



Red Hat Enterprise Linux 8

8.7 发行注记

Red Hat Enterprise Linux 8.7 发行注记

Red Hat Enterprise Linux 8 8.7 发行注记

Red Hat Enterprise Linux 8.7 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 8.7 中已实现的改进和附加组件的高级信息，并记录了本版本中已知的问题，以及重要的 bug 修复、技术预览、已弃用的功能和其他详情。有关安装 Red Hat Enterprise Linux 的详情，请参考 Installation。

目录

对红帽文档提供反馈	5
第 1 章 概述	6
1.1. RHEL 8.7 中的主要变化	6
1.2. 原位升级和操作系统转换	8
1.3. 红帽客户门户网站 LABS	9
1.4. 其他资源	10
第 2 章 构架	11
第 3 章 RHEL 8 中的内容发布	12
3.1. 安装	12
3.2. 软件仓库	12
3.3. 应用程序流	12
3.4. 使用 YUM/DNF 的软件包管理	13
第 4 章 新功能	14
4.1. 安装程序和镜像创建	14
4.2. SHELL 和命令行工具	14
4.3. 基础设施服务	17
4.4. 安全	18
4.5. 网络	20
4.6. 内核	21
4.7. 引导加载程序	22
4.8. 文件系统和存储	23
4.9. 高可用性和集群	23
4.10. 动态编程语言、网页和数据库服务器	24
4.11. 编译器和开发工具	27
4.12. 身份管理	35
4.13. 图形基础结构	37
4.14. WEB 控制台	38
4.15. RED HAT ENTERPRISE LINUX 系统角色	39
4.16. 虚拟化	43
4.17. 云环境中的 RHEL	44
4.18. 容器	45
第 5 章 对外部内核参数的重要更改	47
新内核参数	47
更新的内核参数	48
第 6 章 设备驱动程序	51
6.1. 新驱动程序	51
6.2. 更新的驱动程序	52
第 7 章 可用的 BPF 功能	53
第 8 章 程序错误修复	67
8.1. 安装程序和镜像创建	67
8.2. 软件管理	67
8.3. SHELL 和命令行工具	67
8.4. 基础设施服务	69
8.5. 安全	69
8.6. 网络	70

8.7. 内核	71
8.8. 引导加载程序	71
8.9. 高可用性和集群	72
8.10. 动态编程语言、网页和数据库服务器	72
8.11. 编译器和开发工具	72
8.12. 身份管理	73
8.13. DESKTOP	74
8.14. 图形基础结构	74
8.15. WEB 控制台	75
8.16. RED HAT ENTERPRISE LINUX 系统角色	75
8.17. 虚拟化	77
8.18. 云环境中的 RHEL	77
8.19. 容器	78
第 9 章 技术预览	79
9.1. SHELL 和命令行工具	79
9.2. 网络	79
9.3. 内核	80
9.4. 文件系统和存储	81
9.5. 高可用性和集群	83
9.6. 身份管理	84
9.7. DESKTOP	86
9.8. 图形基础结构	87
9.9. 虚拟化	87
9.10. 云环境中的 RHEL	88
9.11. 容器	88
第 10 章 过时的功能	90
10.1. 安装程序和镜像创建	90
10.2. 软件管理	91
10.3. SHELL 和命令行工具	91
10.4. 安全	92
10.5. 网络	94
10.6. 内核	95
10.7. 引导加载程序	96
10.8. 文件系统和存储	96
10.9. 高可用性和集群	97
10.10. 动态编程语言、网页和数据库服务器	98
10.11. 编译器和开发工具	98
10.12. 身份管理	98
10.13. DESKTOP	101
10.14. 图形基础结构	101
10.15. WEB 控制台	102
10.16. RED HAT ENTERPRISE LINUX 系统角色	102
10.17. 虚拟化	103
10.18. 容器	104
10.19. 已弃用的软件包	104
10.20. 弃用和不维护的设备	141
第 11 章 已知问题	145
11.1. 安装程序和镜像创建	145
11.2. 订阅管理	147
11.3. 软件管理	147
11.4. SHELL 和命令行工具	148

11.5. 基础设施服务	148
11.6. 安全	149
11.7. 网络	154
11.8. 内核	155
11.9. 引导加载程序	160
11.10. 文件系统和存储	160
11.11. 动态编程语言、网页和数据库服务器	162
11.12. 身份管理	163
11.13. DESKTOP	166
11.14. 图形基础结构	167
11.15. WEB 控制台	168
11.16. RED HAT ENTERPRISE LINUX 系统角色	168
11.17. 虚拟化	169
11.18. 云环境中的 RHEL	172
11.19. 支持性	174
11.20. 容器	174
第 12 章 国际化	176
12.1. RED HAT ENTERPRISE LINUX 8 国际语言	176
12.2. RHEL 8 国际化的显著变化	176
附录 A. 按组件划分的问题单列表	177
附录 B. 修改历史记录	185

对红帽文档提供反馈

我们感谢您对我们文档的反馈。让我们了解如何改进它。

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。包括文档相关部分的链接。
5. 点对话框底部的 **Create**。

第 1 章 概述

1.1. RHEL 8.7 中的主要变化

安装程序和镜像创建

以下是 RHEL 8.7 GA 中的镜像构建器关键亮点：

- 内部的镜像构建器现在支持：
 - 将镜像上传到 GCP
 - 自定义 `/boot` 分区
 - 将容器镜像直接推送到注册中心
 - 用户现在可以在镜像创建过程中自定义其蓝图

更多信息请参阅 [第 4.1 节“安装程序和镜像创建”](#)。

安全

现在，**scap-security-guide** (SSG) 软件包中 Red Hat Enterprise Linux 8 的 DISA STIG 配置文件与 DISA 的内容很好地保持一致。这会导致在 SSG 补救后对 DISA 内容较少的发现。

scap-security-guide (SSG) 软件包中的互联网安全中心(CIS)配置文件现在与 CIS Red Hat Enterprise Linux 8 Benchmark 版本 2.0.0 保持一致。这个基准版本添加了新的要求，删除了不再相关的要求，并对一些现有的要求进行了重新排序。更新会影响相关规则中的引用，以及对应配置文件的准确性。

在系统配置和 **clevis-luks-systemd** 子软件包中的更改使 Clevis 加密客户端也能够解锁在引导过程后期挂载的 LUKS 加密的卷，而无需在部署过程中使用 **systemctl enable clevis-luks-askpass.path** 命令。

如需更多信息，请参阅 [新特性 - 安全](#)。

Shell 和命令行工具

RHEL 8.7 引入了一个新的软件包 **xmlstarlet**。有了 XMLStarlet，您可以解析、转换、查询、验证和编辑 XML 文件。

RHEL 8.7 中更新了以下命令行工具：

- **opencryptoki** 更新到版本 3.18.0
- **powerpc-utils** 更新到版本 1.3.10
- **libva** 更新到版本 2.13.0

如需更多信息，请参阅 [新功能 - Shell 和命令行工具](#)

基础架构服务

RHEL 8.7 中更新了以下基础设施服务工具：

- **chrony** 更新到版本 4.2
- **unbound** 更新到版本 1.16.2

如需更多信息，请参阅 [新功能 - 基础设施服务](#)。

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：

- Ruby 3.1
- Mercurial 6.2
- Node.js 18

另外，Redis 6 已升级到版本 6.2.7。

如需更多信息，请参阅 [新特性 - 动态编程语言、Web 和数据库服务器](#)。

编译器和开发工具

更新了性能工具和调试器

RHEL 8.7 中更新了以下性能工具和调试器：

- Valgrind 3.19
- SystemTap 4.7
- Dyninst 12.1.0
- elfutils 0.187

更新了性能监控工具

RHEL 8.7 中更新了以下性能监控工具：

- PCP 5.3.7
- Grafana 7.5.13

更新了编译器工具集

RHEL 8.7: 中更新了以下编译器工具集：

- GCC Toolset 12
- LLVM Toolset 14.0.6
- Rust Toolset 1.62
- Go Toolset 1.18

如需更多信息，请参阅 [新功能 - 编译器和开发工具](#)。

RHEL 8 中的 Java 实现

RHEL 8 AppStream 软件仓库包括：

- **java-17-openjdk** 软件包，提供 OpenJDK 17 Java 运行时环境和 OpenJDK 17 Java 软件开发组件。
- **java-11-openjdk** 软件包，提供 OpenJDK 11 Java 运行时环境和 OpenJDK 11 Java 软件开发组件。
- **java-1.8.0-openjdk** 软件包，提供 OpenJDK 8 Java 运行时环境和开源 JDK 8 Java 软件开发组件。

如需更多信息，请参阅 [OpenJDK 文档](#)。

Java 工具

RHEL 8.7 引入了 **Maven 3.8** 作为新的模块流。

如需更多信息，请参阅 [新功能 - 编译器和开发工具](#)。

身份管理

RHEL 8.7 中的身份管理(IdM)引入了技术预览，您可以在其中将用户身份验证委派给支持 OAuth 2 设备授权流的外部身份提供商(IdP)。当这些用户使用 SSSD 进行身份验证，并在外部 IdP 完成验证和授权后，它们会收到具有 Kerberos 票的 RHEL IdM 单点登录功能。

如需更多信息，请参阅 [技术预览 - 身份管理](#)

Red Hat Enterprise Linux System 角色

8.7 RHEL 系统角色中的主要新功能：

- 现在，RHEL 系统角色也可以在禁用事实收集的 playbook 中提供。
- **ha_cluster** 角色现在支持 SBD 隔离、Corosync 设置的配置和捆绑包资源的配置。
- **network** 角色现在为路由规则配置网络设置，支持使用 **nmstate API** 的网络配置，用户可以使用 IPoIB 功能创建连接。
- **microsoft.sql.server** 角色拥有新的变量，如控制高可用性集群的变量、自动管理防火墙端口的变量，或者在受管节点上搜索 **mssql_tls_cert** 和 **mssql_tls_private_key** 值的变量。
- **logging** 角色支持各种新选项，如文件输入的 **startmsg.regex** 和 **endmsg.regex**，或 **template**、**severity** 和 **facility** 选项。
- **storage** 角色现在包括对精简配置的卷的支持，并且默认情况下，该角色的详细程度也降低了。
- **sshd** 角色验证置入目录的 include 指令，现在可以通过 `/etc/ssh/sshd_config` 管理角色。
- **metrics** 角色现在可以导出 postfix 性能数据。
- **postfix** 角色现在有一个覆盖以前配置的新选项。
- 在配置伪装或 `icmp_block_inversion` 时，**firewall** 角色不需要 `state` 参数。在 **firewall** 角色中，您现在可以使用 `absent` 和 `present` 状态添加、更新或删除服务。该角色也可以提供 Ansible 事实，并使用 PCI 设备 ID 向区域添加或删除接口。**firewall** 角色有覆盖之前配置的新选项。
- **selinux** 角色现在包含 **seuser** 和 **selevel** 参数的设置。

1.2. 原位升级和操作系统转换

RHEL 7 原位升级到 RHEL 8

目前可能的原位升级路径是：

- 从 64 位 Intel、IBM POWER 8(little endian)和 IBM Z 架构上的 RHEL 7.9 升级到 RHEL 8.4 和 RHEL 8.6
- 在需要内核版本 4.14 的架构上，从 RHEL 7.6 到 RHEL 8.4：IBM POWER 9(little endian)和 IBM Z(Structure A)。这是这些架构的最终原位升级路径。
- 从 64 位 Intel 架构上带有 SAP HANA 的系统上的 RHEL 7.9 升级到 RHEL 8.2 和 RHEL 8.6。

要确定您的系统在升级到 RHEL 8.6 后仍然被支持，请把系统更新到最新的 RHEL 8.7 版本，或者确保启用了 RHEL 8.6 Extended Update Support (EUS)存储库。

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](#)。

有关在带有 SAP 环境的系统上进行原位升级的说明，请参阅[如何将 SAP 环境从 RHEL 7 原位升级到 RHEL 8](#)。



注意

对于 IBM POWER 9 (little endian)和 IBM Z (structure A)架构的 RHEL 7.6 成功的原位升级，您必须手动下载特定的 Leapp 数据。如需更多信息，请参阅[原位升级的 Leapp 数据快照](#) 知识库文章。

主要改进包括：

- 现在，在具有 Red Hat Update Infrastructure (RHUI)的 Microsoft Azure 上能够进行 SAP Apps 系统原位升级。
- 现在，在具有 Red Hat Update Infrastructure (RHUI)的 Google Cloud Platform 上能够进行原位升级。

从 RHEL 6 原位升级到 RHEL 8

要从 RHEL 6.10 升级到 RHEL 8，请遵循[从 RHEL 6 升级到 RHEL 8](#) 中的说明。

从 RHEL 8 原位升级到 RHEL 9

文档[从 RHEL 8 升级到 RHEL 9](#) 中提供了有关如何使用 Leapp 工具执行从 RHEL 8 原位升级到 RHEL 9 的说明。RHEL 8 和 RHEL 9 的主要区别记录在[采用 RHEL 9 时的注意事项](#) 中。

从不同的 Linux 发行版转换到 RHEL

如果您使用 CentOS Linux 8 或 Oracle Linux 8，您可以使用红帽支持的 **Convert2RHEL** 程序将操作系统转换为 RHEL 8。如需更多信息，请参阅[从基于 RPM 的 Linux 发行版转换到 RHEL](#)。

如果您使用早期版本的 CentOS Linux 或 Oracle Linux（称为版本 6 或 7），可以将操作系统转换为 RHEL，然后执行 RHEL 8 的原位升级。请注意，CentOS Linux 6 和 Oracle Linux 6 的转换使用不被支持的 **Convert2RHEL** 工具。有关不受支持的转换的更多信息，请参阅[如何执行从派生的 Linux 发行版本到 RHEL 的不受支持的转换](#)。

有关红帽如何支持从其他 Linux 发行本转换到 RHEL 的详情，请参考[Convert2RHEL 支持政策文档](#)。

1.3. 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 Enterprise Linux Upgrade Helper](#)
- [Red Hat Satellite Upgrade Helper](#)

- [Red Hat Code Browser](#)
- [JVM Options Configuration Tool](#)
- [Red Hat CVE Checker](#)
- [Red Hat Product Certificates](#)
- [Load Balancer Configuration Tool](#)
- [Yum Repository Configuration Helper](#)
- [Red Hat Memory Analyzer](#)
- [Kernel Oops Analyzer](#)
- [Red Hat Product Errata Advisory Checker](#)

1.4. 其他资源

- 与其他版本系统相比，Red Hat Enterprise Linux 8 的**能力和限制**可在知识库文章[Red Hat Enterprise Linux 技术能力和限制](#)中获得。
- 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**，包括删除的功能，都记录在 [使用 RHEL 8 时的注意事项](#) 里。
- 有关如何执行 **从 RHEL 7 到 RHEL 8 的原位升级**的说明，请参考 [从 RHEL 7 升级到 RHEL 8](#) 的文档。
- **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](#)。

第 2 章 构架

Red Hat Enterprise Linux 8.7 与内核版本 4.18.0-425 一起发行，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

安装 ISO 镜像的大小为几个 GB，因此可能不适用于光盘介质格式。当使用安装 ISO 镜像时，建议使用 USB 盘或 USB 硬盘驱动器创建可引导安装介质。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 安装的说明，请参阅 [执行标准的 RHEL 8 安装](#) 文档。有关自动 Kickstart 安装和其他高级主题，请参阅 [执行高级的 RHEL 8 安装](#) 文档。

有关在基础 RHEL 安装中由 RPM 创建的用户和组的列表，以及获取此列表的步骤，请查看 [基本 RHEL 安装中的所有用户和组是什么？](#) 知识库文章。

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。应用程序流以熟悉的 RPM 格式，作为 RPM 格式的扩展提供，称为 *模块*，或作为软件集合。有关 AppStream 中提供的软件包列表，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。现在，用户空间组件的多个版本的发布和更新频率会比核心操作系统的发布和更新频率快。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）在带有默认的 **postgresql:10** 流的 **postgresql** 模块中提供。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 中可用模块的列表，请查看 [软件包清单](#)。

3.4. 使用 YUM/DNF 的软件包管理

在 Red Hat Enterprise Linux 8 上，安装软件是由 YUM 工具来保证的，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 主版本保持一致。但是，如果您输入 **dnf** 而不是 **yum**，则命令可以按预期工作，因为 **yum** 是 **dnf** 的别名，以便兼容。

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 新功能

这部分论述了 Red Hat Enterprise Linux 8.7 中引入的新功能和主要改进。

4.1. 安装程序和镜像创建

安装程序中支持自动 FCP SCSI LUN 扫描

当在 IBM Z 系统上附加 FCP SCSI LUN 时，安装程序现在可以使用自动 LUN 扫描。如果没有通过 `zfcplib.allow_lun_scan` 内核模块参数禁用，自动 LUN 扫描可用于在 NPIV 模式下对 FCP 设备的操作。它会被默认启用。它提供了对附加到具有指定的设备总线 ID 的 FCP 设备的存储区域网络中发现的所有 SCSI 设备的访问。不需要指定 WWPN 和 FCP LUN，只提供 FCP 设备总线 ID 就足够了。

(BZ#1497089)

内部镜像构建器现在支持 /boot 分区自定义

内部镜像构建器版本现在支持使用自定义的 `/boot` 挂载点分区大小构建镜像。您可以在蓝图自定义中指定 `/boot` 挂载点分区的大小，以便在默认引导分区太小时增加 `/boot` 分区的大小。例如：

```
[[customizations.filesystem]]
mountpoint = "/boot"
size = "20 GiB"
```

(JIRA:RHELPLAN-130379)

内部镜像构建器现在支持将镜像上传至 GCP

有了这个增强，您可以使用镜像构建器 CLI 来构建 `gce` 镜像，为您要上传镜像的用户或服务帐户提供凭证。因此，镜像构建器会创建镜像，然后将 `gce` 镜像直接上传到您指定的 GCP 环境中。

(BZ#2049492)

内部镜像构建器 CLI 支持将容器镜像直接推送到注册中心

有了这个增强，您可以使用镜像构建器 CLI 将 Edge 的 RHEL 容器镜像直接推送到容器注册中心。要构建容器镜像：

1. 设置上传供应商，并可以选择添加凭证。
2. 构建容器镜像，将容器注册中心和存储库作为参数传给 `composer-cli`。镜像就绪后，它在您设置的容器注册中心中提供。

(JIRA:RHELPLAN-130376)

镜像构建器内部用户现在可以在镜像创建过程中自定义其蓝图

有了这个更新，`Edit Blueprint` 页面已被删除，来统一镜像构建器服务中和 `cockpit-composer` 中镜像构建器应用程序的用户体验。现在，用户可以在镜像创建过程中创建他们的蓝图并添加他们的自定义，如添加软件包和创建用户。蓝图版本也已被删除，这样蓝图只有一个版本：当前版本。用户能够通过其已创建的镜像访问旧的蓝图版本。

(JIRA:RHELPLAN-122735)

4.2. SHELL 和命令行工具

Cronie 添加了对所选范围内随机时间的支持

Cronie 工具现在支持 cronjob 执行的 ~（范围内随机）运算符。因此，您可以在所选范围内的随机时间启动 cronjob。

([BZ#1832510](#))

新软件包：xmlstarlet

XMLStarlet 是一组用于解析、转换、查询、验证和编辑 XML 文件的命令行工具。新的 **xmlstarlet** 软件包提供了一组简单的 shell 命令，您可以像使用 UNIX 命令处理纯文本文件（如 **grep**、**sed**、**awk**、**diff**、**patch**、**join** 等）那样使用它们。

([BZ#1882020](#))

ReaR 添加了在恢复前后执行命令的新变量

有了这个增强，ReaR 引进了两个新变量，用于在恢复前后轻松地自动化要执行的命令：

- **PRE_RECOVERY_COMMANDS** 接受一个命令数组。将在恢复开始前执行这些命令。
- **POST_RECOVERY_COMMANDS** 接受一个命令数组。将在恢复完成后执行这些命令。

这些变量是 **PRE_RECOVERY_SCRIPT** 和 **POST_RECOVERY_SCRIPT** 的替代方案，其有以下不同：

- 早期的 **PRE_RECOVERY_SCRIPT** 和 **POST_RECOVERY_SCRIPT** 变量接受单个 shell 命令。要将多个命令传递给这些变量，您必须使用分号将命令分开。
- 新的 **PRE_RECOVERY_COMMANDS** 和 **POST_RECOVERY_COMMANDS** 变量接受命令数组，数组的每个元素都会作为单独的命令执行。

因此，在恢复前后，在救援系统中提供要执行的多个命令现在更为简单且更不容易出错。

如需更多信息，请参阅 **default.conf** 文件。

([BZ#2035872](#))

libva rebase 到版本 2.13.0

视频加速 API 的 **libva** 库已更新至版本 2.13.0。主要的改进和新功能包括：

- 两个新的 FourCC 视频编码格式：**X2R10G10B10** 和 **X2B10G10R10** 用于捕获、处理以及以 10 位 RGB 格式显示视频（不包括 Alpha）。
- **iris** 和 **crocus** DRI 驱动程序的 VA-API 驱动程序映射。
- 输出缓冲区同步的 **vaSyncBuffer** 功能。
- 用于复制面和缓冲区的 **vaCopy** 接口。
- 用于数字权限管理(DRM)保护的视频的 LibVA Protected Content API。
- 视频处理中的 3DLUT Filter 将输入颜色映射为新的输出值。

([BZ#2099907](#))

PowerPC-utils rebase 到版本 1.3.10

为 PowerPC 平台提供了各种工具的 **powerpc-utils** 软件包已更新至版本 1.3.10。主要改进包括：

- 添加了为 **ppc64_cpu** 工具中的能源和频率解析 Power 架构平台引用(PAPR)信息的能力。
- 改进了 **lparstat** 工具，来在 max 配置系统上 **lparstat -E** 命令失败时显示增强的错误消息。**lparstat** 命令报告逻辑分区有关的信息。
- 修复了 **lparstat** 命令中以传统格式报告的在线内存。
- 添加了对 **acc** 命令的支持，来动态更改 NX GZIP 加速器的服务信誉质量(QoS)。
- 添加了在 **printf()** 和 **sprintf()** 调用中格式化指定符的改进。
- **hcnmgr** 工具为混合虚拟网络提供了 HMC 工具，包括以下改进：
 - 在混合网络虚拟化 **HNV FEATURE** 列表中添加了 **wicked** 功能。**hcnmgr** 工具支持 wicked 混合网络虚拟化(HNV)使用 **wicked** 功能进行绑定。
 - **hcnmgr** 维护一个 **hcnid** 状态，以便稍后进行清理。
 - **hcnmgr** 排除了 NetworkManager (NM) **nmcli** 代码。
 - NM HNV 主从 设置已被修复。
 - **hcnmgr** 支持将虚拟网络接口控制器(vNIC)用作备份设备。
- 修复了 **bootlist** 中无效的十六进制编号系统消息。
- **kpartx** 工具中包含的 **-l** 标志作为 **bootlist** 命令中的 **-p** 分隔符值。
- 在 **sslot** 工具中添加了修复以防止在列出 IO 插槽时的内存泄漏。
- 在 **lsslot** 工具中为最新的外围设备组件互连快速(PCIe)插槽类型添加了 DRC 类型描述字符串。
- 修复了 **errinjct** 工具中 RTAS 的无效配置地址。
- 添加了对 **ofpathname** 工具中结构设备上非易失性内存(NVMf)的支持。此工具提供了一种将逻辑设备名称转换为开放固件设备路径的机制，反之亦然。
- 添加了对 **ofpathname** 工具中非对称名字空间访问 (ANA) 模式下非易失性内存(NVMe)支持的修复。
- 将 **smt.state** 文件作为配置文件安装。

(BZ#2051330)

opencryptoki rebase 到版本 3.18.0

opencryptoki 软件包，它是公钥加密标准(PKCS)#11 的实现，已更新到版本 3.18.0。主要改进包括：

- 默认为符合联邦信息处理标准 (FIPS) 的令牌数据格式 (tokversion=3.12)。
- 添加了对带有全局策略的限制机制和密钥的使用的支持。
- 添加了对机制使用统计计数的支持。
- **ICA/EP11** 令牌现在支持 **libica** 库版本 4。
- **p11sak** 工具可以为公钥和私钥设置不同的属性。

- `C_GetMechanismList` 不会在 EP11 令牌中返回 `CKR_BUFFER_TOO_SMALL`。

`opencryptoki` 支持两个不同的令牌数据格式：

- 较早的数据格式，它使用非FIPS批准的算法（如 DES 和 SHA1）
- 新数据格式，只使用 FIPS 批准的算法。

旧数据格式不再可以正常工作，因为 FIPS 供应商只允许使用 FIPS 批准的算法。



重要

要使 `penCryptoki` 在 RHEL 8 上正常工作，请迁移令牌以便在系统上启用 FIPS 模式之前使用新的数据格式。这是必要的，因为旧数据格式仍然是 `openCryptoki 3.17` 中的默认设置。当系统改为启用 FIPS 时，使用旧令牌数据格式的现有 `openCryptoki` 安装将无法正常工作。

您可以使用 `pkcstok_migrate` 程序（由 `openCryptoki` 提供），将令牌迁移为使用新的数据格式。请注意，`pkcstok_migrate` 在迁移过程中使用非FIPS批准的算法。因此，在系统中启用 FIPS 模式前使用这个工具。如需更多信息，请参阅[迁移到 FIPS 合规性 - pkcstok_migrate 工具程序](#)。

(BZ#2043845)

Redfish 模块现在是 `redhat.rhel_mgmt` Ansible 集合的一部分

`redhat.rhel_mgmt` Ansible 集合现在包含以下模块：

- `redfish_info`
- `redfish_command`
- `redfish_config`

因此，用户可以从管理自动化中受益，通过使用 Redfish 模块来检索服务器健康状态，获得有关硬件和固件清单的信息、执行电源管理、更改 BIOS 设置、配置 Out-Of-Band (OOB) 控制器、配置硬件 RAID，并执行固件更新。

(BZ#2112435)

`sysctl` 现在与 `systemd` 目录顺序匹配

`sysctl` 工具的配置目录顺序现在与 `systemd-sysctl` 目录顺序同步。配置目录在运行时检查并更改内核参数。`/etc/sysctl.d` 目录中的配置文件的优先级高于 `/run/sysctl.d` 中的配置文件，且在 `sysctl` 和 `systemd` 之间文件的优先级不再发生中断。

(BZ#2111915)

4.3. 基础设施服务

`chrony` rebase 到版本 4.2

`chrony` 套件已更新至版本 4.2。与版本 4.1 相比，主要的改进包括：

- 服务器交错模式已得到改进，更加可靠，并支持单一地址转换器（网络地址转换 - NAT）后的多个客户端。

- 添加了对网络时间协议版本 4 (NTPv4)扩展字段的实验性支持，以提高时间同步的稳定性和估计错误的精度。您可以使用 **extfield F323** 选项启用此字段，这可以扩展协议 NTPv4 的功能。
- 添加了精度时间协议 (PTP)上 NTP 转发的实验性支持，以便在对 PTP 数据包有时间戳限制的网络接口卡(NIC)上启用完整的硬件时间戳。您可以使用 **ptpport 319** 指令来启用 PTP 上的 NTP。

(BZ#2062356)

unbound rebase 到版本 1.16.2

unbound 组件已更新至版本 1.16.2。**unbound** 是一种验证、递归和缓存 DNS 解析器。主要改进包括：

- 有了带有 **RFC 8976** 的 ZONEMD 区域验的支持，接收者现在可以验证数据完整性和来源真实性的区域内容。
- 有了 **unbound**，您现在可以配置持久性 TCP 连接。
- 根据 DNS **draft-ietf-dnsop-svcb-https** 文档中的服务绑定和参数规范添加了 SVCB 和 HTTPS 类型及处理。
- **unbound** 从加密策略获取默认的 TLS 密码。
- 您可以根据 **RFC8375** 使用特殊用途域 **home.arpa**。这个域被指定为在住宅家庭网络中非唯一使用。
- **unbound** 现在支持为 stub 或转发区域启用 **tcp-upstream** 查询选择。
- 现在，**aggressive-nsec** 选项默认为 **yes**。
- **ratelimit** 逻辑已更新。
- 当查询被 Unbound 响应策略区域(RPZ) nxdomain 回复阻止时，您可以使用新的 **rpz-signal-nxdomain-ra** 选项来取消 **RA** 标志设置。
- 根据 **RFC8914**，有了对扩展 DNS 错误(EDE)的基本支持，您可以从其他错误信息中受益。

(BZ#2027735)

4.4. 安全

NSS 不再支持少于 1023 位的 RSA 密钥

网络安全服务(NSS)库的更新将所有 RSA 操作的最小密钥大小从 128 改为 1023 位。这意味着 NSS 不再执行以下功能：

- 生成小于 1023 位的 RSA 密钥。
- 使用小于 1023 位的 RSA 密钥进行签名或验证 RSA 签名。
- 使用小于 1023 位的 RSA 密钥的加密或解密值。

(BZ#2097837)

SCAP 安全指南 rebase 到 0.1.63

SCAP 安全指南(SSG)软件包已更新到上游版本 0.1.63。此版本提供各种改进和程序错误修复，最重要的是：

- 添加了 **sysctl**、**grub2**、**pam_pwquality** 的新合规性规则以及构建时间内核配置。
- 强化 PAM 堆栈的规则现在使用 **authselect** 作为配置工具。注意：如果以其他方式编辑 PAM 堆栈，则通过此修改，将不会应用强化 PAM 堆栈的规则。

(BZ#2070564)

SSG CIS 配置文件与 CIS RHEL 8 基准 2.0.0 保持一致

SCAP 安全指南(SSG)现在包含将互联网安全中心(CIS)配置文件与 CIS Red Hat Enterprise Linux 8 基准版本 2.0.0 保持一致的更改。这个基准版本添加了新的要求，删除了不再相关的要求，并对一些现有的要求进行了重新排序。更新会影响相关规则中的引用，以及对应配置文件的准确性。

(BZ#2058203)

RHEL 8 STIG 配置文件现在与 DISA STIG 内容保持一致

包括在 **scap-security-guide** (SSG)软件包中的 Red Hat Enterprise Linux 8 的 DISA STIG 配置文件 (**xccdf_org.ssgproject.content_profile_stig**) 可用于根据国防部信息系统(DISA)制定的安全技术实施指南(STIG)来评估系统。您可以使用 SSG 中的内容修复您的系统，但您可能需要使用 DISA STIG 自动化内容来评估您的系统。有了这个更新，DISA STIG RHEL 8 配置文件与 DISA 的内容更加一致。这会导致 SSG 修复后对 DISA 内容的发现较少。

请注意，以下规则的评估仍然有分歧：

- SV-230264r627750_rule - CCE-80790-9 (**ensure_gpgcheck_globally_activated**)
- SV-230349r833388_rule - CCE-82266-8 (**configure_bashrc_exec_tmux**)
- SV-230311r833305_rule - CCE-82215-5 (**sysctl_kernel_core_pattern**)
- SV-230546r833361_rule - CCE-80953-3 (**sysctl_kernel_yama_ptrace_scope**)
- SV-230287r743951_rule - CCE-82424-3 (**file_permissions_sshd_private_key**)
- SV-230364r627750_rule - CCE-82472-2 (**accounts_password_set_min_life_existing**)
- SV-230343r743981_rule - CCE-86107-0 (**account_passwords_pam_faillock_audit**)

(BZ#1967947)

如果 /tmp 和 /var/tmp 分区不存在，则挂载选项的 SSG 规则不再失败

在以前的版本中，如果系统上不存在此类分区，则 /tmp 和 /var/tmp 分区的挂载选项的 SCAP 安全指南 (SSG)规则会错误地报告 **失败** 结果。

此增强使这些规则不适用，而不是失败。现在，只有在分区存在且系统没有正确的挂载选项时，规则才会失败。

如果这些挂载选项对于特定的策略至关重要，则那么配置文件中应该存在一条规定此类分区存在的规则，而这条规则应该失败。

(BZ#2032403)

STIG 安全配置文件已更新到版本 V1R7

SCAP 安全指南中的 **Red Hat Enterprise Linux 8** 的 **DISA STIG** 配置文件已更新为与最新的 **V1R7** 版本保持一致。

该配置文件更加稳定，并与国防信息系统局(DISA)提供的 RHEL 8 STIG（安全技术实施指南）手动基准保持一致。这个迭代带来了更新，将 **sysctl** 内容与新的 STIG 保持一致。

您应该只使用这个配置文件的当前版本，因为旧版本已不再有效。



警告

自动补救可能会导致系统无法正常工作。先在测试环境中运行补救。

([BZ#2112937](#))

clevis-luks-askpass 现在被默认启用

`/lib/systemd/system-preset/90-default.preset` 文件现在包含 **enable clevis-luks-askpass.path** 配置选项，并且 **clevis-systemd** 子软件包的安装确保 **clevis-luks-askpass.path** 单元文件被启用。这使 Clevis 加密客户端也可解锁在引导过程后期挂载的 LUKS 加密的卷。在此次更新之前，管理员必须使用 **systemctl enable clevis-luks-askpass.path** 命令来使 Clevis 解锁此类卷。

([BZ#2107081](#))

为 Rsyslog 错误文件添加了一个最大大小选项

使用新的 **action.errorfile.maxsize** 选项，您可以为 Rsyslog 日志处理系统指定错误文件的最大字节数。当错误文件达到指定大小时，Rsyslog 无法在其中写入额外的错误或其他数据。这可防止错误文件填满文件系统，并使主机不可用。

([BZ#1962318](#))

fapolicyd rebase 到 1.1.3

fapolicyd 软件包已升级到版本 1.1.3。主要改进和 bug 修复包括：

- 规则现在可以包含新的主题 PPID 属性，该属性与主题的父 PID（进程 ID）匹配。
- OpenSSL 库代替了 Libcrypt 库，来作为哈希计算的加密引擎。
- **fagenrules --load** 命令现在可以正常工作。

([BZ#2100087](#))

4.5. 网络

改进了大 iptables 规则集的保存速度

此增强优化了 **iptables-save** 工具，来减少保存大规则集时的开销。当从 `/etc/protocols` 文件中读取条目时，这个工具已被改进，在不需要的情况下，它不再搜索扩展共享对象文件。因此，当您在具有高存储访问延迟的环境中保存大规则集时，**iptables-save** 的运行时间已显著提高。

([BZ#2058444](#))

NetworkManager rebase 到版本 1.40

NetworkManager 软件包升级到上游版本 1.40，与之前的版本相比，它提供了大量的改进和 bug 修复：

- `/run/NetworkManager/devices/` 目录中的设备状态文件现在有新的部分, `[dhcp4]` 和 `[dhcp6]`, 它们包含当前租期的 DHCP 选项。
- NetworkManager 支持在连接的 `ipv6.mtu` 属性中设置 IPv6 最大传输单元(MTU)。
- NetworkManager 使用 `nm.debug` 内核命令行选项来启用 debug 日志。
- 改进了载体检测。
- 现在, 如果设备上的 MAC 地址改变了, NetworkManager 会为连接重启 DHCP 客户端。
- wifi 热点现在使用稳定的随机通道号, 除非您选择了一个特定的通道。
- 如果将 `wifi.key-mgmt` 属性设为 `wpa-psk`, NetworkManager 现在会禁用 Wi-Fi Protected Access 3 (WPA3)转换模式, 且网络接口不支持受保护的管理帧(PMF)。这个转换模式会在这个场景中的某些设置中造成问题。要明确启用 WPA3 转换模式, 请将 `wifi.key-mgmt` 设为 `sae`。
- NetworkManager 现在将从 DHCP 服务器接收的过长的主机名缩短为第一个点或 64 个字符。

有关显著变化的更多信息, 请参阅 [上游发行注记](#)。

([BZ#2063109](#))

cloud-init 每次在 Microsoft Azure 上引导时都更新网络配置

当管理员在虚拟机离线时更新网络接口配置时, Microsoft Azure 不会更改实例 ID。有了这个增强, `cloud-init` 服务始终会在虚拟机引导时更新网络配置, 以确保 Microsoft Azure 上的 RHEL 使用最新的网络设置。

因此, 如果您在接口上手动配置设置, 如额外的搜索域, `cloud-init` 也许会在重启虚拟机时覆盖它们。有关详情和临时解决方法, 请参阅 [cloud-init-22.1-5 在每次引导时更新网络配置](#) 解决方案。

([BZ#2144898](#))

NetworkManger 现在将 DHCP 租期信息存储在 `/run/NetworkManager/devices/` 目录中

NetworkManager 现在将 DHCP 服务器的租期信息保存在 `/run/NetworkManager/devices/` 目录中。在以前的版本中, 基于文件的 API 不可用, 且此信息仅在 `nmcli -f all devices show DEVICE` 命令的输出中可见。有了这个增强, 其他工具和脚本可以访问 DHCP 选项, 而无需调用 `nmcli`。

([BZ#1943153](#))

4.6. 内核

RHEL 8.7 中的内核版本

Red Hat Enterprise Linux 8.7 与内核版本 4.18.0-425 一起发行。

([BZ#2125545](#))

SSBD 和 STIBP 的默认缓解方案已被更改

`spec_store_bypass_disable` (SSBD)和 `spectre_v2_user` (STIBP)引导参数的默认缓解已从 `seccomp` 模式改为 `prctl`。有了这个更新, `seccomp` 控制下的容器和应用程序的性能得到了提高。

