Red Hat Enterprise Linux 8

8.1 Release Notes

Release Notes for Red Hat Enterprise Linux 8.1

Last Updated: 2019-11-26

Red Hat Enterprise Linux 8 8.1 Release Notes

Release Notes for Red Hat Enterprise Linux 8.1

Legal Notice

Abstract

The Release Notes provide high-level coverage of the improvements and additions that have been implemented in Red Hat Enterprise Linux 8.1 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

Users can now disable modules during a Kickstart installation.

See Section 4.1, "Installer and image creation" for further details.

Red Hat Enterprise Linux System Roles

A new **storage** role has been added to RHEL System Roles.

See Section 4.3, "Red Hat Enterprise Linux System Roles" for details.

Infrastructure services

RHEL 8.1 introduces a new routing protocol stack, **FRR**, which replaces **Quagga** that was used on previous versions of RHEL. **FRR** provides TCP/IP-based routing services with support for multiple IPv4 and IPv6 routing protocols.

The **Tuned** system tuning tool has been rebased to version 2.12, which adds support for negation of CPU list.

The **chrony** suite has been rebased to version 3.5, which adds support for more accurate synchronization of the system clock with hardware timestamping in RHEL 8.1 kernel.

For more information, see Section 4.4, "Infrastructure services" .

Security

RHEL 8.1 introduces a new tool for generating SELinux policies for containers: **udica**. With **udica**, you can create a tailored security policy for better control of how a container accesses host system resources, such as storage, devices, and network. This enables you to harden your container deployments against security violations and it also simplifies achieving and maintaining regulatory compliance.

The **fapolicyd** software framework introduces a form of application whitelisting and blacklisting based on a user-defined policy. The RHEL 8.1 application whitelisting feature provides one of the most efficient ways to prevent running untrusted and possibly malicious applications on the system.

A security compliance suite, **OpenSCAP**, now supports SCAP 1.3 data streams and provides improved reports.

See Section 4.5, "Security" for more information.

Kernel

Live patching for the kernel, **kpatch**, is now available, which enables you to consume Critical and Important CVEs fixes without the need to reboot your system. See Section 4.7, "Kernel" for more information.

File systems and storage

The LUKS version 2 (**LUKS2**) format now supports re-encrypting block devices while the devices are in use.

See Section 4.8, "File systems and storage" for more information.

Dynamic programming languages, web and database servers

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

- **PHP 7.3**

- **Ruby 2.6**
- **Node.js 12**
- **nginx 1.16**

See Section 4.10, “Dynamic programming languages, web and database servers” for details.

Compiler toolsets

RHEL 8.1 introduces a new compiler toolset, **GCC Toolset 9**, an Application Stream packaged as a Software Collection, which provides recent versions of development tools.

In addition, the following compiler toolsets have been upgraded:

- **LLVM 8.0.1**
- **Rust Toolset 1.37**
- **Go Toolset 1.12.8**

See Section 4.11, “Compilers and development tools” for more information.

Identity Management

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

See Section 4.12, “Identity Management” for details.

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 4.12, “Identity Management” 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. For more information,see Section 4.13, “Desktop”.

The **Direct Rendering Manager** (DRM) kernel graphics subsystem has been rebased to upstream Linux kernel version 5.1. This version provides a number of enhancements over the previous version, including support for new GPUs and APUs, and various driver updates. See Section 4.13, “Desktop” for further details.

In-place upgrade from RHEL 7 to RHEL 8

The supported in-place upgrade path is currently **from RHEL 7.6 to RHEL 8.1** The following major enhancements have been introduced:

- Support for an in-place upgrade on the following architectures has been added: 64-bit ARM, IBM POWER (little endian), IBM Z.
- It is now possible to perform a pre-upgrade system assessment in the web console and apply automated remediations using the new **cockpit-leapp** plug-in.
- The /**var** or /**usr** directories can now be mounted on a separate partition.
- UEFI is now supported.

- **Leapp** now upgrades packages from the Supplementary repository.

For details and usage instructions, see Upgrading to RHEL 8 .

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
- 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 8.1 is distributed with the kernel version 4.18, 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. 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. NEW FEATURES

This part describes new features and major enhancements introduced in Red Hat Enterprise Linux 8.1.

4.1. INSTALLER AND IMAGE CREATION

Modules can now be disabled during Kickstart installation

With this enhancement, users can now disable a module to prevent the installation of packages from the module. To disable a module during Kickstart installation, use the command:

module --name=foo --stream=bar --disable

(BZ#1655523)

Support for the repo.git section to blueprints is now available

A new **repo.git** blueprint section allows users to include extra files in their image build. The files must be hosted in git repository that is accessible from the **lorax-composer** build server.

(BZ#1709594)

Image Builder now supports image creation for more cloud providers

With this update, the Image Builder expanded the number of Cloud Providers that the Image Builder can create an image for. As a result, now you can create RHEL images that can be deployed also on Google Cloud and Alibaba Cloud as well as run the custom instances on these platforms.

(BZ#1689140)

4.2. SOFTWARE MANAGEMENT

dnf-utils has been renamed to yum-utils

With this update, the **dnf-utils** package, that is a part of the **YUM** stack, has been renamed to **yum-utils**. For compatibility reasons, the package can still be installed using the **dnf-utils** name, and will automatically replace the original package when upgrading your system.

(BZ#1722093)

4.3. RED HAT ENTERPRISE LINUX SYSTEM ROLES

A new storage role added to RHEL System Roles

The **storage** role has been added to RHEL System Roles provided by the **rhel-system-roles** package. The **storage** role can be used to manage local storage using Ansible.

Currently, the **storage** role supports the following types of tasks:

- Managing file systems on whole disks
- Managing LVM volume groups and their file systems
- Managing logical volumes and their file systems

For more information, see Managing file systems and Configuring and managing logical volumes.

(BZ#1691966)

4.4. INFRASTRUCTURE SERVICES

tuned rebased to version 2.12

The **tuned** packages have been upgraded to upstream version 2.12, which provides a number of bug fixes and enhancements over the previous version, notably:

- Handling of devices that have been removed and reattached has been fixed.
- Support for negation of CPU list has been added.
- Performance of runtime kernel parameter configuration has been improved by switching from the **sysctl** tool to a new implementation specific to **Tuned**.

(BZ#1685585)

chrony rebased to version 3.5

The **chrony** packages have been upgraded to upstream version 3.5, which provides a number of bug fixes and enhancements over the previous version, notably:

- Support for more accurate synchronization of the system clock with hardware timestamping in RHEL 8.1 kernel has been added.
- Hardware timestamping has received significant improvements.
- The range of available polling intervals has been extended.
- The filter option has been added to NTP sources.

(BZ#1685469)

New FRRouting routing protocol stack is available

With this update, **Quagga** has been replaced by **Free Range Routing** (**FRRouting**, or **FRR**), which is a new routing protocol stack. **FRR** is provided by the **frr** package available in the AppStream repository.

FRR provides TCP/IP-based routing services with support for multiple IPv4 and IPv6 routing protocols, such as **BGP**, **IS-IS**, **OSPF**, **PIM**, and **RIP**.

With **FRR** installed, the system can act as a dedicated router, which exchanges routing information with other routers in either internal or external network.

For more information, see Setting the routing protocols for your system .

(BZ#1657029)

GNU enscript now supports ISO-8859-15 encoding

With this update, support for ISO-8859-15 encoding has been added into the GNU enscript program.

(BZ#1664366)

Improved accuracy of measuring system clock offset in phc2sys

The **phc2sys** program from the **linuxptp** packages now supports a more accurate method for measuring the offset of the system clock.

(BZ#1677217)

ptp4l now supports team interfaces in active-backup mode

With this update, support for team interfaces in active-backup mode has been added into the **PTP Boundary/Ordinary Clock** (ptp4l).

(BZ#1685467)

The PTP time synchronization on macvlan interfaces is now supported

This update adds support for hardware timestamping on **macvlan** interfaces into the Linux kernel. As a result, **macvlan** interfaces can now use the **Precision Time Protocol** (PTP) for time synchronization.

(BZ#1664359)

4.5. SECURITY

New package: fapolicyd

The **fapolicyd** software framework introduces a form of application whitelisting and blacklisting based on a user-defined policy. The application whitelisting feature provides one of the most efficient ways to prevent running untrusted and possibly malicious applications on the system.

The **fapolicyd** framework provides the following components:

- **fapolicyd** service
- **fapolicyd** command-line utilities
- **yum** plugin
- rule language

Administrator can define the **allow** and **deny** execution rules, both with possibility of auditing, based on a path, hash, MIME type, or trust for any application.

Note that every **fapolicyd** setup affects overall system performance. The performance hit varies depending on the use case. The application whitelisting slow-downs the **open()** and **exec()** system calls, and therefore primarily affects applications that perform such system calls frequently.

See the Configuring and managing application whitelists section in the RHEL 8 Security hardening title and the **fapolicyd(8)**, **fapolicyd.rules(5)**, and **fapolicyd.conf(5)** man pages for more information.

(BZ#1673323)

New package: udica

The new **udica** package provides a tool for generation SELinux policies for containers. With **udica**, you can create a tailored security policy for better control of how a container accesses host system resources, such as storage, devices, and network. This enables you to harden your container deployments against security violations and it also simplifies achieving and maintaining regulatory compliance.

See the Creating SELinux policies for containers section in the RHEL 8 Using SELinux title for more information.

(BZ#1673643)

SELinux user-space tools updated to version 2.9

The **libsepol**, **libselinux**, **libsemanage**, **policycoreutils**, **checkpolicy**, and **mcstrans** SELinux user-space tools have been upgraded to the latest upstream release 2.9, which provides many bug fixes and enhancements over the previous version.

(BZ#1672638, BZ#1672642, BZ#1672637, BZ#1672640, BZ#1672635, BZ#1672641)

SETools updated to version 4.2.2

The SETools collection of tools and libraries has been upgraded to the latest upstream release 4.2.2, which provides the following changes:

- Removed source policy references from man pages, as loading source policies is no longer supported
- Fixed a performance regression in alias loading

(BZ#1672631)

selinux-policy rebased to 3.14.3

The **selinux-policy** package has been upgraded to upstream version 3.14.3, which provides a number of bug fixes and enhancements to the allow rules over the previous version.

(BZ#1673107)

A new SELinux type: **boltd_t**

A new SELinux type, **boltd_t**, confines **boltd**, a system daemon for managing Thunderbolt 3 devices. As a result, **boltd** now runs as a confined service in SELinux enforcing mode.

(BZ#1684103)

A new SELinux policy class: **bpf**

A new SELinux policy class, **bpf**, has been introduced. The **bpf** class enables users to control the Berkeley Packet Filter (BPF) flow through SElinux, and allows inspection and simple manipulation of Extended Berkeley Packet Filter (eBPF) programs and maps controlled by SELinux.

(BZ#1673056)

OpenSCAP rebased to version 1.3.1

The **openscap** packages have been upgraded to upstream version 1.3.1, which provides many bug fixes and enhancements over the previous version, most notably:

- Support for SCAP 1.3 source data streams: evaluating, XML schemas, and validation
- Tailoring files are included in ARF result files
- OVAL details are always shown in HTML reports, users do not have to provide the **--oval-results** option

- HTML report displays OVAL test details also for OVAL tests included from other OVAL definitions using the OVAL **extend_definition** element
- OVAL test IDs are shown in HTML reports
- Rule IDs are shown in HTML guides

(BZ#1718826)

OpenSCAP now supports SCAP 1.3

The **OpenSCAP** suite now supports data streams conforming to the latest version of the SCAP standard - SCAP 1.3. You can now use SCAP 1.3 data streams, such as those contained in the **scap-security-guide** package, in the same way as SCAP 1.2 data streams without any additional usability restrictions.

(BZ#1709429)

scap-security-guide rebased to version 0.1.44

The **scap-security-guide** packages have been upgraded to upstream version 0.1.44, which provides many bug fixes and enhancements over the previous version, most notably: * SCAP content conforms to the latest version of SCAP standard, SCAP 1.3 * SCAP content supports UBI images

(BZ#1718839)

OpenSSH rebased to 8.0p1

The **openssh** packages have been upgraded to upstream version 8.0p1, which provides many bug fixes and enhancements over the previous version, most notably:

- Increased default RSA key size to 3072 bits for the **ssh-keygen** tool
- Removed support for the **ShowPatchLevel** configuration option
- Applied numerous GSSAPI key exchange code fixes, such as the fix of Kerberos cleanup procedures
- Removed fall back to the **sshd_net_t** SELinux context
- Added support for **Match final** blocks
- Fixed minor issues in the **ssh-copy-id** command
- Fixed Common Vulnerabilities and Exposures (CVE) related to the **scp** utility (CVE-2019-6111, CVE-2018-20685, CVE-2019-6109)

Note, that this release introduces minor incompatibility of **scp** as mitigation of CVE-2019-6111. If your scripts depend on advanced bash expansions of the path during an scp download, you can use the **-T** switch to turn off these mitigations temporarily when connecting to trusted servers.

