



Red Hat Enterprise Linux 8

8.2 发行注记

Red Hat Enterprise Linux 8.2 发行注记

Red Hat Enterprise Linux 8 8.2 发行注记

Red Hat Enterprise Linux 8.2 发行注记

法律通告

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

摘要

本发行注记提供了在 Red Hat Enterprise Linux 8.2 中实施的改进和附加功能的高级信息，以及本版本中已知的问题，以及重要的程序错误修复、技术预览、已弃用的功能和其他详情。

目录

对红帽文档提供反馈	3
第 1 章 概述	4
安装程序和镜像创建	4
基础架构服务	4
安全性	4
动态编程语言、网页和数据库服务器	4
编译器工具集	4
Identity Management	4
Web 控制台	5
Desktop	5
原位升级	5
其它资源	5
红帽客户门户网站 Labs	6
第 2 章 构架	7
第 3 章 RHEL 8 中的内容发布	8
3.1. 安装	8
3.2. 软件仓库	8
3.3. 应用程序流	8
3.4. 使用 YUM/DNF 管理软件包	9
第 4 章 RHEL 8.2.1 发行版本	10
4.1. 新特性	10
第 5 章 RHEL 8.2.0 发行版本	12
5.1. 新特性	12
5.2. 对外部内核参数的重要更改	45
5.3. 设备驱动程序	50
5.4. 程序错误修复	52
5.5. 技术预览	65
5.6. 过时的功能	73
5.7. 已知问题	79
第 6 章 国际化	103
6.1. RED HAT ENTERPRISE LINUX 8 国际语言	103
6.2. RHEL 8 国际化的显著变化	103
附录 A. 按组件划分的问题单列表	104
附录 B. 修改历史记录	111

对红帽文档提供反馈

我们感谢您对文档提供反馈信息。请让我们了解如何改进文档。要做到这一点：

通过 Jira 提交反馈（需要帐户）

1. 登录到 [Jira](#) 网站。
2. 单击顶部导航栏中的 **Create**。
3. 在 **Summary** 字段中输入描述性标题。
4. 在 **Description** 字段中输入您对改进的建议。包括到文档相关部分的链接。
5. 点对话框底部的 **Create**。

第 1 章 概述

安装程序和镜像创建

在 RHEL 8.2 中，您可以在软件包安装前注册您的系统、附加 RHEL 订阅并从 Red Hat Content Delivery Network(CDN)安装。您还可以在安装过程中将您的系统注册到 Red Hat Insights。交互式 GUI 安装以及自动 Kickstart 安装支持这些新功能。

如需更多信息，请参阅 [第 5.1.1 节“安装程序和镜像创建”](#)。

基础架构服务

Tuned 系统调优工具已更新至版本 2.13，它增加了对架构独立调整和多个包含指令的支持。

如需更多信息，请参阅 [第 5.1.4 节“基础架构服务”](#)。

安全性

系统范围的加密策略 现在支持 **自定义**。管理员现在可以定义完整策略或仅修改某些值。

RHEL 8.2 包含 **setools-gui** 和 **setools-console-analyses** 软件包，它们提供了 SELinux-policy 分析和数据流划分的工具。

SCAP Security Guide 现在提供了一个符合 Australian Cyber Security Centre (ACSC) **Essential Eight** Maturity Model 的配置集。

如需更多信息，请参阅 [第 5.1.5 节“安全性”](#)。

动态编程语言、网页和数据库服务器

以下组件的更新版本现在作为新的模块流提供：

- Python 3.8
- maven 3.6

详情请查看 [第 5.1.10 节“动态编程语言、网页和数据库服务器”](#)。

编译器工具集

RHEL 8.2 中更新了以下编译器工具集：

- GCC Toolset 9
- clang 和 LLVM Toolset 9.0.1
- Rust Toolset 1.41
- Go Toolset 1.13

如需更多信息，请参阅 [第 5.1.11 节“编译器和开发工具”](#)。

Identity Management

身份管理引入了一个新的命令行工具：**状况检查**。**状况检查** 可帮助用户发现可能会影响其 IdM 环境适用性的问题。

身份管理现在支持用于安装和管理的 Ansible 角色和模块。在这个版本中，基于 IdM 的解决方案的安装和配置更容易。

如需更多信息，请参阅 [第 5.1.12 节“Identity Management”](#)。

Web 控制台

Web 控制台已被重新设计，以使用 PatternFly 4 用户界面系统设计。

在 web 控制台中添加了会话超时以提高安全性。

如需更多信息，请参阅 [第 5.1.15 节 “Web 控制台”](#)。

Desktop

GNOME Classic 环境中的工作空间交换器已被修改。交换器现在位于底部栏的右侧，它被设计为一个横向的缩略图。点击相关的缩略图可以在不同工作区间切换。

Direct Rendering Manager(DRM)内核图形子系统已更新到上游 Linux 内核版本 5.3。与之前的版本相比，这个版本提供了很多改进，包括对新 GPU 和 APU 以及各种驱动程序更新的支持。

原位升级

RHEL 7 原位升级到 RHEL 8

支持的原位升级路径为：

- 64 位 Intel、IBM POWER 8(little endian)和 IBM Z 架构上，从 RHEL 7.9 升级到 RHEL 8.2
- 从 RHEL 7.6 到 RHEL 8.2 的架构需要内核版本 4.14：64 位 ARM、IBM POWER 9(little endian)和 IBM Z(Structure A)。请注意，这些架构在 RHEL 7 中被完全支持，但从 RHEL 7.7 开始不再发布次发行版本更新。

如需更多信息，请参阅[支持的 Red Hat Enterprise Linux 原位升级路径](#)。有关执行原位升级的步骤，请参阅[从 RHEL 7 升级到 RHEL 8](#)。

主要改进包括：

- 现在，您可以使用额外的自定义软件仓库从 RHEL 7 原位升级到 RHEL 8。也可以在没有任何 Red Hat Subscription Manager 的情况下升级。
- 您可以创建自己的执行器，以使用 Leapp 程序迁移您的自定义或第三方应用程序。

详情请参阅[自定义 Red Hat Enterprise Linux 原位升级](#)。

如果您使用 CentOS Linux 7 或 Oracle Linux 7，您可以在升级到 RHEL 8 之前使用支持的 **convert2rhel** 实用程序将操作系统转换为 RHEL 7。具体步骤请参阅[从基于 RPM 的 Linux 发行版转换到 RHEL](#)。

从 RHEL 6 原位升级到 RHEL 8

要从 RHEL 6.10 升级到 RHEL 8.2，请参阅 [从 RHEL 6 升级到 RHEL 8 部分](#)。

如果您使用 CentOS Linux 6 或 Oracle Linux 6，您可以在升级到 RHEL 8 之前使用不支持的 **转换2rhel** 实用程序将操作系统转换为 RHEL 6。具体步骤请查看[如何从 CentOS Linux 或 Oracle Linux 转换到 RHEL](#)。

其它资源

- [Red Hat Enterprise Linux technology capabilities and limits](#) 包括了与其他版本系统相比的 Red Hat Enterprise Linux 8 的**能力和限制**。
- 有关 Red Hat Enterprise Linux **生命周期** 的详情请查看 [Red Hat Enterprise Linux 生命周期文档](#)。
- [软件包清单文档](#) 为 RHEL 8 **提供软件包列表**。

- RHEL 7 和 RHEL 8 的主要区别包括在使用 [RHEL 8 时的注意事项](#)。
- 文档 [从 RHEL 7 升级到 RHEL 8](#) 提供了有关如何执行 [从 RHEL 7 到 RHEL 8 的原位升级](#) 的说明。
- Red Hat Insights 服务可让您主动发现、检查并解决已知的技术问题，所有 RHEL 订阅都可以使用它。有关如何安装 Red Hat Insights 客户端并将您的系统注册到该服务的说明，请查看 [Red Hat Insights 入门页面](#)。

红帽客户门户网站 Labs

红帽客户门户网站 Labs 是客户门户网站的一个部分中的一组工具，地址为 <https://access.redhat.com/labs/>。红帽客户门户网站 Labs 中的应用程序可帮助您提高性能、快速解决问题、发现安全问题以及快速部署和配置复杂应用程序。一些最常用的应用程序有：

- [Registration Assistant](#)
- [Product Life Cycle Checker](#)
- [Kickstart Generator](#)
- [Kickstart Converter](#)
- [Red Hat Satellite Upgrade Helper](#)
- [Red Hat Code Browser](#)
- [JVM Options Configuration Tool](#)
- [Red Hat CVE Checker](#)
- [Red Hat Product Certificates](#)
- [Load Balancer Configuration Tool](#)
- [Yum Repository Configuration Helper](#)

第 2 章 构架

Red Hat Enterprise Linux 8.2 带有内核版本 4.18.0-193，它支持以下构架：

- AMD 和 Intel 64 位构架
- 64 位 ARM 架构
- IBM Power Systems, Little Endian
- 64-bit IBM Z

请确定为每个构架购买正确的订阅。如需更多信息,请参阅 [Red Hat Enterprise Linux 入门 - 附加构架](#)。有关可用订阅列表, 请查看客户门户网站中的 [订阅使用](#)。

第 3 章 RHEL 8 中的内容发布

3.1. 安装

Red Hat Enterprise Linux 8 使用 ISO 镜像安装。AMD64、Intel 64 位、64 位 ARM、IBM Power Systems 和 IBM Z 架构有两种类型的 ISO 镜像：

- 二进制 DVD ISO：包含 BaseOS 和 AppStream 软件仓库的完整安装镜像,并允许您在没有附加软件仓库的情况下完成安装。



注意

二进制 DVD ISO 镜像大于 4.7 GB，因此它可能不适用于单层 DVD。当使用二进制 DVD ISO 镜像创建可引导安装介质时，建议使用双层 DVD 或者 USB 设备。您还可以使用 Image Builder 工具创建自定义的 RHEL 镜像。有关镜像构建器的更多信息，请参阅 [编写自定义的 RHEL 系统镜像](#) 文档。

- 引导 ISO：用来引导到安装程序的最小引导 ISO 镜像。这个选项需要访问 BaseOS 和 AppStream 软件仓库来安装软件包。软件仓库是二进制 DVD ISO 镜像的一部分。

有关下载 ISO 镜像、创建安装介质和完成 [RHEL 安装的步骤](#)，请参阅 [执行标准 RHEL 8 安装文档](#)。有关自动 Kickstart 安装和其他高级主题，请参阅 [执行高级 RHEL 8 安装文档](#)。

3.2. 软件仓库

Red Hat Enterprise Linux 8 由两个主要软件仓库发布：

- BaseOS
- AppStream

两个软件仓库都需要一个基本的 RHEL 安装，所有 RHEL 订阅都包括它们。

BaseOS 仓库的内容旨在提供底层操作系统功能的核心组件，为所有安装提供基础操作系统的基础。这部分内容采用 RPM 格式，它的支持条款与之前的 RHEL 版本相似。有关通过 BaseOS 发布的软件包列表，请查看 [软件包清单](#)。

Application Stream 仓库的内容包括额外的用户空间应用程序、运行时语言和数据库来支持各种工作负载和使用案例。Application Streams（应用程序流）以熟悉的 RPM 格式，作为 RPM 格式的扩展，名为 *模块 (modules)*，或作为 Software Collections（软件集合）。有关 AppStream 中可用软件包列表，请查看 [软件包清单](#)。

另外，所有 RHEL 订阅都可以使用 CodeReady Linux Builder 软件仓库。它为开发人员提供了额外的软件包。不支持包括在 CodeReady Linux Builder 存储库中的软件包。

有关 RHEL 8 软件仓库的详情，请查看 [软件包清单](#)。

3.3. 应用程序流

Red Hat Enterprise Linux 8 引进了应用程序流（Application Streams）的概念。和操作系统软件包相比，现在为用户空间组件提供了多个版本且会更频繁地进行更新。这为自定义 Red Hat Enterprise Linux 提供了更大的灵活性，不会影响平台或特定部署的基本稳定性。

作为 Application Streams 提供的组件可打包为模块（module）或 RPM 软件包，并通过 RHEL 8 中的 AppStream 软件仓库提供。每个 Application Stream 组件都有其特定的生命周期，可能和 RHEL 8 的生命周期相同或更短。详情请查看 [Red Hat Enterprise Linux 生命周期](#)。

模块是代表逻辑单元的软件包集合：应用程序、语言堆栈、数据库或一组工具。这些软件包被一同构建、测试并发布。

模块流代表 Application Stream 组件的版本。例如，PostgreSQL 数据库服务器的几个流（版本）位于带有默认的 **postgresql: 10 流**的 **postgresql** 模块中。在系统中只能安装一个模块流。不同的容器可以使用不同的版本。

详细的模块命令，请参考 [安装、管理和删除用户空间组件文档](#)。有关 AppStream 中可用模块列表，请查看 [软件包清单](#)。

3.4. 使用 YUM/DNF 管理软件包

在 Red Hat Enterprise Linux 8 中，YUM 工具确保安装软件，该工具基于 DNF 技术。我们有意坚持使用 **yum** 术语，以便与以前的 RHEL 主要版本保持一致。但是，如果您键入 **dnf** 而不是 **yum**，则命令可以正常工作，因为 **yum** 是 **dnf** 的别名以实现兼容性。

如需了解更多详细信息，请参阅以下文档：

- [安装、管理和删除用户空间组件](#)
- [使用 RHEL 8 时的注意事项](#)

第 4 章 RHEL 8.2.1 发行版本

红帽在次发行版本 (8.Y) 之间会每季度提供一次 Red Hat Enterprise Linux 8 的内容。每季度发布的版本使用第三个数字(8.Y.1)进行编号。RHEL 8.2.1 发行版中的新功能如下所述。

4.1. 新特性

JDK Mission Control rebase 到版本 7.1.1

HotSpot JVM 的 JDK Mission Control(JMC)配置集 (由 **jmc:rhel8** 模块流提供) 升级至 RHEL 8.2.1 版本 7.1.1。

这个版本包括很多程序错误修复和增强, 包括 :

- 多规则优化
- 基于标准小工具工具包(SWT)的新 JOverflow 视图.
- 新的 flame 图形视图
- 一种使用高动态范围(HDR)直方图的延迟视觉化的新方法

jmc:rhel8 模块流有两个配置集 :

- **common** 配置文件, 其安装了整个 JMC 应用程序
- **内核 配置文件**, 仅安装核心 Java 库(**jmc-core**)

要安装 **jmc:rhel8** 模块流的 **common** 配置集, 请使用 :

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

将配置文件名称更改为 **core**, 以仅安装 **jmc-core** 软件包。

(BZ#1792519)

Rust Toolset rebase 到版本 1.43

Rust Toolset 已更新至 1.43 版本。主要变更包括 :

- 现在, 有用的行号包含在被调用的 **Option** 和 **Result** panic 消息中。
- 扩展了对子片模式匹配的支持。
- **match!** 宏提供返回布尔值的模式匹配。
- **项目** 片段可插入到特征、模仿和 **extern** 块中。
- 改进了围绕原语的类型推断。
- 浮点和整数的相关常数。

要安装 Rust Toolset 模块, 以 **root** 用户身份运行以下命令 :

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

有关使用信息，请参阅 [使用 Rust Toolset](#) 文档。

(BZ#1811997)

容器 registry 现在支持 skopeo sync 命令

在这个版本中，用户可以使用 **skopeo sync** 命令同步容器 registry 和本地 registry。**skopeo sync** 命令可用于同步本地容器 registry 镜像，以及填充在 Air-gapped 环境中运行的 registry。

skopeo sync 命令需要单独指定源(--src)和目的地(--dst)传输。可用的源和目标传输是 **docker**（托管在容器注册表上的存储库）和 **dir**（本地目录路径中的目录）。源传输还包括 include **yaml**（本地 YAML 文件路径）。有关使用 **skopeo** 同步的详情，请参考 **skopeo-sync** man page。

(BZ#1811779)

配置文件 container.conf 现已正式发布

在这个版本中，用户和管理员可以为容器引擎指定默认配置选项和命令行标志。容器引擎读取 **/usr/share/containers/containers.conf** 和 **/etc/containers/containers.conf** 文件（若存在）。在无根模式中，容器引擎读取 **\$HOME/.config/containers/containers.conf** 文件。

containers.conf 文件中指定的字段会覆盖之前读取 **containers.conf** 文件中的默认选项，以及选项。**container.conf** 文件在 Podman 和 Buildah 之间共享，并替换 **libpod.conf** 文件。

(BZ#11826486)

现在，您可以从 registry 服务器登录和注销

在这个版本中，您可以使用 **skopeo login** 和 **skopeo logout** 命令来登录并从指定的 registry 服务器注销。**skopeo login** 命令读取来自标准输入的用户名和密码。用户名和密码也可以使用 **--username**（或 **-u**）和 **--password**（或 **-p**）选项进行设置。

您可以通过设置 **--authfile** 标志来指定身份验证文件的路径。默认路径为 **\${XDG_RUNTIME_DIR}/containers/auth.json**。有关使用 **skopeo login** 和 **skopeo logout** 的详情，请查看 **skopeo-login** 和 **skopeo-logout** man page。

(JIRA:RHELPLAN-47311)

现在，您可以重置 podman 存储

在这个版本中，用户可以使用 **podman system reset** 命令将 **podman** 存储重置回初始状态。**podman system reset** 命令将移除所有 pod、容器、镜像和卷。如需更多信息，请参阅 **podman-system-reset** man page。

(JIRA:RHELPLAN-48941)

第 5 章 RHEL 8.2.0 发行版本

5.1. 新特性

这部分论述了红帽企业 Linux 8.2 中引入的新功能和主要增强。

5.1.1. 安装程序和镜像创建

注册您的系统、附加 RHEL 订阅以及使用红帽 CDN 安装的功能

在 RHEL 8.2 中，您可以在软件包安装前注册您的系统、附加 RHEL 订阅并从 Red Hat Content Delivery Network(CDN)安装。交互式 GUI 安装以及自动的 Kickstart 安装支持这个特性。优点包括：

- 使用较小的引导 ISO 镜像文件不需要下载更大的 Binary DVD ISO 镜像文件。
- CDN 使用最新的软件包，它会在安装后马上得到全面订阅和最新的系统。安装后无需安装软件包更新。
- 注册是在软件包安装之前执行的，从而缩短和简化的安装过程。
- 提供了对 Red Hat Insights 集成的支持。

(BZ#1748281)

在安装过程中将您的系统注册到 Red Hat Insights

在 RHEL 8.2 中，您可以在安装过程中将您的系统注册到 Red Hat Insights 中。交互式 GUI 安装以及自动的 Kickstart 安装支持这个特性。

优点包括：

- 更容易识别、排列和解决问题，防止业务操作受到影响。
- 通过预测分析主动识别和修复对安全性、性能、可用性和稳定性的威胁。
- 避免在您的环境中出现问题和计划外停机。

(BZ#1746391)

镜像构建器现在提供 cloud-init 支持来创建 Azure 镜像

在这个版本中，Cloud-init 支持由 Image Builder 创建的 Azure 镜像。因此，客户可以使用创建具有快速配置功能的内部镜像以及添加自定义数据的功能。

(BZ#1754711)

添加了新的 kickstart 命令：rh sm 和 zi pl

在这个版本中，添加了以下 kickstart 命令：

- **rhsm**：在安装过程中，使用 **rhsm** 命令在红帽注册系统。
- **zipl**：使用 **zipl** 命令指定 IBM Z 系统上的 **zipl** 配置。

(BZ#1972214)

5.1.2. 软件管理

user-Agent 标头字符串现在包含从 /etc/os-release 文件中读取的信息

在这个版本中，DNF 发出的 HTTP 请求通常包含 **User-Agent** 标头字符串，使用从 `/etc/os-release` 文件读取的信息进行扩展。

要获得更多信息，请参阅 `dnf.conf(5)` man page 中的 `user_agent`。

([BZ#1676891](#))

现在，所有 `dnf-automatic.timer` 计时器单元都会默认使用实时时钟

在以前的版本中，`dnf-automatic.timer` 计时器单元使用单例时钟，这会导致系统引导后激活时间无法预计。在这个版本中，计时器单元在早上 6 点到早上 7 点之间运行。如果系统在该时间段内关闭，则定时器单元将在系统引导后一小时内激活。

([BZ#1754609](#))

`createrepo_c` 工具现在跳过元数据包含禁止控制字符的软件包

为确保有效的 XML，软件包元数据不得包含任何控制字符，但：

- 横向标签
- `newline` 字符
- 回车字符

在这个版本中，`createrepo_c` 工具不会在新创建的仓库中包含元数据的软件包，并返回以下错误消息：

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

([BZ#1743186](#))

5.1.3. Shell 和命令行工具

`opencv` rebase 到版本 3.4.6

`opencv` 软件包已升级到上游版本 3.4.6。主要变更包括：

- 支持新的 Open CL 参数，如 `OPENCV_OPENCL_BUILD_EXTRA_OPTIONS` 和 `OPENCV_OPENCL_DEVICE_MAX_WORK_GROUP_SIZE`。
- `objdetect` 模块现在支持 QR 代码检测算法。
- 多种新方法，如 `MatSize::dims` 或 `VideoCapture::getBackendName`。
- 多个新功能，如 `draw FrameAxes` 或 `getVersionMajor`。
- 各种性能得到了改进，包括在使用 SSE3 指令时 `GaussianBlur` 函数 `v_load_deinterleave` 和 `v_store_interleave` 内部函数的改进。

([BZ#1694647](#))

5.1.4. 基础架构服务

graphviz-python3 现在在 CRB 仓库中发布

在这个版本中，将 **graphviz-python3** 软件包添加到 RHEL 8。软件包提供使用 Python 的 Graphviz 图形可视化软件所需的绑定。

请注意, **graphviz-python3** 软件包包括在不支持的 [CodeReady Linux Builder 仓库\(CRB\)](#)。

[\(BZ#1704875\)](#)

tuned rebase 到版本 2.13.0

tuned 软件包已升级到上游版本 2.13.0。主要改进包括：

- 添加了架构依赖的调优框架。
- 添加了对多个 include 指令的支持。
- 更新了 **sap-hana**、**latency-performance** 和 **realtime** 配置文件中的调优。

[\(BZ#1738250\)](#)

powertop rebase 到版本 2.11

powertop 软件包已升级到 2.11 版本，它提供了以下显著变化：

- 支持 EHL、TGL、ICL/ICX 平台

[\(BZ#1716721\)](#)

BIND 现在支持 .GeoIP2 而不是 GeoLite Legacy GeoIP

GeoLite Legacy GeoIP 库不再被 BIND 支持。在这个版本中，GeoLite Legacy GeoIP 已被 GeoIP2 替代，它以 **libmaxminddb** 数据格式提供。

请注意，新格式可能需要一些配置更改，而且格式还不支持以下旧的 GeoIP 访问控制列表(ACL)设置：

- GeoIP 网络速度
- GeoIP 机构
- ISO 3166 Alpha-3 国家代码

[\(BZ#1564443\)](#)

stale-answer 现在在 DDoS 攻击时提供旧的缓存记录

在以前的版本中，分布式服务(DDoS)攻击会导致权威服务器因为 SERVFAIL 错误而失败。在这个版本中，**stale -answer** 功能提供过期的记录，直到获取了全新的响应。

要启用或禁用 **service -stale** 功能，请使用以下任一功能：

- 配置文件
- 远程控制频道(rndc)

[\(BZ#1664863\)](#)

BIND rebase 到版本 9.11.13

bind 软件包已升级到 9.11.13 版本。主要变更包括：

- 添加了 **tcp-highwater** 统计变量。此变量显示运行期间记录的最大并发 TCP 客户端。
- 添加了基于 **SipHash-2-4** 的 DNS Cookies(RFC 7873)算法。
- 无论 **minimal-responses** 配置选项是如何设置的，都将返回用于根启动查询的粘合地址。
- **named-checkconf** 命令现在确保 **DNS64** 网络前缀的有效性。
- 当 **trusted-keys** 和 **managed-keys** 语句都为同一名称配置时，每个 RFC 5011 的自动滚动不再会失败。相反，会记录警告消息。
- 现在，如果 **dig** 和 **nslookup** 实用程序中未在终端上运行（例如，在脚本中）中，国际化域名 (IDN)处理会被默认禁用。可以使用 **+idnin** 和 **+idnout** 选项打开 **dig** 中的 IDN 处理。

(BZ#1704328)

5.1.5. 安全性

RHEL 8 现在包含 DISA STIG 配置集

安全技术实施指南(STIG)是国防部(DISA)发布的一组基线建议，以增强可能易受损害的信息系统和软件的安全性。这个版本包括这个安全策略的配置集和 Kickstart 文件。在这个版本中，用户可以检查系统的合规性，修复系统是否合规，以及安装与 Red Hat Enterprise Linux 8 的 DISA STIG 兼容的系统。

(BZ#1755447)

crypto-policies 现在可以自定义

在这个版本中，您可以调整任何策略级别的特定算法或协议，或将新的完整策略文件设置为当前系统范围的加密策略。这可让管理员根据不同场景的要求自定义系统范围的加密策略。

RPM 软件包应当将其提供的策略存储在 **/usr/share/crypto-policies/policies** 目录中。**/etc/crypto-policies/policies/policies** 目录包含本地自定义策略。

如需更多信息，请参阅 **update-crypto-policies(8)** 手册页中的 **Custom Policies** 部分和 **update-crypto-policies(8)** 手册页中的 **Crypto Policy Definition Format** 部分。

(BZ#1690565)

SCAP 安全指南现在支持 ACSC Essential Eight

scap-security-guide 软件包现在提供澳大利亚网络安全中心(ACSC)Essential Eight 合规性配置集和相应的 Kickstart 文件。在这个改进中，用户可以安装一个符合这个安全基点的系统。另外，您可以使用 **OpenSCAP** 套件使用 ACSC 定义的最低安全控制规范来检查安全合规和补救。

(BZ#1755194)

现在，提供了用于容器安全性和合规性扫描的 **oscap-podman**

这个 **openscap** 软件包更新引进了新的实用程序，用于容器的安全性和合规性扫描。The **oscap-podman** 工具提供等效的 the **oscap-docker** 实用程序，用于在 RHEL 7 中扫描容器和容器镜像。

(BZ#1642373)

现在，**se troubleshoot** 可以分析和响应 **execmem** 访问拒绝

在这个版本中引进了新的 **setroubleshoot** 插件。插件可以分析 **execmem** 访问拒绝(AVC)，并提供相关建议。因此，**setroubleshoot** 现在可以建议在允许访问时切换布尔值，或者在没有布尔值允许访问时报告问题。

(BZ#1649842)

新软件包：**setools-gui** 和 **setools-console-analyses**

RHEL 8 现在引进了作为 RHEL 7 一部分的 **setools-gui** 软件包。图形工具有助于检查关系和数据流，尤其是在具有高度专业 SELinux 策略的多级别系统中。使用 **setools-gui** 软件包中的 **apol** 图形工具，您可以检查和分析 SELinux 策略的各个方面。**setools-console-analyses** 软件包中的工具可让您分析域转换和 SELinux 策略信息流。

(BZ#1731519)

SELinux 中的受限用户现在可以管理用户会话服务

在以前的版本中，受限制的用户无法管理用户会话服务。因此，它们无法执行 **systemctl --user** 或 **busctl --user** 命令，或者在 RHEL web 控制台中正常工作。在这个版本中，受限用户可以管理用户会话。

(BZ#1727887)

lvmbusd 服务现在被 SELinux 限制

lvmbusd 服务向逻辑卷管理器(LVM)提供 D-Bus API。在以前的版本中，**lvmbusd** 守护进程无法过渡到 **lvm_t** 上下文，即使定义了 **lvm_t** 的 SELinux 策略。因此，**lvm dbusd** 守护进程在 **unconfined_service_t** 域中被执行，将 **lvmbusd** 标记为 **unconfined**。在这个版本中，**lvm dbusd** 可执行文件定义了 **lvm_exec_t** 上下文，现在可以在强制模式下正确使用 **lvmbusd**。

(BZ#1726166)

semanage 现在支持列出和修改 SCTP 和 DCCP 端口。

在以前的版本中，**semanage** 端口 只允许列出和修改 TCP 和 UDP 端口。在这个版本中，增加了对 **semanage port** 的 SCTP 和 DCCP 协议支持。现在，管理员可以检查两台机器是否能够通过 SCTP 通信，并完全启用 SCTP 功能来成功部署基于 SCTP 的应用程序。

(BZ#1563742)

semanage 导出 现在显示与 permissive 域相关的自定义

在这个版本中，**semanage** 实用程序是 SELinux 的 **polycoreutils** 软件包的一部分，可以显示与 permissive 域相关的自定义。系统管理员现在可以使用 **semanage export** 命令在机器之间传输许可的本地修改。

(BZ#1417455)

udica 可以将 SELinux 拒绝生成的新 allow 规则添加到现有容器策略

当在 **udica** 实用程序生成的策略下运行的容器触发 SELinux 拒绝时，**udica** 现在可以更新策略。新参数 **-a** 或 **--append-rules** 可以用来附加 AVC 文件中的规则。

(BZ#1732704)

新的 SELinux 类型使服务能够受限制

在这个版本中引进了新的 SELinux 类型，使以下服务能够以 SELinux enforcing 模式作为受限服务运行，而不是在 **unconfined_service_t** 域中运行：

- **lldpd** 现在运行 as **lldpd_t**
- **rrdcached** 现在作为 **rrdcached_t**运行
- **stratisd** 现在作为 **stratisd_t**运行
- **timedatex** 现在作为 **timedatex_t**运行

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

Clevis 能够列出给定 LUKS 设备的策略

在这个版本中，**c levis luks list** 命令列出 给定 LUKS 设备的 PBD 策略。这样更易于查找正在使用和固定配置的 Clevis pins 信息，如 Tang 服务器地址、**tpm2 策略** 的详情和 SSS 阈值。

([BZ#1766526](#))

Clevis 提供报告密钥状态和重新绑定过期密钥的新命令

clevis luks report 命令现在提供了报告特定绑定的键是否需要轮转的简单方法。Tang 服务器中的常规密钥轮转提高了 Network-Bound Disk 加密(NBDE)部署的安全性，因此客户端应提供已过期密钥的检测。如果密钥已过期，Clevis 建议使用 **clevis luks regen** 命令，该命令使用当前密钥将过期的密钥插槽重新关联。这大大简化了密钥轮转的过程。

([BZ#1564559](#), [BZ#1564566](#))

Clevis 现在可以提取用于绑定 LUKS 设备中特定插槽的密码短语

在这个版本中，您可以提取用于在 LUKS 设备中绑定特定插槽的密码短语。在以前的版本中，如果删除了 LUKS 安装密码，Clevis 无法执行 LUKS 管理任务，如重新加密、使用用户密码启用新密钥插槽，并在管理员需要更改 **ss** 阈值时重新绑定 Clevis。在这个版本中引进了 **clevis luks pass** 命令，显示用于绑定特定插槽的密码短语。

([BZ#1436780](#))

Clevis 现在提供改进的对在引导时解密多个 LUKS 设备的支持

clevis 软件包已经更新，为在启动时解密多个 LUKS 加密设备提供更好的支持。在进行这一改进之前，管理员必须对系统配置执行复杂的更改，以便 Clevis 在引导时正确解密多个设备。在这个发行版本中，您可以使用 **clevis luks bind** 命令并通过 The **dracut -fv --regenerate-all** 命令更新 **initramfs** 来设置解密。

如需了解更多详细信息，请参阅[使用基于策略的解密配置加密卷的自动解锁部分](#)。

([BZ#1784524](#))

openssl-pkcs11 rebase 到 0.4.10

Open **ssl-pkcs11** 软件包已升级到上游版本 0.4.10，它比之前的版本提供了很多程序错误修复和增强。**openssl-pkcs11** 软件包通过引擎接口提供对 PKCS #11 模块的访问。新版本引入的主要更改有：

- 如果在加载 ECDSA 私钥时不提供与私钥对应的公钥对象，则引擎将从匹配的证书加载公钥（如果存在）。
- 您可以使用通用 PKCS #11 URI（如 example **pkcs11:type=public**），因为 **openssl-pkcs11** 引擎搜索与给定 PKCS #11 URI 匹配的所有令牌。
- 只有在单个设备与 URI 搜索匹配时，系统才会尝试使用 PIN 登录。这可防止因为将 PIN 提供给所有匹配的令牌而导致身份验证失败。

- 在访问设备时，**openssl-pkcs11** 引擎现在使用 **RSA_FLAG_FIPS_METHOD** 标志标记 RSA 方法结构。在 FIPS 模式中，OpenSSL 要求在 RSA 方法结构中设置标志。请注意，引擎无法检测设备是否经过 FIPS 认证。

(BZ#1745082)

rsyslog rebase 到 8.1911.0

rsyslog 工具已升级到上游版本 8.1911.0，它提供很多程序错误修复和增强。以下列表包含显著的改进：

- New **omhttp** 模块允许您通过 HTTP REST 接口发送消息。
- 文件输入模块得到了增强，以提高稳定性、错误报告和截断检测。
- 新的 **action.resumeIntervalMax** 参数可与任何操作一起使用，允许将重试间隔增长上限为指定的值。
- TLS 的新 **StreamDriver.PermitExpiredCerts** 选项允许连接，即使证书已过期。
- 现在，您可以根据配置的外部文件内容暂停和恢复输出。这在另一端始终接受消息并在无法全部处理消息时静默丢弃消息时很有用。
- 文件输出模块的错误报告已被改进，现在包含真实的文件名和有关错误原因的更多信息。
- 磁盘队列现在运行多线程，以提高性能。
- 您可以设置更严格的 TLS 操作模式：检查 extend **KeyUsage** 证书字段，并对 **CN/SAN** 证书字段进行更严格的检查。