([BZ#2101938](#))

vmcore 转储文件在 debug 内核变体上正确生成

有了这个更新，当当前内核是 debug 变体时，**kdump** 机制现在使用与捕获内核相同的非调试内核版本。通过将非调试内核用作捕获内核，**kdump** 消耗的内存比 debug 变体要少。因此，**kdump** 可以正确地生成 **vmcore** 文件，并捕获崩溃内核的内存内容。

(BZ#2006000)

Intel E800 设备现在支持 iWARP 和 RoCE 协议

有了这个增强，现在您可以使用 **enable_iwarp** 和 **enable_roce** devlink 参数打开和关闭 iWARP 或 RoCE 协议支持。有了这个强制功能，您可以使用其中一个协议配置设备。Intel E800 设备不同时支持同一端口上的这两个协议。

要为特定 E800 设备启用或禁用 iWARP 协议，首先获取卡的 PCI 位置：

```
$ lspci | awk '/E810/ {print $1}'
44:00.0
44:00.1
$
```

然后，启用或禁用协议。您可以将 **pci/0000:44:00.0** 用作第一个端口，将 **pci/0000:44:00.1** 用于卡的第二个端口，来作为 devlink 命令的参数

```
$ devlink dev param set pci/0000:44:00.0 name enable_iwarp value true cmode runtime
$ devlink dev param set pci/0000:44:00.0 name enable_iwarp value false cmode runtime
```

要为特定 E800 设备启用或禁用 RoCE 协议，请获取卡的 PCI 位置，如上所示。然后使用以下命令之一：

```
$ devlink dev param set pci/0000:44:00.0 name enable_roce value true cmode runtime
$ devlink dev param set pci/0000:44:00.0 name enable_roce value false cmode runtime
```

(BZ#2096127)

4.7. 引导加载程序

GRUB 由新密钥签名

由于安全原因，GRUB 现在使用新密钥签名。因此，如果您在启用了安全引导功能的 IBM POWER 的 little-endian 变体上使用 RHEL，您必须将固件更新至 FW1010.30（或更新版本）或 FW1020 才能引导。

(BZ#2074762)

在 IBM POWER 上启动虚拟机时可配置的磁盘访问重试

现在，当逻辑分区(**lpar**)虚拟机(VM)在 IBM POWER 构架上引导时，您可以配置 GRUB 引导装载程序重试访问远程磁盘的次数。降低重试次数可防止某些情况下引导较慢。

在以前的版本中，当磁盘访问在引导失败时，GRUB 重试访问磁盘 20 次。如果您在连接到慢的存储区域网络(SAN)磁盘的 **lpar** 系统上执行 Live Partition Mobility (LPM)迁移，这会导致问题。因此，启动可能在系统上花费非常长，直到 20 次重试完成为止。

有了这个更新，您可以使用 **ofdisk_retries** GRUB 选项配置并减少磁盘访问重试次数。详情请参阅 [在 IBM POWER 上引导 VM 时配置磁盘访问重试](#)。

因此，在 POWER 上的 LPM 后，**lpar** 引导不再慢，**lpar** 系统可以在没有失败的磁盘的情况下引导。

([BZ#2070347](#))

4.8. 文件系统和存储

nfsrahead 已添加到 RHEL 8 中

随着 **nfsrahead** 工具的引入，您可以使用它来修改 NFS 挂载的 **readahead** 值，因此会影响 NFS 的读性能。

([BZ#1946283](#))

rpcctl 命令现在显示 SunRPC 连接信息

有了这个更新，您可以使用 **rpcctl** 命令来显示 SunRPC **sysfs** 文件中收集的有关系的 SunRPC 对象的信息。您可以通过 **sysfs** 文件系统显示、删除和设置 SunRPC 网络层中的对象。

([BZ#2087187](#))

multipath.conf 现在可以在 DM 多路径中包括特定于协议的配置覆盖

您可以通过各种协议访问多路径设备的路径。由于各种协议可以有各种优化配置，因此在没有 **per-protocol** 选项的情况下，无法为设备映射器多路径功能中的所有协议设置最佳配置。有了这个增强，您可以在 **multipath.conf** 文件中包含特定于协议的配置覆盖。现在，您可以在每协议的基础上配置多路径设备路径，允许通过多协议访问多路径设备的正确配置。

([BZ#2065477](#))

multipathd 现在支持检测 FPIN-Li 事件

当您为 **marginal_pathgroups** 配置选项添加新值 **fpin** 时，您可以启用 **multipathd** 来监控 Link Integrity Fabric Performance Impact 通知(PFIN-Li)事件，并将带有链路完整性问题的路径移到边缘 pathgroup。设置 **fpin** 值后，**multipathd** 会覆盖其现有的边缘路径检测方法，并依赖光纤通道光纤来识别链路完整性问题。

有了这个增强，**multipathd** 方法在可发出 PFIN-Li 事件的光纤通道光纤上检测边缘路径变得更为强壮。

([BZ#2083077](#))

4.9. 高可用性和集群

pcs 命令行支持更新多路径 SCSI 设备，而无需系统重启

现在，您可以使用 **pcs stonith update-scsi-devices** 命令更新多路径 SCSI 设备。这个命令更新 SCSI 设备，而不会导致运行在同一节点上其他集群资源的重启。

([BZ#2023845](#))

支持集群 UUID

在集群设置过程中，**pcs** 命令现在会为每个集群生成一个 UUID。因为集群名称不是一个唯一的集群标识符，因此您可以在管理多个集群时使用集群 UUID 来识别具有相同名称的集群。

您可以使用 **pcs cluster config [show]** 命令来显示当前集群的 UUID。您可以使用 **pcs cluster config uuid generate** 命令来向现有集群添加一个 UUID 到或重新生成一个 UUID（如果其 UUID 已存在）。

([BZ#1950551](#))

现在，**multi-active** 资源参数接受 **stop_unexpected** 的值

当资源在其不应该在的多个节点上处于活跃状态时，**multiple-active** 资源参数决定了恢复行为。默认情况下，这种情况需要资源的全面重启，即使资源在其应该运行的地方在成功运行。有了这个更新，**multiple-active** 资源参数接受 **stop_unexpected** 的值，它允许您指定只停止多活动资源的意外实例。用户负责验证服务及其资源代理是否可以与额外的活跃实例一起正常工作，而无需全面重启。

([BZ#2036815](#))

新的 **allow-unhealthy-node** Pacemaker 资源 meta-attribute

pacemaker 现在支持 **allow-unhealthy-node** 资源 meta-attribute。当此 meta-attribute 设为 **true** 时，由于降级节点健康状况，资源不会强制关闭节点。当健康资源设置了此属性时，集群可以自动检测节点的健康状态恢复，并将资源移回节点。

([BZ#2059638](#))

支持 Red Hat OpenStack 平台上的高可用性

现在，您可以在 Red Hat OpenStack 平台上配置高可用性集群。为了支持这个功能，红帽提供了以下新的集群代理：

- **fence_openstack**: OpenStack 上 HA 集群的隔离代理
- **openstack-info**: 配置 **openstack-info** 克隆资源的资源代理，这是 OpenStack 上 HA 集群所需的
- **openstack-virtual-ip**: 配置虚拟 IP 地址资源的资源代理
- **openstack-floating-ip**: 配置浮动 IP 地址资源的资源代理
- **openstack-cinder-volume**: 配置块存储资源的资源代理

([BZ#1182956](#))

pacemaker 现在支持为系统组指定访问控制列表(ACL)

Pacemaker 之前允许为单个用户指定 ACL，但有时更简单，更符合本地策略，来为系统组指定 ACL，并将其应用到该组中的所有用户。**pcs acl group** 命令出现在早期版本中，但没有影响。现在，用户可以使用这个命令为系统组指定 ACL。

([BZ#1724310](#))

新的 **pcs stonith config** 命令选项来显示重新创建配置的隔离设备的 **pcs** 命令

pcs stonith config 命令现在接受 **--output-format=cmd** 选项。指定这个选项会显示您用来在不同系统上重新创建配置的隔离设备的 **pcs** 命令。

([BZ#1909904](#))

新的 **pcs resource config** 命令选项来显示重新创建配置的资源资源的 **pcs** 命令

pcs resource config 命令现在接受 **--output-format=cmd** 选项。指定这个选项会显示用来在不同系统上重新创建配置的资源资源的 **pcs** 命令。

([BZ#1874624](#))

4.10. 动态编程语言、网页和数据库服务器

nodejs:18 模块流现在完全支持

以前作为技术预览提供的 **nodejs:18** 模块流，现在在 [RHSA-2022:8833](#) 公告版中完全支持。**nodejs:18** 模块流现在提供 **Node.js 18.12**，它是一个长期支持(LTS)版本。

RHEL 8.7 中包含的 **Node.js 18** 提供了许多新功能，以及自 RHEL 8.5 以来对 **Node.js 16** 的 bug 和安全修复。

主要变更包括：

- **V8** 引擎已升级到版本 10.2。
- **npm** 软件包管理器已升级到版本 8.18.0。
- **Node.js** 现在提供了一个新的实验性 **fetch** API。
- **Node.js** 现在提供了一个新的实验性 **node:test** 模块，它便于创建以 test Anything Protocol (TAP)格式报告结果的测试。
- **Node.js** 现在首选 IPv6 地址，而不是 IPv4 地址。

要安装 **nodejs:18** 模块流，请使用：

```
# yum module install nodejs:18
```

如果要从 **nodejs:16** 流升级，请参阅 [切换到更新的流](#)。

(BZ#2083073)

nodejs:18 rebase 到版本 18.14， npm rebase 到版本 9

Node.js 18.14 在 [RHSA-2023:1583](#) 中发布，包括 **npm** 从版本 8 升级到版本 9 的 SemVer 主要升级。由于维护原因，这个更新是必需的，可能需要调整 **npm** 配置。

值得注意的是，不限定到特定注册中心的与身份验证相关的设置不再支持。所做的这一变化是出于安全原因。如果您使用了无范围的身份验证配置，则提供的令牌将被发送到 **.npmrc** 文件中列出的每个注册中心。

如果您使用无范围的身份验证令牌，请在 **.npmrc** 文件中生成并提供注册中心范围的令牌。

如果您有使用 **_auth** 的配置行，如 **.npmrc** 文件中的 **//registry.npmjs.org/:_auth**，请将它们替换为 **//registry.npmjs.org/:_authToken=\${NPM_TOKEN}**，并提供您生成的有范围的令牌。

有关更改的完整列表，请参阅 [上游更改日志](#)。

(BZ#2178087)

新模块流：ruby:3.1

RHEL 8.7 在新的 **ruby:3.1** 模块流中引入了 **Ruby 3.1.2**。这个版本提供了很多性能改进、bug和安全修复，以及与 RHEL 8.5 一起发布的 **Ruby 3.0** 的新功能。

主要改进包括：

- **Interactive Ruby (IRB)** 工具现在提供自动完成功能以及文档对话框
- 新的 **debug** gem，它替换了 **lib/debug.rb**，提供了改进的性能，并支持远程调试和多进程/多线程调试

- **error_highlight** gem 现在在 backtrace 中提供精细的错误位置
- 现在可以省略哈希文本数据类型和关键字参数中的值
- pin 运算符(^)现在接受模式匹配中的表达式
- 现在，可以在单行模式匹配省略括号
- YJIT 一种新的实验性进程内实时(JIT)编译器，现在在 AMD 和 Intel 64 位构架上提供
- 已引进了 **TypeProf For IDE** 工具，这是 IDE 中 **Ruby** 代码的实验性静态类型分析工具。

以下性能改进已在基于方法的实时编译器(MJIT)中实现了：

- 对于像 **Rails** 这样的工作负载，默认的最大 JIT 缓存值从 100 增加到 10000
- 当启用了 class 事件的 **TracePoint** 时，使用 JIT 编译的代码将不再被取消

其他显著变化包括：

- **tracer.rb** 文件已被删除
- 自版本 4.0 起，**Psych** YAML 解析器默认使用 **secure_load** 方法

要安装 **ruby:3.1** 模块流，请使用：

```
# yum module install ruby:3.1
```

如果要从以前的 **ruby** 模块流升级，请参阅 [切换到更新的流](#)。

(BZ#2063772)

新模块流：mercurial:6.2

RHEL 8.7 将 **Mercurial 6.2** 添加为新的模块流。从 RHEL 8.0 开始，此版本提供了很多 bug 修复、增强，以及 **Mercurial 4.8** 方面的性能改进。

主要变更包括：

- **Mercurial 6.2** 支持 **Python 3.6** 或更高版本
- **Mercurial** 不再支持 **Python 2**
- **hg purge** 和 **hg clean** 命令现在提供了一个新的 **-i** 选项，该选项允许您删除忽略的文件，而不是未跟踪的文件
- **hg diff** 和 **hg extdiff** 命令现在支持 **--from <revision>** 和 **--to <revision>** 参数
- 现在提供了一个新的内部合并工具 **internal:mergediff**
- Z 标准(ZSTD)压缩现在默认用于新的存储库（如果有的话）
- 现在，提供了一个指定所需扩展的新方法，其在找不到所需的扩展时防止 **Mercurial** 启动

另外，提供了一个新的 **mercurial-chg** 工具，它为 **hg** 命令提供了一个 C 包装器。当您使用 **chg** 命令时，会创建一个 **Mercurial** 命令服务器后台进程，C 程序连接到该后台进程并执行 **Mercurial** 命令。因此，性能会显著提高。

要安装 **mercurial:6.2** 模块流，请使用：

```
# yum module install mercurial:6.2
```

如果要从 **mercurial:4.8** 流升级，请参阅 [切换到更新的流](#)。

(BZ#2089849)

mariadb-java-client rebase 到版本 2.7.1

为用 Java 开发的应用程序提供 MariaDB 连接器的 **mariadb-java-client** 软件包更新至版本 2.7.1。

这个更新在服务中引入了以下变化：

- 客户端身份验证插件现在被定义为服务。因此，您可以轻松地添加新的客户端身份验证插件。驱动程序包括 **caching_sha2_password** 和 **sha256_password** 插件，用于与 MySQL 兼容。
- 现在，凭证插件允许提供凭证信息。驱动程序包含三个默认插件：**AWS IAM**、**Environment** 和 **Property**。
- SSL 工厂服务现在使您能够使用自定义的 SSL 实现。例如，您可以创建一个新的 **HostnameVerifier** 实现。

其他显著变化包括：

- **enabledSslProtocolSuites** 选项现在默认包含 TLSv1.2。

(BZ#2043212)

Redis rebase 到版本 6.2.7

Redis 6 是提供 **redis:6** 模块流的高级键值存储，现已更新到版本 6.2.7。这个更新提供了 RHEL 8.4 以来的 bug 修复、安全修复以及对 6.0 版本的改进。

(BZ#1999873)

httpd 配置中 LimitRequestBody 指令的新默认值

要修复 [CVE-2022-29404](#)，Apache HTTP 服务器中的 **LimitRequestBody** 指令的默认值已从 **0** (无限) 变为 1 GiB。

在 **httpd** 配置文件中没有明确指定 **LimitRequestBody** 的值系统上，更新 **httpd** 软件包会将 **LimitRequestBody** 设为默认值 1 GiB。因此，如果 HTTP 请求正文的总大小超过这个 1 GiB 默认限制，则 **httpd** 会返回 **413 Request Entity Too Large** 错误码。

如果 HTTP 请求消息正文的新默认允许的大小不满足您的用例，请在相应的上下文中（服务器、每目录、每文件或每位置）更新您的 **httpd** 配置文件，并以字节为单位设置您的首选限制。例如，要设置一个新的 2 GiB 限制，请使用：

```
LimitRequestBody 2147483648
```

已被配置为使用 **LimitRequestBody** 指令的任何显式值的系统不受此更改的影响。

(BZ#2128016)

4.11. 编译器和开发工具

新的 GCC 工具集 12

GCC Toolset 12 是一个编译器工具集，提供了开发工具的最新版本。它在 **AppStream** 存储库中以软件集合的形式作为应用程序流提供。

GCC 编译器已更新至版本 12.1.1，它提供了上游 GCC 中提供的很多程序错误修复和增强。

GCC Toolset 12 提供了以下工具和版本：

工具	版本
GCC	12.1.1
GDB	11.2
binutils	2.35
dwz	0.14
annobin	10.76

要安装 GCC Toolset 12，以 root 用户身份运行以下命令：

```
# yum install gcc-toolset-12
```

要从 GCC Toolset 12 运行工具：

```
$ scl enable gcc-toolset-12 tool
```

要运行一个 shell 会话，其中 GCC Toolset 12 中的工具版本会覆盖这些工具的系统版本：

```
$ scl enable gcc-toolset-12 bash
```

如需更多信息，请参阅 [使用 GCC 工具集](#)。

(BZ#2077276)

GCC 工具集 12：Anobin rebase 到版本 10.76

在 GCC 工具集 12 中，Anobin 软件包已更新至版本 10.76。

重要的程序错误修复和增强包括：

- anochek 的新命令行选项告诉它避免使用 **debuginfod** 服务（如果它无法用其他方式查找调试信息）。使用 **debuginfod** 为 anochek 提供了更多信息，但如果 **debuginfod** 服务器不可用，它也会导致 anochek 的性能下降。
- Annobin 源现在可以使用 **meson** 和 **ninja** 来构建，而不是根据需要进行配置和制作。
- Annocheck 现在支持 Rust 1.18 编译器构建的二进制文件。

另外，已在 Annobin 的 GCC 工具集 12 版本中报告了以下已知问题：

在某些情况下，编译可能会失败，并显示类似以下内容的错误消息：

```
cc1: fatal error: inaccessible plugin file
opt/rh/gcc-toolset-12/root/usr/lib/gcc/architecture-linux-gnu/12/plugin/gcc-annobin.so
expanded from short plugin name gcc-annobin: No such file or directory
```

要临时解决这个问题，请在 `plugins` 目录中创建一个从 `annobin.so` 到 `gcc-annobin.so` 的符号链接：

```
# cd /opt/rh/gcc-toolset-12/root/usr/lib/gcc/architecture-linux-gnu/12/plugin
# ln -s annobin.so gcc-annobin.so
```

其中 `architecture` 被正在使用的构架替换：

- `aarch64`
- `i686`
- `ppc64le`
- `s390x`
- `x86_64`

(BZ#2077447)

GCC 工具集 12：binutils rebase 到版本 2.38

在 GCC 工具集 12 中，`binutils` 软件包已更新至版本 2.38。

重要的程序错误修复和增强包括：

- `binutils` 软件包中的所有工具现在支持显示或警告存在多字节字符的选项。
- `readelf` 和 `objdump` 工具现在默认遵循指向各个 `debuginfo` 文件的任何链接。可以使用 `readelf` 的 `--debug-dump=no-follow-links` 选项或 `objdump` 的 `--dwarf=no-follow-links` 选项来禁用此行为。

(BZ#2077448)

GCC 12 和更高版本支持 `_FORTIFY_SOURCE` 级别 3

有了此增强，用户在使用 GCC 版本 12 或更高版本构建时，可以在编译器命令行中使用 `-D_FORTIFY_SOURCE=3` 来构建应用程序。`_FORTIFY_SOURCE` 级别 3 提高了源代码强化的覆盖率，从而提高了在编译器命令行中使用 `-D_FORTIFY_SOURCE=3` 构建的应用的安全性。这在 GCC 版本 12 及更高版本，以及 Clang 版本 9.0 和之后带有 `__builtin_dynamic_object_size` 内置的版本中支持。

(BZ#2033684)

DNS stub 解析器选项现在支持 `no-aaaa` 选项

有了这个增强，`glibc` 现在识别 `/etc/resolv.conf` 和 `RES_OPTIONS` 环境变量中的 `no-aaaa` stub 解析器选项。当此选项处于活跃状态时，不会通过网络发送 AAAA 查询。系统管理员可以出于诊断目的禁用 AAAA DNS 查询，例如，排除仅在 IPv4 网络上的多余查询不会导致 DNS 问题。

(BZ#2096189)

添加了对 `glibc` 中 IBM Z 系列 z16 的支持

现在，对使用 **glibc** 中的 **IBM z16** 平台设置的 **s390** 指令提供支持。**IBM z16** 提供了两个额外的硬件功能，即 **HWCAP_S390_VXRS_PDE2** 和 **HWCAP_S390_NNPA**。因此，应用程序现在可以使用这些功能来交付优化的库和功能。

(BZ#2077835)

新的 **make-latest** 软件包

这个增强引进了 **make-latest** 软件包，其包括 **make** 工具的最新版本。在以前的版本中，我们通过 **GCC** 工具集提供最新的 **make** 版本。现在，您可以单独安装 **make-latest** 软件包，并使用 **scl enable make43 /bin/bash** 运行最新版本（如果 **make43** 版本是最新的）。

(BZ#2083419)

GCC 工具集 12 : **GDB** 已 rebase 到版本 11.2

在 **GCC** 工具集 12 中，**GDB** 软件包已更新至版本 11.2。

重要的程序错误修复和增强包括：

- 对 Aarch64 MTE 的新支持。请参阅带有 **memory-tag** 前缀的新命令。
- **-break-insert** 和 **-dprintf-insert** 的 **--qualified** 选项。这个选项查找用户事件位置的确切匹配，而不是在所有范围内搜索。
例如，**break --qualified foo** 将在全局范围内查找名为 *foo* 的符号。没有 **--qualified**，**GDB** 将在所有范围内搜索具有该名称的符号。
- **--force-condition**: 任何提供的条件都会被定义，即使当前无效也是如此。
- **-break-condition --force**: 与 **MI** 命令类似。
- **-file-list-exec-source-files** 接受可选 **REGEXP** 来限制输出。
- **.gdbinit** 搜索路径包含配置目录。顺序是：
 - a. **\$XDG_CONFIG_HOME/gdb/gdbinit**
 - b. **\$HOME/.config/gdb/gdbinit**
 - c. **\$HOME/.gdbinit**
- 支持 **~/.config/gdb/gdbearlyinit** 或 **~/.gdbearlyinit**。
- **-eix** 和 **-eiex** 早期初始化文件选项。

终端用户界面(TUI)：

- 支持终端用户界面(TUI)窗口中的鼠标操作。
- 不在聚焦窗口上操作的组合键现在传给 **GDB**。

新命令：

- **show print memory-tag-violations**
- **set print memory-tag-violations**
- **memory-tag show-logical-tag**

- **memory-tag with-logical-tag**
- **memory-tag show-allocation-tag**
- **memory-tag check**
- **show startup-quietly** 和 **set startup-quietly** : 一种在 GDB 脚本中指定 **-q** 或 **-quiet** 的方法。仅在早期初始化文件中有效。
- **show print type hex** 和 **set print type hex** : 告诉 GDB 以十六进制形式而不是十进制形式打印结构成员的大小或偏移量。
- **show python ignore-environment** 和 **set python ignore-environment** : 如果启用了, GDB 的 Python 解释器回忽略 Python 环境变量, 就像将 **-E** 传给 Python 可执行文件。仅在早期初始化文件中有效。
- **show python dont-write-bytecode** 和 **set python dont-write-bytecode**: 如果为 **off**, 则这些命令会阻止 GDB 的 Python 解释器编写导入模块的字节代码编译的对象, 就像将 **-B** 传给 Python 可执行文件一样。仅在早期初始化文件中有效。

更改的命令 :

- **break LOCATION if CONDITION** : 如果 **CONDITION** 无效, 则 GDB 会拒绝设置断点。 **-force-condition** 选项会覆盖它。
- **CONDITION -force N COND** : 与前面的命令相同。
- **inferior [ID]**: 当 ID 被省略时, 这个命令会打印有关当前 inferior 的信息。否则, 没有变化。
- **ptype[/FLAGS] TYPE | EXPRESSION**: 在打印 struct 成员的大小和偏移时使用 **/x** 标记来使用十六进制表示法。使用 **/d** 标志来做同样的事情, 但使用十进制。
- **info sources** : 输出已重构。

Python API :

- inferior 对象包含一个只读 **connection_num** 属性。
- 新的 **gdb.Frame.level()** 方法。
- 新的 **gdb.PendingFrame.level()** 方法。
- 忽略了 **gdb.BreakpointEvent** 而不是 **gdb.Stop**。

(BZ#2077492)

libpfm 现在支持 AMD Zen 2 和 Zen 3 处理器

有了这个增强, 用户现在可以使用 **libpfm** 来访问 AMD Zen 2 和 Zen 3 性能监控硬件。

(BZ#2067218)

papi 现在支持 AMD Zen 2 和 Zen 3 处理器

有了这个增强, 用户现在可以使用 **papi** 来访问 AMD Zen 2 和 Zen 3 性能监控硬件。

(BZ#2071558)

改进了 ARM 处理器的硬件标识

有了这个增强，`papi_avail` 工具可以正确地报告各种 ARM 供应商的供应商字符串和代码信息。这个工具允许 `PAPI_get_hardware_info ()` 函数识别由 ARM 以外的公司生产的处理器，这些公司仅限于 `arch64` 架构。因此，开发人员可以调优所需架构的代码。

(BZ#2037427)

更新了 Fujitsu A64FX 事件映射

为 Fujitsu A64FX 处理器更新了 `PAPI` 库。用户现在可以在可用于分析程序性能的 `papi_avail` 输出中使用额外的预设置。

这包括 IDL 事件预设置：

`PAPI_BRU_IDL`

分支机构闲置

`PAPI_FXU_IDL`

整数单位闲置

`PAPI_FPU_IDL`

浮点单位闲置

`PAPI_LSU_IDL`

载入存储单元闲置

(BZ#2037417)

`dyninst` 更新到版本 12.1

`dyninst` 软件包已更新至 12.1 版本。重要的程序错误修复和增强包括：

- 对 `glibc-2.35` 多个命名空间的初始支持。
- 对 DWARF 并行解析的并发修复。
- 更好地支持 `CUDA` 和 `CDNA2` GPU 二进制文件。
- 更好地支持 IBM POWER 系统(little endian)注册访问。
- 更好地支持 PIE 二进制文件。
- 更正了对捕获块的解析。
- 更正了对 64 位 ARM (`arch64`)浮点寄存器的访问。

(BZ#2057676)

`systemtap` 软件包 rebase 到版本 4.7

`systemtap` 软件包已 rebase 到版本 4.7。重要的程序错误修复和增强包括：

- 使用 MOK 键手动为模块签名的新 `--sign-module` 选项，用于 SecureBoot 系统。
- 生成注释源代码的系统范围配置文件的新的 `stap-profile-annotate` 工具。
- 用于探测函数入口和返回的新的通用 Python tapset。

- 对于用户空间中字符串内核空间探测的扩展 `foo` 处理。
- 非捕获组的扩展常规表达式语言。
- 添加了对一些最近添加的内核系统调用的 tapset 支持。

(BZ#2057565)

Rust Toolset rebase 到版本 1.62.1

Rust Toolset 已更新到版本 1.62.1。主要变更包括：

- 解构赋值允许模式在赋值的左侧赋值现有的变量。例如，元组赋值可以交换变量：`(a, b) = (b, a)`
- 现在，使用 `core::arch::asm!` 宏在 64 位 x86 和 64 位 ARM 上支持内联汇编。有关更多详细信息，请参阅参考资料中的 **Inline assembly** 章节 `/usr/share/doc/rust/html/reference/inline-assembly.html`（线上为 <https://doc.rust-lang.org/reference/inline-assembly.html>）。
- 枚举现在可以使用显式注解的 `#[default]` 变体派生 **Default** 特征。
- **Mutex**、**CondVar** 和 **RwLock** 现在使用基于自定义的 **futex** 实现，而非 pthreads，Rust 语言保证提供了新的优化。
- Rust 现在支持 **main** 中的自定义退出代码，包括实现新稳定的 **Termination** 特征的用户定义的类型。
- cargo 支持更多对依赖项功能的控制。**dep:** 前缀可以在不公开为功能的情况下指向可选的依赖项，如果依赖项在其它地方启用了（如 `package-name?/feature-name`），`?` 才启用依赖项功能。
- cargo 有一个新的 **cargo add** 子命令，用于向 **Cargo.toml** 添加依赖项。
- 详情请查看上游发布公告系列：
 - [宣布 Rust 1.59.0](#)
 - [宣布 Rust 1.60.0](#)
 - [宣布 Rust 1.61.0](#)
 - [宣布 Rust 1.62.0](#)
 - [宣布 Rust 1.62.1](#)

(BZ#2075344)

LLVM 工具集 rebase 到版本 14.0.6

LLVM 工具集已 rebase 到版本 14.0.6。主要变更包括：

- 在 64 位 x86 上，添加了对 **AVX512-FP16** 指令的支持。
- 添加了对 Armv9-A、Armv9.1-A 和 Armv9.2-A 架构的支持。
- 在 PowerPC 上，添加了 `__ibm128` 类型来代表 IBM double-double 格式，也称为 `__attribute__((mode(IF)))`。

clang 更改了：

- 现在为 **C++2b** 实现了 **if consteval**。
- 在 64 位 x86 上，添加了对 **AVX512-FP16** 指令的支持。
- 对处于实验状态的 OpenCL 2021 的 OpenCL C 3.0 和 **C++** 的支持。
- 现在 **-E -P** 预处理器输出始终省略空白行，匹配 GCC 行为。以前，输出中可能会出现高达 8 个空行。
- 不仅仅支持 C89，还支持 **C99** 和更高标准的 **-Wdeclaration-after-statement**，匹配 GCC 的行为。值得注意的用例是支持禁止混合声明和代码的风格指南，但希望迁移到新的 C 标准。

如需更多信息，请参阅 [LLVM 工具集](#) 和 [Clang 上游发行注记](#)。

(BZ#2061042)

Go 工具集 rebase 到版本 1.18.2

Go 工具集已 rebase 到版本 1.18.2。

主要变更包括：

- 在保持与之前版本的 Go 向后兼容性的同时引入泛型。
- 一个新的模糊库。
- 新的 **debug/buildinfo** 和 **net/netip** 软件包。
- **go get** 工具不再构建或安装软件包。现在，它只处理 **go.mod** 中的依赖项。
- 如果主模块的 **go.mod** 文件指定了 **go 1.17** 或更高版本，则使用没有任何其他参数的 **go mod download** 命令来下载主模块的 **go.mod** 文件中明确需要的模块的源代码。要下载传输依赖项的源代码，请使用 **go mod download all** 命令。
- **go mod vendor** 子命令现在支持 **-o** 选项来设置输出目录。
- **go mod tidy** 命令现在为需要其源代码来验证构建列表中只有一个模块提供每个导入软件包的模块在 **go.sum** 文件中保留额外的校验和。这个更改不受主模块的 **go.mod** 文件中 Go 版本的影响。

(BZ#2075162)

LLVM 黄金插件 现在在 IBM Z 构架上提供

有了这个增强，用户可以在 IBM Z (**s390x**) 构架上使用 **clang** 和 **ld.bfd** 创建 LTO 构建。**s390x** 架构现在支持与 **ld.bfd** 和 LTO 的连接。

(BZ#2088315)

新模块流：maven:3.8

RHEL 8.7 引入了 **Maven 3.8** 来作为新模块流。

要安装 **maven:3.8** 模块流，请使用：

```
# yum module install maven:3.8
```

如果要从 **maven:3.6** 流升级，请参阅 [切换到更新的流](#)。

(BZ#2083114, BZ#2064785, [BZ#2088473](#))

.NET 版本 7.0 可用

Red Hat Enterprise Linux 8.7 与 .NET 版本 7.0 一起发布。主要改进包括：

- 支持 IBM Power (**ppc64le**)

如需更多信息，请参阅 [.NET 7.0 RPM 软件包的发行注记](#) 和 [.NET 7.0 容器的发行注记](#)。

(BZ#2112096)

4.12. 身份管理

SSSD 现在支持 SID 请求的内存缓存

有了这个增强，SSSD 现在支持 SID 请求的内存缓存，它们是按 SID 查询的 GID 和 UID，反之亦然。内存缓存提高了性能，例如，当向或从 Samba 服务器拷贝大量文件时。

(JIRA:RHELPLAN-123369)

IdM 现在支持使用 Windows Server 2022 配置 AD Trust

有了这个增强，您可以在身份管理(IdM)域和使用运行 Windows Server 2022 的域控制器的活动目录林之间建立跨林信任。

([BZ#2122716](#))

IdM 现在支持在用户密码过期后对允许的 LDAP 绑定数量的限制

有了这个增强，当身份管理(IdM)用户的密码已过期时，您可以设置允许 LDAP 绑定的数量：

-1

在用户必须重置密码之前，IdM 授予用户无限的 LDAP 绑定。这是默认值，与之前的行为匹配。

0

一旦密码过期，这个值会禁用所有 LDAP 绑定。生效时，用户必须立即重置其密码。

1-MAXINT

输入的值可以完全允许许多绑定后过期。

该值可以在全局密码策略和组策略中设置。

请注意，计数为每个服务器存储。

要让用户重置自己的密码，他们需要与其当前到期的密码绑定。如果用户已耗尽所有到期后绑定，则必须由管理员重置密码。

([BZ#782917](#))

IdM 现在指示在名称搜索过程中给定的名称是否是可信 AD 域中的用户或组

有了这个更新，新的 `getorigbyusername ()` 和 `getorigbygroupname ()` 调用被添加到 `libsss_nss_idmap`，一个基于 SID 查询的工具库。当身份管理(IdM)处于与活动目录(AD)域的信任之中时，这个添加使用户和组查找更加健壮。当执行用户或组查找时，IdM 现在可以显示是否指定的名称属于可信域中的用户或组。

([BZ#2062379](#))

新的 ipasmartcard_server 和 ipasmartcard_client 角色

有了这个更新，**ansible-freeipa** 软件包提供了 Ansible 角色来配置身份管理(IdM)服务器和客户端，以进行智能卡验证。**ipasmartcard_server** 和 **ipasmartcard_client** 角色替代了 **ipa-advise** 脚本来自动化和简化集成。使用与其它 **ansible-freeipa** 角色相同的清单和命名方案。

([BZ#2076554](#))

samba rebase 到版本 4.16.1

samba 软件包升级至上游版本 4.16.1，它提供程序错误修复和增强：

- 默认情况下，**smbd** 进程会根据需要自动启动新的 **samba-dcerpcd** 进程，来为分布式计算环境/远程过程调用(DCERPC)提供服务。请注意，Samba 4.16 及更高版本始终需要 **samba-dcerpcd** 来使用 DCERPC。如果您在 `/etc/samba/smb.conf` 文件中的 `[global]` 部分中禁用了 `rpc start on require helpers` 设置，则您必须创建一个 **systemd** 服务单元来在独立模式下运行 **samba-dcerpcd**。
- Cluster Trivial Database (CTDB) **recovery master** 角色已被重命名为 **leader**。因此，以下 **ctdb** 子命令被重命名为：
 - **recmaster** 变为 **leader**
 - **setrecmasterrole** 变为 **setleaderrole**
- CTDB **recovery lock** 配置已被重命名为 **cluster lock**。
- CTDB 现在使用领导广播和关联的超时来确定是否需要选举。

请注意，从 Samba 4.11 开始，服务器消息块版本 1 (SMB1) 协议已被弃用，并将在以后的版本中删除。

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

更新 Samba 后，使用 **testparm** 工具验证 `/etc/samba/smb.conf` 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](#)。

([BZ#2077468](#))

SSSD 现在支持与 Windows Server 2022 集成

有了这个增强，您可以使用 SSSD 将 RHEL 系统直接与使用运行 Windows Server 2022 的域控制器的活动目录林集成。

([BZ#2070793](#))

目录服务器现在支持取消 Auto Membership 插件任务。

在以前的版本中，如果目录服务器有复杂的配置（大的组、复杂的规则以及与其他插件的交互），则 Auto Membership 插件任务会在服务器上产生很高的 CPU 使用率。有了这个增强，您可以取消 Auto Membership 插件任务。因此，性能问题不会再发生。

([BZ#2052528](#))

目录服务器现在在使用 Idapdelete 时支持递归删除操作

在这个版本中，Directory 服务器支持 **Tree Delete Control [1.2.840.113556.1.4.805]** OpenLDAP 控制。因此，您可以使用 **Idapdelete** 程序以递归方式删除父条目的子条目。

(BZ#2057063)

现在，您可以在目录服务器安装过程中设置基本复制选项

有了这个增强，您可以在实例安装过程中使用 `.inf` 文件配置基本复制选项，如身份验证凭证和 changelog 修剪。

(BZ#2057066)

现在，在目录服务器中默认启用复制 changelog 修剪

在以前的版本中，目录服务器没有配置为默认自动修剪 `changelog` 文件。因此，`changelog` 文件可能会变得非常大。有了这个更新，目录服务器默认配置为修剪 7 天前的 changelog 条目，从而防止 `changelog` 文件的过度增长。

(BZ#2062679)

`pki` 软件包重命名为 `idm-pki`

以下 `pki` 软件包现在被重命名为 `idm-pki`，以更好地区分 IDM 软件包和 Red Hat 证书系统软件包：

- `idm-pki-symkey`
- `idm-pki-tools`
- `idm-pki-symkey-debuginfo`
- `idm-pki-tools-debuginfo`
- `idm-pki-acme`
- `idm-pki-base`
- `idm-pki-base-java`
- `idm-pki-ca`
- `idm-pki-kra`
- `idm-pki-server`
- `python3-idm-pki`

