified with the **-i *instance*** option.

As a consequence, this removed the **pki-tomcat** instance instead of the intended instance, and the **--remove-logs** option did not remove the intended instance's logs. **pkidestroy** now applies the right instance name, removing only the intended instance's leftovers.

(BZ#1698084)

The ldap_user_authorized_service description has been updated in the sssd-ldap man page

The Pluggable authentication modules (PAM) stack has been changed in RHEL 8. For example, the **systemd** user session now starts a PAM conversation using the **systemd-user** PAM service. This service now recursively includes the **system-auth** PAM service, which may include the **pam_sss.so** interface. This means that the SSSD access control is always called.

You should be aware of this change when designing access control rules for RHEL 8 systems. For example, you can add the **systemd-user** service to the allowed services list.

Please note for some access control mechanisms, such as IPA HBAC or AD GPOs, the **systemd-user** service has been added to the allowed services list by default and you do not need to take any action.

The **sssd-ldap** man page has been updated to include this information.

(BZ#1669407)

Information about required DNS records is now displayed when enabling support for AD trust in IdM

Previously, when enabling support for Active Directory (AD) trust in Red Hat Enterprise Linux Identity Management (IdM) installation with external DNS management, no information about required DNS records was displayed. Entering the **ipa dns-update-system-records --dry-run** command manually was necesary to obtain a list of all DNS records required by IdM.

With this update, the **ipa-adtrust-install** command correctly lists the DNS service records for manual addition to the DNS zone.

(BZ#1665051)

8.12. DESKTOP

GNOME Shell on Wayland no longer performs slowly when using a software renderer

Previously, the Wayland back end of GNOME Shell did not use a cacheable framebuffer when using a software renderer. As a consequence, software-rendered GNOME Shell on Wayland was slow compared to software-rendered GNOME Shell on the X.org back end.

With this update, an intermediate shadow framebuffer has been added in GNOME Shell on Wayland. As a result, software-rendered GNOME Shell on Wayland now performs as well as GNOME Shell on X.org.

(BZ#1737553)

8.13. VIRTUALIZATION

Starting a VM on a 10th generation Intel Core processor no longer fails

Previously, starting a virtual machine (VM) failed on a host model that used a 10th generation Intel Core processor, also known as Icelake-Server. With this update, **libvirt** no longer attempts to disable the **pconfig** CPU feature which is not supported by QEMU. As a result, starting a VM on a host model running a 10th generation Intel processor no longer fails.

(BZ#1749672)

Using cloud-init to provision virtual machines on Microsoft Azure now works correctly

Previously, it was not possible to use the **cloud-init** utility to provision a RHEL 8 virtual machine (VM) on the Microsoft Azure platform. This update fixes the **cloud-init** handling of the Azure endpoints, and provisioning RHEL 8 VMs on Azure now proceeds as expected.

(BZ#1641190)

RHEL 8 virtual machines on RHEL 7 hosts can be reliably viewed in higher resolution than 1920x1200

Previously, when using a RHEL 8 virtual machine (VM) running on a RHEL 7 host system, certain methods of displaying the the graphical output of the VM, such as running the application in kiosk mode,

could not use greater resolution than 1920x1200. As a consequence, displaying VMs using those methods only worked in resolutions up to 1920x1200 even if the host hardware supported higher resolutions. This update adjusts DRM and QXL drivers in a way to prevent the described problem from occurring.

(BZ#1635295)

Customizing an ESXi VM using cloud-init and rebooting the VM now works correctly

Previously, if the **cloud-init** service was used to modify a virtual machine (VM) running on the VMware ESXi hypervisor to use static IP and the VM was then cloned, the new cloned VM in some cases took a very long time to reboot. This update modifies **cloud-init** not to rewrite the VM's static IP to DHCP, which prevents the described problem from occurring.

(BZ#1666961, BZ#1706482)

8.14. CONTAINERS

Pulling images from the quay.io registry no longer leads to unintended images

Previously, having the quay.io container image registry listed in the default registries search list provided in **/etc/containers/registries.conf** could allow a user to pull a spoofed image when using a short name. To fix this issue, the quay.io container image registry has been removed from the default registries search list in **/etc/containers/registries.conf**. As a result, pulling images from the **quay.io** registry now requires users to specify the full repository name, such as **quay.io/myorg/myimage**. The quay.io registry can be added back to the default registries search list in **/etc/containers/registries.conf** to reenable pulling container images using short names, however, this is not recommended as it could create a security risk.

(BZ#1784267)

CHAPTER 9. TECHNOLOGY PREVIEWS

This part provides a list of all Technology Previews available in Red Hat Enterprise Linux 8.2.

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

9.1. NETWORKING

nmstate available as a Technology Preview

Nmstate is a network API for hosts. The **nmstate** packages, available as a Technology Preview, provide a library and the **nmstatectl** command-line utility to manage host network settings in a declarative manner. The networking state is described by a pre-defined schema. Reporting of the current state and changes to the desired state both conform to the schema.

For further details, see the **/usr/share/doc/nmstate/README.md** file and the examples in the **/usr/share/doc/nmstate/examples** directory.

(BZ#1674456)

AF_XDP available as a Technology Preview

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

(BZ#1633143)

XDP available as a Technology Preview

The eXpress Data Path (XDP) feature, which is available as a Technology Preview, provides a means to attach extended Berkeley Packet Filter (eBPF) programs for high-performance packet processing at an early point in the kernel ingress data path, allowing efficient programmable packet analysis, filtering, and manipulation.

(BZ#1503672)

KTLS available as a Technology Preview

In Red Hat Enterprise Linux 8, Kernel Transport Layer Security (KTLS) is provided 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 dracut utility now supports creating initrd images with NetworkManager support as a technology preview

By default, the **dracut** utility uses a shell script to manage networking in the initial RAM disk (**initrd**). In certain cases, this could cause problems when the system switches from the RAM disk to the operating system that uses NetworkManager to configure the network. For example, NetworkManager could send another DHCP request, even if the script in the RAM disk already requested an IP address. This request from the RAM disk could result in a time out.

To solve these kind of problems, **dracut** in RHEL 8.2 can now use NetworkManager in the RAM disk. Use the following commands to enable the feature and recreate the RAM disk images:

```
echo 'add_dracutmodules+=" network-manager "' > /etc/dracut.conf.d/enable-nm.conf
dracut -vf --regenerate-all
```

Note that Red Hat does not support technology preview features. However, to provide feedback about this feature, please contact the Red Hat support.

(BZ#1626348)

The mlx5_core driver supports Mellanox ConnectX-6 Dx network adapter as a Technology Preview

This enhancement adds the PCI IDs of the Mellanox ConnectX-6 Dx network adapter to the **mlx5_core** driver. On hosts that use this adapter, RHEL loads the **mlx5_core** driver automatically. Note that Red Hat provides this feature as an unsupported Technology Preview.

(BZ#1687434)

9.2. KERNEL

kexec fast reboot as a Technology Preview

The **kexec fast reboot** feature, continues to be available as a Technology Preview. Rebooting is now significantly faster thanks to **kexec fast reboot**. To use this feature, load the kexec kernel manually, and then reboot the operating system.

(BZ#1769727)

eBPF available as a Technology Preview

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

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

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

There are numerous components shipped by Red Hat that utilize the **eBPF** virtual machine. Each component is in a different development phase, and thus not all components are currently fully supported. All components are available as a Technology Preview, unless a specific component is indicated as supported.

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

- **bpftrace**, a high-level tracing language that utilizes the **eBPF** virtual machine.
- The eXpress Data Path (XDP) feature, a networking technology that enables fast packet processing in the kernel using the **eBPF** virtual machine.

(BZ#1559616)

libbpf is available as a Technology Preview

The **libbpf** package is currently available as a Technology Preview. The **libbpf** package is crucial for bpf related applications like **bpftrace** and **bpf/xdp** development.

It is a mirror of bpf-next linux tree **bpf-next/tools/lib/bpf** directory plus its supporting header files. The version of the package reflects the version of the Application Binary Interface (ABI).

(BZ#1759154)

The igc driver available as a Technology Preview for RHEL 8

The **igc** Intel 2.5G Ethernet Linux wired LAN driver is now available on all architectures for RHEL 8 as a Technology Preview. The **ethtool** utility also supports **igc** wired LANs.

(BZ#1495358)

9.3. FILE SYSTEMS AND STORAGE

NVMe/TCP is available as a Technology Preview

Accessing and sharing Nonvolatile Memory Express (NVMe) storage over TCP/IP networks (NVMe/TCP) and its corresponding **nvme-tcp.ko** and **nvmet-tcp.ko** kernel modules have been added as a Technology Preview.

The use of NVMe/TCP as either a storage client or a target is manageable with tools provided by the **nvme-cli** and **nvmetcli** packages.

The NVMe/TCP target Technology Preview is included only for testing purposes and is not currently planned for full support.

(BZ#1696451)

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

In Red Hat Enterprise Linux 8.2, file system DAX is available as a Technology Preview. DAX provides a means for an application to directly map persistent memory into its address space. To use DAX, a system must have some form of persistent memory available, usually in the form of one or more Non-Volatile Dual In-line Memory Modules (NVDIMMs), and a file system that 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#1627455)

OverlayFS

OverlayFS is a type of union file system. It enables you to overlay one file system on top of another. Changes are recorded in the upper file system, while the lower file system remains unmodified. This allows multiple users to share a file-system image, such as a container or a DVD-ROM, where the base image is on read-only media. See the Linux kernel documentation for additional information: https://www.kernel.org/doc/Documentation/filesystems/overlayfs.txt.

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

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

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

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

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

- The OverlayFS kernel ABI and userspace behavior are not considered stable, and might see changes in future updates.

- OverlayFS provides a restricted set of the POSIX standards. Test your application thoroughly before deploying it with OverlayFS. The following cases are not POSIX-compliant:

  - Lower files opened with **O_RDONLY** do not receive **st_atime** updates when the files are read.

  - Lower files opened with **O_RDONLY**, then mapped with **MAP_SHARED** are inconsistent with subsequent modification.

  - Fully compliant **st_ino** or **d_ino** values are not enabled by default on RHEL 8, but you can enable full POSIX compliance for them with a module option or mount option.
    To get consistent inode numbering, use the **xino=on** mount option.