(BZ#1740683)

rsyslog 现在提供通过 HTTP REST 接口进行通信的 omhttp 插件

在这个版本中，您可以使用 **new omhttp** 插件生成与服务兼容的输出，如 Ceph 存储平台、Amazon S3) 和 Grafana Loki。这个新的 HTTP 输出模块提供了可配置的 REST 路径和消息格式，支持多种批处理格式、压缩和 TLS 加密。

如需了解更多详细信息，请参阅使用 **rsyslog-doc** 软件包安装在系统上的 `/usr/share/doc/rsyslog/html/configuration/modules/omhttp.html` 文件。

(BZ#1676559)

rsyslog 中的 omelasticsearch 现在支持 rebindinginterval

此 **rsyslog** 软件包更新引进了对在 **omelasticsearch** 模块中设置定期重新连接时间的支持。您可以根据您的场景设置这个参数来提高在将记录发送到 Elasticsearch 节点集群时的性能。**rebindinginterval** 参数的值指示 **rsyslog** 关闭连接并建立新的节点时提交到节点的操作数量。默认值为 **-1**，表示 **rsyslog** 不会重新建立连接。

(BZ#1692073)

rsyslog mmkubernetes 现在提供元数据缓存过期时间

在这个版本中，您可以使用 **mmkubernetes** 模块的两个新参数来设置元数据缓存过期时间。这样可确保从 **mmkubernetes** 静态缓存中删除删除的 Kubernetes 对象。**cacheentryttl** 参数的值指示缓存条目的最长期限（以秒为单位）。**cacheexpireinterval** 参数具有以下值：

- **-1** 用于禁用缓存扩展检查

- 0 用于启用缓存扩展检查
- 大于 0 用于常规缓存扩展检查（以秒为单位）

(BZ#1692072)

Audit rebase 到版本 3.0-0.14

审计软件包已升级到上游版本 3.0-0.14，它与之前的版本相比提供了很多程序错误修复和增强，最重要的是：

- 添加了一个选项来解释 syslog 插件中的字段
- 将 **30-ospp-v42.rules** 文件分成更细化的文件
- 将示例规则移到 **/usr/share/audit/sample-rules/** 目录中
- 修复了用于远程日志的 Audit KRB5 传输模式

(BZ#1757986)

Audit 现在包含内核 v5.5-rc1 的许多改进

除了 Linux 内核外，还包含与 Audit 子系统相关的大部分增强功能、漏洞修复和清理，并在 4.18 和 5.5-rc1 版本之间引入。以下列表重点介绍了重要变化：

- 更广泛的使用 **exe** 字段进行过滤
- 支持 v3 命名空间功能
- 在远程文件系统中进行过滤的改进
- 修正 the **gid** 过滤器规则
- 修复无用的内存损坏和内存泄漏
- 事件记录的关联改进
- 清理 **fanotify** 接口、审计配置选项和 **syscall** 接口
- 修复扩展验证模块(EVM)返回值
- 修复和清理几种记录格式
- 虚拟文件系统(VFS)审核的简化和修复

(BZ#1716002)

fapolicyd rebase 到 0.9.1-2

提供 RHEL 应用白名单的 **fapolicyd** 软件包已升级到上游版本 0.9.1-2。重要的程序错误修复和增强包括：

- 进程标识已修复。
- 现在，主题部分和对象部分被严格放在规则中。这两个组件都用冒号分隔，它们包含所需的权限（执行、打开、任意）。
- 主题和对象属性已合并。

- 新规则格式如下：

```
DECISION PERMISSION SUBJECT : OBJECT
```

例如：

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

([BZ#1759895](#))

sudo rebase 到 1.8.29-3.el8

sudo 软件包已升级到上游版本 1.8.29-3，它提供很多程序错误修复和增强。新版本引入的主要更改有：

- **sudo** 现在将可插拔验证模块(PAM)消息写入用户的终端（如果可用），而不是标准输出或标准错误输出。这可防止对 PAM 输出和发送到文件和管道的命令输出造成可能的混淆。
- LDAP 和 SSSD 的 **notBefore** 和 **notAfter** 选项现在可以正常工作，并可以使用 **sudo -l** 命令正确显示。
- 现在，在从 LDIF 转换为 **sudoers** 和 JSON 格式时，**sudoers** 命令会拒绝非 LDAP 数据交换格式(LDIF)输入。
- 使用 **sudoers** 的新 **log_allowed** 和 **log_denied** 设置，您可以禁用允许和拒绝命令的日志和审核。
- 现在，您可以将 **sudo** 与 **-g** 选项搭配使用，以指定与任何目标用户组匹配的组，即使 **runas_spec** 规格中不存在任何组。在以前的版本中，只有在组与目标用户的主组匹配时才可以这样做。
- 修复了一个阻止 **sudo** 将主机名与 **sssd.conf** 中的 **ipa_hostname** 匹配（如果指定）值的错误。
- 现在，解决了一个允许 **sudo** 用户在 **Runas** 规格不允许使用 **ALL** 关键字进行 **root** 访问时以 **root** 身份运行命令的漏洞(CVE-2019-14287)。
- permissive **sudoers** 条目（例如使用 **ALL** 关键字）的未知用户和组 ID 现已禁用。您可以使用 **runas_allow_unknown_id** 设置(CVE-2019-19232)启用它。

([BZ#1733961](#))

pam_namespace 模块现在允许为 tmpfs 指定额外的挂载选项

现在可以在 **/etc/security/namespace.conf** 配置文件中使用时使用 **nosuid**、**noexec** 和 **nodev** 挂载选项，来分别禁用 **setuid** 位效果、禁用运行的可执行文件，并防止文件被解释为已挂载的 **tmpfs** 文件系统上的字符或块设备。

其他挂载选项在 **tmpfs(5)** man page 中指定。

([BZ#1252859](#))

pam_faillock 现在可以从 faillock.conf 配置文件读取设置

pam_faillock 模块是可插拔身份验证模块(PAM)的一部分，现在可以从位于 **/etc/security/faillock.conf** 的配置文件读取设置。这样更易于在身份验证失败时设置帐户锁定，为此功能提供用户配置文件，并通过编辑 **faillock.conf** 文件来处理不同的 PAM 配置。

([BZ#1537242](#))

5.1.6. 网络

用户空间应用程序现在可以检索内核选择的 netns id

用户空间应用程序可以请求内核选择新的 **netns** ID 并将其分配给网络命名空间。有了这个增强，用户可以在向内核发送 **RTM_NETNSID netlink** 消息时指定 **NLM_F_ECHO** 标志。然后，内核将 **netlink** 消息发回给用户。此消息包括设置为内核所选值的 **netns** ID。因此，用户空间应用程序现在可以使用可靠的选项来识别内核选择的 **netlink** ID。

(BZ#1763661)

firewalld 被 rebase 到版本 0.8

firewalld 软件包已更新至 0.8 版本。主要变更包括：

- 这个 **firewalld** 版本包括版本 0.7.0 以来的所有 bug 修复。
- **firewalld** 现在将 **libnftables** JSON 接口用于 **nftables** 子系统。这可以提高规则应用的性能和可靠性。
- 在服务定义中，新的 **helper** 元素取代了 **模块**。
- 此版本允许自定义帮助者使用标准帮助程序模块。

(BZ#1740670)

ndptool 现在可以在 IPv6 标头中指定目标地址

在这个版本中，**ndptool** 实用程序可以通过在 IPv6 标头中指定地址，将邻居请求(NS)或邻居公告(NA)消息发送到特定目的地。因此，可以将消息发送到本地链路地址以外的地址。

(BZ#1697595)

nftables 现在支持多维 IP 设置类型

在这个版本中，**nf tables** 数据包过滤框架支持设置具有串联和间隔的类型。因此，管理员不再需要临时解决方案来创建多组 IP 设置类型。

(BZ#1593711)

nftables rebase 到版本 0.9.3

nftables 软件包已升级到上游版本 0.9.3，它提供很多程序错误修复和增强：

- 在 **libnftables** 库中添加了一个 JSON API。此库提供了一个高级别接口，可用于管理来自第三方应用的 **nftables** 规则集。要在 Python 中使用新 API，请安装 **python3-nftables** 软件包。
- 声明支持 IP 前缀和范围，如 **192.0.2.0/24** 和 **192.0.2.0-192.0.2.30**。
- 添加了对操作系统指纹的支持，以根据猜测的操作系统标记数据包。详情请查看 **nft(8)man** page 中的 **osf 表达式** 部分。
- 添加了透明代理支持，将数据包重定向到本地套接字，而不以任何方式更改数据包标头。详情请查看 **nft(8)man** page 中的 **tproxy 语句** 部分。
- 默认情况下，**nft** 在创建 **nft** 链时显示优先级集的文本名称。要查看标准优先级数字值，请使用 **-y** 选项。
- 添加了安全标记支持。

- 改进了对动态集更新的支持，以设置数据包路径的更新。
- 添加了对传输标头端口匹配的支持。

有关显著变化的更多信息，请在更新前阅读上游发行注记：

- <https://lore.kernel.org/netfilter-devel/20190624164910.defehs5gjqzqnir@salvia/>
- <https://lore.kernel.org/netfilter-devel/20190819115807.myv6owxzbj2bthd@salvia/>
- <https://lore.kernel.org/netfilter-devel/20191202211737.xvmd6e6xxj4xvvlj@salvia/>

(BZ#1643192)

firewalld 服务的规则现在可以将连接跟踪帮助器用于在非标准端口上运行的服务

firewalld 服务中的用户定义的帮助程序现在可以使用标准内核帮助程序模块。这可让管理员创建 **firewalld** 规则，将连接跟踪帮助器用于在非标准端口上运行的服务。

(BZ#1733066)

whois 软件包现在可用

在这个版本中，**whois** 软件包在 RHEL 8.2.0 中可用。因此，现在可以检索特定域名或 IP 地址的信息。

(BZ#1734183)

现在完全支持 tc 的 eBPF

流量控制(tc)内核子系统和 **tc** 工具可以附加扩展 Berkeley Packet 过滤(eBPF)程序作为数据包分类器，以及用于入口和出站队列规则的操作。这启用了内核网络数据路径中的可编程数据包处理。RHEL 8.2 现在完全支持 **tc** 的 eBPF（以前作为技术预览提供）。

(BZ#1755347)

5.1.7. 内核

RHEL 8.2 中的内核版本

Red Hat Enterprise Linux 8.2 带有内核版本 4.18.0-193。

另请参阅[外部内核参数的重要变化](#)和[设备驱动程序](#)。

(BZ#1797671)

RHEL 8.2 的扩展 Berkeley Packet 过滤器

Extended Berkeley Packet Filter(eBPF)是一个内核中的虚拟机,允许在可访问有限功能的受限沙箱环境中在内核空间中执行代码。虚拟机执行类特殊的装配代码。**eBPF** 字节码首先加载到内核，然后进行验证，通过即时编译到原生物理代码转换，然后虚拟机执行代码。

红帽提供大量使用 **eBPF** 虚拟机的组件。每个组件处于不同的开发阶段，因此目前并不完全支持所有组件。在 RHEL 8.2 中，支持以下 **eBPF** 组件：

- **TheBPF Compiler Collection(BCC)** 工具软件包，这是使用 **eBPF** 虚拟机创建高效内核跟踪程序的用户空间集合。**BCC** 提供用于使用 **eBPF** 的 I/O 分析、网络和监控 Linux 操作系统的工具。
- **BCC** 库，它允许开发与 **BCC** 工具软件包中相似的工具。

- **eBPF for Traffic Control (tc)** 功能，可在内核网络数据路径中启用可编程数据包处理。

所有其他 eBPF 组件都作为技术预览提供，除非有特定组件被显示为受支持。

以下显著的 eBPF 组件当前作为技术预览提供：

- **bpfftrace** 追踪语言
- **TheeXpress 数据路径(XDP)** 功能

如需有关技术预览组件的更多信息，请参阅 [技术预览](#)。

([BZ#1780124](#))

Intel® Omni-Path 架构(OPA)主机软件

Red Hat Enterprise Linux 8.2 完全支持 Intel Omni-Path 架构(OPA)主机软件。Intel OPA 为在集群环境中的计算和 I/O 节点之间的高性能数据传输（高带宽、高消息率、低延迟）提供主机 Fabric Interface (HFI) 硬件初始化和设置。

([BZ#1833541](#))

RHEL 8 现在完全支持 control Group v2

控制组 v2 机制是统一的层次结构控制组。控制组 v2 以分层方式组织进程，并以受控和可配置的方式在层次结构中分发系统资源。

与之前的版本不同，控制组 v2 只有一个层次结构。这个单一层次结构使 Linux 内核能够：

- 根据拥有者的角色对进程进行分类。
- 解决多个分级冲突策略的问题。

控制组 v2 支持大量控制器。其中一些示例如下：

- CPU 控制器规定了 CPU 周期的分布。这个控制器实现：
 - 常规调度策略的权重和绝对带宽限制模型。
 - 用于实时调度策略的绝对带宽分配模型。
- cpuset 控制器将处理器和/或内存放置限制为在 **cpuset** 接口文件中指定的资源。
- 内存控制器规定了内存分布。目前，会追踪以下类型的内存用量：
 - Userland memory - 页面缓存和匿名内存。
 - 内核数据结构，如密度和内节点。
 - TCP 套接字缓冲。
- I/O 控制器规定了 I/O 资源的分配。
- 回写控制器与 Memory 和 I/O 控制器交互，并且特定于 **Control Group v2**。

以上信息基于 [Control Group v2](#) 上游文档。您可以参照同一链接来获取有关特定控制组 v2 控制器的更多信息。

请注意，上游文档中提及的所有功能尚未在 RHEL 8 中实施。

(BZ#1401552)

随机化空闲列表：提高直接映射内存缓存的性能和使用

在这个版本中，您可以启用页面分配器来随机化自由列表，并提高直接映射的内存端缓存的平均利用率。内核命令行选项 **page_alloc.shuffle** 允许页面分配器随机化空闲列表并将布尔值标志设置为 **True**。**sysfs** 文件位于 **/sys/module/page_alloc/parameters/shuffle** 中，读取标志状态，缓冲可用列表，从而缓存 Dynamic Random Access Memory(DRAM)，并减少 DRAM 和持久内存之间的延迟带。因此，在通用服务器平台上提供了容量较高的持久内存。

(BZ#1620349)

TPM 用户空间工具已更新至最后一个版本

tpm2-tools 用户空间工具已更新至 3.2.1 版本。这个版本提供了几个程序错误修正，特别是与平台配置寄存器代码和手动页面清理相关的内容。

(BZ#1725714)

C620 排序 PCH 芯片组现在支持 Intel Trace Hub 功能

在这个版本中，在 C620-series Platform Controller Hub(PCH)（也称为 Lewisburg PCH）中添加了对 Intel Trace Hub(TH)的硬件支持。具有 C620 序列的用户现在可以使用 Intel TH。

(BZ#1714486)

perf 工具现在支持 CLX-AP 和 CPX 处理器的每个中断事件聚合

在这个版本中，**perf** 工具支持针对具有多个死锁的一些 Intel CPU 进行每个 die 事件计数。要启用此模式，请在 Xeon Cascade **Lake-AP(CLX-AP)**和 **Cooper Lake(CPX)**系统处理器之外添加 **--per-die** 选项。因此，这个更新会检测到结束之间的任何冲突。**perf stat** 命令捕获事件计数并显示输出结果，如下所示：

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

(BZ#1660368)

IBM Z 上的 `crashkernel=auto` 阈值会降低

现在，在 IBM Z 系统上，**crashkernel=auto** 内核命令行参数的较低阈值从 4G 降至 1G。这个实现允许 IBM Z 与 AMD64 和 Intel 64 系统的阈值一致，在 **崩溃内核=auto** 的较低阈值上共享相同的保留策略。因此，崩溃内核可以在内存不足 4GB 的系统上为 **kdump** 自动保留内存。

(BZ#1780432)

numactl manual 条目阐明内存用量输出

在这个 RHEL 8 发行版本中，**num actl** 的 man page 明确提到内存用量信息只反映系统中的驻留页面。这种添加的原因是消除了用户在内存用量信息与常驻页面还是虚拟内存相关的潜在混淆。

(BZ#1730738)

kexec-tools 文档现已更新为包含 Kdump FCoE 目标支持

在这个发行版本中，**/usr/share/doc/kexec-tools/supported-kdump-targets.txt** 文件已更新，以包含 Kdump Fibre Channel over Ethernet(FCoE)目标支持。现在，用户可以更好地了解 FCoE 目标支持中 **kdump** 崩溃转储机制的状态和详情。

(BZ#1690729)

固件辅助转储现在支持 PowerNV

现在 PowerNV 平台支持固件辅助转储(**fadump**)机制。IBM POWER9 FW941 固件版本及更新的版本支持这个功能。系统故障时，**fadump** 和 **vmcore** 文件也会导出 **opalcore** 文件。**opalcore** 文件包含分类时有关 OpenPOWER Abstraction Layer(OPAL)内存状态的信息。**opalcore** 文件有助于调试基于 OPAL 的系统崩溃。

(BZ#1524687)

kernel-rt 源树现在与最新的 RHEL 8 树匹配

kernel-rt 源已更新为使用最新的 RHEL 内核源树。实时补丁集也更新至最新的上游 v5.2.21-rt13 版本。这两个更新都提供很多程序错误修正和增强。

(BZ#1680161)

rngd 现在能够以非 root 权限运行

随机数生成器守护进程(**rngd**)检查随机性来源提供的数据是否足够随机，然后将数据存储在内核的随机数熵池中。在这个版本中，**rngd** 可以使用非 root 用户权限运行，以增强系统安全性。

(BZ#1692435)

现在，在 POWER 9 中 RHEL 8.2 及更新的版本支持虚拟持久性内存

当在 IBM POWER9 硬件上运行带有 PowerVM 管理程序的 RHEL 8.2 或更高版本主机时，主机现在可以使用 Virtual Persistent Memory(vPMEM)功能。使用 vPMEM 时，数据会在应用程序间持久保留，并在物理服务器关闭前进行分区重启。因此，重启使用 vPMEM 的工作负载要快得多。

您的系统必须使用 vPMEM 必须满足以下要求：

- 硬件管理控制台(HMC)V9R1 M940 或更高版本
- 固件级别 FW940 或更高版本
- E980 系统固件 FW940 或更高版本
- L922 系统固件 FW940 或更高版本
- PowerVM 级别 V3.1.1

请注意，当前在 RHEL 8 中使用 vPMEM 存在几个已知问题。详情请查看以下知识库文章：

- [热插拔/拔下 pmem 内存可能会导致 POWER9 上的内核 panic](#)
- [引导捕获内核需要很长时间，使用 vPMEM 命名空间作为 kdump/fadump 的转储目标](#)

(BZ#1859262)

5.1.8. 文件系统和存储

LVM 现在支持 dm-writecache 缓存方法

LVM 缓存卷现在除了提供现有的 **dm-cache** 方法外，还提供 **dm-writecache** 缓存方法。

dm-cache

这个方法可通过在更快的卷上缓存数据来加快频繁使用数据的访问速度。这个方法会缓存读写操作。

dm-writocache

这个方法只缓存写操作。较快速的卷通常是 SSD 或持久内存(PMEM)磁盘，首先存储写入操作，然后在后台将其迁移到较慢的磁盘。

要配置缓存方法，请使用 `--type cache` 或 `--type writocache` 选项以及 `lvconvert` 实用程序。

如需更多信息，请参阅[启用缓存来提高逻辑卷性能](#)。

(BZ#1600174)

VDO async 策略现在兼容 ACID

在这个版本中，VDO **async** 写入模式与 Atomicity、Consistency、Isolation、Durability(ACID)兼容。如果在 VDO 以 **async** 模式写入数据时系统意外停止，则恢复的数据现在始终一致。

由于 ACID 合规性，**sync** 的性能现在低于上一个版本。要恢复原始性能，您可以将 VDO 卷的写入模式改为 **async-unsafe** 模式，这不兼容 ACID。

如需更多信息，请参阅[选择 VDO 写模式](#)。

(BZ#1657301)

现在，您可以导入 VDO 卷

The **vdo** 工具现在可让您导入系统中当前没有注册的现有 VDO 卷。要导入 VDO 卷，请使用 `vdo import` 命令。

另外，您可以使用 `vdo import` 命令修改 VDO 卷的通用唯一标识符(UUID)。

(BZ#1713749)

mountstats 和 nfsiostat 的输出中现在提供了新的 每个错误 计数器

NFS 客户端系统可使用的一个小的支持性功能：`nfs-utils` 中的 `mountstats` 和 `nfsiostat` 命令的输出有一个 **per-op** 错误计数。在这个版本中，这些工具可以显示 **每个错误** 计数和百分比，以帮助缩小 NFS 客户端计算机上特定 NFS 挂载点上的问题。请注意，这些新统计数据取决于 Red Hat Enterprise Linux 8.2 内核中的内核更改。

(BZ#1719983)

XFS 现在提供了带有 cgroup 感知的 writeback IO

在这个版本中，XFS 支持使用 **cgroup** 感知的 writeback IO。通常，**c group** writeback 需要底层文件系统明确支持。直到现在，XFS 上的 writeback IO 仅仅是 **root cgroup** 的属性。

(BZ#1274406)

FUSE 文件系统现在实施 `copy_file_range ()`

`copy_file_range ()` 系统调用为文件系统提供了一种实施高效数据复制机制的方法。在这个版本中，使用用户空间(FUSE)框架中的 Filesystem 的 GlusterFS 利用了这个机制。由于 FUSE 文件系统的读/写功能涉及多个数据副本，使用 `copy_file_range ()` 可以显著提高性能。

(BZ#1650518)

现在，`mountstats` 和 `nfsiostat` 命令提供了对每个应用程序的统计信息的支持

NFS 客户端系统现在具有支持功能：`/proc/self/mountstats` 文件具有 **每个错误** 计数器。在这个版本中，在每个 **op** 统计行下，第九个数字表示以 `status` 值小于后零完成的操作数。此状态值表示错误。如需更多信息，请参阅 `nfs-utils` 中对 `mountstats` 和 `nfsiostat` 程序的更新，其中显示了这些新错误计数。

(BZ#1636572)

`/proc/self/mountstats` 文件中提供了新的挂载统计信息 `lease_time` 和 `lease_expired`

NFSv4.x 客户端系统提供一项支持功能。`/proc/self/mountstats` 文件在以 `nfsv4:` 开头的行末尾有 `lease_time` 和 `lease_expired` 字段：`lease_time` 字段指示 NFSv4 租用时间内的秒数。`lease_expired` 字段指示自租用过期起的秒数；如果租期未过期，则为 0。

(BZ#1727369)

意外删除 (Surprise removal) NVMe 设备

在这个版本中，您可以在不预先通知操作系统的情况下从 Linux 操作系统中意外删除 NVMe 设备。这可提高 NVMe 设备的可用性，因为不需要额外的步骤来准备设备以按顺序删除，从而可以避免服务器停机时间以确保服务器可用。

注意以下几点：

- 意外删除 NVMe 设备需要 `kernel-4.18.0-193.13.2.el8_2.x86_64` 或更高版本。
- 要成功意外删除 NVMe 设备，可能需要硬件平台或软件的额外要求。
- 不支持意外删除对系统操作很重要的 NVMe 设备。例如：您无法意外删除包含操作系统或者 swap 分区的 NVMe 设备。

(BZ#1634655)

5.1.9. 高可用性和集群

仅在不影响其他资源的情况下禁用资源的新命令选项

有时，只有在这不会影响其他资源时才需要禁用资源。在具有复杂的资源关系时，这可能无法通过手动设置来完成。要满足这个需要，`pcs resource disable` 命令现在支持以下选项：

- `pcs resource disable --simulate`：显示禁用指定资源的效果，而不更改群集配置
- `pcs resource disable --safe`：仅在没有任何方式影响任何其他资源（如从一个节点迁移到另一个节点）时禁用指定的资源
- `pcs resource disable --safe --no-strict`：仅在没有其他资源停止或降级时禁用指定资源

此外，还引进了 `pcs resource secure-disable` 命令，作为 `pcs resource disable --safe` 的别名。

(BZ#1631519)

显示资源关系的新命令

新的 `pcs resource relationship` 命令允许您显示树结构中的群集资源之间的关系。

(BZ#1631514)

新命令可显示主站点和恢复站点集群的状态

如果您已将集群配置为用作恢复站点，则可以使用 `pcs dr` 命令将该集群配置为恢复站点集群。然后，您可以使用 `pcs dr` 命令显示主站点集群和恢复站点集群的状态。

(BZ#1676431)

现在，在列出限制时默认隐藏已过期的资源限制

列出资源约束不再默认显示过期的限制。要包含过期的 constraints，请使用 **pcs constraint** 命令的 **--all** 选项。这将列出已过期的限制，注意显示中的限制和 **相关规则（扩展）**。

(BZ#1442116)

Pacemaker 支持将资源配置为在清理节点关闭时保持停止

当集群节点关闭时，Pacemaker 的默认响应是停止在该节点上运行的所有资源，并在其它位置恢复这些资源。有些用户希望只在失败时具有高可用性，并希望将清理关闭视为调度的中断。要解决这个问题，Pacemaker 现在支持 **shutdown-lock** 和 **shutdown-lock-limit** 集群属性，以指定在关闭时节点上活跃的资源应该一直停止，直到节点下一次重新加入为止。现在，用户可以根据调度的中断使用干净的关闭，而无需手动干预。有关将资源配置为在干净节点关闭中保持停止的详情，请参考 [链接：将资源配置为在清理节点关闭时保持停止](#)。

(BZ#1712584)

支持在单一节点中运行集群环境

仅配置了一个成员的群集现在可以在集群环境中启动和运行资源。这允许用户为使用单一节点进行备份的多节点群集配置单独的灾难恢复站点。请注意，只有一个节点的群集本身不容错。

(BZ#1700104)

5.1.10. 动态编程语言、网页和数据库服务器

新模块：python38

RHEL 8.2 引入了 Python 3.8，它由新模块 **python38** 和 **ubi8/python-38** 容器镜像提供。

与 Python 3.6 相比，显著改进包括：

- 新的 Python 模块，如 **contextvars**、**dataclasses** 或 **importlib.resources**
- 新语言特性，如分配表达式（所谓的 walrus operator **::=**）或仅位置参数
- 改进了关于 **breakpoint ()** 内置功能、**=** 格式字符串规格的开发人员体验，以及 Python 和扩展模块的调试和非调试构建之间的兼容性
- 性能改进
- 改进了对可选静态类型提示的支持
- 为更轻松地调试添加了格式化字符串文字(f-strings)的 **=** specifier
- 软件包的更新版本，如 **pip**、**requests** 或 **Cython**

Python 3.8 和为其构建的软件包可以与 Python 3.6 在同一系统上并行安装。

请注意，**python38** 模块没有包括为 **python 36** 模块提供的系统工具（RPM、DNF 和 SELinux 等）相同的二进制绑定。

要从 **python38** 模块安装软件包，请使用：


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

python38:3.8 模块流将自动启用。

要运行解释器，例如：

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

如需更多信息，请参阅 [安装和使用 Python](#)。

请注意，红帽将继续对 Python 3.6 提供支持，直至 RHEL 8 生命周期结束。Python 3.8 的生命周期较短，请参阅 [RHEL 8 应用程序流生命周期](#)。

(BZ#1747329)

mod_wsgi 安装中的更改

在以前的版本中，当用户尝试使用 `yum install mod_wsgi` 命令安装 `mod_wsgi` 模块时，总是会安装 `python3-mod_wsgi` 软件包。RHEL 8.2 引入了 Python 3.8，作为 Python 3.6 的补充。在这个版本中，您需要指定要安装的 `mod_wsgi` 版本，否则返回错误消息。

要安装 `mod_wsgi` 的 Python 3.6 版本：

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

要安装 `mod_wsgi` 的 Python 3.8 版本：

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

请注意，`python3-mod_wsgi` 和 `python38-mod_wsgi` 软件包相互冲突，并且由于 Apache HTTP 服务器的限制，系统上只能安装一个 `mod_wsgi` 模块。

此更改引入了 [BZ#1829692](#) 中描述的依赖项已知问题。

(BZ#1779705)

支持 IBM Z 上的 zlib 中的硬件加速定义

在这个版本中，为 IBM Z 大型机中的 `zlib` 库增加了对硬件加速定义算法的支持。因此，提高了 IBM Z 向量机器上的压缩和解压缩性能。

(BZ#1659433)

IBM Power Systems、little endian 上的 gzip 解压缩时提高了性能

在这个版本中，32 位 Cyclic Redundancy 检查(CRC32)的优化功能增加了对 IBM Power Systems little endian 的 `zlib` 库的优化。因此，解压缩 `gzip` 文件的性能提高了。

(BZ#1666798)

新模块流：maven:3.6

RHEL 8.2 引入了一个新的模块流 `maven:3.6`。此版本的 Maven 软件项目管理和理解工具针对 RHEL 8.0 提供的 `maven:3.5` 流提供大量程序错误修复和各种增强。

要安装 **maven:3.6** 流，请使用：

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

如果要从 **maven:3.5** 流升级，[请参阅切换到更新的流](#)。

(BZ#1783926)

mod_md 现在支持 ACMEv2 协议

mod_md 模块已更新至 2.0.8 版本。在这个版本中增加了一些功能，特别是支持自动证书管理环境 (ACME) 证书和管理协议的版本 2，即互联网工程任务组(IETF)标准(RFC 8555)。原始 ACMEv1 协议仍然被支持，但已被常见服务提供商弃用。

(BZ#1747923)

PHP 7.3 的新扩展

php:7.3 模块流已更新，提供两个新的 PHP 扩展：**rrd** 和 **Xdebug**。

rrd 扩展提供到 **RRDtool** C 库的绑定。**RRDtool** 是用于时间序列数据的高性能数据日志记录和图形化系统。

包括 **Xdebug** 扩展以帮助您调试和开发。请注意，扩展仅用于开发目的，不应在生产环境中使用。

有关在 RHEL 8 中安装和使用 PHP 的详情，[请参考使用 PHP 脚本语言](#)。

(BZ#1769857, BZ#1764738)

新软件包：perl-LDAP 和 perl-Convert-ASN1

这个更新向 RHEL 8 添加了 **perl-LDAP** 和 **Perl-Convert-ASN1** 软件包。**perl-LDAP** 软件包为 Perl 语言提供 LDAP 客户端。**Perl-LDAP** 需要 **perl-Convert-ASN1** 软件包，该软件包使用基本编码规则(BER)和可辨识的编码规则(DER)来编码和解码 Abstract Syntax Notation One(ASN.1)数据结构。

(BZ#1663063, BZ#1746898)

sscg 现在支持生成受密码保护的私钥文件

sscg 实用程序现在可以生成受密码保护的私钥文件。这为私钥增加了另一级别的保护，而且某些服务（如 FreeRADIUS）需要它。

(BZ#1717880)

5.1.11. 编译器和开发工具

Grafana rebase 到版本 6.3.6

The **grafana** 软件包已升级到 6.3.6 版本，提供多个程序错误修复和增强。主要变更包括：

- Database：重写系统统计信息查询，以获得更好的性能。
- 探索：
 - 在 Safari 浏览器的分割视图中修复查询字段布局。
 - 为支持的数据源添加 Live 选项，将 **orgId** 添加到 URL 中以进行共享。

- 添加了对标签端点的新 **loki start** 和 **end** 参数的支持。
- 添加了对 Explore 中切换原始查询模式的支持，允许在指标和日志之间进行切换。
- 显示日志级别，如果由字段或标签提供，则不会解析日志级别。
- 支持新的 **LogQL** 过滤语法。
- 使用 Grafana/UI 中的新 **TimePicker**。
- 处理 **LogRow** 高亮点中的新行。
- 修复重新浏览到控制面板面板。
- 按日志图中的序列级别修复过滤器。
- 修复加载和图形/表被折叠时的问题。
- 修复日志行的选择/复制。
- 仪表盘：修复仪表盘 **init** 失败的仪表盘加载错误，包含缺少属性的面板链接，并在导出到逗号分隔的值(CSV)数据链接时修复时区仪表盘设置。
- editor：修复仅复制整行的问题。
- LDAP：集成多 **LDAP** 和 **LDAP** 身份验证组件。
- profile/UserAdmin：修复用户代理解析器在 32 位构建中崩溃 **grafana-server**。
- Prometheus:
 - 在切换到 **Prometheus** 数据源时防止面板编辑器崩溃，将大括号更改为更少烦恼。
 - 通过 **label_replace** 修复查询，并在加载查询编辑器时删除 \$1 个匹配项。
 - 一致允许编辑器中的多行查询，并将时区考虑步骤匹配。
 - 将覆盖面板范围用于 **\$__range**，而不是仪表盘范围。
 - 为序列标签查询添加时间范围过滤器，转义 | 插入的 **PromQL** 变量中的字面。
 - 在添加在 Explore 中包含冒号的指标标签时修复。
- auth：允许 API 密钥过期，返回设备、os 和浏览器，同时在 HTTP API 中列出用户身份验证令牌，支持在 UI 中列出和撤销用户身份验证令牌。
- DataLinks：正确地将有作用域的变量应用到数据链路，紧跟时区，同时在图形上下文菜单中显示数据点时间戳，在变量插入时使用数据点时间戳，修复 **\${__series_name}** 的不正确插入情况。
- 图：修复图例问题单击序列行图标，并在窗口中看到水平滚动栏时出现问题，添加新的填空选项。
- 图表：避免单值数组变量的 **Glanceob**，修复最后移动别名函数的问题，修复 **seriesByTag** 和函数与变量参数相关的问题，将 **POST** 用于 **/metrics/find** 请求。
- 时间序列：假定值都是数字。
- 计量型/BarGauge：修复阈值丢失的问题，以及 **avg stat** 加载 Gauge 的问题。