`pki-core` 保持不变（这还包括 `pki-core-debuginfo` 和 `pki-core-debugsource`）。

(BZ#2139821)

4.13. 图形基础结构

`vulkan` 软件包在 64 位 IBM POWER 上提供

现在，支持 Vulkan 3D 图形 API 的软件包在 little-endian 64 位 IBM POWER 架构(`ppc64le`)上提供：

- `vulkan-headers`
- `vulkan-loader`
- `vulkan-loader-devel`

- **vulkan-tools**

有了这些软件包，您可以运行使用 Vulkan 渲染引擎的软件。

在以前的版本中，这些软件包仅在 AMD64 和 Intel 64 构架上提供。

(BZ#2012639)

支持新的 AMD GPU

此发行版本添加了对几个 AMD Radeon RX 6000 系列 GPU 和 AMD Ryzen 6000 系列 CPU 的集成图形的支持。

现在支持以下 AMD Radeon RX 6000 系列 GPU 模型：

- AMD Radeon RX 6400
- AMD Radeon RX 6500 XT
- AMD Radeon RX 6300M
- AMD Radeon RX 6500M

AMD Ryzen 6000 系列包括使用以下 CPU 模型找到的集成的 GPU：

- AMD Ryzen 5 6600U
- AMD Ryzen 5 6600H
- AMD Ryzen 5 6600HS
- AMD Ryzen 7 6800U
- AMD Ryzen 7 6800H
- AMD Ryzen 7 6800HS
- AMD Ryzen 9 6900HS
- AMD Ryzen 9 6900HX
- AMD Ryzen 9 6980HS
- AMD Ryzen 9 6980HX

(JIRA:RHELPLAN-135602)

第 12 代 Intel Core GPU 不再需要 force_probe 选项

在本发行版前，您必须设置 `i915.alpha_support=1` 或 `i915.force_probe=*` 内核选项来启用对第 12 代 Intel Core GPU 的支持，其以前被称为 Alder Lake-S 和 Alder Lake-P。

有了这个更新，您不再必需设置此选项，对这些 GPU 的全支持默认启用。

(JIRA:RHELPLAN-136150)

4.14. WEB 控制台

RHEL web 控制台现在将 RHEL 作为 Download an OS VM 工作流的一个选项

有了这个增强，RHEL web 控制台支持使用默认的 **Download an OS** 工作流的 RHEL 虚拟机的安装。因此，您可以在 web 控制台中直接下载并安装 RHEL OS 作为虚拟机。

(JIRA:RHELPLAN-121982)

RHEL web 控制台中用于单独安装内核补丁的一个新按钮

有了这个更新，RHEL web 控制台提供 **Install kpatch updates** 按钮。您可以使用它来仅安装内核补丁，而无需安装其他更新并重启系统。

(JIRA:RHELPLAN-121981)

现在，诊断报告页面提供了新的功能

在更新的 web 控制台诊断报告(**sos** 报告)页面，您现在可以：

- 给报告打标签
- 使用密码短语加密报告
- 在报告中隐藏私有数据

另外，您可以查看之前生成的报告列表，并下载或删除它们。

(JIRA:RHELPLAN-121983)

从 web 控制台 UI 设置加密策略

有了这个更新，您可以直接从 RHEL web 控制台用户界面(UI)直接更改不同的加密策略级别。您可以从 UI 的 **Overview** 页面中的 **Configuration** 字段访问加密策略配置选项。

请注意，您必须有有效的管理访问权限才能更改设置。

(JIRA:RHELPLAN-121980)

Web 控制台中的更新进度页面现在支持自动重启选项

更新进度页面现在有一个 **Reboot after completion** 开关。这会在安装更新后自动重启系统。

([BZ#2056786](#))

4.15. RED HAT ENTERPRISE LINUX 系统角色

ha_cluster RHEL 系统角色现在支持 SBD 隔离和配置 Corosync 设置

ha_cluster 系统角色现在支持以下功能：

SBD 隔离

隔离是 HA 集群配置的重要组成部分。SBD 为节点提供了一种在需要隔离时提供可靠的自终止的方法。在无法实现传统隔离机制的环境中，SBD 隔离特别有用。现在，可以使用 **ha_cluster** 系统角色配置 SBD 隔离。

corosync 设置

ha_cluster 系统角色现在支持 Corosync 设置的配置，如传输、压缩、加密、链接、totem 和仲裁。当默认设置不合适时，需要使用这些设置来将集群的配置与客户的需要和环境匹配。

([BZ#2065339](#),[BZ#2066868](#))

用户可以使用 network RHEL 系统角色，创建具有 IPoIB 功能的连接

network RHEL 系统角色的 **infiniband** 连接类型现在支持 Internet Protocol over Infiniband (IPoIB) 功能的互联网协议。要启用此功能，请为 **infiniband** 的 **p_key** 选项定义一个值。请注意，如果指定了 **p_key**，则 **network_connections** 变量的 **interface_name** 选项必须保留为未设置。以前的 **network** RHEL 系统角色的实现没有正确地验证 **p_key** 值和用于 **infiniband** 连接类型的 **interface_name** 选项。因此，IPoIB 功能在以前不能工作。如需更多信息，请参阅 `/usr/share/doc/rhel-system-roles/network/` 目录中的 README 文件。

([BZ#2086869](#))

network RHEL 系统角色现在为路由规则配置网络设置

在以前的版本中，您可以根据数据包中的目标地址字段路由数据包，但您不能定义源路由和其他策略路由规则。有了这个增强，**network** RHEL 系统角色支持路由规则，以便用户可以控制数据包传输或路由选择。

([BZ#1996731](#))

网络系统角色现在在其管理的配置文件中 **Ansible managed** 注释

使用 **initscripts** 提供程序时，网络系统角色现在在 `/etc/sysconfig/network-scripts` 目录中生成已注释的 **ifcfg** 文件。Networking 角色使用 Ansible 标准的 **ansible_managed** 变量插入 **Ansible managed** 注释。注释声明 **ifcfg** 文件由 Ansible 管理，并且表示不应直接编辑 **ifcfg** 文件，因为网络角色将覆盖该文件。当提供者是 **initscripts** 时，会添加 **Ansible managed** 注释。当使用具有 **nm** (NetworkManager) 提供者的网络角色时，**ifcfg** 文件是由 NetworkManager 管理的，而不是网络角色。

([BZ#2065670](#))

新的 **previous:replaced** 配置可使 **firewall** 系统角色将防火墙设置重置为默认值

管理不同机器集合（其中每台机器都有不同的预先存在的防火墙设置）的系统管理员，现在可以使用 **firewall** 角色中的 **previous: replaced** 配置，来确保所有机器都有相同的防火墙配置设置。**previous: replaced** 配置可能会清除所有现有的防火墙设置，并使用一致的设置替换它们。

([BZ#2043009](#))

增强的 Microsoft SQL Server RHEL 系统角色

以下新变量现在对 **microsoft.sql.server** RHEL 系统角色可用：

- 带有 **mssql_ha_** 前缀的变量用来控制配置高可用性集群。
- **mssql_tls_remote_src** 变量，用来在受管节点上搜索 **mssql_tls_cert** 和 **mssql_tls_private_key** 值。如果保持默认的 **false** 设置，则该角色会在控制节点上搜索这些文件。
- **mssql_manage_firewall** 变量用来自动管理防火墙端口。如果此变量被设为 **false**，则您必须手动启用防火墙端口。
- **mssql_pre_input_sql_file** 和 **mssql_post_input_sql_file** 变量来控制您要在角色执行前或之后运行的 SQL 脚本。这些新变量替换以前的 **mssql_input_sql_file** 变量，这不会影响 SQL 脚本执行的时间。

([BZ#2066338](#), [BZ#2120713](#), [BZ#2039990](#), [BZ#2120714](#))

logging RHEL 系统角色支持文件输入中的 **startmsg.regex** 和 **endmsg.regex**

有了这个更新，您现在可以使用正则表达式过滤来自文件的日志消息。**startmsg_regex** 和 **endmsg_regex** 选项现在包含在文件的输入中。**startmsg_regex** 代表与消息的开始部分匹配的正则表达式，而 **endmsg_regex** 代表与消息的最后部分匹配的正则表达式。因此，您可以根据日期、优先级和严重性等属性过滤消息。

(BZ#2112143)

storage RHEL 系统角色中提供了对精简置备的卷的支持

storage RHEL 系统角色现在可以创建和管理精简配置的 LVM 逻辑卷。精简配置的 LV 会在写时分配，因此在创建卷时具有更好的灵活性，因为提供给精简配置的 LV 的物理存储可以在以后需要时增加。LVM 精简配置还允许创建效率更高的快照，因为精简 LV 的通用数据块及其任何快照都是共享的。

(BZ#2066876)

logging RHEL 系统角色现在支持 **template,severity and facility** 选项

logging RHEL 系统角色现在对文件输入提供新的有用的 **severity** 和 **facility** 选项，以及对文件和转发输出提供新的 **template** 选项。使用 **template** 选项使用 **traditional** 参数指定传统时间格式，使用参数 **syslog** 指定 syslog 协议 23 格式，使用参数 **modern** 指定现代风格格式。您现在可以使用 **logging** 角色根据 **severity** 和 **facility** 进行过滤，并根据 **template** 指定输出格式。

(BZ#2075116)

RHEL 系统角色现在也在禁用了事实收集的 **playbook** 中提供

由于性能或其他原因，可能会在您的环境中禁用 Ansible 事实收集。在以前的版本中，无法在这样的配置中使用 RHEL 系统角色。有了这个更新，系统会在您的配置中检测 **ANSIBLE_GATHERING=explicit** 参数，在 **playbook** 中检测 **gather_facts: false** 参数，并使用 **setup:** 模块来仅收集给定角色所需要的事实（如果在事实缓存中没有）。



注意

如果您因为性能原因而禁用了 Ansible 事实收集，则您可以启用 Ansible 事实缓存，这不会导致从源中检索它们的性能下降。

(BZ#2079008)

sshd RHEL 系统角色验证用于置入目录的 **include** 指令

RHEL 9 上的 **sshd** RHEL 系统角色只管理置入目录中的文件，但之前不验证目录是否包含在主 **sshd_config** 文件中。有了这个更新，该角色会验证 **sshd_config** 是否包含置入目录的 **include** 指令。因此，该角色可以更可靠地应用提供的配置。

(BZ#2086934)

sshd RHEL 系统角色可以通过 **/etc/ssh/sshd_config** 进行管理

应用到 RHEL 9 受管节点的 **sshd** RHEL 系统角色将 SSHD 配置放在置入目录（默认为 **/etc/ssh/sshd_config.d/00-ansible_system_role.conf**）中。在以前的版本中，对 **/etc/ssh/sshd_config** 文件的任何更改都会覆盖 **00-ansible_system_role.conf** 中的默认值。在这个版本中，您可以使用 **/etc/ssh/sshd_config** 而不是 **00-ansible_system_role.conf** 管理 SSHD，同时在 **00-ansible_system_role.conf** 中保留系统的默认值。

(BZ#2086935)

在配置 **masquerade** 或 **icmp_block_inversion** 时，**firewall** RHEL 系统角色不需要 **state** 参数

在配置自定义防火墙区时，变量 `masquerade` 和 `icmp_block_inversion` 是布尔值设置。`true` 值表示 `state: present` 和 `false` 值表示 `state: absent`。因此，配置 `masquerade` 或 `icmp_block_inversion` 时不需要 `state` 参数。

([BZ#2093437](#))

metrics 角色可以导出 postfix 性能数据

现在，您可以使用 `metrics` 角色中的新 `metrics_from_postfix` 布尔变量来记录并进行详细的性能分析。有了这个增强，设置变量会在系统上启用 `pmdapostfix` 指标代理，生成关于 `postfix` 的统计信息。

([BZ#2079114](#))

存储系统角色 现在默认具有较少的详细程度

`storage` 角色输出现在默认具有较少的详细信息。有了这个更新，用户可以提高 `storage` 角色输出的详细信息，使其只有在使用 Ansible 详细程度 1 或更高程度时才生成调试输出。

([BZ#2056480](#))

metrics 系统角色现在生成标头中有正确的 ansible_managed 注释的文件

在以前的版本中，`metrics` 角色不会向角色生成的文件中添加 `ansible_managed` 标头注释。有了这个修复，`metrics` 角色将 `ansible_managed` 标头注释添加到它生成的文件中，因此用户可以轻松地识别 `metrics` 角色生成的文件。

([BZ#2065215](#))

postfix 系统角色现在生成标头中有合适的 ansible_managed 注释的文件

在以前的版本中，`postfix` 角色不会向角色生成的文件中添加 `ansible_managed` 标头注释。有了这个修复，`postfix` 角色向其生成的文件中添加 `ansible_managed` 标头注释，因此用户可以轻松地识别 `postfix` 角色生成的文件。

([BZ#2065216](#))

postfix RHEL 系统角色中用于覆盖以前配置的新选项

如果您管理一组具有 `postfix` 配置不一致的系统，则可能要在所有这些配置上保持一致。在这个版本中，您可以指定 `postfix_conf` 字典中的 `previous: replaced` 选项，以移除任何现有配置并在清 `postfix` 安装之上应用所需的配置。因此，您可以清除任何现有的 `postfix` 配置并确保所有被管理的系统上的一致性。

([BZ#2065218](#))

现在，您可以在 firewall RHEL 系统角色中使用 absent 和 present 状态添加、更新或删除服务

在这个版本中，您可以使用 `present` 状态来添加端口、模块、协议、服务和目标地址，或使用 `absent` 状态来删除它们。请注意，要在 `firewall` RHEL 系统角色中使用 `absent` 和 `present` 状态，请将 `permanent` 选项设置为 `true`。将 `permanent` 选项设置为 `true` 时，状态设置将应用至更改后，并且保持不受角色重新加载的影响。

([BZ#2100297](#))

firewall 系统角色可以使用 PCI 设备 ID 向区添加或删除接口

使用 PCI 设备 ID，`firewall` 系统角色现在可以为区分配或删除网络接口。在以前的版本中，如果只有 PCI 设备 ID 已知而不是接口名称，用户必须首先识别对应的接口名称才能使用 `firewall` 系统角色。在这个版本中，`firewall` 系统角色可以使用 PCI 设备 ID 来管理区中的网络接口。

[\(BZ#2100939\)](#)

network RHEL 系统角色支持使用 nmstate API 的网络配置

有了此更新，**network** RHEL 系统角色支持通过 **nmstate** API 进行网络配置。用户现在可以将所需网络状态的配置直接应用到网络接口，而不是创建连接配置集。该功能还允许部分配置网络。因此，存在以下优点：

- 网络配置复杂性降低
- 应用网络状态更改的可靠方法
- 不需要跟踪整个网络配置

[\(BZ#2100979\)](#)

用于设置自定义监听端口的新的 cockpit 系统角色变量

cockpit 系统角色引入了 **cockpit_port** 变量，它允许您设置默认的 9090 端口以外的自定义监听端口。请注意，如果您决定设置自定义监听端口，您还需要调整 SELinux 策略以允许 Web 控制台监听该端口。

[\(BZ#2115159\)](#)

firewall RHEL 系统角色可以提供 Ansible 事实

有了此增强，您现在可以通过在 playbook 中包含不带参数的 **firewall:** 变量，来从所有系统中收集 **firewall** RHEL 系统角色的 Ansible 事实。要收集更多 Ansible 事实的详细版本，请使用 **detailed: true** 参数，例如：

```
vars:
  firewall:
    detailed: true
```

[\(BZ#2115160\)](#)

将 seuser 和 selevel 的设置添加到 selinux RHEL 系统角色中

有时，在设置 SELinux 上下文文件系统映射时，需要设置 **seuser** 和 **selevel** 参数。在这个版本中，您可以使用 **selinux_fcontext** 中的 **seuser** 和 **selevel** 可选参数来指定 SELinux 上下文文件系统映射中的 SELinux 用户和级别。

[\(BZ#2115162\)](#)

4.16. 虚拟化

ap-check 现在在 RHEL 8 中提供

mdevctl 工具现在提供了一个新的 **ap-check** 支持工具。您可以使用 **mdevctl** 永久配置加密适配器和域，允许将它们直通到虚拟机以及 **矩阵** 和 **vfio-ap** 设备。使用 **mdevctl**，您无需在每次 IPL 后重新配置这些适配器、域和设备。此外，**mdevctl** 可以防止服务器发明其他方法来重新配置它们。

当对 **vfio-ap** 设备调用 **mdevctl** 命令时，会将新的 **ap-check** 支持工具作为 **mdevctl** 命令的一部分来调用，来对 **vfio-ap** 设备配置执行额外的有效性检查。

另外，**chzdev** 工具现在提供了管理系统范围 Adjunct Processor (AP)掩码设置的功能，它决定了 **vfio-ap** 设备可以使用哪些 AP 资源。使用时，**chzdev** 通过生成关联的 **udev** 规则来持久保留这些设置。使用 **lszdev**，您现在可以查询系统范围的 AP 掩码设置。

(BZ#1660911)

IBM Z 上所选的虚拟机现在可以使用大于 896 字节的内核命令行启动

在以前的版本中，如果虚拟机的内核命令行超过 896 字节，启动 RHEL 8 IBM Z 主机上的虚拟机(VM)会始终失败。有了这个更新，QEMU 模拟器可以处理大于 896 字节的内核命令行。现在，如果 VM 内核支持，您可以对虚拟机使用具有非常长的内核命令行的 QEMU 直接内核引导。具体来说，要使用大于 896 字节的命令行，虚拟机必须使用 Linux 内核版本 5.16-rc1 或更高版本。

(BZ#2043830)

使用多线程的虚拟机内存预分配

现在，您可以在域 XML 配置中为虚拟机(VM)内存分配定义多个 CPU 线程，例如，如下所示：

```
<memoryBacking>
  <allocation threads='8'/>
</memoryBacking>
```

这样可确保在启动虚拟机时，有多个线程用于分配内存页。因此，配置了多个分配线程的虚拟机会很快启动，特别是当虚拟机分配了大量 RAM 且由巨页支持时。

(BZ#2067126)

现在完全支持 ESXi hypervisor 和 SEV-ES

现在，您可以启用 AMD Secure Encrypted Virtualization-Encrypted State(SEV-ES)来保护 VMware 的 ESXi hypervisor 版本 7.0.2 及更新版本上的 RHEL 虚拟机(VM)。此功能以前作为技术预览在 RHEL 8.4 中引入。它现已被全面支持。

(BZ#1904496)

IBM Z 上的安全执行现在支持远程认证

IBM Z 架构上的安全执行功能现在支持远程认证。**pvattest** 工具可以创建远程认证请求，以验证启用了安全执行的虚拟机(VM)的完整性。

另外，Guest Interruption State Area (GISA)机制现已为安全执行虚拟机启用，它允许通过完全绕过主机操作系统来直接向虚拟机发送中断。

(JIRA:RHELPLAN-98420, BZ#1984905, BZ#2043870)

4.17. 云环境中的 RHEL

现在，在 Ampere Altra 构架上支持 RHEL 虚拟机

有了这个更新，在带有基于 Ampere® Altra® 构架的 Azure 虚拟机上支持运行 RHEL 操作系统。

(JIRA:RHELPLAN-121252)

open-vm-tools rebase 到 12.0.5

open-vm-tools 软件包升级至版本 12.0.5，它引入了大量的 bug 修复和新功能。最值得注意的是，增加了对通过客户机操作系统变量管理的 Salt Minion 工具的支持。

(BZ#2061193)

cloud-init 的新的 SSH 模块

有了这个更新，SSH 模块已添加到 **cloud-init** 工具中，它会在实例创建过程中自动生成主机密钥。

请注意，有了这个更改，默认的 **cloud-init** 配置已被更新。因此，如果您有一个本地修改，请确保 `/etc/cloud/cloud.cfg` 包含 `"ssh_genkeytypes: ['rsa', 'ecdsa', 'ed25519']"` 行。

否则，**cloud-init** 会创建一个无法启动 **sshd** 服务的镜像。如果出现这种情况，请执行以下操作来临时解决这个问题：

1. 确保 `/etc/cloud/cloud.cfg` 文件包含以下行：

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

2. 检查实例中是否存在 `/etc/ssh/ssh_host_*` 文件。
3. 如果 `/etc/ssh/ssh_host_*` 文件不存在，请使用以下命令生成主机密钥：

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

4. 重启 **sshd** 服务：

```
systemctl restart sshd
```

(BZ#2115791)

4.18. 容器

Container Tools 软件包已更新

包含 Podman、Buildah、Skopeo、crun 和 runc 工具的 Container Tools 现在可用。与之前的版本相比，这个更新提供了 bug 修复和增强的列表。

主要变更包括：

- **podman pod create** 命令现在支持设置 CPU 和内存限制。您可以为 pod 中的所有容器设置限制，pod 中的单个容器可以有自己的限制。
- **podman pod clone** 命令创建现有 pod 的一个副本。
- **podman play kube** 命令现在支持使用 **BlockDevice** 和 **CharDevice** 卷的安全上下文设置。
- **podman play kube** 创建的 Pod 现在可使用 **podman-kube@<service>.service** 的 systemd 单元文件管理（例如 **systemctl --user start podman-play-kube@\$(systemd-escape my.yaml).service**）。
- **podman push** 和 **podman push manifest** 命令现在支持 sigstore 签名。
- Podman 网络现在可以使用 **podman network --opt isolate** 命令来隔离。

Podman 已升级至版本 4.2，要了解有关显著变化的更多信息，请参阅 [上游发行注记](#)。

(JIRA:RHELPLAN-118463)

GitLab Runner 现在可使用 Podman 在 RHEL 上提供

从 GitLab Runner 15.1 开始，您可以使用 Podman 作为 GitLab Runner Docker Executor 中的容器运行时。如需了解更多详细信息，请参阅 [GitLab 的发行注记](#)。

(JIRA:RHELPLAN-100037)

Podman 现在支持 `--health-on-failure` 选项

`podman run` 和 `podman create` 命令现在支持 `--health-on-failure` 选项，来确定容器状态变为不健康时要执行的操作。

`--health-on-failure` 选项支持四个操作：

- **none**：不执行任何操作，这是默认操作。
- **kill**：杀掉容器。
- **restart**：重启容器。
- **stop**：停止容器。



注意

不要将 **restart** 操作与 `--restart` 选项组合。在 `systemd` 单元内运行时，请考虑使用 **kill** 或 **stop** 操作来使用 `systemd` 的重启策略。

(BZ#2097708)

Netavark 网络堆栈现在可用

从 Podman 4.0 开始，新网络堆栈包括两个工具：Netavark 网络设置工具和 Aardvark DNS 服务器。在 RHEL 8 中，以前作为技术预览提供的 Netavark 堆栈现在完全支持。

此网络堆栈具有以下功能：

- 使用 JSON 配置文件配置容器网络
- 创建、管理和删除网络接口，包括网桥和 MACVLAN 接口
- 配置防火墙设置，如网络地址转换(NAT)和端口映射规则
- IPv4 和 IPv6
- 改进了多个网络中容器的功能
- 使用 [aardvark-dns 项目](#) 的容器 DNS 解析



注意

您必须使用相同的 Netavark 堆栈和 Aardvark 权威 DNS 服务器版本。

(JIRA:RHELPLAN-100039)

第 5 章 对外部内核参数的重要更改

本章为系统管理员提供了与 Red Hat Enterprise Linux 8.7 一起分发的内核中显著变化的总结。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、内核配置选项或者任何可见的行为更改。

新内核参数

idxd.tc_override = [HW]

使用 **<bool>** 格式的此参数，您可以允许覆盖设备的默认流量类配置。
默认值设为 **false (0)**。

kvm.eager_page_split = [KVM,X86]

有了这个参数，您可以控制 KVM 在脏日志记录期间是否主动分割所有巨页。Eager 页面分割通过消除写保护错误和内存管理单元(MMU)锁定争用而减少 vCPU 执行中断，否则需要延迟分割大页面。很少执行写或只写入虚拟机内存的一小部分区域的虚拟机工作负载可从禁用 eager 页分割中受益，从而使巨页仍被用于读。

eager 页分割行为取决于是否启用或禁用了 **KVM_DIRTY_LOG_INITIALLY_SET** 选项。

- 如果禁用了，则 **memslot** 中的所有巨页在对那个 **memslot** 启用脏日志时，都会被急切地分割。
- 如果启用了，在 **KVM_CLEAR_DIRTY ioctl ()** 系统调用期间会执行 eager 页分割，并且仅对被清除的页。
Eager 页分割目前只支持分割由两个维度的分页(TDP) MMU 映射的巨页。

默认值设为 **Y (on)**。

kvm.nx_huge_pages_recovery_period_ms = [KVM]

有了这个参数，您可以控制 KVM zaps 4 KiB 页面回到巨页的时间段。

- 如果值为非零 **N**，KVM 会每 **N** 毫秒切换一部分页面。
- 如果值为 **0**，KVM 根据比例选择一个时间段，从而使页面平均 1 小时后切换。
默认值设为 **0**。

mmio_stale_data = [X86,INTEL]

有了这个参数，您可以控制对 Processor Memory-mapped I/O (MMIO) Stale 数据漏洞的缓解。处理器 MMIO Stale Data 是一个漏洞，其可在 MMIO 操作后暴露数据。暴露的数据可能源自或结束于受元数据服务器(MDS)和事务性异步中止(TAA)影响的相同的 CPU 缓冲区。因此，与 MDS 和 TAA 类似，缓解是为了清除受影响的 CPU 缓冲区。

可用的选项有：

- **full**: 对存在安全漏洞的 CPU 启用缓解
- **full,nosmt**: 对存在安全漏洞的 CPU 启用缓解并禁用 SMT。
- **off**: 无条件禁用缓解
在 MDS 或 TAA 受影响的机器上，活跃的 MDS 或 TAA 缓解可以防止 **mmio_stale_data=off**，因为这些漏洞使用同样的机制缓解。因此，为了禁用这个缓解，您还需要指定 **mmds=off** 和 **tsx_async_abort=off**。

不指定这个选项等同于 `mmio_stale_data=full`。

如需更多信息，请参阅 [Documentation/admin-guide/hw-vuln/processor_mmio_stale_data.rst](#)。

`rcutree.rcu_delay_page_cache_fill_msec = [KNL]`

有了这个参数，您可以在响应低内存情况下设置页面缓冲回填延迟（毫秒）。允许的值的范围是 `0:100000`。

`rcuscale.kfree_rcu_test_double = [KNL]`

有了这个参数，您可以测试 `kfree_rcu ()` 函数的双参数变体。如果此参数有与 `rcuscale.kfree_rcu_test_single` 一样的值，则将测试单参数和双参数变体。

`rcuscale.kfree_rcu_test_single = [KNL]`

有了此参数，您可以测试 `kfree_rcu ()` 函数的单参数变体。如果此参数的值与 `rcuscale.kfree_rcu_test_double` 的值相同，则将测试单参数和双参数变体。

`retbleed = [X86]`

有了这个参数，您可以控制具有返回说明(RETBleed)漏洞的 Arbitrary Speculative Code Execution 的缓解。可用的选项有：

- **off**: 没有缓解
- **auto**: 自动选择一个缓解
- **auto,nosmt**: 自动选择一个缓解，如果需要完全缓解，禁用 SMT（只在 Zen1 和没有 STIBP 的旧 Zen1 上）。
- **ibpb** : 缓解基本块边界上的短期投机窗口。安全，最高的性能影响。
- **unret** : 强制启用未训练的返回 thunks，仅对基于 AMD f15h-f17h 的系统有效。
- **unret,nosmt**: 与 **unret** 选项一样，当 STIBP 不可用时禁用 SMT。根据 CPU，选择 **auto** 选项来在运行时选择缓解方法。

不指定这个选项等同于 `retbleed=auto`。

`s390_iommu_aperture = [KNL,S390]`

有了此参数，您可以指定通过 DMA 和 IOMMU API 可访问的每个设备 DMA 地址空间的大小，来作为主内存大小的十进制因子。

- 默认值设为 **1**，意味着您可以并发使用尽可能多的安装的物理内存的 DMA 地址，如果硬件支持，因此一次性映射所有内存。
- 有了值 **2**，您可以映射所有内存两次。
- 值 **0** 不会对硬件提供的内存进行限制，代价是需要大量额外的内存用于表。

更新的内核参数

`acpi_sleep = [HW,ACPI]`

格式：{s3_bios, s3_mode, s3_beep, s4_hwsig, s4_nohwsig, old_ordering, nonvs, sci_force_enable, nobl}

- 有关 **s3_bios** 和 **s3_mode** 的更多信息，请参见 [Documentation/power/video.rst](#)。

- **s3_beep** 用于调试；它使得 PC 的扬声器在调用内核实模式入口点时发出嘟嘟声。
- **s4_hwsig** 会导致内核在从休眠状态恢复过程中检查 ACPI 硬件签名，如果其更改了，则拒绝恢复。默认行为是允许恢复，并在签名更改时简单警告，除非启用了 **s4_hwsig** 选项。
- **s4_nohwsig** 可防止在恢复过程中使用 ACPI 硬件签名或警告。**old_ordering** 会导致关于将设备置于低功耗状态的 **_PTS** 控制方法的 ACPI 1.0 排序的强制执行。默认使用 **_PTS** 的 ACPI 2.0 排序。
- **nonvs** 可防止内核在挂起、休眠和恢复过程中保存和恢复 ACPI NVS 内存。
- **sci_force_enable** 可使内核在从 S1/S3 恢复时直接设置 **SCI_EN**。即使此行为与 ACPI 规范相反，一些损坏的系统没有它也无法正常工作。
- **nobl** 会导致已知的在某些方面与系统挂起和恢复的不正确的行为有关的系统的内部拒绝列表被忽略。明智地使用这个选项。
如需更多信息，请参阅 [Documentation/power/video.rst](#)。

crashkernel=size[KMG],high = [KNL, X86-64, ARM64]

有了这个参数，您可以按如下方式从顶部分配物理内存区域：

- 如果系统安装了超过 4 GB RAM，则物理内存区域可能会超过 4 GB。
- 如果系统安装了少于 4 GB RAM，则物理内存区域将在 4 GB 下分配（如果可用的话）。如果指定了 **crashkernel=X** 参数，则忽略此参数。

crashkernel=size[KMG],low = [KNL, X86-64]

当您传递 **crashkernel=X,high** 时，内核可以分配超过 4 GB 的物理内存区域。这会导致在需要一些低内存的系统上出现第二次内核崩溃（例如，**swiotlb** 需要至少 64M+32K 低内存），需要足够的低内存以确保 32 位设备的 DMA 缓冲区不会被耗尽。内核尝试自动分配低于 4 GB 的至少 256 M 内存。有了此参数，您可以为第二个内核指定 4 GB 的低范围。

- **0**：禁用低分配。当 **crashkernel=X,high** 未使用或保留的内存低于 4 GB 时，它将会被忽略。

kvm.nx_huge_pages_recovery_ratio = [KVM]

有了这个参数，您可以控制有多少 4KiB 页会被定期切换回巨页：

- **0** 禁用恢复
- **N** KVM 每个时间段监控切换 4KiB 页的 **1/Nth**。
默认值设为 **60**。

module.sig_enforce = norid [S390]

有了这个参数，您可以忽略 RID 字段，并强制每个 PCI 功能使用一个 PCI 域。

rcu_nocbs[=cpu-list] = [KNL]

可选参数是一个 CPU 列表。

在使用 **CONFIG_RCU_NOCB_CPU=y** 构建的内核中，您可以启用 no-callback CPU 模式，这可防止从 softirq 上下文调用此类 CPU 回调。这类 CPU 的 RCU 回调的调用将会被卸载到为那个目的创建的 **rcuox/N kthreads**，其中 **x** 是 RCU-preempt 的 **p**，RCU-sched 的 **s**，调解宽限期的 **kthreads** 的 **g**；**N** 是 CPU 号。这可减少卸载 CPU 上的 OS 抖动，这对于 HPC 和实时工作负载非常有用。它还可提高非对称多处理器的能源效率。

- 如果将 **cpulist** 作为参数传递，则指定的 CPU 列表被设置为启动的 no-callback 模式。

- 如果省略 `=` 符号和 `cpulist` 参数，则不会将 CPU 设置为启动的 `no-callback` 模式，但您可以使用 `cpusets` 在运行时切换模式。

`spectre_v2_user = [X86]`

有了这个参数，您可以控制用户空间任务之间的 Spectre 变体 2（直接分支推测）漏洞的缓解。

- **auto**：内核根据可用的 CPU 功能和漏洞选择缓解。
- 默认缓解被设置为 `prctl`。
- 不指定这个选项等同于 `spectre_v2_user=auto`。

`spec_store_bypass_disable = [X86]`

有了这个参数，您可以控制是否使用 Speculative Store Bypass (SSB) 优化来缓解 SSB 漏洞。

- 不指定这个选项等同于 `spec_store_bypass_disable=auto`。
- 默认缓解被设置为 `prctl`。

新的 `sysctl` 参数

`perf_user_access = [ARM64]`

有了这个参数，您可以控制读取性能事件计数器的用户空间的访问。

- 当设置为 **1** 时，用户空间可以直接读取性能监控计数器寄存器。
- 默认值设为 **0**，这表示 **访问被禁用**。
如需更多信息，请参阅 [Documentation/arm64/perf.rst](#)。

`force_cgroup_v2_swappiness`

有了此参数，您可以弃用仅在 `cgroupV1` 中可用的每个组的 `swappiness` 值。由于 `systemd` 设计选择，大多数系统和用户进程都在 `cgroup` 中运行。另外，这些 `cgroup swappiness` 值默认为 **60**。这会导致不希望的影响，其中系统 `swappiness` 值对系统的交换行为有较小的影响。

如果要使用每个 `cgroup swappiness` 功能，您可以使用 `force_cgroup_v2_swappiness=1` 配置系统，以便在整个系统中有更加一致的 `swappiness` 行为。

请注意，这是 RHEL 的专有功能。

第 6 章 设备驱动程序

6.1. 新驱动程序

网络驱动程序

- Maxlinear 以太网 GPY 驱动程序 (**mxl-gpy**)
- Realtek 802.11ax 无线 8852A 驱动程序(**rtw89_8852a**)
- Realtek 802.11ax 无线 8852AE 驱动程序(**rtw89_8852ae**)

图形驱动程序和各种驱动程序

- MHI 主机接口(**mhi**)
- Modem 主机接口(MHI) PCI 控制器驱动程序(**mhi_pci_generic**)
- IDXD 驱动程序 dsa_bus_type 驱动程序 (**idxd_bus**)
- AMD PassThru DMA 驱动程序(**ptdma**)
- Cirrus Logic DSP 支持(**cs_dsp**)
- DRM DisplayPort 帮助程序 (**drm_dp_helper**)
- DRM Buddy Allocator (**drm_buddy**)
- DRM SHMEM 内存管理帮助程序(**drm_shmem_helper**)
- 使用 bochs dispi 接口(**bochs**)的 DRM 驱动程序。
- Intel® PMT Class 驱动程序(**pmt_class**)
- Intel® PMT Crashlog 驱动程序(**pmt_crashlog**)
- Intel® PMT Telemetry 驱动程序(**pmt_telemetry**)
- Intel® speed select interface 驱动程序 (**isst_if_common**)
- Intel® speed select interface mailbox 驱动程序(**isst_if_mbox_msr**)
- Intel® speed select interface pci mailbox 驱动程序 (**isst_if_mbox_pci**)
- Intel® speed select interface mmio 驱动程序(**isst_if_mmio**)
- Intel® Software Defined Silicon 驱动程序(**intel_sdsi**)
- Intel® 扩展功能辅助总线驱动程序(**intel_vsec**)
- ISH ISHTP eclite 客户端 opregion 驱动程序(**ishtp_eclite**)
- 串行 multi 实例化伪设备驱动程序(**serial-multi-instantiate**)
- AMD® SPI Master Controller 驱动程序(**spi-amd**)

6.2. 更新的驱动程序

网络驱动程序

- VMware vmxnet3 虚拟 NIC 驱动程序(**vmxnet3**)已更新至版本 1.7.0.0-k。
- Intel® PRO/1000 网络驱动程序(**e1000e**)已更新至版本 4.18.0-425.3.1。
- Intel® Ethernet Switch Host Interface Driver (**fm10k**)已更新至版本 4.18.0-425.3.1。
- Intel® Ethernet Connection XL710 Network Driver (**i40e**)更新至版本 4.18.0-425.3.1。
- Intel® Ethernet Adaptive Virtual Function Network Driver (**iavf**)已更新至版本 4.18.0-425.3.1。
- Intel® Gigabit Ethernet Network Driver (**igb**)已被更新到版本 4.18.0-425.3.1。
- Intel® Gigabit Virtual Function Network Driver (**igbvf**)已更新到版本 4.18.0-425.3.1。
- Intel® 2.5G Ethernet Linux Driver (**igc**)已更新至版本 4.18.0-425.3.1。
- Intel® 10 Gigabit PCI Express Network Driver (**ixgbe**)已更新至版本 4.18.0-425.3.1。
- Intel® 10 Gigabit Virtual Function Network Driver (**ixgbev**)已更新至版本 4.18.0-425.3.1。
- Mellanox 5 代网络适配器(ConnectX 系列)核心驱动程序(**mlx5_core**)已更新至版本 4.18.0-425.3.1。