(BZ#1691045)

libssh now complies with the system-wide crypto-policies

The **libssh** client and server now automatically load the **/etc/libssh/libssh_client.config** file and the **/etc/libssh/libssh_server.config**, respectively. This configuration file includes the options set by the system-wide **crypto-policies** component for the **libssh** back end and the options set in the

/etc/ssh/ssh_config or **/etc/ssh/sshd_config** OpenSSH configuration file. With automatic loading of the configuration file, **libssh** now use the system-wide cryptographic settings set by **crypto-policies**. This change simplifies control over the set of used cryptographic algorithms by applications.

(BZ#1610883, BZ#1610884)

An option for **rsyslog** to preserve case of **FROMHOST** is available

This update to the **rsyslog** service introduces the option to manage letter case preservation of the **FROMHOST** property for the **imudp** and **imtcp** modules. Setting the **preservecase** value to **on** means the **FROMHOST** property is handled in a case sensitive manner. To avoid breaking existing configurations, the default values of **preservecase** are **on** for **imtcp** and **off** for **imudp**.

(BZ#1614181)

4.6. NETWORKING

PMTU discovery and route redirection is now supported with VXLAN and GENEVE tunnels

The kernel in Red Hat Enterprise Linux (RHEL) 8.0 did not handle Internet Control Message Protocol (ICMP) and ICMPv6 messages for Virtual Extensible LAN (VXLAN) and Generic Network Virtualization Encapsulation (GENEVE) tunnels. As a consequence, Path MTU (PMTU) discovery and route redirection was not supported with VXLAN and GENEVE tunnels in RHEL releases prior to 8.1. With this update, the kernel handles ICMP "Destination Unreachable" and "Redirect Message", as well as ICMPv6 "Packet Too Big" and "Destination Unreachable" error messages by adjusting the PMTU and modifying forwarding information. As a result, RHEL 8.1 supports PMTU discovery and route redirection with VXLAN and GENEVE tunnels.

(BZ#1652222)

Notable changes in XDP and networking eBPF features in kernel

The XDP and the networking eBPF features in the **kernel** package have been upgraded to upstream version 5.0, which provides a number of bug fixes and enhancements over the previous version:

- eBPF programs can now better interact with the TCP/IP stack, perform flow dissection, have wider range of **bpf** helpers available, and have access to new map types.
- XDP metadata are now available to AF_XDP sockets.

(BZ#1687459)

The new **PTP_SYS_OFFSET_EXTENDED** control for **ioctl()** improves the accuracy of measured system-PHC ofsets

This enhancement adds the **PTP_SYS_OFFSET_EXTENDED** control for more accurate measurements of the system precision time protocol (PTP) hardware clock (PHC) offset to the **ioctl()** function. The **PTP_SYS_OFFSET** control which, for example, the **chrony** service uses to measure the offset between a PHC and the system clock is not accurate enough. With the new **PTP_SYS_OFFSET_EXTENDED** control, drivers can isolate the reading of the lowest bits. This improves the accuracy of the measured offset. Network drivers typically read multiple PCI registers, and the driver does not read the lowest bits of the PHC time stamp between two readings of the system clock.

(BZ#1677215)

ipset rebased to version 7.1

The **ipset** packages have been upgraded to upstream version 7.1, which provides a number of bug fixes and enhancements over the previous version:

- The **ipset** protocol version 7 introduces the **IPSET_CMD_GET_BYNAME** and **IPSET_CMD_GET_BYINDEX** operations. Additionally, the user space component can now detect the exact compatibility level that the kernel component supports.
- A significant number of bugs have been fixed, such as memory leaks and use-after-free bugs.

(BZ#1649090)

4.7. KERNEL

Live patching for the kernel is now available

Live patching for the kernel, **kpatch**, provides a mechanism to patch the running kernel without rebooting or restarting any processes. Live kernel patches will be provided for selected minor release streams of RHEL covered under the Extended Update Support (EUS) policy to remediate Critical and Important CVEs.

To subscribe to the **kpatch** stream for the RHEL 8.1 version of the kernel, install the **kpatch-patch-4_18_0-147** package provided by the RHEA-2019:3695 advisory.

For more information, see Applying patches with kernel live patching in Managing, monitoring and updating the kernel.

(BZ#1763780)

Red Hat Enterprise Linux 8 now supports **early kdump**

The **early kdump** feature allows the crash kernel and initramfs to load early enough to capture the **vmcore** information even for early crashes.

For more details about **early kdump**, see the **/usr/share/doc/kexec-tools/early-kdump-howto.txt** file.

(BZ#1520209)

RHEL 8 now supports **ipcmni_extend**

A new kernel command line parameter **ipcmni_extend** has been added to Red Hat Enterprise Linux 8. The parameter extends a number of unique System V Inter-process Communication (IPC) identifiers from the current maximum of 32 KB (15 bits) up to 16 MB (24 bits). As a result, users whose applications produce a lot of shared memory segments are able to create a stronger IPC identifier without exceeding the 32 KB limit.

Note that in some cases using **ipcmni_extend** results in a small performance overhead and it should be used only if the applications need more than 32 KB of unique IPC identifier.

(BZ#1710480)

The persistent memory initialization code supports parallel initialization

The persistent memory initialization code enables parallel initialization on systems with multiple nodes of persistent memory. The parallel initialization greatly reduces the overall memory initialization time on systems with large amounts of persistent memory. As a result, these systems can now boot much faster.

(BZ#1634343)

TPM userspace tool has been updated to the last version

The **tpm2-tools** userspace tool has been updated to version 2.0. With this update, **tpm2-tools** is able to fix many defects.

(BZ#1664498)

The rngd daemon 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)

Intel ® Omni-Path Architecture (OPA) Host Software

Intel Omni-Path Architecture (OPA) host software is fully supported in Red Hat Enterprise Linux 8.1. Intel OPA provides Host Fabric Interface (HFI) hardware with initialization and setup for high performance data transfers (high bandwidth, high message rate, low latency) between compute and I/O nodes in a clustered environment.

For instructions on installing Intel Omni-Path Architecture documentation, see: https://cdrdv2.intel.com/v1/dl/getContent/616368

(BZ#1766186)

bcc-tools is now fully supported only on x86_64 in RHEL 8

BPF Compiler Collection (BCC) is a userspace tool kit for creating efficient kernel tracing and manipulation programs. **BCC** provides tools for I/O analysis, networking, and monitoring of Linux operating systems using the **extended Berkeley Packet Filtering (eBPF)**. Any other use of **eBPF** including **bcc library**, **bpftrace** and **bcc-tools** on other than x86_64 architecture remains Technology Preview.

(BZ#1667043)

UBSan has been enabled in the debug kernel in RHEL 8

The **Undefined Behavior Sanitizer** (**UBSan**) utility exposes undefined behavior flaws in C code languages at runtime. This utility has now been enabled in the debug kernel because the compiler behavior was, in some cases, different than developers' expectations. Especially, in the case of compiler optimization, where subtle, obscure bugs would appear. As a result, running the debug kernel with **UBSan** enabled allows the system to easily detect such bugs.

(BZ#1571628)

The fadump infrastructure now supports re-registering in RHEL 8

The support has been added for re-registering (unregistering and registering) of the firmware-assisted dump (**fadump**) infrastructure after any memory hot add/remove operation to update the crash memory ranges. The feature aims to prevent the system from potential racing issues during unregistering and registering **fadump** from userspace during **udev** events.

(BZ#1710288)

The determine_maximum_mpps.sh script has been introduced in RHEL for Real Time 8

The **determine_maximum_mpps.sh** script has been introduced to help use the **queuelat** test program. The script executes **queuelat** to determine the maximum packets per second a machine can handle.

(BZ#1686494)

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

The **kernel-rt** sources have been upgraded to be based on the latest Red Hat Enterprise Linux kernel source tree, which provides a number of bug fixes and enhancements over the previous version.

(BZ#1678887)

The ssdd test has been added to RHEL for Real Time 8

The **ssdd** test has been added to enable stress testing of the tracing subsystem. The test runs multiple tracing threads to verify locking is correct within the tracing system.

(BZ#1666351)

4.8. FILE SYSTEMS AND STORAGE

Support for Data Integrity Field/Data Integrity Extension (DIF/DIX)

DIF/DIX is supported on configurations where the hardware vendor has qualified it and provides full support for the particular host bus adapter (HBA) and storage array configuration on RHEL.

DIF/DIX is not supported on the following configurations:

- It is not supported for use on the boot device.
- It is not supported on virtualized guests.
- Red Hat does not support using the Automatic Storage Management library (ASMLib) when DIF/DIX is enabled.

DIF/DIX is enabled or disabled at the storage device, which involves various layers up to (and including) the application. The method for activating the DIF on storage devices is device-dependent.

For further information on the DIF/DIX feature, see What is DIF/DIX.

(BZ#1649493)

Optane DC memory systems now supports EDAC reports

Previously, EDAC was not reporting memory corrected/uncorrected events if the memory address was within a NVDIMM module. With this update, EDAC can properly report the events with the correct memory module information.

(BZ#1571534)

The VDO Ansible module has been moved to Ansible packages

Previously, the VDO Ansible module was provided by the **vdo** RPM package. Starting with this release, the module is provided by the **ansible** package instead.

The original location of the VDO Ansible module file was:

```
/usr/share/doc/vdo/examples/ansible/vdo.py
```

The new location of the file is:

```
/usr/lib/python3.6/site-packages/ansible/modules/system/vdo.py
```

The **vdo** package continues to distribute Ansible playbooks.

For more information on Ansible, see http://docs.ansible.com/.

(BZ#1669534)

Aero adapters are now fully supported

The following Aero adapters, previously available as a Technology Preview, are now fully supported:

- PCI ID 0x1000:0x00e2 and 0x1000:0x00e6, controlled by the **mpt3sas** driver
- PCI ID 0x1000:Ox10e5 and 0x1000:0x10e6, controlled by the **megaraid_sas** driver

(BZ#1663281)

LUKS2 now supports online re-encryption

The Linux Unified Key Setup version 2 (LUKS2) format now supports re-encrypting encrypted devices while the devices are in use. For example, you do not have to unmount the file system on the device to perform the following tasks:

- Change the volume key
- Change the encryption algorithm

When encrypting a non-encrypted device, you must still unmount the file system, but the encryption is now significantly faster. You can remount the file system after a short initialization of the encryption.

Additionally, the LUKS2 re-encryption is now more resilient. You can select between several options that prioritize performance or data protection during the re-encryption process.

To perform the LUKS2 re-encryption, use the **cryptsetup reencrypt** subcommand. Red Hat no longer recommends using the **cryptsetup-reencrypt** utility for the LUKS2 format.

Note that the LUKS1 format does not support online re-encryption, and the **cryptsetup reencrypt** subcommand is not compatible with LUKS1. To encrypt or re-encrypt a LUKS1 device, use the **cryptsetup-reencrypt** utility.

For more information on disk encryption, see Encrypting block devices using LUKS.

(BZ#1676622)

New features of ext4 available in RHEL 8

In RHEL8, following are the new fully supported features of ext4:

- Non-default features:
  - **project**
  - **quota**
  - **mmp**

- Non-default mount options:
  - **bsddf|minixdf**
  - **grpid|bsdgroups and nogrpid|sysvgroups**
  - **resgid=n and resuid=n**
  - **errors={continue|remount-ro|panic}**
  - **commit=nrsec**
  - **max_batch_time=usec**
  - **min_batch_time=usec**
  - **grpquota|noquota|quota|usrquota**
  - **prjquota**
  - **dax**
  - **lazytime|nolazytime**
  - **discard|nodiscard**
  - **init_itable|noinit_itable**
  - **jqfmt={vfsold|vfsv0|vfsv1}**
  - **usrjquota=aquota.user|grpjquota=aquota.group**

For more information on features and mount options, see the **ext4** man page. Other ext4 features, mount options or both, or combination of features, mount options or both may not be fully supported by Red Hat. If your special workload requires a feature or mount option that is not fully supported in the Red Hat release, contact Red Hat support to evaluate it for inclusion in our supported list.

(BZ#1741531)

NVMe over RDMA now supports an **Infiniband** in the target mode for IBM Coral systems

In RHEL 8.1, NVMe over RDMA now supports an **Infiniband** in the target mode for IBM Coral systems, with a single NVMe PCIe add in card as the target.

(BZ#1721683)

4.9. HIGH AVAILABILITY AND CLUSTERS

Pacemaker now defaults the **concurrent-fencing** cluster property to **true**

If multiple cluster nodes need to be fenced at the same time, and they use different configured fence devices, Pacemaker will now execute the fencing simultaneously, rather than serialized as before. This can result in greatly sped up recovery in a large cluster when multiple nodes must be fenced.

(BZ#1715426)

Extending a shared logical volume no longer requires a refresh on every cluster node

With this release, extending a shared logical volume no longer requires a refresh on every cluster node after running the **lvextend** command on one cluster node. For the full procedure to extend the size of a GFS2 file system, see Growing a GFS2 file system .

(BZ#1649086)

Maximum size of a supported RHEL HA cluster increased from 16 to 32 nodes

With this release, Red Hat supports cluster deployments of up to 32 full cluster nodes.