- PanelLinks：使用面板链接修复 Gauge 和 Bar Gauge 面板中的崩溃问题（浏览链接），修复显示问题，而未显示面板描述。
- OAuth：修复了因 SameSite cookie 策略而丢失的保存状态 OAuth 登录失败，修复了 DS 代理中对 **OAuth** 刷新的错误令牌更新。
- auth Proxy：包含其他标头作为缓存密钥的一部分。
- **CLI**：修复在 dev 模式中识别时的 sql 错误，修复 **encrypt-datasource-passwords** 失败的问题。
- 权限：在导航中为非管理员用户显示插件，但隐藏插件配置。
- TimePicker：提高自定义范围弹出的最大快速范围下拉菜单高度和修复风格问题。
- Loki：按照 Explore 中的正确顺序显示实时跟踪的日志。
- Timerange：修复自定义时间范围不遵循通用时间协调(UTC)的错误。
- **remote_cache**：修复 **redis connstr** 解析。
- 警报：添加标签到警报规则，尝试发送电子邮件通知到所有给定的电子邮件地址，改进警报规则测试，支持为 **Discord** 警报通知器配置内容字段。
- Alertmanager：在标签名称中将非法字符替换为下划线。
- AzureMonitor：为 Azure Logs 更改内置 Grafana 变量或宏名称。
- cloudwatch：对 Amazon Web Services(AWS)Cloudwatch Expressions 可见的区域，添加 **AWS DocDB** 指标。
- GraphPanel：当图例列表和排序列不可见时，请勿排序系列。
- InfluxDB：支持可视化内存中的日志。
- MySQL/Postgres/MSSQL：在宏中添加每天、周和年间隔的解析，增加了对定期重新加载客户端证书的支持。
- plugins：使用 **plugin.json** 文件中的 **skipDataQuery** 标志替换 **dataFormats** 列表。
- 刷新提取器：处理空间隔。
- singlestat：将 **y min/max** 配置添加到 singlestat sparklines。
- 模板：在页面重新加载后，在多值变量中显示 **__text**，支持选择多值变量的所有筛选值。
- frontend：修复 Json 树组件无法正常工作的问题。
- InfluxDB：修复标签值过滤器中未转义的单引号的问题。
- config：修复 **defaults.ini** 文件中的 **remote_cache** 的连接字符串选项。
- Elasticsearch：修复空查询（通过模板变量）应该以通配符形式发送，修复默认的最大并发分片请求，支持在 Explore 中可视化日志。
- TablePanel：修复注释显示。

- Grafana-CLI : 通过命令行修复接收标记, **RPM/DEB** 软件包中的 **grafana-cli** 的 wrapper 和 **config/homepath** 现在是全局标记。
- HTTPServer : 修复 **X-XSS-Protection** 标头格式, 用于返回新标头的选项 **X-Content-Type-Options**、**X-XSS-Protection** 和 **Strict-Transport-Security**, 修复 **Strict-Transport-Security** 标头, 使用自定义 URL 路径前缀提供 Grafana。

([BZ#1725278](#))

pcp rebase 到版本 5.0.2

pcp 软件包已升级到 5.0.2 版本, 提供多个程序错误修复和增强。主要变更包括 :

- **pcp-webapp-*** 软件包现在由 **grafana-pcp** 软件包和 **pmproxy** 替代。
- **pcp-collectl** 工具现在被 **pmrep** 配置替代。
- 新的且改进的性能指标域代理(PMDA) :
 - **pmdamssql** : 适用于 Microsoft SQL Server 实施的新 PMDA。
 - **pmdanetcheck** : 用于执行网络检查的新 PMDA。
 - **pmdaopenmetrics** : 将 **prometheus** 代理重新命名为 **openmetrics**。
 - **pmdanfsclient** : 添加 **per-op** 和 **每个mount rpc** 错误指标。
 - **pmdalmsensors** : 改进名称解析和错误处理。
 - **pmdaperfevent** : 支持多节点系统上的 **hv_24x7** 嵌套事件。
 - **pmdalinux**:
 - 正确处理稀疏或停止的数值节点。
 - 使用 **cpu instname**, 而不是 **每个cpu numa stats** 的 **instid**。
 - 为 **slab info v2** 解析添加一个活跃和总计 **slab s**
 - 修复了几个 **unix socket**, **icmp6** 指标数据、巨页指标值. 计算、带有大量 CPU 数的中断代码的 **segfault**
 - 在 **--container** 命名空间中获取更多网络指标。
 - **pmdabcc** : 修复了 **bcc** 0.10.0 及更高版本的 **tracepoints** 模块
 - **pmdabpfttrace** : **b pfttrace** 脚本中指标的新 PMDA
 - **pmdaproc**:
 - 修复 **pidlist** 刷新中的内存泄漏。
 - 避免 **cgroups_scan** 中过多的 **stat** 调用。
 - 保留 **cgroup** 路径, 并且仅保留 **un-escape** 实例名称。
 - **pmdaroot** : 改进了缓存或不活跃的的 **cgroup** 行为的处理, 并刷新了对 **cgroup fs** 更改的容器 **indom**。

- 收集器（服务器）工具修复：
 - **pmproxy**：通过 **/metrics** 端点提供 **Openmetrics** 支持，整合 **pmseries/grafana** REST API，并添加新的 **async PMWEBAPI(3)** REST API 实施。
 - **SELinux**：大量 **pcp** 策略更新。
 - Python **pmdas**：启用身份验证支持，使用新的 **set_comm_flags** 方法来设置通信标记。
 - **Python api**：导出 **pmdaGetContext ()** 并添加调试打包程序。
 - **Perl api**：确保为 **PMDA** 存储设置的上下文与 **python wrapper** 一样。
 - **systemd**：在所有服务中添加 **120s** 超时，并修复启动 **pmlogger** 服务失败。
- 分析（客户端）工具修复：
 - **pmchart**：修复获取错误条件下的 **Chart** 自动扩展。
 - **pmrep**：修复了 **collectl-dm-sD** 和 **collectl-sD** 的 **wait.formula**。
 - **pmseries**：提供对 **delta** 关键字和更好的时间戳的支持。
 - **pcp-atop**：修复写入模式(**-w**)以处理 **proc** vs **hotproc** 指标。
 - **pcp-atopsar**：修复几个命令行参数的错误处理。
 - **pcp-dstat**：在 **CSV** 输出中修复不匹配的标头，并处理 **--bits** 命令行选项。
 - **libpcp**：使用本地上下文修复 **cockpit-pcp segv**，针对损坏的归档修复多存档错误处理。

([BZ#1723598](#))

Grafana-pcp 现在包括在 RHEL 8.2 中

The **grafana-pcp** 软件包提供新的 **grafana** 数据源和应用程序插件，将 **PCP** 与 **grafana** 连接。使用 **grafana-pcp** 软件包，您可以分别使用 **pmseries** 查询语言和 **pmwebapi** 实时服务分析历史 **PCP** 指标和实时 **PCP** 指标。如需更多信息，请参阅 [Performance Co-Pilot Grafana 插件](#)。

([BZ#1685315](#))

更新了 GCC 工具集 9

GCC 工具集 9 是提供开发工具最新版本的编译器工具集。它以 **AppStream** 存储库中的 **Software Collection** 的形式作为 **Application Stream** 提供。

RHEL 8.2 的主要变化包括：

- **GCC 编译器**已更新至版本 **9.2.1**，它在上游 **GCC** 中提供了很多程序错误修复和增强。
- **GCC Toolset 9** 组件现在在两个容器镜像中可用：
 - **rhel8/gcc-toolset-9-toolchain**，其中包括 **GCC 编译器**、**GDB 调试器**和 **make 自动化工具**。
 - **rhel8/gcc-toolset-9-perftools**，其中包括性能监控工具，如 **SystemTap** 和 **Valgrind**。要拉取容器镜像，以 **root** 身份运行以下命令：

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

GCC 工具集 9 提供以下工具和版本：

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

要安装 GCC Toolset 9，以 root 用户身份运行以下命令：

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

要从 GCC Toolset 9 运行工具：

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

要运行 shell 会话，GCC Toolset 9 的工具版本优先于这些工具的系统版本：

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

如需更多信息，[请参阅使用 GCC Toolset。](#)

([BZ#1789401](#))

GCC Toolset 9 现在支持 NVIDIA PTX 目标卸载

GCC Toolset 9 中的 GCC 编译器现在支持 NVIDIA PTX 的 OpenMP 目标卸载。

(BZ#1698607)

更新的 GCC 编译器现在可用于 RHEL 8.2

系统 GCC 编译器 8.3.1 版已更新，包括了上游 GCC 中提供的众多程序错误修复和增强。

GNU 编译器集合(GCC)提供了用于使用 C、C++ 和 Fortran 编程语言开发应用程序的工具。

有关使用信息，请参阅在 [RHEL 8 中开发 C 和 C++ 应用程序](#)。

(BZ#1747157)

用于更改 glibc 中最大 fastbin 大小的新可调项

malloc 函数使用一系列将可重复使用的内存区块保留为特定大小的 fastbin。32 位系统中的默认块大小为 80 字节，64 位系统中的默认块大小为 160 字节。这个功能在 **glibc** 中引入了一个新的 **glibc.malloc.mxfast** 可调项，可让您更改最大 fastbin 大小。

(BZ#1764218)

现在在 GCC Toolset 9 中为 GNU Fortran 启用向量化数学库

在这个版本中，GCC 工具集的 GNU Fortran 可以使用向量化数学库 **libmvec** 中的例程。在以前的版本中，GCC Toolset 中的 Fortran 编译器需要一个 Fortran 标头文件，然后才能使用 GNU C 库 **glibc** 提供的 **libmvec** 提供的例程。

(BZ#1764238)

glibc.malloc.tcache 可调项已被改进

glibc.malloc.tcache_count 可调项可用于设置可存储在每个线程缓存(tcache)的每个大小的最大内存区块数。在这个版本中，**glibc.malloc.tcache_count** 可调项的上限从 127 增加到 65535。

(BZ#1746933)

glibc 动态加载程序被改进，以提供非继承库预加载机制

在这个版本中，可以调用加载程序来加载使用 **--preload** 选项的用户程序，后跟用于预加载的以冒号分隔的库列表。此功能允许用户通过加载器直接调用其程序，并包含非继承库预加载列表。

在以前的版本中，用户必须使用 **LD_PRELOAD** 环境变量，该变量由所有子进程通过自己的环境继承。

(BZ#1747453)

GDB 现在支持 IBM Z 架构中的 ARCH(13)扩展

在这个版本中，GNU Debugger(GDB)支持 IBM Z 架构中的 ARCH(13)扩展实施的新指令。

(BZ#1768593)

elfutils rebase 到版本 0.178

The **elfutils** 软件包已升级到 0.178 版本，它提供多个程序错误修复和增强。主要变更包括：

- **elfclassify**：分析 ELF 对象的新工具。
- **debuginfod**：通过 HTTP 从文件和 RPM 存档中自动获取 ELF、DWARF 和源的新服务器、客户端工具和库。
- **libeb1** 现在直接编译到 **libdw.so** 中。

- **eu-readelf** 具有多个注释、章节编号和符号表的新标志。
- **libdw** 改进了多线程支持。
- **libdw** 支持额外的 GNU DWARF 扩展。

(BZ#1744992)

SystemTap rebase 到版本 4.2

SystemTap 工具已更新至版本 4.2。主要改进包括：

- 回溯追踪现在可以包含源文件名和行号。
- 现在提供了许多 Berkeley Packet Filter(BPF)后端扩展，例如用于循环、计时和其他进程。
- 提供了用于管理 SystemTap 脚本的新服务。此服务将指标发送到与 Prometheus 兼容的监控系统。
- SystemTap 继承了新的 HTTP 文件服务器 for **elfutils**（名为 **debuginfod**）的功能。此服务器会自动向 SystemTap 发送调试资源。

(BZ#1744989)

IBM Z 系列性能计数器的改进

IBM Z 系列类型 0x8561、0x8562 和 0x3907(z14 ZR1)机器现在可以被 **libpfm** 识别。现在提供了在 IBM Z 系列上监控 elliptic-curve 加密(ECC)操作的性能事件。这允许监控 IBM Z 系列机器上的其他子系统。

(BZ#1731019)

Rust Toolset rebase 到版本 1.41

Rust Toolset 已更新至 1.41 版本。主要变更包括：

- 现在更容易实施新的特征，因为孤立规则不太严格。
- 现在，您可以将 **#[non_exustive]** 属性附加到一个结构、**enum** 或 **enum** 变体。
- 在规范 **功能接口(FFI)**中使用 **Box<T>**；现在可以提供更多保障。**box<T>** 拥有与 FFI 中的 **T*** 指针相同的 Application Binary Interface(ABI)。
- Trust 应该在编译时检测到内存安全错误，但之前的租借检查器存在限制，并允许未定义的行为和内存不安全。新的非编配生命周期(NLL)借机检查器可将内存不安全问题报告为困难错误。它现在适用于 Rust 2015 和 Rust 2018 版本。在以前的版本中，在 Rust 2015 中，NLL 租借检查器只引发有关此类问题的警告。

要安装 **rust-toolset** 模块，以 root 用户身份运行以下命令：

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

有关使用信息，[请参阅使用 Rust Toolset](#)。

(BZ#1776847)

LLVM Toolset rebase 到版本 9.0.1

LLVM Toolset 已升级至版本 9.0.1。在这个版本中，支持 **asm goto** 语句。此更改允许在 AMD64 和 Intel 64 构架中编译 Linux 内核。

要安装 **llvm-toolset** 模块，以 root 用户身份运行以下命令：

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

如需更多信息，请参阅[使用 LLVM Toolset](#)。

(BZ#1747139)

Go Toolset rebase 到版本 1.13

Go Toolset 已升级到 1.13 版本。主要改进包括：

- 现在，当 RHEL 系统以 FIPS 模式引导时，Git 可使用 FIPS 认证的加密模块。用户可以使用 **GOLANG_FIPS=1** 环境变量手动启用此模式。
- Delve debugger 版本 1.3.2 现在可用于 Go。它是 Go(**golang**)编程语言的源级调试器。

要安装 **go-toolset** 模块，以 root 用户身份运行以下命令：

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

要安装 Delve 调试程序，以 root 用户身份运行以下命令：

```
# yum install delve
```

要使用 Delve 调试 **helloworld.go** 程序，请运行以下命令：

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

有关 Go Toolset 的详情，请参考[使用 Go Toolset](#)。

有关 Delve 的更多信息，请参见上游 [Delve 文档](#)。

(BZ#1747150)

OpenJDK 现在也支持 secp256k1

在以前的版本中，OpenJDK (OpenJDK) 只能使用 NSS 库中的 curves。因此，对于椭圆曲线加密 (ECC)，OpenJDK 只提供了 secp256r1、secp384r1 和 secp521r1 curves。在这个版本中，OpenJDK 使用内部 ECC 实现，并支持 secp256k1 curve。

(BZ#1746875,BZ#1746879)

5.1.12. Identity Management

IdM 现在支持新的 Ansible 管理模块

此更新引入了几个 **ansible-freeipa** 模块，用于使用 Ansible playbook 自动完成通用身份管理(IdM)任务：

- **ipauser** 模块可自动添加和删除用户。
- **ipagroup** 模块自动向用户组和用户组添加和删除用户组。

- **ipahost** 模块自动添加和删除主机。
- **ipahostgroup** 模块自动向主机组和主机组添加和删除主机和主机组。
- **ipasudorule** 模块自动管理 **sudo** 命令和 **sudo** 规则。
- **ipapwpolicy** 模块会在 IdM 中自动配置密码策略。
- **ipahbacrule** 模块在 IdM 中自动管理基于主机的访问控制。

请注意，您可以将两个或多个 **ipauser** 调用组合一个与 **users** 变量，或者使用包含用户的 JSON 文件。类似地，您可以将两个或多个 **ipahost** 调用组合一个与 **hosts** 变量，或者使用包含主机的 JSON 文件。**ipahost** 模块还可确保主机存在或不存在多个 IPv4 和 IPv6 地址。

(JIRA:RHELPLAN-37713)

IdM Healthcheck 现在支持强制 DNS 记录

在这个版本中引进了对 Identity Management(IdM)服务器上的 DNS 记录的独立手动测试。

该测试使用 **Healthcheck** 工具，并使用 **etc/resolv.conf** 文件中的本地解析器执行 DNS 查询。测试可确保自动发现所需的预期 DNS 记录可以解析。

(JIRA:RHELPLAN-37777)

使用 SSSD 将 RHEL 直接集成到 AD 中现在支持 FIPS

在这个版本中，系统服务安全守护进程(SSSD)与 Active Directory(AD)部署集成，它们的身份验证机制使用由联邦信息处理标准(FIPS)批准的加密类型。这个增强可让您在必须满足 FIPS 条件的环境中直接将 RHEL 系统集成到 AD 中。

(BZ#1841170)

默认情况下，Samba 服务器和客户端实用程序中已禁用 SMB1 协议

在 Samba 4.11 中，服务器 **min** 协议和客户端 **min** 协议 参数的默认值已从 **NT1** 改为 **SMB2_02**，因为服务器消息块版本 1(SMB1)协议已弃用。如果您还没有在 **/etc/samba/smb.conf** 文件中设置这些参数：

- 仅支持 SMB1 的客户端不再能够连接到 Samba 服务器。
- Samba 客户端实用程序（如 **smbclient** 和 **libsmbclient** 库）无法连接到仅支持 SMB1 的服务器。

红帽建议不要使用 SMB1 协议。但是，如果您的环境需要 SMB1，您可以手动重新启用该协议。

要在 Samba 服务器中重新启用 SMB1：

- 在 **/etc/samba/smb.conf** 文件中添加以下设置：

```
server min protocol = NT1
```

- 重启 **smb** 服务：

```
# systemctl restart smb
```

要为 Samba 客户端实用程序和 **libsmbclient** 库重新启用 SMB1：

- 在 **/etc/samba/smb.conf** 文件中添加以下设置：

```
client min protocol = NT1
```

- 重启 **smb** 服务：

```
# systemctl restart smb
```

请注意，SMB1 协议将在以后的 Samba 发行版中删除。

([BZ#1785248](#))

Samba rebase 到版本 4.11.2

samba 软件包已升级到上游版本 4.11.2，它提供很多程序错误修复和增强。主要变更包括：

- 默认情况下，Samba 服务器、客户端实用程序和 **libsmbclient** 库中现在禁用服务器消息块版本 1(SMB1)协议。但是，您仍然可以手动将 **服务器 min 协议** 和 **客户端 min 协议** 参数设置为 **NT1** 以重新启用 SMB1。红帽不建议重新启用 SMB1 协议。
- The **lanman auth** 和 **encrypt password** 参数已弃用。这些参数启用不安全的身份验证，且仅在已弃用的 SMB1 协议中使用。
- **o 参数** 已从 **onode** 集群的简单数据库(CTDB)实用程序中删除。
- Samba 现在使用 GnuTLS 库进行加密。因此，如果启用了 RHEL 中的 FIPS 模式，Samba 符合 FIPS 标准。
- 现在，当使用超过 90% 的 CPU 线程时，**c tdbd** 服务会进行日志。
- 弃用的 Python 2 支持已被删除。

当 **smbd**、**nmbd** 或者 **winbind** 服务启动时，Samba 会自动更新它的 **tdb** 数据库文件。在启动 Samba 前备份数据库文件。请注意，红帽不支持降级 **tdb** 数据库文件。

有关显著变化的更多信息，请在更新前阅读上游发行注记：

<https://www.samba.org/samba/history/samba-4.11.0.html>

([BZ#1754409](#))

目录服务器 rebase 到版本 1.4.2.4

389-ds-base 软件包已升级到上游版本 1.4.2.4，与之前的版本相比，它提供了一些程序错误修复和增强。如需显著变化的完整列表，请在更新前阅读归档的上游发行注记：

[389 目录服务器发行注记存档](#)包括以下版本的发行注记：[* 389 Directory Server 1.4.2.4](#) * [389 Directory Server 1.4.2.3](#) * [389 Directory Server 1.4.2.2](#) * [389 Directory Server 1.4.2.1](#)

([BZ#1748994](#))

目录服务器中替换了一些旧脚本

这个增强为不支持的 **dbverify**、**validate-syntax.pl**、**cl-dump.pl**、**fixup-memberuid.pl** 和 **repl-monitor.pl** 传统脚本提供了目录服务器中的替换。以下命令替换了这些脚本：

- **dbverify:dsctl instance_name dbverify**
- **validate-syntax.pl:dsconf schema validate-syntax**
- **CL-dump.pl:dsconf 复制 dump-changelog**

- **fixup-memberuid.pl: dsconf plugin posix-winsync fixup**
- **rEPL-monitor.pl : dsconf 复制监控器**

有关所有旧脚本及其替换列表，请参阅 [Red Hat Directory Server 11 中的命令行实用程序](#)。

(BZ#1739718)

现在完全支持将 IdM 设置为隐藏的副本

RHEL 8.2 中的身份管理(IdM)完全支持将 IdM 服务器设置为隐藏的副本。隐藏的副本是一个 IdM 服务器，它具有所有运行的服务并可用。但是，它不会公告给其他客户端或主控机，因为 DNS 中不存在服务的 **SRV** 记录，并且未启用 LDAP 服务器角色。因此，客户端无法使用服务发现来检测隐藏的副本。

隐藏副本主要针对可能会破坏客户端的专用服务设计。例如，IdM 的完整备份需要关闭 master 或副本中的所有 IdM 服务。因为没有客户端使用隐藏的副本，管理员可以在不影响任何客户端的情况下暂时关闭这个主机上的服务。其他用例包括 IdM API 或 LDAP 服务器上的高负载操作，如大量导入或广泛查询。

若要安装新的隐藏副本，请使用 **ipa-replica-install --hidden-replica** 命令。要更改现有副本的状态，请使用 **ipa server-state** 命令。

如需了解更多详细信息，请参阅[安装 IdM 隐藏副本](#)。

(BZ#1719767)

Kerberos 票据策略现在支持验证指示符

身份验证指示符会根据用来获取票据的预身份验证机制附加到 Kerberos 票据中：

- **OTP 用于 双因素身份验证（密码 + OTP）**
- **用于 RADIUS 身份验证的 radius**
- **用于 PKINIT、智能卡或证书身份验证的 PKINIT**
- **强化强化的密码（SPAKE 或 FAST）**

Kerberos 分发中心(KDC)可以对基于身份验证指示器的服务票据请求实施策略，如服务访问控制、最长票据生命周期和最长可续订期限。

在这个版本中，管理员可以要求来自用户票据的特定身份验证指示器更精细地控制服务票据。

(BZ#1777564)

krb5 软件包现在与 FIPS 兼容

在这个版本中，禁止使用不合规的加密。因此，管理员可以在 FIPS 管理的环境中使用 Kerberos。

(BZ#1754690)

目录服务器根据系统范围的加密策略设置 **sslVersionMin** 参数

默认情况下，Directory 服务器现在根据系统范围的加密策略设置 **sslVersionMin** 参数的值。如果您在 **/etc/crypto-policies/config** 文件中将 crypto 策略配置集设置为：

- **DEFAULT、FUTURE 或 FIPS** 目录服务器将 **sslVersionMin** 设置为 **TLS1.2**
- **criACY**, Directory Server 将 **sslVersionMin** 设置为 **TLS1.0**

或者，您可以将 **sslVersionMin** 手动设置为比 `crypto` 策略中定义的值更高：

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

(BZ#1828727)

SSSD 现在默认强制使用 AD GPOs

SSSD 选项 **ad_gpo_access_control** 的默认设置现在是 **enforcing**。在 RHEL 8 中，SSSD 默认根据 Active Directory Group 策略对象(GPO)强制实施访问控制规则。

红帽建议确保在从 RHEL 7 升级到 RHEL 8 前，在 Active Directory 中正确配置 GPO。如果您不想强制执行 GPOs，请将 `/etc/sss/sss.conf` 文件中的 **ad_gpo_access_control** 选项的值改为 **permissive**。

(JIRA:RHELPLAN-51289)

5.1.13. Desktop

Wayland 现在在双GPU系统上启用

在以前的版本中，GNOME 环境默认为笔记本电脑和其他具有两个图形处理单元(GPU)的 X11 会话。在这个版本中，GNOME 默认为双GPU 系统上的 **Wayland** 会话，这与单GPU 系统上的行为相同。

(BZ#1749960)

5.1.14. 图形基础结构

支持新图形卡

现在支持以下图形卡：

- Intel HD Graphics 610、620 和 630，可在 Intel Comet Lake H 和 U 处理器中找到
- Intel Ice Lake UHD Graphics 910 和 Iris Plus Graphics 930、940 和 950。
您不再需要设置 **alpha_support** 内核选项来启用对 Intel Ice Lake 图形的支持。
- AMD Navi 10 系列，其包括以下型号：
 - Radeon RX 5600
 - Radeon RX 5600 XT
 - Radeon RX 5700
 - Radeon RX 5700 XT
 - Radeon Pro W5700
- Nvidia Turing TU116 系列，包括以下模型：
请注意，**nouveau** 图形驱动程序尚不支持使用 Nvidia Turing TU116 系列的 3D 加速。
 - GeForce GTX 1650 Super
 - GeForce GTX 1660
 - GeForce GTX 1660 Super

- GeForce GTX 1660 Ti
- GeForce GTX 1660 Ti Max-Q

另外，以下图形驱动程序已被更新：

- Matrox **mgag2000** 驱动程序
- A fast **ast** 驱动程序
- Intel **i915** 驱动程序

(JIRA:RHELPLAN-41384)

5.1.15. Web 控制台

管理员现在可以使用客户端证书在 RHEL 8 web 控制台中进行身份验证

有了这个 web 控制台增强，系统管理员可以使用客户端证书本地访问 RHEL 8 系统，或使用带有内置证书身份验证的浏览器远程访问 RHEL 8 系统。不需要额外的客户端软件。这些证书通常由智能卡或 Yubikey 提供，也可以导入到浏览器中。

使用证书登录时，用户目前无法在 Web 控制台中执行管理操作。但是，用户可以在使用密码进行身份验证后通过 **sudo 命令** 在 Terminal 页面上执行它们。

(JIRA:RHELPLAN-2507)

使用 TLS 客户端证书登录到 web 控制台的选项

在这个版本中，可以配置 Web 控制台，以使用浏览器或智能卡或 YubiKey 等设备提供的 TLS 客户端证书登录。

(BZ#1678465)

更改 Web 控制台登录

RHEL web 控制台已更新了以下更改：

- 在 15 分钟不活动后，Web 控制台将自动从当前会话注销。您可以在 `/etc/cockpit/cockpit.conf` 文件中几分钟内配置超时。
- 与 SSH 类似，Web 控制台现在可以选择性地显示登录屏幕上横幅文件的内容。用户需要在 `/etc/cockpit/cockpit.conf` 文件中配置功能。

如需更多信息，请参阅 **cockpit.conf(5)** 手册页。

(BZ#1754163)

RHEL web 控制台已被重新设计为使用 PatternFly 4 用户界面设计系统

新的设计提供了更好的可访问性，与 OpenShift 4 的设计相匹配。更新包括：

- Overview 页面已被完全重新设计。例如，信息被分组为更容易理解的面板，健康信息更突出，资源图已移至自己的页面，现在硬件信息页面更容易找到。
- 用户可以使用导航菜单中的新搜索字段轻松查找基于关键字的特定页面。

有关 PatternFly 的更多信息，请参阅 [PatternFly 项目页面](#)。

([BZ#1784455](#))

Virtual Machines 页面更新

Web 控制台的 **Virtual Machines** 页面有几个存储改进：

- 存储卷创建现在可以适用于所有 libvirt 支持的类型。
- 可以在 LVM 或 iSCSI 上创建存储池。

另外，**Virtual Machines** 页面现在支持创建和删除虚拟网络接口。

([BZ#1676506](#),[BZ#1672753](#))

Web 控制台 存储 页面更新

可用性测试表明 RHEL Web 控制台 **存储** 页面 中的默认挂载点概念难以获取，并导致很多混淆。在这个版本中，在挂载文件系统时，Web 控制台不再提供默认选择。现在，创建新文件系统总是需要指定挂载点。

此外，Web 控制台现在隐藏了配置(/etc/fstab)和运行时状态(/proc/mounts)之间的区别。Web 控制台中所做的更改始终同时应用到配置和运行时状态。当配置和运行时状态相互不同时，Web 控制台会显示警告信息，并允许用户轻松将其重新同步。

([BZ#1784456](#))

5.1.16. 虚拟化

现在，尝试从安装树创建 RHEL 虚拟机会返回更有用的错误消息。

使用 **virt-install** 实用程序和 **--location** 选项创建的 RHEL 7 和 RHEL 8 虚拟机在某些情况下无法引导。这个更新添加了 **virt-install** 错误消息，它提供了有关如何临时解决这个问题的说明。

([BZ#1677019](#))

KVM 客户机上支持 Intel Xeon Platinum 9200 系列处理器

现在，在 KVM 管理程序和内核代码以及 libvirt API 中增加了对 Intel Xeon Platinum 9200 系列处理器（以前称为 **Cascade Lake**）的支持。这使得 KVM 虚拟机能够使用 Intel Xeon Platinum 9200 系列处理器。

([JIRA:RHELPLAN-13995](#))

EDK2 被 rebase 到版本 stable201908

EDK2 软件包已升级至 **stable201908** 版本，提供多个改进。值得注意的是：

- EDK2 现在包括对 OpenSSL-1.1.1 的支持。
- 为了满足上游项目的许可要求，EDK 2 软件包许可证已从 **BSD 和 OpenSSL 和 MIT** 改为 **BSD - 2-Clause-Patent 和 OpenSSL 和 MIT**。

([BZ#1748180](#))

创建嵌套虚拟机

在这个版本中，在带有 RHEL 8 的 Intel 64 主机上运行的 KVM 虚拟机(VM)完全支持嵌套虚拟化。使用此功能，在物理 RHEL 8 主机上运行的 RHEL 7 或 RHEL 8 虚拟机可作为虚拟机监控程序，并托管自己的虚拟机。

请注意，在 AMD64 系统中，嵌套的 KVM 虚拟化仍然是技术预览。

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

5.1.17. 容器

/etc/containers/registries.conf 中的默认 registry 搜索列表已被更新

/etc/containers/registries.conf 中的默认 **registries.search** 列表已更新，仅包含受信任的 registry，这些 registry 提供由红帽及其合作伙伴设置的容器镜像。

红帽建议您始终使用完全限定的镜像名称，包括：

- 注册表服务器（完整 DNS 名称）
- 命名空间
- 镜像名称
- tag（如 **registry.redhat.io/ubi8/ubi:latest**）

在使用短名称时，例如欺骗风险始终存在，用户希望从注册表中提取名为 **foobar** 的镜像，并且希望它来自 **myregistry.com**。如果 **myregistry.com** 先不在搜索列表中，攻击者可以在搜索列表中的 registry 放置不同的 **foobar** 镜像。用户会意外拉取并运行攻击者镜像和代码，而不是预期的内容。红帽建议您只添加受信任的 registry，即不允许未知或匿名用户创建带有任意名称的帐户的 registry。这可以防止镜像被欺骗、排序或者变得不安全。

([BZ#1810053](#))

Podman 不再依赖于 oci-systemd-hook

Podman 不需要或依赖 **oci-systemd-hook** 软件包，该软件包已从 **container-tools:rhel8** 和 **container-tools:2.0** 模块流中删除了。

([BZ#1645280](#))

5.2. 对外部内核参数的重要更改

本章为系统管理员提供了随 Red Hat Enterprise Linux 8.2 发布的内核的显著变化摘要。这些更改包括添加或更新的 **proc** 条目、**sysctl** 和 **sysfs** 默认值、引导参数、内核配置选项或任何可见的行为更改。

5.2.1. 新内核参数

cpuidle.governor = [CPU_IDLE]

要使用的 **cpuidle** 调控器的名称。

deferred_probe_timeout = [KNL]

这是用于设置超时（以秒为单位）的调试参数，以便延迟探测放弃等待依赖关系到探测。

只有已选择的特定依赖程序（子系统或驱动程序）才会被忽略。超时为 0，将在 **initcalls** 结束时超时。此参数还会在重试后在延迟探测列表中转储仍然在延迟探测列表中的设备。

kvm.nx_huge_pages = [KVM]

这个参数控制 **X86_BUG_ITLB_MULTIHIT** bug 的软件临时解决方案。

这些选项是：

- **force** - 始终部署临时解决方案。
- **off** - 无法部署临时解决方案。
- **auto** (默认) - 根据 **X86_BUG_ITLB_MULTIHIT** 的存在部署临时解决方案。

如果为主机启用了软件临时解决方案，则客户机不需要为嵌套的客户机启用它。

kvm.nx_huge_pages_recovery_ratio = [KVM]

此参数控制定期将多少 4KiB 页面配置回大页面。0 可禁用恢复；否则，如果值为 N，基于内核的虚拟机(KVM)将每分钟中断 4KiB 页面的 1/N。默认值为 60。

page_alloc.shuffle = [KNL]