    You can also use the **redirect_dir=on** and **index=on** options to improve POSIX compliance. These two options make the format of the upper layer incompatible with an overlay without these options. That is, you might get unexpected results or errors if you create an overlay with **redirect_dir=on** or **index=on**, unmount the overlay, then mount the overlay without these options.

- Commands used with XFS:

  - XFS file systems must be created with the **-n ftype=1** option enabled for use as an overlay.

  - With the rootfs and any file systems created during system installation, set the **--mkfsoptions=-n ftype=1** parameters in the Anaconda kickstart.

  - When creating a new file system after the installation, run the **# mkfs -t xfs -n ftype=1 /PATH/TO/DEVICE** command.

  - To determine whether an existing file system is eligible for use as an overlay, run the **# xfs_info /PATH/TO/DEVICE | grep ftype** command to see if the **ftype=1** option is enabled.

- SELinux security labels are enabled by default in all supported container engines with OverlayFS.

- There are several known issues associated with OverlayFS in this release. For details, see *Non-standard behavior* in the Linux kernel documentation: https://www.kernel.org/doc/Documentation/filesystems/overlayfs.txt.

(BZ#1690207)

Stratis is now available as a Technology Preview

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

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

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

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

Stratis is provided as a Technology Preview.

For more information, see the Stratis documentation: Managing layered local storage with Stratis .

RHEL 8.2 updates Stratis to version 2.0.0. This version improves reliability and the Stratis DBus API. For more information about version 2.0.0, see Stratis 2.0.0 Release Notes .

(JIRA:RHELPLAN-1212)

IdM now supports setting up a Samba server on an IdM domain member as a Technology Preview

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

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

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

(JIRA:RHELPLAN-13195)

9.4. HIGH AVAILABILITY AND CLUSTERS

Pacemaker podman bundles available as a Technology Preview

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

(BZ#1619620)

Heuristics in corosync-qdevice available as a Technology Preview

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

(BZ#1784200)

New fence-agents-heuristics-ping fence agent

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

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

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

(BZ#1775847)

9.5. IDENTITY MANAGEMENT

Identity Management JSON-RPC API available as Technology Preview

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

In Red Hat Enterprise Linux 7.3, the IdM API was enhanced to enable multiple versions of API commands. Previously, enhancements could change the behavior of a command in an incompatible way. Users are now able to continue using existing tools and scripts even if the IdM API changes. This enables:

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

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

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

(BZ#1664719)

DNSSEC available as Technology Preview in IdM

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

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

- DNSSEC Operational Practices, Version 2: http://tools.ietf.org/html/rfc6781#section-2
- Secure Domain Name System (DNS) Deployment Guide: http://dx.doi.org/10.6028/NIST.SP.800-81-2
- DNSSEC Key Rollover Timing Considerations: http://tools.ietf.org/html/rfc7583

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

(BZ#1664718)

Checking the overall health of your public key infrastructure is now available as a Technology Preview

With this update, the public key infrastructure (PKI) Healthcheck tool reports the health of the PKI subsystem to the Identity Management (IdM) Healthcheck tool, which was introduced in RHEL 8.1. Executing the IdM Healthcheck invokes the PKI Healthcheck, which collects and returns the health report of the PKI subsystem.

The **pki-healthcheck** tool is available on any deployed RHEL IdM server or replica. All the checks provided by **pki-healthcheck** are also integrated into the **ipa-healthcheck** tool. **ipa-healthcheck** can be installed separately from the **idm:DL1** module stream.

Note that **pki-healthcheck** can also work in a standalone Red Hat Certificate System (RHCS) infrastructure.

(BZ#1303254)

9.6. DESKTOP

GNOME Desktop on ARM is available as a Technology Preview

The GNOME Desktop is now available as a Technology Preview on the 64-bit ARM architecture. Users who require a graphical session to configure and manage their servers can now connect to a remote graphical session running GNOME Desktop using VNC.

(BZ#1724302)

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

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

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

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

9.7. GRAPHICS INFRASTRUCTURES

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

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

(BZ#1698565)

9.8. RED HAT ENTERPRISE LINUX SYSTEM ROLES

The postfix role of RHEL System Roles available as a Technology Preview

Red Hat Enterprise Linux System Roles provides a configuration interface for Red Hat Enterprise Linux subsystems, which makes system configuration easier through the inclusion of Ansible Roles. This interface enables managing system configurations across multiple versions of Red Hat Enterprise Linux, as well as adopting new major releases.

The **rhel-system-roles** packages are distributed through the AppStream repository.

The **postfix** role is available as a Technology Preview.

The following roles are fully supported:

- **kdump**
- **network**
- **selinux**
- **storage**
- **timesync**

For more information, see the Knowledgebase article about RHEL System Roles.

(BZ#1812552)

rhel-system-roles-sap available as a Technology Preview

The **rhel-system-roles-sap** package provides Red Hat Enterprise Linux (RHEL) System Roles for SAP, which can be used to automate the configuration of a RHEL system to run SAP workloads. These roles greatly reduce the time to configure a system to run SAP workloads by automatically applying the optimal settings that are based on best practices outlined in relevant SAP Notes. Access is limited to RHEL for SAP Solutions offerings. Please contact Red Hat Customer Support if you need assistance with your subscription.

The following new roles in the **rhel-system-roles-sap** package are available as a Technology Preview:

- **sap-preconfigure**
- **sap-netweaver-preconfigure**
- **sap-hana-preconfigure**

For more information, see Red Hat Enterprise Linux System Roles for SAP .

Note: RHEL 8.2 for SAP Solutions is scheduled to be validated for use with SAP HANA on Intel 64 architecture and IBM POWER9. Support for other SAP applications and database products, for

example, SAP NetWeaver and SAP ASE, are tied to GA releases, and customers can use RHEL 8.2 features upon GA. Please consult SAP Notes 2369910 and 2235581 for the latest information about validated releases and SAP support.

(BZ#1660832)

9.9. VIRTUALIZATION

Select Intel network adapters now support SR-IOV in RHEL guests on Hyper-V

As a Technology Preview, Red Hat Enterprise Linux guest operating systems running on a Hyper-V hypervisor can now use the single-root I/O virtualization (SR-IOV) feature for Intel network adapters supported by the **ixgbevf** and **i40evf** drivers. This feature is enabled when the following conditions are met:

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

The feature is currently supported with Microsoft Windows Server 2019 and 2016.

(BZ#1348508)

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

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

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

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

(BZ#1519039)

AMD SEV for KVM virtual machines

As a Technology Preview, RHEL 8 introduces 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 VM memory so that the host cannot access data on the VM. This increases the security of the VM if the host is successfully infected by malware.

Note that the number of VMs that can use this feature at a time on a single host is determined by the host hardware. Current AMD EPYC processors support up to 509 running VMs using SEV.

Also note that for VMs with SEV configured to be able to boot, you must also configure the VM with a hard memory limit. To do so, add the following to the VM's XML configuration:

```
<memtune>
<hard_limit unit='KiB'>N</hard_limit>
</memtune>
```

The recommended value for N is equal to or greater then the guest RAM + 256 MiB. For example, if the guest is assigned 2 GiB RAM, N should be 2359296 or greater.

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

Intel vGPU

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

Note that only selected Intel GPUs are compatible with the vGPU feature. In addition, assigning a physical GPU to VMs makes it impossible for the host to use the GPU, and may prevent graphical display output on the host from working.

(BZ#1528684)

9.10. CONTAINERS

skopeo container image is available as a Technology Preview

The **registry.redhat.io/rhel8/skopeo** container image is a containerized implementation of the **skopeo** package. The **skopeo** is a command-line tool utility that performs various operations on container images and image repositories. This container image allows you to inspect and copy container images from one unauthenticated container registry to another.

(BZ#1627900)

buildah container image is available as a Technology Preview

The **registry.redhat.io/rhel8/buildah** container image is a containerized implementation of the **buildah** package. The **buildah** is a tool that facilitates building OCI container images. This container image allows you to build container images without the need to install the **buildah** package on your system. The use-case does not cover running this image in rootless mode as a non-root user.

(BZ#1627898)

CHAPTER 10. DEPRECATED FUNCTIONALITY

This part provides an overview of functionality that has been *deprecated* in Red Hat Enterprise Linux 8.2.

Deprecated functionality continues to be supported until the end of life of Red Hat Enterprise Linux 8. Deprecated functionality will likely not be supported in future major releases of this product and is not recommended for new deployments. For the most recent list of deprecated functionality within a particular major release, refer to the latest version of release documentation.

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

A package can be deprecated and not recommended for further use. Under certain circumstances, a package can be removed from a product. Product documentation then identifies more recent packages that offer functionality similar, identical, or more advanced to the one deprecated, and provides further recommendations.

For information regarding functionality that is present in RHEL 7 but has been *removed* in RHEL 8, see Considerations in adopting RHEL 8 .

10.1. INSTALLER AND IMAGE CREATION

Several Kickstart commands and options have been deprecated

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

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

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

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

(BZ#1642765)

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

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

(BZ#1637872)

10.2. SOFTWARE MANAGEMENT

rpmbuild --sign is deprecated

With this update, the **rpmbuild --sign** command has become deprecated. Using this command in future releases of Red Hat Enterprise Linux can result in an error. It is recommended that you use the **rpmsign** command instead.

(BZ#1688849)

10.3. SECURITY

NSS SEED ciphers are deprecated

The Mozilla Network Security Services (**NSS**) library will not support TLS cipher suites that use a SEED cipher in a future release. For deployments that rely on SEED ciphers, Red Hat recommends enabling support for other cipher suites. This way, you ensure smooth transitions when NSS will remove support for them.

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

(BZ#1817533)

TLS 1.0 and TLS 1.1 are deprecated

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

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

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

(BZ#1660839)

DSA is deprecated in RHEL 8

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

(BZ#1646541)

SSL2 Client Hello has been deprecated in NSS

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

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

(BZ#1645153)

TPM 1.2 is deprecated

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

(BZ#1657927)

10.4. NETWORKING

Network scripts are deprecated in RHEL 8

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

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

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

```
~]# yum install network-scripts
```

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

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

(BZ#1647725)

10.5. KERNEL

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

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

(BZ#1748980)

The qla3xxx driver is deprecated

The **qla3xxx** driver has been deprecated in RHEL 8. The driver will likely not be supported in future major releases of this product, and thus it is not recommended for new deployments.