(BZ#1693491)

4.10. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

A new module stream: php:7.3

RHEL 8.1 introduces **PHP 7.3**, which provides a number of new features and enhancements. Notable changes include:

- Enhanced and more flexible **heredoc** and **nowdoc** syntaxes
- The PCRE extension upgraded to PCRE2
- Improved multibyte string handling
- Support for LDAP controls
- Improved FastCGI Process Manager (FPM) logging
- Several deprecations and backward incompatible changes

For more information, see Migrating from PHP 7.2.x to PHP 7.3.x .

Note that the RHEL 8 version of **PHP 7.3** does not support the **Argon2** password hashing algorithm.

To install the **php:7.3** stream, use:

```
# yum install @php:7.3
```

If you want to upgrade from the **php:7.2** stream, see Switching to a later stream .

(BZ#1653109)

A new module stream: ruby:2.6

A new module stream, **ruby:2.6**, is now available. **Ruby 2.6.3**, included in RHEL 8.1, provides numerous new features, enhancements, bug and security fixes, and performance improvements over version 2.5 distributed in RHEL 8.0.

Notable enhancements include:

- Constant names are now allowed to begin with a non-ASCII capital letter.
- Support for an endless range has been added.
- A new **Binding#source_location** method has been provided.

- **$SAFE** is now a process global state and it can be set back to **0**.

The following performance improvements have been implemented:

- The **Proc#call** and **block.call** processes have been optimized.
- A new garbage collector managed heap, Transient heap (**theap**), has been introduced.
- Native implementations of coroutines for individual architectures have been introduced.

In addition, **Ruby 2.5**, provided by the **ruby:2.5** stream, has been upgraded to version 2.5.5, which provides a number of bug and security fixes.

To install the **ruby:2.6** stream, use:

```
# yum install @ruby:2.6
```

If you want to upgrade from the **ruby:2.5** stream, see Switching to a later stream .

(BZ#1672575)

A new module stream: nodejs:12

RHEL 8.1 introduces **Node.js 12**, which provides a number of new features and enhancements over version 10. Notable changes include:

- The V8 engine upgraded to version 7.4
- A new default HTTP parser, **llhttp** (no longer experimental)
- Integrated capability of heap dump generation
- Support for ECMAScript 2015 (ES6) modules
- Improved support for native modules
- Worker threads no longer require a flag
- A new experimental diagnostic report feature
- Improved performance

To install the **nodejs:12** stream, use:

```
# yum install @nodejs:12
```

If you want to upgrade from the **nodejs:10** stream, see Switching to a later stream .

(BZ#1685191)

Judy-devel available in CRB

The **Judy-devel** package is now available as a part of the **mariadb-devel:10.3** module in the CodeReady Linux Builder repository (CRB). As a result, developers are now able to build applications with the **Judy** library.

To install the **Judy-devel** package, enable the **mariadb-devel:10.3** module first:

```
# yum module enable mariadb-devel:10.3
# yum install Judy-devel
```

(BZ#1657053)

FIPS compliance in Python 3

This update adds support for OpenSSL FIPS mode to **Python 3**. Namely:

- In FIPS mode, the **blake2**, **sha3**, and **shake** hashes use the OpenSSL wrappers and do not offer extended functionality (such as keys, tree hashing, or custom digest size).

- In FIPS mode, the **hmac.HMAC** class can be instantiated only with an OpenSSL wrapper or a string with OpenSSL hash name as the **digestmod** argument. The argument must be specified (instead of defaulting to the **md5** algorithm).

Note that hash functions support the **usedforsecurity** argument, which allows using insecure hashes in OpenSSL FIPS mode. The user is responsible for ensuring compliance with any relevant standards.

(BZ#1731424)

FIPS compliance changes in python3-wheel

This update of the **python3-wheel** package removes a built-in implementation for signing and verifying data that is not compliant with FIPS.

(BZ#1731526)

A new module stream: nginx:1.16

The **nginx 1.16** web and proxy server, which provides a number of new features and enhancements over version 1.14, is now available. For example:

- Numerous updates related to SSL (loading of SSL certificates and secret keys from variables, variable support in the **ssl_certificate** and **ssl_certificate_key** directives, a new **ssl_early_data** directive)

- New **keepalive**-related directives

- A new **random** directive for distributed load balancing

- New parameters and improvements to existing directives (port ranges for the **listen** directive, a new **delay** parameter for the **limit_req** directive, which enables two-stage rate limiting)

- A new **$upstream_bytes_sent** variable

- Improvements to User Datagram Protocol (UDP) proxying

Other notable changes include:

- In the **nginx:1.16** stream, the **nginx** package does not require the **nginx-all-modules** package, therefore **nginx** modules must be installed explicitly. When you install **nginx** as module, the **nginx-all-modules** package is installed as a part of the **common** profile, which is the default profile.

- The **ssl** directive has been deprecated; use the **ssl** parameter for the **listen** directive instead.

- **nginx** now detects missing SSL certificates during configuration testing.

- When using a host name in the **listen** directive, **nginx** now creates listening sockets for all addresses that the host name resolves to.

To install the **nginx:1.16** stream, use:

```
# yum install @nginx:1.16
```

If you want to upgrade from the **nginx:1.14** stream, see Switching to a later stream .

(BZ#1690292)

perl-IO-Socket-SSL rebased to version 2.066

The **perl-IO-Socket-SSL** package has been upgraded to version 2.066, which provides a number of bug fixes and enhancements over the previous version, for example:

- Improved support for TLS 1.3, notably a session reuse and an automatic post-handshake authentication on the client side
- Added support for multiple curves, automatic setting of curves, partial trust chains, and support for RSA and ECDSA certificates on the same domain

(BZ#1632600)

perl-Net-SSLeay rebased to version 1.88

The **perl-Net-SSLeay** package has been upgraded to version 1.88, which provides multiple bug fixes and enhancements. Notable changes include:

- Improved compatibility with OpenSSL 1.1.1, such as manipulating a stack of certificates and X509 stores, and selecting elliptic curves and groups
- Improved compatibility with TLS 1.3, for example, a session reuse and a post-handshake authentication
- Fixed memory leak in the **cb_data_advanced_put()** subroutine.

(BZ#1632597)

4.11. COMPILERS AND DEVELOPMENT TOOLS

GCC Toolset 9 available

Red Hat Enterprise Linux 8.1 introduces GCC Toolset 9, an Application Stream containing more up-to-date versions of development tools.

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

Tool	Version
GCC	9.1.1
GDB	8.3

Tool	Version
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	8.79

GCC Toolset 9 is available as an Application Stream in the form of a Software Collection in the **AppStream** repository. GCC Toolset is a set of tools similar to Red Hat Developer Toolset for RHEL 7.