布尔值标志，以控制页面分配器是否应该随机化其空闲列表。

如果内核检测到它正在具有直接映射内存端缓存的平台上运行，可自动启用随机化。此参数可用于覆盖/禁用该行为。

标志的状态可以从 **/sys/module/page_alloc/parameters/shuffle** 文件中的 **sysfs** 伪文件系统读取。

panic_print =

发生异常时，位掩码用于打印系统信息。

用户可以选择以下位的组合：

- 位 0：打印所有任务信息
- 位 1：打印系统内存信息
- 第 2 位：打印计时器信息
- 位 3：如果 **CONFIG_LOCKDEP** 内核配置启用，打印锁定信息
- 位 4：打印 **ftrace** 缓冲区
- 位 5：打印缓冲区中的所有打印信息

rcutree.sysrq_rcu = [KNL]

命令提供一个 **sysrq** 键，以转储 Tree RCU 的 **rcu_node** 树，并了解为什么新的宽限期尚未启动。

rcutorture.fwd_progress = [KNL]

为支持此概念的 RCU 类型启用 Read-copy update(RCU)grace-period forward-progress 测试。

rcutorture.fwd_progress_div = [KNL]

指定 CPU-stall-warning 期间的比例来进行紧凑的正向测试。

rcutorture.fwd_progress_holdoff = [KNL]

连续的 forward-progress 测试之间等待的秒数。

rcutorture.fwd_progress_need_resched = [KNL]

在紧循环前向进度测试过程中，将 **cond_resched()** 调用括在 **need_resched()** 的检查中。

tsx = [X86]

此参数控制支持 TSX 控制的 Intel 处理器中的事务同步扩展(TSX)功能。

这些选项是：

- **on** - 在系统上启用 TSX。虽然所有已知的安全漏洞都有缓解措施，但 TSX 加速了几个与假设相关的 CVE。因此，在启用它时可能会存在未知的安全风险。

- **off** - 在系统上禁用 TSX。此选项仅对不易受 Microarchitectural Data Sampling(MDS)的较新的 CPU 起作用。换句话说，他们有 **MSR_IA32_ARCH_CAPABILITIES.MDS_NO=1**，并通过 microcode 更新获取新的 **IA32_TSX_CTRL** 模型特定寄存器(MSR)。这个新的 MSR 可以可靠地取消激活 TSX 地功能。
- **auto** - 如果存在 **X86_BUG_TAA**，则禁用 TSX，否则在系统上启用 TSX。

不指定这个参数等同于 **tsx=off**。

[详情请查看上游内核文档。](#)

tsx_async_abort = [X86,INTEL]

此参数用来控制对 TSX Async Abort (TAA) 漏洞的缓解方案。

与 MDS (Micro-architectural Data Sampling) 类似，某些支持 TSX (Transactional Synchronization Extensions) 的 CPU 可能会受到 CPU 内部缓冲区的攻击。该漏洞在某些情况下可将信息转发给泄漏小工具。

在存在安全漏洞的处理器中，缓存侧频道攻击可以利用预测的数据转发，访问到应该无法直接访问到的数据。

这些选项是：

- **Complete** - 如果启用了 TSX，则对存在安全漏洞的 CPU 启用 TAA 缓解方案。
- **完全, nosmt** - 在存在安全漏洞的 CPU 上启用 TAA 缓解并禁用同步多线程(SMT)如果已禁用了 TSX，则不会禁用 SMT，因为 CPU 不会受到跨线程 TAA 攻击。
- **off - 不** 条件禁用 TAA 缓解方案。
在 MDS 受影响的机器上，可以通过活跃的 MDS 缓解来防止 **tsx_async_abort=off** 参数，因为这两个漏洞都使用相同的机制缓解。因此，要禁用这个缓解方案，还需要指定 **sds=off** 参数。

不指定这个选项等同于 **tsx_async_abort=full**。对于受 MDS 影响并实施了 MDS 缓解方案的 CPU 中，则不需要使用 TAA 缓解方案，它并不会提供任何额外的缓解。

[详情请查看上游内核文档。](#)

5.2.2. 更新的内核参数

intel_iommu = [DMAR]

Intel IOMMU 驱动程序 Direct Memory Access Remapping(DMAR)。

这些选项是：

- **sm_on** [Default Off] - 默认情况下，可扩展模式将被禁用，即使硬件公告其支持可扩展模式转换。设定此选项后，可扩展模式将用于声明要支持它的硬件。

isolcpus = [KNL,SMP,ISOL]

此参数将给定的 CPU 集与干扰隔离。

- **managed_irq** - 子参数，可防止隔离的 CPU 被托管中断作为目标，其具有包含隔离 CPU 的中断掩码。管理的中断的关联性由内核处理，不能通过 **/proc/irq/*** 接口更改。

这种隔离是尽力的，只有在设备队列自动分配的中断掩码包含隔离和内务处理 CPU 时才有效。如果内务处理 CPU 在线，则此类中断会被定向到内务处理 CPU，这样在日常维护 CPU 上提交的 I/O 就不会干扰隔离的 CPU。

如果队列的关联性掩码仅包含隔离的 CPU，则此参数不会影响中断路由决策。但是，只有在这些隔离 CPU 上运行的任务提交 I/O 时，才会交付中断。在内务处理 CPU 上提交的 I/O 对这些队列没有影响。

mds = [X86,INTEL]

对选项的更改：

- **off** - TSX Async Abort(TAA)受影响的机器, **mds=off** 可以被活跃的 TAA 缓解措施阻止，因为这两个漏洞都可以使用相同的机制缓解。因此，要禁用这个缓解方案，您需要同时指定 **tsx_async_abort=off** 内核参数。

不指定这个参数等同于 **mds=full**。

[详情请查看上游内核文档。](#)

mem_encrypt = [X86-64]

AMD 安全内存加密(SME)控制

...

[有关何时激活内存加密的详情，请查看上游内核文档。](#)

缓解方案 =

对选项的更改：

- **off** - 禁用所有可选 CPU 缓解方案。这会提高系统性能，但可能会使用户暴露一些 CPU 漏洞。
等同于：
 - **nopti [X86,PPC]**
 - **kpti=0 [ARM64]**
 - **nospectre_v1 [X86,PPC]**
 - **nobp=0 [S390]**
 - **nospectre_v2 [X86,PPC,S390,ARM64]**
 - **spectre_v2_user=off [X86]**
 - **spec_store_bypass_disable=off [X86,PPC]**
 - **ssbd=force-off [ARM64]**
 - **l1tf=off [X86]**
 - **mds=off [X86]**
 - **tsx_async_abort=off [X86]**
 - **kvm.nx_huge_pages=off [X86]**

例外：

当 `kvm.nx_huge_pages=force` 时，这不会影响 `kvm.nx_huge_pages`。

- **auto,nosmt** - 匹配所有 CPU 漏洞，根据需要禁用 Simultaneous Multi Threading(SMT)。这个选项适用于始终希望完全缓解的用户，即使它意味着丢失 SMT。
等同于：

- `l1tf=flush,nosmt [X86]`
- `mds=full,nosmt [X86]`
- `tsx_async_abort=full,nosmt [X86]`

`rcutree.jiffies_till_sched_qs = [KNL]`

此参数在读取-复制更新(RCU)开始从 `rcu_note_context_switch()` 和 `cond_resched()` 函数请求静默状态帮助之前给所需的年龄设置宽限期（以jiffies为单位）。如果没有指定，内核将根据 `rcutree.jiffies_till_first_fqs` 和 `rcutree.jiffies_next_fqs` 内核参数的最新设置计算值。此计算的值可以在 `rcutree.jiffies_to_sched_qs` 内核参数中查看。设置 `rcutree.jiffies_to_sched_qs` 的任何尝试都将被覆盖。

`tsc =`

此参数禁用时间戳计数器(TSC)的时钟源稳定性检查。

格式：<string>

这些选项是：

- **可靠 [x86]** - 将 TSC 时钟源标记为可靠。这个选项禁用运行时时钟源验证，以及在引导时执行的稳定性检查。选项还可在较旧的硬件和虚拟环境中启用高分辨率计时器模式。
- **noirqtime [x86]** - 不要使用 TSC 进行中断请求(IRQ)记帐。用于在 Read Time-Stamp counter(RDTSC)的平台上禁用 `IRQ_TIME_ACCOUNTING`，此记帐可能会增加开销。
- **不稳定 [x86]** - 将 TSC 时钟源标记为不稳定。这个选项在启动时标记 TSC 无条件不稳定，并避免在 TSC watchdog 通知后再出现问题。
- **nowatchdog [x86]** - 禁用时钟源监控。该选项用于具有严格延迟要求的情况，其中不接受来自时钟源监视器的中断。

5.2.3. 新的 /proc/sys/kernel 参数

`panic_print`

发生异常时，位掩码用于打印系统信息。

用户可以选择以下位的组合：

- 位 0：打印所有任务信息
- 位 1：打印系统内存信息
- 第 2 位：打印计时器信息
- 位 3：如果 `CONFIG_LOCKDEP` 内核配置项目正在运行，打印锁定信息
- 位 4：打印 `ftrace` 缓冲区

例如，要打印 panic 上的任务和内存信息，请执行：

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

5.2.4. 更新了 /proc/sys/kernel 参数

threads-max

这个参数控制 `fork ()` 函数可以创建的线程的最大数量。

在初始化过程中，内核会设置这个值，即使创建了最大线程数，线程结构只占据可用 RAM 页面的部分 (1/8th)。

可写入 `threads-max` 的最小值为 1。最大值由常量 `FUTEX_TID_MASK(0x3ffff)` 提供。

如果此范围以外的值写入到 `threads-max`，则会出现错误 `EINVAL`。

5.2.5. 更新了 /proc/sys/net 参数

bpf_jit_enable

此参数启用 Berkeley Packet 过滤器 Just-in-time(BPF JIT)编译器。

BPF 是一个灵活而高效的基础架构，允许在不同 hook 点执行字节码。它用于多个 Linux 内核子系统，如网络（如 XDP、tc）、追踪（如 kprobes、uprobes、追踪点）和安全性（如 seccomp）。

LLVM 具有 aBPF 后端，可将受限 C 编译到一系列 BPF 指令中。在程序通过 `bpf ()` 系统调用并在内核中传递验证器后，JIT 会将这些 BPF 程序转换为原生 CPU 指令。

JIT 有两种类型，以下 CPU 架构中目前支持较新的 eBPF JIT：

- x86_64
- arm64
- ppc64（小端和大端）
- s390x

5.3. 设备驱动程序

本章提供了在 Red Hat Enterprise Linux 8.2 中新增或已更新的所有设备驱动程序的完整列表。

5.3.1. 新驱动程序

网络驱动程序

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

图形驱动程序和各种驱动程序

- DRM VRAM memory-management helpers (drm_vram_helper.ko.xz)
- cpuidle driver for haltpoll governor (cpuidle-haltpoll.ko.xz)
- stm_ftrace driver (stm_ftrace.ko.xz)
- stm_console driver (stm_console.ko.xz)
- System Trace Module device class (stm_core.ko.xz)
- dummy_stm device (dummy_stm.ko.xz)
- stm_heartbeat driver (stm_heartbeat.ko.xz)
- Intel® Trace Hub Global Trace Hub driver (intel_th_gth.ko.xz)
- Intel® Trace Hub PTI/LPP 输出驱动程序(intel_th_pti.ko.xz)
- Intel® Trace Hub controller driver (intel_th.ko.xz)
- Intel® Trace Hub Memory Storage 单元驱动程序(intel_th_msu.ko.xz)
- Intel® Trace Hub Software Trace Hub driver (intel_th_sth.ko.xz)
- Intel® Trace Hub 内存存储单元软件 sink(intel_th_msu_sink.ko.xz)
- Intel® Trace Hub PCI controller driver (intel_th_pci.ko.xz)
- Intel® Trace Hub ACPI controller driver (intel_th_acpi.ko.xz)
- MC Driver for Intel 10nm server processors (i10nm_edac.ko.xz)
- 设备 DAX : 直接访问映射设备(dax_pmem_core.ko.xz)
- PMEM DAX : 直接访问持久内存(dax_pmem.ko.xz)
- PMEM DAX : 支持已弃用的 /sys/class/dax 接口(dax_pmem_compat.ko.xz)
- Intel PMC Core platform init (intel_pmc_core_pltdrv.ko.xz)
- Intel RAPL(Running Average Power Limit)控制通过 MSR 接口(intel_rapl_msr.ko.xz)
- Intel Runtime Average Power Limit (RAPL) common code (intel_rapl_common.ko.xz)

存储驱动程序

- 对 MD 的集群支持(md-cluster.ko.xz)

5.3.2. 更新的驱动程序

网络驱动程序更新

- VMware vmxnet3 virtual NIC driver(vmxnet3.ko.xz)已更新至 1.4.17.0-k 版本。
- Intel® 10 Gigabit Virtual Function Network Driver(ixgbevf.ko.xz)已更新至 4.1.0-k-rh8.2.0 版本。
- Intel® 10 Gigabit PCI Express Network Driver(ixgbe.ko.xz)已更新至版本 5.1.0-k-rh8.2.0。

- Intel® Ethernet Connection E800 Series Linux Driver(ice.ko.xz)更新至 0.8.1-k 版本。
- Netronome Flow Processor(NFP)driver(nfp.ko.xz)更新至版本 4.18.0-185.el8.x86_64。
- Elastic Network Adapter(ENA)(ena.ko.xz)更新至 2.1.0K 版本。

图形和各种驱动程序更新

- HPE watchdog driver(hpwdt.ko.xz)更新至 2.0.3 版本。
- Intel I/OAT DMA Linux 驱动程序(ioatdma.ko.xz)更新至版本 5.00。

存储驱动程序更新

- HPE Smart Array Controller(hpsa.ko.xz)的驱动更新至 3.4.20-170-RH4 版本。
- LSI MPT Fusion SAS 3.0 Device Driver(mpt3sas.ko.xz)更新至版本 32.100.00.00。
- QLogic FCoE Driver(bnx2fc.ko.xz)更新至 2.12.10 版本。
- Emulex LightPulse Fibre Channel SCSI 驱动程序(lpfc.ko.xz)更新至版本 0:12.6.0.2。
- QLogic FastLinQ 4xxx FCoE Module(qedf.ko.xz)更新至 8.42.3.0 版本。
- QLogic Fibre Channel HBA Driver(qla2xxx.ko.xz)更新至 10.01.00.21.08.2-k 版本。
- Microsemi Smart Family Controller 版本的驱动程序(smartpqi.ko.xz)更新至 1.2.10-025 版本。
- QLogic FastLinQ 4xxx iSCSI Module(qedi.ko.xz)更新至 8.37.0.20 版本。
- Broadcom MegaRAID SAS Driver(megaraid_sas.ko.xz)更新至版本 07.710.50.00-rc1。

5.4. 程序错误修复

这部分论述了在 Red Hat Enterprise Linux 8.2 中修复的对用户有严重影响的错误。

5.4.1. 安装程序和镜像创建

使用 `版本` 或 `inst.version` 内核引导参数不再停止安装程序

在以前的版本中，使用 `version` 或 `inst.version` 引导参数从内核命令行引导安装程序可打印该版本，如 **anaconda 30.25.6**，并停止安装程序。

在这个版本中，当安装程序从内核命令行引导时，`版本` 和 `inst.version` 参数会被忽略，因此安装程序不会被停止。

(BZ#1637472)

支持安装程序中的 s390x 安全引导

在以前的版本中，RHEL 8.1 支持准备引导磁盘，以便在 IBM Z 环境中使用，这些磁盘强制使用安全引导。安装期间使用的服务器和虚拟机监控程序的功能决定了生成的磁盘上格式是否包含安全引导支持。安装期间无法影响磁盘格式。因此，如果您在一个支持安全引导的环境中安装了 RHEL 8.1，则该系统无法在移到一个缺乏安全引导支持的环境中引导，例如在有些故障转移中完成的操作。

有了这个更新，您现在可以配置 `zipl` 工具的安全引导选项。要做到这一点，您可以使用：

- Kickstart **zipl** 命令及其选项之一，例如：**--secure-boot**、**--no-secure-boot** 和 **--force-secure-boot**。
- 在 GUI 中的 **安装概述** 窗口中，您可以选择 **System > Installation Destination > Full disk summary** 和 **boot loader** 链接并设置引导设备。因此，现在可以在缺乏安全引导支持的环境中引导安装。

(BZ#1659400)

安全引导功能现在可用

在以前的版本中，**secure =** 引导选项的默认值没有被设置为 **auto**，因此无法使用安全引导功能。在这个版本中，除非之前配置了，否则默认值被设置为 **auto**，安全引导功能现在可用。

(BZ#1750326)

/etc/sysconfig/kernel 文件不再引用 new-kernel-pkg 脚本

在以前的版本中，**/etc/sysconfig/kernel** 文件引用 **new-kernel-pkg** 脚本。但是，RHEL 8 系统中没有包括 **new-kernel-pkg** 脚本。在这个版本中，对 **new-kernel-pkg** 脚本的引用已从 **/etc/sysconfig/kernel** 文件中被删除。

(BZ#1747382)

安装不会超过 boot-device NVRAM 变量中允许设备的最大数量

在以前的版本中，RHEL 8 安装程序在 **boot-device NVRAM** 变量中设置大于允许设备的最大数量。因此，在设备数量超过最大设备的系统中安装会失败。在这个版本中，RHEL 8 安装程序会检查最大设备设置，且只添加允许的设备数量。

(BZ#1748756)

安装适用于在非网络位置的 Kickstart 文件中使用 URL 命令的镜像位置

在以前的版本中，当镜像远程位置触发的网络激活在非网络位置的 Kickstart 文件中通过 URL 命令指定时，安装过程会在早期失败。在这个版本中解决了这个问题，以及使用位于非网络位置的 Kickstart 文件中的 URL 命令提供镜像位置的安装（如 CD-ROM 或本地块设备）可以正常工作。

(BZ#1649359)

RHEL 8 安装程序只检查 ECKD DASD 的未格式化设备

在以前的版本中，当检查未格式化的设备时，安装程序会检查所有 DASD 设备。但是，安装程序应该只检查了 ECKD DASD 设备。因此，当使用带有 SWAPGEN 的 FBA DASD 设备时，安装会失败。在这个版本中，安装程序不会检查 FBA DASD 设备，安装可以成功完成。

(BZ#1715303)

5.4.2. 软件管理

yum repolist 不再以第一个不可用存储库结束

在以前的版本中，存储库配置选项 **skip_if_unavailable** 默认设置为如下：

```
skip_if_unavailable=false
```

此设置强制 **yum repolist** 命令以第一个不可用存储库结束，并显示错误并退出状态 1。因此，**yum repolist** 不会继续列出可用的软件仓库。

有了这个更新，**yum repolist** 已被修复，不再需要任何下载。因此，**yum repolist** 不会提供任何需要元数据的输出，命令现在继续按预期列出可用的存储库。

请注意，可用软件包的数量仅由 **yum repolist --verbose** 或 **yum repoinfo** 返回，后者仍需要可用元数据。因此，这些命令将结束第一个不可用存储库。

(BZ#1697472)

5.4.3. Shell 和命令行工具

ReaR 更新

RHEL 8.2 对 Relax-and-Recover(**ReaR**)程序引入了多个更新。

构建目录处理已被更改。在以前的版本中，如果 **ReaR** 失败，构建目录会临时保留。在这个版本中，构建目录默认在非交互式运行中删除，以防止消耗磁盘空间。

KEEP_BUILD_DIR 配置变量的语义经过增强，以包含新的 **错误** 值。您可以将 **KEEP_BUILD_DIR** 变量设置为以下值：

- 在 **出错** 时保留构建目录以用于调试的错误（之前的行为）
- **y (true)** 以始终保留构建目录
- **N (false)** 永不保留构建目录

默认值为空字符串，其含义为以交互方式（在终端中）执行 **ReaR** 时的 **错误**；如果以非交互方式执行 **ReaR**，则默认值为 **false**。请注意，**KEEP_BUILD_DIR** 在调试模式(-d)和 debugscript 模式(-D)中自动设置为 **true**；此行为尚未更改。

主要程序错误修复包括：

- 修复了对 NetBackup 8.0 的支持。
- **Rear** 不再会中止 bash 错误与 **xrealloc** 类似的问题：无法分配给每个组有大量用户、组和用户的系统。
- **bconsole** 命令现在显示其提示，这可让您在使用 Bacula 集成时执行恢复操作。
- 当 **docker** 服务正在运行，但没有定义 **docker** root 目录，或者无法确定 **docker** 服务的状态时，**ReaR** 现在可以正确地备份文件。
- 当使用精简池或在 Migration Mode 中恢复系统时，恢复不再会失败。
- 在使用 LVM 的恢复过程中，**initramfs** 的重建非常慢。
- 现在，在使用 UEFI 引导装载程序时，**ReaR** 在 AMD 和 Intel 64 位构架中创建一个可引导 ISO 镜像。在这个设置中引导救援镜像不再在 Grub 中中止，并显示错误消息 **Unknown command 'configfile'(...).Entering rescue mode...**。此设置中对 GRUB_RESCUE 的支持（之前因为缺少 XFS 文件系统支持而失败）也已被修复。

(BZ#1729501)

mlocate-updatedb.timer 现在在 **mlocate** 软件包安装过程中启用

在以前的版本中，文件数据库重新索引不会被自动执行，因为 `mlocate-updatedb.timer` 计时器在 `mlocate` 软件包安装后被禁用。在这个版本中，`mlocate-updatedb.timer` 定时器现在是 `90-default.preset` 文件的一部分，在 `mlocate` 软件包安装后默认启用。因此，文件数据库会自动更新。

(BZ#1817591)

5.4.4. 基础架构服务

`dnsmasq` 现在可以正确地处理非递归 DNS 查询

在以前的版本中，`dnsmasq` 将所有非递归查询转发到上游服务器，这会导致不同的响应。在这个版本中，对本地已知名称（如 DHCP 主机租用名称或从 `/etc/hosts` 文件读取的主机）的非递归查询由 `dnsmasq` 处理，且不会转发到上游服务器。因此，返回与对已知名称的递归查询相同的响应。

(BZ#1700916)

在系统时间更改后，`dhclient` 不再无法续订 IP 地址

在以前的版本中，如果系统时间更改，系统可能会因为内核删除而丢失分配的 IP 地址。在这个版本中，`dhclient` 使用单调计时器检测向后时间跳转，并在系统时间停止跳转时发出 `DHCPREQUEST` 消息用于租用扩展。因此，系统不会在上述场景中丢失 IP 地址。

(BZ#1729211)

`ipcalc` 现在返回正确的 /31 网络的广播地址

在这个版本中，`ip calc` 工具被正确遵循 RFC 3021 标准。因此，当接口中使用 /31 前缀时，`ip calc` 会返回正确的广播地址。

(BZ#1638834)

`/etc/services` 现在包含正确的 NRPE 端口定义

在这个版本中，在 `/etc/services` 文件中添加了正确的 Nagios Remote Plug-in Executor(NRPE)服务端口定义。

(BZ#1730396)

`postfix` DNS 解析器代码现在使用 `res_search` 而不是 `res_query`

在 `postfix` 中的上一更新之后，DNS 解析器代码使用 `res_query` 功能而不是 `res_search` 功能。因此，DNS 解析器不会使用以下 `postfix` 配置在当前和父域中搜索主机名：

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

例如：

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

并且 `example.com` 格式的域名，DNS 解析器没有使用 `smtp.example.com` SMTP 服务器进行转发。

在这个版本中，DNS 解析器代码已被修改为使用 `res_search` 而不是 `res_query`，它现在可以正确地搜索当前和父域中的主机名。

(BZ#1723950)

PCRE、CDB 和 SQLite 现在可以与 Postfix 一起使用

在 RHEL 8 中，`postfix` 软件包被分成多个子软件包，每个子软件包为特定数据库提供一个插件。在以前的版本中，包含 `postfix-pcre`、`postfix-cdb` 和 `postfix-sqlite` 插件的 RPM 软件包不会被分发。因此，使用这些插件的数据库无法与 Postfix 一起使用。在这个版本中，将包含 PCRE、CDB 和 SQLite 插件的 RPM 软件包添加到 AppStream 仓库中。因此，可以在安装相应的 RPM 软件包后使用这些插件。

([BZ#1745321](#))

5.4.5. 安全性

`fapolicyd` 不再阻止 RHEL 更新

当更新替换了正在运行的应用程序的二进制文件时，内核会通过附加 "(deleted)" 后缀来修改内存中的应用程序二进制路径。在以前的版本中，`fapolicyd` 文件访问策略守护进程将此类应用程序视为不受信任的应用程序，并阻止它们打开和执行任何其他文件。因此，在应用更新后系统有时无法引导。

在 [RHBA-2020:5243](#) 公告提供发行版本中，`fapolicyd` 忽略二进制路径中的后缀，以便二进制文件与信任数据库匹配。因此，`fapolicyd` 会正确强制执行规则，更新过程也可以完成。

([BZ#1897091](#))

`openssl-pkcs11` 不再通过尝试登录到多个设备锁定设备

在以前的版本中，`openssl-pkcs11` 引擎会尝试使用提供的 PKCS #11 URI 登录搜索的第一个结果，即使第一个结果不是预期的设备和 PIN 匹配另一个设备。这些失败的身份验证尝试锁定该设备。

`OpenSSL-pkcs11` 现在，只有在提供的 PKCS #11 URI 匹配单个设备时才会尝试登录到设备。现在，如果 PKCS #11 搜索发现多个设备，引擎现在特意失败。因此，您必须提供一个 PKCS #11 URI，在使用 `openssl-pkcs11` 时仅匹配单个设备来登录该设备。

([BZ#1705505](#))

使用 `rpmverifyfile` 的 OpenSCAP 离线扫描现在可以正常工作

在以前的版本中，OpenSCAP 扫描程序没有以离线模式正确更改当前工作目录，并且未使用 `OpenSCAP rpmverifyfile` 探测中正确的参数调用 `fchdir` 功能。已修复 `OpenSCAP` 扫描程序，以离线模式正确更改当前工作目录，并且修复 `fchdir` 功能以在 `rpmverifyfile` 中使用正确的参数。因此，OpenSCAP 可以使用包含 OVAL `rpmverifyfile` 的 SCAP 内容扫描任意文件系统。

([BZ#1636431](#))

现在，如果在 PKCS #11 设备中没有匹配的公钥的情况下使用 ECDSA 私钥，`httpd` 现在可以正确启动

与 RSA 密钥不同，ECDSA 私钥不一定包含公钥信息。在这种情况下，您无法从 ECDSA 私钥获取公钥。因此，PKCS #11 设备将公钥信息存储在单独的对象中，无论是公钥对象还是证书对象。OpenSSL 预期引擎提供的 `EVP_PKEY` 结构用于存放公钥信息的私钥。填写要提供给 OpenSSL 的 `EVP_PKEY` 结构时，`openssl-pkcs11` 软件包中的引擎会尝试仅从匹配的公钥对象获取公钥信息，并忽略当前证书对象。

当 OpenSSL 从引擎请求 ECDSA 私钥时，提供的 `EVP_PKEY` 结构不包含公钥信息，如果 PKCS #11 设备中没有公钥，即使有匹配的证书可用，也是如此。因此，由于 Apache `httpd` web 服务器调用 `X509_check_private_key ()` 函数，这在启动过程中需要公钥，因此 `httpd` 在此场景下无法启动。如果公钥对象不可用，可以通过从证书加载 EC 公钥来解决此问题。现在，当 ECDSA 密钥存储在 PKCS #11 设备中时，`httpd` 可以被正确启动。

([BZ#1664807](#))

现在，审计规则的 SCAP-security-guide PCI-DSS 补救可以正常工作

在以前的版本中，**scap-security-guide** 软件包包含补救的组合和可导致以下情况之一的检查：

- 错误修复审计规则
- 扫描评估包含通过的规则被标记为失败的假正值

因此，在 RHEL 安装过程中，扫描安装的系统会报告一些审计规则为失败或错误。

在这个版本中，补救已被修复，使用 PCI-DSS 安全策略安装的系统不再为审计规则报告假的正状态。

([BZ#1754919](#))

OpenSCAP 现在提供虚拟机和容器的离线扫描

在以前的版本中，重构 OpenSCAP 代码库会导致某些 RPM 探测无法以离线模式扫描虚拟机和容器文件系统。因此，下列工具无法包含在 **openscap-utils** 软件包中：**oscap-vm** 和 **oscap-chroot**。另外，**openscap-containers** 软件包已从 RHEL 8 中完全删除。在这个版本中，探测的问题已被解决。

因此，RHEL 8 现在在 **openscap-utils** 软件包中包含 **oscap-podman**、**oscap-vm** 和 **oscap-chroot** 工具。

([BZ#1618489](#))

OpenSCAP rpmverifypackage 现在可以正常工作

在以前的版本中，rpm **verifypackage** 探测调用 **chdir** 和 **chroot** 系统调用两次。因此，在使用自定义 Open Vulnerability 和评估语言(OVAL)内容的 OpenSCAP 扫描过程中使用该探测时会出现错误。已修复 **rpmverifypackage** 探测，以正确使用 **chdir** 和 **chroot** 系统调用。因此，rpm **verifypackage** 现在可以正常工作。

([BZ#1646197](#))

5.4.6. 网络

现在锁定在 qdisc_run 功能中不会导致内核崩溃

在以前的版本中，当删除流量时 **pfifo_fast** 队列强制重置时，在数据包释放后会导致数据包传输的竞争条件。因此，有时内核被意外终止。有了这个更新，**qdisc_run** 函数中的锁已被改进。因此，内核不会在上述情况下崩溃。

([BZ#1744397](#))

org.fedoraproject 中的 DBus API.FirewallD1.config.service 可以按预期工作

在以前的版本中，DBus API **getIncludes**、**setIncludes** 和 **queryIncludes** 功能在 **org.fedoraproject.FirewallD1** 中返回错误消息：**org.fedoraproject.FirewallD1.Exception**：因为索引错误，列出索引范围。在这个版本中，DBus API **getIncludes**、**setIncludes** 和 **queryIncludes** 功能可以正常工作。

([BZ#1737045](#))

在卸载 ipv6 模块时，RHEL 不再记录内核警告

在以前的版本中，IP 虚拟服务器(**ipvs**)模块使用不正确的参考计数，这会导致卸载模块时出现竞争条件。因此，RHEL 记录了内核警告。在这个版本中解决了竞争条件。因此，当您卸载 **ipvs** 模块时，内核不再记录警告。

(BZ#1687094)

nft 实用程序在第一个非选项参数后不再将参数解释为命令行选项

在以前的版本中，**nft** 工具在 **nft** 命令中接受任何位置的选项。例如，管理员可以在非选项参数之间或之后使用选项。因此，由于前导短划线，**nft** 将负优先级值解析为选项，命令会失败。**nft** 实用程序的命令行解析器已更新，不解释读取第一个非选项参数后以短划线开头的参数。因此，管理员不再需要将负优先级值传递给 **nft**。

请注意，由于此更改，您必须在第一个非选项参数之前将所有命令选项传递到 **nft**。在更新之前，请验证 **nftables** 脚本是否满足这一新条件，以确保安装此更新后脚本可以按预期工作。

(BZ#1778883)

/etc/hosts.allow 和 /etc/hosts.deny 文件不再包含对删除的 tcp_wrappers 的过时引用

在以前的版本中，**/etc/hosts.allow** 和 **/etc/hosts.deny** 文件包含有关 **tcp_wrappers** 软件包的过期信息。这些文件在 RHEL 8 中被删除，因为删除的 **tcp_wrappers** 不再需要这些文件。

(BZ#1663556)

在 firewalld 中添加了一个配置参数来禁用区域偏移

在以前的版本中，**firewalld** 服务包含一个未记录的行为，称为"zone drifting"。RHEL 8.0 删除了这个行为，因为它可能会造成负面影响。因此，在使用这个行为配置 **catch-all** 或 **fallback zone** 的主机上，**firewalld** 会拒绝之前允许的连接。在这个版本中，重新添加了区 **drifting** 功能，但作为一个可配置的功能。现在，用户可以决定使用区 **drifting** 功能，也可以禁用它并使用防火墙进行安全设置。

默认情况下，在 RHEL 8.2 中，**/etc/firewalld/firewalld.conf** 文件中的新 **AllowZoneDrifting** 参数被设置为 **yes**。请注意，如果启用了该参数，**firewalld** 日志：

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

(BZ#1772208)

5.4.7. 内核

现在完全支持子部分内存热插拔

在以前的版本中，一些平台与物理内存区域一致，如 **Dual In-Line Modules(DIMM)** 和 **interleave** 设置为 **64MiB** 内存边界。但是，由于 Linux 热插拔子系统使用 **128MiB** 的内存大小，热插拔新设备会导致多个内存区域在单个热插拔内存窗口中重叠。因此，这会导致使用以下或类似的调用 **trace** 列出可用持久内存命名空间失败：

```
WARNING: CPU: 38 PID: 928 at arch/x86/mm/init_64.c:850
add_pages+0x5c/0x60
[.]
RIP: 0010:add_pages+0x5c/0x60
[.]
Call Trace:
devm_memremap_pages+0x460/0x6e0
```

```
pmem_attach_disk+0x29e/0x680 [nd_pmem]
? nd_dax_probe+0xfc/0x120 [libnvdimm]
nvdimm_bus_probe+0x66/0x160 [libnvdimm]
```

在这个版本中解决了这个问题，并支持 Linux 热插拔子系统来启用多个内存区域共享一个热插拔内存窗口。

(BZ#1724969)