(BZ#1658840)

The dl2k, dnet, ethoc, and dlci drivers are deprecated

The **dl2k**, **dnet**, **ethoc**, and **dlci** drivers have been deprecated in RHEL 8. The drivers will likely not be supported in future major releases of this product, and thus they are not recommended for new deployments.

(BZ#1660627)

10.6. FILE SYSTEMS AND STORAGE

The elevator kernel command line parameter is deprecated

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

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

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

For more information, see Setting the disk scheduler.

(BZ#1665295)

LVM mirror is deprecated

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

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

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

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

(BZ#1827628)

NFSv3 over UDP has been disabled

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

NFS over UDP is no longer supported in RHEL 8.

(BZ#1592011)

10.7. DESKTOP

The libgnome-keyring library has been deprecated

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

(BZ#1607766)

10.8. GRAPHICS INFRASTRUCTURES

AGP graphics cards are no longer supported

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

(BZ#1569610)

10.9. THE WEB CONSOLE

The web console no longer supports incomplete translations

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

(BZ#1666722)

10.10. VIRTUALIZATION

virt-manager has been deprecated

The Virtual Machine Manager application, also known as **virt-manager**, has been deprecated. The RHEL 8 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 the RHEL 8 web console.

(JIRA:RHELPLAN-10304)

Virtual machine snapshots are not properly supported in RHEL 8

The current mechanism of creating virtual machine (VM) snapshots has been deprecated, as it is not working reliably. As a consequence, it is recommended not to use VM snapshots in RHEL 8.

Note that a new VM snapshot mechanism is under development and will be fully implemented in a future minor release of RHEL 8.

(BZ#1686057)

The Cirrus VGA virtual GPU type has been deprecated

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

(BZ#1651994)

The cpu64-rhel6 CPU model has been deprecated and removed

The **cpu64-rhel6** QEMU virtual CPU model has been deprecated in RHEL 8.1, and has been removed from RHEL 8.2. It is recommended that you use the other CPU models provided by QEMU and libvirt, according to the CPU present on the host machine.

(BZ#1741346)

10.11. DEPRECATED PACKAGES

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

- 389-ds-base-legacy-tools
- authd
- custodia
- hostname
- libidn
- net-tools
- network-scripts
- nss-pam-ldapd
- sendmail
- yp-tools
- ypbind
- ypserv

CHAPTER 11. KNOWN ISSUES

This part describes known issues in Red Hat Enterprise Linux 8.2.

11.1. INSTALLER AND IMAGE CREATION

The auth and authconfig Kickstart commands require the AppStream repository

The **authselect-compat** package is required by the **auth** and **authconfig** Kickstart commands during installation. Without this package, the installation fails if **auth** or **authconfig** are used. However, by design, the **authselect-compat** package is only available in the AppStream repository.

To work around this problem, verify that the BaseOS and AppStream repositories are available to the installer or use the **authselect** Kickstart command during installation.

(BZ#1640697)

The reboot --kexec and inst.kexec commands do not provide a predictable system state

Performing a RHEL installation with the **reboot --kexec** Kickstart command or the **inst.kexec** kernel boot parameters do not provide the same predictable system state as a full reboot. As a consequence, switching to the installed system without rebooting can produce unpredictable results.

Note that the **kexec** feature is deprecated and will be removed in a future release of Red Hat Enterprise Linux.

(BZ#1697896)

Anaconda installation includes low limits of minimal resources setting requirements

Anaconda initiates the installation on systems with minimal resource settings required available and do not provide previous message warning about the required resources for performing the installation successfully. As a result, the installation can fail and the output errors do not provide clear messages for possible debug and recovery. To work around this problem, make sure that the system has the minimal resources settings required for installation: 2GB memory on PPC64(LE) and 1GB on x86_64. As a result, it should be possible to perform a successful installation.

(BZ#1696609)

Installation fails when using the reboot --kexec command

The RHEL 8 installation fails when using a Kickstart file that contains the **reboot --kexec** command. To avoid the problem, use the **reboot** command instead of **reboot --kexec** in your Kickstart file.

(BZ#1672405)

RHEL 8 initial setup cannot be performed via SSH

Currently, the RHEL 8 initial setup interface does not display when logged in to the system using SSH. As a consequence, it is impossible to perform the initial setup on a RHEL 8 machine managed via SSH. To work around this problem, perform the initial setup in the main system console (ttyS0) and, afterwards, log in using SSH.

(BZ#1676439)

Network access is not enabled by default in the installation program

Several installation features require network access, for example, registration of a system using the

Content Delivery Network (CDN), NTP server support, and network installation sources. However, network access is not enabled by default, and as a result, these features cannot be used until network access is enabled.

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

(BZ#1757877)

Registration fails for user accounts that belong to multiple organizations

Currently, when you attempt to register a system with a user account that belongs to multiple organizations, the registration process fails with the error message **You must specify an organization for new units**.

To work around this problem, you can either:

- Use a different user account that does not belong to multiple organizations.
- Use the **Activation Key** authentication method available in the Connect to Red Hat feature for GUI and Kickstart installations.
- Skip the registration step in Connect to Red Hat and use Subscription Manager to register your system post-installation.

(BZ#1822880)

A GUI installation using the Binary DVD ISO image can sometimes not proceed without CDN registration

When performing a GUI installation using the Binary DVD ISO image file, a race condition in the installer can sometimes prevent the installation from proceeding until you register the system using the Connect to Red Hat feature. To work around this problem, complete the following steps:

1. Select **Installation Source** from the **Installation Summary** window of the GUI installation.
2. Verify that **Auto-detected installation media** is selected.
3. Click **Done** to confirm the selection and return to the **Installation Summary** window.
4. Verify that **Local Media** is displayed as the **Installation Source** status in the **Installation Summary** window.

As a result, you can proceed with the installation without registering the system using the Connect to Red Hat feature.

(BZ#1823578)

Copying the content of the Binary DVD.iso file to a partition omits the .treeinfo and .discinfo files

During local installation, while copying the content of the RHEL 8 Binary DVD.iso image file to a partition, the * in the **cp <path>/* <mounted partition>/dir** command fails to copy the **.treeinfo** and **.discinfo** files. These files are required for a successful installation. As a result, the BaseOS and AppStream repositories are not loaded, and a debug-related log message in the **anaconda.log** file is the only record of the problem.

To work around the problem, copy the missing **.treeinfo** and **.discinfo** files to the partition.

(BZ#1687747)

Self-signed HTTPS server cannot be used in Kickstart installation

Currently, the installer fails to install from a self-signed https server when the installation source is specified in the kickstart file and the **--noverifyssl** option is used:

```
url --url=https://SERVER/PATH --noverifyssl
```

To work around this problem, append the **inst.noverifyssl** parameter to the kernel command line when starting the kickstart installation.

For example:

```
inst.ks=<URL> inst.noverifyssl
```

(BZ#1745064)

GUI installation might fail if an attempt to unregister using the CDN is made before the repository refresh is completed

In RHEL 8.2, when registering your system and attaching subscriptions using the Content Delivery Network (CDN), a refresh of the repository metadata is started by the GUI installation program. The refresh process is not part of the registration and subscription process, and as a consequence, the **Unregister** button is enabled in the **Connect to Red Hat** window. Depending on the network connection, the refresh process might take more than a minute to complete. If you click the **Unregister** button before the refresh process is completed, the GUI installation might fail as the unregister process removes the CDN repository files and the certificates required by the installation program to communicate with the CDN.

To work around this problem, complete the following steps in the GUI installation after you have clicked the **Register** button in the **Connect to Red Hat** window:

1. From the **Connect to Red Hat** window, click **Done** to return to the **Installation Summary** window.
2. From the **Installation Summary** window, verify that the **Installation Source** and **Software Selection** status messages in italics are not displaying any processing information.
3. When the Installation Source and Software Selection categories are ready, click **Connect to Red Hat**.
4. Click the **Unregister** button.

After performing these steps, you can safely unregister the system during the GUI installation.

(BZ#1821192)

11.2. SUBSCRIPTION MANAGEMENT

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

In Red Hat Enterprise Linux 8, four attributes of the **syspurpose** command-line tool have been added: **role**,**usage**, **service_level_agreement** and **addons**. Currently, only **role**, **usage** and

service_level_agreement affect the output of running the **subscription-manager attach --auto** command. Users who attempt to set values to the **addons** argument will not observe any effect on the subscriptions that are auto-attached.

(BZ#1687900)

11.3. SHELLS AND COMMAND-LINE TOOLS

Applications using Wayland protocol cannot be forwarded to remote display servers

In Red Hat Enterprise Linux 8, most applications use the Wayland protocol by default instead of the X11 protocol. As a consequence, the ssh server cannot forward the applications that use the Wayland protocol but is able to forward the applications that use the X11 protocol to a remote display server.

To work around this problem, set the environment variable **GDK_BACKEND=x11** before starting the applications. As a result, the application can be forwarded to remote display servers.

(BZ#1686892)

systemd-resolved.service fails to start on boot

The **systemd-resolved** service occasionally fails to start on boot. If this happens, restart the service manually after the boot finishes by using the following command:

```
# systemctl start systemd-resolved
```

However, the failure of **systemd-resolved** on boot does not impact any other services.

(BZ#1640802)

11.4. SECURITY

Audit executable watches on symlinks do not work

File monitoring provided by the **-w** option cannot directly track a path. It has to resolve the path to a device and an inode to make a comparison with the executed program. A watch monitoring an executable symlink monitors the device and an inode of the symlink itself instead of the program executed in memory, which is found from the resolution of the symlink. Even if the watch resolves the symlink to get the resulting executable program, the rule triggers on any multi-call binary called from a different symlink. This results in flooding logs with false positives. Consequently, Audit executable watches on symlinks do not work.

To work around the problem, set up a watch for the resolved path of the program executable, and filter the resulting log messages using the last component listed in the **comm=** or **proctitle=** fields.