存储驱动程序

- Emulex LightPulse Fibre Channel SCSI 驱动程序(**lpfc**)已更新至版本 14.0.0.15。
- MPI3 Storage Controller Device Driver (**mpi3mr**)已更新至版本 8.0.0.69.0。
- LSI MPT Fusion SAS 3.0 Device Driver (**mpt3sas**)已更新至版本 42.100.00.00。
- QLogic Fibre Channel HBA 驱动程序(**qla2xxx**)更新至版本 10.02.07.400-k。
- Microchip Smart Family Controller 的驱动程序 (**smartpqi**)已更新至版本 2.1.18-045。

图形和各种驱动程序更新

- VMware SVGA 设备的独立 drm 驱动程序(**vmwgfx**)已更新至版本 2.20.0.0。

第 7 章 可用的 BPF 功能

本章提供了这个 Red Hat Enterprise Linux 8 次版本的 kernel 中 **Berkeley Packet Filter (BPF)**功能的完整列表。表包括以下列表：

- [系统配置和其他选项](#)
- [可用的程序类型和支持的帮助程序](#)
- [可用的映射类型](#)

本章包含 **bpf tool feature** 命令自动生成的输出。

表 7.1. 系统配置和其他选项

选项	值
unprivileged_bpf_disabled	1 (bpf()系统调用限制为特权用户，无需恢复。)
JIT 编译器	1 (启用)
JIT 编译器强化	1 (为非特权用户启用)
JIT 编译器 kallsyms 导出	1 (为 root 用户启用)
非特权用户的 JIT 的内存限制	264241152
CONFIG_BPF	y
CONFIG_BPF_SYSCALL	y
CONFIG_HAVE_EBPF_JIT	y
CONFIG_BPF_JIT	y
CONFIG_BPF_JIT_ALWAYS_ON	y
CONFIG_DEBUG_INFO_BTF	y
CONFIG_DEBUG_INFO_BTF_MODULES	n
CONFIG_CGROUPS	y
CONFIG_CGROUP_BPF	y
CONFIG_CGROUP_NET_CLASSID	y
CONFIG_SOCK_CGROUP_DATA	y

选项	值
CONFIG_BPF_EVENTS	y
CONFIG_KPROBE_EVENTS	y
CONFIG_UPROBE_EVENTS	y
CONFIG_TRACING	y
CONFIG_FTRACE_SYSCALLS	y
CONFIG_FUNCTION_ERROR_INJECTION	y
CONFIG_BPF_KPROBE_OVERRIDE	y
CONFIG_NET	y
CONFIG_XDP_SOCKETS	y
CONFIG_LWTUNNEL_BPF	y
CONFIG_NET_ACT_BPF	m
CONFIG_NET_CLS_BPF	m
CONFIG_NET_CLS_ACT	y
CONFIG_NET_SCH_INGRESS	m
CONFIG_XFRM	y
CONFIG_IP_ROUTE_CLASSID	y
CONFIG_IPV6_SEG6_BPF	n
CONFIG_BPF_LIRC_MODE2	n
CONFIG_BPF_STREAM_PARSER	y
CONFIG_NETFILTER_XT_MATCH_BPF	m
CONFIG_BPFILTER	n
CONFIG_BPFILTER_UMH	n

选项	值
CONFIG_TEST_BPF	m
CONFIG_HZ	1000
bpf() syscall	可用
大型程序大小限制	可用

表 7.2. 可用的程序类型和支持的帮助程序

程序类型	可用的帮助程序
socket_filter	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_skb_load_bytes, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_skb_load_bytes_relative, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
kprobe	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_current_task bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_override_return, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

程序类型	可用的帮助程序
sched_cls	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_l3_csum_replace, bpf_l4_csum_replace, bpf_tail_call, bpf_clone_redirect, bpf_get_cgroup_classid, bpf_skb_vlan_push, bpf_skb_vlan_pop, bpf_skb_get_key, bpf_skb_key bpf_skb_set_tunnel_key, bpf_redirect, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_get_tunnel_opt, bpf_skb_set_forward_opt bpf_skb_change_proto, bpf_skb_change_type, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_change_tail, bpf_skb_pull_data, bpf_csum_update, bpf_set_hash_invalid, bpf_get_numa_node_id, bpf_skb_change_head, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_set_hash, bpf_skb_adjust_room, bpf_skb_get_xfrm_state, bpf_skb_load_bytes_relative, bpf_fib_lookup, bpf_skb_cgroup_id, bpf_skb_ancestor_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sk_fullsk_fulls bpf_tcp_sock, bpf_skb_ecn_set_ce, bpf_get_listener_sock, bpf_skc_lookup_tcp, bpf_tcp_check_syncookie, bpf_sk_get, bpf_sk_storage_delete, bpf_tcp_gen_syncookie, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_sk_assign, bpf_ktime_get_boot_ns bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_csum_level, bpf_skc_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_bt, bpf_skb_cgroup_classid, bpf_redirect_neigh, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_redirect_peer, bpf_ktime_get_coarse_ns, bpf_check_mt, bpf_for_each_map_elem, bpf_snprintf
sched_act	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_l3_csum_replace, bpf_l4_csum_replace, bpf_tail_call, bpf_clone_redirect, bpf_get_cgroup_classid, bpf_skb_vlan_push, bpf_skb_vlan_pop, bpf_skb_get_key, bpf_skb_key bpf_skb_set_tunnel_key, bpf_redirect, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_get_tunnel_opt, bpf_skb_set_forward_opt bpf_skb_change_proto, bpf_skb_change_type, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_change_tail, bpf_skb_pull_data, bpf_csum_update, bpf_set_hash_invalid, bpf_get_numa_node_id, bpf_skb_change_head, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_set_hash, bpf_skb_adjust_room, bpf_skb_get_xfrm_state, bpf_skb_load_bytes_relative, bpf_fib_lookup, bpf_skb_cgroup_id, bpf_skb_ancestor_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sk_fullsk_fulls bpf_tcp_sock, bpf_skb_ecn_set_ce, bpf_get_listener_sock, bpf_skc_lookup_tcp, bpf_tcp_check_syncookie, bpf_sk_get, bpf_sk_storage_delete, bpf_tcp_gen_syncookie, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_sk_assign, bpf_ktime_get_boot_ns bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_csum_level, bpf_skc_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_bt, bpf_skb_cgroup_classid, bpf_redirect_neigh, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_redirect_peer, bpf_ktime_get_coarse_ns, bpf_check_mt, bpf_for_each_map_elem, bpf_snprintf

程序类型	可用的帮助程序
tracepoint	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_current_task bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_peek, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
xdp	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_redirect, bpf_perf_event_output, bpf_csum_diff, bpf_get_current_task, bpf_get_numa_node_id, bpf_xdp_adjust_head, bpf_redirect_map, bpf_xdp_adjust_meta, bpf_xdp_adjust_tail, bpf_fib_lookup, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_skc_lookup_tcp, bpf_tcp_check_syncookie, bpf_tcp_gen_syncookie, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_check_mtu, bpf_for_each_map_elem, bpf_snprintf
perf_event	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_current_task bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_perf_prog_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_read_branch_records, bpf_get_ns_current_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_snprintf_bt bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

程序类型	可用的帮助程序
cgroup_skb	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_skb_load_bytes, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_skb_load_bytes_relative, bpf_skb_cgroup_id, bpf_get_local_storage, bpf_skb_ancestor_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sk_fullsock, bpf_tcp_sock, bpf_skb_ecn_set_ce, bpf_get_listener_sock, bpf_skc_lookup_tcp, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_skb_cgroup_id, bpf_skb_ancestor_cgroup_id, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
cgroup_sock	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_get_cgroup_classid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_id, bpf_get_socket_cookie, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sk_storage_get, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_netns_cookie, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
lwt_in	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_cgroup_classid, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_pull_data, bpf_get_numa_node_id, bpf_lwt_push_encap, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_skc_to_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

程序类型	可用的帮助程序
lwt_out	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_cgroup_classid, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_pull_data, bpf_get_numa_node_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_ocks, bpf_skc_waits_waits_ocks, bpf_ringbuf_ock bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
lwt_xmit	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_l3_csum_replace, bpf_l4_csum_replace, bpf_tail_call, bpf_clone_redirect, bpf_get_cgroup_classid, bpf_skb_get_tunnel_key, bpf_skb_tunnel_key, bpf_redirect, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_get_tunnel_opt, bpf_skb_set_tunnel_opt, bpf_skb_under_group, bpf_skb_undercgroup bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_change_tail, bpf_skb_pull_data, bpf_csum_update, bpf_set_hash_invalid, bpf_get_numa_node_id, bpf_skb_change_head, bpf_lwt_push_encap, bpf_map_push_elem, bpf_map_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_serve bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_csum_level, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_waits_ocks bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
sock_ops	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_setsockopt, bpf_sock_map_update, bpf_getsockopt, bpf_s_ockops_ock_flags_set, bpf_sock_hash_update, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_tcp_sock, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_load_hdr_opt, bpf_store_hdr_opt, bpf_reserve_hdr_opt, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

程序类型	可用的帮助程序
sk_skb	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_tail_call, bpf_perf_event_output, bpf_skb_load_bytes, bpf_get_current_task, bpf_skb_change_tail, bpf_skb_pull_data, bpf_get_numa_node_id, bpf_skb_change_head, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_skb_adjust_room, bpf_sk_redirect_map, bpf_sk_redirect_hash, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_skc_lookup_tcp, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
cgroup_device	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_uid_gid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf, bpf_ringbuf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
sk_msg	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_cgroup_classid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_msg_redirect_map, bpf_msg_apply_bytes, bpf_msg_cork_bytes, bpf_msg_pull_data, bpf_msg_redirect_hash, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_msg_push_data, bpf_msg_pop_data, bpf_spin_lock, bpf_spin_unlock, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_socks, bpf_skc_waits_waits_ocks, bpf_ringbuf_ock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

程序类型	可用的帮助程序
raw_tracepoint	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
cgroup_sock_addr	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_get_cgroup_classid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_id, bpf_get_socket_cookie, bpf_setsockopt, bpf_getsockopt, bpf_bind, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_skc_lookup_tcp, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_netns_cookie, bpf_get_get_current_tance_id_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_bt, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
lwt_seg6local	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_cgroup_classid, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_pull_data, bpf_get_numa_node_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_ocks, bpf_skc_waits_waits_ocks, bpf_ringbuf_ock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
lirc_mode2	不支持

程序类型	可用的帮助程序
sk_reuseport	<p> bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_skb_load_bytes, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_skb_load_bytes_relative, bpf_sk_reuseport, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf, bpf_ringbuf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf </p>
flow_dissector	<p> bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_skb_load_bytes, bpf_get_current_task, bpf_get_numa_node_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_socks, bpf_skc_waits_waits_socks, bpf_ringbuf_ock bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf </p>
cgroup_sysctl	<p> bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_uid_gid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sysctl_get_name, bpf_sysctl_get_current_value, bpf_sysctl_get_new_value, bpf_sysctl_set_new_value, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_bt, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf </p>
raw_tracepoint_writable	<p> bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf </p>

程序类型	可用的帮助程序
cgroup_sockopt	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_uid_gid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_tcp_sock, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
tracing	不支持

程序类型	可用的帮助程序
struct_ops	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_l3_csum_replace, bpf_l4_csum_replace, bpf_tail_call, bpf_clone_redirect, bpf_get_current_pid_tgid, bpf_get_gid, bpf_uid_gid, bpf_get_current_m bpf_get_cgroup_classid, bpf_skb_vlan_push, bpf_skb_vlan_vlan_pop, bpf_skb_get_tunnel_key, bpf_skb_set_tunnel_key, bpf_perf_event_read, bpf_redirect, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_get_stackid, bpf_csum_diff, bpf_skb_get_tunnel_opt, bpf_skb_set_tunnel_opt, bpf_skb_change_proto, bpf_skb_change_type, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_skb_change_tail bpf_skb_pull_data, bpf_csum_update, bpf_set_hash_invalid, bpf_get_numa_node_id, bpf_skb_change_head, bpf_xdp_adjust_head, bpf_probe_read_str, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_set_hash, bpf_setsockopt, bpf_skb_adjust_room, bpf_redirect_map, bpf_sk_redirect_map, bpf_ocks_map, bpf_xdp_adjust_meta, bpf_perf_event_read_value, bpf_perf_prog_read_value, bpf_getsockopt, bpf_override_return, bpf_sock_ops_cb_flags_set, bpf_msg_redirect_map, bpf_msg_apply_bytes, bpf_msg_cork_bytes, bpf_msg_pull_data, bpf_bind, bpf_xdp_adjust_tail, bpf_skb_get_xfrm_state, bpf_get_stack, bpf_skb_load_bytes_reativel bpf_fib_lookup, bpf_sock_hash_update, bpf_msg_redirect_hash, bpf_sk_redirect_hash, bpf_lwt_push_encap, bpf_lwt_seg6_store_bytes, bpf_lwt_seg6_adjust_rhs, bpf_lwt_seg6_action, bpf_rc_repeat, bpf_rc_keydown, bpf_skb_cgroup_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_sk_sk_reuseport, bpf_skb_ancestor_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_msg_push_data, bpf_msg_pop_data, bpf_rc_pointer_rel, bpf_spin_lock, bpf_spin_unlock, bpf_sk_fullsock, bpf_tcp_sock, bpf_skb_ecn_ce_ce bpf_get_listener_sock, bpf_skc_lookup_tcp, bpf_tcp_check_syncookie, bpf_sysctl_get_name, bpf_sysctl_get_current_value, bpf_sysctl_get_new_value, bpf_sysctl_set_new_value, bpf_strtol, bpf_strtoul, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_send_signal, bpf_tcp_gen_syncookie, bpf_skb_output, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_tcp_send_ack, bpf_send_signal_thread, bpf_jiffies64, bpf_read_branch_records, bpf_get_ns_current_pid_tgid, bpf_xdp_output, bpf_get_netns_cookie, bpf_get_current_ancestor_cgroup_id, bpf_sk_assign, bpf_ktime_get_boot_ns, bpf_seq_printf, bpf_seq_write, bpf_sk_cgroup_id, bpf_sk_ancestor_cgroup_id, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringf_discard, bpf_ringbuf_query, bpf_csum_level, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_get_task_stack, bpf_load_hdr_opt, bpf_store_hdr_opt, bpf_reserve_hdr_opt, bpf_inode_storage_get, bpf_inode_storage_delete, bpf_d_path, bpf_copy_from_user, bpf_snprintf_btf, bpf_seq_printf_btf, bpf_skb_cgroup_classid, bpf_redirect_neigh, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_redirect_peer, bpf_task_storage_get, bpf_task_storage_delete, bpf_get_current_task_btf, bpf_bprm_opts_set, bpf_ktime_get_coarse_ns, bpf_ima_inode_hash, bpf_sock_from_file, bpf_check_mtu, bpf_for_each_map_elem, bpf_snprintf, bpf_sys_bpf, bpf_btf_find_by_name_kind, bpf_sys_close
ext	不支持
lsm	不支持

程序类型	可用的帮助程序
sk_lookup	bpf_map_lookup_elem, bpf_map_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_sk_assign, bpf_ktime_get_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_skc_to_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

表 7.3. 可用的映射类型

映射类型	可用
hash	是
array	是
prog_array	是
perf_event_array	是
percpu_hash	是
percpu_array	是
stack_trace	是
cgroup_array	是
lru_hash	是
lru_percpu_hash	是
lpm_trie	是
array_of_maps	是
hash_of_maps	是
devmap	是
sockmap	是

映射类型	可用
cpumap	是
xskmap	是
sockhash	是
cgroup_storage	是
reuseport_sockarray	是
percpu_cgroup_storage	是
queue	是
queue	是
sk_storage	是
devmap_hash	是
struct_ops	否
ringbuf	是
inode_storage	是
task_storage	否

第 8 章 程序错误修复

这部分论述了 Red Hat Enterprise Linux 8.7 中对用户有严重影响的 bug 修复。

8.1. 安装程序和镜像创建

安装程序不再安装早期版本的软件包

在以前的版本中，在安装过程中安装程序无法正确载入 DNF 配置文件。因此，安装程序有时会在 RPM 事务中安装所选软件包的早期版本。

这个 bug 已经修复，现在只安装存储仓库中最新版本的软件包。如果无法安装最新版本的软件包，安装会失败。

([BZ#1899494](#))

即使在阶段 2 中更改网络配置，Anaconda 安装也会成功

在以前的版本中，当使用 `rd.live.ram` 引导参数时，Anaconda 不会卸载在 `initramfs` 中用来将安装镜像提取到内存中的 NFS 挂载点。因此，如果阶段 2 中更改了网络配置，安装过程可能会变得没有响应，或因为超时错误而失败。

要解决这个问题，在 `switchroot` 之前，在 `initramfs` 中卸载用来提取安装镜像的 NFS 挂载点。因此，安装过程会在不中断的情况下完成。

([BZ#1970726](#))

在安装过程中安装程序会要求在 Kickstart 文件中缺少的加密设备的密码短语

在以前的版本中，当在图形模式运行安装程序时，如果 Kickstart 文件中没有指定密码短语，安装程序不会要求输入加密设备的密码短语。因此，在安装过程中不会应用在 Kickstart 文件中指定的分区。

这个更新添加了一个在安装过程中出现的对话框窗口，要求输入缺少的密码短语。因此，安装程序可以正确地应用在 Kickstart 文件中指定的分区方案。

([BZ#2029101](#))

镜像现在可以成功构建包含有条件依赖项的蓝图中的软件包

在以前的版本中，当使用 Web 控制台自定义带有软件包的包含条件依赖项的蓝图时（如 `ipa-client`、`cockpit`，`podman`），会由于缺失依赖项而导致构建失败。因此，`dep-solve` 软件包过程中不会满足条件依赖项。这个问题现已解决，当分解条件依赖项时，构建将不再失败。

([BZ#2065734](#))

8.2. 软件管理

DNF 现在可以正确地回滚一个带有 Reason Change Action 类型的事务

在以前的版本中，对包含 `Reason Change Action` 类型的事务运行 `dnf history rollback` 命令会失败。有了这个更新，此问题已被解决，`dnf history rollback` 现在可以按预期工作。

([BZ#2060815](#))

8.3. SHELL 和命令行工具

没有参数的 **cmx** 操作不再使 CIM 客户端崩溃

cmx 操作调用一个方法，并返回 XML，一个参数指定调用的方法的名称。在以前的版本中，当在运行没有附加参数的 **cmx** 操作时，命令行 **sblim-wbemcli** 公共信息模型(CIM)客户端会崩溃。有了这个更新，**cmx** 操作需要定义调用的方法的名称。在没有此参数的情况下调用 **cmx** 操作会导致错误消息，CIM 客户端不再崩溃。

([BZ#2075807](#))

opencv 库中的 **cvSaveImage** 函数不再终止用户应用程序

在以前的版本中，**opencv** 库无法正确使用 **cvSaveImage** 函数。因此，用户应用程序会意外终止。有了这个更新，**cvSaveImage** 函数会在磁盘上写镜像数据，不再终止用户应用程序。

([BZ#2104776](#))

如果没有更新 **/etc/fstab** 中的 UUID，ReaR 不再失败并显示错误消息

在以前的版本中，当 ReaR 不能更新 **/etc/fstab** 中的通用唯一标识符(UUID)，来匹配在 UUID 不同的情况下新创建的分区时，ReaR 不会在恢复过程中显示一条错误消息。如果救援镜像与备份不同步，则会出现这种情况。有了这个更新，如果恢复的基本系统文件与重新创建系统不匹配，则在恢复过程中会显示一条错误消息。

([BZ#2072978](#))

带有 PXE 输出方法的 ReaR 不再无法在 **rsync OUTPUT_URL** 位置存储输出文件

在 RHEL 8.5 中，删除了带有 **OUTPUT=PX**E 和 **BACKUP=RSYNC** 选项的 **OUTPUT_URL** 变量的处理。因此，当为 **OUTPUT_URL** 使用 **rsync** 位置时，ReaR 无法将 **initrd** 和内核文件拷贝到此位置，虽然其它们上传到了 **BACKUP_URL** 指定的位置。有了这个更新，RHEL 8.4 和更早版本的行为被恢复了。ReaR 使用 **rsync** 在指定的 **OUTPUT_URL** 目的地中创建所需的文件。

([BZ#2115918](#))

ReaR 现在支持使用 NetBackup 版本 9 恢复系统

在以前的版本中，使用 NetBackup 版本 9 或更高版本的 NetBackup (NBU) 恢复系统会因为缺少库和其他文件而失败。有了这个更新，**NBU_LD_LIBRARY_PATH** 变量包含所需的库路径，救援系统现在包含所需的文件，ReaR 可以使用 NetBackup 方法。

([BZ#2077404](#))

ReaR 不再显示有关丢失符号链接目标的假错误消息

在以前的版本中，ReaR 在创建救援镜像时，会对 **/usr/lib/modules/** 下的 **build** 和 **source** 符号链接缺失符号链接目标而显示不正确的错误消息。这个情况是无害的，您可以安全地忽略此错误消息。有了这个更新，在这种情况下，ReaR 不会报告缺少符号链接目标的假错误消息。

([BZ#2021935](#))

SR-IOV 设备的回退现在可以成功完成

在以前的版本中，因为 **hcnmgr** 脚本中使用了不正确的 **active_slave** 属性而不是 **primary** 属性，在设备故障切换后，单根 I/O 虚拟化(SR-IOV)设备不会回退。有了这个更新，**hcnmgr** 脚本对 SR-IOV 设备使用正确的属性可以成功完成。

([BZ#2078514](#))

ppc64-diag rebase 到版本 2.7.8

平台诊断的 **ppc64-diag** 软件包已更新至版本 2.7.8。主要改进和 bug 修复包括：

- 更新了构建依赖项，以使用 **libvdpd** 工具版本 2.2.9 或更高版本
- 修复了在不支持平台上的 **extract_opal_dump** 错误消息
- 使用 **GCC-8.5** 和 **GCC-11** 编译器修复了构建警告

(BZ#2051313)

lsvdpd rebase 到版本 1.7.14

提供构成硬件清单系统命令的 **lsvdpd** 软件包已更新至版本 1.7.14。有了这个更新，**lsvdpd** 工具可防止在运行 **vpdupdate** 命令时数据库文件的崩溃。

(BZ#2051316)

libvdpd rebase 到版本 2.2.9

包含访问 Vital 产品数据(VPD)的类的 **libvdpd** 软件包已更新至版本 2.2.9。主要改进和 bug 修复包括：

- 修复了数据库锁定
- 更新了 **libtool** 工具版本信息

(BZ#2051319)

8.4. 基础设施服务

RHEL 8 中的打印机测试页面布局已更改

在以前的版本中，如果目标文档格式为 PDF，则不会打印打印测试页。这个更新引入了一个新的测试页面布局，用于更广泛的打印机。请注意，测试页面不包含有关打印机或测试页面打印作业的任何信息。

(BZ#2064606)

frr 二进制文件和脚本有一个新位置

在以前的版本中，管理动态路由堆栈的 **frr** 软件包将其二进制文件和脚本包含在 **/usr/lib/frr** 目录中，这会在应用新目标 SELinux 策略时引起某些问题。因此，SELinux 会在向量缓存(AVC)中记录拒绝信息，并阻止 **frr** 正常启动。

有了这个更新，**/usr/libexec/frr** 是 **frr** 二进制文件和脚本的新位置。因此，SELinux 为 **/usr/libexec/frr** 中的二进制文件和脚本，以及 **/usr/lib64/frr** 中的其他 **frr** 库应用规则，且不再产生拒绝消息。

(BZ#1714984, BZ#1941765)

8.5. 安全

OpenSCAP 补救为 /etc/tmux.conf 设置正确的权限

在以前的版本中，当重新修复 SCAP 规则 **configure_tmux_lock_after_time** 时，**/etc/tmux.conf** 文件是使用与 **umask (600)** 相关的权限创建的。这会导致 **/etc/tmux.conf** 对常规用户不可读。如果常规用户登录，它们会收到一条错误消息，必须在超时到期前等待几分钟，然后才能登录。有了这个更新，规则 **configure_tmux_lock_after_time** 的修复将 **/etc/tmux.conf** 的特定权限设为 644。因此，常规用户不会再遇到错误消息或登录延迟。

(BZ#2064696)

Rsyslog 的 SCAP 规则可以正确识别 .conf 文件

在以前的版本中，规则“确保系统日志文件拥有正确的权限”

(`xccdf_org.ssgproject.content_rule_rsyslog_files_permissions`)没有在 Rsyslog include 语句中扩展 glob 表达式。因此，该规则没有解析所有相关配置文件，有些日志文件没有检查它们的权限。有了这个更新，该规则可以正确地扩展 glob 表达式，以识别它需要解析的 .conf 文件。现在，该规则可以正确地处理所需的 .conf 文件，以确保所有配置的日志文件都有正确的权限。

(BZ#2075384)

chronyd 的规则不需要显式的 chrony 用户配置

RHEL 默认在 `chrony` 用户下运行 `chronyd`。在以前的版本中，`chronyd` 服务配置用户的检查和修复过于严格。过于严格的检查导致误报和过度的补救。在这个版本中，规则

`xccdf_org.ssgproject.content_rule_chronyd_run_as_chrony_user` 的检查和补救会更新，以便最低限度的正确配置和传统的显式正确配置可以通过。因此，规则尊重默认的 RHEL 行为，不需要显式的 `chrony` 用户配置。

(BZ#2077531)

警告添加到 rsyslog_remote_loghost 中

SCAP 规则 `xccdf_org.ssgproject.content_rule_rsyslog_remote_loghost` 确保 Rsyslog 守护进程被配置为将日志消息发送到远程日志主机。但是，该规则没有配置 TCP 队列。因此，如果没有配置 TCP 队列，系统会挂起，远程日志主机变得不可用。这个更新添加了一条警告消息，解释了如何配置 TCP 队列。如果您在使用这个规则时遇到系统挂起，请读取警告并正确配置系统。

(BZ#2078974)

sudo_custom_logfile 的修复也适用于自定义 sudo 日志文件

在以前的版本中，SCAP 安全指南规则 `xccdf_org.ssgproject.content_sudo_custom_logfile` 的修复不适用于与 `/var/log/sudo.log` 不同路径的自定义 `sudo` 日志文件。有了这个更新，规则已被修复，如果系统有一个与预期路径不匹配的自定义的 `sudo` 日志文件，则该规则可以正确修复。

(BZ#2083109)

firewalld_sshd_port_enabled 的修复现在可以正常工作

在以前的版本中，SCAP 规则 `xccdf_org.ssgproject.content_rule_firewalld_sshd_port_enabled` 的 Bash 修复会错误地处理网络接口列表。另外，配置文件的名称与所需名称不同。这个更新已修复了补救。因此，补救可以正确地处理所有网络接口，配置文件有可预测的名称。

(BZ#2109602)

fagenrules --load 现在可以正常工作

在以前的版本中，`fapolicyd` 服务无法正确处理信号挂起(SIGHUP)。因此，在收到 SIGHUP 信号后 `fapolicyd` 会终止，`fagenrules --load` 命令无法正常工作。此更新包含了对此问题的修复。因此，`fagenrules --load` 现在可以正常工作，规则更新不再需要手动重启 `fapolicyd`。

(BZ#2070639)

8.6. 网络

NetworkManager 工具会强制对来自各种源的 IPv6 地址进行正确排序

通常，IPv6 地址的顺序会影响源地址选择的优先级。例如，当您进行传出 TCP 连接时，在以前的版本中，通过 **manual**、**dhcpv6** 和 **autoconf6** 方法添加 IPv6 地址的相对优先级是不正确的。有了这个更新，此问题已被解决，排序优先级现在反映了这个逻辑：**manual** > **dhcpv6** > **autoconf6**。但是，**ipv6.addresses** 设置下的地址顺序没有变化，最后添加的地址仍具有最高的优先级。

(BZ#2097270)

非对称路由现在可以正常工作

以前的 RHEL 8 次要版本包含一个更改，其在某些情况下会导致连接跟踪失败。因此，非对称路由无法正常工作。此发行版本恢复了在 RHEL 8.6 中引入的更改。因此，非对称路由可以正常工作。

(BZ#2062870)

8.7. 内核

弃用 CgroupV1 memory.swappiness 的新功能，允许一致的交换行为

CgroupV1 包括 **memory.swappiness** 每 cgroup swappiness 值，用于控制给定 cgroup 的交换行为。

但是，**systemd** 进程在 **cgroups** 内运行，而 **sysctl** swappiness 值对 **swap** 启发的影响很小。此 cgroup 忽略 **sysctl** 或 **tuned** 配置中的值，系统上运行的进程会被分配一个默认的 swappiness 值 **60**。因此，在高内存压力和页面回收的情况下，与分配的 swappiness 值相比，可能会发生更早或更激进的交换。

这个更新引入了一个新的 **sysctl** 变量 **/proc/sys/vm/force_cgroupv2_swappiness**，默认值为 **0**。当设置为 **1** 时，**memory.swappiness** 值将被弃用，所有每 cgroups swappiness 值都反映了 **/proc/sys/vm/swappiness** 文件中系统范围的 swappiness 值。因此，cgroups 的内存交换行为更为一致。

(BZ#2084242)

在为加密设备输入密码短语后，Anaconda 不再失败

在以前的版本中，如果在准备安装时禁用了 **kdump**，并且用户所选了加密磁盘分区，则 Anaconda 安装程序在为加密设备输入密码短语后会失败，并出现回溯。

这个更新解决了这个问题，用户不再需要启用 **kdump** 来创建加密的磁盘分区。

(BZ#2086100)

v1 模式中的 net_prio 或 net_cls 控制器现在可以正常工作

在以前的版本中，在 **cgroup-v2** 环境中，在 v1 模式下使用 **net_prio** 或 **net_cls** 控制器禁用了套接字数据的层次结构跟踪。因此，套接字数据跟踪控制器的 **cgroup-v2** 层次结构不处于活跃状态，**dmesg** 命令会报告以下信息：

```
cgroup: cgroup: disabling cgroup2 socket matching due to net_prio or net_cls activation
```

这个更新确保 **cgroup-v2** 在重启后被正确激活。

(BZ#2046396)

8.8. 引导加载程序

grubby 现在将参数传递给未来的内核

当安装较新版本的内核时，**grubby** 工具不会传递来自上一内核版本的内核命令行参数。因此，GRUB 引导装载程序会忽略用户设置。有了这个修复，在安装了新内核版本后，用户设置仍然有效。

([BZ#1978226](#))

8.9. 高可用性和集群

在创建新 Booth 票据时，**pcs** 现在识别 **mode** 选项

在以前的版本中，当用户在添加新的 Booth 票据时指定了 **mode** 选项，**pcs** 会报告错误 **invalid booth ticket option 'mode'**。有了这个修复，您可以在创建 Booth 票据时指定 **mode** 选项。

([BZ#1786964](#))

pcs 现在验证 **stonith-watchdog-timeout** 的值

在以前的版本中，可以将 **stonith-watchdog-timeout** 属性设为与 SBD 配置不兼容的值。这可能会导致隔离循环，或者可能导致集群将隔离操作视为成功，即使操作没有完成。有了这个修复，**pcs** 会在您设置它时验证 **stonith-watchdog-property** 的值，以防止不正确的配置。

([BZ#1954099](#))

8.10. 动态编程语言、网页和数据库服务器

现在，当启用 **OQGraph** 插件时，**MariaDB 10.5** 会发出删除不存在的表的警告

在以前的版本中，当 **OQGraph** 存储引擎插件加载到 **MariaDB 10.5** 服务器时，**MariaDB** 不会发出删除不存在的表的警告。特别是，当用户尝试使用 **DROP TABLE** 或 **DROP TABLE IF EXISTS** SQL 命令删除不存在的表时，**MariaDB** 不会返回错误消息，也不会记录警告。这个 bug 已解决，会在上述场景中显示一条警告。

([BZ#1944653](#))

8.11. 编译器和开发工具

当从 **fork** 处理程序回调调用 **pthread_atfork** 或 **dclose** 时，应用程序不再死锁

在以前的版本中，当 **glibc** 获取内部锁时，应用程序会调用 **pthread_atfork** 处理程序回调。因此，注册 **fork** 处理程序或从 **fork** 处理程序调用 **dclose** 可能会导致应用程序死锁。

现在，在 **fork** 处理程序运行时，可使用不同的同步机制来保护内部数据结构。因此，当从 **fork** 处理程序回调调用 **pthread_atfork** 或 **dclose** 时，应用程序不再死锁。

([BZ#1888660](#))

当只有目录时，**Makefile** 中的通配符功能不再返回符号链接

在以前的版本中，**glob()** 使用的 **GLOB_ONLYDIR** 提示会将符号链接错误地报告为某个 XFS 文件系统上的目录。使用 **glob()** 时，**make** 不会确认提示是否是实际的目录，因此，当只有目录时，**Makefile** 中的通配符功能会返回符号链接。

这个 bug 已解决，当只有目录时，**Makefile** 中的通配符功能不再返回符号链接。

([BZ#1982608](#))

popen() 不再导致多线程进程崩溃

在以前的版本中，在使用多线程进程中的接口时，`popen()` 会导致应用程序崩溃。有了这个更新，bug 已修复，在使用 `popen()` 时，多线程进程不会再崩溃。

(BZ#2065588)

有些 IBM 字符集的 0xBC 代码点的映射现在是 U+00AF MACRON

在以前的版本中，IBM256、IBM277、IBM278、IBM280、IBM284、IBM297 和 IBM424 字符集编码了 EBCDIC 代码点 0xBC 来作为 Unicode 字符 U+203E OVERLINE。因此，当使用 `glibc` 提供的 `iconv` 程序时，在这些字符集中包含 0xBC 代码点的转换文本对于非 Unicode 字符集（比如 ISO-8859-1）会失败，因为它们不能编码 U+203E OVERLINE 字符。

有了这个更新，此 bug 已被修复。因此，在所有情况下，IBM277、IBM278、IBM280、IBM284 和 IBM297 字符集都可以转换为 ISO-8859-1。对于 IBM256 和 IBM424 字符集，如果输入文本包含 0xBC 代码点，则转换不再失败，且对应的输出为 U+00AF MACRON。

(BZ#1961109)

tempnam 函数现在使用 getrandom 来增加生成的文件名的随机性

在以前的版本中，Red Hat Enterprise Linux 8.4 及之后的版本中的 `tempnam` 函数使用时间派生的随机性来选择路径。因此，在快速连续调用时，`tempnam` 函数不会生成可能文件名的全集。这个 bug 已被一个新的实现修复，它使用 `getrandom` 函数来提高生成的文件名称的随机性。现在，`tempnam` 函数可以生成更多不同的文件名。

(BZ#2089247)

POWER9-optimized strncpy 功能不再给出不正确的结果

在以前的版本中，POWER9 `strncpy` 功能没有使用正确的注册作为 padding NUL 字节的来源。因此，输出缓冲包含未初始化的寄存器内容，而不是 NUL padding。在这个版本中，`strncpy` 功能已被修复，输出缓冲区的末尾会正确添加 NUL 字节。

(BZ#2091553)

en_US@ampm 区域现在可以使用 locale -a 正确列出

在以前的版本中，在 `locale -a` 命令输出的 `en_US@ampm` 列表中有一个缺陷。因此，当尝试使用 `locale -a` 打印的名称/别名设置这个区域时，`setlocale` API 会失败。有了这个更新，`en_US@ampm` 现在可被正确列出，并为 `locale -a` 打印的所有区域成功调用 `setlocale`。

(BZ#2104907)

事件的单元掩码现在都包括在 papi_xml_event_info 输出中

在以前的版本中，`papi_xml_event_info` 中测试事件单元掩码信息不完整。在某些情况下，事件的单元掩码不包含在 `papi_xml_event_info` 输出中。这个 bug 已修复，因此 `papi_xml_event_command` 现在可以打印出事件的所有单元掩码。

(BZ#2037426)

8.12. 身份管理

默认情况下，调试消息不再记录到 /var/log/messages

在以前的版本中，`ipa-dnskeysyncd` 和 `ipa-ods-exporter` 守护进程会默认将所有调试消息记录到 `/var/log/messages`，导致日志大幅增大。现在，您可以通过在 `/etc/ipa/dns.conf` 文件中设置 `debug=True` 来配置调试日志级别。有关详细信息，请参阅 `default.conf (5)` 手册页。

(BZ#2059396)

保留用户帐户

在以前的版本中，如果您运行 `ipa user-del --preserve user_login` 命令来保留用户帐户，输出会错误地返回信息 **Deleted user "user_login"**。这个消息错误地表示用户已被删除，且未按预期保留。有了这个更新，输出现在返回 **Preserved user "user_login"**。

(BZ#2022028)

传输大于 4 GB 的 Kerberos 数据库

在以前的版本中，`kprop` 服务和 `kproptd` 命令在存储 Kerberos KDC 数据库的大小时使用 32 位值。因此，如果数据库大小超过 4 GB，则将 Kerberos 数据库转储文件从主 Kerberos 服务器传输到副本服务器会失败。

这个更新修改了 Kerberos，并且它现在可以传输大于 4 GB 的 KDC 数据库。

(BZ#2026462)