数据崩溃现在触发 BUG 而不是 WARN 消息

在这个版本中，lib `/list_debug.c` 列表会触发 BUG，它会生成带有 **vmcore** 的报告。在以前的版本中，当遇到数据崩溃时，会生成一个简单的 WARN，这很可能不被留意。设置

CONFIG_BUG_ON_DATA_CORRUPTION 后，内核现在会创建一个崩溃并触发 BUG 来应对数据损坏。这可防止进一步损坏并降低安全风险。**kdump** 现在会生成 **vmcore**，它可改进数据损坏错误报告。

(BZ#1714330)

提供了对 Intel Carlsville 卡的支持，但不在 RHEL 8.2 中验证

提供了 **Intel Carlsville** 卡支持，但不在 Red Hat Enterprise Linux 8.2 上进行测试。

(BZ#1720227)

RPS 和 XPS 不再将作业放在隔离的 CPU 中

在以前的版本中，接收数据包提取(RPS)软件队列机制和传输包服务关联(XPS)可传输在所有 CPU 集合上分配的队列选择机制，包括隔离的 CPU。因此，当对延迟敏感的工作负载使用与运行 RPS 或 XPS 作业相同的 CPU 时，可能会导致实时环境中出现意外延迟。在这个版本中，`storage_rps_map ()` 函数不包括任何隔离的 CPU 用于 RPS 配置。同样，用于 XPS 配置的内核驱动程序也尊重 CPU 隔离。因此，在上述场景中，RPS 和 XPS 不再将作业放在隔离的 CPU 上。如果在 `/sys/devices/pci*/net/dev/queues/rx-*/rps_cpus` 文件中配置隔离 CPU，会出现以下错误：

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

但是，在 `/sys/devices/pci*/net/dev/queues/tx-*/xps_cpus` 文件中手动配置隔离 CPU 成功在隔离的 CPU 上分配 XPS 作业。

请注意，在具有隔离 CPU 的环境中网络工作负载可能会遇到一些性能变化。

(BZ#1867174)

5.4.8. 文件系统和存储

SCSI 驱动程序不再使用过多内存

在以前的版本中，某些 SCSI 驱动程序使用的内存比 RHEL 7 中的内存更大。在某些情况下，比如在光纤通道主机总线适配器(HBA)上创建 vPort，内存用量过量，具体取决于系统配置。

内存用量增加是由块层中内存预分配造成的。多队列块设备调度(BLK-MQ)和每个 I/O 请求的多队列 SCSI 堆栈(SCSI-MQ)预分配内存，从而增加内存用量。

在这个版本中，块层限制预分配的内存量，因此 SCSI 驱动程序不再使用过多内存。

(BZ#1698297)

现在，VDO 可以在 UDS 完成重建前挂起

在以前的版本中，如果您试图在 UDS 索引重新构建时挂起 VDO 卷，`dmsetup suspend` 命令就会变得无响应。该命令仅在重建后完成。

在这个版本中，这个问题已被解决。`dmsetup suspend` 命令可以在 UDS 重新构建之前完成，而不会变得无响应。

(BZ#1737639)

5.4.9. 动态编程语言、网页和数据库服务器

mod_cgid 日志记录中的问题已被解决

在更新之前，如果在线程多处理模块(MPM)下使用了 `mod_cgid` Apache `httpd` 模块，则会出现以下日志记录问题：

- CGI 脚本的 `stderr` 输出没有使用标准时间戳信息作为前缀。
- 如果已配置，CGI 脚本的 `stderr` 输出没有正确重定向到特定于 `VirtualHost` 的日志文件。

在这个版本中解决了这个问题，`mod_cgid` 日志记录现在可以正常工作。

(BZ#1633224)

5.4.10. 编译器和开发工具

如果 `dlopen` 失败，`unrelocated` 和未初始化共享对象不再会导致失败

在以前的版本中，如果 `dlopen` 调用失败，`glibc` 动态链接程序不会在报告错误前删除带有 `NODELETE` 标记的共享对象。因此，未重新定位和未初始化的共享对象会保留在进程镜像中，最终会导致断言失败或崩溃。在这个版本中，动态加载器使用待处理的 `NODELETE` 状态在 `dlopen` 失败时删除共享对象，然后将其标记为 `NODELETE`。因此，进程不会留下任何未重新定位的对象。另外，在 ELF 构造函数和析构函数运行时的惰性绑定失败现在会终止进程。

(BZ#1410154)

64 位 ARM 架构中的高级 SIMD 功能在临时解决方案后不再复杂

在以前的版本中，当延迟解析高级 SIMD 功能时，用于高级 SIMD 的新向量标准(PCS)无法正确保存和恢复某些调用者保存的寄存器。因此，二进制文件可能会在运行时发生错误。在这个版本中，符号表中的 Advanced SIMD 和 SVE 向量函数标记为 `.variant_pcs`，因此动态链接器会早期绑定这些函数。

(BZ#1726641)

sudo wrapper 脚本现在解析选项

在以前的版本中，`/opt/redhat/devtoolset*/root/usr/bin/sudo` wrapper 脚本无法正确解析 `sudo` 选项。因此，无法执行一些 `sudo` 选项（如 `sudo -i`）。在这个版本中，可以正确地解析更多 `sudo` 选项，因此 `sudo` 打包程序脚本的工作方式更像 `/usr/bin/sudo`。

(BZ#1774118)

修复了 `glibc` 中的 TLS 变量对齐的问题

在以前的版本中，在某些情况下，一致的线程本地存储(TLS)数据可能会实例化而无需预期的对齐。在这个版本中，POSIX Thread Library `libpthread` 已被改进，以确保在任何条件下都正确一致。因此，为所有带有正确对齐的线程正确实例化了一致的 TLS 数据。

(BZ#1764214)

EINTR 或 EAGAIN 错误后重复的 pututxline 调用不再破坏 utmp 文件

当 `pututxline` 函数试图获取锁定且不成功时，该函数会返回 `EINTR` 或 `EAGAIN` 错误代码。在以前的版本中，如果立即调用 `pututxline` 并被管理来获取锁定，则它没有在 `utmp` 文件中使用已分配的匹配插槽，而是添加另一个条目。因此，这些未使用的条目极大地增加了 `utmp` 文件的大小。在这个版本中解决了这个问题，条目已正确添加到 `utmp` 文件中。

(BZ#1749439)

当发生内部故障时，mtrace 不再挂起

在以前的版本中，`mtrace` 工具实现中的一个缺陷可能会导致内存追踪挂起。为解决这个问题，`mtrace` 内存追踪实现功能更强，即使在内部故障的情况下也能避免挂起。现在，用户可以调用 `mtrace`，它不再挂起，从而在有限的时间里完成。

(BZ#1764235)

fork 功能可避免使用 pthread_atfork 相关的某些死锁

在以前的版本中，如果程序注册了 `in fork` 处理程序并从异步信号处理程序调用 `fork`，则依赖内部实现的锁定中的缺陷可能会导致程序停滞。在这个版本中，对 `fork` 及其 `for fork 处理程序` 的实施进行了调整，以避免单线程程序中出现死锁。

(BZ#1746928)

strstr 不再返回对截断模式的不正确的匹配

在某些 IBM Z 平台 (z15, 以前称为 arch13) 上，`strstr` 函数在处理跨页面边界的搜索模式时无法正确更新 CPU 寄存器。因此，`strstr` 返回了不正确的匹配项。在这个版本中解决了这个问题，`strstr` 在上面提到的场景中可以正常工作。

(BZ#1777241)

glibc 中的 C.UTF-8 本地源省略表达式已被修复

在以前的版本中，C.UTF-8 源区域中的一个缺陷会导致 U+10000 之外的所有 Unicode 代码点都缺乏校验权重。因此，所有超过 U+10000 的代码点均无法按预期互操作。C.UTF-8 源区域设置已被修正，新编译的二进制区域现在对所有 Unicode 代码点具有排序权重。因为这个修复，编译的 C.UTF-8 区域设置会较大 5.3MiB。

(BZ#1361965)

当在不调用 setpwent() 的情况下调用 getpwent () 时，glibc 不再会失败。

如果您的 `/etc/nsswitch.conf` 文件指向 Berkeley DB(db) 密码提供商，则可以使用 `getpwent ()` 函数来请求数据，而无需首先调用 `setpwent ()`。当您调用 `endpwent ()` 函数时，在没有首先调用 `setpwent()` 的情况下进一步调用 `getpwent()` 会导致 `glibc` 失败，因为 `endpwent ()` 无法重置间隔以允许新的查询。在这个版本中解决了这个问题。因此，在使用 `endpwent ()` 结束一个查询后，对 `getpwent ()` 的其他调用将启动一个新的查询，即使您没有调用 `setpwent ()`。

(BZ#1747502)

ltrace 现在可以跟踪强化二进制文件中的系统调用

在以前的版本中，`ltrace` 在 AMD 和 Intel 64 位构架中不会在特定强化的二进制文件（如系统二进制文件）中产生任何结果。在这个版本中，`ltrace` 可以跟踪强化的二进制文件中的系统调用。

(BZ#1655368)

Intel 的 JCC 漏洞不再会在 GCC 编译器中造成大量性能损失

某些 Intel CPU 受到 Jump Conditional Code(JCC)错误影响，从而导致机器指令被错误地执行。因此，受影响的 CPU 可能无法正确执行程序。完整的修复涉及更新有安全漏洞的 CPU 的微码，这可能会导致性能下降。在这个版本中，在 assembler 中启用了临时解决方案，可帮助减少性能丢失。**默认情况下不启用临时解决方案。**

要应用该临时解决方案，请使用 GCC 重新编译带有 **-Wa,-mbranches-with-32B-boundaries** 命令行选项的程序。使用这个命令行选项重新编译的程序不会受到 JCC 缺陷的影响，但 microcode 更新仍然需要完全保护系统。

请注意，应用临时解决方案会增加程序的大小，并可能导致性能下降，尽管没有重新编译应该小于它。

([BZ#1777002](#))

使用并行构建时 make 不再减慢

在以前的版本中，当运行并行构建时，当等待其轮流运行时，子进程可能会临时变得无响应。因此，带有高 **-j** 值的构建会减慢或以较低有效 **-j** 值运行。在这个版本中，**make** 的作业控制逻辑没有阻塞。因此，具有高 **-j** 值的构建会以全 **-j** 速度运行。

([BZ#1774790](#))

ltrace 工具现在正确报告功能调用

由于改进了应用于所有 RHEL 组件的二进制强化，**ltrace** 工具之前无法在 RHEL 组件的二进制文件中检测功能调用。因此，**ltrace** 输出为空，因为它没有报告在此类二进制文件中使用任何检测到的调用。这个更新解决了 **ltrace** 处理函数调用的方式，这防止了上面描述的问题的发生。

([BZ#1618748](#))

5.4.11. Identity Management

dsctl 实用程序不再无法在其名称中管理带有连字符的实例

在以前的版本中，**dsctl** 实用程序无法正确解析 Directory Server 实例名称中的连字符。因此，管理员无法使用 **dsctl** 管理名称中带有连字符的实例。在这个版本中解决了这个问题，**dsctl** 现在可以正常工作。

([BZ#1715406](#))

目录服务器实例名称现在可以有最多 103 个字符

当 LDAP 客户端建立与目录服务器的连接时，服务器会将与客户端地址相关的信息存储在本地缓冲区中。在以前的版本中，这个缓冲区的大小太小，无法存储超过 46 个字符的 LDAP 路径名称。例如，如果 Directory Server 实例的名称太长，就会出现这种情况。因此，服务器因为缓冲区溢出而意外终止。在这个版本中，缓冲区大小增加到 Netscape Portable Runtime(NSPR)库对路径名称支持的最大大小。因此，在上述场景中 Directory 服务器不再崩溃。

请注意，由于 NSPR 库中的限制，实例名称可以最多 103 个字符。

([BZ#1748016](#))

pkidestroy 工具现在选择正确的实例

在以前的版本中，在一个半删除的实例上执行的 **pkidestroy --force** 命令默认选择 **pki-tomcat** 实例，无论使用 **-i** 实例选项指定的实例名称是什么。

因此，这会删除 **pki-tomcat** 实例，而不是预期的实例，**--remove-logs** 选项不会删除预期的实例日志。**pkidestroy** 现在应用正确的实例名称，仅删除预期实例的左边。

(BZ#1698084)

sssd-ldap man page 中更新了 **ldap_user_authorized_service** 描述

RHEL 8 中更改了可插拔验证模块(PAM)堆栈。例如，**systemd** 用户会话现在使用 **systemd -user** PAM 服务启动 PAM 对话。此服务现在递归包含 **system-auth** PAM 服务，其中可能包括 **pam_sss.so** 接口。这意味着始终调用 SSSD 访问控制。

在为 RHEL 8 系统设计访问控制规则时，您应该了解这一变化。例如，您可以将 **systemd-user** 服务添加到允许的服务列表中。

请注意一些访问控制机制，如 IPA HBAC 或 AD GPOs，默认情况下 **systemd-user** 服务已添加到允许的服务列表中，您不需要进行任何操作。

sssd-ldap man page 已更新，以包含此信息。

(BZ#1669407)

现在，在 IdM 中启用 AD 信任支持时会显示有关所需 DNS 记录的信息

在以前的版本中，当启用对带有外部 DNS 管理的 Red Hat Enterprise Linux Identity Management(IdM)安装中的 Active Directory(AD)信任时，不会显示有关所需 DNS 记录的信息。手动输入 **ipa dns-update-system-records --dry-run** 命令是 necessary，以获取 IdM 所需的所有 DNS 记录列表。

在这个版本中，**ipa -adtrust-install** 命令可以正确地列出 DNS 服务记录，以便手动添加 DNS 区域。

(BZ#1665051)

现在，在带有集成 DNS 的 IdM 服务器中默认禁用递归 DNS 查询

在以前的版本中，在使用带有集成 DNS 的身份管理(IdM)服务器时，默认启用递归查询。因此，有可能使用服务器进行 DNS 放大攻击。有了这个更新，现在默认禁用递归 DNS 查询，且不再使用服务器进行 DNS 放大攻击。

您可以通过修改 IdM 服务器上的集成 DNS 配置来手动允许对特定客户端的递归查询。例如，要允许任何人查询服务器以获取权威数据，但只允许可信客户端访问您的缓存和递归，请在 **trusted** 访问控制列表 (ACL)中列出客户端：

1. 在 **/etc/named/ipa-ext.conf** 文件中创建一个 **trusted** ACL：

```
acl "trusted" {
    192.168.0.0/16;
    10.153.154.0/24;
    localhost;
    localnets;
};
```

2. 将 **trusted** ACL 添加到 **/etc/named/ipa-options-ext.conf** 文件中：

```
allow-query { any; };
allow-recursion { trusted; };
allow-query-cache { trusted; };
```

(BZ#2151696)

5.4.12. Desktop

当使用软件渲染器时，Wayland 上的 GNOME Shell 不再缓慢

在以前的版本中，当使用软件渲染器时，GNOME Shell 的 Wayland 后端没有使用可缓存的帧缓冲程序。因此，与 X.org 后端的软件渲染的 GNOME Shell 相比，Wayland 上的软件渲染 GNOME Shell 速度较慢。

在这个版本中，Wayland 的 GNOME Shell 中添加了中间的 shadow 帧缓冲。因此，Wayland 上的软件渲染的 GNOME Shell 现在和 X.org 上的 GNOME Shell 一样执行。

(BZ#1737553)

5.4.13. 虚拟化

在 10 代 Intel Core 处理器中启动虚拟机不再失败

在以前的版本中，在使用第 10 代 Intel Core 处理器（也称为 Icelake-Server）的主机模型中启动虚拟机 (VM) 会失败。在这个版本中，lib virt 不再尝试禁用 QEMU 不支持的 **pconfig** CPU 功能。因此，在运行 10 代 Intel 处理器的主机模型中启动虚拟机不再会失败。

(BZ#1749672)

使用 cloud-init 在 Microsoft Azure 上置备虚拟机现在可以正常工作

在以前的版本中，无法使用 **cloud-init** 工具在 Microsoft Azure 平台上置备 RHEL 8 虚拟机 (VM)。在这个版本中，Azure 端点的 **cloud-init** 处理已被修复，Azure 上的 RHEL 8 虚拟机现在可以按预期进行。

(BZ#1641190)

RHEL 7 主机上的 RHEL 8 虚拟机可在高于 1920x1200 的分辨率中可靠地查看

在以前的版本中，当使用在 RHEL 7 主机系统中运行的 RHEL 8 虚拟机 (VM) 时，显示虚拟机图形输出（如在 kiosk 模式下运行应用程序）的某些方法不能超过 1920x1200。因此，即使主机硬件支持更高的分辨率，使用这些方法显示虚拟机也只能达到 1920x1200 的分辨率。这个版本会调整 DRM 和 QXL 驱动程序，以防止上述问题的发生。

(BZ#1635295)

使用 cloud-init 自定义 ESXi 虚拟机并重启虚拟机现在可以正常工作

在以前的版本中，如果 **cloud-init** 服务用于修改在 VMware ESXi hypervisor 上运行的虚拟机 (VM) 以使用静态 IP，然后克隆虚拟机，则新的克隆虚拟机在某些情况下需要很长时间才能重启。在这个版本中，**cloud-init** 不会将虚拟机的静态 IP 重新写入 DHCP，这可防止上述问题发生。

(BZ#1666961, BZ#1706482)

5.4.14. 容器

从 quay.io 注册表中提取镜像不再会导致意外的镜像

在以前的版本中，在 `/etc/containers/registries.conf` 提供的默认 **registry** 搜索列表中列出 **quay.io** 容器镜像 registry 可允许用户在使用简短名称时拉取欺骗镜像。为解决这个问题，quay.io 容器镜像 registry 已从 `/etc/containers/registries.conf` 的默认 **registry** 搜索列表中删除。因此，从 **quay.io** registry 中拉取镜像现在要求用户指定完整的存储库名称，如 **quay.io/myorg/myimage**。quay.io registry 可以添加回 `/etc/containers/registries.conf` 中的默认 **registry** 搜索列表中，以使用短名称重新启用拉取容器镜像，但不建议这样做，因为它可能会造成安全风险。

(BZ#1784267)

5.5. 技术预览

这部分提供了 Red Hat Enterprise Linux 8.2 中所有可用技术预览的列表。

如需有关红帽对技术预览功能支持范围的信息，请参阅 [技术预览功能支持范围](#)。

5.5.1. 网络

nmstate 作为技术预览提供

nmstate 是主机的网络 API。**nmstate** 软件包作为技术预览提供库和 **nmstatectl** 命令行实用程序以声明性方式管理主机网络设置。网络状态由预定义的 schema 描述。报告当前状态以及对所需状态的更改都符合 schema。

详情请查看 `/usr/share/doc/nmstate/README.md` 文件以及 `/usr/share/doc/nmstate/examples` 目录中的示例。

(BZ#1674456)

AF_XDP 作为技术预览

Address Family eXpress Data Path (AF_XDP) 是设计用于处理高性能数据包。它包含 **XDP**，并允许通过编程方式将选定的数据包高效地重定向到用户空间应用，以便进一步处理。

(BZ#1633143)

XDP 作为技术预览提供

eXpress Data Path (XDP) 功能作为技术预览提供。它提供了在内核入口数据路径早期为高性能数据包处理附加扩展 Berkeley Packet Filter (eBPF) 程序的方法，从而可以进行高效的可编程数据包分析、过滤和操作。

(BZ#1503672)

KTLS 作为技术预览提供

在 Red Hat Enterprise Linux 8 中，KTLS 是作为技术预览提供的。KTLS 使用内核中的对称加密或者解密算法为 AES-GCM 密码处理 TLS 记录。KTLS 还提供将 TLS 记录加密卸载到支持此功能的网络接口控制器(NIC)的接口。

(BZ#1570255)

The dracut 工具现在支持使用 NetworkManager 作为技术预览创建 initrd 镜像

默认情况下，**dracut** 实用程序使用 shell 脚本管理初始 RAM 磁盘(**initrd**)中的网络。在某些情况下，当系统从 RAM 磁盘切换到使用 NetworkManager 配置网络的操作系统时，这可能会造成问题。例如，NetworkManager 可以发送另一个 DHCP 请求，即使 RAM 磁盘中的脚本已请求了 IP 地址。这个来自 RAM 磁盘的请求可能会导致超时。

要解决这类问题，RHEL 8.2 中的 **dracut** 现在可以在 RAM 磁盘中使用 NetworkManager。使用以下命令启用该功能并重新创建 RAM 磁盘镜像：

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

请注意，红帽不支持技术预览功能。但是，为了提供有关此功能的反馈，请联系红帽支持。

(BZ#1626348)

mlx5_core 驱动程序支持 Mellanox ConnectX-6 Dx 网络适配器作为技术预览

在这个版本中，Mellanox ConnectX-6 Dx 网络适配器的 PCI ID 被添加到 **mlx5_core** 驱动中。在使用这个适配器的主机上，RHEL 会自动加载 **mlx5_core** 驱动程序。请注意，红帽将此功能作为不受支持的技术预览提供。

(BZ#1687434)

systemd-resolved 服务现在作为技术预览提供

systemd-resolved 服务为本地应用提供名字解析。该服务实现了缓存和验证 DNS stub 解析器、链接本地多播名称解析 (LLMNR) 以及多播 DNS 解析器和响应程序。

请注意，即使 **systemd** 软件包提供 **systemd-resolved**，这个服务还是一个不受支持的技术预览。

(BZ#1906489)

5.5.2. 内核

kexec fast reboot 作为技术预览

kexec fast reboot 功能仍作为技术预览提供。重启现在要快得多，这归功于 **kexec 快速重启**。要使用这个功能，请手动加载 **kexec** 内核，然后重启操作系统。

(BZ#1769727)

eBPF 作为技术预览

Extended Berkeley Packet Filter(eBPF) 是一个内核中的虚拟机,允许在可访问有限功能的受限沙箱环境中在内核空间中执行代码。

虚拟机包含新系统调用 **bpf()**，它支持生成各种映射类型，同时还允许在特殊的装配类代码中载入程序。然后，代码被加载到内核，并使用即时编译方式转换为原生机器代码。请注意,只有具有 **CAP_SYS_ADMIN** 能力的用户（如 root 用户）才可以成功使用 **bpf()** syscall。更多信息，请参阅 **bpf(2)** man page。

载入的程序可附加到不同的点（套接字、追踪点、数据包）来接收和处理数据。

红帽提供的很多组件都使用 **eBPF** 虚拟机。每个组件处于不同的开发阶段，因此目前并不完全支持所有组件。所有组件都作为技术预览提供，除非有特定组件被显示为受支持。

以下显著的 **eBPF** 组件当前还作为技术预览提供：

- **bpftrace** 是使用 **eBPF** 虚拟机的高级别追踪语言。
- **eXpress Data Path(XDP)**功能，是一种网络技术，它允许使用 **eBPF** 虚拟机在内核中快速处理数据包。

(BZ#1559616)

libbpf 作为技术预览提供

libbpf 软件包目前作为技术预览提供。**libbpf** 软件包对于 **bpf** 相关应用程序（如 **bpftrace** 和 **bpf/xdp** 开发）至关重要。

它是 bpf-next linux tree **bpf-next/tools/lib/bpf** 目录及其支持的标题文件镜像。软件包的版本反映了应用程序二进制接口(ABI)的版本。

(BZ#1759154)

igc 驱动程序作为 RHEL 8 的一个技术预览

igc Intel 2.5G 以太网 Linux 有线 LAN 驱动程序现在可在 RHEL 8 的所有架构中作为技术预览使用。**ethtool** 还支持 **igc** 有线 LAN。

(BZ#1495358)

soft-RoCE 作为技术预览提供

通过融合以太网的远程直接内存访问(RDMA)是一个网络协议，通过以太网实现 RDMA。soft-RoCE 是 RoCE 的软件实施，它支持两种协议版本：RoCE v1 和 RoCE v2。在 RHEL 8 中，Soft-RoCE 驱动程序 **rdma_rxe** 作为不受支持的技术预览提供。

(BZ#1605216)

5.5.3. 文件系统和存储

NVMe/TCP 作为技术预览提供

通过 TCP/IP 网络(NVMe/TCP)访问和共享 Nonvolatile Memory Express(NVMe/TCP)存储及其对应的 **nvme-tcp.ko** 和 **nvmet-tcp.ko** 内核模块已被添加为技术预览。

使用 **nvme-cli** 和 **nvmetcli** 软件包提供的工具可以管理 NVMe/TCP 作为存储客户端或目标。

NVMe/TCP 目标技术预览仅用于测试目的，目前没有计划提供全面支持。

(BZ#1696451)

现在 ext4 和 XFS 作为技术预览提供文件系统 DAX

在 Red Hat Enterprise Linux 8.2 中，文件系统 DAX 作为技术预览提供。DAX 提供了将持久内存直接映射到其地址空间的方法。要使用 DAX，系统必须有某种可用的持久内存，通常使用一个或多个非线内存模块(NVDIMM)，且必须在 NVDIMM 上创建支持 DAX 的文件系统。另外，该文件系统必须使用 **dax** 挂载选项挂载。然后，在 **dax** 挂载的文件系统中的一个文件 **mmap** 会把存储直接映射到应用程序的地址空间中。

(BZ#1627455)

OverlayFS

OverlayFS 是一种联合文件系统。它允许您在另一个文件系统上覆盖一个文件系统。更改记录在上面的文件系统中，而较小的文件系统则未修改。这允许多个用户共享文件系统镜像，如容器或 DVD-ROM，基础镜像使用只读介质。

在大多数情况下，OverlayFS 仍是一个技术预览。因此，当这个技术被激活时，内核会记录警告信息。

与支持的容器引擎 (**podman**、**cri-o** 或 **buildah**) 一同使用时，对 OverlayFS 提供的全面支持包括以下限制：

- OverlayFS 仅支持作为容器引擎图形驱动程序或其他专用用例使用，如 squashed **kdump** initramfs。它主要用于容器 COW 内容，不支持持久性存储。您必须将任何持久性存储放在非 OverlayFS 卷中。您只能使用默认容器引擎配置：一个级别的覆盖、一个较低 **dir** 以及较低级别和上一级都位于同一个文件系统中。

- 目前只支持 XFS 作为较低层文件系统使用。

另外，以下规则和限制适用于使用 OverlayFS:

- OverlayFS 内核 ABI 和用户空间的行为被视为不稳定，将来的更新可能会改变。
- OverlayFS 提供一组受限的 POSIX 标准。在使用 OverlayFS 部署前，先测试您的应用程序。以下情况与 POSIX 不兼容：
 - **O_RDONLY** 打开的较低文件在读取文件时不会接收 **st_atime** 更新。
 - 使用 **O_RDONLY** 打开的较低文件，然后与 **MAP_SHARED** 映射与后续修改不一致。
 - RHEL 8 中不默认启用完全兼容 **st_ino** 或 **d_ino** 值，但您可以使用模块选项或挂载选项为它们启用完整的 POSIX 合规性。
要获得一致的内节点编号，请使用 **xino=on** 挂载选项。

您还可以使用 **redirect_dir=on** 和 **index=on** 选项提高 POSIX 合规性。这两个选项使上层的格式与没有这些选项的 overlay 不兼容。也就是说，如果您使用 **redirect_dir=on** 或 **index=on** 创建一个覆盖得到了意外的结果或错误，请卸载覆盖，然后不使用这些选项挂载覆盖。

- 要确定现有 XFS 文件系统是否有资格用作 overlay，请使用以下命令查看是否启用了 **ftype=1** 选项：

```
# xfs_info /mount-point | grep ftype
```

- 使用 OverlayFS 在所有支持的容器引擎中默认启用 SELinux 安全标签。
- 本发行版本中与 OverlayFS 相关的几个已知问题。详情请查看 [Linux 内核文档](#) 中的 *非标准行为*。

有关 OverlayFS 的更多信息，请参阅 [Linux 内核文档](#)。

(BZ#1690207)

Stratis 现在作为技术预览提供

Stratis 是一个新的本地存储管理器。它在存储池的上面为用户提供额外的功能。

Stratis 可让您更轻松地执行存储任务，比如：

- 管理快照和精简配置
- 根据需要自动增大文件系统大小
- 维护文件系统

要管理 Stratis 存储，使用 **stratis** 工具来与 **stratisd** 后台服务进行通信。

Stratis 作为技术预览提供。

如需更多信息，请参阅 Stratis 文档：[设置 Stratis 文件系统](#)。

RHEL 8.2 将 Stratis 更新至 2.0.0 版本。此版本提高了可靠性和 Stratis DBus API。

(JIRA:RHELPLAN-1212)

IdM 现在支持在 IdM 域成员中将 Samba 服务器设置为技术预览

在这个版本中，您可以在 Identity Management(IdM)域成员中设置 Samba 服务器。由同名软件包提供的新 `ipa-client-samba` 程序为 IdM 添加了特定于 Samba 的 Kerberos 服务主体并准备 IdM 客户端。例如，实用程序使用 `sss` ID 映射后端的 ID 映射配置创建 `/etc/samba/smb.conf`。现在，管理员可以在 IdM 域成员中设置 Samba。

由于 IdM Trust Controller 不支持全局目录服务，AD-enrolled Windows 主机无法在 Windows 中找到 IdM 用户和组。另外，IdM Trust Controller 不支持使用分布式计算环境/远程过程调用(DCE/RPC)协议解析 IdM 组。因此，AD 用户只能访问 IdM 客户端的 Samba 共享和打印机。

详情请查看在 [IdM 域成员中设置 Samba](#)。

(JIRA:RHELPLAN-13195)

5.5.4. 高可用性和集群

pacemaker podman bundles 作为技术预览

pacemaker 容器捆绑包现在在 `podman` 容器平台上运行，容器捆绑包功能作为一个技术预览可用。其中一个功能例外于技术预览：红帽完全支持在 Red Hat Openstack 中使用 Pacemaker 捆绑包。

(BZ#1619620)

作为技术预览的 `corosync-qdevice` 中的 Heuristics

Heuristics 是一组在启动、集群成员资格更改、成功连接到 `corosync-qnetd` 时本地执行的命令，以及可选的定期执行的命令。当所有命令及时成功完成（返回的错误代码为零），代表 heuristics 通过，否则代表失败。Heuristics 结果发送到 `corosync-qnetd`，在计算中用来决定哪个分区应该是 quorate。

([BZ#1784200](#))

新的 `fence-agents-heuristics-ping` 保护代理

作为技术预览，Pacemaker 现在支持 `fence_heuristics_ping` 代理。这个代理旨在打开一组实验性保护代理，它们本身没有实际隔离，而是以新的方式利用隔离级别。

如果 heuristics 代理的配置与用于实现实际隔离代理有相同的隔离级别，但在代理之前配置，隔离会在试图进行隔离前，在 heuristics 代理上发出一个 `off` 操作。如果 heuristics 代理给出了 `off` 操作的一个负结果，则代表隔离不成功，从而导致 Pacemaker 隔离跳过对实现隔离的代理发出 `off` 动作的步骤。heuristics 代理可以利用这个行为来防止实际上进行隔离的代理在特定情况下隔离节点。

用户可能希望使用这个代理，特别是在双节点集群中，如果节点可以预先知道无法正确接管该服务，则节点可以隔离这个代理。例如，如果节点在网络连接链接出现问题，使服务无法访问客户端，则节点接管服务可能不真实。在这种情况下，向路由器的 ping 可能会探测到这个情况。

(BZ#1775847)

5.5.5. 身份管理

身份管理 JSON-RPC API 作为技术预览

一个 API 可用于 Identity Management(IdM)。要查看 API，IdM 还提供了一个 API 浏览器作为技术预览。

在 Red Hat Enterprise Linux 7.3 中，IdM API 被改进来启用多个 API 命令版本。在以前的版本中，增强功能可能会以不兼容的方式改变命令的行为。用户现在可以继续使用已有的工具和脚本，即使 IdM API 发生了变化。这可启用：

- 管理员要在服务器中使用之前或更高版本的 IdM，而不是在管理客户端中使用。
- 开发人员使用 IdM 调用的特定版本，即使 IdM 版本在服务器上发生了变化。

在所有情况下，与服务器进行通信是可能的，无论是否一方使用，例如，一个新的版本会为这个功能引进新的选项。

有关使用 API 的详细信息，请参阅[使用身份管理 API 与 IdM 服务器通信\(TECHNOLOGY PREVIEW\)](#)。

(BZ#1664719)

DNSSEC 在 IdM 中作为技术预览提供

带有集成 DNS 的身份管理 (IdM) 服务器现在支持 DNS 安全扩展 (DNSSEC)，这是一组增强 DNS 协议安全性的 DNS 扩展。托管在 IdM 服务器上的 DNS 区可以使用 DNSSEC 自动签名。加密密钥是自动生成和轮转的。

建议那些决定使用 DNSSEC 保护 DNS 区的用户读取并遵循这些文档：

- 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>

请注意，集成了 DNSSEC 的 IdM 服务器验证从其他 DNS 服务器获取的 DNS 答案。这可能会影响未按照推荐的命名方法配置的 DNS 区域可用性。

(BZ#1664718)

现在，检查公钥基础架构的整体健康状况已作为技术预览提供

在这个版本中，公钥基础架构(PKI)Healthcheck 工具将 PKI 子系统的健康状况报告到 RHEL 8.1 中的身份管理(IdM)Healthcheck 工具。执行 IdM Healthcheck 会调用 PKI Healthcheck，它会收集并返回 PKI 子系统的健康状况报告。

所有部署的 RHEL IdM 服务器或副本都提供了 **pki-healthcheck** 工具。**pki-healthcheck** 提供的所有检查也集成到 **ipa-healthcheck** 工具中。**ipa-healthcheck** 可以独立于 **idm:DL1** 模块流安装。