(BZ#1846345)

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

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

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

(JIRA:RHELPLAN-34199)

libselinux-python is available only through its module

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

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

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

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

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

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

(BZ#1666328)

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

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

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

(BZ#1763210)

Removing the rpm-plugin-selinux package leads to removing all selinux-policy packages from the system

Removing the **rpm-plugin-selinux** package disables SELinux on the machine. It also removes all **selinux-policy** packages from the system. Repeated installation of the **rpm-plugin-selinux** package then installs the **selinux-policy-minimum** SELinux policy, even if the **selinux-policy-targeted** policy was previously present on the system. However, the repeated installation does not update the SELinux configuration file to account for the change in policy. As a consequence, SELinux is disabled even upon reinstallation of the **rpm-plugin-selinux** package.

To work around this problem:

1. Enter the **umount /sys/fs/selinux/** command.
2. Manually install the missing **selinux-policy-targeted** package.
3. Edit the **/etc/selinux/config** file so that the policy is equal to **SELINUX=enforcing**.
4. Enter the command **load_policy -i**.

As a result, SELinux is enabled and running the same policy as before.

(BZ#1641631)

SELinux prevents systemd-journal-gatewayd to call newfstatat() on shared memory files created by corosync

SELinux policy does not contain a rule that allows the **systemd-journal-gatewayd** daemon to access files created by the **corosync** service. As a consequence, SELinux denies **systemd-journal-gatewayd** to call the **newfstatat()** function on shared memory files created by **corosync**.

To work around this problem, create a local policy module with an allow rule which enables the described scenario. See the **audit2allow(1)** man page for more information on generating SELinux policy *allow* and *dontaudit* rules. As a result of the previous workaround, **systemd-journal-gatewayd** can call the function on shared memory files created by **corosync** with SELinux in enforcing mode.

(BZ#1746398)

SELinux prevents auditd to halt or power off the system

The SELinux policy does not contain a rule that allows the Audit daemon to start a **power_unit_file_t systemd** unit. Consequently, **auditd** cannot halt or power off the system even when configured to do so in cases such as no space left on a logging disk partition.

To work around this problem, create a custom SELinux policy module. As a result, **auditd** can properly halt or power off the system only if you apply the workaround.

(BZ#1826788)

users can run sudo commands as locked users

In systems where **sudoers** permissions are defined with the **ALL** keyword, **sudo** users with permissions can run **sudo** commands as users whose accounts are locked. Consequently, locked and expired accounts can still be used to execute commands.

To work around this problem, enable the newly implemented **runas_check_shell** option together with proper settings of valid shells in **/etc/shells**. This prevents attackers from running commands under system accounts such as **bin**.

(BZ#1786990)

Negative effects of the default logging setup on performance

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

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

(JIRA:RHELPLAN-10431)

Parameter not known errors in the rsyslog output with config.enabled

In the **rsyslog** output, an unexpected bug occurs in configuration processing errors using the **config.enabled** directive. As a consequence, **parameter not known** errors are displayed while using the **config.enabled** directive except for the **include()** statements.

To work around this problem, set **config.enabled=on** or use **include()** statements.

(BZ#1659383)

Certain rsyslog priority strings do not work correctly

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

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

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

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

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

(BZ#1679512)

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

SHA-1 signatures in certificates are rejected by the GnuTLS secure communications library as insecure. Consequently, applications that use GnuTLS as a TLS backend cannot establish a TLS connection to peers that offer such certificates. This behavior is inconsistent with other system cryptographic libraries. To work around this problem, upgrade the server to use certificates signed with SHA-256 or stronger hash, or switch to the LEGACY policy.

(BZ#1628553)

TLS 1.3 does not work in NSS in FIPS mode

TLS 1.3 is not supported on systems working in FIPS mode. As a result, connections that require TLS 1.3 for interoperability do not function on a system working in FIPS mode.

To enable the connections, disable the system's FIPS mode or enable support for TLS 1.2 in the peer.

(BZ#1724250)

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)

OpenSSL generates a malformed **status_request** extension in the **CertificateRequest** message in TLS 1.3

OpenSSL servers send a malformed **status_request** extension in the **CertificateRequest** message if support for the **status_request** extension and client certificate-based authentication are enabled. In

such case, OpenSSL does not interoperate with implementations compliant with the **RFC 8446** protocol. As a result, clients that properly verify extensions in the **CertificateRequest** message abort connections with the OpenSSL server. To work around this problem, disable support for the TLS 1.3 protocol on either side of the connection or disable support for **status_request** on the OpenSSL server. This will prevent the server from sending malformed messages.

(BZ#1749068)

ssh-keyscan cannot retrieve RSA keys of servers in FIPS mode

The **SHA-1** algorithm is disabled for RSA signatures in FIPS mode, which prevents the **ssh-keyscan** utility from retrieving RSA keys of servers operating in that mode.

To work around this problem, use ECDSA keys instead, or retrieve the keys locally from the **/etc/ssh/ssh_host_rsa_key.pub** file on the server.

(BZ#1744108)

Libreswan does not work properly with seccomp=enabled on all configurations

The set of allowed syscalls in the **Libreswan** SECCOMP support implementation is currently not complete. Consequently, when SECCOMP is enabled in the **ipsec.conf** file, the syscall filtering rejects even syscalls needed for the proper functioning of the **pluto** daemon; the daemon is killed, and the **ipsec** service is restarted.

To work around this problem, set the **seccomp=** option back to the **disabled** state. SECCOMP support must remain disabled to run **ipsec** properly.

(BZ#1777474)

Certain sets of interdependent rules in SSG can fail

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

(BZ#1750755)

SCAP Workbench fails to generate results-based remediations from tailored profiles

The following error occurs when trying to generate results-based remediation roles from a customized profile using the **SCAP Workbench** tool:

```
Error generating remediation role .../remediation.sh: Exit code of oscap was 1: [output truncated]
```

To work around this problem, use the **oscap** command with the **--tailoring-file** option.

(BZ#1640715)

Kickstart uses org_fedora_oscap instead of com_redhat_oscap in RHEL 8

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

(BZ#1665082)

OSCAP Anaconda Addon does not install all packages in text mode

The **OSCAP Anaconda Addon** plugin cannot modify the list of packages selected for installation by the system installer if the installation is running in text mode. Consequently, when a security policy profile is specified using Kickstart and the installation is running in text mode, any additional packages required by the security policy are not installed during installation.

To work around this problem, either run the installation in graphical mode or specify all packages that are required by the security policy profile in the security policy in the **%packages** section in your Kickstart file.

As a result, packages that are required by the security policy profile are not installed during RHEL installation without one of the described workarounds, and the installed system is not compliant with the given security policy profile.

(BZ#1674001)

OSCAP Anaconda Addon does not correctly handle customized profiles

The **OSCAP Anaconda Addon** plugin does not properly handle security profiles with customizations in separate files. Consequently, the customized profile is not available in the RHEL graphical installation even when you properly specify it in the corresponding Kickstart section.

To work around this problem, follow the instructions in the Creating a single SCAP data stream from an original DS and a tailoring file Knowledgebase article. As a result of this workaround, you can use a customized SCAP profile in the RHEL graphical installation.

(BZ#1691305)

GnuTLS fails to resume current session with the NSS server

When resuming a TLS (Transport Layer Security) 1.3 session, the **GnuTLS** client waits 60 milliseconds plus an estimated round trip time for the server to send session resumption data. If the server does not send the resumption data within this time, the client creates a new session instead of resuming the current session. This incurs no serious adverse effects except for a minor performance impact on a regular session negotiation.

(BZ#1677754)

The oscap-ssh utility fails when scanning a remote system with --sudo

When performing a Security Content Automation Protocol (SCAP) scan of a remote system using the **oscap-ssh** tool with the **--sudo** option, the **oscap** tool on the remote system saves scan result files and report files into a temporary directory as the **root** user. If the **umask** settings on the remote machine have been changed, **oscap-ssh** might not have access to these files. To work around this problem, modify the **oscap-ssh** tool as described in this solution "oscap-ssh --sudo" fails to retrieve the result files with "scp: ...: Permission denied" error. As a result, **oscap** saves the files as the target user, and **oscap-ssh** accesses the files normally.

(BZ#1803116)

OpenSCAP produces false positives caused by removing blank lines from YAML multi-line strings

When OpenSCAP generates Ansible remediations from a datastream, it removes blank lines from YAML multi-line strings. Because some Ansible remediations contain literal configuration file content, removing blank lines affects the corresponding remediations. This causes the **openscap** utility to fail the corresponding Open Vulnerability and Assessment Language (OVAL) checks, even though the blank

lines do not have any effect. To work around this problem, check the rule descriptions and skip scan results that failed because of missing blank lines. Alternatively, use Bash remediations instead of Ansible remediations, because Bash remediations do not produce these false positive results.

(BZ#1795563)

OSPP-based profiles are incompatible with GUI package groups.

GNOME packages installed by the *Server with GUI* package group require the **nfs-utils** package that is not compliant with the Operating System Protection Profile (OSPP). As a consequence, selecting the *Server with GUI* package group during the installation of a system with OSPP or OSPP-based profiles, for example, Security Technical Implementation Guide (STIG), aborts the installation. If the OSPP-based profile is applied after the installation, the system is not bootable. To work around this problem, do not install the *Server with GUI* package group or any other groups that install GUI when using the OSPP profile and OSPP-based profiles. When you use the *Server* or *Minimal Install* package groups instead, the system installs without issues and works correctly.

(BZ#1787156)

RHEL8 system with the *Server with GUI* package group cannot be remediated using the e8 profile

Using the OpenSCAP Anaconda Add-on to harden the system on the *Server With GUI* package group with profiles that select rules from the *Verify Integrity with RPM* group requires an extreme amount of RAM on the system. This problem is caused by the OpenSCAP scanner; for more details see Scanning large numbers of files with OpenSCAP causes systems to run out of memory. As a consequence, the hardening of the system using the RHEL8 Essential Eight (e8) profile is not successful. To work around this problem, choose a smaller package group, for example, Server, and install additional packages that you require after the installation. As a result, the system will have a smaller number of packages, the scanning will require less memory, and therefore the system can be hardened automatically.