处理 LDAP 组成员列表中的不可读对象

在此次更新之前，SSSD 处理 LDAP 组成员列表中的不可读取对象不一致，这会导致不可读取对象出错或在某些情况下不可读取对象被忽略。

有了这个更新，SSSD 有一个新的选项 `ldap_ignore_unreadable_references` 来修改此行为。如果 `ldap_ignore_unreadable_references` 选项被设为 `false`，不可读对象会导致错误，如果设为 `true`，不可读对象会被忽略。由于最初的不一致行为，默认值被设为 `false`，更新后，一些组查找可能会失败。在这种情况下，在 `/etc/sss/sss.conf` 文件中相应 `[domain/name of the domain]` 部分中设置 `ldap_ignore_unreadable_references = True`。

这允许以一致的方法处理不可读对象，并使用新的 `ldap_ignore_unreadable_references` 选项调整行为。

(BZ#2069379)

8.13. DESKTOP

Airplane Mode 开关始终显示

在以前的版本中，在启用了 airplane 模式后，`Settings` 应用的 `Wi-Fi` 部分中的 `Airplane Mode` 切换消失了。有了这个更新，这个问题已被解决，并且 `Settings` 始终会显示 `Airplane Mode` 开关，无论其状态是什么。

(BZ#2079139)

8.14. 图形基础结构

Motif 应用程序中的热键激活正确的项目

在以前的版本中，菜单热键使用 Motif 工具包激活应用程序中错误的菜单项。当子菜单处于打开状态时，您按与其项目关联的热键，应用程序会改为在父菜单中激活一个项目。

有了这个更新，这个问题已被解决，热键现在激活正确的子菜单项。

(BZ#2060571)

禁用 IPv6 和 DisallowTCP=false 后，桌面不再无法启动

在以前的版本中，在以下情况下登录后，X11 桌面会话无法启动：

- 在您的系统上 IPv6 网络被禁用。
- 在 GDM 配置中启用了 **DisallowTCP=false** 选项。

有了这个更新，这个问题已被解决，您可以使用描述的配置按预期登录到 X11 会话。

(BZ#2075132)

8.15. WEB 控制台

使用 Web 控制台删除 USB 主机设备现在可以按预期工作

在以前的版本中，当您把 USB 设备附加到虚拟机时，USB 设备的设备号和总线号会在传给虚拟机后改变。因此，由于设备和总线号关联不正确，因此使用 Web 控制台删除这样的设备会失败。有了这个更新，这个问题已被解决，您可以使用 web 控制台删除 USB 主机设备。

(JIRA:RHELPLAN-109067)

使用 Web 控制台附加多个主机设备现在可以按预期工作

在以前的版本中，当使用 web 控制台选择多个设备来附加到虚拟机(VM)时，只会附加一个设备，剩余的设备都会被忽略。有了这个更新，这个问题已被解决，现在您可以使用 web 控制台同时附加多个主机设备。

(JIRA:RHELPLAN-115603)

8.16. RED HAT ENTERPRISE LINUX 系统角色

修复了一个拼写错误，为正确的绑定模式支持 active-backup

在以前的版本中，在指定 **active-backup** 绑定模式时，在支持 InfiniBand 端口时，**active_backup** 有一个拼写错误。由于这个拼写错误，对于 InfiniBand 绑定端口，连接无法支持正确的绑定模式。这个更新通过将绑定模式改为 **active-backup** 解决了拼写错误。现在，连接可以成功支持 InfiniBand 绑定端口。

(BZ#2064067)

IPRouteUtils.get_route_tables_mapping() 函数现在接受任何空白序列

在以前的版本中，**iproute2** 路由表数据库的一个解析程序（如 **/etc/iproute2/rt_tables**）断言，文件中条目的格式为 **254 main**，只有一个空格字符将数字 ID 和名称分开。因此，解析器无法缓存路由表名称和表 ID 之间的所有映射。因此用户无法通过定义路由表名称来在路由表中添加静态路由。有了这个更新，解析器会接受表 ID 和表名称之间的任何空格序列。因此，因为解析器会缓存路由表名称和表 ID 之间的所有映射，所以用户可以通过定义路由表名称来在路由表中添加静态路由。

(BZ#2115884)

metrics RHEL 系统角色的配置会正确遵循符号链接

安装 **mssql pcp** 软件包后，**mssql.conf** 文件位于 **/etc/pcp/mssql/** 中，由被符号链接 **/var/lib/pcp/pmdas/mssql/mssql.conf** 定为目标。但是，**metrics** 角色覆盖了符号链接，而不是遵循它，并配置 **mssql.conf**。因此，运行 **metrics** 角色会更改到常规文件的符号链接，因此配置只影响

`/var/lib/pcp/pmdas/mssql/mssql.conf` 文件。这会导致符号链接失败，而主配置文件 `/etc/pcp/mssql.conf` 没有受到配置的影响。这个问题现已解决，遵循符号链接的 **follow: yes** 选项已添加到 **metrics** 角色中。因此，**metrics** 角色会保留符号链接，并正确配置主配置文件。

(BZ#2060377)

tlog RHEL 系统角色现在可以被 SSSD 正确覆盖

在以前的版本中，**tlog** RHEL 系统角色依赖系统安全服务守护进程(SSSD)文件提供者以及启用的 **authselect** 选项 **with-files-domain**，来在 `nsswitch.conf` 文件中设置正确的 **passwd** 条目。有了这个修复，**tlog** 角色现在可以更新 `nsswitch.conf`，来确保 **tlog-rec-session** 被 SSSD 正确覆盖。

(BZ#2072749)

卷的 **mount_options** 参数现在对卷有效

在以前的版本中，该参数会从卷的有效参数列表中意外删除。因此，用户无法为卷设置 **mount_options** 参数。有了这个 bug 修复，**mount_options** 参数已被添加到有效参数列表中，并且已重构了代码来捕获错误。因此，**storage** RHEL 系统角色可以为卷设置 **mount_options** 参数。

(BZ#2083378)

metrics RHEL 系统角色 README 和文档现在明确指定角色在 RHEL 的特定版本上支持的 Redis 和 Grafana 版本

在以前的版本中，当尝试在不支持的平台上使用具有不支持的 Redis 和 Grafana 版本的 **metrics** 角色时，角色会失败。这个更新澄清了关于角色在哪个 RHEL 版本上支持哪个 Redis 和 Grafana 版本的文档。因此，您可以避免在不支持的平台上使用不支持的 Redis 和 Grafana 的版本。

(BZ#2100285)

kernel_settings RHEL 系统角色现在可以正确地安装 **python3-configobj**

在以前的版本中，**kernel_settings** 角色返回一个错误：**python3-configobj** package could not be found。角色找不到软件包，是因为它没有在受管主机上安装 **python3-configobj**。有了这个更新，该角色现在可以在受管主机上安装 **python3-configobj**，并可以正常工作。

(BZ#2060378)

storage RHEL 系统角色现在可以正确地支持 LVM 卷的 条状 和 **raid0** 级别

storage RHEL 系统角色以前错误地将 RAID 级别 **striped** 和 **raid0** 报告为 LVM 卷不支持。现在，这个问题已被解决，角色现在可以正确创建由 LVM 支持的所有 RAID 级别的 LVM 卷：**raid0**、**raid1**、**raid4**、**raid5**、**raid6**、**raid10**、**striped** 和 **mirror**。

(BZ#2083426)

metrics RHEL 系统角色在更新其配置后自动重启 **pmie** 和 **pmlogger** 服务

在以前的版本中，**pmie** 和 **pmlogger** 服务在配置更改后不会重启，并等待处理程序执行。这会导致其他 **metrics** 服务出现错误，这些服务需要 **pmie** 和 **pmlogger** 配置来匹配其运行时行为。有了这个更新，角色会在配置更新后立即重启 **pmie** 和 **pmlogger**，其配置与依赖 **metrics** 服务的运行时行为匹配，且它们可以正常工作。

(BZ#2100298)

forward_port 参数现在接受 字符串和字典选项

在以前的版本中，在 **firewall** RHEL 系统角色中，**forward_port** 参数只接受 **string** 选项。但是，该角色文档声明了 **字符串** 和 **字典** 选项的支持。因此，用户读取并遵循文档收到错误。这个程序错误已通过使 **forward_port** 接受这两个选项来解决。因此，用户可以安全地按照文档配置端口转发。

(BZ#2101607)

nbde_client 系统角色现在在指定额外的 Dracut 命令行参数时使用正确的空间

Dracut 框架需要在指定附加参数时正确启动，如内核命令行参数。如果没有通过正确的空间指定参数，则 Dracut 可能不会将指定的额外参数附加到内核命令行中。在这个版本中，**nbde_client** 系统角色在创建附加组件 Dracut 配置文件时使用正确的空间。因此，该角色可以正确地设置 Dracut 命令行参数。

(BZ#2115161)

ssh 和 **sshd** RHEL 系统角色中的最小 RSA 密钥位长度选项

意外使用短 RSA 密钥可能会使系统更易受到攻击。在这个版本中，您可以使用 **ssh** 和 **sshd** RHEL 系统角色中的 **RSAMinSize** 选项为 OpenSSH 客户端和服务器设置 RSA 密钥最小长度。

(BZ#2109997)

NBDE 客户端系统角色支持静态 IP 地址

在之前的 RHEL 版本中，重启具有静态 IP 地址的系统，并使用 Network Bound Disk Encryption (NBDE)Client 系统角色配置将更改系统的 IP 地址。有了此更改，NBDE 客户端系统角色支持具有静态 IP 地址的系统，重启后它们的 IP 地址不会改变。

请注意，默认情况下，NBDE 角色在启动时使用 DHCP，并在系统引导时切换到配置的静态 IP。

(BZ#2071011)

8.17. 虚拟化

带有故障转移 VF 的虚拟机的实时预复制迁移现在可以正常工作

在以前的版本中，如果虚拟机使用启用了虚拟功能(VF)故障转移功能的设备，则尝试预复制迁移正在运行的虚拟机(VM)会失败。这个更新解决了这个问题，现在可在上述场景中正常迁移虚拟机。

(BZ#2054656)

8.18. 云环境中的 RHEL

现在，即使在 Alibaba 云上启动 nm-cloud-setup 服务后，实例也可以保留主 IP 地址

在以前的版本中，在 Alibaba 云上启动实例后，nm-cloud-setup 服务会在多 IPv4 地址的情况下将不正确的 IP 地址配置为主 IP 地址。因此，这会影响到出站连接的 IPv4 源地址的选择。有了这个更新，在手动配置二级 IP 地址后，NetworkManager 软件包会从 primary-ip-address 元数据中获取主 IP 地址，并正确配置主和辅 IP 地址。

(BZ#2082000)

SR-IOV 在 Azure 上的 ARM 64 RHEL 8 虚拟机中不再表现不佳

在以前的版本中，在运行在 Microsoft Azure 平台上的 ARM 64 RHEL 8 虚拟机中，SR-IOV 网络设备比预期的吞吐量要低的多，延迟要高的多。这个问题已被解决，受影响的虚拟机现在运行正常。

(BZ#2068429)

在 AWS 上使用 cloud-init 启动 RHEL 8 虚拟机不再超过预期的时间

在以前的版本中，在 Amazon Web Services (AWS) 上使用 **cloud-init** 服务初始化 RHEL 8 的 EC2 实例需要更长的时间。RHEL 8 的 Amazon 机器镜像(AMI)已被更新来包括此问题的修复，初始化 RHEL 8 的 EC2 实例现在可以正常工作。

但是，在自定义和上传自己的 RHEL 8 镜像时，您可能仍然会遇到缓慢的情况。要避免这个问题，请在将镜像上传到 AWS 之前，从您要用于创建虚拟机的镜像中删除 **/etc/resolv.conf** 文件。

(BZ#1862930)

8.19. 容器

由于不匹配存储库 ID，DNF 和 YUM 不再失败

在以前的版本中，DNF 和 YUM 存储库 ID 与 DNF 或 YUM 期望的格式不匹配。例如，如果您运行以下示例，则会发生错误：

```
# podman run -ti ubi8-ubi
# dnf debuginfo-install dnsmasq
...
This system is not registered with an entitlement server. You can use subscription-manager to register.
```

在这个版本中，这个问题已被解决。后缀 **--debug-rpms** 被添加到所有 debug 存储库名称中（如 **ubi-8-appstream-debug-rpms**），后缀 **-rpms** 被添加到所有 UBI 存储库名称中（如 **ubi-8-appstream-rpms**）。

如需更多信息，请参阅 [通用基础镜像\(UBI\)：镜像、存储库、软件包和源代码](#)。

(BZ#2120378)

现在，可以拉取使用 Beta GPG 密钥签名的容器镜像

在以前的版本中，当您拉取 RHEL Beta 容器镜像时，Podman 会失败，并显示错误消息：**Error: Source image rejected: None of the signatures are accepted**。由于当前构建被配置为默认不信任 RHEL Beta GPG 密钥，所以镜像无法被拉取。有了这个更新，**/etc/containers/policy.json** 文件支持一个新的 **keyPaths** 字段，该字段接受一个包含可信密钥的文件列表。因此，在默认配置中，现在接受使用 GA 和 Beta GPG 密钥签名的容器镜像。

(BZ#2020301)

第 9 章 技术预览

这部分列出了 Red Hat Enterprise Linux 8.7 中提供的所有技术预览。

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](#)。

9.1. SHELL 和命令行工具

ReaR 在 64 位 IBM Z 构架中作为技术预览提供

64 位 IBM Z 构架中现在作为技术预览提供了基本的 Relax 和 Recover(ReaR)功能。您只能在 z/VM 环境中的 IBM Z 上创建 ReaR 救援镜像。备份和恢复逻辑分区(LPAR)还没有测试。

当前唯一可用的输出方法是 Initial Program Load(IPL)。IPL 生成一个内核和一个初始 ramdisk(initrd)，可与 **ziPL** 引导装载程序一起使用。



警告

目前，救援过程会重新格式化连接到系统的所有 DASD（直接附加的存储设备）。如果系统存储设备中存有宝贵的数据，则不要尝试进行系统恢复。这还包括用于引导到救援环境的 **ziPL** 引导装载程序、ReaR 内核和 initrd 的设备。确保保留一个副本。

如需更多信息，请参阅在 [64 位 IBM Z 架构中使用 ReaR 救援镜像](#)。

(BZ#1868421)

9.2. 网络

KTLS 作为技术预览提供

RHEL 作为技术预览提供内核传输层(KTLS)。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 还包括将 TLS 记录加密卸载到提供此功能的网络接口控制器(NIC)的接口。

(BZ#1570255)

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。它伴随 **XDP**，并将编程选择的数据包的高效重定向授予给用户空间应用程序，以便做进一步处理。

(BZ#1633143)

可作为技术预览的 XDP 功能

红帽提供了以下 eXpress Data Path(XDP)功能作为不受支持的技术预览：

- 在 AMD 和 Intel 64 位以外的构架中载入 XDP 程序。请注意，**libxdp** 库不适用于 AMD 和 Intel 64 位的构架。
- XDP 硬件卸载。

[\(BZ#1889737\)](#)

TC 的多协议标签交换，作为技术预览提供

Multi-protocol Label Switching (MPLS) 是一个内核内数据转发机制，用于跨企业网络路由流量。在 MPLS 网络中，接收数据包的路由器根据附加到数据包的标签决定数据包的其他路由。使用标签时，MPLS 网络可以处理带有特定特征的数据包。例如，您可以添加 **tc 过滤器**，以一致的方式管理从特定端口接收的数据包或执行特定类型的流量。

数据包进入企业网络后，MPLS 路由器对数据包执行多个操作，如 **push** 来添加标签、**swap** 来更新标签，以及 **pop** 来删除标签。MPLS 允许基于 RHEL 中的一个或多个标签在本地定义操作。您可以配置路由器并设置流量控制(**tc**)过滤器，根据 MPLS 标签堆栈条目(**lse**)元素来对数据包执行相应的操作，如 **label**、**traffic class**、**bottom of stack** 以及 **time to live**。

例如，以下命令在 `enp0s1` 网络接口中添加过滤器以匹配传入的数据包，第一个标签为 `12323`，第二个标签为 `45832`。在匹配的数据包中，会执行以下操作：

- 第一个 MPLS TTL 被缩减（如果 TTL 到达 0，则会丢弃数据包）
- 第一个 MPLS 标签被改为 `549386`
- 生成的数据包通过 `enp0s2` 传输，目标 MAC 地址为 `00:00:5E:00:53:01`，源 MAC 地址为 `00:00:5E:00:53:02`

```
# tc filter add dev enp0s1 ingress protocol mpls_uc flower mpls lse depth 1 label 12323 lse
depth 2 label 45832 \
action mpls dec_ttl pipe \
action mpls modify label 549386 pipe \
action pedit ex munge eth dst set 00:00:5E:00:53:01 pipe \
action pedit ex munge eth src set 00:00:5E:00:53:02 pipe \
action mirrored egress redirect dev enp0s2
```

[\(BZ#1814836, BZ#1856415\)](#)

systemd-resolved 服务现在作为技术预览提供

systemd-resolved 为本地应用程序提供名字解析。该服务实现了缓存和验证 DNS 存根解析器、链接本地多播名称解析(LLMNR)和多播 DNS 解析器和响应器。

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

[\(BZ#1906489\)](#)

9.3. 内核

kexec 快速重启功能作为一种技术预览提供

kexec 快速重启功能作为一种技术预览继续提供。**kexec** 快速重启可以显著加快引导过程，因为内核允许直接引导到第二个内核，而无需首先通过基本输入/输出系统(BIOS)。要使用这个功能：

1. 手动加载 **kexec** 内核。
2. 重启操作系统。

[\(BZ#1769727\)](#)

accel-config 软件包作为技术预览提供

accel-config 软件包现在作为技术预览在 Intel **EM64T** 和 **AMD64** 构架上提供。这个软件包有助于控制并配置 Linux 内核中的数据流化器 (DSA) 子系统。另外, 它还通过 **sysfs** (pseudo-文件系统)配置设备, 以 **json** 格式保存并载入配置。

(BZ#1843266)

SGX 作为技术预览

软件扩展 (SGX) 是一个 Intel® 技术, 用于保护软件代码和数据不受公开和修改的影响。RHEL 内核部分提供了 SGX v1 和 v1.5 功能。版本 1 启用使用 **Flexible Launch Control** 机制的平台使用 SGX 技术。

(BZ#1660337)

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。

虚拟机包含一个新的系统调用 **bpf ()**, 它允许创建各种类型的映射, 还允许在特殊的装配式代码中加载程序。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。如需更多信息, 请参阅 **bpf(2)** 手册页。

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。每个组件都处于不同的开发阶段。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 :

- **AF_XDP**, 将 **eXpress Data Path (XDP)** 连接到用户空间的套接字, 用于优先考虑数据包处理性能的应用程序。

(BZ#1559616)

用于内核的 Intel 数据流加速器驱动程序作为技术预览提供

内核的 Intel 数据流加速器驱动程序(IDXD)目前作为技术预览提供。它是一个 Intel CPU 集成加速器, 包括一个带有处理地址空间 ID(pasid)提交和共享虚拟内存(SVM)的共享工作队列。

(BZ#1837187)

Soft-RoCE 作为一个技术预览提供

通过融合以太网的远程直接内存访问(RDMA)是一个通过以太网实现 RDMA 的网络协议。Soft-RoCE 是 RoCE 的软件实现, 它维护两个协议版本 : 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)

stmmac 驱动程序作为技术预览提供

红帽将芯片(SoC)上 Intel® Elkhart Lake 系统的 **stmmac** 作为不受支持的技术预览提供。

(BZ#1905243)

9.4. 文件系统和存储

现在 ext4 和 XFS 作为技术预览提供文件系统 DAX

在 Red Hat Enterprise Linux 8 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，系统必须有某种形式的持久性内存可用，通常是以一个或多个非易失性双内联内存模块(NVDIMM)的形式，并且提供 DAX 能力的文件系统必须在 NVDIMM 上创建。另外，该文件系统必须使用 **dax** 挂载选项挂载。然后，挂载了 dax 的文件系统上的文件的 **mmap** 会导致存储直接映射到应用程序的地址空间。

(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，卸载覆盖，然后在没有这些选项的情况下挂载覆盖，则可能会出现意想不到的结果或错误。

- 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 8.3 将 Stratis 更新至 2.1.0 版本。如需更多信息，请参阅 [Stratis 2.1.0 发行注记](#)。

(JIRA:RHELPLAN-1212)

在 IdM 域成员中设置 Samba 服务器作为技术预览提供

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)

NVMe/TCP 主机作为技术预览提供

访问和共享通过 TCP/IP 网络(NVMe/TCP)的 Nonvolatile Memory Express(NVMe/TCP)存储及其相应的 **nvme_tcp.ko** 内核模块已被添加为技术预览。使用 **nvme-cli** 软件包提供的工具可以将 NVMe/TCP 作为主机管理。NVMe/TCP 主机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)

9.5. 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 上运行，容器捆绑包功能作为技术预览提供。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 现在提供 **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](#))

在资源移动后自动删除位置约束作为技术预览提供

当您执行 **pcs resource move** 命令时，这会为资源添加一个约束，以防止其在当前运行的节点上运行。**pcs resource move** 命令的新 **--autodelete** 选项现在作为技术预览提供。当您指定这个选项时，命令创建的位置约束会在资源移动后自动删除。

([BZ#1847102](#))

9.6. 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。

在以前的版本中，IdM API 被改进来启用多个 API 命令版本。这些增强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。
- 开发人员可以使用 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)服务器现在实现了 DNS 安全扩展(DNSSEC)，这是一组增强 DNS 协议安全的 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换的。

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 操作实践, 版本 2](#)
- [安全域名系统\(DNS\)部署指南](#)
- [DNSSEC 键翻滚时间注意事项](#)

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)

ACME 作为技术预览提供

自动证书管理环境(ACME)服务现在作为技术预览在 Identity Management(IdM)中提供。ACME 是一个用于自动标识符验证和证书颁发的协议。它的目标是通过缩短证书生命周期并避免证书生命周期管理中的手动过程来提高安全性。

在 RHEL 中，ACME 服务使用红帽认证系统(RHCS)PKI ACME 响应程序。RHCS ACME 子系统自动部署到 IdM 部署中的每个证书颁发机构(CA)服务器上，但只有管理员启用它之后，它才会为请求提供服务。RHCS 在发布 ACME 证书时使用 **acmeIPAServerCert** 配置文件。签发的证书的有效期为 90 天。启用或禁用 ACME 服务会影响整个 IdM 部署。



重要

建议仅在所有服务器都运行 RHEL 8.4 或以上版本的 IdM 部署中启用 ACME。早期的 RHEL 版本不包括 ACME 服务，这可能会在混合版本部署中引起问题。例如，没有 ACME 的 CA 服务器可能会导致客户端连接失败，因为它使用不同的 DNS Subject Alternative Name(SAN)。



警告

目前，RHCS 不会删除过期的证书。由于 ACME 证书在 90 天后过期，因此过期的证书可能会累积，这会影响性能。

- 要在整个 IdM 部署中启用 ACME，请使用 **ipa-acme-manage enable** 命令：

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

- 要在整个 IdM 部署中禁用 ACME，请使用 **ipa-acme-manage disable** 命令：

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

- 要检查是否安装了 ACME 服务，以及它是否启用或禁用了，请使用 **ipa-acme-manage status** 命令：

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

(BZ#1628987)

RHEL IdM 允许将用户身份验证委派给外部身份提供程序作为技术预览

在 RHEL IdM 中，您可以把用户与支持 OAuth 2 设备授权流的外部身份提供程序(IdP)关联。当这些用户使用 RHEL 8.7 中提供的 SSSD 版本进行身份验证时，它们将在外部 IdP 执行身份验证和授权后会接收带有 Kerberos 票据的 RHEL IdM 单点登录功能。

主要特性包括：

- 使用 `ipa idp-*` 命令为外部 IdP 添加、修改和删除引用
- 使用 `ipa user-mod --user-auth-type=idp` 命令为用户启用 IdP 验证

如需更多信息，请参阅 [使用外部身份提供程序向 IdM 进行身份验证](#)。

(BZ#2101770)

sssd-idp 子软件包作为技术预览提供

SSSD 的 `sssd-idp` 子软件包包含 `oidc_child` 和 `krb5 idp` 插件，它们是对身份管理(IdM)服务器执行 OAuth2 身份验证的客户端组件。这个功能只在 RHEL 8.7 及更高版本，以及 RHEL 9.1 及更高版本上的 IdM 服务器中提供。

(BZ#2065692)

SSSD 内部 krb5 idp 插件作为技术预览提供

SSSD `krb5 idp` 插件允许您使用 OAuth2 协议对外部身份提供者(IdP)进行身份验证。这个功能只在 RHEL 8.7 及更高版本，以及 RHEL 9.1 及更高版本上的 IdM 服务器中提供。

(BZ#2056483)

9.7. 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：磁盘使用率分析器 (`baobab`)、防火墙配置 (`firewall-config`)、红帽订阅管理器(`subscription-manager-cockpit`)或 Firefox Web 浏览器。使用 Firefox，管理员可以远程连接到本地 Cockpit 守护进程。

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

IBM Z 上的 GNOME 桌面作为技术预览提供

GNOME 桌面，包括 Firefox 网页浏览器，现在在 IBM Z 构架中作为技术预览提供。现在，您可以使用 VNC 连接到运行 GNOME 的远程图形会话来配置和管理您的 IBM Z 服务器。

(JIRA:RHELPLAN-27737)

9.8. 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)

9.9. 虚拟化

用于 KVM 虚拟机的 AMD SEV 和 SEV-ES

作为技术预览, RHEL 8 为使用 KVM 虚拟机的 AMD EPYC 主机提供安全加密虚拟化 (SEV) 功能。如果在虚拟机(VM)上启用, SEV 会加密虚拟机的内存来保护虚拟机被主机访问。这提高了虚拟机的安全性。

另外, 增强的 Encrypted State 版本 SEV-ES) 也作为技术预览提供。SEV-ES 在虚拟机停止运行时加密所有 CPU 注册内容。这可防止主机修改虚拟机的 CPU 注册或读取它们中的任何信息。

请注意, SEV 和 SEV-ES 仅适用于第 2 代 AMD EPYC CPU (代号 Rome) 或更新版本。另请注意, RHEL 8 包括 SEV 和 SEV-ES 加密, 但不包括 SEV 和 SEV-ES 安全认证。

(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, 也可以启用由 Intel vGPU 操作的 VNC 控制台。通过启用它, 用户可以连接到虚拟机的 VNC 控制台, 并查看由 Intel vGPU 托管的虚拟机桌面。但是, 目前这仅适用于 RHEL 客户机操作系统。

(BZ#1528684)

创建嵌套虚拟机

对于装有 RHEL 8 的 Intel、AMD64、IBM POWER 以及 IBM Z 系统主机上运行的 KVM 虚拟机, 嵌套的 KVM 虚拟化作为技术预览提供。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。

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

技术预览：选择 Intel 网络适配器现在在 Hyper-V 上的 RHEL 客户端中提供 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对 **ixgbevf** 和 **iaanf** 驱动程序支持的 Intel 网络适配器使用单根 I/O 虚拟化(SR-IOV)功能。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

该功能目前随 Microsoft Windows Server 2016 及更高版本一起提供。

(BZ#1348508)

使用 virtiofs 在主机和虚拟机之间共享文件

作为技术预览，RHEL 8 现在提供 virtio 文件系统(**virtiofs**)。使用 **virtiofs**，您可以在主机系统及其虚拟机 (VM)之间高效地共享文件。

(BZ#1741615)

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 和 AMD 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)

9.10. 云环境中的 RHEL

RHEL 机密虚拟机现在在 Azure 上作为技术预览提供

有了更新的 RHEL 内核，现在您可以在 Microsoft Azure 上创建并运行机密虚拟机(VM)作为技术预览。但是，在 Azure 上引导时无法加密 RHEL 机密虚拟机镜像。

(JIRA:RHELPLAN-122316)

9.11. 容器

Toolbox 作为技术预览提供

在以前的版本中，Toolbox 工具是基于 RHEL CoreOS [coreos/toolbox](#) 的。有了这个版本，Toolbox 已被 [containers/toolbox](#) 替代。

(JIRA:RHELPLAN-77238)

sigstore 签名现在作为技术预览提供

从 Podman 4.2 开始，您可以使用容器镜像签名的 sigstore 格式。sigstore 签名会与容器镜像一起存储在容器注册中心，而无需有单独的签名服务器来存储镜像签名。

(JIRA:RHELPLAN-75165)

多个签名镜像的可信 GPG 密钥的功能是作为技术预览提供

`/etc/containers/policy.json` 文件支持一个新的 **keyPaths** 字段，该字段接受一个包含可信密钥的文件的列表。因此，在默认配置中，现在接受使用 GA 和 Beta GPG 密钥签名的容器镜像。

例如：

```
"registry.redhat.io": [
  {
```

```
    "type": "signedBy",
    "keyType": "GPGKeys",
    "keyPaths": ["/etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release", "/etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-beta"]
  }
]
```

(JIRA:RHELPLAN-118470)

podman-machine 命令不被支持

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)

第 10 章 过时的功能

这部分提供在 Red Hat Enterprise Linux 8 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保持不变。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](#)。

10.1. 安装程序和镜像创建

弃用了一些 Kickstart 命令和选项

在 RHEL 8 Kickstart 文件中使用以下命令和选项将会在日志中打印警告信息：

- **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)

Kickstart autostep 命令已弃用

autostep 命令已弃用。有关这个命令的相关部分已从 [RHEL 8 文档](#) 中删除。

(BZ#1904251)

10.2. 软件管理

rpmbuild --sign 已弃用

从 RHEL 8.1 开始，**rpmbuild --sign** 命令被弃用。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)

10.3. SHELL 和命令行工具

OpenEXR 组件已弃用

OpenEXR 组件已弃用。因此，对 **EXR** 镜像格式的支持已从 **imagecodecs** 模块中去掉了。

(BZ#1886310)

dump 软件包中的 dump 工具已弃用

用于文件系统备份的 **dump** 工具已弃用，在 RHEL 9 中将不再提供。

在 RHEL 9 中，红帽建议根据使用情况使用 **tar**、**dd** 或 **bacula**、备份工具，对 ext2、ext3 和 ext4 文件系统提供了完整和安全的备份。

请注意，**dump** 软件包中的 **restore** 工具仍可用，在 RHEL 9 中也被支持，并作为 **restore** 软件包提供。

(BZ#1997366)

ABRT 工具已被弃用

用于检测和报告应用程序崩溃的自动错误报告工具(ABRT)在 RHEL 8 中已弃用。作为替代，使用 **systemd-coredump** 工具记录和存储核心转储，其是程序崩溃后自动生成的文件。

(BZ#2055826)

ReaR crontab 已被弃用

rear 软件包中的 **/etc/cron.d/rear** crontab 已在 RHEL 8 中弃用，且不在 RHEL 9 中提供。crontab 会每晚检查磁盘布局是否已更改，如果发生了更改，则运行 **rear mkrescue** 命令。

如果您需要这个功能，在升级到 RHEL 9 后，请手动配置定期运行 ReaR。

(BZ#2083301)

Bacula 中的 SQLite 数据库后端已被弃用

Bacula 备份系统支持多个数据库后端：PostgreSQL、MySQL 和 SQLite。SQLite 后端已被弃用，并将在今后的 RHEL 版本中不被支持。作为一种替代，迁移到另一种后端(PostgreSQL 或 MySQL)，且在新部署中不使用 SQLite 后端。

([BZ#2089399](#))

RHEL 8 systemd 不再支持 hidepid=n 挂载选项

挂载选项 `hidepid=n`，其控制谁可以访问 `/proc/[pid]` 目录中的信息，与 RHEL 8 提供的 `systemd` 基础架构不兼容。

另外，使用这个选项可能会导致 `systemd` 启动的某些服务生成 SELinux AVC 拒绝信息，并阻止其他操作完成。

如需更多信息，请参阅有关的知识库解决方案 [RHEL7 和 RHEL8 建议使用"hidepid=2"挂载 /proc ?](#)。

([BZ#2038929](#))

/usr/lib/udev/rename_device 工具已被弃用

用于重命名网络接口的 `udev` 帮助工具 `/usr/lib/udev/rename_device` 已被弃用。

([BZ#1875485](#))

raw 命令已被弃用

`raw (/usr/bin/raw)` 命令已被弃用。在以后的 Red Hat Enterprise Linux 版本中使用这个命令可能会导致错误。

(JIRA:RHELPLAN-133171)

10.4. 安全

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。为确保在 NSS 取消支持时依赖 SEED 密码的部署平稳过渡，红帽推荐对其它密码套件的支持。

请注意，在 RHEL 中，SEED 密码已经被默认禁用。

([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，请参阅红帽客户门户网站中的知识库文章 [RHEL 8 中的强加密默认值和弱加密算法的弃用](#) 和 [update-crypto-policies\(8\)](#) 手册页。

([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)

SSL2 Client Hello 在 NSS 中已弃用

传输层安全性(TLS)协议版本 1.2 和更早版本允许以与安全套接字层(SSL)协议版本 2 向后兼容方式与 **Client 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

可信平台模块(TPM)安全加密处理器标准将在 2016 年更新至版本 2.0。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)

crypto-policies 派生的属性现已被弃用

随着自定义策略中 **crypto-policies** 指令作用域的引入，以下派生属性已被弃用：

tls_cipher、**ssh_cipher**、**ssh_group**、**ike_protocol** 和 **sha1_in_dnssec**。另外，使用 **protocol** 属性而不指定范围现也已被弃用。有关推荐的替代品，请参阅 **crypto-policies(7)** 手册页。

(BZ#2011208)

使用 /etc/selinux/config 运行时禁用 SELinux 现已弃用

使用 **/etc/selinux/config** 文件中的 **SELINUX=disabled** 选项禁用 SELinux 已被弃用。在 RHEL 9 中，当您只通过 **/etc/selinux/config** 禁用 SELinux 时，系统启动时会启用 SELinux，但没有载入任何策略。

如果您的情况确实需要完全禁用 SELinux，红帽建议通过在内核命令行中添加 **selinux=0** 参数来禁用 SELinux，如在标题为 [使用 SELinux 的在引导时更改 SELinux 模式](#) 部分中所述。

(BZ#1932222)

ipa SELinux 模块从 selinux-policy 中删除了

ipa SELinux 模块已从 **selinux-policy** 软件包中删除，因为不再维护它了。这个功能现在包括在 **ipa-selinux** 子软件包中。

如果您的场景需要在本地 SELinux 策略中使用 **ipa** 模块的类型或接口，请安装 **ipa-selinux** 软件包。

(BZ#1461914)

fapolicyd.rules 已被弃用

包含允许和拒绝执行规则的文件 **/etc/fapolicyd/rules.d/** 目录替代了 **/etc/fapolicyd/fapolicyd.rules** 文件。**fagenrules** 脚本现在将此目录中的所有组件规则文件合并到 **/etc/fapolicyd/compiled.rules** 文件。**/etc/fapolicyd/fapolicyd** 中的规则仍由 **fapolicyd** 框架处理，但只是为了保证向后兼容。

(BZ#2054741)

10.5. 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)

dropwatch 工具已弃用

dropwatch 工具已弃用。以后的发行版本中不再支持该工具，因此不建议在新部署中使用它。作为这个软件包的替代品，红帽建议使用 **perf** 命令行工具。

有关使用 **perf** 命令行工具的更多信息，请参阅红帽客户门户网站的 [Perf入门](#) 部分或 [perf](#) 手册页。

(BZ#1929173)

cgdcboxd 软件包已被弃用

控制组数据中心桥接交换守护进程(**cgdcboxd**)是监控数据中心桥接(DCB)netlink 事件和管理 **net_prio** 控制组子系统的服务。从 RHEL 8.5 开始，**cgdcboxd** 软件包被弃用，并将在下一个主要 RHEL 发行版本中删除。

(BZ#2006665)

xinetd 服务已被弃用

xinetd 服务已被弃用，并将在 RHEL 9 中删除。作为替换，请使用 **systemd**。详情请查看 [如何将 xinetd 服务转换为 systemd](#)。

(BZ#2009113)

WEP Wi-Fi 连接方法已被弃用

在 RHEL 8 中弃用了不安全的有线等效隐私(WEP) Wi-Fi 连接方法，并将在 RHEL 9.0 中删除。对于安全的 Wi-Fi 连接，请使用 Wi-Fi Protected Access 3(WPA3)或 WPA2 连接方法。

(BZ#2029338)

不受支持的 xt_u32 模块现已被弃用

使用不受支持的 **xt_u32** 模块，**iptables** 用户可以匹配数据包头或有效负载中的任意 32 位数据。从 RHEL 8.6 开始，**xt_u32** 模块已被弃用，并将在 RHEL 9 中删除。

如果使用 **xt_u32**，请迁移到 **nftables** 数据包过滤框架。例如，首先将您的防火墙更改为使用带有原生匹配项的 **iptables**，以便逐步替换单个规则，然后使用 **iptables-translate** 和附带的工具迁移到 **nftables**。

如果 **nftables** 中没有原生匹配，请使用 **nftables** 的原始有效负载匹配功能。详情请查看 **nft(8)** 手册页中 **原始有效负载表达式** 部分。

([BZ#2061288](#))