请注意，**pki-healthcheck** 也可以在独立 Red Hat 证书系统(RHCS)基础架构中正常工作。

(BZ#1303254)

5.5.6. Desktop

GNOME 用于 64 位 ARM 架构，作为一个技术预览

GNOME 桌面环境现在可作为技术预览用于 64 位 ARM 架构。这可以让管理员使用 VNC 会话从图形用户界面(GUI)远程配置和管理服务器。

因此，在 64 位 ARM 架构中提供了新的管理应用程序。例如：**Disk Usage Analyzer (baobab)**，**防火墙配置 (firewall-config)**，**Red Hat Subscription Manager (subscription-manager)**，或 **Firefox web 浏览器**。使用 **Firefox**，管理员可以远程地连接到本地 Cockpit 守护进程。

(JIRA:RHELPLAN-27394, BZ#1667516, BZ#1667225, BZ#1724302)

5.5.7. 图形基础结构

VNC 远程控制台作为 64 位 ARM 架构的一个技术预览提供

在 64 位 ARM 架构中,虚拟网络计算(VNC)远程控制台可作为技术预览使用。请注意,在 64 位 ARM 架构中,目前图形堆栈的其它部分没有被验证。

(BZ#1698565)

5.5.8. Red Hat Enterprise Linux 系统角色

RHEL 系统角色的 postfix 角色作为技术预览

Red Hat Enterprise Linux 系统角色为 Red Hat Enterprise Linux 子系统提供了一个配置接口, 通过包含 Ansible 角色来简化系统配置。这个界面支持在多个 Red Hat Enterprise Linux 版本间管理系统配置, 并使用新的主发行版本。

rhel-system-roles 软件包通过 AppStream 软件仓库发布。

postfix 角色是作为技术预览提供的。

以下角色被完全支持：

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

如需更多信息, 请参阅有关 [RHEL 系统角色](#) 的知识库文章。

(BZ#1812552)

rhel-system-roles-sap 作为技术预览

rhel-system-roles-sap 软件包为 SAP 提供 Red Hat Enterprise Linux (RHEL)系统角色, 可用于自动化 RHEL 系统的配置以运行 SAP 工作负载。这些角色通过自动应用基于相关 SAP 备注中最佳实践的优化设置,大大减少了将系统配置为运行 SAP 工作负载的时间。访问只限制为 RHEL for SAP Solutions。如果需要帮助, 请联络红帽客户支持团队。

rhel-system-roles-sap 软件包中的以下新角色可作为技术预览提供：

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

如需更多信息, 请参阅 [SAP 的 Red Hat Enterprise Linux 系统角色](#)。

注：RHEL 8.2 for SAP Solutions 计划进行验证, 以便在 Intel 64 架构和 IBM POWER9 上与 SAP HANA 一起使用。对其他 SAP 应用和数据库产品（如 SAP NetWeaver 和 SAP ASE）的支持与 GA 版本绑定, 客户可在 GA 后使用 RHEL 8.2 功能。如需了解有关验证的版本和 SAP 支持的最新信息, 请查阅 SAP Notes 2369910 和 2235581。

(BZ#1660832)

5.5.9. 虚拟化

选择 Intel 网络适配器现在支持 Hyper-V 的 RHEL 客户端中的 SR-IOV

作为技术预览，运行在 Hyper-V hypervisor 上的 Red Hat Enterprise Linux 客户机操作系统现在可以将单根 I/O 虚拟化(SR-IOV)功能用于 **ixgbev** 和 **i40evf** 驱动程序支持的 Intel 网络适配器。此功能在满足以下条件时启用：

- 对网络接口控制器(NIC)启用了 SR-IOV 支持
- 对虚拟 NIC 启用了 SR-IOV 支持
- 对虚拟交换机启用 SR-IOV 支持
- NIC 中的虚拟功能(VF)附加到虚拟机。

目前，Microsoft Windows Server 2019 和 2016 支持该功能。

(BZ#1348508)

KVM 虚拟化可用于 RHEL 8 Hyper-V 虚拟机

作为技术预览，现在可将嵌套的 KVM 虚拟化用于 Microsoft Hyper-V hypervisor。因此，您可以在运行在 Hyper-V 主机的 RHEL 8 虚拟机中创建虚拟机。

请注意目前，这个功能仅适用于 Intel 系统。另外，在一些情况下，Hyper-V 中不默认启用嵌套虚拟化。要启用它，请参阅以下文档：

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

(BZ#1519039)

用于 KVM 虚拟机的 AMD SEV

作为技术预览，RHEL 8 为使用 KVM 管理程序的 AMD EPYC 主机引入了安全加密虚拟化(SEV)功能。如果在虚拟机(VM)上启用，SEV 会加密虚拟机内存，因此主机不能访问虚拟机上的数据。如果主机被恶意软件成功损坏，这可以提高虚拟机的安全性。

请注意，在单一主机上同时可以使用此功能的虚拟机数量是由主机硬件决定的。当前的 AMD EPYC 处理器支持使用 SEV 运行最多 509 个运行虚拟机。

也请注意，对于将 SEV 配置为可以引导的虚拟机，还必须使用硬内存限制配置虚拟机。要做到这一点，请在虚拟机 XML 配置中添加以下内容：

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

建议 N 的值等于或大于客户机 RAM + 256 MiB。例如：如果为客户端分配 2 GiB RAM，则 N 应该为 2359296 或更高。

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

Intel vGPU

作为技术预览，现在可以将物理 Intel GPU 设备划分为多个虚拟设备（称为 **mediated devices**）。然后可将这些 mediated devices 分配给多个虚拟机(VM)作为虚拟 GPU。因此,这些虚拟机共享单个物理 Intel GPU 的性能。

请注意,只有所选 Intel GPU 与 vGPU 功能兼容。另外,为虚拟机分配物理 GPU 使得主机无法使用 GPU,并可能阻止主机的图形显示输出工作。

(BZ#1528684)

5.5.10. 容器

Skopeo 容器镜像作为技术预览提供

registry.redhat.io/rhel8/skopeo 容器镜像是 **skopeo** 软件包的容器化实施。**skopeo** 是一个命令行工具，可以对容器镜像和镜像存储库执行各种操作。通过此容器镜像，您可以将容器镜像从一个未经身份验证的容器注册表检查并复制到另一个注册表。

(BZ#1627900)

Buildah 容器镜像作为技术预览提供

registry.redhat.io/rhel8/buildah 容器镜像是 **buildah** 软件包的容器化实施。**buildah** 是一个有助于构建 OCI 容器镜像的工具。此容器镜像允许您构建容器镜像，而无需在系统中安装 **buildah** 软件包。使用案例不包括在无根模式下作为非 root 用户运行此镜像。

(BZ#1627898)

podman-machine 命令不受支持

用于管理虚拟机的 **podman-machine** 命令仅作为技术预览提供。相反，请直接从命令行运行 Podman。

(JIRA:RHELDPCS-16861)

5.6. 过时的功能

这部分提供了在 Red Hat Enterprise Linux 8.2 中弃用的功能概述。

弃用的功能可能在以后的主要发行本中不被支持，因此不建议在新的部署中使用。有关特定主要发行本中已弃用功能的最新列表，请参考最新版本的发行文档。

在 Red Hat Enterprise Linux 8 中，已弃用的功能的支持状态不会改变。有关支持长度的详情，请查看 [Red Hat Enterprise Linux 生命周期](#)和 [Red Hat Enterprise Linux 应用程序流生命周期](#)。

对于当前或将来的主发行版本中的新部署，我们不推荐使用已弃用的硬件组件。硬件驱动程序更新仅限于安全和关键修复。红帽建议尽快替换这个硬件。

一个软件包可以被弃用，我们不推荐在以后使用。在某些情况下，软件包可从产品中删除。然后，产品文档可识别提供类似、完全相同或者更高级功能的最新软件包，并提供进一步建议。

有关 RHEL 7 中存在，但已从 RHEL 8 中删除的功能信息，请参阅 [使用 RHEL 8 的注意事项](#)。

5.6.1. 安装程序和镜像创建

弃用了一些 Kickstart 命令和选项

使用 RHEL 8 Kickstart 文件中的以下命令和选项会在日志中显示警告信息。

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

如果只列出具体的选项，则基础命令及其它选项仍可用且没有弃用。

有关 Kickstart 中的详情和相关更改，请参阅 *使用 RHEL 8 的参考事项* 文档中的 [Kickstart 更改](#) 部分。

(BZ#1642765)

ignoredisk Kickstart 命令的 --interactive 选项已被弃用

在以后的 Red Hat Enterprise Linux 版本中使用 **--interactive** 选项会导致严重安装错误。建议您修改 Kickstart 文件删除该选项。

(BZ#1637872)

5.6.2. 软件管理

rpmbuild --sign 已弃用

在这个版本中，**rpmbuild --sign** 命令已过时。在以后的 Red Hat Enterprise Linux 版本中使用这个命令可能会导致错误。建议您使用 **rpmsign** 命令替代。

(BZ#1688849)

5.6.3. Shell 和命令行工具

禁用对 curl 的 metalink 支持

在 curl 功能中发现了一个缺陷，它在处理使用 Metalink 下载的内容时处理凭证和文件哈希不匹配的方式存在问题。此漏洞允许控制托管服务器的恶意执行器：

- 欺骗用户下载恶意内容

- 在用户不知情的情况下获取对提供的凭证的未授权的访问

来自此漏洞的最大威胁是机密性和完整性。为避免这种情况，在 Red Hat Enterprise Linux 8.2.0.z 中禁用了对 curl 的 Metalink 支持。

作为临时解决方案，在下载 Metalink 文件后执行以下命令：

```
wget --trust-server-names --input-metalink`
```

例如：

```
wget --trust-server-names --input-metalink <(curl -s $URL)
```

(BZ#1999620)

5.6.4. 安全性

NSS SEED 密码已弃用

Mozilla Network Security Services (**NSS**) 库将不支持在以后的版本中使用 SEED 密码的 TLS 密码组合。对于要依赖 SEED 加密的部署，红帽建议启用对其它密码套件的支持。这样，您可以确保在 NSS 不再支持时可以平稳过渡。

请注意，在 RHEL 中，SEED 加密已被默认禁用。

(BZ#1817533)

TLS 1.0 和 TLS 1.1 已弃用

TLS 1.0 和 TLS 1.1 协议在 **DEFAULT** 系统范围的加密策略级别被禁用。如果需要使用启用的协议，如 Firefox 网页浏览器中的视频检查程序，把系统范围的加密策略切换到 **LEGACY** 级别：

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

如需更多信息，请参阅 [RHEL 8 中的 Strong crypto 默认值](#)，并弃用红帽客户门户网站中的弱加密算法 知识库文章和 [update-crypto-policies\(8\)man page](#)。

(BZ#1660839)

在 RHEL 8 中弃用 DSA

数字签名算法(DSA)在 Red Hat Enterprise Linux 8 中被视为已弃用。依赖于 DSA 密钥的身份验证机制在默认配置中不起作用。请注意，即使使用系统范围的 **LEGACY** 加密策略级别中，**OpenSSH** 客户端都不接受 DSA 主机密钥。

(BZ#1646541)

在 NSS 中弃用了 SSL2 Client Hello

传输层安全性(TLS)协议版本 1.2 和更早版本允许以向后兼容安全套接字层(SSL)协议版本 2 的方式与客户端 **Hello** 消息开始协商。网络安全服务(NSS)库中对这个功能的支持已被弃用，默认是禁用的。

需要这个功能支持的应用程序需要使用新的 **SSL_ENABLE_V2_COMPATIBLE_HELLO** API 启用它。以后的 Red Hat Enterprise Linux 8 版本中可以完全删除对这个功能的支持。

(BZ#1645153)

TPM 1.2 已被弃用

Trusted Platform Module (TPM) 安全加密处理器标准版本在 2016 年更新至 2.0 版本。TPM 2.0 比 TPM 1.2 提供了很多改进，它和之前的版本不向后兼容。在 RHEL 8 中弃用了 TPM 1.2，它可能会在下一个主发行版本中删除。

(BZ#1657927)

5.6.5. 网络

在 RHEL 8 中已弃用网络脚本

网络脚本在 Red Hat Enterprise Linux 8 中已弃用，且不再默认提供。基本安装提供了 **ifup** 和 **ifdown** 脚本的新版本，它们通过 **nmcli** 工具调用 **NetworkManager** 服务。在 Red Hat Enterprise Linux 8 中，要运行 **ifup** 和 **ifdown** 脚本，**NetworkManager** 必须正在运行。

请注意，**/sbin/ifup-local**、**ifdown-pre-local** 和 **ifdown-local** 脚本中的自定义命令不会执行。

如果需要这些脚本，您仍可以使用以下命令在系统中安装已弃用的网络脚本：

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

ifup 和 **ifdown** 脚本链接到已安装的旧网络脚本。

调用旧的网络脚本会显示一个关于它们已过时的警告。

(BZ#1647725)

5.6.6. 内核

使用无磁盘引导为 Real Time 8 安装 RHEL 现已弃用

无磁盘引导允许多个系统通过网络共享一个 root 文件系统。无盘引导是方便的，它在实时工作负载中容易引入网络延迟。在以后的 RHEL for Real Time 8 的更新中，无盘引导功能将不再被支持。

(BZ#1748980)

qla3xxx 驱动程序已弃用

在 RHEL 8 中已弃用 **qla3xxx** 驱动程序。本产品的将来主发行版本可能不支持该驱动程序，因此不建议在新的部署中使用该驱动程序。

(BZ#1658840)

dl2k、dnet、ethoc 和 dlci 驱动程序已弃用

在 RHEL 8 中弃用了 **dl2k**、**dnet**、**ethoc** 和 **dlci** 驱动程序。在以后的主要发行本中，这些驱动程序可能不被支持，因此不建议在新的部署中使用这些驱动程序。

(BZ#1660627)

The rdma_rxe Soft-RoCE 驱动程序已弃用

软件直接内存通过融合以太网(Soft-RoCE) (也称为 RXE) 是模拟远程直接内存访问(RDMA)的功能。在 RHEL 8 中，Soft-RoCE 功能作为一个不受支持的技术预览提供。但是，由于稳定性问题，此功能已被弃用，并将在 RHEL 9 中删除。

(BZ#1878207)

5.6.7. 文件系统和存储

elevator 内核命令行参数已弃用

在之前的 RHEL 版本中使用 **elevator** 内核命令行参数为所有设备设置磁盘调度程序。在 RHEL 8 中，该参数已弃用。

上游 Linux 内核删除了对 **elevator** 参数的支持，但出于兼容性的原因，在 RHEL 8 中仍提供此支持。

请注意，内核会根据设备类型选择默认磁盘调度程序。这通常是最佳设置。如果您需要不同的调度程序，红帽建议您使用 **udev** 规则或 Tuned 服务来配置它。匹配所选设备并只为那些设备切换调度程序。

如需更多信息，请参阅[设置磁盘调度程序](#)。

(BZ#1665295)

LVM mirror 已弃用

LVM **mirror** segment 类型已弃用。以后的 RHEL 主发行版本中会删除对 **mirror** 的支持。

红帽建议使用 segment 类型为 **raid1** 的 LVM RAID 1 设备来替代 **mirror**。**raid1** segment 类型是默认的 RAID 配置类型，它作为推荐的解决方案替换 **mirror**。

要将 **mirror** 设备转换为 **raid1**，请参阅[将镜像 LVM 设备转换为 RAID1 逻辑卷](#)。

LVM **mirror** 有几个已知问题。详情请查看[文件系统和存储中的已知问题](#)。

(BZ#1827628)

禁用了 NFSv3 over UDP

默认情况下，NFS 服务器不再默认在 User Datagram Protocol(UDP)套接字上打开或监听。这个变化只影响 NFS 版本 3，因为版本 4 需要传输控制协议(TCP)。

RHEL 8 不再支持通过 UDP 的 NFS。

(BZ#1592011)

5.6.8. 身份管理

SMB1 协议在 Samba 中被弃用

从 Samba 4.11 开始，不安全的服务器消息块版本 1 (SMB1) 协议已弃用，并将在以后的发行版本中删除。

为提高安全性，在 Samba 服务器和客户端工具中默认禁用 SMB1。

(JIRA:RHELDPCS-16612)

5.6.9. Desktop

libgnome-keyring 库已弃用

libgnome-keyring 库已弃用，现在使用 **libsecret** 库，因为 **libgnome-keyring** 没有被上游维护，且不会遵循 RHEL 所需的加密策略。新的 **libsecret** 库是符合所需安全标准的替换。

(BZ#1607766)

5.6.10. 图形基础结构

不再支持 AGP 图形卡

Red Hat Enterprise Linux 8 不支持使用图形端口(AGP)总线的图形卡。推荐使用 PCI-Express bus 图形卡替换。

(BZ#1569610)

5.6.11. Web 控制台

Web 控制台不再支持不完整翻译

RHEL web 控制台不再提供翻译少于 50% 的语言支持。如果浏览器要求转换成这种语言，用户界面将为英语。

(BZ#1666722)

5.6.12. 虚拟化

virt-manager 已被弃用

虚拟机管理器（也称 **virt-manager**）已弃用。RHEL 8 web 控制台（也称 **Cockpit**）旨在在以后的版本中成为它替换。因此，建议您使用 web 控制台使用 GUI 管理虚拟化。但请注意，**virt-manager** 中的一些功能可能还不能使用 RHEL 8 web 控制台。

(JIRA:RHELPLAN-10304)

RHEL 8 不支持虚拟机快照

当前创建虚拟机(VM)快照的机制已经被弃用，因为它无法可靠工作。因此，建议在 RHEL 8 中使用虚拟机快照。

请注意，一个新的 VM 快照机制正在开发中，并将在以后的 RHEL 8 次要发行本中完全实现。

(BZ#1686057)

Cirrus VGA 虚拟 GPU 类型已弃用

随着 Red Hat Enterprise Linux 的主要更新，**Cirrus VGA** GPU 设备将在 KVM 虚拟机中不再被支持。因此，红帽建议您使用 **stdvga**、**virtio-vga** 或者 **qxl** 设备而不是 Cirrus VGA。

(BZ#1651994)

已弃用并删除 **cpu64-rhel6** CPU 模型

cpu64-rhel6 QEMU 虚拟 CPU 模型已在 RHEL 8.1 中弃用，已从 RHEL 8.2 中删除。建议您根据主机计算机上存在的 CPU，使用 QEMU 和 libvirt 提供的其他 CPU 型号。

(BZ#1741346)

5.6.13. 已弃用的软件包

下列软件包已弃用，可能不会包括在 Red Hat Enterprise Linux 未来的主发行版本中：

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

5.7. 已知问题

这部分论述了 Red Hat Enterprise Linux 8.2 中已知的问题。

5.7.1. 安装程序和镜像创建

auth 和 authconfig Kickstart 命令需要 AppStream 软件仓库

auth 和 **authconfig** Kickstart 命令在安装过程中需要 **authselect-compat** 软件包。如果没有这个软件包，如果使用了 **auth** 或 **authconfig**，则安装会失败。但根据设计，**authselect-compat** 软件包只包括在 AppStream 仓库中。

要临时解决这个问题，请确定安装程序可使用 BaseOS 和 AppStream 软件仓库，或者在安装过程中使用 **authselect** Kickstart 命令。

(BZ#1640697)

reboot --kexec 和 inst.kexec 命令不提供可预测的系统状态

使用 **reboot --kexec** Kickstart 命令或 **inst.kexec** 内核引导参数执行 RHEL 安装不会提供与完全重启相同的可预期系统状态。因此，在不重启的情况下切换安装的系统可能会导致无法预计的结果。

请注意，**kexec** 功能已弃用，并将在以后的 Red Hat Enterprise Linux 版本中删除。

(BZ#1697896)

Anaconda 安装包括最小资源设置要求的低限制

Anaconda 以最少的资源设置在系统中启动安装，并且不要提供有关成功执行安装所需的资源的先前消息警告。因此，安装可能会失败，输出错误不会为可能的调试和恢复提供清晰的信息。要临时解决这个问题，请确保系统具有安装所需的最少资源设置：2GB 内存在 PPC64(LE)和 1GB on x86_64 上。因此，应该可以成功执行安装。

(BZ#1696609)

使用 `reboot --kexec` 命令安装失败

当使用包含 `reboot --kexec` 命令的 Kickstart 文件时，RHEL 8 安装会失败。为避免问题，请在 Kickstart 文件中使用 `reboot` 命令而不是 `reboot --kexec`。

(BZ#1672405)

RHEL 8 初始设置无法通过 SSH 执行

目前，当使用 SSH 登录到系统时，RHEL 8 初始设置接口不会显示。因此，无法在通过 SSH 管理的 RHEL 8 机器上执行初始设置。要临时解决这个问题，请在主系统控制台(ttySO)中执行初始设置，然后再使用 SSH 登录。

(BZ#1676439)

在安装程序中不默认启用网络访问

几个安装功能需要网络访问，例如：使用 Content Delivery Network(CDN)、NTP 服务器支持和网络安装源注册系统。但默认情况下不启用网络访问，因此在启用网络访问前无法使用这些功能。

要临时解决这个问题，请添加 `ip=dhcp` 在启动安装时启用网络访问。另外，使用引导选项传递 Kickstart 文件或位于网络中的库也会解决这个问题。因此可以使用基于网络的安装功能。

(BZ#1757877)

属于多个机构的用户帐户注册失败

目前，当试图使用属于多个机构的用户帐户注册系统时，注册过程会失败并显示出错信息，**You must specify an organization for new units.**

要临时解决这个问题，您可以：

- 使用不属于多个机构的不同用户帐户。
- 使用 GUI 和 Kickstart 安装的的 Connect to Red Hat 中的 **Activation Key** 验证方法。
- 跳过连接到红帽的注册步骤，并使用 Subscription Manager 在安装后注册您的系统。

(BZ#1822880)

在没有 CDN 注册的情况下，有时无法进行使用 Binary DVD ISO 镜像的 GUI 安装

当使用 Binary DVD ISO 镜像文件执行 GUI 安装时，安装程序的竞争条件有时可能会阻止安装进行，直到您使用连接到红帽功能注册系统。要临时解决这个问题，请完成以下步骤：

1. 在 GUI 安装的 **安装概述** 窗口中选择 **安装源**。
2. 验证是否选择了 **自动探测的安装介质**。
3. 点击 **完成** 确认选择并返回 **安装概述** 窗口。
4. 验证 **Local Media** 在 **Installation Summary** 窗口中是否显示为 **Installation Source** 状态。

因此，您可以继续安装，而不必使用连接到红帽的功能注册系统。

(BZ#1823578)

将 Binary DVD.iso 文件的内容复制到分区会省略 .treeinfo 和 .discinfo 文件

在本地安装过程中，当将 RHEL 8 Binary DVD.iso 镜像文件的内容复制到分区时，`cp <path>/^* <mounted 分区>/dir` 命令中的 `*` 无法复制 `.treeinfo` 和 `.discinfo` 文件。成功安装时需要这些文件。因此，BaseOS 和 AppStream 软件仓库不会被加载，在 `anaconda.log` 文件中与 `debug` 相关的日志消息是问题的唯一记录。

要临时解决这个问题，将 `missing .treeinfo` 和 `.discinfo` 文件复制到分区中。

(BZ#1687747)

Kickstart 安装无法使用自签名 HTTPS 服务器

目前，当在 kickstart 文件中指定安装源并使用 `--noverifyssl` 选项时，安装程序无法从自签名的 https 服务器安装：

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

要临时解决这个问题，请在开始 kickstart 安装时将 `inst.noverifyssl` 参数附加到内核命令行中。

例如：

```
inst.ks=<URL> inst.noverifyssl
```

(BZ#1745064)

如果在仓库刷新完成前尝试使用 CDN 取消注册，则 GUI 安装可能会失败

在 RHEL 8.2 中，当使用 Content Delivery Network(CDN)注册您的系统并附加订阅时，GUI 安装程序会启动对仓库元数据的刷新。刷新过程不是注册和订阅过程的一部分，因此在 **Connect to Red Hat** 窗口中启用了 **Unregister** 按钮。根据网络连接，刷新过程可能需要一分钟以上的时间完成。如果您在刷新过程完成前点 **Unregister** 按钮，则 GUI 安装可能会失败，因为未注册过程会删除 CDN 仓库文件和安装程序与 CDN 通信所需的证书。

要临时解决这个问题，点 **连接到红帽** 窗口中的 **Register** 按钮后在 GUI 安装中完成以下步骤：

1. 在 **连接到红帽** 的窗口中点 **完成** 返回 **安装概述** 窗口。
2. 在 **安装概述** 窗口中验证 **安装源** 和 **软件选择状态** 信息是否以斜体显示任何处理信息。
3. 当安装源和软件选择类别准备好后，点 **连接到红帽**。
4. 点 **Unregister** 按钮。

执行这些步骤后，您可以在 GUI 安装过程中安全地取消注册系统。

(BZ#1821192)

5.7.2. 订阅管理

`syspurpose addons` 对 `subscription-manager attach --auto` 输出没有影响。

在 Red Hat Enterprise Linux 8 中，添加了 `syspurpose` 命令行工具的四个属性：`role`、`usage`、`service_level_agreement` 和 `addons`。目前，只有 `role`、`usage` 和 `service_level_agreement` 会影响到运行 `subscription-manager attach --auto` 命令的输出。试图为 `addons` 参数设置值的用户不会观察到对自动附加的订阅有任何影响。

(BZ#1687900)

使用 Kickstart 文件安装 RHEL 时，多路径存储设备中的数据会丢失

使用 Kickstart 文件安装 RHEL 时，附加到主机的多路径存储设备中的数据会丢失。出现这个问题的原因是，安装程序无法忽略您使用 `ignoredisk --drives` 命令指定的多路径存储设备。因此，设备中的数据会丢失。

要临时解决这个问题，请在安装前分离设备，或使用 `ignoredisk --only-use` 命令指定设备进行安装。

(BZ#1862131)

5.7.3. Shell 和命令行工具

使用 Wayland 协议的应用无法转发到远程显示服务器

在 Red Hat Enterprise Linux 8 中，大多数应用程序默认使用 Wayland 协议，而不是 X11 协议。因此，ssh 服务器无法转发使用 Wayland 协议的应用，但能够将使用 X11 协议的应用转发到远程显示服务器。

要临时解决这个问题，请在启动应用程序前设置环境变量 `GDK_BACKEND=x11`。因此，可以将应用转发到远程显示服务器。

(BZ#1686892)

systemd-resolved.service 无法在引导时启动

`systemd-resolved` 服务偶尔无法在引导时启动。如果发生这种情况，请在引导完成后手动重启该服务：

```
# systemctl start systemd-resolved
```

但是，在引导时解析 `systemd` 失败不会影响任何其他服务。

(BZ#1640802)

5.7.4. 安全性

可执行审核监控在符号链接上无法正常工作

由 `-w` 选项提供的文件监控无法直接跟踪路径。它需要解析到设备的路径以及内节点，才能与已执行程序进行比较。监控可执行符号链接的观察监控将监控设备和符号链接本身的索引节点，而不是内存中执行的程序，这些程序可从符号链接的解析中找到。即使监视解析了符号链接以获取生成的可执行程序，规则也会对从不同符号链接调用的多调用二进制文件触发。这会导致大量日志带有假的正状态。因此，可执行的审计监控符号链接无法正常工作。

要临时解决这个问题，请为可执行程序的解析路径设置一个监视，并使用 `comm=` 或 `proctitle=` 字段中列出的最后一个组件过滤生成的日志消息。

(BZ#1846345)

/etc/selinux/config 中的 SELINUX=disabled 无法正常工作

在 `/etc/selinux/config` 中使用 `SELINUX=disabled` 选项禁用 SELinux 会导致内核在启用了 SELinux 的情况下引导，并在稍后的引导过程中切换到禁用模式。这可能导致内存泄漏。

要临时解决这个问题，请在内核命令行中添加 `selinux=0` 参数来禁用 SELinux，如 [使用 SELinux 中的在引导时更改 SELinux 模式](#) 部分所述。

(JIRA:RHELPLAN-34199)

libselinux-python 只能通过其模块提供

libselinux-python 软件包只包含用于开发 SELinux 应用程序的 Python 2 绑定，它用于向后兼容。因此，通过 `dnf install libselinux-python` 命令，默认的 RHEL 8 软件仓库不再提供 **libselinux-python**。

要临时解决这个问题，请启用 **libselinux-python** 和 **python27** 模块，并使用以下命令安装 **libselinux-python** 软件包及其相依性软件包：

```
# dnf module enable libselinux-python
# dnf install libselinux-python
```

或者，使用它的安装配置集在一个命令中安装 **libselinux-python**:

```
# dnf module install libselinux-python:2.8/common
```

因此，您可以使用相关的模块安装 **libselinux-python**。

(BZ#1666328)

UDICA 仅在使用 `--env container=podman` 启动时才会处理 UBI 8 容器

Red Hat Universal Base Image 8(UBI 8)容器将 **container** 环境变量设置为 **oci** 值，而不是 **podman** 值。这可以防止 **udica** 工具分析容器 JavaScript 对象表示法(JSON)文件。

要临时解决这个问题，请使用带有 `--env container=podman` 参数的 **podman** 命令启动 UBI 8 容器。因此，只有使用上述临时解决方案时，**udica** 才可以为 UBI 8 容器生成 SELinux 策略。

(BZ#1763210)

删除 rpm-plugin-selinux 软件包会导致从系统中删除所有 selinux-policy 软件包

删除 **rpm-plugin-selinux** 软件包会禁用机器中的 SELinux。它还会从系统中删除所有 **selinux-policy** 软件包。重复安装 **rpm-plugin-selinux** 软件包后会安装 **selinux-policy-minimum** SELinux 策略，即使之前系统中存在 **selinux-policy-targeted** 策略。但是，重复安装不会更新 SELinux 配置文件来考虑策略的改变。因此，即使重新安装 **rpm-plugin-selinux** 软件包也会禁用 SELinux。

要临时解决这个问题：

1. 输入 `umount /sys/fs/selinux/` 命令。
2. 手动安装缺少的 **selinux-policy-targeted** 软件包。
3. 编辑 `/etc/selinux/config` 文件以便策略等同于 **SELINUX=enforcing**。
4. 输入命令 `load_policy -i`。

因此，SELinux 被启用并运行和以前相同的策略。

(BZ#1641631)

SELinux 会阻止 **systemd-journal-gatewayd** 在由 **corosync** 创建的共享内存文件中调用 `newfstatat ()`

SELinux 策略不包含允许 **systemd-journal-gatewayd** 守护进程访问由 **corosync** 服务创建的文件的规定。因此，SELinux 拒绝 **systemd-journal-gatewayd** 在由 **corosync** 创建的共享内存文件中调用 **newfstatat ()** 功能。

要临时解决这个问题，请使用启用上述场景的 allow 规则创建一个本地策略模块。有关生成 SELinux 策略 *允许* 和 *dontaudit 规则* 的更多信息，请参阅 **audit2allow(1)** man page。由于前面的临时解决方案，**systemd-journal-gatewayd** 可以在 enforcing 模式中使用 SELinux 的 **corosync** 创建的共享内存文件上调用该功能。

(BZ#1746398)

SELinux 会阻止 auditd 停止或关闭系统

SELinux 策略不包含允许 Audit 守护进程启动 **power_unit_file_t systemd** 单元的规则。因此，在日志记录磁盘分区没有剩余空间的情况下，**audit d** 也无法停止或关闭系统。

要临时解决这个问题，请创建自定义 SELinux 策略模块。因此，只有应用了临时解决方案，**audit d** 才可以正确地停止或关闭系统。

(BZ#1826788)

锁定用户可以运行 sudo 命令

在使用 **ALL** 关键字定义 **sudoers** 权限的系统中，拥有 **权限的 sudo** 用户可以作为帐户被锁定的用户运行 **sudo** 命令。因此，仍然可以使用锁定的和过期的帐号来执行命令。

要临时解决这个问题，请启用新实现的 **runas_check_shell** 选项，并在 **/etc/shells** 中正确设置有效 shell。这样可防止攻击者在比如 **bin** 的系统帐户中运行命令。

(BZ#1786990)

默认日志设置在性能上的负面影响

默认日志环境设置可能会消耗 4 GB 内存甚至更多，当 **systemd-journald** 使用 **rsyslog** 运行时，速率限制值的调整会很复杂。

如需更多信息，请参阅 [RHEL 默认日志设置对性能的负面影响及环境方案](#)。

(JIRA:RHELPLAN-10431)

使用 config.enabled 的 rsyslog 输出中的 Parameter not known 错误

在 **rsyslog** 输出中，使用 **config.enabled** 指令在配置处理错误中出现意外错误。因此，在使用 **config.enabled** 指令时会显示 **参数未知的错误**，但 **include ()** 语句除外。

要临时解决这个问题，请设置 **config.enabled=on** 或者使用 **include ()** 语句。

(BZ#1659383)

某些 rsyslog 优先级字符串无法正常工作

对允许精细控制加密的 **imtcp** 的 **GnuTLS** 优先级字符串的支持并不完整。因此，以下优先级字符串无法在 **rsyslog** 中正常工作：

```
NONE:+VERS-ALL:-VERS-TLS1.3:+MAC-ALL:+DHE-RSA:+AES-256-GCM:+SIGN-RSA-SHA384:+COMP-ALL:+GROUP-ALL
```

要临时解决这个问题，请只使用正确的优先级字符串：


```
NONE:+VERS-ALL:-VERS-TLS1.3:+MAC-ALL:+ECDHE-RSA:+AES-128-CBC:+SIGN-RSA-
SHA1:+COMP-ALL:+GROUP-ALL
```