(BZ#1816199)

Scanning large numbers of files with OpenSCAP causes systems to run out of memory

The OpenSCAP scanner stores all the collected results in the memory until the scan finishes. As a consequence, the system might run out of memory on systems with low RAM when scanning large numbers of files, for example from the large package groups *Server with GUI* and *Workstation*. To work around this problem, use smaller package groups, for example, *Server* and *Minimal Install* on systems with limited RAM. If you need to use large package groups, you can test whether your system has sufficient memory in a virtual or staging environment. Alternatively, you can tailor the scanning profile to deselect rules that involve recursion over the entire / filesystem:

- **rpm_verify_hashes**
- **rpm_verify_permissions**
- **rpm_verify_ownership**
- **file_permissions_unauthorized_world_writable**
- **no_files_unowned_by_user**
- **dir_perms_world_writable_system_owned**
- **file_permissions_unauthorized_suid**
- **file_permissions_unauthorized_sgid**

- **file_permissions_ungroupowned**
- **dir_perms_world_writable_sticky_bits**

This will prevent OpenSCAP scan from causing the system to run out of memory.

(BZ#1824152)

11.5. NETWORKING

IPsec network traffic fails during IPsec offloading when GRO is disabled

IPsec offloading is not expected to work when Generic Receive Offload (GRO) is disabled on the device. If IPsec offloading is configured on a network interface and GRO is disabled on that device, IPsec network traffic fails.

To work around this problem, keep GRO enabled on the device.

(BZ#1649647)

iptables does not request module loading for commands that update a chain if the specified chain type is not known

Note: This problem causes spurious errors with no functional implication when stopping the **iptables systemd** service if you are using the services default configuration.

When setting a chain's policy with **iptables-nft**, the resulting update chain command sent to the kernel will fail if the associated kernel module is not loaded already. To work around the problem, use the following commands to cause the modules to load:

+

```
# iptables -t nat -n -L
# iptables -t mangle -n -L
```

(BZ#1812666)

Automatic loading of address family-specific LOG back end modules by the nft_compat module can hang

When the **nft_compat** module loads address family-specific **LOG** target back ends while an operation on network namespaces (**netns**) happens in parallel, a lock collision can occur. As a consequence, loading the address family-specific **LOG** target back ends can hang. To work around the problem, manually load the relevant **LOG** target back ends, such as **nf_log_ipv4.ko** and **nf_log_ipv6.ko**, before executing the **iptables-restore** utility. As a result, loading the **LOG** target back ends does not hang. However, if the problem appears during the system boots, no workaround is available.

Note that other services, such as **libvirtd**, also execute **iptables** commands, which can cause the problem to occur.

(BZ#1757933)

11.6. KERNEL

Accidental patch removal causes huge_page_setup_helper.py to show error

A patch that updates the **huge_page_setup_helper.py** script, was accidentally removed. Consequently, after executing the **huge_page_setup_helper.py** script, the following error message appears:

```
SyntaxError: Missing parentheses in call to 'print'
```

To work around this problem, copy the **huge_page_setup_helper.py** script from RHEL 8.1 and install it to the **/usr/bin/** directory:

1. Download the **libhugetlbfs-utils-2.21-3.el8.x86_64.rpm** package from the RHEL-8.1.0 Installation Media or from the Red Hat Customer Portal .
2. Execute the **rpm2cpio** command:

   ```
   # rpm2cpio libhugetlbfs-utils-2.21-3.el8.x86_64.rpm | cpio -D / -iduv
   '*/huge_page_setup_helper.py'
   ```

   The command extracts the **huge_page_setup_helper.py** script from the RHEL 8.1 RPM and saves it to the **/usr/bin/** directory.

As a result, the **huge_page_setup_helper.py** script works correctly.

(BZ#1823398)

Systems with a large amount of persistent memory experience delays during the boot process

Systems with a large amount of persistent memory take a long time to boot because the initialization of the memory is serialized. Consequently, if there are persistent memory file systems listed in the **/etc/fstab** file, the system might timeout while waiting for devices to become available. To work around this problem, configure the **DefaultTimeoutStartSec** option in the **/etc/systemd/system.conf** file to a sufficiently large value.

(BZ#1666538)

KSM sometimes ignores NUMA memory policies

When the kernel shared memory (KSM) feature is enabled with the **merge_across_nodes=1** parameter, KSM ignores memory policies set by the mbind() function, and may merge pages from some memory areas to Non-Uniform Memory Access (NUMA) nodes that do not match the policies.

To work around this problem, disable KSM or set the **merge_across_nodes** parameter to **0** if using NUMA memory binding with QEMU. As a result, NUMA memory policies configured for the KVM VM will work as expected.

(BZ#1153521)

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

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

(BZ#1659609)

zlib may slow down a vmcore capture in some compression functions

The **kdump** configuration file uses the **lzo** compression format (**makedumpfile -l**) by default. When you modify the configuration file using the **zlib** compression format, (**makedumpfile -c**) it is likely to bring a better compression factor at the expense of slowing down the **vmcore** capture process. As a consequence, it takes the **kdump** upto four times longer to capture a **vmcore** with **zlib**, as compared to **lzo**.

As a result, Red Hat recommends using the default **lzo** for cases where speed is the main driving factor. However, if the target machine is low on available space, **zlib** is a better option.

(BZ#1790635)

A vmcore capture fails after memory hot-plug or unplug operation

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

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

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

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

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

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

(BZ#1793389)

The fadump dumping mechanism renames the network interface to kdump-<interface-name>

When using firmware-assisted dump (**fadump**) to capture a **vmcore** and store it to a remote machine using SSH or NFS protocol, renames the network interface to **kdump-<interface-name>**. The renaming happens when the **<interface-name>** is generic, for example, *eth#, or net# and so on. This problem occurs because the **vmcore** capture scripts in the initial RAM disk (**initrd**) add the kdump- prefix to the network interface name to secure persistent naming. Since the same **initrd** is also used for a regular boot, the interface name is changed for the production kernel too.

(BZ#1745507)

The system enters the emergency mode at boot-time when fadump is enabled

The system enters the emergency mode when **fadump** (**kdump**) or **dracut** squash module is enabled in the **initramfs** scheme because **systemd** manager fails to fetch the mount information and configure the LV partition to mount. To work around this problem, add the following kernel command line parameter **rd.lvm.lv=<VG>/<LV>** to discover and mount the failed LV partition appropriately. As a result, the system will boot successfully in the described scenario.

(BZ#1750278)

Using irqpoll causes vmcore generation failure

Due to an existing problem with the **nvme** driver on the 64-bit ARM architectures that run on the Amazon Web Services (AWS) cloud platforms, the **vmcore** generation fails when you provide the **irqpoll** kernel command line parameter to the first kernel. Consequently, no **vmcore** file is dumped in the

/var/crash/ directory after a kernel crash. To work around this problem:

1. Add **irqpoll** to the **KDUMP_COMMANDLINE_REMOVE** key in the **/etc/sysconfig/kdump** file.
2. Restart the **kdump** service by running the **systemctl restart kdump** command.

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

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

(BZ#1654962)

Using vPMEM memory as dump target delays the kernel crash capture process

When you use Virtual Persistent Memory (vPEM) namespaces as **kdump** or **fadump** target, the **papr_scm** module is forced to unmap and remap the memory backed by vPMEM and re-add the memory to its linear map. Consequently, this behavior triggers Hypervisor Calls (HCalls) to the POWER Hypervisor, and the total time taken, slows the capture kernel boot considerably. Therefore, it is recommended not to use vPMEM namespaces as a dump target for kdump or fadump.

If you must use vPMEM, to work around this problem execute the following commands:

1. Create the **/etc/dracut.conf.d/99-pmem-workaround.conf** file and add:

```
add_drivers+="nd_pmem nd_btt libnvdimm papr_scm"
```

2. Rebuild the initial RAM disk (initrd) file system:

```
# touch /etc/kdump.conf
# systemctl restart kdump.service
```

(BZ#1792125)

The HP NMI watchdog does not always generate a crash dump

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

The missing NMI is initiated by one of two conditions:

1. The **Generate NMI** button on the Integrated Lights-Out (iLO) server management software. This button is triggered by a user.
2. The **hpwdt** watchdog. The expiration by default sends an NMI to the server.

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

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

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

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

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

(BZ#1602962)

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

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

(BZ#1609288)

The cxgb4 driver causes crash in the kdump kernel

The **kdump** kernel crashes while trying to save information in the **vmcore** file. Consequently, the **cxgb4** driver prevents the **kdump** kernel from saving a core for later analysis. To work around this problem, add the **novmcoredd** parameter to the kdump kernel command line to allow saving core files.

(BZ#1708456)

Attempting to add ICE driver NIC port to a mode 5 (balance-tlb) bonding master interface might lead to failure

Attempting to add **ICE** driver NIC port to a mode 5 (balance-tlb) bonding master interface might lead to a failure with an error **Master 'bond0', Slave 'ens1f0': Error: Enslave failed**. Consequently, you experience an intermittent failure to add the NIC port to the bonding master interface. To workaround this problem, attempt to retry adding the interface.

(BZ#1791664)

Attaching the Virtual Function to virtual machine with interface type='hostdev' might fails at times

Attaching a Virtual Function (VF) to a virtual machine using an .XML file, following the **Assignment with <interface type='hostdev'>** method, might fail at times. This occurs because using the **Assignment with <interface type='hostdev'>** method prevents the VM from attaching to the VF NIC presented to this virtual machine. To workaround this problem, attach the VF to the VM using the .XML file using the **Assignment with <hostdev>** method. As a result, the **virsh attach-device** command succeeds without error. For more details about the difference between **Assignment with <hostdev>** and **Assignment with <interface type='hostdev'>** (SRIOV devices only), see PCI Passthrough of host network devices .

(BZ#1792691)

11.7. FILE SYSTEMS AND STORAGE

The /boot file system cannot be placed on LVM

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

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

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

(BZ#1496229)

LVM no longer allows creating volume groups with mixed block sizes

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

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

(BZ#1768536)

DM Multipath might fail to start when too many LUNs are connected

The **multipathd** service might time out and fail to start if too many logical units (LUNs) are connected to the system. The exact number of LUNs that causes the problem depends on several factors, including the number of devices, the response time of the storage array, the memory and CPU configuration, and system load.

To work around the problem, increase the timeout value in the **multipathd** unit file:

1. Open the **multipathd** unit in the unit editor:

```
# systemctl edit multipathd
```

2. Enter the following configuration to override the timeout value:

```
[Service]
TimeoutSec=300
```

   Red Hat recommends increasing the value to 300 from the default 90, but you can also test other values above 90.

3. Save the file in the editor.
4. Reload **systemd** units to apply the change:

```
# systemctl daemon-reload
```

As a result, **multipathd** can now successfully start with a larger number of LUNs.