术语 **slave** 在 **nmstate** API 中已弃用

红帽承诺使用适当的语言。因此，**slaves** 术语在 **Nmstate** API 中被弃用。使用 **nmstatectl** 时使用术语 **port**。

([JIRA:RHELDPCS-17641](#))

10.6. 内核

内核实时补丁现在涵盖所有 RHEL 次版本

从 RHEL 8.1 开始，已为延长更新支持(EUS)策略涵盖的所选定的 RHEL 次版本提供了内核实时补丁，以修复级别为关键(Critical)和重要(Important)的通用漏洞披露(CVE)。要容纳同时涵盖内核和用例的最大数量，对于内核的每个次版本、主要版本和 zStream 版本，每个实时补丁的支持窗口已从 12 月减少到 6 个月。这意味着，在内核实时补丁发布的那天，它将覆盖过去 6 个月所交付的每个次要版本和计划勘误表内核。

有关此功能的更多信息，请参阅 [使用内核实时修补应用补丁](#)。

有关可用的内核实时补丁的详情，请查看 [内核实时补丁生命周期](#)。

([BZ#1958250](#))

crash-ptdump-command 软件包已被弃用

crash-ptdump-command 软件包（这是 **crash** 工具的 **ptdump** 扩展模块）已被弃用，且可能在以后的 RHEL 版本中不提供。在 **Single Range Output** 模式下工作时，**ptdump** 命令无法检索日志缓冲区，且只能在物理地址(ToPA)表模式下工作。**crash-ptdump-command** 当前没有在上游维护

([BZ#1838927](#))

使用无磁盘引导为 **Real Time 8** 安装 RHEL 现已弃用

无盘引导允许多个系统通过网络共享根文件系统。虽然方便，无盘引导会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](#))

Linux **firewire** 子系统及其关联的用户空间组件在 RHEL 8 中已被弃用

firewire 子系统提供了接口来使用和维护 IEEE 1394 总线上的任何资源。在 RHEL 9 中，**内核** 软件包将不再支持 **firewire**。请注意，**firewire** 包含几个由 **libavc1394**、**libdc1394**、**libraw1394** 软件包提供的用户空间组件。这些软件包也会被弃用。

([BZ#1871863](#))

rdma_rxe Soft-RoCE 驱动程序已被弃用

通过融合以太网(Soft-RoCE)的软件远程直接内存（也称为 RXE）是模拟远程直接内存访问(RDMA)的一个特性。在 RHEL 8 中，Soft-RoCE 特性作为一个不受支持的技术预览提供。但是，由于稳定性问题，此特性已被弃用，并将在 RHEL 9 中删除。

([BZ#1878207](#))

10.7. 引导加载程序

kernelopts 环境变量已被弃用

在 RHEL 8 中，使用 GRUB2 引导装载程序的系统的内核命令行参数定义在 **kernelopts** 环境变量中。变量保存在每个内核引导条目的 `/boot/grub2/grubenv` 文件中。但是，使用 **kernelopts** 存储内核命令行参数并不可靠。因此，在以后的 RHEL 主更新中，**kernelopts** 将被删除，且内核命令行参数会存储在 Boot Loader Specification(BLS)片断中。

([BZ#2060759](#))

10.8. 文件系统和存储

除 **async** 之外的 VDO 写模式已被弃用

VDO 支持 RHEL 8 中的几种写入模式：

- **sync**
- **async**
- **async-unsafe**
- **auto**

从 RHEL 8.4 开始，以下写入模式已弃用：

sync

VDO 层之上的设备无法识别 VDO 是否同步，因此设备无法利用 VDO 同步模式。

async-unsafe

VDO 添加了这个写入模式来作为降低 **async** 模式性能的一个临时解决方案，异步模式符合原子性、一致性、隔离性和持久性(ACID)。对于大多数用例，红帽不推荐使用 **async-unsafe**，也不知道任何依赖它的用户。

auto

这个写入模式只选择其它写入模式之一。当 VDO 只支持单个写入模式时，不再需要它。

这些写入模式将在以后的主 RHEL 发行版本中删除。

推荐的 VDO 写入模式现在是 **async** 的。

如需有关 VDO 写入模式的更多信息，请参阅[选择 VDO 写入模式](#)。

([JIRA:RHELPLAN-70700](#))

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](#))

cramfs 已被弃用

由于缺少用户，**cramfs** 内核模块已被弃用。建议将 **squashfs** 作为一个替代解决方案。

(BZ#1794513)

VDO 管理器已被弃用

基于 python 的 VDO 管理软件已被弃用，并将从 RHEL 9 中删除。在 RHEL 9 中，它将被 LVM-VDO 集成替代。因此，建议您使用 **lvcreate** 命令创建 VDO 卷。

使用 VDO 管理软件创建的现有卷可以使用 **lvm2** 软件包提供的 `/usr/sbin/lvm_import_vdo` 脚本来进行转换。有关 LVM-VDO 实现的更多信息，请参阅 [在 RHEL 上去重和压缩逻辑卷](#)。

(BZ#1949163)

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)

peripety 已被弃用

从 RHEL 8.3 开始，**pipety** 软件包已弃用。

Peripety 存储事件通知守护进程将系统存储日志解析为结构化的存储事件。它帮助您调查存储问题。

(BZ#1871953)

10.9. 高可用性和集群

支持 clutter 工具的 pcs 命令已被弃用

支持 **clutter** 工具来分析群集配置格式的 **pcs** 命令已被弃用。现在，这些命令会显示一个警告信息，提示该命令已弃用，并且与这些命令相关的部分已从 **pcs** 帮助显示和 **pcs(8)** 手册页中删除。

以下命令已被弃用：

- 用于导入 CMAN / RHEL6 HA 群集配置的 **pcs config import-cman**

- **pcs config export** 用于将集群配置导出到可重新创建同一集群的 **pcs** 命令列表中

(BZ#1851335)

10.10. 动态编程语言、网页和数据库服务器

PHP 提供的与 Apache HTTP 服务器一起使用的 **mod_php** 模块已被弃用

PHP 提供的与 RHEL 8 中的 Apache HTTP 服务器一起使用的 **mod_php** 模块可用，但在默认配置中未启用。RHEL 9 不再提供该模块。

从 RHEL 8 开始，PHP 脚本默认使用 FastCGI Process Manager (**php-fpm**) 运行。如需更多信息，请参阅 [将 PHP 与 Apache HTTP 服务器一起使用](#)。

(BZ#2225332)

10.11. 编译器和开发工具

libdwarf 已弃用

libdwarf 库在 RHEL 8 中已弃用。将来的主版本中可能也不支持该程序库。对于打算处理 ELF/DWARF 文件的应用程序，请使用 **elfutils** 和 **libdw** 库。

libdwarf-tools dwarfdump 程序的替代方案是 **binutils readelf** 程序或 **elfutils readelf** 程序，它们都通过传递 **--debug-dump** 标志来使用。

(BZ#1920624)

gdb.i686 软件包已弃用

在 RHEL 8.1 中，GNU Debugger (GDB) **gdb.i686** 的 32 位版本因为另一个软件包中的依赖问题而提供。因为 RHEL 8 不支持 32 位硬件，所以 **gdb.i686** 软件包从 RHEL 8.4 开始已弃用。GDB 的 64 位版本 **gdb.x86_64** 完全能够调试 32 位应用程序。

如果使用 **gdb.i686**，请注意以下重要问题：

- **gdb.i686** 软件包将不再更新。用户必须安装 **gdb.x86_64**。
- 如果您安装了 **gdb.i686**，安装 **gdb.x86_64** 将导致 **yum** 报告 **package gdb-8.2-14.el8.x86_64 obsoletes gdb < 8.2-14.el8 provided by gdb-8.2-12.el8.i686**。这是预期的。卸载 **gdb.i686** 或向 **dnf** 传递 **--allow-erase** 选项来删除 **gdb.i686**，并安装 **gdb.x86_64**。
- 用户将不能在 64 位系统中安装 **gdb.i686** 软件包，也就是那些带有 **libc.so.6(i686)** 软件包的系统。

(BZ#1853140)

10.12. 身份管理

openssh-ldap 已被弃用

在 Red Hat Enterprise Linux 8 中弃用 **openssh-ldap** 子软件包，并将在 RHEL 9 中删除。因为 **openssh-ldap** 子软件包没有被上游维护，红帽建议您使用 **SSSD** 和 **sss_ssh_authorizedkeys** 帮助程序，它们与其他 IdM 解决方案更好地集成且更安全。

默认情况下，SSSD `ldap` 和 `ipa` 供应商会读取用户对象的 `sshPublicKey` LDAP 属性（如果可用）。请注意，您无法为 `ad` provider 或 `IdM` 可信域使用默认的 SSSD 配置从 Active Directory(AD)检索 SSH 公钥，因为 AD 没有存储公钥的默认 LDAP 属性。

要允许 `sss_ssh_authorizedkeys` 帮助程序从 SSSD 获取密钥，在 `sssd.conf` 文件的 `services` 选项中添加 `ssh` 来启用 `ssh` 响应程序。详情请查看 [sssd.conf\(5\)](#) 手册页。

要允许 `sshd` 使用 `sss_ssh_authorizedkeys`，添加 `AuthorizedKeysCommand/usr/bin/sss_ssh_authorizedkeys` 和 `AuthorizedKeysCommandUser nobody` 选项到 `/etc/ssh/sshd_config` 文件，如 [ss_ssh_authorizedkeys\(1\)](#) 手册页所述。

([BZ#1871025](#))

已经删除了 DES 和 3DES 加密类型

由于安全考虑，自 RHEL 7 开始，数据加密标准(DES)算法已被弃用并默认禁用。通过最近重新构建 Kerberos 软件包，已从 RHEL 8 中删除了 `single-DES(DES)` 和 `triple-DES (3DES)` 加密类型。

如果您已经将服务或用户配置为只使用 DES 或 3DES 加密，您可能会遇到服务中断，例如：

- Kerberos authentication 错误
- **unknown enctype** 加密错误
- 带有 DES 加密数据库主密钥(K/M)的 KDC 无法启动

执行以下操作准备升级：

1. 检查您的 KDC 是否使用 DES 或者 3DES 加密，并使用 `krb5check` 打开源 Python 脚本。请参阅 GitHub 上的 [krb5check](#)。
2. 如果您要将 DES 或 3DES 加密用于任何 Kerberos 主体，请使用支持的加密类型重新加密，比如高级加密标准(AES)。有关重新打包的步骤，请参考 MIT Kerberos 文档中的 [Retiring DES](#)。
3. 通过在升级前临时设置以下 Kerberos 选项，从 DES 和 3DES 测试测试：
 - a. 在 KDC 上的 `/var/kerberos/krb5kdc/kdc.conf` 中设置 `support_encypes` 且不包含 `des` 或 `des3`。
 - b. 对于每个主机，在 `/etc/krb5.conf` 以及 `/etc/krb5.conf.d` 中的任意文件，将 `allow_weak_crypto` 设为 `false`。默认为 `false`。
 - c. 对于每个主机，在 `/etc/krb5.conf` 以及 `/etc/krb5.conf.d` 中的任何文件中，设置 `permitted_encypes`、`default_tgs_encypes` 和 `default_tkt_encypes`，但不包括 `des` 或 `des3`。
4. 如果您没有遇到上一步中测试 Kerberos 设置的服务中断，请删除并升级它们。升级到最新的 Kerberos 软件包后您不需要这些设置。

([BZ#1877991](#))

单独使用 ctdb 服务已弃用

从 RHEL 8.4 开始，建议客户仅在满足以下条件时使用 `ctdb` 集群 Samba 服务：

- `ctdb` 服务通过资源代理 `ctdb` 作为 `pacemaker` 资源进行管理。

- **ctdb** 服务使用包含由红帽 Gluster 存储产品提供的 GlusterFS 文件系统或 GFS2 文件系统的存储卷。

ctdb 服务的独立用例已被弃用，其不会包含在 Red Hat Enterprise Linux 的下一个主版本中。有关 Samba 支持政策的更多信息，请参阅 [RHEL Resilient Storage - ctdb 常规政策的支持政策](#)。

(BZ#1916296)

以 PDC 或 BDC 的形式运行 Samba 已被弃用

传统的域控制器模式使管理员能够作为类似 NT4 的主域控制器 (PDC) 和备份域控制器 (BDC) 运行 Samba。用于配置这些模式的代码和设置将在以后的 Samba 发行版本中删除。

只要 RHEL 8 中的 Samba 版本提供 PDC 和 BDC 模式,红帽就仅在带有支持 NT4 域的 Windows 版本的现有安装中支持这些模式。红帽建议不要设置新的 Samba NT4 域，因为 Microsoft 操作系统稍后于 Windows 7 和 Windows Server 2008 R2 不支持 NT4 域。

如果您使用 PDC 仅验证 Linux 用户，红帽建议迁移到 RHEL 订阅中包含的 [Red Hat Identity Management \(IdM\)](#)。但是，您无法将 Windows 系统加入到 IdM 域中。请注意，红帽继续支持在后台使用 PDC 功能 IdM。

红帽不支持将 Samba 作为 AD 域控制器 (DC) 运行。

(BZ#1926114)

通过 WinSync 与 IdM 间接集成的 AD 已被弃用

由于一些功能的限制，不会在 RHEL 8 中对 WinSync 进行积极的开发：

- WinSync 只支持一个活动目录(AD)域。
- 密码同步需要在 AD 域控制器上安装额外的软件。

对于具有更好的资源和安全分离的更健壮的解决方案，红帽建议对于与活动目录的间接集成使用 [跨林信任](#)。请参阅 [间接集成](#) 文档。

(JIRA:RHELPLAN-100400)

libwbclient 的 SSSD 版本已被删除

libwbclient 软件包的 SSSD 实现在 RHEL 8.4 中已被弃用。因为无法与 Samba 的最新版本一起使用，**libwbclient** 的 SSSD 实现现已被删除。

(BZ#1947671)

SMB1 协议在 Samba 中已弃用

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)

对 FreeRADIUS 的有限支持

在 RHEL 8 中，以下外部身份验证模块作为 FreeRADIUS 产品的一部分被弃用：

- MySQL、PostgreSQL、SQLite 和 unixODBC 数据库连接器

- Perl 语言模块
- REST API 模块



注意

PAM 身份验证模块以及其它作为基础软件包的一部分提供的身份验证模块不受影响。

您可以在社区支持的软件包中找到已弃用模块的替代品，例如在 Fedora 项目中。

另外，在以后的 RHEL 版本中，对 **freeradius** 软件包的支持范围将限制为以下用例：

- 使用 FreeRADIUS 作为无线身份验证提供商，身份管理(IdM)作为身份验证的后端源。身份验证通过 **krb5** 和 LDAP 身份验证软件包或作为主 FreeRADIUS 软件包中的 PAM 身份验证发生。
- 使用 FreeRADIUS，通过 Python 3 身份验证软件包对 IdM 中的身份验证提供真相来源。

与这些弃用相反，红帽将通过 FreeRADIUS 增强对以下外部身份验证模块的支持：

- 基于 **krb5** 和 LDAP 的身份验证
- **Python 3** 身份验证

这些集成选项的重点在于与红帽 IdM 的战略方向紧密一致。

(Jira:RHELDPCS-17573)

10.13. 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)

10.14. 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)

Motif 已被弃用

Motif 小部件工具包已在 RHEL 中被弃用，因为上游 Motif 社区的开发不活跃。

以下 Motif 软件包已被弃用，包括其开发和调试变体：

- **motif**
- **openmotif**
- **openmotif21**

- **openmotif22**

另外，**motif-static** 软件包已删除。

红帽建议使用 GTK 工具包作为替代品。与 Motif 相比，GTK 更易于维护，并提供了新功能。

(JIRA:RHELPLAN-98983)

10.15. 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](#))

10.16. RED HAT ENTERPRISE LINUX 系统角色

当在 RHEL 9 节点上配置 team 时，networking 系统角色显示一条弃用警告

RHEL 9 中弃用了网络协作功能。因此，在 RHEL 8 控制器上使用 **networking** RHEL 系统角色在 RHEL 9 节点上配置网络 team 时，显示一条有关其弃用的警告。

([BZ#2021685](#))

Ansible Engine 已被弃用

以前的 RHEL 8 版本提供对 Ansible Engine 存储库的访问（有限范围的支持），以启用支持的 RHEL Automation 用例，如 RHEL 系统角色和 Insights 补救措施。Ansible Engine 已被弃用，2023 年 9 月 29 日之后将不再支持 Ansible Engine 2.9。有关支持的用例的详情，请参阅 [RHEL 9 AppStream 中包含的 Ansible Core 软件包的支持范围](#)。

用户必须手动将他们的系统从 Ansible Engine 迁移到 Ansible Core。为此，请按照以下步骤操作：

流程

1. 检查系统是否在运行 RHEL 8.7:

```
# cat /etc/redhat-release
```

2. 卸载 Ansible Engine 2.9 :

```
# yum remove ansible
```

3. 禁用 **ansible-2-for-rhel-8-x86_64-rpms** 存储库 :

```
# subscription-manager repos --disable  
ansible-2-for-rhel-8-x86_64-rpms
```

4. 从 RHEL 8 AppStream 存储库安装 Ansible Core 软件包 :

```
# yum install ansible-core
```

如需了解更多详细信息，请参阅：[在 RHEL 8.6 及之后的版本中使用 Ansible](#)。

([BZ#2006081](#))

geoipupdate 软件包已弃用

geoipupdate 软件包需要第三方订阅，同时下载专有内容。因此，**geoipupdate** 软件包已被弃用，并将在下一个主要 RHEL 版本中删除。

([BZ#1874892](#))

10.17. 虚拟化

virsh iface-* 命令已被弃用

virsh iface-* 命令（如 **virsh iface-start** 和 **virsh iface-destroy**）现已被弃用，并将在以后的 RHEL 主版本中删除。另外，这些命令会因为配置依赖而经常失败。

因此，建议您不要使用 **virsh iface-*** 命令来配置和管理主机网络连接。反之，使用 NetworkManager 程序及其相关的管理应用程序，如 **nmcli**。

([BZ#1664592](#))

virt-manager 已被弃用

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

([JIRA:RHELPLAN-10304](#))

对虚拟机快照的支持有限

目前只对使用 UEFI 固件的虚拟机支持创建虚拟机(VM)的快照。另外，在快照操作过程中，QEMU 监控可能会被阻止，这会影响某些工作负载的 hypervisor 性能。

另请注意，创建虚拟机快照的当前机制已被弃用，红帽不推荐在生产环境中使用虚拟机快照。

([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，红帽建议使用 **stdvga** 或 **virtio-vga** 设备，而不是 **Cirrus VGA**。

([BZ#1651994](#))

IBM POWER 上的 KVM 已被弃用

在 IBM POWER 硬件中使用 KVM 虚拟化已被弃用。因此，RHEL 8 仍支持 IBM POWER 上的 KVM，但在以后的 RHEL 主发行版本中将不被支持。

([JIRA:RHELPLAN-71200](#))

使用基于 SHA1 的签名进行 SecureBoot 镜像验证已弃用

在 UEFI (PE/COFF) 可执行文件中使用基于 SHA1 的签名执行 SecureBoot 镜像验证已过时。反之，红帽建议使用基于 SHA2 算法或更新版本的签名。

(BZ#1935497)

使用 SPICE 将智能卡读取器附加到虚拟机已被弃用

SPICE 远程显示协议已在 RHEL 8 中被弃用。由于将智能卡读取器附加到虚拟机(VM)的唯一方法取决于 SPICE 协议，因此在虚拟机中使用智能卡已在 RHEL 8 中被弃用。

在 RHEL 未来的主版本中，将智能卡读取器附加到虚拟机的功能仅被第三方远程视觉解决方案支持。

(BZ#2059626)

SPICE 已被弃用

SPICE 远程显示协议已弃用。因此，RHEL 8 中仍支持 SPICE，但红帽建议对远程显示流使用备用解决方案：

- 要访问远程控制台，请使用 VNC 协议。
- 对于高级远程显示功能，请使用 RDP、HP RGS 或 Mechdyne TGX 等第三方工具。

请注意，SPICE 使用的 QXL 图形设备也已被弃用。(BZ#1849563)

10.18. 容器

基于 Podman varlink 的 API v1.0 已被删除

基于 Podman varlink 的 API v1.0 在之前的 RHEL 8 版本中已弃用。podman v2.0 引入了一个新的 Podman v2.0 RESTful API。Podman v3.0 发行版本中，基于 varlink 的 API v1.0 已被完全删除。

(JIRA:RHELPLAN-45858)

container-tools:1.0 已弃用

container-tools:1.0 模块已弃用，将不再接收安全更新。建议您使用较新支持的稳定模块流，如 **container-tools:2.0** 或 **container-tools:3.0**。

(JIRA:RHELPLAN-59825)

container-tools:2.0 模块已被弃用

container-tools:2.0 模块已被弃用，并将不再接受安全更新。建议您使用更新的支持的稳定模块流，如 **container-tools:3.0**。

(JIRA:RHELPLAN-85066)

除了 GIMP，Flatpak 镜像已弃用

rhel8/firefox-flatpak, **rhel8/thunderbird-flatpak**, **rhel8/inkscape-flatpak** 和 **rhel8/libreoffice-flatpak** RHEL 8 Flatpak 应用程序已被弃用，并被 RHEL 9 版本替代。**rhel8/gimp-flatpak** Flatpak 应用程序已被弃用，因为在 RHEL 9 中还没有替代品。

(BZ#2142499)

10.19. 已弃用的软件包

本节列出了已弃用的软件包，可能不会包括在 Red Hat Enterprise Linux 未来的主发行版本中。

有关 RHEL 7 和 RHEL 8 之间的软件包变化，请参阅 [采用 RHEL 8 的注意事项](#) 文档中的 [对软件包的更改](#)。

在 RHEL 8 生命周期结束前，以下软件包已被弃用，但仍然被支持：

- 389-ds-base-legacy-tools
- abrt
- abrt-addon-ccpp
- abrt-addon-kerneloops
- abrt-addon-pstoreoops
- abrt-addon-vmcore
- abrt-addon-xorg
- abrt-cli
- abrt-console-notification
- abrt-dbus
- abrt-desktop
- abrt-gui
- abrt-gui-libs
- abrt-libs
- abrt-tui
- adobe-source-sans-pro-fonts
- adwaita-qt
- alsa-plugins-pulseaudio
- amanda
- amanda-client
- amanda-libs
- amanda-server
- ant-contrib
- antlr3
- antlr32
- aopalliance
- apache-commons-collections

- apache-commons-compress
- apache-commons-exec
- apache-commons-jxpath
- apache-commons-parent
- apache-ivy
- apache-parent
- apache-resource-bundles
- apache-sshd
- apiguardian
- aspnetcore-runtime-3.0
- aspnetcore-runtime-3.1
- aspnetcore-runtime-5.0
- aspnetcore-targeting-pack-3.0
- aspnetcore-targeting-pack-3.1
- aspnetcore-targeting-pack-5.0
- assertj-core
- authd
- auto
- autoconf213
- autogen
- autogen-libopts
- awscli
- base64coder
- batik
- batik-css
- batik-util
- bea-stax
- bea-stax-api
- bind-export-devel

- bind-export-libs
- bind-libs-lite
- bind-pkcs11
- bind-pkcs11-devel
- bind-pkcs11-libs
- bind-pkcs11-utils
- bind-sdb
- bind-sdb
- bind-sdb-chroot
- bluez-hid2hci
- boost-jam
- boost-signals
- bouncycastle
- bpg-algeti-fonts
- bpg-chveulebrivi-fonts
- bpg-classic-fonts
- bpg-courier-fonts
- bpg-courier-s-fonts
- bpg-dedaena-block-fonts
- bpg-dejavu-sans-fonts
- bpg-elite-fonts
- bpg-excelsior-caps-fonts
- bpg-excelsior-condenced-fonts
- bpg-excelsior-fonts
- bpg-fonts-common
- bpg-glaho-fonts
- bpg-gorda-fonts
- bpg-ingiri-fonts
- bpg-irubaqidze-fonts

- bpg-mikhail-stephan-fonts
- bpg-mrgvlovani-caps-fonts
- bpg-mrgvlovani-fonts
- bpg-nateli-caps-fonts
- bpg-nateli-condenced-fonts
- bpg-nateli-fonts
- bpg-nino-medium-cond-fonts
- bpg-nino-medium-fonts
- bpg-sans-fonts
- bpg-sans-medium-fonts
- bpg-sans-modern-fonts
- bpg-sans-regular-fonts
- bpg-serif-fonts
- bpg-serif-modern-fonts
- bpg-ucnobi-fonts
- brlapi-java
- bsh
- buildnumber-maven-plugin
- byaccj
- call0n
- cbi-plugins
- cdparanoia
- cdparanoia-devel
- cdparanoia-libs
- cdrdao
- cmirror
- codehaus-parent
- codemodel
- compat-exiv2-026

- compat-guile18
- compat-hwloc1
- compat-libpthread-nonshared
- compat-libtiff3
- compat-openssl10
- compat-sap-c++-11
- compat-sap-c++-10
- compat-sap-c++-9
- createrepo_c-devel
- ctags
- ctags-etags
- custodia
- cyrus-imapd-vzic
- dbus-c++
- dbus-c++-devel
- dbus-c++-glib
- dbxtool
- dhcp-libs
- directory-maven-plugin
- directory-maven-plugin-javadoc
- dirsplit
- dleyna-connector-dbus
- dleyna-core
- dleyna-renderer
- dleyna-server
- dnssec-trigger
- dnssec-trigger-panel
- dotnet-apphost-pack-3.0
- dotnet-apphost-pack-3.1

- dotnet-apphost-pack-5.0
- dotnet-host-fxr-2.1
- dotnet-host-fxr-2.1
- dotnet-hostfxr-3.0
- dotnet-hostfxr-3.1
- dotnet-hostfxr-5.0
- dotnet-runtime-2.1
- dotnet-runtime-3.0
- dotnet-runtime-3.1
- dotnet-runtime-5.0
- dotnet-sdk-2.1
- dotnet-sdk-2.1.5xx
- dotnet-sdk-3.0
- dotnet-sdk-3.1
- dotnet-sdk-5.0
- dotnet-targeting-pack-3.0
- dotnet-targeting-pack-3.1
- dotnet-targeting-pack-5.0
- dotnet-templates-3.0
- dotnet-templates-3.1
- dotnet-templates-5.0
- dotnet5.0-build-reference-packages
- dptfextract
- drpm
- drpm-devel
- dump
- dvd+rw-tools
- dyninst-static
- eclipse-ecf

- eclipse-ecf-core
- eclipse-ecf-runtime
- eclipse-emf
- eclipse-emf-core
- eclipse-emf-runtime
- eclipse-emf-xsd
- eclipse-equinox-osgi
- eclipse-jdt
- eclipse-license
- eclipse-p2-discovery
- eclipse-pde
- eclipse-platform
- eclipse-swt
- ed25519-java
- ee4j-parent
- elfutils-devel-static
- elfutils-libelf-devel-static
- enca
- enca-devel
- environment-modules-compat
- evince-browser-plugin
- exec-maven-plugin
- farstream02
- felix-gogo-command
- felix-gogo-runtime
- felix-gogo-shell
- felix-scr
- felix-osgi-compendium
- felix-osgi-core

- felix-osgi-foundation
- felix-parent
- file-roller
- fipscheck
- fipscheck-devel
- fipscheck-lib
- firewire
- fonts-tweak-tool
- forge-parent
- freeradius-mysql
- freeradius-perl
- freeradius-postgresql
- freeradius-rest
- freeradius-sqlite
- freeradius-unixODBC
- fuse-sshfs
- fusesource-pom
- future
- gamin
- gamin-devel
- gavl
- gcc-toolset-10
- gcc-toolset-10-annobin
- gcc-toolset-10-binutils
- gcc-toolset-10-binutils-devel
- gcc-toolset-10-build
- gcc-toolset-10-dwz
- gcc-toolset-10-dyninst
- gcc-toolset-10-dyninst-devel

- gcc-toolset-10-elfutils
- gcc-toolset-10-elfutils-debuginfod-client
- gcc-toolset-10-elfutils-debuginfod-client-devel
- gcc-toolset-10-elfutils-devel
- gcc-toolset-10-elfutils-libelf
- gcc-toolset-10-elfutils-libelf-devel
- gcc-toolset-10-elfutils-libs
- gcc-toolset-10-gcc
- gcc-toolset-10-gcc-c++
- gcc-toolset-10-gcc-gdb-plugin
- gcc-toolset-10-gcc-gfortran
- gcc-toolset-10-gdb
- gcc-toolset-10-gdb-doc
- gcc-toolset-10-gdb-gdbserver
- gcc-toolset-10-libasan-devel
- gcc-toolset-10-libatomic-devel
- gcc-toolset-10-libitm-devel
- gcc-toolset-10-libsan-devel
- gcc-toolset-10-libquadmath-devel
- gcc-toolset-10-libstdc++-devel
- gcc-toolset-10-libstdc++-docs
- gcc-toolset-10-libtsan-devel
- gcc-toolset-10-libubsan-devel
- gcc-toolset-10-ltrace
- gcc-toolset-10-make
- gcc-toolset-10-make-devel
- gcc-toolset-10-perfutils
- gcc-toolset-10-runtime
- gcc-toolset-10-strace

- gcc-toolset-10-systemtap
- gcc-toolset-10-systemtap-client
- gcc-toolset-10-systemtap-devel
- gcc-toolset-10-systemtap-initscript
- gcc-toolset-10-systemtap-runtime
- gcc-toolset-10-systemtap-sdt-devel
- gcc-toolset-10-systemtap-server
- gcc-toolset-10-toolchain
- gcc-toolset-10-valgrind
- gcc-toolset-10-valgrind-devel
- gcc-toolset-9
- gcc-toolset-9-annobin
- gcc-toolset-9-build
- gcc-toolset-9-perftools
- gcc-toolset-9-runtime
- gcc-toolset-9-toolchain
- gcc-toolset-11-make-devel
- GConf2
- GConf2-devel
- gegl
- genisoimage
- genwqe-tools
- genwqe-vpd
- genwqe-zlib
- genwqe-zlib-devel
- geoipupdate
- geronimo-annotation
- geronimo-jms
- geronimo-jpa

- geronimo-parent-poms
- gfbgraph
- gflags
- gflags-devel
- glassfish-annotation-api
- glassfish-el
- glassfish-fastinfoset
- glassfish-jaxb-core
- glassfish-jaxb-txw2
- glassfish-jsp
- glassfish-jsp-api
- glassfish-legal
- glassfish-master-pom
- glassfish-servlet-api
- glew-devel
- glib2-fam
- glog
- glog-devel
- gmock
- gmock-devel
- gnome-abrt
- gnome-boxes
- gnome-menus-devel
- gnome-online-miners
- gnome-shell-extension-disable-screenshield
- gnome-shell-extension-horizontal-workspaces
- gnome-shell-extension-no-hot-corner
- gnome-shell-extension-window-grouper
- gnome-themes-standard

- gnu-free-fonts-common
- gnu-free-mono-fonts
- gnu-free-sans-fonts
- gnu-free-serif-fonts
- gnupg2-smime
- gnuplot
- gnuplot-common
- gobject-introspection-devel
- google-gson
- google-noto-sans-syriac-eastern-fonts
- google-noto-sans-syriac-estrangela-fonts
- google-noto-sans-syriac-western-fonts
- google-noto-sans-tibetan-fonts
- google-noto-sans-ui-fonts
- gphoto2
- gsl-devel
- gssntlmssp
- gtest
- gtest-devel
- gtkmm24
- gtkmm24-devel
- gtkmm24-docs
- gtksourceview3
- gtksourceview3-devel
- gtkspell
- gtkspell-devel
- gtkspell3
- guile
- gutenprint-gimp

- gutenprint-libs-ui
- gvfs-afc
- gvfs-afp
- gvfs-archive
- hamcrest-core
- hawtjni
- hawtjni
- hawtjni-runtime
- HdrHistogram
- HdrHistogram-javadoc
- highlight-gui
- hivex-devel
- hostname
- hplip-gui
- httpcomponents-project
- hwloc-plugins
- hyphen-fo
- hyphen-grc
- hyphen-hsb
- hyphen-ia
- hyphen-is
- hyphen-ku
- hyphen-mi
- hyphen-mn
- hyphen-sa
- hyphen-tk
- ibus-sayura
- icedax
- icu4j

- idm-console-framework
- iptables
- ipython
- isl
- isl-devel
- isorelax
- istack-commons-runtime
- istack-commons-tools
- iwl3945-firmware
- iwl4965-firmware
- iwl6000-firmware
- jacoco
- jaf
- jaf-javadoc
- jakarta-oro
- janino
- jansi-native
- jarjar
- java-1.8.0-ibm
- java-1.8.0-ibm-demo
- java-1.8.0-ibm-devel
- java-1.8.0-ibm-headless
- java-1.8.0-ibm-jdbc
- java-1.8.0-ibm-plugin
- java-1.8.0-ibm-src
- java-1.8.0-ibm-webstart
- java-1.8.0-openjdk-accessibility
- java-1.8.0-openjdk-accessibility-slowdebug
- java_cup

- java-atk-wrapper
- javacc
- javacc-maven-plugin
- javaewah
- javaparser
- javapoet
- javassist
- javassist-javadoc
- jaxen
- jboss-annotations-1.2-api
- jboss-interceptors-1.2-api
- jboss-logmanager
- jboss-parent
- jctools
- jdepend
- jdependency
- jdom
- jdom2
- jetty
- jetty-continuation
- jetty-http
- jetty-io
- jetty-security
- jetty-server
- jetty-servlet
- jetty-util
- jffi
- jflex
- jgit

- jline
- jmc
- jnr-netdb
- jolokia-jvm-agent
- js-uglify
- jsch
- json_simple
- jss-javadoc
- jtidy
- junit5
- jvnet-parent
- jzlib
- kernel-cross-headers
- ksc
- kurdit-unikurd-web-fonts
- kyotocabinet-libs
- ldapjdk-javadoc
- lensfun
- lensfun-devel
- lftp-scripts
- libaec
- libaec-devel
- libappindicator-gtk3
- libappindicator-gtk3-devel
- libatomic-static
- libavc1394
- libblocksruntime
- libcacard
- libcacard-devel

- libcgroup
- libcgroup-tools
- libchamplain
- libchamplain-devel
- libchamplain-gtk
- libcroco
- libcroco-devel
- libcxl
- libcxl-devel
- libdap
- libdap-devel
- libdazzle-devel
- libdbusmenu
- libdbusmenu-devel
- libdbusmenu-doc
- libdbusmenu-gtk3
- libdbusmenu-gtk3-devel
- libdc1394
- libdnet
- libdnet-devel
- libdv
- libdwarf
- libdwarf-devel
- libdwarf-static
- libdwarf-tools
- libeasyfc
- libeasyfc-gobject
- libepubgen-devel
- libertas-sd8686-firmware

- `libertas-usb8388-firmware`
- `libertas-usb8388-olpc-firmware`
- `libgdither`
- `libGLEW`
- `libgovirt`
- `libguestfs-benchmarking`
- `libguestfs-devel`
- `libguestfs-gfs2`
- `libguestfs-gobject`
- `libguestfs-gobject-devel`
- `libguestfs-java`
- `libguestfs-java-devel`
- `libguestfs-javadoc`
- `libguestfs-man-pages-ja`
- `libguestfs-man-pages-uk`
- `libguestfs-tools`
- `libguestfs-tools-c`
- `libhugetlbfs`
- `libhugetlbfs-devel`
- `libhugetlbfs-utils`
- `libIDL`
- `libIDL-devel`
- `libidn`
- `libiec61883`
- `libindicator-gtk3`
- `libindicator-gtk3-devel`
- `libiscsi-devel`
- `libjose-devel`
- `libkcc`

- libkkc-common
- libkkc-data
- libldb-devel
- liblogging
- libluksmeta-devel
- libmalaga
- libmcpp
- libmemcached
- libmemcached-libs
- libmetalink
- libmodulemd1
- libmongocrypt
- libmtp-devel
- libmusicbrainz5
- libmusicbrainz5-devel
- libnbd-devel
- liboauth
- liboauth-devel
- libpfm-static
- libpng12
- libpurple
- libpurple-devel
- libraw1394
- libreport-plugin-mailx
- libreport-plugin-rhtsupport
- libreport-plugin-ureport
- libreport-rhel
- libreport-rhel-bugzilla
- librpmem

- librpmem-debug
- librpmem-devel
- libsass
- libsass-devel
- libselinux-python
- libsqlite3x
- libtalloc-devel
- libtar
- libtdb-devel
- libtevent-devel
- libtpms-devel
- libunwind
- libusal
- libvarlink
- libverto-libevent
- libvirt-admin
- libvirt-bash-completion
- libvirt-daemon-driver-storage-gluster
- libvirt-daemon-driver-storage-iscsi-direct
- libvirt-devel
- libvirt-docs
- libvirt-gconfig
- libvirt-gobject
- libvirt-lock-sanlock
- libvirt-wireshark
- libvmem
- libvmem-debug
- libvmem-devel
- libvmmalloc