因此，当前的配置必须仅限于可正常工作的字符串。

([BZ#1679512](#))

到带有 SHA-1 签名的服务器的连接无法使用 GnuTLS

GnuTLS 安全通讯库以 `insecure` 形式拒绝 SHA-1 证书签名。因此，使用 GnuTLS 作为 TLS 后端的应用程序无法建立与提供此类证书的对等的 TLS 连接。这个行为与其他系统加密库不一致。要临时解决这个问题，请升级服务器以使用 SHA-256 或更强大的哈希签名的证书，或切换到 LEGACY 策略。

([BZ#1628553](#))

在 FIPS 模式下，TLS 1.3 无法在 NSS 中工作

在使用 FIPS 模式的系统中不支持 TLS 1.3。因此，需要 TLS 1.3 进行互操作性的连接无法在 FIPS 模式下工作的系统上正常工作。

要启用连接，请禁用系统的 FIPS 模式，或者启用对 `peer` 中 TLS 1.2 的支持。

([BZ#1724250](#))

OpenSSL 错误处理 PKCS #11 tokens 不支持原始 RSA 或 RSA-PSS 签名

OpenSSL 库不会检测到 PKCS #11 令牌的与键相关的功能。因此，当使用不支持原始 RSA 或 RSA-PSS 签名的令牌创建签名时，建立 TLS 连接会失败。

要临时解决这个问题，请在 `/etc/pki/tls/openssl.cnf` 文件的 `crypto_policy` 部分的 `.include` 行后面添加以下行：

```
SignatureAlgorithms =
RSA+SHA256:RSA+SHA512:RSA+SHA384:ECDSA+SHA256:ECDSA+SHA512:ECDSA+SHA384
MaxProtocol = TLSv1.2
```

因此，可以在描述的场景中建立 TLS 连接。

([BZ#1685470](#))

OpenSSL 在 TLS 1.3 中的 CertificateRequest 消息中生成一个不正确的 status_request 扩展

如果启用了 `status_request` 扩展和基于客户端证书的身份验证，OpenSSL 服务器会在 `CertificateRequest` 消息中发送错误的 `status_request` 扩展。在这种情况下，OpenSSL 无法与 RFC 8446 协议兼容的实施互操作。因此，正确验证 `CertificateRequest` 消息中的扩展的客户端会中止与 OpenSSL 服务器的连接。要临时解决这个问题，在连接两侧禁用对 TLS 1.3 协议的支持，或者禁用对 OpenSSL 服务器上的 `status_request` 的支持。这将阻止服务器发送错误的消息。

([BZ#1749068](#))

ssh-keyscan 无法在 FIPS 模式中检索服务器的 RSA 密钥

在 FIPS 模式中的 RSA 签名禁用了 `SHA-1` 算法，这样可防止 `ssh-keyscan` 工具程序获取在那个模式下运行的服务器的 RSA 密钥。

要临时解决这个问题，使用 ECDSA 密钥，或者使用服务器中的 `/etc/ssh/ssh_host_rsa_key.pub` 文件在本地检索密钥。

(BZ#1744108)

Libreswan 在所有配置中使用 `seccomp=enabled` 无法正常工作

Libreswan SECCOMP 支持实施中的允许系统调用集合目前还不完整。因此，当 `ipsec.conf` 文件中启用 SECCOMP 时，`syscall` 过滤也会拒绝 `pluto` 守护进程正常工作所需的 `syscalls`；守护进程已被终止，`ipsec` 服务重启。

要临时解决这个问题，将 `seccomp=` 选项设置回 `disabled` 状态。为了正确运行 `ipsec`，SECCOMP 支持必须保持禁用状态。

(BZ#1777474)

某些 SSG 中的规则组可能会失败

由于规则及其依赖项未定义，在基准中修复 **SCAP 安全指南** (SSG) 规则可能会失败。如果需要以特定顺序执行两个或多个规则，例如，当一条规则安装组件和另一个规则配置同一组件时，它们可按错误的顺序运行，并报告错误。要临时解决这个问题，请执行补救两次，第二次运行会修复依赖规则。

(BZ#1750755)

SCAP Workbench 无法从定制的配置集生成基于结果的补救方法

当尝试使用 SCAP Workbench 工具从自定义配置集生成基于结果的补救角色时，会出现以下错误：

```
Error generating remediation role .../remediation.sh: Exit code of oscap was 1: [output truncated]
```

要临时解决这个问题，请使用带有 `--tailoring-file` 选项的 `oscap` 命令。

(BZ#1640715)

Kickstart 在 RHEL 8 中使用 `org_fedora_oscap` 而不是 `com_redhat_oscap`

Kickstart 将 Open Security Content Automation Protocol (OSCAP) Anaconda 附加组件作为 `org_fedora_oscap` 而不是 `com_redhat_oscap` 来引用，这可能会导致混淆。这样做可以保持与 Red Hat Enterprise Linux 7 的向后兼容性。

(BZ#1665082)

OSCAP Anaconda Addon 不会在文本模式中安装所有软件包

如果安装以文本模式运行，则 **OSCAP Anaconda Addon** 插件无法修改为系统安装程序安装而选择的软件包列表。因此，当使用 Kickstart 指定安全策略配置集且安装以文本模式运行时，安全策略所需的附加软件包不会在安装过程中安装。

要临时解决这个问题，可以使用图形模式运行安装，或者在 Kickstart 文件的 `%packages` 部分指定安全策略配置集所需的所有软件包。

因此，在没有描述的一个临时解决方案的情况下，安全策略配置集所需的软件包不会在 RHEL 安装过程中安装，且安装的系统与给定的安全策略配置集不兼容。

(BZ#1674001)

OSCAP Anaconda Addon 组件无法正确处理自定义配置集

OSCAP Anaconda Addon 插件无法以独立文件中自定义的方式正确处理安全配置集。因此，即使您在对应的 Kickstart 部分正确指定了自定义配置集，RHEL 图形安装中也不会提供自定义配置集。

要临时解决这个问题，请遵循 [从原始 DS 创建单一 SCAP 数据流中的说明](#)，以及 [一个定制文件](#) 知识库文章。因此，您可以在 RHEL 图形安装中使用自定义的 SCAP 配置集。

(BZ#1691305)

gnutls 无法恢复使用 NSS 服务器的当前会话

当恢复 TLS（传输层安全）1.3 会话时，**GnuTLS** 客户端会等待 60 毫秒，加上服务器发送会话恢复数据的预计往返时间。如果服务器没有在此时间内发送恢复数据，客户端会创建一个新的会话，而不是恢复当前会话。这不会造成严重的负面影响，只会对常规会话协商的性能有小的影响。

(BZ#1677754)

当使用 --sudo 扫描远程系统时，oscap-ssh 工具会失败

在使用带 `--sudo` 选项的 **oscap-ssh** 工具对远程系统执行安全内容自动化协议(SCAP)扫描时，远程系统上的 **oscap** 工具将扫描结果文件保存并报告到一个临时目录中，以 **root** 用户身份将文件报告到一个临时目录中。如果远程计算机上的 **umask** 设置已被更改，**oscap-ssh** 可能无法访问这些文件。要临时解决这个问题，请修改 **oscap-ssh** 工具，如本解决方案 ["oscap-ssh --sudo" 错误所述](#)，检索 ["scp: ...: Permission denied" 错误的结果文件](#)。因此，**oscap** 会以目标用户身份保存文件，**oscap-ssh** 会正常访问这些文件。

(BZ#1803116)

OpenSCAP 会在从 YAML 多行字符串中删除空白行时生成假的正数

当 OpenSCAP 从 `datastream` 中生成 Ansible 修复时，它会从 YAML 多行字符串中删除空白行。由于某些 Ansible 修复包含字面配置文件内容，因此删除空行会影响对应的修复。这会导致 **openscap** 实用程序无法进行相应的 Open Vulnerability 和评估语言(OVAL)检查，即使空白行没有任何作用。要临时解决这个问题，请检查规则描述并跳过因为缺少空白行而失败的扫描结果。或者，使用 Bash 修复而不是 Ansible 修复，因为 Bash 修复不会生成这些假的正结果。

(BZ#1795563)

基于 OSPP 的配置集与 GUI 软件包组不兼容。

服务器安装的 GUI 软件包组的 **GNOME** 软件包需要 **nfs-utils** 软件包，该软件包与操作系统保护配置集 (OSPP) 不兼容。因此，在安装带有 OSPP 或 OSPP 配置集的系统（例如安全技术实施指南(STIG)）的过程中，选择 *Server with GUI* 软件包组。如果在安装后应用了基于 OSPP 的配置集，则系统将无法引导。要临时解决这个问题，请不要在使用 OSPP 配置集和基于 OSPP 的配置集时使用 *Server with GUI* 或者其它安装 GUI 的组。在使用 *Server* 或 *Minimal Install* 软件包组时，系统不会有任何问题并可正常工作。

(BZ#1787156)

无法使用 e8 配置集修复 Server with Server with GUI 软件包组的 RHEL8 系统

使用 OpenSCAP Anaconda 附加组件强化 *Server with GUI* 软件包组上的系统，带配置文件（从 *Verify Integrity with RPM* 组中选择规则）需要系统中的超大 RAM。这个问题是由 OpenSCAP 扫描程序造成的。如需更多详情，请参阅 [扫描大量使用 OpenSCAP 的文件，从而导致系统运行内存不足](#)。因此，使用 RHEL8 Essential Eight(e8)配置集强化系统将无法成功。要临时解决这个问题，请选择一个较小的软件包组，如 *Server*，并在安装后安装您需要的附加软件包。因此，该系统会使用较少的软件包，扫描需要较少的内存，因此可自动强化该系统。

(BZ#1816199)

使用 OpenSCAP 扫描大量文件会导致系统内存不足

OpenSCAP 扫描程序将所有收集的结果存储在内存中，直到扫描完成为止。因此，系统在扫描大量文件时，比如从大软件包组 *Server with GUI* 和 *Workstation* 中进行扫描时，系统可能会出现系统没有足够的问题。要临时解决这个问题，在内存较小的系统中，使用较小的软件包组，如 *Server* 和 *Minimal Install*。

如果您需要使用大的软件包组，测试您的系统在虚拟或临时环境中是否有足够内存。或者，您可以将扫描配置集定制为取消选择涉及整个 / 文件系统递归的规则：

- `rpm_verify_hashes`
- `rpm_verify_permissions`
- `rpm_verify_ownership`
- `file_permissions_unauthorized_world_writable`
- `no_files_unowned_by_user`
- `dir_perms_world_writable_system_owned`
- `file_permissions_unauthorized_suid`
- `file_permissions_unauthorized_sgid`
- `file_permissions_ungroupowned`
- `dir_perms_world_writable_sticky_bits`

这可防止 OpenSCAP 扫描造成系统内存不足。

([BZ#1824152](#))

5.7.5. 网络

当禁用 GRO 时，IPsec 网络流量在 IPsec 卸载过程中失败

当在该设备中禁用通用接收 Offload (GRO) 时，IPsec 卸载将不会正常工作。如果在一个网络接口中配置了 IPsec 卸载，且在该设备中禁用 GRO，IPsec 网络流量会失败。

要临时解决这个问题，在该设备中启用 GRO。

([BZ#1649647](#))

如果指定链类型未知，则 iptables 不会为更新链的命令请求模块加载

注意：如果您使用服务默认配置，则此问题会在停止 `iptables systemd` 服务时导致无功能隐患的错误。

当使用 `iptables-nft` 设置链策略时，发送到内核的更新链命令将失败（如果尚未加载相关的内核模块）。要临时解决这个问题，请使用以下命令使模块载入：

```
+
# iptables -t nat -n -L
# iptables -t mangle -n -L
```

([BZ#1812666](#))

nft_compat 模块自动加载特定于地址系列的 LOG 后端模块可能会挂起

当 `nft_compat` 模块同时加载网络命名空间(`netns`)上的操作时，可能会发生锁定冲突。因此，载入特定于地址的 LOG 目标后端可能会挂起。要临时解决这个问题，请在执行 `iptables-restore` 工具之前手动加载相关的 LOG 目标后端，如 `nf_log_ipv4.ko` 和 `nf_log_ipv6.ko`。因此，载入 LOG 目标后端不会挂起。

但是，如果在系统引导过程中出现问题，则没有可用的临时解决方案。

请注意，`libvirt d` 等其他服务也执行 `iptables` 命令，这可能会导致问题发生。

(BZ#1757933)

5.7.6. 内核

意外删除补丁会导致 `huge_page_setup_helper.py` 显示错误

更新 `huge_page_setup_helper.py` 脚本的补丁被意外删除。因此，执行 `huge_page_setup_helper.py` 脚本后会出现以下出错信息：

```
SyntaxError: Missing parentheses in call to 'print'
```

要临时解决这个问题，从 RHEL 8.1 中复制 `huge_page_setup_helper.py` 脚本并将其安装到 `/usr/bin/` 目录中：

1. 从 RHEL-8.1.0 安装介质或从 [红帽客户门户网站](#) 下载 `libhugetlbfs-utils-2.21-3.el8.x86_64.rpm` 软件包。
2. 执行 `rpm2cpio` 命令：

```
# rpm2cpio libhugetlbfs-utils-2.21-3.el8.x86_64.rpm | cpio -D / -iduv
*/huge_page_setup_helper.py'
```

该命令从 RHEL 8.1 RPM 中提取 `huge_page_setup_helper.py` 脚本并将其保存到 `/usr/bin/` 目录中。

因此，`huge_page_setup_helper.py` 脚本可以正常工作。

(BZ#1823398)

有大量持久内存的系统在引导过程中出现延迟

有大量持久内存的系统需要很长时间才能引导，因为初始化内存是序列化的。因此，如果 `/etc/fstab` 文件中列出了持久的内存文件系统，系统在等待设备可用时可能会超时。要临时解决这个问题，请将 `/etc/systemd/system.conf` 文件中的 `DefaultTimeoutStartSec` 选项配置为足够大的值。

(BZ#1666538)

KSM 有时会忽略 NUMA 内存策略

当内核共享内存(KSM)功能通过 `merge_across_nodes=1` 参数启用时，KSM 会忽略 `mbind()` 函数设置的内存策略，并且可能会将某些内存区域的页面合并到与策略不匹配的非一致性内存访问(NUMA)节点。

要临时解决这个问题，如果使用 NUMA 内存与 QEMU 绑定，请禁用 KSM 或将 `merge_across_nodes` 参数设置为 `0`。因此，为 KVM 虚拟机配置的 NUMA 内存策略可以正常工作。

(BZ#1153521)

Debug 内核无法在 RHEL 8 的崩溃捕获环境中引导

由于 debug 内核的内存需求特性，会在使用 debug 内核并触发内核 panic 时出现问题。因此，调试内核无法作为捕获内核引导，而是生成一个堆栈追踪。要临时解决这个问题，相应地增大崩溃内核内存。因此，debug 内核可以在崩溃捕获环境中成功引导。

(BZ#1659609)

zlib 可能会在某些压缩功能中减慢 vmcore 捕获速度

kdump 配置文件默认使用 **lzo** 压缩格式 (**makedumpfile -l**)。当您修改使用 **zlib** 压缩格式 (**makedumpfile -c**) 的配置文件时，可能会带来更好的压缩效果，但会牺牲 **vmcore** 捕获过程的速度。因此，与 **lzo** 相比，捕获有 **zlib** 的 **vmcore** 文件所需的时间最多可能比使用 **kdump** 要长四倍。

因此，如果速度对您非常重要，红帽推荐使用默认的 **lzo**。但是，如果目标机器在可用空间中较低，**zlib** 就是一个更好的选项。

(BZ#1790635)

在内存热插拔或拔出操作后，vmcore 捕获失败

执行内存 hot-plug 或 hot-unplug 操作后，会更新包含内存布局信息的设备树。因此，**makedumpfile** 实用程序会尝试访问不存在的物理地址。如果以下条件都满足，就会出现这个问题：

- IBM Power System 的 little-endian 变体运行 RHEL 8。
- 在系统中启用了 **kdump** 或者 **fadump** 服务。

因此，如果在内存 hot-plug 或 hot-unplug 操作后触发了内核崩溃，捕获内核将无法保存 **vmcore**。

要临时解决这个问题，在 hot-plug 或 hot-unplug 后重启 **kdump** 服务：

```
# systemctl restart kdump.service
```

因此，**vmcore** 在上述场景中被成功保存。

(BZ#1793389)

fadump 转储机制将网络接口重命名为 kdump-<interface-name>

当使用固件辅助转储(**fadump**)捕获 **vmcore** 并使用 SSH 或 NFS 协议将其保存到远程机器时，请将网络接口重命名为 **kdump-<interface-name>**。当 **<interface-name>** 为通用时（如 ***eth#** 或 **net#** 等等）进行重命名。这是因为初始 RAM 磁盘(**initrd**)中的 **vmcore** 捕获脚本在网络接口名称中添加 **kdump-** 前缀来保护持久性命名。由于同一 **initrd** 也用于常规启动，因此生产内核的接口名称也会更改。

(BZ#1745507)

启用 fadump 时，系统在引导时进入紧急模式

当 **initramfs** 方案中启用了 **fadump(kdump)** 或 **dracut squash** 模块时，系统进入紧急模式，因为 **systemd** Manager 无法获取挂载信息并将 LV 分区配置为挂载。要临时解决这个问题，请添加以下内核命令行参数 **rd.lvm.lv=<VG>/<LV>** 以正确发现并挂载失败的 LV 分区。因此，系统将在上述场景中成功引导。

(BZ#1750278)

使用 irqpoll 会导致 vmcore 生成失败

由于在 Amazon Web Services(AWS)云平台上运行的 64 位 ARM 架构中的 **nvme** 驱动程序存在问题，在第一个内核提供了 **irqpoll** 内核命令行参数时 **vmcore** 生成会失败。因此，在内核崩溃后，**/var/crash/** 目录中不会转储 **vmcore** 文件。要临时解决这个问题：

1. 在 **/etc/sysconfig/kdump** 文件的 **KDUMP_COMMANDLINE_REMOVE** 键里添加 **irqpoll**。

2. 运行 `systemctl restart kdump` 命令重启 `kdump` 服务。

因此，第一个内核会正确引导，在内核崩溃时可以捕获 `vmcore` 文件。

请注意，`kdump` 服务可能会使用大量崩溃内核内存转储 `vmcore` 文件。确定捕获内核有足够的内存可用于 `kdump` 服务。

(BZ#1654962)

使用 vPMEM 内存作为转储目标会延迟内核崩溃捕获过程

当您使用 Virtual Persistent Memory (vPMEM) 命名空间用作 `kdump` 或 `fadump` 目标时，`papr_scm` 模块必须取消 `map` 并重新 `map` 由 vPMEM 支持的内存，并将内存重新添加到其线性映射中。因此，这个行为会向 POWER Hypervisor (H Calls) 触发 Hypervisor (H Calls)，从而大大缩短捕获内核引导的速度。因此，建议您不要将 vPMEM 命名空间用作 `kdump` 或 `fadump` 的转储目标。

如果您必须使用 vPMEM，要解决这个问题请执行以下命令：

1. 创建 `/etc/dracut.conf.d/99-pmem-workaround.conf` 文件并添加：

```
add_drivers+="nd_pmem nd_btt libnvdimm papr_scm"
```

2. 重建初始 RAM 磁盘 (initrd) 文件系统：

```
# touch /etc/kdump.conf
# systemctl restart kdump.service
```

(BZ#1792125)

HP NMI 监视器并不总是生成崩溃转储

在某些情况下，HP NMI watchdog 的 `hpwdt` 驱动无法声明一个由 HPE watchdog timer 生成的不可屏蔽中断 (NMI)，因为 NMI 被 `perfmon` 驱动所消耗。

缺少的 NMI 是由以下两个条件之一引发的：

1. Integrated Lights-Out (iLO) 服务器管理软件中的 **Generate NMI** 按钮。这个按钮由用户触发。
2. `hpwdt` watchdog。默认过期会向服务器发送一个 NMI。

在系统无响应时通常会出现这两个序列。在一般情况下，用于这两种情况的 NMI 处理程序调用 `kernel panic()` 功能，如果配置了，`kdump` 服务会生成 `vmcore` 文件。

由于缺少 NMI，没有调用 `kernel panic()` 且不收集 `vmcore`。

第一种情况(1.)，如果系统不响应，它会一直处于这个状态。要临时解决这种情况，请使用虚拟 **Power** 按钮来重置或者启用服务器。

在第二个示例中 (2.)，缺少的 NMI 之后会在 9 秒后被自动系统恢复 (ASR) 重置。

HPE Gen9 服务器行以单位数字显示这个问题。Gen10 频率更小。

(BZ#1602962)

`tuned-adm profile powerave` 命令会导致系统变得无响应

执行 **tuned-adm 配置集 powersave** 命令会导致 Penguin Valkyrie 2000 2-socket 系统具有较旧 Thunderx(CN88x)处理器的无响应状态。因此，需要重启系统以便恢复工作。要临时解决这个问题，如果您的系统符合上述说明，请避免使用 **powersave** 配置集。

(BZ#1609288)

cxgb4 驱动会导致 kdump 内核崩溃

在 **vmcore** 文件中保存信息时 **kdump** 内核会崩溃。因此，**cxgb4** 驱动程序可防止 **kdump** 内核保存内核以便稍后进行分析。要临时解决这个问题，在 **kdump** 内核命令行中添加 **novmcoredd** 参数以便保存核心文件。

(BZ#1708456)

尝试将 ICE 驱动程序 NIC 端口添加到模式 5(balance-tlb)绑定 master 接口可能会导致失败

尝试将 **ICE** 驱动程序 NIC 端口添加到模式 5(balance-tlb)绑定 master 接口可能会导致失败，并显示 **Master 'bond0', Slave 'ens1f0': Error: Enslave failed.** 因此，您会出现将 NIC 端口添加到绑定 master 接口的间歇性失败的情况。要解决这个问题，请尝试重试添加接口。

(BZ#1791664)

将虚拟功能附加到接口 type='hostdev' 可能会有时失败

使用 .XML 文件将虚拟功能(VF)附加到虚拟机时，可能会按照带有 **<interface type='hostdev'>** 方法的 **Assignment** 方法失败。这是因为，使用带有 **<interface type='hostdev'>** 方法的 **Assignment** 可以防止虚拟机附加到显示在此虚拟机的 VF NIC 中。要解决这个问题，请使用 **Assignment with <hostdev>** 方法使用 .XML 文件将 VF 附加到虚拟机。因此，**vir sh attach-device** 命令可以成功且没有错误。有关 **Assignment with <hostdev>** 和 **Assignment with <interface type='hostdev'>** (仅限 SRIOV 设备) 之间区别的详情，请参阅 [主机网络设备的 PCI Passthrough](#)。

(BZ#1792691)

5.7.7. 文件系统和存储

无法将 /boot 文件系统放在 LVM 中

您不能将 **/boot** 文件系统放在 LVM 逻辑卷中。这种限制的原因如下：

- 在 EFI 系统中，**EFI** 系统分区通常充当 **/boot** 文件系统。UEFI 标准要求有特定的 GPT 分区类型和具体文件系统类型。
- RHEL 8 在系统引导条目中使用 *Boot Loader 规格 (BLS)*。这个规格要求 **/boot** 文件系统可由平台固件可读。在 EFI 系统中，平台固件只能读取 uEFI 标准定义的 **/boot** 配置。
- 在 GRUB 2 引导装载程序中不支持 LVM 逻辑卷。红帽没有计划进行改进，因为如 uEFI 和 BLS 的标准，这个功能的使用情况正在下降。

红帽不计划在 LVM 中支持 **/boot**。反之，红帽提供了管理系统快照和回滚的工具，这些工具不需要将 **/boot** 文件系统放在 LVM 逻辑卷中。

(BZ#1496229)

LVM 不再允许使用混合块大小创建卷组

LVM 工具（如 **vgcreate** 或 **vgextend**）不再允许您创建有不同逻辑块大小的物理卷 (PV) 的卷组(VG)。LVM 启用了这个更改，因为如果您使用不同块大小的 PV 扩展了基本逻辑卷 (LV)，文件系统将无法挂载。

要重新创建带有混合块大小的 VG，在 `lvm.conf` 文件中设置 `allow_mixed_block_sizes=1` 选项。

(BZ#1768536)

当连接太多 LUN 时，DM 多路径可能无法启动

如果太多逻辑单元(LUN)连接到系统，`multipathd` 服务可能会超时且无法启动。造成此问题的确切 LUN 数量取决于多个因素，包括设备数量、存储阵列的响应时间、内存和 CPU 配置以及系统负载。

要临时解决这个问题，在 `multipathd` 单元文件中增加超时值：

1. 在单元编辑器中打开 `multipathd` 单元：

```
# systemctl edit multipathd
```

2. 输入以下配置来覆盖超时值：

```
[Service]
TimeoutSec=300
```

红帽建议将默认值从默认值 90 增加到 300，但您也可以测试 90 以上的其他值。

3. 在编辑器中保存文件。
4. 重新载入 `systemd` 单元以应用更改：

```
# systemctl daemon-reload
```

现在，`multipathd` 可以从更多 LUN 成功启动。

(BZ#1797660)

LVM writecache 的限制

`writecache` LVM 缓存方法有以下限制，这些限制不会出现在 `cache` 方法中：

- 当逻辑卷使用 `writecache` 时，您无法为逻辑卷生成快照。
- 当逻辑卷处于活跃状态时，您无法附加或分离 `writecache`。
- 将 `writecache` 附加到不活跃的逻辑卷时，您必须使用与现有文件系统块大小匹配的 `writecache` 块大小。
详情请查看 `lvmcache(7)` man page。
- 您不能在附加 `writecache` 时调整逻辑卷的大小。
- 您不能对与 `writecache` 一起使用的设备使用 `pvmove` 命令。
- 您不能将带有 `writecache` 的逻辑卷与精简池或 VDO 结合使用。

(JIRA:RHELPLAN-27987、BZ#1798631、BZ#1808012)

保存一个 LUKS 卷的 LVM mirror 设备有时将变为无响应

在某些情况下，保存 LUKS 卷的片段类型的 `mirror` LVM 设备可能会变得无响应。无响应设备会拒绝所有 I/O 操作。

要解决这个问题，红帽建议在有弹性软件定义的存储之上使用带 **raid1** 的片段类型的 LVM RAID 1 设备而不是镜像 (**mirror**)。

raid1 segment 类型是默认的 RAID 配置类型，它作为推荐的解决方案替换 **mirror**。

要将 **mirror** 设备转换为 **raid1**，请参阅[将镜像 LVM 设备转换为 RAID1 设备](#)。

(BZ#1730502)

NFS 4.0 补丁可能会导致 open-heavy 工作负载性能降低。

在以前的版本中，存在一个程序错误，在某些情况下，可能会导致 NFS 打开操作覆盖文件已被删除或重命名在服务器中的事实。但是，这个修复可能会在需要很多打开操作的工作负载中造成性能下降。要临时解决这个问题，您可能需要使用 NFS 版本 4.1 或更高版本，这些版本已被改进为客户端在本地、快速和安全地执行开放操作。

(BZ#1748451)

5.7.8. 动态编程语言、网页和数据库服务器

当有 32 位应用程序调用 `getpwnam()` 时，可能会失败

当 NIS 用户使用 32 位应用程序调用 `getpwnam()` 函数时，如果没有 `nss_nis.i686` 软件包，则调用会失败。要临时解决这个问题，使用 `yum install nss_nis.i686` 手动安装缺少的软件包。

(BZ#1803161)

nginx 无法从硬件安全令牌加载服务器证书

nginx web 服务器支持直接从 PKCS#11 模块的硬件安全令牌加载 TLS 私钥。但是，目前无法通过 PKCS#11 URI 从硬件安全令牌加载服务器证书。要临时解决这个问题，在文件系统中存储服务器证书

(BZ#1668717)

当使用 PHP 7.2 安装 `php-opcache` 时，`php-fpm` 会引起 SELinux AVC 拒绝

安装 `php-opcache` 软件包后，FastCGI Process Manager (`php-fpm`) 会导致 SELinux AVC 拒绝。要临时解决这个问题，将 `/etc/php.d/10-opcache.ini` 文件中的默认配置改为：

```
opcache.huge_code_pages=0
```

请注意，此问题仅影响 `php:7.2` 流，而非 `php:7.3` 流。

(BZ#1670386)

作为依赖项安装时，`mod_wsgi` 软件包名称缺失

随着 `mod_wsgi` 安装中的更改（如 [BZ#1779705](#) 所述），`python3-mod_wsgi` 软件包不再提供名称 `mod_wsgi`。安装 `mod_wsgi` 模块时，您必须指定完整的软件包名称。此更改会导致第三方软件包依赖项出现问题。

如果您尝试安装需要名为 `mod_wsgi` 的依赖项的第三方软件包，则会返回类似如下的错误：

```
Error:
Problem: conflicting requests
- nothing provides mod_wsgi needed by package-requires-mod_wsgi.el8.noarch
```

要临时解决这个问题，请选择以下之一：

- a. 重新构建软件包（或向第三方供应商询问新构建），以要求完整软件包名称 **python3-mod_wsgi**。
- b. 创建一个带有缺失软件包名称的 meta 软件包：
 1. 构建自己的空 meta 软件包，它提供名称 **mod_wsgi**。
 2. 将 **module_hotfixes=True** 行添加到包含 meta 软件包的存储库的 **.repo** 配置文件。
 3. 手动安装 **python3-mod_wsgi**。

([BZ#1829692](#))

5.7.9. 编译器和开发工具

GCC confuse SystemTap 生成的复合功能

GCC 优化可以为其他函数的部分内嵌副本生成复合函数。SystemTap 和 GDB 等工具无法将这些复合功能与实际功能区分开来。因此，SystemTap 将探测放置在复合和实际函数入口点上，因此为单个实际函数调用注册多个探测命中。

要临时解决这个问题，修改 SystemTap 脚本以检测递归并防止放置与内联部分功能相关的探测。

这个示例脚本

```
probe kernel.function("can_nice").call { }
```

可以通过这种方式修改：

```
global in_can_nice%

probe kernel.function("can_nice").call {
  in_can_nice[tid()] ++;
  if (in_can_nice[tid()] > 1) { next }
  /* code for real probe handler */
}

probe kernel.function("can_nice").return {
  in_can_nice[tid()] --;
}
```

请注意，这个示例脚本不会考虑所有可能的情况，如错过的 kprobes 或 kretprobes 或真正的预期递归。

([BZ#1169184](#))

5.7.10. Identity Management

更改 `/etc/nsswitch.conf` 需要手动重启系统

对 `/etc/nsswitch.conf` 文件的任何更改（例如运行 **authselect select profile_id** 命令）都需要重启系统，以便所有相关进程使用更新版本的 `/etc/nsswitch.conf` 文件。如果无法重新启动系统，请重新启动将您的系统加入 Active Directory 的服务，即 **系统安全服务后台程序 (SSSD)** 或 **winbind**。

([BZ#1657295](#))

启用文件域时，SSSD 为本地用户返回不正确的 LDAP 组成员资格

如果系统安全服务守护进程(SSSD)从本地文件和 `ldap_rfc2307_fallback_to_local_users` 属性的 `sssd.conf` 文件的 `[domain/LDAP]` 部分为用户提供服务，则文件提供程序不包含来自其他域的组成员资格。因此，如果本地用户是 LDAP 组的成员，`id local_user` 命令不会返回用户的 LDAP 组成员资格。要临时解决这个问题，请通过添加 `enable_files_domain` 来禁用隐式文件域

```
enable_files_domain=False
```

到 `/etc/sss/sss.conf` 文件中的 `[sss]` 部分：

因此，`id local_user` 会为本地用户返回正确的 LDAP 组成员资格。

(BZ#1652562)

SSSD 无法正确处理具有相同优先级的多个证书匹配规则

如果给定证书与多个具有相同优先级的证书匹配规则匹配，系统安全服务守护进程(SSSD)只使用其中一个规则。作为临时解决方案，请使用单个证书匹配规则，该规则 LDAP 过滤器由与 `|`（或）运算符串联的单独规则的过滤器组成。有关证书匹配规则的示例，请参阅 `sss-certamp(5)man page`。

(BZ#1447945)

当定义了多个域时，无法使用 `auto_private_group = 混合创建专用组`

如果定义了多个域，并且第一个域以外的任何域使用混合选项，则专用组无法通过选项 `auto_private_group = 混合创建`。如果隐式文件域与 `sssd.conf` 文件中的 AD 或 LDAP 域一起定义，且未标记为 `MPG_HYBRID`，那么 SSSD 无法为具有 `uid=gid` 且具有此 `gid` 的组在 AD 或 LDAP 中创建私有组。

`sssd_nss` 响应程序仅检查第一个域中 `auto_private_groups` 选项的值。因此，在配置多个域的设置中，在 RHEL 8 中包括默认设置的设置中，选项 `auto_private_group` 无效。

要临时解决这个问题，请在 `sssd.conf` 的 `sss` 部分中设置 `enable_files_domain = false`。因此，如果 `enable_files_domain` 选项被设置为 `false`，则 `sssd` 不会在活跃域列表的开头添加 `id_provider=files` 的域，因此不会出现这个程序错误。

(BZ#1754871)

python-ply 不兼容 FIPS

`python-ply` 软件包的 YACC 模块使用 MD5 哈希算法来生成 YACC 签名的指纹。但是，FIPS 模式会阻止使用 MD5，只有非安全上下文中才允许这样做。因此，`python-ply` 不兼容 FIPS。在 FIPS 模式中的系统中，对 `ply.yacc.yacc()` 的所有调用都会失败，并显示错误消息：

```
UnboundLocalError: local variable 'sig' referenced before assignment
```