(BZ#1797660)

Limitations of LVM writecache

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

- You cannot take a snapshot of a logical volume while the logical volume is using **writecache**.
- You cannot attach or detach **writecache** while a logical volume is active.
- When attaching **writecache** to an inactive logical volume, you must use a **writecache** block size that matches the existing file system block size.
  For details, see the **lvmcache(7)** man page.
- You cannot resize a logical volume while **writecache** is attached to it.
- You cannot use **pvmove** commands on devices that are used with **writecache**.
- You cannot use logical volumes with **writecache** in combination with thin pools or VDO.

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

LVM mirror devices that store a LUKS volume sometimes become unresponsive

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

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

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

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

(BZ#1730502)

An NFS 4.0 patch can result in reduced performance under an open-heavy workload

Previously, a bug was fixed that, in some cases, could cause an NFS open operation to overlook the fact that a file had been removed or renamed on the server. However, the fix may cause slower performance with workloads that require many open operations. To work around this problem, it might help to use NFS version 4.1 or higher, which have been improved to grant delegations to clients in more cases, allowing clients to perform open operations locally, quickly, and safely.

(BZ#1748451)

11.8. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

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

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

(BZ#1803161)

nginx cannot load server certificates from hardware security tokens

The **nginx** web server supports loading TLS private keys from hardware security tokens directly from PKCS#11 modules. However, it is currently impossible to load server certificates from hardware security tokens through the PKCS#11 URI. To work around this problem, store server certificates on the file system

(BZ#1668717)

php-fpm causes SELinux AVC denials when php-opcache is installed with PHP 7.2

When the **php-opcache** package is installed, the FastCGI Process Manager (**php-fpm**) causes SELinux AVC denials. To work around this problem, change the default configuration in the **/etc/php.d/10-opcache.ini** file to the following:

```
opcache.huge_code_pages=0
```

Note that this problem affects only the **php:7.2** stream, not the **php:7.3** one.

(BZ#1670386)

The mod_wsgi package name is missing when being installed as a dependency

With a change in **mod_wsgi** installation, described in BZ#1779705, the **python3-mod_wsgi** package no longer provides the name **mod_wsgi**. When installing the **mod_wsgi** module, you must specify the full package name. This change causes problems with dependencies of third-party packages.

If you try to install a third-party package that requires a dependency named **mod_wsgi**, an error similar to the following is returned:

```
Error:
 Problem: conflicting requests
  - nothing provides mod_wsgi needed by package-requires-mod_wsgi.el8.noarch
```

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

a. Rebuild the package (or ask the third-party vendor for a new build) to require the full package name **python3-mod_wsgi**.

b. Create a meta package with the missing package name:

   1. Build your own empty meta package that provides the name **mod_wsgi**.

   2. Add the **module_hotfixes=True** line to the **.repo** configuration file of the repository that includes the meta package.

   3. Manually install **python3-mod_wsgi**.

(BZ#1829692)

11.9. COMPILERS AND DEVELOPMENT TOOLS

Synthetic functions generated by GCC confuse SystemTap

GCC optimization can generate synthetic functions for partially inlined copies of other functions. Tools

such as SystemTap and GDB cannot distinguish these synthetic functions from real functions. As a consequence, SystemTap places probes on both synthetic and real function entry points and, thus, registers multiple probe hits for a single real function call.

To work around this problem, modify SystemTap scripts to detect recursion and prevent placing of probes related to inlined partial functions.

This example script

```
probe kernel.function("can_nice").call { }
```

can be modified this way:

```
global in_can_nice%

probe kernel.function("can_nice").call {
  in_can_nice[tid()] ++;
  if (in_can_nice[tid()] > 1) { next }
  /* code for real probe handler */
}

probe kernel.function("can_nice").return {
  in_can_nice[tid()] --;
}
```

Note that this example script does not consider all possible scenarios, such as missed kprobes or kretprobes, or genuine intended recursion.

(BZ#1169184)

11.10. IDENTITY MANAGEMENT

Changing /etc/nsswitch.conf requires a manual system reboot

Any change to the **/etc/nsswitch.conf** file, for example running the **authselect select profile_id** command, requires a system reboot so that all relevant processes use the updated version of the **/etc/nsswitch.conf** file. If a system reboot is not possible, restart the service that joins your system to Active Directory, which is the **System Security Services Daemon** (SSSD) or **winbind**.

(BZ#1657295)

SSSD returns incorrect LDAP group membership for local users when the files domain is enabled

If the System Security Services Daemon (SSSD) serves users from the local files and the **ldap_rfc2307_fallback_to_local_users** attribute in the [domain/LDAP] section of the **sssd.conf** file is set to True, then the files provider does not include group memberships from other domains. As a consequence, if a local user is a member of an LDAP group, the **id local_user** command does not return the user's LDAP group membership. To work around this problem, disable the implicit **files** domain by adding

```
enable_files_domain=False
```

to the **[sssd]** section in the **/etc/sssd/sssd.conf** file.

As a result, **id local_user** returns correct LDAP group membership for local users.

(BZ#1652562)

SSSD does not correctly handle multiple certificate matching rules with the same priority

If a given certificate matches multiple certificate matching rules with the same priority, the System Security Services Daemon (SSSD) uses only one of the rules. As a workaround, use a single certificate matching rule whose LDAP filter consists of the filters of the individual rules concatenated with the | (or) operator. For examples of certificate matching rules, see the sss-certamp(5) man page.

(BZ#1447945)

Private groups fail to be created with auto_private_group = hybrid when multiple domains are defined

Private groups fail to be created with the option auto_private_group = hybrid when multiple domains are defined and the hybrid option is used by any domain other than the first one. If an implicit files domain is defined along with an AD or LDAP domain in the **sssd.conf** file and is not marked as **MPG_HYBRID**, then SSSD fails to create a private group for a user who has uid=gid and the group with this gid does not exist in AD or LDAP.

The sssd_nss responder checks for the value of the **auto_private_groups** option in the first domain only. As a consequence, in setups where multiple domains are configured, which includes the default setup on RHEL 8, the option **auto_private_group** has no effect.

To work around this problem, set **enable_files_domain = false** in the sssd section of of **sssd.conf**. As a result, If the **enable_files_domain** option is set to false, then sssd does not add a domain with **id_provider=files** at the start of the list of active domains, and therefore this bug does not occur.

(BZ#1754871)

python-ply is not FIPS compatible

The YACC module of the **python-ply** package uses the MD5 hashing algorithm to generate the fingerprint of a YACC signature. However, FIPS mode blocks the use of MD5, which is only allowed in non-security contexts. As a consequence, python-ply is not FIPS compatible. On a system in FIPS mode, all calls to **ply.yacc.yacc()** fail with the error message:

```
UnboundLocalError: local variable 'sig' referenced before assignment
```

The problem affects **python-pycparser** and some use cases of **python-cffi**. To work around this problem, modify the line 2966 of the file **/usr/lib/python3.6/site-packages/ply/yacc.py**, replacing **sig = md5()** with **sig = md5(usedforsecurity=False)**. As a result, **python-ply** can be used in FIPS mode.

(BZ#1747490)

FreeRADIUS silently truncates Tunnel-Passwords longer than 249 characters

If a Tunnel-Password is longer than 249 characters, the FreeRADIUS service silently truncates it. This may lead to unexpected password incompatibilities with other systems.

To work around the problem, choose a password that is 249 characters or fewer.

(BZ#1723362)

Installing KRA fails if all KRA members are hidden replicas

The **ipa-kra-install** utility fails on a cluster where the Key Recovery Authority (KRA) is already present if the first KRA instance is installed on a hidden replica. Consequently, you cannot add further KRA instances to the cluster.

To work around this problem, unhide the hidden replica that has the KRA role before you add new KRA instances. You can hide it again when **ipa-kra-install** completes successfully.

(BZ#1816784)

Directory Server warns about missing attributes in the schema if those attributes are used in a search filter

If you set the **nsslapd-verify-filter-schema** parameter to **warn-invalid**, Directory Server processes search operations with attributes that are not defined in the schema and logs a warning. With this setting, Directory Server returns requested attributes in search results, regardless whether the attributes is defined in the schema or not.

A future version of Directory Server will change the default setting of **nsslapd-verify-filter-schema** to enforce stricter checks. The new default will warn about attributes that are missing in the schema, and reject requests or return only partial results.

(BZ#1790259)

ipa-healthcheck-0.4 does not obsolete older versions of ipa-healthcheck

The **Healthcheck** tool has been split into two sub-packages: **ipa-healthcheck** and **ipa-healthcheck-core**. However, only the **ipa-healthcheck-core** sub-package is correctly set to obsolete older versions of **ipa-healthcheck**. As a result, updating **Healthcheck** only installs **ipa-healthcheck-core** and the **ipa-healthcheck** command does not work after the update.

To work around this problem, install the **ipa-healthcheck-0.4** sub-package manually using **yum install ipa-healthcheck-0.4**.

(BZ#1852244)

11.11. DESKTOP

Limitations of the Wayland session

With Red Hat Enterprise Linux 8, the GNOME environment and the GNOME Display Manager (GDM) use **Wayland** as the default session type instead of the **X11** session, which was used with the previous major version of RHEL.

The following features are currently unavailable or do not work as expected under **Wayland**:

- **X11** configuration utilities, such as **xrandr**, do not work under **Wayland** due to its different approach to handling, resolutions, rotations, and layout. You can configure the display features using GNOME settings.
- Screen recording and remote desktop require applications to support the portal API on **Wayland**. Certain legacy applications do not support the portal API.
- Pointer accessibility is not available on **Wayland**.
- No clipboard manager is available.
- GNOME Shell on **Wayland** ignores keyboard grabs issued by most legacy **X11** applications. You

can enable an **X11** application to issue keyboard grabs using the **/org/gnome/mutter/wayland/xwayland-grab-access-rules** GSettings key. By default, GNOME Shell on **Wayland** enables the following applications to issue keyboard grabs:

  - **GNOME Boxes**
  - **Vinagre**
  - **Xephyr**
  - **virt-manager**, **virt-viewer**, and **remote-viewer**
  - **vncviewer**

- **Wayland** inside guest virtual machines (VMs) has stability and performance problems. RHEL automatically falls back to the **X11** session when running in a VM.