- libvmmalloc-debug
- libvmmalloc-devel
- libvncserver
- libwinpr-devel
- libwmf
- libwmf-devel
- libwmf-lite
- libXNVCtrl
- libyami
- log4j12
- log4j12-javadoc
- lohit-malayalam-fonts
- lohit-nepali-fonts
- lorax-composer
- lua-guestfs
- lucene
- lucene-analysis
- lucene-analyzers-smartcn
- lucene-queries
- lucene-queryparser
- Lucene-sandbox
- lz4-java
- lz4-java-javadoc
- mailman
- mailx
- make-devel
- malaga
- malaga-suomi-voikko
- marisa

- maven-antrun-plugin
- maven-assembly-plugin
- maven-clean-plugin
- maven-dependency-analyzer
- maven-dependency-plugin
- maven-doxia
- maven-doxia-sitetools
- maven-install-plugin
- maven-invoker
- maven-invoker-plugin
- maven-parent
- maven-plugins-pom
- maven-reporting-api
- maven-reporting-impl
- maven-resolver-api
- maven-resolver-connector-basic
- maven-resolver-impl
- maven-resolver-spi
- maven-resolver-transport-wagon
- maven-resolver-util
- maven-scm
- maven-script-interpreter
- maven-shade-plugin
- maven-shared
- maven-verifier
- maven-wagon-file
- maven-wagon-http
- maven-wagon-http-shared
- maven-wagon-provider-api

- maven2
- meanwhile
- mercurial
- mercurial-hgk
- meetis
- metis-devel
- mingw32-bzip2
- mingw32-bzip2-static
- mingw32-cairo
- mingw32-expat
- mingw32-fontconfig
- mingw32-freetype
- mingw32-freetype-static
- mingw32-gstreamer1
- mingw32-harfbuzz
- mingw32-harfbuzz-static
- mingw32-icu
- mingw32-libjpeg-turbo
- mingw32-libjpeg-turbo-static
- mingw32-libpng
- mingw32-libpng-static
- mingw32-libtiff
- mingw32-libtiff-static
- mingw32-openssl
- mingw32-readline
- mingw32-sqlite
- mingw32-sqlite-static
- mingw64-adwaita-icon-theme
- mingw64-bzip2

- mingw64-bzip2-static
- mingw64-cairo
- mingw64-expat
- mingw64-fontconfig
- mingw64-freetype
- mingw64-freetype-static
- mingw64-gstreamer1
- mingw64-harfbuzz
- mingw64-harfbuzz-static
- mingw64-icu
- mingw64-libjpeg-turbo
- mingw64-libjpeg-turbo-static
- mingw64-libpng
- mingw64-libpng-static
- mingw64-libtiff
- mingw64-libtiff-static
- mingw64-nettle
- mingw64-openssl
- mingw64-readline
- mingw64-sqlite
- mingw64-sqlite-static
- modello
- mojo-parent
- mongo-c-driver
- mousetweaks
- mozjs52
- mozjs52-devel
- mozjs60
- mozjs60-devel

- mozvoikko
- msv-javadoc
- msv-manual
- munge-maven-plugin
- mythes-mi
- mythes-ne
- nafees-web-naskh-fonts
- nbd
- nbdkit-devel
- nbdkit-example-plugins
- nbdkit-gzip-plugin
- nbdkit-plugin-python-common
- nbdkit-plugin-vddk
- ncompress
- ncurses-compat-libs
- net-tools
- netcf
- netcf-devel
- netcf-libs
- network-scripts
- network-scripts-ppp
- nkf
- nodejs-devel
- nodejs-packaging
- nss_nis
- nss-pam-ldapd
- objectweb-asm
- objectweb-asm-javadoc
- objectweb-pom

- ocaml-bisect-ppx
- ocaml-camlp4
- ocaml-camlp4-devel
- ocaml-lwt
- ocaml-mmap
- ocaml-ocplib-endian
- ocaml-ounit
- ocaml-result
- ocaml-seq
- opencryptoki-tpmtok
- opencv-contrib
- opencv-core
- opencv-devel
- openhpi
- openhpi-libs
- OpenIPMI-perl
- openssh-cavs
- openssh-ldap
- openssl-ibmpkcs11
- opentest4j
- os-maven-plugin
- pakchois
- pandoc
- paps-libs
- paranamer
- parfait
- parfait-examples
- parfait-javadoc
- pcp-parfait-agent

- pcp-pmda-rpm
- pcp-pmda-vmware
- pcsc-lite-doc
- peripety
- perl-B-Debug
- perl-B-Lint
- perl-Class-Factory-Util
- perl-Class-ISA
- perl-DateTime-Format-HTTP
- perl-DateTime-Format-Mail
- perl-File-CheckTree
- perl-homedir
- perl-libxml-perl
- perl-Locale-Codes
- perl-Mozilla-LDAP
- perl-NKF
- perl-Object-HashBase-tools
- perl-Package-DeprecationManager
- perl-Pod-LaTeX
- perl-Pod-Plainer
- perl-prefork
- perl-String-CRC32
- perl-SUPER
- perl-Sys-Virt
- perl-tests
- perl-YAML-Syck
- phodav
- php-recode
- php-xmlrpc

- pidgin
- pidgin-devel
- pidgin-sipe
- pinentry-emacs
- pinentry-gtk
- pipewire0.2-devel
- pipewire0.2-libs
- platform-python-coverage
- plexus-ant-factory
- plexus-bsh-factory
- plexus-cli
- plexus-component-api
- plexus-component-factories-pom
- plexus-components-pom
- plexus-i18n
- plexus-interactive
- plexus-pom
- plexus-velocity
- plymouth-plugin-throbgress
- pmreorder
- postgresql-test-rpm-macros
- powermock
- prometheus-jmx-exporter
- prometheus-jmx-exporter-openjdk11
- ptscotch-mpich
- ptscotch-mpich-devel
- ptscotch-mpich-devel-parmetis
- ptscotch-openmpi
- ptscotch-openmpi-devel

- purple-sipe
- pygobject2-doc
- pygtk2
- pygtk2-codegen
- pygtk2-devel
- pygtk2-doc
- python-nose-docs
- python-nss-doc
- python-podman-api
- python-psycopg2-doc
- python-pymongo-doc
- python-redis
- python-schedutils
- python-slip
- python-sqlalchemy-doc
- python-varlink
- python-virtualenv-doc
- python2-backports
- python2-backports-ssl_match_hostname
- python2-bson
- python2-coverage
- python2-docs
- python2-docs-info
- python2-funcsigs
- python2-ipaddress
- python2-mock
- python2-nose
- python2-numpy-doc
- python2-psycopg2-debug

- python2-psycopg2-tests
- python2-pymongo
- python2-pymongo-gridfs
- python2-pytest-mock
- python2-sqlalchemy
- python2-tools
- python2-virtualenv
- python3-bson
- python3-click
- python3-coverage
- python3-cpio
- python3-custodia
- python3-docs
- python3-flask
- python3-gevent
- python3-gobject-base
- python3-hivex
- python3-html5lib
- python3-hypothesis
- python3-ipatests
- python3-itsdangerous
- python3-jwt
- python3-libguestfs
- python3-mock
- python3-networkx-core
- python3-nose
- python3-nss
- python3-openipmi
- python3-pillow

- python3-ptyprocess
- python3-pydbus
- python3-pymongo
- python3-pymongo-gridfs
- python3-pyOpenSSL
- python3-pytml
- python3-reportlab
- python3-schedutils
- python3-scons
- python3-semantic_version
- python3-slip
- python3-slip-dbus
- python3-sqlalchemy
- python3-syspurpose
- python3-virtualenv
- python3-webencodings
- python3-werkzeug
- python38-asn1crypto
- python38-numpy-doc
- python38-psycopg2-doc
- python38-psycopg2-tests
- python39-numpy-doc
- python39-psycopg2-doc
- python39-psycopg2-tests
- qemu-kvm-block-gluster
- qemu-kvm-block-iscsi
- qemu-kvm-block-ssh
- qemu-kvm-hw-usbredir
- qemu-kvm-tests

- qpdf
- qpdf-doc
- qpid-proton
- qrencode
- qrencode-devel
- qrencode-libs
- qt5-qtcanvas3d
- qt5-qtcanvas3d-examples
- rarian
- rarian-compat
- re2c
- recode
- redhat-lsb
- redhat-lsb-core
- redhat-lsb-cxx
- redhat-lsb-desktop
- redhat-lsb-languages
- redhat-lsb-printing
- redhat-lsb-submod-multimedia
- redhat-lsb-submod-security
- redhat-lsb-supplemental
- redhat-lsb-trialuse
- redhat-menus
- redhat-support-lib-python
- redhat-support-tool
- reflections
- regexp
- relaxngDatatype
- rhsm-gtk

-
- rpm-plugin-priorreset
 - rpmemd
 - rsyslog-udp spoof
 - ruby-hivex
 - ruby-libguestfs
 - rubygem-abrt
 - rubygem-abrt-doc
 - rubygem-bson
 - rubygem-bson-doc
 - rubygem-bundler-doc
 - rubygem-mongo
 - rubygem-mongo-doc
 - rubygem-net-telnet
 - rubygem-xmlrpc
 - s390utils-cmsfs
 - samba-pidl
 - samba-test
 - samba-test-libs
 - samyak-devanagari-fonts
 - samyak-fonts-common
 - samyak-gujarati-fonts
 - samyak-malayalam-fonts
 - samyak-odia-fonts
 - samyak-tamil-fonts
 - sane-frontends
 - sanlk-reset
 - sat4j
 - scala
 - scotch

- scotch-devel
- SDL_sound
- selinux-policy-minimum
- sendmail
- sgabios
- sgabios-bin
- shrinkwrap
- sisu-inject
- sisu-mojos
- sisu-plexus
- skkdic
- SLOF
- smc-anjalioldlipi-fonts
- smc-dyuthi-fonts
- smc-fonts-common
- smc-kalyani-fonts
- smc-raghumalayalam-fonts
- smc-suruma-fonts
- softsm-devel
- sonatype-oss-parent
- sonatype-plugins-parent
- sos-collector
- sparsehash-devel
- spax
- spec-version-maven-plugin
- spice
- spice-client-win-x64
- spice-client-win-x86
- spice-glib

- spice-glib-devel
- spice-gtk
- spice-gtk-tools
- spice-gtk3
- spice-gtk3-devel
- spice-gtk3-vala
- spice-parent
- spice-protocol
- spice-qxl-wddm-dod
- spice-server
- spice-server-devel
- spice-qxl-xddm
- spice-server
- spice-streaming-agent
- spice-vdagent-win-x64
- spice-vdagent-win-x86
- sssd-libwbclient
- star
- stax-ex
- stax2-api
- stringtemplate
- stringtemplate4
- subscription-manager-initial-setup-addon
- subscription-manager-migration
- subscription-manager-migration-data
- subversion-javahl
- SuperLU
- SuperLU-devel
- supermin-devel

- swig
- swig-doc
- swig-gdb
- swtpm-devel
- swtpm-tools-pkcs11
- system-storage-manager
- tcl-brlapi
- testng
- tibetan-machine-uni-fonts
- timedatex
- tpm-quote-tools
- tpm-tools
- tpm-tools-pkcs11
- treelayout
- trousers
- trousers-lib
- tuned-profiles-compat
- tuned-profiles-nfv-host-bin
- tuned-utils-systemtap
- tycho
- uglify-js
- unbound-devel
- univocity-output-tester
- univocity-parsers
- usbguard-notifier
- usbredir-devel
- utf8cpp
- uthash
- velocity

- vinagre
- vino
- virt-dib
- virt-p2v-maker
- vm-dump-metrics-devel
- weld-parent
- wodim
- woodstox-core
- wqy-microhei-fonts
- wqy-unibit-fonts
- xdelta
- xmlgraphics-commons
- xmlstreambuffer
- xinetd
- xorg-x11-apps
- xorg-x11-drv-qxl
- xorg-x11-server-Xspice
- xpp3
- xsane-gimp
- xsom
- xz-java
- xz-java-javadoc
- yajl-devel
- yp-tools
- ypbind
- ypserv

10.20. 弃用和不维护的设备

本节列出了继续被支持的设备（驱动程序、适配器）

- ，直到 RHEL 8 生命周期结束，但可能在这个产品以后的主要发行本中不被支持，不建议在新的部署中使用。对列出的设备的支持不会改变。这些是 **弃用的** 设备。
- 可用，但不再在 RHEL 8 中进行常规测试或更新。红帽可自行决定修复严重的 bug，包括安全性 bug。这些设备不应再在生产环境中使用，很可能在下一个主发行版本中被禁用。这些是 **不维护的** 设备。

PCI 设备 ID 采用 `vendor:device:subvendor:subdevice` 的格式。如果没有列出设备 ID，则与相应驱动程序关联的所有设备都会被弃用。要在您的系统中检查硬件的 PCI ID，请运行 `lspci -nn` 命令。

表 10.1. 弃用的设备

设备 ID	驱动	设备名称
	bnx2	QLogic BCM5706/5708/5709/5716 驱动程序
	hpsa	惠普公司: 智能阵列控制器
0x10df:0x0724	lpfc	Emulex 公司 : OneConnect FCoE Initiator(Skyhawk)
0x10df:0xe200	lpfc	Emulex 公司 : LPe15000/LPe16000 系列 8Gb/16Gb 光纤通道适配器
0x10df:0xf011	lpfc	Emulex 公司 : Saturn: LightPulse 光纤通道主机适配器
0x10df:0xf015	lpfc	Emulex 公司 : Saturn: LightPulse 光纤通道主机适配器
0x10df:0xf100	lpfc	Emulex 公司 : LPe12000 系列 8Gb 光纤通道适配器
0x10df:0xfc40	lpfc	Emulex 公司 : Saturn-X: LightPulse 光纤通道主机适配器
0x10df:0xe220	be2net	Emulex 公司 : OneConnect NIC(Lancer)
0x1000:0x005b	megaraid_sas	Broadcom / LSI: MegaRAID SAS 2208 [Thunderbolt]
0x1000:0x006E	mpt3sas	Broadcom / LSI: SAS2308 PCI-Express Fusion-MPT SAS-2
0x1000:0x0080	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0081	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0082	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0083	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0084	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0085	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2

设备 ID	驱动	设备名称
0x1000:0x0086	mpt3sas	Broadcom / LSI: SAS2308 PCI-Express Fusion-MPT SAS-2
0x1000:0x0087	mpt3sas	Broadcom / LSI: SAS2308 PCI-Express Fusion-MPT SAS-2
	myri10ge	Myricom 10G 驱动程序(10GbE)
	netxen_nic	QLogic/NetXen(1/10)GbE 智能以太网驱动程序
0x1077:0x2031	qla2xxx	QLogic 公司: 基于 ISP8324 的 16Gb 光纤通道到 PCI Express 适配器
0x1077:0x2532	qla2xxx	QLogic 公司: 基于 ISP2532 的 8Gb 光纤通道到 PCI Express 适配器
0x1077:0x8031	qla2xxx	QLogic 公司: 8300 系列 10GbE 聚合网络适配器(FCoE)
	qla3xxx	QLogic ISP3XXX 网络驱动程序 v2.03.00-k5
0x1924:0x0803	sfc	Solarflare 通信: SFC9020 10G 以太网控制器
0x1924:0x0813	sfc	Solarflare 通信: SFL9021 10GBASE-T 以太网控制器
	Soft-RoCE (rdma_rxe)	
	HNS-RoCE	HNS GE/10GE/25GE/50GE/100GE RDMA Network Controller
	liquidio	Cavium LiquidIO 智能服务器适配器驱动程序
	liquidio_vf	Cavium LiquidIO 智能服务器适配器虚拟功能驱动程序

表 10.2. 不维护的设备

设备 ID	驱动	设备名称
	e1000	Intel® PRO/1000 网络驱动程序
	mptbase	Fusion MPT SAS 主机驱动程序

设备 ID	驱动	设备名称
	mptsas	Fusion MPT SAS 主机驱动程序
	mptscsih	Fusion MPT SCSI 主机驱动程序
	mptspi	Fusion MPT SAS 主机驱动程序
0x1000:0x0071 ^[a]	megaraid_sas	Broadcom / LSI: MR SAS HBA 2004
0x1000:0x0073 ^[a]	megaraid_sas	Broadcom / LSI: MegaRAID SAS 2008 [Falcon]
0x1000:0x0079 ^[a]	megaraid_sas	Broadcom / LSI: MegaRAID SAS 2108 [Liberator]
	nvmet_tcp	NVMe/TCP 目标驱动程序

[a] 在 RHEL 8.0 中禁用了，由于客户请求，在 RHEL 8.4 中被重新启用了。

第 11 章 已知问题

这部分论述了 Red Hat Enterprise Linux 8.7 中已知的问题。

11.1. 安装程序和镜像创建

在具有 LPAR 并启用了 secure boot 的 IBM Power 10 系统上安装失败

RHEL 安装程序没有与 IBM Power 10 系统上静态密钥安全引导集成。因此，当使用 secure 引导选项启用逻辑分区(LPAR)时，安装会失败并显示错误，**Unable to proceed with RHEL-x.x Installation**。

要临时解决这个问题，请安装 RHEL 但不启用 secure boot。引导系统后：

1. 使用 **dd** 命令将签名的内核复制到 PReP 分区。
2. 重启系统并启用 secure boot。

一旦固件验证了引导加载程序和内核，系统就可以成功引导。

如需更多信息，请参阅 <https://www.ibm.com/support/pages/node/6528884>

(BZ#2025814)

在 Anaconda 作为应用程序运行的系统上意外的 SELinux 策略

当 Anaconda 作为应用程序运行在已安装的系统上（例如，使用 **-image anaconda** 选项对镜像文件执行另一次安装）时，不禁止系统在安装过程中修改 SELinux 类型和属性。因此，某些 SELinux 策略的元素可能会在运行 Anaconda 的系统上发生更改。要临时解决这个问题，请不要在生产环境系统上运行 Anaconda，而在临时虚拟机中执行它。因此，生产系统上的 SELinux 策略没有被修改。作为系统安装过程的一部分运行 anaconda，如从 **boot.iso** 或 **dvd.iso** 安装不会受此问题的影响。

(BZ#2050140)

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)

USB CD-ROM 驱动器作为 Anaconda 中的安装源不可用

当源为 USB CD-ROM 驱动器，并且指定了 Kickstart **ignoredisk --only-use=** 命令时，安装会失败。在这种情况下，Anaconda 无法找到并使用这个源磁盘。

要临时解决这个问题，请使用 `harddrive --partition=sdX --dir=/` 命令从 USB CD-ROM 驱动器安装。因此，安装不会失败。

(BZ#1914955)

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)

使用 iso9660 文件系统的硬盘分区安装失败

您不能在使用 `iso9660` 文件系统进行分区的系统中安装 RHEL。这是因为将设置为忽略包含 `iso9660` 文件系统分区的硬盘的更新安装代码。即使在没有使用 DVD 的情况下安装 RHEL，也会发生这种情况。

要解决这个问题，请在 `kickstart` 文件中添加以下脚本，以在安装开始前格式化磁盘。

注：在执行临时解决方案前，请备份磁盘上的数据。`erafs` 命令对磁盘中的所有现有数据进行格式化。

```
%pre
wipefs -a /dev/sda
%end
```

因此，安装可以正常工作，且没有任何错误。

(BZ#1929105)

具有 HASH MMU 模式的 IBM Power 系统无法引导，显示内存分配失败

具有 `HASH 内存分配单元(MMU)` 模式的 IBM Power Systems 最多支持 192 个核的 `kdump`。因此，如果在超过 192 个核上启用了 `kdump`，则系统会无法引导，显示内存分配失败。这个限制是因为 `HASH MMU` 模式下早期引导过程中的 RMA 内存分配。要临时解决这个问题，请使用启用了 `fadump` 的 `Radix MMU` 模式，而不是使用 `kdump`。

(BZ#2028361)

RHEL for Edge 安装程序镜像在安装 rpm-ostree 有效负载时无法创建挂载点

当部署 `rpm-ostree` 有效负载时，例如在 RHEL for Edge 安装程序镜像中，安装程序不会为自定义分区正确创建一些挂载点。因此，安装会中止，并报以下错误：

```
The command 'mount --bind /mnt/sysimage/data /mnt/sysroot/data' exited with the code 32.
```

要临时解决这个问题：

- 使用自动分区方案，且不手动添加任何挂载点。
- 只在 `/var` 目录中手动分配挂载点。例如：`/var/my-mount-point` 和以下标准目录：`/`、`/boot`、`/var`。

因此，安装过程成功完成。

(BZ#2126506)

composer-cli compose 的 --size 参数将值视为字节，而不是 MiB

当使用 `composer-cli compose start --size size_value blueprint_name image_type` 命令时，`--size` 参数应该以 MiB 格式使用其参数。但是，设置中的一个 bug 导致 `composer-cli` 工具将此参数视为字节。

要临时解决这个问题，将大小值乘以 1048576。或者，在蓝图中使用。自定义允许对文件系统更精细的控制，并接受 MiB 或 GiB 这样的单位。请参阅 [支持的镜像自定义](#)。

([BZ#2033192](#))

11.2. 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](#))

11.3. 软件管理

cr_compress_file_with_stat () 可能会导致内存泄漏

`createrepo_c` C 库包含 API `cr_compress_file_with_stat ()` 函数。这个函数被声明为以 `char **dst` 作为第二个参数。根据其他参数，`cr_compress_file_with_stat ()` 要么使用 `dst` 作为输入参数，要么使用它来返回一个分配的字符串。这种无法预测的行为可能会导致内存泄漏，因为它不会通知用户何时释放 `dst` 内容。

要临时解决这个问题，增加了一个新的 API `cr_compress_file_with_stat_v2` 函数，它只使用 `dst` 参数作为输入。它被声明为 `char *dst`。这可防止内存泄漏。

请注意，`cr_compress_file_with_stat_v2` 函数为临时的，仅存在于 RHEL 8 中。之后，`cr_compress_file_with_stat ()` 将会被修复。

([BZ#1973588](#))

当 scriptlet 失败时，YUM 事务被报告为成功

从 RPM 版本 4.6 开始，允许安装后 scriptlet 失败，而不会对事务造成致命影响。这个行为也被传播到 YUM。这导致 scriptlets 可能会偶尔失败，而整个软件包事务报告为成功。

目前还没有可用的临时解决方案。

请注意，这是预期行为，其在 RPM 和 YUM 之间保持一致。scriptlet 中的任何问题都应在软件包层面上解决。

([BZ#1986657](#))

对通过升级更改其架构的软件包的安全 YUM 升级会失败

与 [RHBA-2022:7711](#) 公告一起发布的 [BZ#2088149](#) 的补丁引入了以下回归：对通过升级将其架构从或 `noarch` 更改为 `noarch` 的使用安全过滤器的 YUM 升级会失败。因此，它可以使系统处于脆弱的状态。

要临时解决这个问题，执行没有安全过滤器的常规升级。

([BZ#2088149](#))

11.4. SHELL 和命令行工具

ipmitool 与某些服务器平台不兼容

ipmitool 工具为监控、配置和管理支持智能平台管理接口(IPMI)的设备提供服务。**ipmitool** 的当前版本默认使用 Cipher Suite 17，而不是之前的 Cipher Suite 3。因此，**ipmitool** 无法与在协商过程中与声称支持 Cipher Suite 17 的某些裸机节点通信，但实际上不支持这个加密套件。因此，**ipmitool** 会中止，并发出 **no matching cipher suite** 错误消息。

如需了解更多详细信息，请参阅相关的 [知识库文章](#)。

要解决这个问题，请更新您的基板管理控制器(BMC)固件以使用 Cipher Suite 17。

另外，如果 BMC 固件更新不可用，您可以通过强制 **ipmitool** 使用特定加密套件来临时解决这个问题。当使用 **ipmitool** 调用管理任务时，向 **ipmitool** 命令添加 **-C** 选项，以及您要使用的加密套件的 *编号*。请参见以下示例：

```
# ipmitool -l lanplus -H _myserver.example.com_ -P _mypass_ -C 3 chassis power status
```

([BZ#1873614](#))

在不使用清洁磁盘进行恢复时，ReaR 无法重新创建卷组

当您恢复到包含现有数据的磁盘时，ReaR 无法执行恢复。

要临时解决这个问题，如果磁盘之前使用过，请在恢复之前手动擦除磁盘。要在救援环境中擦除磁盘，请在运行 **rear recover** 命令前使用以下命令之一：

- 使用 **dd** 命令覆盖磁盘。
- 使用带有 **-a** 标志的 **wipefs** 命令来清除所有可用的元数据。

请参见以下擦除 **/dev/sda** 磁盘的元数据的示例：

```
# wipefs -a /dev/sda[1-9] /dev/sda
```

此命令首先从 **/dev/sda** 上的分区擦除元数据，然后是分区表本身。

([BZ#1925531](#))

coreutils 可能会报告误导性的 EPERM 错误代码

GNU Core 工具(**coreutils**)使用 **statx ()** 系统调用启动了。如果 **seccomp** 过滤器返回一个未知系统调用的 EPERM 错误代码，**coreutils** 可能会报告误导性 EPERM 错误代码，因为无法将 EPERM 与正常工作的 **statx ()** 系统调用所返回的实际的 *Operation not permitted* 错误区分开。

要临时解决这个问题，请更新 **seccomp** 过滤器，使其要么允许 **statx ()** 系统调用，要么对其它不知道的系统调用返回 ENOSYS 错误代码。

([BZ#2030661](#))

11.5. 基础设施服务

FIPS 模式中的 postfix TLS 指纹算法需要改为 SHA-256

默认情况下，在 RHEL 8 中，**postfix** 使用带有 TLS 的 MD5 指纹来进行向后兼容。但是在 FIPS 模式中，MD5 哈希功能不可用，这可能会导致 TLS 在默认 postfix 配置中不正确正常工作。要解决这个问题，在 postfix 配置文件中需要将哈希功能改为 SHA-256。

如需了解更多详细信息，请参阅 [Fix postfix TLS in the FIPS mode by switching to SHA-256 instead of MD5](#)。

([BZ#1711885](#))

在将 `--delete` 和 `--filter '-x string.*'` 选项一起使用时，`rsync` 会失败

用于传输和同步文件的 `rsync` 工具无法正确处理 RHEL 8 中的扩展属性。因此，如果您将 `--delete` 选项与扩展属性的 `--filter '-x string.*'` 选项一起传给 `rsync` 命令，并且系统上的文件满足正则表达式，则会发生协议不兼容错误。例如，如果您使用 `--filter '-x system.*'` 选项，过滤器会找到您系统上存在的 `system.mwmrc` 文件，则 `rsync` 会失败。请参阅使用 `--filter '-x system.*'` 选项后发生的以下错误信息：

```
# /usr/bin/rsync -a --delete --filter '-x system.*' / 192.0.2.2::some/test/dir/
ERROR: rejecting excluded file-list name: path/to/excluded/system.mwmrc
rsync error: protocol incompatibility (code 2) at flist.c(912) [receiver=3.1.3]
rsync error: protocol incompatibility (code 2) at io.c(1649) [generator=3.1.3]
```

要防止这个问题，请对扩展属性谨慎使用正则表达式。

([BZ#2139118](#))

`brlitty` 软件包与 `multilib` 不兼容

不可能同时安装 32 位和 64 位版本的 `brlitty` 软件包。您要么安装 32 位 (`brlitty.i686`)，要么安装 64 位 (`brlitty.x86_64`) 版本的软件包。建议使用 64 位版本。

([BZ#2008197](#))

11.6. 安全

`/etc/passwd-` 的文件权限与 CIS RHEL 8 基准 1.0.0 不一致

由于 CIS Benchmark 存在问题，修正 SCAP 规则可以确保 `/etc/passwd-` backup 文件中的权限被配置为 `0644`。但是 **CIS Red Hat Enterprise Linux 8 Benchmark 1.0.0** 需要该文件的文件权限 `0600`。因此，在修复后，`/etc/passwd-` 的文件权限与基准数据不一致。

([BZ#1858866](#))

`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 `yum install libselinux-python` 命令，默认的 RHEL 8 软件仓库中不再提供 `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：

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

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

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

因此，您可以使用相关的模块安装 `libselenium-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，只有当使用上述的临时解决方案时，`udica` 才可以为 UBI 8 容器生成一个 SELinux 策略。

(BZ#1763210)

`/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 会导致一个进程，在这个进程中，内核启动时启用了 SELinux，并在稍后的启动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)

`sshd -T` 提供关于 Ciphers、MAC 和 KeX 算法的不准确的信息

`sshd -T` 命令的输出不包含系统范围的加密策略配置或可能来自于环境文件 `/etc/sysconfig/sshd` 的其他选项，它们作为 `sshd` 命令的参数被应用。这种情况的发生是因为上游 OpenSSH 项目不支持 `Include` 指令，以支持 RHEL 8 中红帽提供的加密默认值。在使用 `EnvironmentFile` 启动 `sshd.service` 单元服务的过程中，加密策略作为命令行参数被应用到 `sshd` 可执行文件中。要临时解决这个问题，请对环境文件使用 `source` 命令，并将加密策略作为参数传给 `sshd` 命令，如 `sshd -T $CRYPTO_POLICY`。如需更多信息，请参阅 [Ciphers、MAC 或 KeX 算法与 sshd -T 的不同](#)，以了解当前加密策略级别所提供的内容。因此，`sshd -T` 的输出与当前配置的加密策略匹配。

(BZ#2044354)

FIPS 模式中的 `openssl` 只接受特定的 D-H 参数

在 FIPS 模式中，使用 OpenSSL 的 TLS 客户端返回一个 `bad dh value` 错误，并中止与使用手动生成参数的服务器的 TLS 连接。这是因为 OpenSSL 当配置为符合 FIPS 140-2 时，只可用于符合 NIST SP 800-56A rev3 附加 D (RFC 3526 中定义的组 14、15、16、17 和 18，以及 RFC 7919) 中定义的组。另，使用 OpenSSL 的服务器会忽略所有其他参数，并选择类似大小的已知参数。要临时解决这个问题，请只使用兼容的组。

(BZ#1810911)

`crypto-policies` 错误地允许 Camellia 密码

RHEL 8 系统范围的加密策略应该在所有策略级别禁用 Camellia 密码，如产品文档中所述。但是 Kerberos 协议默认启用密码。

要临时解决这个问题，请应用 `NO-CAMELLIA` 子策略：

```
# update-crypto-policies --set DEFAULT:NO-CAMELLIA
```

在上个命令中，如果您之前已从 `DEFAULT` 进行了切换，请将 `DEFAULT` 替换为加密级别名称。

因此，只有在您通过临时解决方案禁用系统范围的加密策略的所有应用程序中，Camellia 密码才会被正确禁止。

([BZ#1919155](#))

通过 OpenSC pkcs15-init 的智能卡配置过程无法正常工作

`file_caching` 选项在默认的 OpenSC 配置中是启用的，文件缓存功能无法正确处理 `pkcs15-init` 工具中的一些命令。因此，通过 OpenSC 进行智能卡置备过程会失败。

要临时解决这个问题，请在 `/etc/opensc.conf` 文件中添加以下代码段：

```
app pkcs15-init {
    framework pkcs15 {
        use_file_caching = false;
    }
}
```

只有在应用了前面描述的临时解决方案时，通过 `pkcs15-init` 进行智能卡配置才能正常工作。

([BZ#1947025](#))

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](#))

TCP 连接上的 IKE 无法在自定义的 TCP 端口上工作

`tcp-remoteport` Libreswan 配置选项无法正常工作。因此，当场景需要指定非默认 TCP 端口时，TCP 连接上的 IKE 无法建立。

([BZ#1989050](#))

在安装过程中强化系统时，RHV hypervisor 可能无法正常工作

安装 Red Hat Virtualization Hypervisor(RHV-H),并应用 Red Hat Enterprise Linux 8 STIG 配置文件时，OSCAP Anaconda Add-on 可将系统强化为 RHEL 而不是 RVH-H，并删除 RHV-H 的基本软件包。因此，RHV hypervisor 可能无法工作。要临时解决这个问题，请在不应用任何配置文件强化的情况下安装 RHV-H 系统，并在安装完成后使用 OpenSCAP 来应用配置文件。因此，RHV hypervisor 可以正常工作。

([BZ#2075508](#))

红帽以压缩格式提供 CVE OVAL 报告

红帽以 `bzip2` 压缩格式提供 CVE OVAL 源，不再提供 XML 文件格式。RHEL 8 源的位置已相应更新，以反映这个变化。因为引用压缩的内容不是标准化的，因此第三方 SCAP 扫描程序在扫描使用源的规则时可能会遇到问题。

([BZ#2028428](#))

某些 SSG 中的规则组可能会失败

由于未定义规则及其依赖项的顺序，基准中 **SCAP 安全指南** (SSG) 规则的修复可能会失败。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](#))

Server with GUI 和 Workstation 安装可能无法使用 CIS 服务器配置文件

CIS Server Level 1 和 Level 2 安全配置文件与 **Server with GUI** 和 **Workstation** 软件选择不兼容。因此，无法使用 **Server with GUI** 软件选择和 CIS Server 配置文件进行 RHEL 8 安装。使用 CIS Server Level 1 或 Level 2 配置以及这些软件选择之一尝试安装会产生错误信息：

```
package xorg-x11-server-common has been added to the list of excluded packages, but it can't be removed from the current software selection without breaking the installation.
```

如果您需要根据 CIS 基准使系统与 **Server with GUI** 或 **Workstation** 软件选择保持一致，请使用 CIS Workstation Level 1 或 Level 2 配置文件。

([BZ#1843932](#))

Kickstart 在 RHEL 8 中使用 org_fedora_oscaped 而不是 com_redhat_oscaped

Kickstart 将 Open Security Content Automation Protocol (OSCAP) Anaconda 附加组件引用为 **org_fedora_oscaped** 而不是 **com_redhat_oscaped**，后者可能会导致混淆。这是向后兼容 Red Hat Enterprise Linux 7 所必需的。

([BZ#1665082](#))

STIG 配置文件中的 SSH 超时规则配置了不正确的选项

OpenSSH 的更新会影响以下国防信息系统局安全技术实施指南 (DISA STIG) 概要中的规则：

- DISA STIG for RHEL 8 (**xccdf_org.ssgproject.content_profile_stig**)
- DISA STIG with GUI for RHEL 8 (**xccdf_org.ssgproject.content_profile_stig_gui**)

在每个配置文件中，以下两条规则都会受到影响：

```
Title: Set SSH Client Alive Count Max to zero
CCE Identifier: CCE-83405-1
Rule ID: xccdf_org.ssgproject.content_rule_sshd_set_keepalive_0
STIG ID: RHEL-08-010200
```

```
Title: Set SSH Idle Timeout Interval
CCE Identifier: CCE-80906-1
Rule ID: xccdf_org.ssgproject.content_rule_sshd_set_idle_timeout
STIG ID: RHEL-08-010201
```

当应用到 SSH 服务器时，每个规则都会配置一个选项 (**ClientAliveCountMax** 和 **ClientAliveInterval**)，其行为不再像之前一样。因此，当 OpenSSH 达到这些规则配置的超时时间时，OpenSSH 不再断开空闲的 SSH 用户。作为临时解决方案，这些规则已从 DISA STIG for RHEL 8 和 DISA STIG with GUI for RHEL 8 中临时删除，直到开发出解决方案为止。

([BZ#2038977](#))

某些审计规则的 Bash 补救无法正常工作

以下 SCAP 规则的 SCAP 安全指南 (SSG) Bash 补救不会添加审计密钥：

- `audit_rules_login_events`
- `audit_rules_login_events_faillock`
- `audit_rules_login_events_lastlog`
- `audit_rules_login_events_tallylog`
- `audit_rules_usergroup_modification`
- `audit_rules_usergroup_modification_group`
- `audit_rules_usergroup_modification_gshadow`
- `audit_rules_usergroup_modification_opasswd`
- `audit_rules_usergroup_modification_passwd`
- `audit_rules_usergroup_modification_shadow`
- `audit_rules_time_watch_localtime`
- `audit_rules_mac_modification`
- `audit_rules_networkconfig_modification`
- `audit_rules_sysadmin_actions`
- `audit_rules_session_events`
- `audit_rules_sudoers`
- `audit_rules_sudoers_d`

因此，修复脚本修复了补救规则中的访问位和路径，但没有 Audit 密钥的规则不符合 OVAL 检查。因此，在此类规则修复后进行扫描会报告 **FAIL**。要临时解决这个问题，请手动向受影响的规则添加密钥。

([BZ#2119356](#))

某些 rsyslog 优先级字符串不能正常工作

对允许对加密进行精细控制的 `imtcp` 的 GnuTLS 优先级字符串的支持不完整。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](#))

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)

在 kickstart 安装过程中修复与服务相关的规则可能会失败

在 kickstart 安装过程中，OpenSCAP 工具有时会错误地显示服务的 **enable** 或 **disable** 状态补救不需要。因此，OpenSCAP 可能会将安装的系统上的服务设置为不合规的状态。作为临时解决方案，您可以在 kickstart 安装后扫描并修复该系统。这可以解决与服务相关的问题。

(BZ#1834716)

11.7. 网络

NetworkManager 不支持以特定顺序激活绑定和团队端口

NetworkManager 按接口名称的字母顺序激活接口。但是，如果接口在启动过程中稍后出现，例如，由于内核需要更多时间发现它，NetworkManager 会稍后激活这个接口。NetworkManager 不支持对绑定和团队端口设置优先级。因此，NetworkManager 激活这些设备的端口的顺序始终无法预测。要临时解决这个问题，请编写一个调度程序脚本。