问题会影响 `python-pycparser` 和 `python-cffi` 的一些用例。要临时解决这个问题，修改 `/usr/lib/python3.6/site-packages/ply/yacc.py` 文件的第 2966 行，将 `sig = md5()` 替换为 `sig = md5(usedforsecurity=False)`。因此，`python -ply` 可以在 FIPS 模式中使用。

(BZ#1747490)

FreeRADIUS 会默默地截断大于 249 个字符的 Tunnel-Password

如果 Tunnel-Password 大于 249 个字符，则 FreeRADIUS 服务会默默地截断它。这可能导致无法预期的，与其它系统不兼容的密码。

要临时解决这个问题，请选择 249 个字符或更少的密码。

(BZ#1723362)

如果所有 KRA 成员都是隐藏的副本，则安装 KRA 会失败

如果在隐藏的副本中安装第一个 KRA 实例，`ipa a-kra-install` 工具程序会在已存在密钥恢复授权(KRA)的集群中失败。因此，您无法向集群添加更多 KRA 实例。

要临时解决这个问题，请在添加新的 KRA 实例前，清除具有 KRA 角色的隐藏副本。`ipa-kra-install` 成功后您可以再次隐藏它。

(BZ#1816784)

如果在搜索过滤器中使用了这些属性，目录服务器会警告 schema 中缺少的属性

如果您将 `then sslapd-verify-filter-schema` 参数设置为 `warn-invalid`，Directory Server 将处理搜索操作（带有架构中未定义的属性），并记录警告。使用这个设置时，Directory 服务器会在搜索结果中返回请求的属性，无论这些属性是否在架构中定义。

目录服务器的未来版本将更改默认设置 `nsslapd-verify-filter-schema` 来强制进行更严格的检查。新默认值将针对架构中缺少的属性发出警告，并且拒绝请求或仅返回部分结果。

(BZ#1790259)

ipa-healthcheck-0.4 没有过时的 ipa-healthcheck 的旧版本

`Healthcheck` 工具已分成两个子软件包：`ipa-healthcheck` 和 `ipa-healthcheck-core`。但是，只有 `ipa-healthcheck-core` 子软件包被正确设置为过时的 `ipa-healthcheck` 版本。因此，更新 `Healthcheck` 只会安装 `ipa-healthcheck-core`，`ipa-healthcheck` 命令在更新后无法正常工作。

要临时解决这个问题，使用 `yum install ipa-healthcheck-0.4` 手动安装 `ipa-healthcheck-0.4` 子软件包。

(BZ#1852244)

对 `ldap_id_use_start_tls` 选项使用默认值时潜在的风险

当使用没有 TLS 的 `ldap://` 进行身份查找时，可能会带来攻击向量的风险。特别是中间人(MITM)攻击，其使攻击者可以通过更改例如在 LDAP 搜索中返回的对象的 UID 或 GID 来冒充用户。

目前，强制 TLS 的 SSSD 配置选项 `ldap_id_use_start_tls` 默认为 `false`。确保您的设置操作在可信环境中进行，并决定对 `id_provider = ldap` 使用未加密的通信是否是安全的。注意 `id_provider = ad` 和 `id_provider = ipa` 不受影响，因为它们使用 SASL 和 GSSAPI 保护的加密连接。

如果使用未加密的通信不安全，请在 `/etc/sss/sss.conf` 文件中通过将 `ldap_id_use_start_tls` 选项设置为 `true` 来强制使用 TLS。计划在以后的 RHEL 版本中更改的默认行为。

(JIRA:RHELPLAN-155168)

5.7.11. Desktop

Wayland 会话的限制

在 Red Hat Enterprise Linux 8 中，GNOME 环境和 GNOME 显示管理器(GDM)使用 `Wayland` 作为默认会话类型，而不是 `X11` 会话，这些会话与之前的 RHEL 主要版本一起使用。

当前无法使用以下功能，或者在 `Wayland` 下无法正常工作：

- **X11 配置实用程序**（如 **xrandr**）因为处理、解决方案、轮转和布局的方法不同而无法在 **Wayland** 下工作。您可以使用 **GNOME 设置配置显示功能**。
- 屏幕记录和远程桌面需要应用程序来支持 **Wayland** 上的门户 API。某些传统应用程序不支持门户 API。
- **Wayland** 上不提供指针可访问性。
- 没有可用的剪贴板管理器。
- **Wayland** 上的 **GNOME Shell** 忽略了大多数传统 **X11** 应用发布的**键盘粒度**。您可以使用 **/org/gnome/mutter/wayland/xwayland-grab-access-rules** **GSettings** 键启用 **X11** 应用程序发布键盘 grabs。默认情况下，**Wayland** 上的 **GNOME Shell** 允许以下应用发布**键盘** grabs：
 - **GNOME Boxes**
 - **vinagre**
 - **Xephyr**
 - **virt-manager**、**virt-viewer** 和 **remote-viewer**
 - **vncviewer**
- 客户机虚拟机(VM)中的 **Wayland** 具有稳定性和性能问题。在虚拟机中运行时，**RHEL** 会自动回退到 **X11** 会话。

如果您从使用 **X11** **GNOME** 会话的 **RHEL 7** 系统升级到 **RHEL 8**，您的系统将继续使用 **X11**。使用以下图形驱动程序时，系统还会自动回退到 **X11**：

- 专有 **NVIDIA** 驱动程序
- **cirrus** 驱动程序
- **mga** 驱动程序
- 一个**速度** 驱动程序

您可以手动禁用 **Wayland** 的使用：

- 要在 **GDM** 中禁用 **Wayland**，请在 **/etc/gdm/custom.conf** 文件中设置 **WaylandEnable=false** 选项。
- 要在 **GNOME** 会话中禁用 **Wayland**，请在输入登录名称后使用登录屏幕上的 **cogwheel** 菜单来选择旧的 **X11** 选项。

有关 **Wayland** 的详情，请参考 <https://wayland.freedesktop.org/>。

([BZ#1797409](#))

在桌面和应用程序间进行拖放操作无法正常工作

由于 **gnome-shell-extensions** 软件包中的一个 bug，**drag-and-drop** 功能目前在桌面和应用程序间无法正常工作。以后的发行版本中将重新添加对这个功能的支持。

([BZ#1717947](#))

无法从软件仓库中禁用 **flatpak** 程序库

目前，在 GNOME 软件工具中的软件程序库工具中无法禁用或删除 **flatpak** 程序库。

(BZ#1668760)

第二代 RHEL 8 虚拟机有时无法在 Hyper-V Server 2016 主机上引导

当使用 RHEL 8 作为在 Microsoft Hyper-V Server 2016 主机上运行的虚拟机(VM)中的客户机操作系统时，虚拟机在某些情况下无法引导，并返回到 GRUB 引导菜单。另外，会在 Hyper-V 事件日志中记录以下错误：

```
The guest operating system reported that it failed with the following error code: 0x1E
```

这个错误是由 Hyper-V 主机上的 UEFI 固件错误造成的。要临时解决这个问题，使用 Hyper-V Server 2019 作为主机。

(BZ#1583445)

系统崩溃可能会导致 fadump 配置丢失

在启用了固件辅助转储(fadump)且引导分区位于 XFS 等日志记录文件系统系统中的系统中会出现此问题。系统崩溃可能会导致引导装载程序加载未启用转储捕获支持的较早 **initrd**。因此，恢复后，系统不会捕获 **vmcore** 文件，这会导致 fadump 配置丢失。

要临时解决这个问题：

- 如果 **/boot** 是一个独立的分区，请执行以下操作：
 1. 重启 kdump 服务
 2. 以 root 用户身份运行以下命令，或使用具有 CAP_SYS_ADMIN 权限的用户帐户：

```
# fsfreeze -f
# fsfreeze -u
```

- 如果 **/boot** 不是单独的分区，请重启该系统。

(BZ#1723501)

5.7.12. 图形基础结构

无法使用 sudo 命令运行图形应用程序

当试图以具有更高权限的用户运行图形应用程序时，应用程序无法打开并带有一个出错信息。发生故障的原因是 **Xwayland** 被 **Xauthority** 文件限制，为使用常规用户凭证进行验证。

要临时解决这个问题，使用 **sudo -E** 命令以 **root** 用户运行图形程序。

(BZ#1673073)

radeon 无法正确重置硬件

radeon 内核驱动程序目前没有能在 kexec 上下文中正确重置硬件。相反，**radeon** 无法工作，从而导致剩余的 **kdump** 服务失败。

要临时解决这个问题，在 **kdump** 中通过在 **/etc/kdump.conf** 文件中添加以下行来使用 blacklis Trade on：

```
dracut_args --omit-drivers "radeon"  
force_rebuild 1
```

重启机器和 `kdump`。启动 `kdump` 后，`force_rebuild 1` 行可能会从配置文件中删除。

请注意，在这种情况下，`kdump` 不会提供图形，但 `kdump` 可成功运行。

(BZ#1694705)

多个 HDR 显示在单个 MST 拓扑上可能无法打开

在使用带有 `nouveau` 驱动程序的 NVIDIA Turing GPU 的系统上，使用带有多个监视器的 `DisplayPort` hub（如便携式计算机 dock）支持 HDR 插入的多个监视器可能会导致所有显示无法打开，尽管之前的 RHEL 版本上仍然这样做。这是因为系统错误地认为 hub 中没有足够的带宽来支持所有显示器。

(BZ#1812577)

5.7.13. Web 控制台

非特权用户可以访问订阅页面

如果非管理员访问 web 控制台的 `Subscriptions` 页面，Web 控制台会显示一个通用错误消息 `Cockpit had an unexpected internal error`。

要临时解决这个问题，使用特权用户登录到 web 控制台，并选择 `Reuse my password for privileged tasks` 复选框。

(BZ#1674337)

5.7.14. 虚拟化

Windows Server 2019 主机上的 RHEL 8 虚拟机中的低 GUI 显示性能

当在 Windows Server 2019 主机上以图形模式使用 RHEL 8 作为客户机操作系统时，GUI 显示性能较低，并连接到客户机的控制台输出所需的时间比预期的要长得多。

这是 Windows 2019 主机上的已知问题，并由 Microsoft 解决。要临时解决这个问题，请使用 SSH 连接到客户端，或使用 Windows Server 2016 作为主机。

(BZ#1706541)

无法通过 QXL 显示多个使用 Wayland 的虚拟机的监控器

使用 `remote-viewer` 工具来显示使用 Wayland 显示服务器的虚拟机(VM)的多个显示器，会导致 VM 变得无响应，并永久显示 `Waiting for display` 状态信息。

要临时解决这个问题，使用 `virtio-gpu` 而不是 `qxl` 作为使用 Wayland 的虚拟机的 GPU 设备。

(BZ#1642887)

`virsh iface-*` 命令无法一致性地工作

因为配置的依赖关系，目前 `virsh iface-*` 命令（如 `virsh iface-start` 和 `virsh iface-destroy` 会经常失败。因此，建议您不要使用 `virsh iface-*` 命令配置和管理主机网络连接。反之，使用 `NetworkManager` 程序及其相关管理程序。

(BZ#1664592)

RHEL 8 虚拟机有时无法引导至 Witherspoon 主机

在某些情况下，使用 **pseries-rhel7.6.0-sxxm** 机器类型的 RHEL 8 虚拟机(VM)无法针对使用 DD2.2 或 DD2.3 CPU 的 HPC 主机（也称为 *Witherspoon*）在 Power9 S922LC 上启动。

尝试引导这样的虚拟机会生成以下出错信息：

```
qemu-kvm: Requested safe indirect branch capability level not supported by kvm
```

要临时解决这个问题，请按如下方式配置虚拟机的 XML 配置：

```
<domain type='qemu' xmlns:qemu='http://libvirt.org/schemas/domain/qemu/1.0'>
  <qemu:commandline>
    <qemu:arg value='-machine'/>
    <qemu:arg value='cap-ibs=workaround'/>
  </qemu:commandline>
```

([BZ#1732726](#),[BZ#1751054](#))

IBM POWER 虚拟机无法在空 NUMA 节点中正常工作

目前，当在 RHEL 8 主机上运行的 IBM POWER 虚拟机(VM)配置为使用零内存(**memory='0'**)和零 CPU 的 NUMA 节点时，虚拟机将无法启动。因此，红帽强烈建议不要在 RHEL 8 中使用具有空 NUMA 节点的 IBM POWER 虚拟机。

([BZ#1651474](#))

当在 AMD EPYC 上使用主机透传模式时，虚拟机不会检测到 SMT CPU 拓扑

当在 AMD EPYC 主机上使用 CPU 主机 passthrough 模式引导虚拟机(VM)时，**TOPOEXT** CPU 功能标志不存在。因此，虚拟机无法检测到每个内核有多个线程的虚拟 CPU 拓扑。要临时解决这个问题，使用 EPYC CPU 模型而不是主机透传引导虚拟机。

([BZ#1740002](#))

RHEL 8.2 虚拟机中的磁盘标识符可能会在虚拟机重启时改变。

当将带有 RHEL 8.2 的虚拟机(VM)用作 Hyper-V 管理程序中的客户机操作系统时，虚拟机虚拟磁盘的设备标识符在某些情况下会在虚拟机重启时改变。例如，最初标识为 **/dev/sda** 的磁盘可能会变为 **/dev/sdb**。因此，虚拟机可能无法引导，引用 VM 磁盘的脚本可能会停止工作。

为避免这个问题，红帽强烈建议为虚拟机中的磁盘设置持久名称。详情请参阅 [Microsoft Azure 文档](#)：

([BZ#1777283](#))

当使用很多 virtio-blk 磁盘时，虚拟机有时无法启动

在虚拟机(VM)中添加大量 virtio-blk 设备可能会耗尽平台中可用的中断向量。如果发生了这种情况，VM 的客户机操作系统无法引导，并显示 **dracut-initqueue[392]: Warning: Could not boot** 错误。

([BZ#1719687](#))

使用 virtio-blk 将 LUN 设备附加到虚拟机中无法正常工作

q35 机器类型不支持 virtio 1.0 设备，因此 RHEL 8 不支持 virtio 1.0 中弃用的功能。特别是，RHEL 8 主机无法从 virtio-blk 设备发送 SCSI 命令。因此，使用 virtio-blk 控制器时，将物理磁盘作为 LUN 设备附加到虚拟机会失败。

请注意，物理磁盘仍可被传递给客户端操作系统，但应该使用 **device='disk'** 选项而不是 **device='lun'** 选项进行配置。

(BZ#1777138)

将 POWER9 客户端从 RHEL 7-ALT 主机迁移到 RHEL 8 会失败

目前，将 POWER9 虚拟机从 RHEL 7-ALT 主机系统迁移到 RHEL 8 变得无响应，并带有 "Migration status: active" 状态。

要临时解决这个问题，在 RHEL 7-ALT 主机上禁用 Transparent Huge Pages (THP)，这样可使迁移成功完成。

(BZ#1741436)

5.7.15. 支持性

redhat-support-tool 无法用于 FUTURE 加密策略

因为客户门户网站 API 中的证书使用的加密密钥不满足 **FUTURE** 系统范围的加密策略的要求，所以 **redhat-support-tool** 程序目前无法使用这个策略级别。

要临时解决这个问题，在连接到客户门户网站 API 时使用 **DEFAULT** 加密策略。

(BZ#1802026)

5.7.16. 容器

UDICA 无法与 1.0 稳定流工作

UDICA 是为容器生成 SELinux 策略的工具，不能与通过 **container-tools:1.0** 模块流中的 podman 1.0.x 运行的容器一起使用。

(JIRA:RHELPLAN-25571)

关于 Podman 支持 FIPS 的备注

联邦信息处理标准(FIPS)要求使用经过认证的模块。在以前的版本中，Podman 通过在启动时启用正确的标记在容器中正确安装认证模块。但是，在此发行版本中，Podman 没有以 FIPS 系统范围 crypto-policy 的形式正确设置系统通常提供的额外应用程序帮助程序。尽管认证模块不需要设置系统范围的 crypto-policy，但它确实提高了应用程序以兼容方式使用加密模块的能力。要临时解决这个问题，请在执行任何其他应用程序代码前更改容器以运行 **update-crypto-policies --set FIPS** 命令。

(BZ#1804193)

第 6 章 国际化

6.1. RED HAT ENTERPRISE LINUX 8 国际语言

Red Hat Enterprise Linux 8 支持多种语言的安装，并根据您的需要更改语言。

- 东亚语言 - 日语、韩语、简体中文和繁体中文。
- 欧洲语言 - 英语、德语、西班牙语、法语、意大利语、葡萄牙语和俄语。

下表列出了为各种主要语言提供的字体和输入法。

语言	默认字体（字体软件包）	输入法
English	dejavu-sans-fonts	
法语	dejavu-sans-fonts	
德语	dejavu-sans-fonts	
意大利语	dejavu-sans-fonts	
俄语	dejavu-sans-fonts	
西班牙语	dejavu-sans-fonts	
葡萄牙语	dejavu-sans-fonts	
简体中文	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-libpinyin, libpinyin
繁体中文	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-libzhuyin, libzhuyin
日语	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-kkc, libkc
韩语	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-hanguk, libhanguk

6.2. RHEL 8 国际化的显著变化

RHEL 8 与 RHEL 7 相比，对国际化进行了以下更改：

- 添加了对 **Unicode 11** 计算行业标准的支持。
- 国际化发布在多个软件包中，这样就可以进行较小的内存占用安装。如需更多信息，[请参阅使用语言包](#)。
- 现在，多个区域的 **glibc** 软件包更新与 Common Locale Data Repository (CLDR) 同步。

附录 A. 按组件划分的问题单列表

Bugzilla 和 JIRA ID 列在本文档中以便参考。可公开访问的 Bugzilla 程序错误包括到 ticket 的链接。

组件	票证
389-ds-base	BZ#1715406 , BZ#1748016 , BZ#1790259 , BZ#1748994 , BZ#1739718
NetworkManager	BZ#1626348
anaconda	BZ#1747382 , BZ#1637472 , BZ#1748756 , BZ#1649359 , BZ#1715303 , BZ#1696609 , BZ#1672405 , BZ#1687747 , BZ#1745064 , BZ#1659400 , BZ#1821192 , BZ#1822880 , BZ#1823578 , BZ#1748281 , BZ#1746391
Audit	BZ#1757986
authselect	BZ#1657295
bind	BZ#1564443 , BZ#1664863 , BZ#1704328
binutils	BZ#1777002 , BZ#1618748
buildah-container	BZ#1627898
clevis	BZ#1766526 , BZ#1564559 , BZ#1436780 , BZ#1784524
cloud-init	BZ#1641190 , BZ#1666961
cockpit-appstream	BZ#1676506
cockpit	BZ#1678465 , BZ#1754163 , BZ#1666722
container-tools-rhel8-module	BZ#1784267
corosync-qdevice	BZ#1784200
createrepo_c	BZ#1743186
crypto-policies	BZ#1690565 , BZ#1660839
device-mapper-multipath	BZ#1797660
dhcp	BZ#1729211
distribution	BZ#1657927

组件	票证
dnf	BZ#1676891 , BZ#1754609
dnsmasq	BZ#1700916
edk2	BZ#1748180
elfutils	BZ#1744992
fapolicyd	BZ#1759895
fence-agents	BZ#1775847
firewalld	BZ#1737045 , BZ#1740670 , BZ#1733066
freeradius	BZ#1723362
gcc-toolset-9	BZ#1774118
gcc	BZ#1726641 , BZ#1698607 , BZ#1747157
gdb	BZ#1768593
gdm	BZ#1749960
glibc	BZ#1410154 , BZ#1764214 , BZ#1749439 , BZ#1764235 , BZ#1746928 , BZ#177241 , BZ#1361965 , BZ#1747502 , BZ#1764218 , BZ#1764238 , BZ#1746933 , BZ#1747453
gnome-shell-extensions	BZ#1717947
gnome-shell	BZ#1724302
gnome-software	BZ#1668760
gnutls	BZ#1628553 , BZ#1677754
go-toolset	BZ#1747150
grafana-pcp	BZ#1685315
grafana	BZ#1725278
graphviz	BZ#1704875
grub2	BZ#1583445 , BZ#1723501

组件	票证
httpd-2.4-module	BZ#1747923
httpd	BZ#1633224
initial-setup	BZ#1676439
ipa	BZ#1665051 , BZ#1816784 , BZ#1719767 , BZ#1777564 , BZ#1664719 , BZ#1664718
ipcalc	BZ#1638834
java-11-openjdk	BZ#1746875
kernel-rt	BZ#1680161
kernel	BZ#1744397, BZ#1698297, BZ#1687094, BZ#1720227, BZ#1846345, BZ#1635295, BZ#1793389, BZ#1706541, BZ#1666538, BZ#1602962, BZ#1649647, BZ#1153521, BZ#1694705, BZ#1348508, BZ#1748451, BZ#1708456, BZ#1654962, BZ#1609288, BZ#1777283, BZ#1791664, BZ#1792125, BZ#1792691, BZ#1812666, BZ#1812577, BZ#1757933, BZ#1763661, BZ#1780432, BZ#1401552, BZ#1716002, BZ#1593711, BZ#1620349, BZ#1724969, BZ#1714330, BZ#1714486, BZ#1660368, BZ#1524687, BZ#1274406, BZ#1650518, BZ#1636572, BZ#1727369, BZ#1519039, BZ#1627455, BZ#1501618, BZ#1495358, BZ#1633143, BZ#1503672, BZ#1570255, BZ#1696451, BZ#1665295, BZ#1658840, BZ#1660627, BZ#1569610, BZ#1730502
kexec-tools	BZ#1750278, BZ#1690729
kmod-kvdo	BZ#1737639 , BZ#1657301
krb5	BZ#1754690
libbpf	BZ#1759154
libdnf	BZ#1697472
libgnome-keyring	BZ#1607766
libndp	BZ#1697595
libpfm	BZ#1731019
libreswan	BZ#1777474

组件	票证
libselinux-python-2.8-module	BZ#1666328
libvirt	BZ#1749672 , BZ#1664592, BZ#1528684
llvm-toolset	BZ#1747139
lorax	BZ#1754711
ltrace	BZ#1655368
lvm2	BZ#1600174, BZ#1496229, BZ#1768536
make	BZ#1774790
maven	BZ#1783926
mod_wsgi	BZ#1829692 , BZ#1779705
Mutter	BZ#1737553
nfs-utils	BZ#1719983 , BZ#1592011
nftables	BZ#1778883 , BZ#1643192
nginx	BZ#1668717
nmstate	BZ#1674456
nss_nis	BZ#1803161
nss	BZ#1724250 , BZ#1817533 , BZ#1645153
numactl	BZ#1730738
opencv	BZ#1694647
openscap	BZ#1636431, BZ#1618489, BZ#1646197, BZ#1803116 , BZ#1795563 , BZ#1824152 , BZ#1642373
openssh	BZ#1744108
openssl-pkcs11	BZ#1705505 , BZ#1664807 , BZ#1745082
openssl	BZ#1685470 , BZ#1749068

组件	票证
oscap-anaconda-addon	BZ#1665082, BZ#1674001 , BZ#1691305, BZ#1787156 , BZ#1816199
pacemaker	BZ#1712584 , BZ#1700104
pam	BZ#1252859, BZ#1537242
pcp	BZ#1723598
pcs	BZ#1631519 , BZ#1631514, BZ#1676431, BZ#1442116 , BZ#1619620
perl-LDAP	BZ#1663063
php-7.2-module	BZ#1670386
php-pecl-xdebug	BZ#1769857
pki-core	BZ#1698084 , BZ#1303254
podman	BZ#1804193 , BZ#1645280
policycoreutils	BZ#1563742, BZ#1417455
postfix	BZ#1723950 , BZ#1745321
powertop	BZ#1716721
pykickstart	BZ#1637872
python-ply	BZ#1747490
python38-3.8-module	BZ#1747329
qemu-kvm	BZ#1651474, BZ#1740002 , BZ#1719687 , BZ#1651994, BZ#1741346
rear	BZ#1729501
redhat-release	BZ#1817591
redhat-support-tool	BZ#1802026
rhel-system-roles-sap	BZ#1660832
rng-tools	BZ#1692435
rpm	BZ#1688849

组件	票证
rsyslog	JIRA:RHELPLAN-10431, BZ#1659383, BZ#1679512 , BZ#1740683, BZ#1676559 , BZ#1692073 , BZ#1692072
rust-toolset	BZ#1776847
s390utils	BZ#1750326
samba	BZ#1754409 , JIRA:RHELPLAN-13195
scap-security-guide	BZ#1755447 , BZ#1754919 , BZ#1750755 , BZ#1755194
scap-workbench	BZ#1640715
selinux-policy	BZ#1641631, BZ#1746398, BZ#1826788 , BZ#1727887 , BZ#1726166 , BZ#1726246
setools	BZ#1731519
setroubleshoot-plugins	BZ#1649842
设置	BZ#1730396 , BZ#1663556
skopeo-container	BZ#1627900
skopeo	BZ#1810053
sscg	BZ#1717880
sssd	BZ#1669407 , BZ#1652562 , BZ#1447945, BZ#1754871
subscription-manager	BZ#1674337
sudo	BZ#1786990, BZ#1733961
systemd	BZ#1686892 , BZ#1640802
systemtap	BZ#1744989
tpm2-tools	BZ#1725714
tuned	BZ#1738250
udica	BZ#1763210 , BZ#1732704
vdo	BZ#1713749

组件	票证
virt-manager	BZ#1677019
wayland	BZ#1673073
whois	BZ#1734183
xorg-x11-drv-qxl	BZ#1642887
xorg-x11-server	BZ#1698565
zlib	BZ#1659433, BZ#1666798
其他	BZ#1640697, BZ#1659609, BZ#1687900 , BZ#1697896, BZ#1797409 , BZ#1790635, BZ#1823398, BZ#1745507, BZ#1732726 , BZ#1757877, JIRA:RHELPLAN-25571, BZ#1777138, JIRA:RHELPLAN-27987, BZ#1797671 , BZ#1780124 , JIRA:RHELPLAN-2507, JIRA:RHELPLAN-37713, JIRA:RHELPLAN-37777, BZ#1841170 , JIRA:RHELPLAN-13995, BZ#1785248 , BZ#1755347 , BZ#1784455 , BZ#1784456 , BZ#1789401 , JIRA:RHELPLAN-41384, BZ#1690207, JIRA:RHELPLAN-1212, BZ#1559616, BZ#1812552 , JIRA:RHELPLAN-14047, BZ#1769727 , JIRA:RHELPLAN-27394, BZ#1642765, JIRA:RHELPLAN-10304, BZ#1646541, BZ#1647725, BZ#1686057 , BZ#1748980 , BZ#1827628

附录 B. 修改历史记录

0.3-7

2024 年 5 月 9 日星期四, Brian Angelica (bangelic@redhat.com)

- 在 [BZROX0207](#) 中更新了技术预览。

0.3-6

2024 年 5 月 9 日星期四, Gabriela Fialova (gfialova@redhat.com)

- 更新了一个已知问题 [BZcriu0502](#) (Storage)。

0.3-5

2023 年 12 月 7 日星期四, Lucie Vaááková(lvarakova@redhat.com)

- 添加了一个已弃用的功能 [JIRA:RHELDOCS-16612](#) (身份管理)。

0.3-4

2023 年 11 月 10 日星期五, Gabriela Fialová(gfialova@redhat.com)

- 更新了 RHEL 文档中的提供反馈模块。

0.3-3

2023 年 11 月 7 日星期二, Gabriela Fialová(gfialova@redhat.com)

- 修复损坏的链接。

0.3-2

2023 年 10 月 13 日星期五, Gabriela Fialová(gfialova@redhat.com)

- 添加了一个技术预览 [JIRA:RHELDOCS-16861](#) (容器)。

0.3-1

2023 年 4 月 27 日星期四, Gabriela Fialová(gfialova@redhat.com)

- 添加了一个已知问题 [JIRA:RHELPLAN-155168](#) (身份管理)。

0.3-0

2023 年 4 月 13 日星期四, Gabriela Fialová(gfialova@redhat.com)

- 使用到 Rust Toolset 启动页的链接更新了 [BZ#1811997](#)。

0.2-9

2023 年 3 月 1 日星期三, Lucie Vařáková (lvarakova@redhat.com)

- 添加了一个 bug 修复 [BZ#2151696](#) (身份管理)。

0.2-8

2022 年 4 月 29 日星期五, Lenka Špačková (lspackova@redhat.com)

- 更新了 [已弃用的功能](#) 简介。

- 修复了 [BZ#1605216](#) 中的拼写错误。
- 修复了损坏链接。

0.2-7

2021 年 12 月 23 日星期四, Lenka Špačková (lspackova@redhat.com)

- 向技术预览 [BZ#1605216](#) 和已弃用的功能 [BZ #1878207](#) (Kernel) 添加了有关 Soft-RoCE 驱动程序 `rdma_rxe` 的信息。

0.2-6

2021 年 10 月 5 日星期二, Lucie Maňásková (Imanasko@redhat.com)

- 添加了已弃用的功能 [BZ#1999620](#) (Shells 和命令行工具)。

0.2-5

Thu Aug 19 2021, Lucie Maňásková (Imanasko@redhat.com)

- 向分发一章中添加了 [使用 YUM/DNF 的软件包管理](#)。

0.2-4

Fri Jul 9 2021, Lucie Maňásková (Imanasko@redhat.com)

- 更新了 [JIRA : RHELPLAN-34199](#) (安全性)。

0.2-3

Wed Jun 23 2021, Lucie Maňásková (Imanasko@redhat.com)

- 更新了新功能部分(Installer)。

0.2-2

2021 年 5 月 21 日星期五, Lenka Špačková (lspackova@redhat.com)

- 更新了 [Overview](#) 中有关操作系统转换的信息。

0.2-1

Tue Apr 06 2021, Lenka Špačková(lspackova@redhat.com)

- 改进了支持的构架列表。

0.2-0

Thu Mar 11 2021, Lucie Maňásková (Imanasko@redhat.com)

- 更新了新功能部分 (文件系统和存储)。

0.1-9

Thu Feb 25 2021, Lenka Špačková(lspackova@redhat.com)

- 修复了 CentOS Linux 名称。

0.1-8

Wed Feb 10 2021, Lucie Maňásková (Imanasko@redhat.com)

- 添加了一个已知问题（虚拟化）。

0.1-7

Thu Jan 28 2021, Lucie Maňásková (Imanasko@redhat.com)

- 更新了技术预览章节。

0.1-6

Thu Dec 10 2020, Lenka Špačková (lspackova@redhat.com)

- 向新功能（身份管理）中添加了有关在 SSSD 中处理 AD GPO 的信息。

0.1-5

Tue Dec 01 2020, Lucie Maňásková (Imanasko@redhat.com)

- 添加了对 fapolicyd(Security)问题的错误修复。
- 添加了一个已知问题(Installer)。

0.1-4

Tue Nov 24 2020, Lucie Maňásková(Imanasko@redhat.com)

- 更新了新功能部分（网络）。

0.1-3

2020 年 10 月 30 日星期五, Lenka Špačková (lspackova@redhat.com)

- 更新了 Repositories 部分中的 Application Streams 描述。

0.1-2

周一 05 2020, Lucie Maňásková(Imanasko@redhat.com)

- 添加了程序错误修复（网络）。

0.1-1

Tue Sep 29 2020, Lenka Špačková (lspackova@redhat.com)

- 使用 RHEL 7.9 发行版本更新了原位升级路径。

0.1-0

Thu Aug 27 2020, Lucie Maňásková (Imanasko@redhat.com)

- 添加了程序错误修复(Kernel)。

0.0-9

Mon Aug 10 2020, Lucie Maňásková (Imanasko@redhat.com)

- 添加了一个已知问题（身份管理）。

0.0-8

Tue Jul 21 2020, Lucie Maňásková (Imanasko@redhat.com)

- 发行 Red Hat Enterprise Linux 8.2.1 发行注记。

0.0-7

Thu Jul 16 2020, Lucie Maňásková (Imanasko@redhat.com)

- 添加了技术预览（网络）。
- 更新了新功能部分。

0.0-6

Thu Jun 25 2020, Jaroslav Klech (jklech@redhat.com)

- 升级内核参数章节。
- 在设备驱动程序章节中添加了各种改进。

0.0-5

Fri Jun 19 2020, Lucie Maňásková (Imanasko@redhat.com)

- 添加了新的已知问题。
- 对其他发行注记进行的一些更新。

0.0-4

Thu Jun 04 2020, Lucie Maňásková (Imanasko@redhat.com)

- 更新了新功能部分。
- 添加了一个已知问题（容器）。

0.0-3

2020 年 5 月 20 日星期三, Lenka Špačková (lspackova@redhat.com)

- 添加了一个已知问题（Dynamic 编程语言、Web 和数据库服务器）。
- 添加了程序错误修复（Compilers 和开发工具）。
- 对其他发行注记进行的一些更新。

0.0-2

Tue Apr 28 2020, Lucie Maňásková (Imanasko@redhat.com)

- 发行 Red Hat Enterprise Linux 8.2 发行注记。

0.0-1

Mon Mar 09 2020, Jaroslav Klech (jklech@redhat.com)

- 提供了对外部内核参数和新驱动程序的重要更改章节。

0.0-0

Tue Jan 21 2020, Lucie Maňásková (Imanasko@redhat.com)

- 发布 Red Hat Enterprise Linux 8.2 Beta 发行注记。