If you upgrade to RHEL 8 from a RHEL 7 system where you used the **X11** GNOME session, your system continues to use **X11**. The system also automatically falls back to **X11** when the following graphics drivers are in use:

- The proprietary NVIDIA driver
- The **cirrus** driver
- The **mga** driver
- The **aspeed** driver

You can disable the use of **Wayland** manually:

- To disable **Wayland** in GDM, set the **WaylandEnable=false** option in the **/etc/gdm/custom.conf** file.
- To disable **Wayland** in the GNOME session, select the legacy **X11** option by using the cogwheel menu on the login screen after entering your login name.

For more details on **Wayland**, see https://wayland.freedesktop.org/.

(BZ#1797409)

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

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

(BZ#1717947)

Disabling flatpak repositories from Software Repositories is not possible

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

(BZ#1668760)

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

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

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

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

(BZ#1583445)

System crash may result in fadump configuration loss

This issue is observed on systems where firmware-assisted dump (fadump) is enabled, and the boot partition is located on a journaling file system such as XFS. A system crash might cause the boot loader to load an older **initrd** that does not have the dump capturing support enabled. Consequently, after recovery, the system does not capture the **vmcore** file, which results in fadump configuration loss.

To work around this problem:

- If **/boot** is a separate partition, perform the following:
  1. Restart the kdump service
  2. Run the following commands as the root user, or using a user account with CAP_SYS_ADMIN rights:

```
# fsfreeze -f
# fsfreeze -u
```

- If **/boot** is not a separate partition, reboot the system.

(BZ#1723501)

11.12. GRAPHICS INFRASTRUCTURES

Unable to run graphical applications using sudo command

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

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

(BZ#1673073)

radeon fails to reset hardware correctly

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

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

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

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

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

(BZ#1694705)

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

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

(BZ#1812577)

11.13. THE WEB CONSOLE

Unprivileged users can access the Subscriptions page

If a non-administrator navigates to the **Subscriptions** page of the web console, the web console displays a generic error message **Cockpit had an unexpected internal error**.

To work around this problem, sign in to the web console with a privileged user and make sure to check the **Reuse my password for privileged tasks**checkbox.

(BZ#1674337)

11.14. VIRTUALIZATION

Low GUI display performance in RHEL 8 virtual machines on a Windows Server 2019 host

When using RHEL 8 as a guest operating system in graphical mode on a Windows Server 2019 host, the GUI display performance is low, and connecting to a console output of the guest currently takes significantly longer than expected.

This is a known issue on Windows 2019 hosts and is pending a fix by Microsoft. To work around this problem, connect to the guest using SSH or use Windows Server 2016 as the host.

(BZ#1706541)

Displaying multiple monitors of virtual machines that use Wayland is not possible with QXL

Using the **remote-viewer** utility to display more than one monitor of a virtual machine (VM) that is using the Wayland display server causes the VM to become unresponsive and the *Waiting for display* status message to be displayed indefinitely.

To work around this problem, use **virtio-gpu** instead of **qxl** as the GPU device for VMs that use Wayland.

(BZ#1642887)

virsh iface-* commands do not work consistently

Currently, **virsh iface-*** commands, such as **virsh iface-start** and **virsh iface-destroy**, frequently fail due to configuration dependencies. Therefore, it is recommended not to use **virsh iface-*** commands for configuring and managing host network connections. Instead, use the NetworkManager program and

its related management applications.

(BZ#1664592)

RHEL 8 virtual machines sometimes cannot boot on Witherspoon hosts

RHEL 8 virtual machines (VMs) that use the **pseries-rhel7.6.0-sxxm** machine type in some cases fail to boot on *Power9 S922LC for HPC* hosts (also known as Witherspoon) that use the DD2.2 or DD2.3 CPU.

Attempting to boot such a VM instead generates the following error message:

```
qemu-kvm: Requested safe indirect branch capability level not supported by kvm
```

To work around this problem, configure the virtual machine's XML configuration as follows:

```
<domain type='qemu' xmlns:qemu='http://libvirt.org/schemas/domain/qemu/1.0'>
  <qemu:commandline>
    <qemu:arg value='-machine'/>
    <qemu:arg value='cap-ibs=workaround'/>
  </qemu:commandline>
```

(BZ#1732726, BZ#1751054)

IBM POWER virtual machines do not work correctly with empty NUMA nodes

Currently, when an IBM POWER virtual machine (VM) running on a RHEL 8 host is configured with a NUMA node that uses zero memory (**memory='0'**) and zero CPUs, the VM cannot start. Therefore, Red Hat strongly recommends not using IBM POWER VMs with such empty NUMA nodes on RHEL 8.

(BZ#1651474)

SMT CPU topology is not detected by VMs when using host passthrough mode on AMD EPYC

When a virtual machine (VM) boots with the CPU host passthrough mode on an AMD EPYC host, the **TOPOEXT** CPU feature flag is not present. Consequently, the VM is not able to detect a virtual CPU topology with multiple threads per core. To work around this problem, boot the VM with the EPYC CPU model instead of host passthrough.

(BZ#1740002)

Disk identifiers in RHEL 8.2 VMs may change on VM reboot.

When using a virtual machine (VM) with RHEL 8.2 as the guest operating system on a Hyper-V hypervisor, the device identifiers for the VM's virtual disks in some cases change when the VM reboots. For example, a disk originally identified as **/dev/sda** may become **/dev/sdb**. As a consequence, the VM might fail to boot, and scripts that reference disks of the VM might stop working.

To avoid this issue, Red Hat strongly recommends to set persistent names for the disks in the VM. For detailed information, see the Microsoft Azure documentation: https://docs.microsoft.com/en-us/azure/virtual-machines/troubleshooting/troubleshoot-device-names-problems.

(BZ#1777283)

Virtual machines sometimes fail to start when using many virtio-blk disks

Adding a large number of virtio-blk devices to a virtual machine (VM) may exhaust the number of interrupt vectors available in the platform. If this occurs, the VM's guest OS fails to boot, and displays a **dracut-initqueue[392]: Warning: Could not boot** error.

(BZ#1719687)

Attaching LUN devices to virtual machines using virtio-blk does not work

The q35 machine type does not support transitional virtio 1.0 devices, and RHEL 8 therefore lacks support for features that were deprecated in virtio 1.0. In particular, it is not possible on a RHEL 8 host to send SCSI commands from virtio-blk devices. As a consequence, attaching a physical disk as a LUN device to a virtual machine fails when using the virtio-blk controller.

Note that physical disks can still be passed through to the guest operating system, but they should be configured with the **device='disk'** option rather than **device='lun'**.

(BZ#1777138)

11.15. SUPPORTABILITY

redhat-support-tool does not work with the **FUTURE** crypto policy

Because a cryptographic key used by a certificate on the Customer Portal API does not meet the requirements by the **FUTURE** system-wide cryptographic policy, the **redhat-support-tool** utility does not work with this policy level at the moment.

To work around this problem, use the **DEFAULT** crypto policy while connecting to the Customer Portal API.

(BZ#1802026)

11.16. CONTAINERS

UDICA is not expected to work with 1.0 stable stream

UDICA, the tool to generate SELinux policies for containers, is not expected to work with containers that are run via podman 1.0.x in the **container-tools:1.0** module stream.

(JIRA:RHELPLAN-25571)

Notes on FIPS support with Podman

The Federal Information Processing Standard (FIPS) requires certified modules to be used. Previously, Podman correctly installed certified modules in containers by enabling the proper flags at startup. However, in this release, Podman does not properly set up the additional application helpers normally provided by the system in the form of the FIPS system-wide crypto-policy. Although setting the system-wide crypto-policy is not required by the certified modules it does improve the ability of applications to use crypto modules in compliant ways. To work around this problem, change your container to run the **update-crypto-policies --set FIPS** command before any other application code is executed.

(BZ#1804193)

CHAPTER 12. INTERNATIONALIZATION

12.1. RED HAT ENTERPRISE LINUX 8 INTERNATIONAL LANGUAGES

Red Hat Enterprise Linux 8 supports the installation of multiple languages and the changing of languages based on your requirements.

- East Asian Languages - Japanese, Korean, Simplified Chinese, and Traditional Chinese.
- European Languages - English, German, Spanish, French, Italian, Portuguese, and Russian.

The following table lists the fonts and input methods provided for various major languages.

Language	Default Font (Font Package)	Input Methods
English	dejavu-sans-fonts	
French	dejavu-sans-fonts	
German	dejavu-sans-fonts	
Italian	dejavu-sans-fonts	
Russian	dejavu-sans-fonts	
Spanish	dejavu-sans-fonts	
Portuguese	dejavu-sans-fonts	
Simplified Chinese	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-libpinyin, libpinyin
Traditional Chinese	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-libzhuyin, libzhuyin
Japanese	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-kkc, libkkc
Korean	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-hangul, libhangu

12.2. NOTABLE CHANGES TO INTERNATIONALIZATION IN RHEL 8

RHEL 8 introduces the following changes to internationalization compared to RHEL 7:

- Support for the **Unicode 11** computing industry standard has been added.
- Internationalization is distributed in multiple packages, which allows for smaller footprint installations. For more information, see Using langpacks.

- The **glibc** package updates for multiple locales are now synchronized with the Common Locale Data Repository (CLDR).

APPENDIX A. LIST OF TICKETS BY COMPONENT

Bugzilla and JIRA IDs are listed in this document for reference. Bugzilla bugs that are publicly accessible include a link to the ticket.