To install GCC Toolset 9:

```
# 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 detailed instructions regarding usage, see Using GCC Toolset.

(BZ#1685482)

Upgraded compiler toolsets

The following compiler toolsets, distributed as Application Streams, have been upgraded with RHEL 8.1:

- Clang and LLVM Toolset, which provides the LLVM compiler infrastructure framework, the Clang compiler for the C and C++ languages, the LLDB debugger, and related tools for code analysis, to version 8.0.1

- Rust Toolset, which provides the Rust programming language compiler **rustc**, the **cargo** build tool and dependency manager, and required libraries, to version 1.37

- Go Toolset, which provides the Go (**golang**) programming language tools and libraries, to version 1.12.8.

(BZ#1731502, BZ#1691975, BZ#1680091, BZ#1677819, BZ#1681643)

SystemTap rebased to version 4.1

The SystemTap instrumentation tool has been updated to upstream version 4.1. Notable improvements include:

- The eBPF runtime backend can handle more features of the scripting language such as string variables and rich formatted printing.

- Performance of the translator has been significantly improved.

- More types of data in optimized C code can now be extracted with DWARF4 debuginfo constructs.

(BZ#1675740)

General availability of the DHAT tool

Red Hat Enterprise Linux 8.1 introduces the general availability of the **DHAT** tool. It is based on the **valgrind** tool version 3.15.0.

You can find changes/improvements in **valgrind** tool functionality below:

- use *--tool=dhat* instead of *--tool=exp-dhat*,

- *--show-top-n* and *--sort-by* options have been removed because **dhat** tool now prints the minimal data after the program ends,

- a new viewer **dh_view.html**, which is a JavaScript programm, contains the profile results. A short message explains how to view the results after the run is ended,

- the documentation for a viewer is located: */usr/libexec/valgrind/dh_view.html*,

- the documentation for the **DHAT** tool is located: */usr/share/doc/valgrind/html/dh-manual.html*,

- the support for amd64 (x86_64): the **RDRAND** and **F16C insn** set extensions is added,

- in **cachegrind** the **cg_annotate** command has a new option, *--show-percs*, which prints percentages next to all event counts,

- in **callgrind** the **callgrind_annotate** command has a new option, *--show-percs*, which prints percentages next to all event counts,

- in **massif** the default value for *--read-inline-info* is now **yes**,

- in **memcheck** option *--xtree-leak=yes*, which outputs leak result in **xtree** format, automatically activates the option *--show-leak-kinds=all*,

- the new option *--show-error-list=no|yes* displays the list of the detected errors and the used suppression at the end of the run. Previously, the user could specify the option *-v* for **valgrind** command, which shows a lot of information that might be confusing. The option *-s* is an equivalent to the option *--show-error-list=yes* .

(BZ#1683715)

elfutils rebased to version 0.176

The elfutils packages have been updated to upstream version 0.176. This version brings various bug fixes, and resolves the following vulnerabilities:

- CVE-2019-7146
- CVE-2019-7149
- CVE-2019-7150
- CVE-2019-7664
- CVE-2019-7665

Notable improvements include:

- The **libdw** library has been extended with the **dwelf_elf_begin()** function which is a variant of **elf_begin()** that handles compressed files.
- A new **--reloc-debug-sections-only** option has been added to the **eu-strip** tool to resolve all trivial relocations between debug sections in place without any other stripping. This functionality is relevant only for **ET_REL** files in certain circumstances.

(BZ#1683705)

Additional memory allocation checks in **glibc**

Application memory corruption is a leading cause of application and security defects. Early detection of such corruption, balanced against the cost of detection, can provide significant benefits to application developers.

To improve detection, six additional memory corruption checks have been added to the **malloc** metadata in the GNU C Library (**glibc**), which is the core C library in RHEL. These additional checks have been added at a very low cost to runtime performance.

(BZ#1651283)

GDB can access more POWER8 registers

With this update, the GNU debugger (GDB) and its remote stub **gdbserver** can access the following additional registers and register sets of the POWER8 processor line of IBM:

- **PPR**
- **DSCR**
- **TAR**
- **EBB/PMU**
- **HTM**

(BZ#1187581)

binutils disassembler can handle NFP binary files

The disassembler tool from the **binutils** package has been extended to handle binary files for the Netronome Flow Processor (NFP) hardware series. This functionality is required to enable further features in the **bpftool** Berkeley Packet Filter (BPF) code compiler.

(BZ#1644391)

Partially writable GOT sections are now supported on the IBM Z architecture

The IBM Z binaries using the "lazy binding" feature of the loader can now be hardened by generating partially writable Global offset table (GOT) sections. These binaries require a read-write GOT, but not all entries to be writable. This update provides protection for the entries from potential attacks.

(BZ#1525406)

binutils now supports Arch13 processors of IBM Z

This update adds support for the extensions related to the Arch13 processors into the **binutils** packages on IBM Z architecture. As a result, it is now possible to build kernels that can use features available in arch13-enabled CPUs on IBM Z.

(BZ#1659437)

Dyninst rebased to version 10.1.0

The **Dyninst** instrumentation library has been updated to upstream version 10.1.0. Notable changes include:

- Dyninst supports the Linux PowerPC Little Endian (**ppcle**) and 64-bit ARM (**aarch64**) architectures.
- Start-up time has been improved by using parallel code analysis.

(BZ#1648441)

Date formatting updates for the Japanese Reiwa era

The GNU C Library now provides correct Japanese era name formatting for the Reiwa era starting on May 1st, 2019. The time handling API data has been updated, including the data used by the **strftime** and **strptime** functions. All APIs will correctly print the Reiwa era including when **strftime** is used along with one of the era conversion specifiers such as **%EC**, **%EY**, or **%Ey**.

(BZ#1577438)

Performance Co-Pilot rebased to version 4.3.2

In RHEL 8.1, the Performance Co-Pilot (PCP) tool has been updated to upstream version 4.3.2. Notable improvements include:

- New metrics have been added - Linux kernel entropy, pressure stall information, Nvidia GPU statistics, and more.
- Tools such as **pcp-dstat**, **pcp-atop**, the **perfevent** PMDA, and others have been updated to report the new metrics.
- The **pmseries** and **pmproxy** utilities for a performant PCP integration with Grafana have been updated.

This release is backward compatible for libraries, over-the-wire protocol and on-disk PCP archive format.

(BZ#1685302)

4.12. IDENTITY MANAGEMENT

IdM now supports Ansible roles and modules for installation and management

This update introduces the **ansible-freeipa** package, which provides Ansible roles and modules for Identity Management (IdM) deployment and management. You can use Ansible roles to install and uninstall IdM servers, replicas, and clients. You can use Ansible modules to manage IdM groups, topology, and users. There are also example playbooks available.

This update simplifies the installation and configuration of IdM based solutions.

(JIRA:RHELPLAN-2542)

New tool to test the overall fitness of IdM deployment: **Healthcheck**

This update introduces the **Healthcheck** tool in Identity Management (IdM). The tool provides tests verifying that the current IdM server is configured and running correctly.

The major areas currently covered are: * Certificate configuration and expiration dates * Replication errors * Replication topology * AD Trust configuration * Service status * File permissions of important configuration files * Filesystem space

The **Healthcheck** tool is available in the command-line interface (CLI).

(JIRA:RHELPLAN-13066)

IdM now supports renewing expired system certificates when the server is offline

With this enhancement, administrators can renew expired system certificates when Identity Management (IdM) is offline. When a system certificate expires, IdM fails to start. The new **ipa-cert-fix** command replaces the workaround to manually set the date back to proceed with the renewal process. As a result, the downtime and support costs reduce in the mentioned scenario.

(JIRA:RHELPLAN-13074)

Identity Management supports trust with Windows Server 2019

When using Identity Management, you can now establish a supported forest trust to Active Directory forests that run by Windows Server 2019. The supported forest and domain functional levels are unchanged and supported up to level Windows Server 2016.

(JIRA:RHELPLAN-15036)

samba rebased to version 4.10.4

The *samba* packages have been upgraded to upstream version 4.10.4, which provides a number of bug fixes and enhancements over the previous version:

- Samba 4.10 fully supports Python 3. Note that future Samba versions will not have any runtime support for Python 2.

- The JavaScript Object Notation (JSON) logging feature now logs the Windows event ID and logon type for authentication messages.

- The new **vfs_glusterfs_fuse** file system in user space (FUSE) module improves the performance when Samba accesses a GlusterFS volume. To enable this module, add **glusterfs_fuse** to the **vfs_objects** parameter of the share in the **/etc/samba/smb.conf** file. Note that **vfs_glusterfs_fuse** does not replace the existing **vfs_glusterfs** module.

- The server message block (SMB) client Python bindings are now deprecated and will be removed in a future Samba release. This only affects users who use the Samba Python bindings to write their own utilities.

Samba automatically updates its **tdb** database files when the **smbd**, **nmbd**, or **winbind** service starts. Back up the databases 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-4.10.0.html

(BZ#1638001)

Updated system-wide certificate store location for OpenLDAP

The default location for trusted CAs for OpenLDAP has been updated to use the system-wide certificate store (**/etc/pki/ca-trust/source**) instead of **/etc/openldap/certs**. This change has been made to simplify the setting up of CA trust.

No additional setup is required to set up CA trust, unless you have service-specific requirements. For example, if you require an LDAP server's certificate to be only trusted for LDAP client connections, in this case you must set up the CA certificates as you did previously.

(JIRA:RHELPLAN-7109)

New **ipa-crl-generation** commands have been introduced to simplify managing IdM CRL master

This update introduces the **ipa-crl-generation status/enable/disable** commands. These commands, run by the root user, simplify work with the Certificate Revocation List (CRL) in IdM. Previously, moving the CRL generation master from one IdM CA server to another was a lengthy, manual and error-prone procedure.

The **ipa-crl-generation status** command checks if the current host is the CRL generation master. The **ipa-crl-generation enable** command makes the current host the CRL generation master in IdM if the current host is an IdM CA server. The **ipa-crl-generation disable** command stops CRL generation on the current host.

Additionally, the **ipa-server-install --uninstall** command now includes a safeguard checking whether the host is the CRL generation master. This way, IdM ensures that the system administrator does not remove the CRL generation master from the topology.

(JIRA:RHELPLAN-13068)

OpenID Connect support in **keycloak-httpd-client-install**

The **keycloak-httpd-client-install** identity provider previously supported only the SAML (Security Assertion Markup Language) authentication with the **mod_auth_mellon** authentication module. This rebase introduces the **mod_auth_openidc** authentication module support, which allows you to configure also the OpenID Connect authentication.

The **keycloak-httpd-client-install** identity provider allows an apache instance to be configured as an OpenID Connect client by configuring **mod_auth_openidc**.

(BZ#1553890)

Setting up IdM as a hidden replica is now available as a Technology Preview

This enhancement enables administrators to set up an Identity Management (IdM) replica as a hidden replica. 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.

(BZ#1719767)

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.

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

(JIRA:RHELPLAN-13195)

4.13. DESKTOP

Modified workspace switcher in GNOME Classic

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. Alternatively, you can also use the combination of **Ctrl**+**Alt**+**down**/**up arrow** keys to switch between workspaces. The content of the active workspace is shown in the left part of the bottom bar in form of the *window list*.

When you press the **Super** key within the particular workspace, you can see the *window picker*, which includes all windows that are open in this workspace. However, the *window picker* no longer displays the following elements that were available in the previous release of RHEL:

- *dock* (vertical bar on the left side of the screen)
- *workspace switcher* (vertical bar on the right side of the screen)
- *search entry*

For particular tasks that were previously achieved with the help of these elements, adopt the following approaches:

- To launch applications, instead of using *dock*, you can:

  - Use the **Applications** menu on the top bar
  - Press the kdb:[Alt + F2] keys to make the **Enter a Command** screen appear, and write the name of the executable into this screen.

- To switch between workspaces, instead of using the vertical *workspace switcher*, use the horizontal *workspace switcher* in the right bottom bar.
- If you require the *search entry* or the vertical *workspace switcher*, use GNOME Standard environment instead of GNOME Classic.

(BZ#1704360)

4.14. GRAPHICS INFRASTRUCTURES

DRM rebased to Linux kernel version 5.1

The **Direct Rendering Manager** (DRM) kernel graphics subsystem has been rebased to upstream Linux kernel version 5.1, which provides a number of bug fixes and enhancements over the previous version. Most notably:

- The **mgag200** driver has been updated. The driver continues providing support for HPE Proliant Gen10 Systems, which use Matrox G200 eH3 GPUs. The updated driver also supports current and new Dell EMC PowerEdge Servers.
- The **nouveau** driver has been updated to provide hardware enablement to current and future Lenovo platforms that use NVIDIA GPUs.
- The **i915** display driver has been updated for continued support of current and new Intel GPUs.
- Bug fixes for Aspeed AST BMC display chips have been added.
- Support for AMD Raven 2 set of Accelerated Processing Units (APUs) has been added.
- Support for AMD Picasso APUs has been added.
- Support for AMD Vega GPUs has been added.
- Support for Intel Amber Lake-Y and Intel Comet Lake-U GPUs has been added.

(BZ#1685552)

Support for AMD Picasso graphic cards

This update introduces the **amdgpu** graphics driver. As a result AMD Picasso graphics cards are now fully supported on RHEL 8.

(BZ#1685427)

4.15. THE WEB CONSOLE

Enabling and disabling SMT

Simultaneous Multi-Threading (SMT) configuration is now available in RHEL 8. Disabling SMT in the web console allows you to mitigate a class of CPU security vulnerabilities such as:

- Microarchitectural Data Sampling

- L1 Terminal Fault Attack

(BZ#1678956)

Adding a search box in the Services page

The Services page now has a search box for filtering services by:

- Name
- Description
- State

In addition, service states have been merged into one list. The switcher buttons at the top of the page have also been changed to tabs to improve user experience of the **Services** page.

(BZ#1657752)

Adding support for firewall zones

The firewall settings on the **Networking** page now supports:

- Adding and removing zones
- Adding or removing services to arbitrary zones and
- Configuring custom ports in addition to **firewalld** services.

(BZ#1678473)

Adding improvements to Virtual Machines configuration

With this update, the RHEL 8 web console includes a lot of improvements in the Virtual Machines page. You can now:

- Manage various types of storage pools
- Configure VM autostart
- Import existing qcow images
- Install VMs through PXE boot
- Change memory allocation
- Pause/resume VMs
- Configure cache characteristics (directsync, writeback)
- Change the boot order

(BZ#1658847)

4.16. VIRTUALIZATION

Windows automatically finds the needed virtio-win drivers

Windows can now automatically find the virtio-win drivers it needs from the driver ISO without requiring the user to select the folder in which they are located.

(BZ#1223668)

KVM supports 5-level paging

With Red Hat Enterprise Linux 8, KVM virtualization supports the 5-level paging feature. On selected host CPUs, this significantly increases the physical and virtual address space that the host and guest systems can use.

(BZ#1526548)

Smart card sharing is now supported on Windows guests with ActivClient drivers

This update adds support for smart card sharing in virtual machines (VMs) that use a Windows guest OS and ActivClient drivers. This enables smart card authentication for user logins using emulated or shared smart cards on these VMs.

(BZ#1615840)

New options have been added for virt-xml

The **virt-xml** utility can now use the following command-line options:

- **--no-define** - Changes done to the virtual machine (VM) by the **virt-xml** command are not saved into persistent configuration.

- **--start** - Starts the VM after performing requested changes.

Using these two options together allows users to change the configuration of a VM and start the VM with the new configuration without making the changes persistent. For example, the following command changes the boot order of the *testguest* VM to network for the next boot, and initiates the boot:

```
virt-xml testguest  --start --no-define --edit --boot network
```

(JIRA:RHELPLAN-13960)

IBM z14 GA2 CPUs supported by KVM

With this update, KVM supports the IBM z14 GA2 CPU model. This makes it possible to create virtual machines on IBM z14 GA2 hosts that use RHEL 8 as the host OS with an IBM z14 GA2 CPU in the guest.

(JIRA:RHELPLAN-13649)

Nvidia NVLink2 is now compatible with virtual machines on IBM POWER9

Nvidia VGPUs that support the NVLink2 feature can now be assigned to virtual machines (VMs) running in a RHEL 8 host on an IBM POWER9 system. This makes it possible for these VMs to use the full performance potential of NVLink2.

(JIRA:RHELPLAN-12811)

CHAPTER 5. BUG FIXES

This part describes bugs fixed in Red Hat Enterprise Linux 8.1 that have a significant impact on users.

5.1. 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)

The **xorg-x11-drv-fbdev**, **xorg-x11-drv-vesa**, and **xorg-x11-drv-vmware** video drivers are now installed by default

Previously, workstations with specific models of NVIDIA graphics cards and workstations with specific AMD accelerated processing units did not display the graphical login window after a RHEL 8.0 Server installation. This issue also impacted virtual machines relying on EFI for graphics support, such as Hyper-V. With this update, the **xorg-x11-drv-fbdev**, **xorg-x11-drv-vesa**, and **xorg-x11-drv-vmware** video drivers are installed by default and the graphical login window is displayed after a RHEL 8.0 and later Server installation.

(BZ#1687489)

Rescue mode no longer fails without displaying an error message

Previously, running rescue mode on a system with no Linux partitions resulted in the installation program failing with an exception. With this update, the installation program displays the error message "You don't have any Linux partitions" when a system with no Linux partitions is detected.

(BZ#1628653)

The installation program now sets the **lvm_metadata_backup** Blivet flag for image installations

Previously, the installation program failed to set the **lvm_metadata_backup** Blivet flag for image installations. As a consequence, LVM backup files were located in the **/etc/lvm/** subdirectory after an image installation. With this update, the installation program sets the **lvm_metadata_backup** Blivet flag, and as a result, there are no LVM backup files located in the **/etc/lvm/** subdirectory after an image installation.

(BZ#1673901)

The RHEL 8 installation program now handles strings from RPM

Previously, when the **python3-rpm** library returned a string, the installation program failed with an exception. With this update, the installation program can now handle strings from RPM.

(BZ#1689909)

The **inst.repo** kernel boot parameter now works for a repository on a hard drive that has a non-root path

Previously, the RHEL 8 installation process could not proceed without manual intervention if the **inst.repo=hd:<device>:<path>** kernel boot parameter was pointing to a repository (not an ISO image) on a hard drive, and a non-root (/) path was used. With this update, the installation program can now propagate any **<path>** for a repository located on a hard drive, ensuring the installation proceeds as normal.

(BZ#1689194)

The **--changesok** option now allows the installation program to change the root password

Previously, using the **--changesok** option when installing Red Hat Enterprise Linux 8 from a Kickstart file did not allow the installation program to change the root password. With this update, the **--changesok** option is successfully passed by Kickstart, and as a result, users specifying the **pwpolicy root –changesok** option in their Kickstart file can now change the root password using the GUI, even if the password has already been set by Kickstart.

(BZ#1584145)

Image Building no longer fails when using **lorax-composer** API

Previously, when using **lorax-composer** API from a subscribed RHEL system, the image building process always failed. Anaconda could not access the repositories, because the subscription certificates from the host are not passed through. To fix the issue update **lorax-composer**, **pykickstart**, and **Anaconda** packages. That will allow to pass supported CDN certificates.

(BZ#1663950)

5.2. SHELLS AND COMMAND-LINE TOOLS

systemd in debug mode no longer produces unnecessary log messages

When using the **systemd** system and service manager in debug mode, **systemd** previously produced unnecessary and harmless log messages that started with:

> "Failed to add rule for system call ..."

With this update, **systemd** has been fixed to no longer produce these unnecessary debug messages.

(BZ#1658691)

5.3. SECURITY

lockdev now runs correctly with SELinux

Previously, the **lockdev** tool could not transition into the **lockdev_t** context even though the SELinux policy for **lockdev_t** was defined. As a consequence, **lockdev** was allowed to run in the ‘unconfined_t’ domain when used by the root user. This introduced vulnerabilities into the system. With this update, the transition into **lockdev_t** has been defined, and **lockdev** can now be used correctly with SELinux in enforcing mode.

(BZ#1673269)

iotop now runs correctly with SELinux

Previously, the **iotop** tool could not transition into the **iotop_t** context even though the SELinux policy for **iotop_t** was defined. As a consequence, **iotop** was allowed to run in the 'unconfined_t' domain when used by the root user. This introduced vulnerabilities into the system. With this update, the transition into **iotop_t** has been defined, and **iotop** can now be used correctly with SELinux in enforcing mode.

(BZ#1671241)

SELinux now properly handles NFS 'crossmnt'

The NFS protocol with the **crossmnt** option automatically creates internal mounts when a process accesses a subdirectory already used as a mount point on the server. Previously, this caused SELinux to check whether the process accessing an NFS mounted directory had a mount permission, which caused AVC denials. In the current version, SELinux permission checking skips these internal mounts. As a result, accessing an NFS directory that is mounted on the server side does not require mount permission.

(BZ#1647723)

An SELinux policy reload no longer causes false ENOMEM errors

Reloading the SELinux policy previously caused the internal security context lookup table to become unresponsive. Consequently, when the kernel encountered a new security context during a policy reload, the operation failed with a false "Out of memory" (ENOMEM) error. With this update, the internal Security Identifier (SID) lookup table has been redesigned and no longer freezes. As a result, the kernel no longer returns misleading ENOMEM errors during an SELinux policy reload.

(BZ#1656787)

Unconfined domains can now use **smc_socket**

Previously, the SELinux policy did not have the allow rules for the **smc_socket** class. Consequently, SELinux blocked an access to **smc_socket** for the unconfined domains. With this update, the allow rules have been added to the SELinux policy. As a result, the unconfined domains can use **smc_socket**.

(BZ#1683642)

Kerberos cleanup procedures are now compatible with **GSSAPIDelegateCredentials** and default cache from **krb5.conf**

Previously, when the **default_ccache_name** option was configured in the **krb5.conf** file, the kerberos credentials were not cleaned up with the **GSSAPIDelegateCredentials** and **GSSAPICleanupCredentials** options set. This bug is now fixed by updating the source code to clean up credential caches in the described use cases. After the configuration, the credential cache gets cleaned up on exit if the user configures it.

(BZ#1683295)

OpenSSH now correctly handles PKCS #11 URIs for keys with mismatching labels

Previously, specifying PKCS #11 URIs with the object part (key label) could prevent OpenSSH from finding related objects in PKCS #11. With this update, the label is ignored if the matching objects are not found, and keys are matched only by their IDs. As a result, OpenSSH is now able to use keys on smart cards referenced using full PKCS #11 URIs.

(BZ#1671262)

SSH connections with VMware-hosted systems now work properly

The previous version of the **OpenSSH** suite introduced a change of the default IP Quality of Service (IPQoS) flags in SSH packets, which was not correctly handled by the VMware virtualization platform.

Consequently, it was not possible to establish an SSH connection with systems on VMware. The problem has been fixed in VMWare Workstation 15, and SSH connections with VMware-hosted systems now work correctly.

(BZ#1651763)

curve25519-sha256 is now supported by default in OpenSSH

Previously, the **curve25519-sha256** SSH key exchange algorithm was missing in the system-wide crypto policies configurations for the OpenSSH client and server even though it was compliant with the default policy level. As a consequence, if a client or a server used **curve25519-sha256** and this algorithm was not supported by the host, the connection might fail. This update of the **crypto-policies** package fixes the bug, and SSH connections no longer fail in the described scenario.

(BZ#1678661)

Ansible playbooks for OSPP and PCI-DSS profiles no longer exit after encountering a failure

Previously, Ansible remediations for the Security Content Automation Protocol (OSPP) and the Payment Card Industry Data Security Standard (PCI-DSS) profiles failed due to incorrect ordering and other errors in the remediations. This update fixes the ordering and errors in generated Ansible remediation playbooks, and Ansible remediations now work correctly.

(BZ#1741455)

Audit **transport=KRB5** now works properly

Prior to this update, Audit KRB5 transport mode did not work correctly. Consequently, Audit remote logging using the Kerberos peer authentication did not work. With this update, the problem has been fixed, and Audit remote logging now works properly in the described scenario.

(BZ#1730382)

5.4. NETWORKING

The kernel now supports destination MAC addresses in **bitmap:ipmac**, **hash:ipmac**, and **hash:mac** IP set types

Previously, the kernel implementation of the **bitmap:ipmac**, **hash:ipmac**, and **hash:mac** IP set types only allowed matching on the source MAC address, while destination MAC addresses could be specified, but were not matched against set entries. As a consequence, administrators could create **iptables** rules that used a destination MAC address in one of these IP set types, but packets matching the given specification were not actually classified. With this update, the kernel compares the destination MAC address and returns a match if the specified classification corresponds to the destination MAC address of a packet. As a result, rules that match packets against the destination MAC address now work correctly.

(BZ#1649087)

The **gnome-control-center** application now supports editing advanced IPsec settings

Previously, the **gnome-control-center** application only displayed the advanced options of IPsec VPN connections. Consequently, users could not change these settings. With this update, the fields in the advanced settings are now editable, and users can save the changes.

(BZ#1697329)

The TRACE target in the **iptables-extensions(8)** man page has been updated

Previously, the description of the **TRACE** target in the **iptables-extensions(8)** man page referred only to the **compat** variant, but Red Hat Enterprise Linux 8 uses the **nf_tables** variant. As a consequence, the man page did not reference the **xtables-monitor** command-line utility to display **TRACE** events. The man page has been updated and, as a result, now mentions **xtables-monitor**.

(BZ#1658734)

Error logging in the **ipset** service has been improved

Previously, the **ipset** service did not report configuration errors with a meaningful severity in the **systemd** logs. The severity level for invalid configuration entries was only **informational**, and the service did not report errors for an unusable configuration. As a consequence, it was difficult for administrators to identify and troubleshoot issues in the **ipset** service's configuration. With this update, **ipset** reports configuration issues as **warnings** in **systemd** logs and, if the service fails to start, it logs an entry with the **error** severity including further details. As a result, it is now easier to troubleshoot issues in the configuration of the **ipset** service.

(BZ#1683711)

The **ipset** service now ignores invalid configuration entries during startup

The **ipset** service stores configurations as sets in separate files. Previously, when the service started, it restored the configuration from all sets in a single operation, without filtering invalid entries that can be inserted by manually editing a set. As a consequence, if a single configuration entry was invalid, the service did not restore further unrelated sets. The problem has been fixed. As a result, the **ipset** service detects and removes invalid configuration entries during the restore operation, and ignores invalid configuration entries.

(BZ#1683713)

The **ipset list** command reports consistent memory for **hash** set types

When you add entries to a **hash** set type, the **ipset** utility must resize the in-memory representation to for new entries by allocating an additional memory block. Previously, **ipset** set the total per-set allocated size to only the size of the new block instead of adding the value to the current in-memory size. As a consequence, the **ip list** command reported an inconsistent memory size. With this update, **ipset** correctly calculates the in-memory size. As a result, the **ipset list** command now displays the correct in-memory size of the set, and the output matches the actual allocated memory for **hash** set types.

(BZ#1714111)

The kernel now correctly updates PMTU when receiving ICMPv6 **Packet Too Big** message

In certain situations, such as for link-local addresses, more than one route can match a source address. Previously, the kernel did not check the input interface when receiving Internet Control Message Protocol Version 6 (ICMPv6) packets. Therefore, the route lookup could return a destination that did not match the input interface. Consequently, when receiving an ICMPv6 **Packet Too Big** message, the kernel could update the Path Maximum Transmission Unit (PMTU) for a different input interface. With this update, the kernel checks the input interface during the route lookup. As a result, the kernel now updates the correct destination based on the source address and PMTU works as expected in the described scenario.

(BZ#1721961)

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)

5.5. KERNEL

tpm2-abrmd-selinux now has a proper dependency on selinux-policy-targeted

Previously, the **tpm2-abrmd-selinux** package had a dependency on the **selinux-policy-base** package instead of the **selinux-policy-targeted** package. Consequently, if a system had **selinux-policy-minimum** installed instead of **selinux-policy-targeted**, installation of the **tpm2-abrmd-selinux** package failed. This update fixes the bug and **tpm2-abrmd-selinux** can be installed correctly in the described scenario.

(BZ#1642000)

All /sys/kernel/debug files can be accessed

Previously, the return value for "Operation not permitted" (EPERM) error remained set until the end of the function regardless of the error. Consequently, any attempts to access certain **/sys/kernel/debug** (debugfs) files failed with an unwarranted EPERM error. This update moves the EPERM return value to the following block. As a result, **debugfs** files can be accessed without problems in the described scenario.

(BZ#1686755)

NICs are no longer affected by a bug in the qede driver for the 41000 and 45000 FastLinQ series

Previously, firmware upgrade and debug data collection operations failed due to a bug in the **qede** driver for the 41000 and 45000 FastLinQ series. It made the NIC unusable. The reboot (PCI reset) of the host made the NIC operational again.

This issue could occur in the following scenarios:

- during the upgrade of Firmware of the NIC using the inbox driver
- during the collection of debug data running the **ethtool -d ethx** command
- while running an **sosreport** command that included **ethtool -d ethx.**
- during the initiation of automatic debug data collection by the inbox driver, such as I/O timeout, Mail Box Command time-out and a Hardware Attention.

To fix this issue, Red Hat released an erratum via Red Hat Bug Advisory (RHBA). Before the release of RHBA, it was recommended to create a case in https://access.redhat.com/support to request for supported fix.

(BZ#1697310)

The generic EDAC GHES driver now detects which DIMM reported an error

Previously, the **EDAC GHES** driver was not able to detect which DIMM reported an error. Consequently, the following error message appeared:

```
DIMM location: not present. DMI handle: 0x<ADDRESS>
```

The driver has been now updated to scan the **DMI (SMBIOS)** tables to detect the specific DIMM that matches the Desktop Management Interface (DMI) handle **0x<ADDRESS>**. As a result, **EDAC GHES** correctly detects which specific DIMM reported a hardware error.

(BZ#1721386)

podman is able to checkpoint containers in RHEL 8

Previously, the version of the Checkpoint and Restore In Userspace (CRIU) package was outdated. Consequently, CRIU did not support container checkpoint and restore functionality, and the **podman** utility failed to checkpoint containers. When running the **podman container checkpoint** command, the following error message was displayed:

```
'checkpointing a container requires at least CRIU 31100'
```

This update fixes the problem by upgrading the version of the CRIU package. As a result, **podman** now supports container checkpoint and restore functionality.

(BZ#1689746)

early-kdump and standard kdump no longer fail if the add_dracutmodules+=earlykdump option is used in dracut.conf

Previously, an inconsistency occurred between the kernel version being installed for **early-kdump** and the kernel version **initramfs** was generated for. As a consequence, booting failed when **early-kdump** was enabled. In addition, if **early-kdump** detected that it was being included in a standard **kdump** initramfs image, it forced an exit. Therefore the standard **kdump** service also failed when trying to rebuild **kdump** initramfs if **early-kdump** was added as a default **dracut** module. As a consequence, **early-kdump** and standard **kdump** both failed. With this update, **early-kdump** uses the consistent kernel name during the installation, only the version differs from the running kernel. Also, the standard **kdump** service will forcibly drop **early-kdump** to avoid image generation failure. As a result, **early-kdump** and standard **kdump** no longer fail in the described scenario.

(BZ#1662911)

The first kernel with SME enabled now succeeds in dumping the vmcore

Previously, the encrypted memory in the first kernel with the active Secure Memory Encryption (SME) feature caused a failure of the **kdump** mechanism. Consequently, the first kernel was not able to dump the contents (vmcore) of its memory. With this update, the **ioremap_encrypted()** function has been added to remap the encrypted memory and modify the related code. As a result, the encrypted first kernel’s memory is now properly accessed, and the vmcore can be dumped and parsed by the crash tools in the described scenario.

(BZ#1564427)

The first kernel with SEV enabled now succeeds in dumping the vmcore

Previously, the encrypted memory in the first kernel with the active Secure Encrypted Virtualization (SEV) feature caused a failure of the **kdump** mechanism. Consequently, the first kernel was not able to dump the contents (vmcore) of its memory. With this update, the **ioremap_encrypted()** function has been added to remap the encrypted memory and modify the related code. As a result, the first kernel’s encrypted memory is now properly accessed, and the vmcore can be dumped and parsed by the crash tools in the described scenario.

(BZ#1646810)

Kernel now reserves more space for SWIOTLB

Previously, when Secure Encrypted Virtualization (SEV) or Secure Memory Encryption (SME) features was enabled in the kernel, the Software Input Output Translation Lookaside Buffer (SWIOTLB) technology had to be enabled as well and consumed a significant amount of memory. Consequently, the capture kernel failed to boot or got an out-of-memory error. This update fixes the bug by reserving extra crashkernel memory for SWIOTLB while SEV/SME is active. As a result, the capture kernel has more memory reserved for SWIOTLB and the bug no longer appears in the described scenario.

(BZ#1728519)

C-state transitions can now be disabled during **hwlatdetect** runs

To achieve real-time performance, the **hwlatdetect** utility needs to be able to disable power saving in the CPU during test runs. This update allows **hwlatdetect** to turn off C-state transitions for the duration of the test run and **hwlatdetect** is now able to detect hardware latencies more accurately.

(BZ#1707505)

5.6. FILE SYSTEMS AND STORAGE

The RHEL 8 installation program now uses the entry ID to set the default boot entry

Previously, the RHEL 8 installation program used the index of the first boot entry as the default, instead of using the entry ID. As a consequence, adding a new boot entry became the default, as it was sorted first and set to the first index. With this update, the installation program uses the entry ID to set the default boot entry, and as a result, the default entry is not changed, even if boot entries are added and sorted before the default.

(BZ#1671047)

The system now boots successfully when SME is enabled with smartpqi

Previously, the system failed to boot on certain AMD machines when the Secure Memory Encryption (SME) feature was enabled and the root disk was using the **smartpqi** driver.

When the boot failed, the system displayed a message similar to the following in the boot log:

```
smartpqi 0000:23:00.0: failed to allocate PQI error buffer
```

This problem was caused by the **smartpqi** driver, which was falling back to the Software Input Output Translation Lookaside Buffer (SWIOTLB) because the coherent Direct Memory Access (DMA) mask was not set.

With this update, the coherent DMA mask is now correctly set. As a result, the system now boots successfully when SME is enabled on machines that use the **smartpqi** driver for the root disk.

(BZ#1712272)

FCoE LUNs do not disappear after being created on the **bnx2fc** cards

Previously, after creating a FCoE LUN on the **bnx2fc** cards, the FCoE LUNs were not attached correctly. As a consequence, FCoE LUNs disappeared after being created on the **bnx2fc** cards on RHEL 8.0. With this update, FCoE LUNs are attached correctly. As a result, it is now possible to discover the FCoE LUNs after they are created on the **bnx2fc** cards.

(BZ#1685894)

VDO volumes no longer lose deduplication advice after moving to a different-endian platform

Previously, the Universal Deduplication Service (UDS) index lost all deduplication advice after moving the VDO volume to a platform that used a different endian. As a consequence, VDO was unable to deduplicate newly written data against the data that was stored before you moved the volume, leading to lower space savings.

With this update, you can now move VDO volumes between platforms that use different endians without losing deduplication advice.

(BZ#1696492)

kdump service works on large IBM POWER systems

Previously, RHEL8 **kdump** kernel did not start. As a consequence, the **kdump initrd** file on large **IBM POWER** systems was not created. With this update, **squashfs-tools-4.3-19.el8** component is added. This update adds a limit (128) to the number of CPUs which the **squashfs-tools-4.3-19.el8** component can use from the available pool (instead of using all the available CPUs). This fixes the running out of resources error. As a result, **kdump** service now works on large **IBM POWER** systems.

(BZ#1716278)

Verbosity debug options now added to nfs.conf

Previously, the **/etc/nfs.conf** file and the **nfs.conf(5)** man page did not include the following options:

- verbosity
- rpc-verbosity

As a consequence, users were unaware of the availability of these debug flags. With this update, these flags are now included in the **[gssd]** section of the **/etc/nfs.conf** file and are also documented in the **nfs.conf(8)** man page.

(BZ#1668026)

5.7. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

Socket::inet_aton() can now be used from multiple threads safely

Previously, the **Socket::inet_aton()** function, used for resolving a domain name from multiple Perl threads, called the unsafe **gethostbyname() glibc** function. Consequently, an incorrect IPv4 address was occasionally returned, or the Perl interpreter terminated unexpectedly. With this update, the **Socket::inet_aton()** implementation has been changed to use the thread-safe **getaddrinfo() glibc** function instead of **gethostbyname()**. As a result, the **inet_aton()** function from Perl **Socket** module can be used from multiple threads safely.

(BZ#1699793, BZ#1699958)

5.8. COMPILERS AND DEVELOPMENT TOOLS

gettext returns untranslated text even when out of memory

Previously, the **gettext()** function for text localization returned the NULL value instead of text when out of memory, resulting in applications lacking text output or labels. The bug has been fixed and now, **gettext()** - returns untranslated text when out of memory as expected.

(BZ#1663035)

The **locale** command now warns about **LOCPATH** being set whenever it encounters an error during execution

Previously, the **locale** command did not provide any diagnostics for the **LOCPATH** environment variable when it encountered errors due to an invalid **LOCPATH**. The **locale** command is now set to warn that **LOCPATH** has been set any time it encounters an error during execution. As a result, **locale** now reports **LOCPATH** along with any underlying errors that it encounters.

(BZ#1701605)

gdb now can read and correctly represent **z** registers in **core** files on aarch64 SVE

Previously, the **gdb** component failed to read **z** registers from **core** files with aarch64 scalable vector extension (SVE) architecture. With this update, the **gdb** component is now able to read **z** registers from **core** files. As a result, the **info register** command successfully shows the **z** register contents.

(BZ#1669953)

GCC rebased to version 8.3.1

The GNU Compiler Collection (GCC) has been updated to upstream version 8.3.1. This version brings a large number of miscellaneous bug fixes.

(BZ#1680182)

5.9. IDENTITY MANAGEMENT

FreeRADIUS now resolves hostnames pointing to IPv6 addresses

In previous RHEL 8 versions of FreeRADIUS, the **ipaddr** utility only supported IPv4 addresses. Consequently, for the **radiusd** daemon to resolve IPv6 addresses, a manual update of the configuration was required after an upgrade of the system from RHEL 7 to RHEL 8. This update fixes the underlying code, and **ipaddr** in FreeRADIUS now uses IPv6 addresses, too.

(BZ#1685546)

The **Nuxwdog** service no longer fails to start the PKI server in HSM environments

Previously, due to bugs, the **keyutils** package was not installed as a dependency of the **pki-core** package. Additionally, the **Nuxwdog** watchdog service failed to start the public key infrastructure (PKI) server in environments that use a hardware security module (HSM). These problems have been fixed. As a result, the required **keyutils** package is now installed automatically as a dependency, and **Nuxwdog** starts the PKI server as expected in environments with HSM.

(BZ#1695302)

The IdM server now works correctly in the FIPS mode

Previously, the SSL connector for Tomcat server was incompletely implemented. As a consequence, the Identity Management (IdM) server with an installed certificate server did not work on machines with the FIPS mode enabled. This bug has been fixed by adding **JSSTrustManager** and **JSSKeyManager**. As a result, the IdM server works correctly in the described scenario.

Note that there are several bugs that prevent the IdM server from running in the FIPS mode in RHEL 8. This update fixes just one of them.

(BZ#1673296)

The KCM credential cache is now suitable for a large number of credentials in a single credential cache

Previously, if the Kerberos Credential Manager (KCM) contained a large number of credentials, Kerberos operations, such as **kinit**, failed due to a limitation of the size of entries in the database and the number of these entries.

This update introduces the following new configuration options to the **kcm** section of the **sssd.conf** file:

- **max_ccaches (integer)**
- **max_uid_ccaches (integer)**
- **max_ccache_size (integer)**

As a result, KCM can now handle a large number of credentials in a single ccache.

For further information on the configuration options, see sssd-kcm man page.

(BZ#1448094)

Samba no longer denies access when using the **sss** ID mapping plug-in

Previously, when you ran Samba on the domain member with this configuration and added a configuration that used the **sss** ID mapping back end to the **/etc/samba/smb.conf** file to share directories, changes in the ID mapping back end caused errors. Consequently, Samba denied access to files in certain cases, even if the user or group existed and it was known by SSSD. The problem has been fixed. As a result, Samba no longer denies access when using the **sss** plug-in.

(BZ#1657665)

Default SSSD time-out values no longer conflict with each other

Previously, there was a conflict between the default time-out values. The default values for the following options have been changed to improve the failover capability:

- dns_resolver_op_timeout - set to 2s (previously 6s)
- dns_resolver_timeout - set to 4s (previously 6s)
- ldap_opt_timeout - set to 8s (previously 6s)

Also, a new **dns_resolver_server_timeout** option, with default value of 1000 ms has been added, which specifies the time out duration for SSSD to switch from one DNS server to another.

(BZ#1382750)

5.10. DESKTOP

systemctl isolate multi-user.target now displays the console prompt

When running the **systemctl isolate multi-user.target** command from GNOME Terminal in a GNOME Desktop session, only a cursor was displayed, and not the console prompt. This update fixes **gdm**, and the console prompt is now displayed as expected in the described situation.

(BZ#1678627)

5.11. GRAPHICS INFRASTRUCTURES

The 'i915' display driver now supports display configurations up to 3×4K.

Previously, it was not possible to have display configurations larger than 2×4K when using the 'i915' display driver in an **Xorg** session. With this update, the 'i915' driver now supports display configurations up to 3×4K.

(BZ#1664969)

Linux guests no longer display an error when initializing the GPU driver

Previously, Linux guests returned a warning when initializing the GPU driver. This happened because Intel Graphics Virtualization Technology –g (GVT -g) only simulates the **DisplayPort** (DP) interface for guest and leaves the ‘EDP_PSR_IMR’ and ‘EDP_PSR_IIR’ registers as default memory-mapped I/O (MMIO) read/write registers. To resolve this issue, handlers have been added to these registers and the warning is no longer returned.

(BZ#1643980)

5.12. THE WEB CONSOLE

It is possible to login to RHEL web console with session_recording shell

Previously, it was not possible for users of the **tlog** shell (which enables session recording) to log in to the RHEL web console. This update fixes the bug. The previous workaround of adding the **tlog-rec-session** shell to **/etc/shells/** should be reverted after installing this update.

(BZ#1631905)

5.13. VIRTUALIZATION

Hot-plugging PCI devices to a pcie-to-pci bridge controller works correctly

Previously, if a guest virtual machine configuration contained a pcie-to-pci-bridge controller that had no endpoint devices attached to it at the time the guest was started, hot-plugging new devices to that controller was not possible. This update improves how hot-plugging legacy PCI devices on a PCIe system is handled, which prevents the problem from occurring.

(BZ#1619884)

Enabling nested virtualization no longer blocks live migration

Previously, the nested virtualization feature was incompatible with live migration. As a consequence, enabling nested virtualization on a RHEL 8 host prevented migrating any virtual machines (VMs) from the host, as well as saving VM state snapshots to disk. This update fixes the described problem, and the impacted VMs are now possible to migrate.

(BZ#1689216)

5.14. SUPPORTABILITY

redhat-support-tool now creates an **sosreport** archive

Previously, the **redhat-support-tool** utility was unable to create an **sosreport** archive. The workaround was running the **sosreport** command separately and then entering the **redhat-support-tool addattachment -c** command to upload the archive. Users can also use the web UI on Customer Portal which creates the customer case and uploads the **sosreport** archive.

In addition, command options such as **findkerneldebugs**, **btextract**, **analyze**, or **diagnose** do not work as expected and will be fixed in a future release.

(BZ#1688274)

CHAPTER 6. TECHNOLOGY PREVIEWS

This part provides a list of all Technology Previews available in Red Hat Enterprise Linux 8.1.

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

6.1. RED HAT ENTERPRISE LINUX SYSTEM ROLES

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 8.1 for SAP Solutions is scheduled to be validated for use with SAP HANA on Intel 64 architecture and IBM POWER8. Other SAP applications and database products, for example, SAP NetWeaver and SAP ASE, can use RHEL 8.1 features. Please consult SAP Notes 2369910 and 2235581 for the latest information about validated releases and SAP support.

(BZ#1660832)

6.2. NETWORKING

TIPC has full support

The Transparent Inter Process Communication (**TIPC**) is a protocol specially designed for efficient communication within clusters of loosely paired nodes. It works as a kernel module and provides a **tipc** tool in **iproute2** package to allow designers to create applications that can communicate quickly and reliably with other applications regardless of their location within the cluster. This feature is now fully supported in RHEL 8.

(BZ#1581898)

eBPF for tc available as a Technology Preview

As a Technology Preview, 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.

(BZ#1699825)

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)

6.3. KERNEL

Control Group v2 available as a Technology Preview 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:

- 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.

- 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 link: https://www.kernel.org/doc/Documentation/cgroup-v2.txt. You can refer to the same link to obtain more information about particular **Control Group v2** controllers.

(BZ#1401552)

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

The **extended Berkeley Packet Filtering (eBPF)**feature is available as a Technology Preview for both networking and tracing. **eBPF** enables the user space to attach custom programs onto a variety of points (sockets, trace points, packet reception) to receive and process data. The feature includes a new system call **bpf()**, which supports creating various types of maps, and also to insert various types of programs into the kernel. Note that the **bpf()** syscall can be successfully used only by a user with the **CAP_SYS_ADMIN** capability, such as a root user. See the **bpf**(2) man page for more information.

(BZ#1559616)

The **bpftrace** is available as a Technology Preview in RHEL 8

Red Hat Enterprise Linux 8 provides the **bpftrace** language as a Technology Preview. **bpftrace** is a high-level tracing language for the **extended Berkeley Packet Filter (eBPF)** feature and is used for very specific tracing tasks while being easy to use. As a result, the user is able to trace arbitrary system data and system performance with only a few lines of code.

(BZ#1687802)

6.4. 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.

NVMe/TCP provides a storage transport option along with the existing NVMe over Fabrics (NVMe-oF) transport, which include Remote Direct Memory Access (RDMA) and Fibre Channel (NVMe/FC).

(BZ#1696451)

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

In Red Hat Enterprise Linux 8.1, 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.

(JIRA:RHELPLAN-1212)

6.5. 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)

6.6. IDENTITY MANAGEMENT

Identity Management JSON-RPC API available as Technology Preview

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

6.7. 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)

6.8. 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 15 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)

Nested virtualization now available on IBM POWER 9

As a Technology Preview, it is now possible to use the nested virtualization features on RHEL 8 host machines running on IBM POWER 9 systems. Nested virtualization enables KVM virtual machines (VMs) to act as hypervisors, which allows for running VMs inside VMs.

Note that nested virtualization also remains a Technology Preview on AMD64 and Intel 64 systems.

Also note that for nested virtualization to work on IBM POWER 9, the host, the guest, and the nested guests currently all need to run one of the following operating systems:

- RHEL 8
- RHEL 7 for POWER 9

(BZ#1505999, BZ#1518937)

Creating nested virtual machines

As a Technology Preview, nested virtualization is available for KVM virtual machines (VMs) in RHEL 8. With this feature, a VM that runs on a physical host can act as a hypervisor, and host its own VMs.

Note that nested virtualization is only available on AMD64 and Intel 64 architectures, and the nested host must be a RHEL 7 or RHEL 8 VM.

(JIRA:RHELPLAN-14047)

CHAPTER 7. DEPRECATED FUNCTIONALITY

This part provides an overview of functionality that has been *deprecated* in Red Hat Enterprise Linux 8.1.

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 .

7.1. 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)

7.2. SOFTWARE MANAGEMENT

The rpmbuild --sign command has been 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)

7.3. SECURITY

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)

7.4. 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)

7.5. KERNEL

Diskless boot has been deprecated

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

(BZ#1748980)

7.6. 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)

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)

7.7. 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)

7.8. 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)

7.9. 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)

7.10. 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)

7.11. 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 8. KNOWN ISSUES

This part describes known issues in Red Hat Enterprise Linux 8.

8.1. 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)

Support secure boot for s390x in the installer

RHEL 8.1 provides support for preparing boot disks for use in IBM Z environments that enforce the use of secure boot. The capabilities of the server and Hypervisor used during installation determine if the resulting on-disk format contains secure boot support or not. There is no way to influence the on-disk format during installation.

Consequently, if you install RHEL 8.1 in an environment that supports secure boot, the system is unable to boot when moved to an environment lacking secure boot support, as it is done in some fail-over scenarios.

To work around this problem, you need to configure the **zipl** tool that controls the on-disk boot format. **zipl** can be configured to write the previous on-disk format even if the environment in which it is run supports secure boot. Perform the following manual steps as root user once the installation of RHEL 8.1 is completed:

1. Edit the configuration file **/etc/zipl.conf**

2. Add a line containing "secure=0" to the section labelled "defaultboot".

   > Example contents of the `zipl.conf` file after the change:

   ```
   [defaultboot]
   defaultauto
   prompt=1
   timeout=5
   target=/boot
   secure=0
   ```

3. Run the **zipl** tool without parameters

After performing these steps, the on-disk format of the RHEL 8.1 boot disk will no longer contain secure boot support. As a result, the installation can be booted in environments that lack secure boot support.

(BZ#1659400)

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)

The default value for the **secure=** boot option is not set to auto

Currently, the default value for the **secure=** boot option is not set to auto. As a consequence, the secure boot feature is not available because the current default is disabled. To work around this problem, manually set **secure=auto** in the **[defaultboot]** section of the **/etc/zipl.conf** file. As a result, the secure boot feature is made available. For more information, see the **zipl.conf** man page.

(BZ#1750326)

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)

8.2. SOFTWARE MANAGEMENT

yum repolist ends on first unavailable repository with skip_if_unavailable=false

The repository configuration option **skip_if_unavailable** is by default set as follows:

```
skip_if_unavailable=false
```

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

Note that it is possible to override this setting in each repository's ***.repo** file.

However, if you want to keep the default settings, you can work around the problem by using **yum repolist** with the following option:

```
--setopt=*.skip_if_unavailable=True
```

(BZ#1697472)

8.3. SHELLS AND COMMAND-LINE TOOLS

Applications using Wayland protocol cannot be forwarded to remote display servers

In Red Hat Enterprise Linux 8.1, 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)

8.4. INFRASTRUCTURE SERVICES

Support for DNSSEC in dnsmasq

The **dnsmasq** package introduces Domain Name System Security Extensions (DNSSEC) support for verifying hostname information received from root servers.

Note that DNSSEC validation in dnsmasq is not compliant with FIPS 140-2. Do not enable DNSSEC in dnsmasq on Federal Information Processing Standard (FIPS) systems, and use the compliant validating resolver as a forwarder on the localhost.

(BZ#1549507)

8.5. SECURITY

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)

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)

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

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

To work around this problem, configure server or client to use the **TLS-1.2** version as the highest **TLS** protocol version available.

(BZ#1681178)

OpenSSL 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)

scap-security-guide PCI-DSS remediation of Audit rules does not work properly

The **scap-security-guide** package contains a combination of remediation and a check that can result in one of the following scenarios:

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

Consequently, during the RHEL 8.1 installation process, scanning of the installed system reports some Audit rules as either failed or errored.

To work around this problem, follow the instructions in the RHEL-8.1 workaround for remediating and scanning with the scap-security-guide PCI-DSS profile Knowledgebase article.

(BZ#1754919)

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)

A utility for security and compliance scanning of containers is not available

In Red Hat Enterprise Linux 7, the **oscap-docker** utility can be used for scanning of Docker containers based on Atomic technologies. In Red Hat Enterprise Linux 8, the Docker- and Atomic-related **OpenSCAP** commands are not available.

To work around this problem, see the Using OpenSCAP for scanning containers in RHEL 8 article on the Customer Portal. As a result, you can use only an unsupported and limited way for security and compliance scanning of containers in RHEL 8 at the moment.

(BZ#1642373)

OpenSCAP does not provide offline scanning of virtual machines and containers

Refactoring of **OpenSCAP** codebase caused certain RPM probes to fail to scan VM and containers file systems in offline mode. For that reason, the following tools were removed from the **openscap-utils** package: **oscap-vm** and **oscap-chroot**. Also, the **openscap-containers** package was completely removed.

(BZ#1618489)

OpenSCAP **rpmverifypackage** does not work correctly

The **chdir** and **chroot** system calls are called twice by the **rpmverifypackage** probe. Consequently, an error occurs when the probe is utilized during an **OpenSCAP** scan with custom Open Vulnerability and Assessment Language (OVAL) content.

To work around this problem, do not use the **rpmverifypackage_test** OVAL test in your content or use only the content from the **scap-security-guide** package where **rpmverifypackage_test** is not used.

(BZ#1646197)

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)

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)

8.6. NETWORKING

The formatting of the verbose output of **arptables** now matches the format of the utility on RHEL 7

In RHEL 8, the **iptables-arptables** package provides an **nftables**-based replacement of the **arptables** utility. Previously, the verbose output of **arptables** separated counter values only with a comma, while **arptables** on RHEL 7 separated the described output with both a space and a comma. As a consequence, if you used scripts created on RHEL 7 that parsed the output of the **arptables -v -L** command, you had to adjust these scripts. This incompatibility has been fixed. As a result, **arptables** on RHEL 8.1 now also separates counter values with both a space and a comma.

(BZ#1676968)

nftables does not support multi-dimensional IP set types

The **nftables** packet-filtering framework does not support set types with concatenations and intervals. Consequently, you cannot use multi-dimensional IP set types, such as **hash:net,port**, with **nftables**.

To work around this problem, use the **iptables** framework with the **ipset** tool if you require multi-dimensional IP set types.

(BZ#1593711)

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)

8.7. KERNEL

The i40iw module does not load automatically on boot

Due to many i40e NICs not supporting iWarp and the **i40iw** module not fully supporting suspend/resume, this module is not automatically loaded by default to ensure suspend/resume works properly. To work around this problem, manually edit the **/lib/udev/rules.d/90-rdma-hw-modules.rules** file to enable automated load of **i40iw**.

Also note that if there is another RDMA device installed with a i40e device on the same machine, the non-i40e RDMA device triggers the **rdma** service, which loads all enabled RDMA stack modules, including the **i40iw** module.

(BZ#1623712)

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)

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** in the kdump kernel command line causes a vmcore generation failure

Due to an existing underlying problem with the **nvme** driver on the 64-bit ARM architectures running on the Amazon Web Services (AWS) cloud platforms, the vmcore generation fails if the **irqpoll** kdump command line argument is provided to the first kernel. Consequently, no vmcore 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 correctly boots and the vmcore is expected to be captured upon the kernel crash.

(BZ#1654962)

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)

8.8. FILE SYSTEMS AND STORAGE

Certain SCSI drivers might sometimes use an excessive amount of memory

Certain SCSI drivers use 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 might be excessive, depending upon the system configuration.

The increased memory usage is caused by memory preallocation in the block layer. Both the multiqueue block device scheduling (BLK-MQ) and the multiqueue SCSI stack (SCSI-MQ) preallocate memory for each I/O request in RHEL 8, leading to the increased memory usage.

(BZ#1698297)

VDO cannot suspend until UDS has finished rebuilding

When a Virtual Data Optimizer (VDO) volume starts after an unclean system shutdown, it rebuilds the Universal Deduplication Service (UDS) index. If you try to suspend the VDO volume using the **dmsetup suspend** command while the UDS index is rebuilding, the suspend command might become unresponsive. The command finishes only after the rebuild is done.

The unresponsiveness is noticeable only with VDO volumes that have a large UDS index, which causes the rebuild to take a longer time.

(BZ#1737639)

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)

8.9. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

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)

8.10. COMPILERS AND DEVELOPMENT TOOLS

The ltrace tool does not report function calls

Because of improvements to binary hardening applied to all RHEL components, the **ltrace** tool can no longer detect function calls in binary files coming from RHEL components. As a consequence, **ltrace** output is empty because it does not report any detected calls when used on such binary files. There is no workaround currently available.

As a note, **ltrace** can correctly report calls in custom binary files built without the respective hardening flags.

(BZ#1618748)

8.11. IDENTITY MANAGEMENT

AD users with expired accounts can be allowed to log in when using GSSAPI authentication

The **accountExpires** attribute that SSSD uses to see whether an account has expired is not replicated to the global catalog by default. As a result, users with expired accounts can log in when using GSSAPI authentication. To work around this problem, the global catalog support can be disabled by specifying **ad_enable_gc=False** in the **sssd.conf** file. With this setting, users with expired accounts will be denied access when using GSSAPI authentication.

Note that SSSD connects to each LDAP server individually in this scenario, which can increase the connection count.

(BZ#1081046)

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

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

(BZ#1729215)

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)

No information about required DNS records displayed when enabling support for AD trust in IdM

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 is displayed. Forest trust to AD is not successful until the required DNS records are added. To work around this problem, run the 'ipa dns-update-system-records --dry-run' command to obtain a list of all DNS records required by IdM. When external DNS for IdM domain defines the required DNS records, establishing forest trust to AD is possible.

(BZ#1665051)

SSSD returns incorrect LDAP group membership for local users

If the System Security Services Daemon (SSSD) serves users from the local files, 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 the problem, either revert the order of the databases where the system is looking up the group membership of users in the **/etc/nsswitch.conf** file, replacing **sss files** with **files sss**, or 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)

Default PAM settings for **systemd-user** have changed in RHEL 8 which may influence SSSD behavior

The Pluggable authentication modules (PAM) stack has changed in Red Hat Enterprise Linux 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.

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

(BZ#1669407)

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)

8.12. DESKTOP

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)

GNOME Shell on Wayland performs slowly when using a software renderer

When using a software renderer, GNOME Shell as a Wayland compositor (**GNOME Shell on Wayland**) does not use a cacheable framebuffer for rendering the screen. Consequently, **GNOME Shell on Wayland** is slow. To workaround the problem, go to the GNOME Display Manager (GDM) login screen and switch to a session that uses the **X11** protocol instead. As a result, the **Xorg** display server, which uses cacheable memory, is used, and **GNOME Shell on Xorg** in the described situation performs faster compared to **GNOME Shell on Wayland**.

(BZ#1737553)

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)

8.13. GRAPHICS INFRASTRUCTURES

radeon fails to reset hardware correctly

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

To work around this problem, 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)

8.14. 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)

8.15. VIRTUALIZATION

Using cloud-init to provision virtual machines on Microsoft Azure fails

Currently, it is not possible to use the **cloud-init** utility to provision a RHEL 8 virtual machine (VM) on the Microsoft Azure platform. To work around this problem, use one of the following methods:

- Use the **WALinuxAgent** package instead of **cloud-init** to provision VMs on Microsoft Azure.
- Add the following setting to the **[main]** section in the **/etc/NetworkManager/NetworkManager.conf** file:

```
[main]
dhcp=dhclient
```

(BZ#1641190)

RHEL 8 virtual machines on RHEL 7 hosts in some cases cannot be viewed in higher resolution than 1920x1200

Currently, 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, cannot use greater resolution than 1920x1200. As a consequence, displaying VMs using those methods only works in resolutions up to 1920x1200, even if the host hardware supports higher resolutions.

(BZ#1635295)

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)

Installing RHEL virtual machines sometimes fails

Under certain circumstances, RHEL 7 and RHEL 8 virtual machines created using the **virt-install** utility fail to boot if the **--location** option is used.

To work around this problem, use the **--extra-args** option instead and specify an installation tree reachable by the network, for example:

```
--extra-args="inst.repo=https://some/url/tree/path"
```

This ensures that the RHEL installer finds the installation files correctly.

(BZ#1677019)

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)

Customizing an ESXi VM using cloud-init and rebooting the VM causes IP setting loss and makes booting the VM very slow

Currently, if the **cloud-init** service is used to modify a virtual machine (VM) that runs on the VMware ESXi hypervisor to use static IP and the VM is then cloned, the new cloned VM in some cases takes a very long time to reboot. This is caused **cloud-init** rewriting the VM's static IP to DHCP and then searching for an available datasource.

To work around this problem, you can uninstall **cloud-init** after the VM is booted for the first time. As a result, the subsequent reboots will not be slowed down.

(BZ#1666961, BZ#1706482)

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, add one of the following strings to the kernel command line of the VM, depending on the CPU firmware:

- For DD2.2: **pseries-rhel7.6.0-sxxm,cap-ibs=fixed-ibs**
- For DD2.3: **pseries-rhel7.6.0-sxxm,cap-ibs=workaround**

(BZ#1732726, BZ#1751054)

IBM POWER virtual machines do not work correctly with zero memory 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'**), the VM cannot boot. Therefore, Red Hat strongly recommends not using IBM POWER VMs with zero-memory 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)

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)

CHAPTER 9. NOTABLE CHANGES TO CONTAINERS

A set of container images is available for Red Hat Enterprise Linux (RHEL) 8.1. Notable changes include:

- Rootless containers are fully supported in RHEL 8.1.
  Rootless containers are containers that are created and managed by regular system users without administrative permissions. This allows users to maintain their identity, including such things as credentials to container registries.

  You can try rootless containers using the podman and buildah commands. For more information:

  - for rootless containers, see Running containers as root or rootless .
  - for **buildah**, see Building container images with Buildah .
  - for **podman**, see Building, running, and managing containers.

- The **toolbox** RPM package is fully supported in RHEL 8.1.
  The **toolbox** command is a utility often used with container-oriented operating systems, such as Red Hat CoreOS. With **toolbox**, you can troubleshoot and debug host operating systems by launching a container that includes a large set of troubleshooting tools for you to use, without having to install those tools on the host system.

  Running the **toolbox** command starts a **rhel-tools** container that provides root access to the host, for fixing or otherwise working with that host.

  For more information see Troubleshooting container hosts with toolbox .

- See the new documentation on Running containers with runlabels.

- The **podman** package has been upgraded to upstream version 1.4.2. For information on features added to **podman** since version 1.0.0, which was used in RHEL 8.0, refer to descriptions of the latest **podman** releases on Github.

CHAPTER 10. INTERNATIONALIZATION

10.1. 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

10.2. 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.

- The **glibc** package updates for multiple locales are now synchronized with the Common Locale Data Repository (CLDR).

APPENDIX A. LIST OF TICKETS BY COMPONENT

Component	Tickets
NetworkManager-libreswan	BZ#1697329
anaconda	BZ#1628653, BZ#1673901, BZ#1671047, BZ#1689909, BZ#1689194, BZ#1584145, BZ#1637472, BZ#1696609, BZ#1672405, BZ#1687747, BZ#1745064, BZ#1659400, BZ#1655523
audit	BZ#1730382
authselect	BZ#1657295
bcc	BZ#1667043
binutils	BZ#1618748, BZ#1644391, BZ#1525406, BZ#1659437
bpftrace	BZ#1687802
chrony	BZ#1685469
cloud-init	BZ#1641190, BZ#1666961
cockpit-appstream	BZ#1658847
cockpit	BZ#1631905, BZ#1678956, BZ#1657752, BZ#1678473, BZ#1666722
corosync	BZ#1693491
criu	BZ#1689746
crypto-policies	BZ#1678661, BZ#1660839
cryptsetup	BZ#1676622
distribution	BZ#1685191, BZ#1657927
dnf-plugins-core	BZ#1722093
dnsmasq	BZ#1549507
dyninst	BZ#1648441
elfutils	BZ#1683705
enscript	BZ#1664366

Component	Tickets
fapolicyd	BZ#1673323
freeradius	BZ#1685546
frr	BZ#1657029
gcc-toolset-9	BZ#1685482
gcc	BZ#1680182
gdb	BZ#1669953, BZ#1187581
gdm	BZ#1678627
glibc	BZ#1663035, BZ#1701605, BZ#1651283, BZ#1577438
gnome-shell-extensions	BZ#1717947
gnome-shell	BZ#1704360
gnome-software	BZ#1668760
gnutls	BZ#1628553
grub2	BZ#1583445, BZ#1723501
initial-setup	BZ#1676439
ipa	BZ#1665051, JIRA:RHELPLAN-15036, BZ#1664719, BZ#1664718, BZ#1719767
ipset	BZ#1683711, BZ#1683713, BZ#1649090
iptables	BZ#1658734, BZ#1676968
kernel-rt	BZ#1678887

Component	Tickets
kernel	BZ#1647723, BZ#1656787, BZ#1649087, BZ#1721386, BZ#1564427, BZ#1686755, BZ#1664969, BZ#1714111, BZ#1712272, BZ#1646810, BZ#1728519, BZ#1721961, BZ#1654962, BZ#1635295, BZ#1706541, BZ#1666538, BZ#1685894, BZ#1643980, BZ#1602962, BZ#1697310, BZ#1593711, BZ#1649647, BZ#1153521, BZ#1694705, BZ#1698297, BZ#1348508, BZ#1748451, BZ#1743456, BZ#1708456, BZ#1710480, BZ#1634343, BZ#1652222, BZ#1687459, BZ#1571628, BZ#1571534, BZ#1685552, BZ#1685427, BZ#1663281, BZ#1664359, BZ#1677215, BZ#1659399, BZ#1665717, BZ#1581898, BZ#1519039, BZ#1627455, BZ#1501618, BZ#1401552, BZ#1495358, BZ#1633143, BZ#1503672, BZ#1505999, BZ#1570255, BZ#1696451, BZ#1665295, BZ#1658840, BZ#1660627, BZ#1569610
kexec-tools	BZ#1662911, BZ#1750278, BZ#1520209, BZ#1710288
keycloak-httpd-client-install	BZ#1553890
kmod-kvdo	BZ#1696492, BZ#1737639
kpatch	BZ#1763780
libcacard	BZ#1615840
libdnf	BZ#1697472
libgnome-keyring	BZ#1607766
libselinux-python-2.8-module	BZ#1666328
libsemanage	BZ#1672638
libssh	BZ#1610883
libstoragemgmt	BZ#1626415
libvirt	BZ#1664592, BZ#1526548, BZ#1528684
linuxptp	BZ#1677217, BZ#1685467
lorax	BZ#1663950, BZ#1709594, BZ#1689140
lvm2	BZ#1649086
mariadb-10.3-module	BZ#1657053

Component	Tickets
mutter	BZ#1737553
nfs-utils	BZ#1668026, BZ#1592011
nginx	BZ#1668717, BZ#1690292
nmstate	BZ#1674456
nss	BZ#1724250, BZ#1645153
openmpi	BZ#1717289
openscap	BZ#1642373, BZ#1618489, BZ#1646197, BZ#1718826, BZ#1709429
openssh	BZ#1683295, BZ#1671262, BZ#1651763, BZ#1744108, BZ#1691045
openssl	BZ#1685470, BZ#1681178, BZ#1749068
oscap-anaconda-addon	BZ#1674001, BZ#1691305
pacemaker	BZ#1715426
pcp	BZ#1685302
pcs	BZ#1619620
perl-IO-Socket-SSL	BZ#1632600
perl-Net-SSLeay	BZ#1632597
perl-Socket	BZ#1699793
php-7.2-module	BZ#1670386
php	BZ#1653109
pki-core	BZ#1695302, BZ#1673296, BZ#1729215
pykickstart	BZ#1637872
python-ply	BZ#1747490
python-wheel	BZ#1731526
python3	BZ#1731424

Component	Tickets
qemu-kvm	BZ#1619884, BZ#1689216, BZ#1651474, BZ#1740002, BZ#1719687, BZ#1651994
redhat-support-tool	BZ#1688274
rhel-system-roles-sap	BZ#1660832
rhel-system-roles	BZ#1691966
rng-tools	BZ#1692435
rpm	BZ#1688849
rsyslog	JIRA:RHELPLAN-10431, BZ#1659383, BZ#1679512, BZ#1614181
rt-tests	BZ#1686494, BZ#1707505, BZ#1666351
ruby-2.6-module	BZ#1672575
s390utils	BZ#1750326
samba	BZ#1638001, JIRA:RHELPLAN-13195
scap-security-guide	BZ#1741455, BZ#1754919, BZ#1750755, BZ#1718839
scap-workbench	BZ#1640715
selinux-policy	BZ#1673269, BZ#1671241, BZ#1683642, BZ#1641631, BZ#1746398, BZ#1673107, BZ#1684103, BZ#1673056
setools	BZ#1672631
setup	BZ#1663556
squashfs-tools	BZ#1716278
sssd	BZ#1448094, BZ#1081046, BZ#1657665, BZ#1652562, BZ#1669407, BZ#1447945, BZ#1382750, BZ#1754871
subscription-manager	BZ#1674337
systemd	BZ#1658691, BZ#1686892, BZ#1640802
systemtap	BZ#1675740

Component	Tickets
tpm2-abrmd-selinux	BZ#1642000
tpm2-tools	BZ#1664498
tuned	BZ#1685585
udica	BZ#1763210, BZ#1673643
valgrind	BZ#1683715
vdo	BZ#1669534
virt-manager	BZ#1677019
virtio-win	BZ#1223668
xorg-x11-drv-qxl	BZ#1642887
xorg-x11-server	BZ#1687489, BZ#1698565
other	BZ#1640697, BZ#1623712, BZ#1659609, BZ#1697896, BZ#1732726, JIRA:RHELPLAN-2542, JIRA:RHELPLAN-13066, JIRA:RHELPLAN-13074, BZ#1731502, BZ#1649493, BZ#1718422, JIRA:RHELPLAN-7109, JIRA:RHELPLAN-13068, JIRA:RHELPLAN-13960, JIRA:RHELPLAN-13649, JIRA:RHELPLAN-12811, BZ#1766186, BZ#1741531, BZ#1721683, BZ#1690207, JIRA:RHELPLAN-1212, BZ#1559616, BZ#1699825, JIRA:RHELPLAN-14047, BZ#1769727, BZ#1642765, JIRA:RHELPLAN-10304, BZ#1646541, BZ#1647725, BZ#1686057, BZ#1748980

APPENDIX B. REVISION HISTORY

0.0-3

Tue Nov 26 2019, Lucie Maňásková (lmanasko@redhat.com)

- Updated the Bug Fixes section.
- Updated the Technology Preview section.
- Added a Known Issue related to **irqpoll** (Kernel).

0.0-2

Thu Nov 14 2019, Lucie Maňásková (lmanasko@redhat.com)

- Added a note that TIPC now has full support.
- Added a note that **bcc-tool** is now supported on x86_64 architectures only.
- Updated Overview with information about live patching for kernel, **kpatch**.
- Updated the the Technology Previews section.

0.0-1

Tue Nov 05 2019, Lucie Maňásková (lmanasko@redhat.com)

- Release of the Red Hat Enterprise Linux 8.1 Release Notes.

0.0-0

Wed Jul 24 2019, Lucie Maňásková (lmanasko@redhat.com)

- Release of the Red Hat Enterprise Linux 8.1 Beta Release Notes.