Component	Tickets
389-ds-base	BZ#1715406, BZ#1748016, BZ#1790259, BZ#1748994, BZ#1739718
NetworkManager	BZ#1626348
anaconda	BZ#1747382, BZ#1637472, BZ#1748756, BZ#1649359, BZ#1715303, BZ#1696609, BZ#1672405, BZ#1687747, BZ#1745064, BZ#1659400, BZ#1821192, BZ#1822880, BZ#1823578, BZ#1748281, BZ#1746391
audit	BZ#1757986
authselect	BZ#1657295
bind	BZ#1564443, BZ#1664863, BZ#1704328
binutils	BZ#1777002, BZ#1618748
buildah-container	BZ#1627898
clevis	BZ#1766526, BZ#1564559, BZ#1436780, BZ#1784524
cloud-init	BZ#1641190, BZ#1666961
cockpit-appstream	BZ#1676506
cockpit	BZ#1678465, BZ#1754163, BZ#1666722
container-tools-rhel8-module	BZ#1784267
corosync-qdevice	BZ#1784200
createrepo_c	BZ#1743186
crypto-policies	BZ#1690565, BZ#1660839
device-mapper-multipath	BZ#1797660
dhcp	BZ#1729211
distribution	BZ#1657927

Component	Tickets
dnf	BZ#1676891, BZ#1754609
dnsmasq	BZ#1700916
edk2	BZ#1748180
elfutils	BZ#1744992
fapolicyd	BZ#1759895
fence-agents	BZ#1775847
firewalld	BZ#1737045, BZ#1740670, BZ#1733066
freeradius	BZ#1723362
gcc-toolset-9	BZ#1774118
gcc	BZ#1726641, BZ#1698607, BZ#1747157
gdb	BZ#1768593
gdm	BZ#1749960
glibc	BZ#1410154, BZ#1764214, BZ#1749439, BZ#1764235, BZ#1746928, BZ#1777241, BZ#1361965, BZ#1747502, BZ#1764218, BZ#1764238, BZ#1746933, BZ#1747453
gnome-shell-extensions	BZ#1717947
gnome-shell	BZ#1724302
gnome-software	BZ#1668760
gnutls	BZ#1628553, BZ#1677754
go-toolset	BZ#1747150
grafana-pcp	BZ#1685315
grafana	BZ#1725278
graphviz	BZ#1704875
grub2	BZ#1583445, BZ#1723501

Component	Tickets
httpd-2.4-module	BZ#1747923
httpd	BZ#1633224
initial-setup	BZ#1676439
ipa	BZ#1665051, BZ#1816784, BZ#1719767, BZ#1777564, BZ#1664719, BZ#1664718
ipcalc	BZ#1638834
java-11-openjdk	BZ#1746875
kernel-rt	BZ#1680161
kernel	BZ#1744397, BZ#1698297, BZ#1687094, BZ#1720227, BZ#1846345, BZ#1635295, BZ#1793389, BZ#1706541, BZ#1666538, BZ#1602962, BZ#1649647, BZ#1153521, BZ#1694705, BZ#1348508, BZ#1748451, BZ#1708456, BZ#1654962, BZ#1609288, BZ#1777283, BZ#1791664, BZ#1792125, BZ#1792691, BZ#1812666, BZ#1812577, BZ#1757933, BZ#1763661, BZ#1780432, BZ#1401552, BZ#1716002, BZ#1593711, BZ#1620349, BZ#1724969, BZ#1714330, BZ#1714486, BZ#1660368, BZ#1524687, BZ#1274406, BZ#1650518, BZ#1636572, BZ#1727369, BZ#1519039, BZ#1627455, BZ#1501618, BZ#1495358, BZ#1633143, BZ#1503672, BZ#1570255, BZ#1696451, BZ#1665295, BZ#1658840, BZ#1660627, BZ#1569610, BZ#1730502
kexec-tools	BZ#1750278, BZ#1690729
kmod-kvdo	BZ#1737639, BZ#1657301
krb5	BZ#1754690
libbpf	BZ#1759154
libdnf	BZ#1697472
libgnome-keyring	BZ#1607766
libndp	BZ#1697595
libpfm	BZ#1731019

Component	Tickets
libreswan	BZ#1777474
libselinux-python-2.8-module	BZ#1666328
libvirt	BZ#1749672, BZ#1664592, BZ#1528684
llvm-toolset	BZ#1747139
lorax	BZ#1754711
ltrace	BZ#1655368
lvm2	BZ#1600174, BZ#1496229, BZ#1768536
make	BZ#1774790
maven	BZ#1783926
mod_wsgi	BZ#1829692, BZ#1779705
mutter	BZ#1737553
nfs-utils	BZ#1719983, BZ#1592011
nftables	BZ#1778883, BZ#1643192
nginx	BZ#1668717
nmstate	BZ#1674456
nss_nis	BZ#1803161
nss	BZ#1724250, BZ#1817533, BZ#1645153
numactl	BZ#1730738
opencv	BZ#1694647
openscap	BZ#1636431, BZ#1618489, BZ#1646197, BZ#1803116, BZ#1795563, BZ#1824152, BZ#1642373
openssh	BZ#1744108

Component	Tickets
openssl-pkcs11	BZ#1705505, BZ#1664807, BZ#1745082
openssl	BZ#1685470, BZ#1749068
oscap-anaconda-addon	BZ#1665082, BZ#1674001, BZ#1691305, BZ#1787156, BZ#1816199
pacemaker	BZ#1712584, BZ#1700104
pam	BZ#1252859, BZ#1537242
pcp	BZ#1723598
pcs	BZ#1631519, BZ#1631514, BZ#1676431, BZ#1442116, BZ#1619620
perl-LDAP	BZ#1663063
php-7.2-module	BZ#1670386
php-pecl-xdebug	BZ#1769857
pki-core	BZ#1698084, BZ#1303254
podman	BZ#1804193, BZ#1645280
policycoreutils	BZ#1563742, BZ#1417455
postfix	BZ#1723950, BZ#1745321
powertop	BZ#1716721
pykickstart	BZ#1637872
python-ply	BZ#1747490
python38-3.8-module	BZ#1747329
qemu-kvm	BZ#1651474, BZ#1740002, BZ#1719687, BZ#1651994, BZ#1741346
rear	BZ#1729501
redhat-release	BZ#1817591
redhat-support-tool	BZ#1802026
rhel-system-roles-sap	BZ#1660832

Component	Tickets
rng-tools	BZ#1692435
rpm	BZ#1688849
rsyslog	JIRA:RHELPLAN-10431, BZ#1659383, BZ#1679512, BZ#1740683, BZ#1676559, BZ#1692073, BZ#1692072
rust-toolset	BZ#1776847
s390utils	BZ#1750326
samba	BZ#1754409, JIRA:RHELPLAN-13195
scap-security-guide	BZ#1755447, BZ#1754919, BZ#1750755, BZ#1755194
scap-workbench	BZ#1640715
selinux-policy	BZ#1641631, BZ#1746398, BZ#1826788, BZ#1727887, BZ#1726166, BZ#1726246
setools	BZ#1731519
setroubleshoot-plugins	BZ#1649842
setup	BZ#1730396, BZ#1663556
skopeo-container	BZ#1627900
skopeo	BZ#1810053
sscg	BZ#1717880
sssd	BZ#1669407, BZ#1652562, BZ#1447945, BZ#1754871
subscription-manager	BZ#1674337
sudo	BZ#1786990, BZ#1733961
systemd	BZ#1686892, BZ#1640802
systemtap	BZ#1744989
tpm2-tools	BZ#1725714
tuned	BZ#1738250

Component	Tickets
udica	BZ#1763210, BZ#1732704
vdo	BZ#1713749
virt-manager	BZ#1677019
wayland	BZ#1673073
whois	BZ#1734183
xorg-x11-drv-qxl	BZ#1642887
xorg-x11-server	BZ#1698565
zlib	BZ#1659433, BZ#1666798
other	BZ#1640697, BZ#1659609, BZ#1687900, BZ#1697896, BZ#1797409, BZ#1790635, BZ#1823398, BZ#1745507, BZ#1732726, BZ#1757877, JIRA:RHELPLAN-25571, BZ#1777138, JIRA:RHELPLAN-27987, BZ#1797671, BZ#1780124, JIRA:RHELPLAN-2507, JIRA:RHELPLAN-37713, JIRA:RHELPLAN-37777, BZ#1841170, JIRA:RHELPLAN-13995, BZ#1785248, BZ#1755347, BZ#1784455, BZ#1784456, BZ#1789401, JIRA:RHELPLAN-41384, BZ#1690207, JIRA:RHELPLAN-1212, BZ#1559616, BZ#1812552, JIRA:RHELPLAN-14047, BZ#1769727, JIRA:RHELPLAN-27394, BZ#1642765, JIRA:RHELPLAN-10304, BZ#1646541, BZ#1647725, BZ#1686057, BZ#1748980, BZ#1827628

APPENDIX B. REVISION HISTORY

0.1-2

Mon Oct 05 2020, Lucie Maňásková (lmanasko@redhat.com)

- Added a bug fix (Networking).

0.1-1

Tue Sep 29 2020, Lenka Špačková (lspackova@redhat.com)

- Updated the in-place upgrade path with the release of RHEL 7.9.

0.1-0

Thu Aug 27 2020, Lucie Maňásková (lmanasko@redhat.com)

- Added a bug fix (Kernel).

0.0-9

Mon Aug 10 2020, Lucie Maňásková (lmanasko@redhat.com)

- Added a known issue (Identity Management).

0.0-8

Tue Jul 21 2020, Lucie Maňásková (lmanasko@redhat.com)

- Release of the Red Hat Enterprise Linux 8.2.1 Release Notes.

0.0-7

Thu Jul 16 2020, Lucie Maňásková (lmanasko@redhat.com)

- Added a Technology Preview (Networking).
- Updated the New Features section.

0.0-6

Thu Jun 25 2020, Jaroslav Klech (jklech@redhat.com)

- Granulated the kernel parameters chapter.
- Added various improvements to the device drivers chapter.

0.0-5

Fri Jun 19 2020, Lucie Maňásková (lmanasko@redhat.com)

- Added new known issues.
- Several updates to other release notes.

0.0-4

Thu Jun 04 2020, Lucie Maňásková (lmanasko@redhat.com)

- Updated the New features section.

- Added a known issue (Containers).

0.0-3

Wed May 20 2020, Lenka Špačková (lspackova@redhat.com)

- Added a known issue (Dynamic programming languages, web and database servers).
- Added a bug fix (Compilers and development tools).
- Several updates to other release notes.

0.0-2

Tue Apr 28 2020, Lucie Maňásková (lmanasko@redhat.com)

- Release of the Red Hat Enterprise Linux 8.2 Release Notes.

0.0-1

Mon Mar 09 2020, Jaroslav Klech (jklech@redhat.com)

- Provided Important Changes to External Kernel Parameters and New Drivers chapters.

0.0-0

Tue Jan 21 2020, Lucie Maňásková (lmanasko@redhat.com)

- Release of the Red Hat Enterprise Linux 8.2 Beta Release Notes.