有关此类脚本的示例，请参阅问题单中相应的 [注释](#)。

(BZ#1920398)

nm-cloud-setup 服务从接口中删除了手动配置的辅助 IP 地址

根据从云环境收到的信息，**nm-cloud-setup** 服务配置网络接口。禁用 **nm-cloud-setup** 以手动配置接口。然而，在某些情况下，主机上的其他服务也可以配置接口。例如，这些服务可以添加辅助 IP 地址。为了避免 **nm-cloud-setup** 删除辅助 IP 地址：

1. 停止并禁用 **nm-cloud-setup** 服务和计时器：

```
# systemctl disable --now nm-cloud-setup.service nm-cloud-setup.timer
```

2. 显示可用的连接配置文件：

```
# nmcli connection show
```

3. 重新激活受影响的连接配置文件：

```
# nmcli connection up "<profile_name>"
```

因此，该服务不再从接口中删除手动配置的辅助 IP 地址。

(BZ#2132754)

启用了 IPv6_rpfilter 选项的系统的网络吞吐量较低

在 **firewalld.conf** 文件中启用了 **IPv6_rpfilter** 选项的系统目前在高流量场景（如 100-Gbps 连接）中性能欠佳，网络吞吐量较低。要临时解决这个问题，请禁用 **IPv6_rpfilter** 选项。为此，请在 **/etc/firewalld/firewalld.conf** 文件中添加以下行：

IPv6_rpfilter=no

因此，系统性能更好，但安全性也有所降低。

(BZ#1871860)

由于网络接口名称的意外更改，IBM Z 上的 RoCE 接口会丢失其 IP 设置

在 RHEL 8.6 及更早版本中，**udev** 设备管理器在 IBM Z 平台上将不可预测的设备名称分配给由唯一标识符(UID)枚举的 RoCE 接口。但是，在 RHEL 8.7 及更高版本中，**udev** 将带有 **eno** 前缀的可预测设备名称分配给这些接口。

如果您从 RHEL 8.6 或更早版本更新至 8.7 或更高版本，则这些 UID 枚举的接口有新名称，且不再与 NetworkManager 连接配置文件中的设备名称匹配。因此，这些接口在更新后没有 IP 配置。

如果您已经更新了系统，有关在更新前可以应用的临时解决方案和修复，请参阅 [IBM Z 上的 RoCE 接口在更新至 RHEL 8.7 或更高版本后丢失其 IP 设置](#)。

(BZ#2169382)

11.8. 内核

IBM Power 系统上的安全引导不支持迁移

目前在 IBM Power 系统上，在成功迁移物理卷(PV)后不会引导逻辑分区(LPAR)。因此，在分区上启用了安全引导的任何类型的自动迁移都会失败。

(BZ#2126777)

使用 `page_poison=1` 可能会导致内核崩溃

当在带有错误 EFI 实现的固件上使用 `page_poison=1` 作为内核参数时，操作系统可能会导致内核崩溃。默认情况下，这个选项被禁用，不推荐启用它，特别是在生产系统中。

(BZ#2050411)

kmod 中的 `weak-modules` 无法处理模块间的依赖关系

kmod 软件包提供的 `weak-modules` 脚本决定了哪些模块与安装的内核兼容 kABI。但是，在检查模块的内核兼容性时，`weak-modules` 处理从构建它们的内核的高版本到低版本的模块符号依赖项。因此，针对不同内核版本构建的具有相互依赖关系的模块可能会被解释为不兼容，因此 `weak-modules` 脚本无法在此场景下工作。

要临时解决这个问题，请在安装新内核之前针对最新的库存内核构建或放置额外的模块。

(BZ#2103605)

重新加载相同的崩溃扩展可能会导致分段错误

当加载已经载入的崩溃扩展文件的副本时，它可能会触发一个分段错误。目前，`crash` 工具会检测原始文件是否已被加载。因此，由于两个完全相同的文件在 `crash` 工具中并存，因此会发生命名空间冲突，这会触发 `crash` 工具，从而导致分段错误。

您可以通过只加载 `crash` 扩展文件一次来临时解决此问题。因此，在描述的场景中不再会出现分段错误。

(BZ#1906482)

内存热插拔操作后，`vmcore` 捕获会失败

执行内存热插拔操作后，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)

在 RHEL 8 上，调试内核无法在崩溃捕获环境中引导

由于调试内核的内存密集型特性，当调试内核在使用，且触发了内核 panic 时会出现问题。因此，调试内核无法作为捕获内核启动，而是生成堆栈跟踪。要临时解决这个问题，请根据需要增大崩溃内核内存。因此，调试内核会在崩溃捕获环境中成功引导。

(BZ#1659609)

在引导时分配崩溃内核内存失败

在某些 Ampere Altra 系统上，当 BIOS 设置中禁用了 32 位区域时，在引导过程中分配崩溃内核内存会失败。因此，**kdump** 服务无法启动。这是因为 4 GB 以下的内存碎片导致，没有足够大的碎片来包含崩溃内核内存。

要临时解决这个问题，请在 BIOS 中启用 32 位内存区域，如下所示：

1. 在您的系统中打开 BIOS 设置。
2. 打开 **Chipset** 菜单。
3. 在 **Memory Configuration** 下，启用 **SSlave 32-bit** 选项。

因此，在 32 位区域中的崩溃内核内存分配会成功，**kdump** 服务可以按预期工作。

(BZ#1940674)

QAT 管理器没有为 LKCF 保留备用设备

Intel® QuickAssist Technology (QAT) 管理器 (**qatmgr**) 是一个用户空间进程，默认情况下会使用系统中的所有 QAT 设备。因此，没有为 Linux 内核加密框架 (LKCF) 保留任何 QAT 设备。不需要临时解决这种情况，因为此行为是预期的，大多数用户都将使用来自用户空间的加速。

(BZ#1920086)

内核 ACPI 驱动程序报告无法访问 PCIe ECAM 内存区域

固件提供的高级配置和电源接口 (ACPI) 表没有在 PCI 总线设备中定义内存区域。因此，在系统引导时会出现以下警告信息：


```
[ 2.817152] acpi PNP0A08:00: [Firmware Bug]: ECAM area [mem 0x30000000-0x31ffffff] not reserved in ACPI namespace
[ 2.827911] acpi PNP0A08:00: ECAM at [mem 0x30000000-0x31ffffff] for [bus 00-1f]
```

但是，内核仍然可以访问 **0x30000000-0x31ffff** 内存区域，并可以正确地将该内存区域分配给 PCI 增强配置访问机制(ECAM)。您可以通过以下输出通过 256 字节偏移访问 PCIe 配置空间来验证 PCI 是正常工作的：

```
03:00.0 Non-Volatile memory controller: Sandisk Corp WD Black 2018/PC SN720 NVMe SSD (prog-if 02 [NVM Express])
...
Capabilities: [900 v1] L1 PM Substates
    L1SubCap: PCI-PM_L1.2- PCI-PM_L1.1- ASPM_L1.2+ ASPM_L1.1- L1_PM_Substates+
    PortCommonModeRestoreTime=255us PortTPowerOnTime=10us
    L1SubCtl1: PCI-PM_L1.2- PCI-PM_L1.1- ASPM_L1.2- ASPM_L1.1-
    T_CommonMode=0us LTR1.2_Threshold=0ns
    L1SubCtl2: T_PwrOn=10us
```

因此，您可以忽略警告信息。

有关此问题的详情，请参阅 ["Firmware Bug: ECAM area mem 0x30000000-0x31ffffff not reserved in ACPI namespace" appears during system boot.](#)

(BZ#1868526)

tuned-adm profile powerave 命令会导致系统变得无响应

执行 **tuned-adm profile powersave** 命令会导致具有旧的 Thunderx(CN88x)处理器的 Penguin Valkyrie 2000 2-socket 系统的无响应状态。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)

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)

使用 irqpoll 会导致 vmcore 生成失败

由于在 Amazon Web Services Graviton 1 处理器上运行的 64 位 ARM 架构上的 **nvme** 驱动程序存在问题，因此导致在给第一个内核提供 **irqpoll** 内核命令行参数时 **vmcore** 生成失败。因此，当内核崩溃时，**vmcore** 文件不会在 `/var/crash/` 目录中转储。要临时解决这个问题：

1. 在 `/etc/sysconfig/kdump` 文件中，将 **irqpoll** 追加到 **KDUMP_COMMANDLINE_REMOVE** 变量中。

```
# KDUMP_COMMANDLINE_REMOVE="hugepages hugepagesz slub_debug quiet
log_buf_len swiotlb"
```

2. 删除 `/etc/sysconfig/kdump` 文件中 **KDUMP_COMMANDLINE_APPEND** 变量中的 **irqpoll**。

```
# KDUMP_COMMANDLINE_APPEND="irqpoll nr_cpus=1 reset_devices
cgroup_disable=memory udev.children-max=2 panic=10 swiotlb=noforce novmcoredd"
```

3. 重启 **kdump** 服务：

```
# systemctl restart kdump
```

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，Amazon Web Services Graviton 2 和 Amazon Web Services Graviton 3 处理器不需要手动删除 `/etc/sysconfig/kdump` 文件中的 **irqpoll** 参数。

kdump 服务可能会使用大量崩溃内核内存来转储 **vmcore** 文件。确定捕获内核有足够的内存可用于 **kdump** 服务。

有关此已知问题的详情，请查看 [irqpoll 内核命令行参数可能会导致 vmcore 生成失败](#) 的文章。

(BZ#1654962)

将虚拟功能附加到虚拟机时连接会失败

使用传奇 **ionic** 设备驱动程序的 Pensando 网卡会默默地接受 VLAN 标签配置请求，并在将网络虚拟功能 (**VF**) 附加到虚拟机 (**VM**) 上时尝试配置网络连接。这些网络连接会失败，因为卡的固件还没有支持这个功能。

(BZ#1930576)

OPEN MPI 库可能会使用默认 PML 的触发程序运行时失败

在 OPEN 消息密码界面 (OPEN MPI) 实现 4.0.x 系列中，Unified communicating X (UCX) 是默认的点点到点通信器 (PML)。OPEN MPI 4.0.x 系列之后的版本弃用了 **openib** Byte Transfer Layer (BTL)。

但是，当 OPEN MPI 在同构集群 (相同的硬件和软件配置) 上运行时，UCX 仍会为 MPI 单边操作使用 **openib** BTL。因此，这可能会导致触发器执行错误。要临时解决这个问题：

- 使用以下参数运行 **mpirun** 命令：

```
-mca btl openib -mca pml ucx -x UCX_NET_DEVICES=mlx5_ib0
```

其中，

- **-mca btl openib** 参数禁用 **openib** BTL

- `-mca pml ucx` 参数将 OPEN MPI 配置为使用 `ucx` PML。
- `x UCX_NET_DEVICES=` 参数限制 UCX 使用指定的设备

OPEN MPI 在使用异构集群（不同硬件和软件配置）中运行时，使用 UCX 作为默认的 PML。因此，这可能会导致 OPEN MPI 任务在运行时出现错误的性能、不响应性行为或崩溃问题。要临时解决这个问题，将 UCX 优先级设置为：

- 使用以下参数运行 `mpirun` 命令：

```
-mca pml_ucx_priority 5
```

因此，OPEN MPI 库可以选择使用 UCX 的可替代传输层。

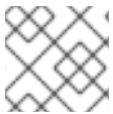
(BZ#1866402)

Solarflare 无法创建最大数量的虚拟函数 (VF)

因为资源不足，Solarflare NIC 无法创建最大数量的 VF。您可以检查 PCIe 设备可以在 `/sys/bus/pci/devices/PCI_ID/sriov_totalvfs` 文件中创建的 VF 的最大数量。要临时解决这个问题，您可以将 VF 的数量或 VF MSI 中断值调整为较低的值，可以在启动时从 **Solarflare Boot Manager** 调整或使用 Solarflare `sfboot` 工具调整。默认的 VF MSI 中断值为 **8**。

- 使用 `sfboot` 来调整 VF MSI 中断值：

```
# sfboot vf-msix-limit=2
```



注意

调整 VF MSI 中断值会影响 VF 性能。

有关调整的参数更多信息，请参阅 **Solarflare 服务器适配器用户指南**。

(BZ#1971506)

iwl7260-firmware 破坏了 Intel Wi-Fi 6 AX200、AX210 和 Lenovo ThinkPad P1 Gen 4 上的 Wi-Fi

在将 `iwl7260-firmware` 或 `iwl7260-wifi` 驱动程序更新到 RHEL 8.7 和/或 RHEL 9.1（及更高版本）提供的版本后，硬件会进入不正确的内部状态。报告其状态不正确。因此，Intel Wifi 6 卡可能无法正常工作，并显示错误信息：

```
kernel: iwlwifi 0000:09:00.0: Failed to start RT ucode: -110
kernel: iwlwifi 0000:09:00.0: WRT: Collecting data: ini trigger 13 fired (delay=0ms)
kernel: iwlwifi 0000:09:00.0: Failed to run INIT ucode: -110
```

一个没有确认的临时解决方法是关闭并再次打开系统。不要重启。

(BZ#2106341)

kdump 的内存分配在 64 位 ARM 构架上失败

在某些基于 64 位 ARM 的系统上，固件使用非连续内存分配方法，此方法在不同的分散位置随机保留内存。因此，由于连续内存块不可用，崩溃内核无法为 `kdump` 机制保留内存空间。

要临时解决这个问题，请安装 RHEL 8.8 及更新版本提供的内核版本。RHEL 的最新版本支持 **fallback** 转储捕获机制，其帮助在上述场景中找到合适的内存区域。

([BZ#2214235](#))

在具有大核数的系统上实时内核的硬件认证可能需要传递 `skew-tick=1` 引导参数以避免锁争用

具有大量插槽和大核数的大型系统可能会因为在计时系统中使用的 `xtime_lock` 上的锁争用而遇到延迟峰值。因此，在多处理系统中可能会发生硬件认证的延迟峰值和延迟。作为临时解决方案，您可以通过添加 `skew_tick=1` 引导参数，来偏移每个 CPU 的计时器刻度，以便在不同的时间启动。

要避免锁冲突，请启用 `skew_tick=1`：

1. 使用 `grubby` 启用 `skew_tick=1` 参数。

```
# grubby --update-kernel=ALL --args="skew_tick=1"
```

2. 重启以使更改生效。
3. 运行 `cat /proc/cmdline` 命令来验证新设置。

请注意，启用 `skew_tick=1` 会导致功耗大量增加，因此只有在您运行延迟敏感的实时工作负载时才必须启用。

([BZ#2214508](#))

11.9. 引导加载程序

`grubby` 的行为与其文档有区别

当您使用 `grubby` 工具添加了一个新内核而不指定任何参数时，`grubby` 会将默认参数传递给新条目。即使不传递 `--copy-default` 参数，也会发生此行为。使用 `--args` 和 `--copy-default` 选项确保这些参数附加到默认参数中，如 `grubby` 文档中所述。

但是，当添加额外的参数（如 `$tuned_params`）时，`grubby` 工具不会传递这些参数，除非调用了 `--copy-default` 选项。

在这种情况下，有两个临时解决方案：

- 设置 `root=` 参数，并保留 `--args` 为空：

```
# grubby --add-kernel /boot/my_kernel --initrd /boot/my_initrd --args "root=/dev/mapper/rhel-root" --title "entry_with_root_set"
```

- 或者设置 `root=` 参数和指定的参数，但不是默认的参数：

```
# grubby --add-kernel /boot/my_kernel --initrd /boot/my_initrd --args "root=/dev/mapper/rhel-root some_args and_some_more" --title "entry_with_root_set_and_other_args_too"
```

([BZ#1900829](#))

11.10. 文件系统和存储

LVM writecache 的限制

writocache LVM 缓存方法有以下限制，这些限制不会出现在 **cache** 方法中：

- 使用 **pvmove** 命令时您无法命名 **writocache** 逻辑卷。
- 您不能将带有 **writocache** 的逻辑卷与精简池或 VDO 结合使用。

以下限制也适用于 **cache** 方法：

- 您不能在将 **cache** 或 **writocache** 附加到逻辑卷时重新定义大小。

(JIRA:RHELPLAN-27987、[BZ#1798631](#)、[BZ#1808012](#))

XFS 配额警告经常被触发

使用配额计时器会导致配额警告的触发过于频繁，这导致软配额的强制执行的速度比应该执行的速度更快。要临时解决这个问题，请不要使用软配额，这会防止触发警告。因此，警告信息量将不再强制软配额限制，会尊重配置的超时时间。

([BZ#2059262](#))

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](#))

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)

blk-availability systemd 服务停用了复杂的设备堆栈

在 **systemd** 中，默认的块停用代码并不总是正确处理虚拟块设备的复杂堆栈。在一些配置中，虚拟设备在关闭过程中可能无法被删除，这会导致记录错误信息。要临时解决这个问题，请执行以下命令来停用复杂块设备堆栈：

```
# systemctl enable --now blk-availability.service
```

因此，复杂虚拟设备堆栈会在关闭过程中被正确停用，且不会生成错误消息。

(BZ#2011699)

VDO 驱动程序 bug 可能会导致设备通过日志块冻结

在跟踪设备映射器 **suspend** 操作时，VDO 驱动程序中的一个 bug 会导致系统将一些日志块标记为等待元数据更新。自 **suspend** 调用以来，更新已应用。

当日志返回到相同的物理块时，块将停止可用。最后，所有写操作都会停止，直到块再次可用。VDO 设备上的 **growPhysical**、**growLogical** 和 **setWritePolicy** 操作包含一个 **suspend/resume** 循环，这会导致在几次日志更新后出现设备冻结。

在它之上增加 VDO 池或逻辑卷的大小，或对 LVM 工具管理的 VDO 设备使用 **pvmove** 和 **lvchange** 操作也会触发此问题。

作为临时解决方案，以任何涉及 **suspend/resume** 循环的方式更改 VDO 设备的设置，完全关闭 VDO 设备，然后再次启动它。这会清除不正确的内存状态，并重置日志块。因此，该设备不再被冻结，可以正常工作。

(BZ#2109047)

在启动 VDO 卷时，系统会因为软锁定而挂起

为了修复 **pv_mmu_ops** 结构中的内核 ABI 中断，内核版本为 **4.18.0-425.10.1.el8_7** 的 RHEL 8.7 系统，即 RHEL-8.7.0.2-BaseOS，会在启动 Virtual Data Optimizer (VDO) 卷时挂起或遇到内核 panic。要临时解决这个问题，请在引导到 **kernel-4.18.0-425.10.1.el8_7** 前禁用任何 VDO 卷，以防止系统挂起，或降级到内核的早期版本，即 **4.18.0-425.3.1.el8**，以保留 VDO 功能。

(BZ#2158783)

11.11. 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)

PAM 插件版本 1.0 在 MariaDB 中无法正常工作

MariaDB 10.3 提供了可插拔验证模块(PAM)插件版本 1.0。**MariaDB 10.5** 提供插件版本 1.0 和 2.0，版本 2.0 是默认版本。

MariaDB PAM 插件版本 1.0 在 RHEL 8 中无法正常工作。要临时解决这个问题，请使用 **mariadb:10.5** 模块流提供的 PAM 插件版本 2.0。

[\(BZ#1942330\)](#)

OpenLDAP 库之间的符号冲突可能会导致 httpd 中的崩溃

当 OpenLDAP 提供的 `libldap` 和 `libldap_r` 库被加载，并在单个进程中使用，这些库之间可能会出现符号冲突。因此，如果 `mod_security` 或 `mod_auth_openidc` 模块也被 `httpd` 配置加载，使用 PHP `ldap` 扩展的 Apache `httpd` 子进程可能会意外终止。

从因为 RHEL 8.3 更新到 Apache Portable Runtime (APR) 库，您可以通过设置 `APR_DEEPCBIND` 环境变量来临时解决这个问题，该变量在载入 `httpd` 模块时，允许使用 `RTLD_DEEPCBIND` 动态链接器选项。当 `APR_DEEPCBIND` 环境变量启用时，在加载冲突库的 `httpd` 配置中不再崩溃。

[\(BZ#1819607\)](#)

11.12. 身份管理

将 `cert-fix` 程序与 `--agent-uid pkidbuser` 选项一同使用会破坏证书系统

使用带有 `--agent-uid pkidbuser` 选项的 `cert-fix` 工具可破坏证书系统的 LDAP 配置。因此，证书系统可能会变得不稳定，需要手动步骤才能恢复该系统。

[\(BZ#1729215\)](#)

IdM 主机上的 `/var/log/lastlog` 稀疏文件可能会导致性能问题

在 IdM 安装过程中，从总计 10,000 个可能范围内会随机选择并分配一个 200,000 UID 范围。当您决定以后合并两个独立的 IdM 域时，以这种方法选择一个随机范围可显著降低冲突 ID 的可能性。

但是，高 UID 可能会导致 `/var/log/lastlog` 文件出现问题。例如，如果 UID 为 1280000008 的用户登录到 IdM 客户端，则本地 `/var/log/lastlog` 文件的大小会增加到近 400 GB。虽然实际文件是稀疏的，且没有使用所有空间，但某些应用程序默认不是为识别稀疏文件而设计的，且可能需要一个特定的选项来处理这些文件。例如，如果设置比较复杂，备份和复制应用程序无法正确处理稀疏文件，则该文件会像大小为 400 GB 一样被复制。这个行为可能会导致性能问题。

要临时解决这个问题：

- 如果是标准软件包，请参考其文档来识别处理稀疏文件的选项。
- 如果是自定义应用程序，请确保它能够正确管理稀疏文件，如 `/var/log/lastlog`。

[\(JIRA:RHELPLAN-59111\)](#)

FIPS 模式不支持使用共享 `secret` 建立跨林信任

在 FIPS 模式中使用共享 `secret` 建立跨林信任会失败，因为 NTLMSSP 身份验证不兼容 FIPS。要临时解决这个问题，在启用了 FIPS 模式的 IdM 域和 AD 域间建立信任时，使用 Active Directory (AD) 管理帐户进行身份验证。

[\(BZ#1924707\)](#)

FreeRADIUS 服务器无法在 FIPS 模式下运行

默认情况下，在 FIPS 模式下，OpenSSL 禁止使用 MD5 摘要算法。由于 RADIUS 协议需要 MD5 来在 RADIUS 客户端和 RADIUS 服务器之间加密 `secret`，因此这会导致 FreeRADIUS 服务器在 FIPS 模式下失败。

要临时解决这个问题，请遵循以下步骤：

流程

1. 为 **radiusd** 服务创建环境变量 **RADIUS_MD5_FIPS_OVERRIDE** :

```
systemctl edit radiusd

[Service]
Environment=RADIUS_MD5_FIPS_OVERRIDE=1
```

2. 要应用这些更改，请重新加载 **systemd** 配置，并启动 **radiusd** 服务：

```
# systemctl daemon-reload
# systemctl start radiusd
```

3. 要在调试模式下运行 FreeRADIUS:

```
# RADIUS_MD5_FIPS_OVERRIDE=1 radiusd -X
```

请注意，虽然 FreeRADIUS 可以在 FIPS 模式下运行，但这并不意味着它符合 FIPS，因为它在 FIPS 模式下使用弱密码和功能。

有关在 FIPS 模式下配置 FreeRADIUS 身份验证的更多信息，请参阅 [在 FIPS 模式下如何配置 FreeRADIUS 身份验证](#)。

([BZ#1958979](#))

IdM 到 AD 跨域 TGS 请求失败

IdM Kerberos 票据中的 Privilege Attribute 证书(PAC) 信息现在使用 AES SHA-2 HMAC 加密进行签名，这是 Active Directory (AD) 不支持的。

因此，IdM 到 AD 跨域 TGS 请求（即双向信任设置）失败，并显示以下错误：

```
"Generic error (see e-text) while getting credentials for <service principal>"
```

([BZ#2125182](#))

迁移的 IdM 用户可能会因为不匹的域 SID 而无法登录

如果您使用 **ipa migrate-ds** 脚本将用户从一个 IdM 部署迁移到另一个，则这些用户可能会在使用 IdM 服务时有问题，因为它们之前存在的安全标识符(SID)没有当前 IdM 环境的域 SID。例如，这些用户可以使用 **kinit** 工具检索 Kerberos 票据，但不能登录。要临时解决这个问题，请参阅以下知识库文章：[Migrated IdM 用户因为不匹配的域 SID 而无法登录](#)。

([JIRA:RHELPLAN-109613](#))

FIPS 模式下的 IdM 不支持使用 NTLMSSP 协议来建立双向跨林信任

在活动目录(AD)和启用了 FIPS 模式的身份管理(IdM)之间建立双向跨林信任会失败，因为新技术局域网管理器安全支持提供程序 (NTLMSSP)身份验证不符合 FIPS。FIPS 模式下的 IdM 不接受在尝试验证时 AD 域控制器使用的 RC4 NTLM 哈希。

([BZ#2120572](#))

IdM Vault 加密和解密在 FIPS 模式下失败

如果启用了 FIPS 模式，则 OpenSSL RSA-PKCS1v15 填充加密会被阻止。因此，身份管理(IdM) Vault 无法正常工作，因为 IdM 目前正在使用 PKCS1v15 填充来用传输证书包装会话密钥。

(BZ#2122919)

将 Samba 作为打印服务器运行，并从 RHEL 8.4 或更早版本更新时所需的操作

有了这个更新，**samba** 软件包不再创建 `/var/spool/samba/` 目录。如果您将 Samba 用作打印服务器，并将 **[printers]** 共享中的 `/var/spool/samba/` 用于假脱机打印作业，则 SELinux 会阻止 Samba 用户在此目录中创建文件。因此，打印作业失败，**auditd** 服务会在 `/var/log/audit/audit.log` 中记录一跳 **denied** 消息。要在更新 8.4 及更早版本的系统后避免此问题：

1. 在 `/etc/samba/smb.conf` 文件中搜索 **[printers]** 共享。
2. 如果共享定义包含 `path = /var/spool/samba/`，请更新设置，并将 `path` 参数设为 `/var/tmp/`。
3. 重启 **smbd** 服务：

```
# systemctl restart smbd
```

如果在 RHEL 8.5 或更高版本上新安装了 Samba，则不需要任何操作。在本例中，**samba-common** 软件包提供的默认 `/etc/samba/smb.conf` 文件已将 `/var/tmp/` 目录用于假脱机打印作业。

(BZ#2009213)

在 rebase 到版本 1.2.2 后，降级 authselect 会破坏系统身份验证

authselect 软件包已 rebase 到最新的上游版本 **1.2.2**。不支持降级 **authselect**，这会中断所有用户的系统身份验证，包括 **root** 用户。

如果您将 **authselect** 软件包降级到 **1.2.1** 或更早版本，请执行以下步骤来解决这个问题：

1. 在 GRUB 引导屏幕中，选择带有您要引导的内核版本的 **Red Hat Enterprise Linux**，然后按 **e** 编辑该条目。
2. 在以 **linux** 开头的行的末尾输入 **single** 作为单独的单词，然后按 **Ctrl+X** 启动引导过程。
3. 以单用户模式引导时，输入 **root** 密码。
4. 使用以下命令恢复 **authselect** 配置：

```
# authselect select sssd --force
```

(BZ#1892761)

NSS 中启用的密码的 **default** 关键字无法与其他密码一起使用

在目录服务器中，您可以使用 **default** 关键字来指向网络安全服务(NSS)中启用的默认密码。但是，如果您要使用命令行或 Web 控制台来启用默认的密码和附加密码，目录服务器将无法解析 **default** 关键字。因此，服务器只启用额外指定的密码，并记录类似如下的错误：

```
Security Initialization - SSL alert: Failed to set SSL cipher preference information: invalid ciphers
<default,+cipher_names>: format is +cipher1,-cipher2... (Netscape Portable Runtime error 0 - no error)
```

作为临时解决方案，指定 NSS 中默认启用的所有密码，包括您要额外启用的密码。

(BZ#1817505)

从 RHEL 8.6 到 RHEL 8.7 的 pki-core-debuginfo 更新会失败

将 **pki-core-debuginfo** 软件包从 RHEL 8.6 更新至 RHEL 8.7 会失败。要临时解决这个问题，请运行以下命令：

1. **yum remove pki-core-debuginfo**
2. **yum update -y**
3. **yum install pki-core-debuginfo**
4. **yum install idm-pki-symkey-debuginfo idm-pki-tools-debuginfo**

([BZ#2134093](#))

对 ldap_id_use_start_tls 选项使用默认值时的潜在风险

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，其允许攻击者，例如通过更改 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运行，并决定对 **id_provider = ldap** 使用未加密的通信是否安全。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](#))

11.13. DESKTOP

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，请将 Hyper-V Server 2019 或更高版本用作主机。

([BZ#1583445](#))

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](#))

11.14. 图形基础结构

radeon 无法正确重置硬件

radeon 内核驱动程序目前没有在有 `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** 中禁用 **radeon**，方法是在 `/etc/kdump.conf` 文件中添加以下行：

```
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 图灵 GPU 的系统上，使用一个有支持 HDR 的多个显示器插入其中的 **DisplayPort** 集线器（如笔记本电脑底座），可能会导致打开失败。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)

ESXi 中的 GUI 可能会因为视频内存较低而崩溃

VMware ESXi 7.0.1 带有 vCenter Server 7.0.1 的 RHEL 虚拟机上的图形用户界面（GUI）需要一定的视频内存。如果您将多个控制台或高分辨率监控器连接到虚拟机，则 GUI 至少需要 16 MB 的视频内存。如果您使用较少的视频内存启动 GUI，则 GUI 可能会意外终止。

要临时解决这个问题，请配置虚拟机监控程序为虚拟机分配至少 16MB 视频内存。因此，虚拟机上的 GUI 不会崩溃。

如果您遇到这个问题，红帽建议您将其报告给 VMware。

另请参阅以下 VMware 文章：[具有高分辨率 VM 控制台的虚拟机可能会在 ESXi 7.0.1\(83194\)上出现崩溃](#)。

(BZ#1910358)

VNC Viewer 显示在 IBM Z 中带有 16 位颜色深度的错误颜色

当您连接到带有 16 位色彩深度的 IBM Z 服务器上的 VNC 会话时，VNC Viewer 应用程序会显示错误的颜色。

要临时解决这个问题，请在 VNC 服务器中设置 24 位颜色深度。使用 **Xvnc** 服务器将 `-depth 16` 选项替换为 `-depth 24`（在 **Xvnc** 配置中）。

因此，VNC 客户端会显示正确的颜色，但在服务器中使用更多的网络带宽。

(BZ#1886147)

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](#))

ARM 不支持硬件加速

内置图形驱动程序不支持 64 位 ARM 架构中的硬件加速或 Vulkan API。

要启用硬件加速或者 ARM 上的 Vulkan，安装专有 Nvidia 驱动程序。

([JIRA:RHELPLAN-57914](#))

Matrox G200e 在 VGA 显示器上没有显示输出

如果您使用以下系统配置，您的显示可能不会显示图形输出：

- Matrox G200e GPU
- 通过 VGA 控制器连接的显示

因此，您不能在这种配置中使用或安装 RHEL。

要临时解决这个问题，请使用以下步骤：

1. 将系统启动到引导装载程序菜单。
2. 向内核命令行中添加 **module_blacklist=mgag200** 选项。

因此，RHEL 会按预期引导并显示图形输出，但最大分辨率限制为 16 位颜色深度的 1024x768。

([BZ#2130159](#))

11.15. WEB 控制台

VNC 控制台在某些分辨率下无法正常工作

当在某些显示分辨率下使用虚拟网络计算(VNC)控制台时，您可能会遇到鼠标偏移问题，或者您可能只看到一部分界面。因此，可能无法使用 VNC 控制台。要临时解决这个问题，您可以尝试扩展 VNC 控制台的大小，或使用 Console 选项卡中的 Desktop Viewer 来启动远程查看器。

([BZ#2030836](#))

11.16. RED HAT ENTERPRISE LINUX 系统角色

使用 playbook 或清单中的 localhost 主机名无法管理 localhost

RHEL 中包含了 **ansible-core 2.13** 软件包，如果您在您管理节点的同一台主机上运行 Ansible，则无法通过在 playbook 或清单中使用 **localhost** 主机名来实现它。这是因为 **ansible-core 2.13** 使用 **python38** 模块，并且缺少许多库，例如，用于 **storage** 角色的 **blivet**，用于 **network** 角色的 **gobject**。要临时解决这个问题，如果您已在 playbook 或清单中使用了 **localhost** 主机名，您可以使用 **ansible_connection=local** 来添加一个连接，或者使用 **ansible_connection=local** 选项创建一个列出 **localhost** 的清单文件。使用这种方式，您可以管理 **localhost** 上的资源。如需了解更多详细信息，请参阅文章在 [localhost 上运行时，RHEL 系统角色 playbook 会失败](#)。

([BZ#2041997](#))

11.17. 虚拟化

使用大量队列可能会导致 Windows 虚拟机失败

当启用了虚拟可信平台模块(vTPM)设备，且将 *multi-queue virtio-net* 功能配置为使用超过 250 个队列时，Windows 虚拟机(VM)可能会失败。

这个问题是由 vTPM 设备的限制造成的。vTPM 设备对于打开的文件描述符的最大数量有一个硬性的限制。因为会为每个新队列打开多个文件描述符，因此可能会超过内部 vTPM 的限值，从而导致虚拟机失败。

要临时解决这个问题，请选择以下两个选项之一：

- 保持 vTPM 设备启用，但使用少于 250 个队列。
- 禁用 vTPM 设备以使用超过 250 个队列。

([BZ#2020133](#))

AMD Milan 系统上有时无法提供 Milan VM CPU 类型

在某些 AMD Milan 系统上，默认在 BIOS 中禁用了增强 REP MOVSB(**erms**)和 Fast Short REP MOVSB(**fsrm**)功能标记。因此，在这些系统上可能无法使用 **Milan** CPU 类型。另外，在具有不同功能标志设置的 Milan 主机之间的虚拟机实时迁移可能会失败。要临时解决这个问题，请在主机的 BIOS 中手动打开 **erms** 和 **fsrm**。

([BZ#2077770](#))

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](#))

带有 iommu_platform=on 的虚拟机无法在 IBM POWER 上启动

RHEL 8 目前不支持 IBM POWER 系统上的虚拟机(VM)的 **iommu_platform=on** 参数。因此，在 IBM POWER 硬件上启动使用这个参数的虚拟机会导致虚拟机在引导过程中变得无响应。

([BZ#1910848](#))

当使用ibmvfc 驱动程序时，IBM POWER 主机可能会崩溃

当在 PowerVM 逻辑分区(LPAR)上运行 RHEL 8 时，目前可能会因为 **ibmvfc** 驱动程序的问题而出现各种错误。因此，在某些情况下主机的内核可能会出现 panic，例如：

- 使用 Live Partition Mobility(LPM)功能
- 重置主机适配器
- 使用 SCSI 错误处理(SCSI EH)功能

([BZ#1961722](#))

在 IBM POWER 系统中使用 `perf kvm record` 可能会导致虚拟机崩溃

当在 IBM POWER 硬件的 little-endian 变体中使用 RHEL 8 主机时，使用 `perf kvm record` 命令为 KVM 虚拟机(VM)收集追踪事件样本会导致虚拟机变得没有响应。在以下情况下发生这种情况：

- `perf` 工具由非特权用户使用，`-p` 选项用来标识虚拟机 - 例如 `perf kvm record -e trace_cycles -p 12345`。
- 虚拟机是使用 `virsh shell` 启动的。

要临时解决这个问题，请使用带有 `-i` 选项的 `perf kvm` 工具来监控使用 `virsh shell` 创建的虚拟机。例如：

```
# perf kvm record -e trace_imc/trace_cycles/ -p <guest pid> -i
```

请注意，在使用 `-i` 选项时，子任务不会继承计数器，因此线程不会被监控。

(BZ#1924016)

带有启用 Hyper-V 的 Windows Server 2016 虚拟机无法引导

目前，无法引导使用 Windows Server 2016 作为客户端操作系统的虚拟机，它启用了 Hyper-V 角色，并使用以下 CPU 模型之一：

- EPYC-IBPB
- EPYC

要临时解决这个问题，使用 `EPYC-v3` CPU 模型，或者为虚拟机手动启用 `xsaves` CPU 标记。

(BZ#1942888)

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 会变得没有响应，显示 **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)

使用 `virt-customize` 有时会导致 `guestfs-firstboot` 失败

使用 `virt-customize` 工具修改了虚拟机(VM)磁盘镜像后，`guestfs-firstboot` 服务在某些情况下会因为 SELinux 权限不正确而失败。这会导致虚拟机启动过程中出现各种问题，如创建用户失败或系统注册失败。

要避免这个问题，请在使用 `virt-customize` 修改虚拟机的磁盘镜像后，将 `--selinux-relabel` 添加到虚拟机的内核命令中。

(BZ#1554735)

从 `macvtap` 虚拟网络中删除转发接口会重置这个网络的所有连接计数

目前，从带有多个转发接口的 `macvtap` 虚拟网络中删除一个转发接口也会重置网络的其他转发接口的连接状态。因此，实时网络 XML 中的连接信息是不正确的。但请注意，这不会影响虚拟网络的功能。要临时解决这个问题，请重启主机上的 `libvirtd` 服务。

(BZ#1332758)

带有 SLOF 的虚拟机无法在 netcat 界面中引导

当使用 netcat(nc)接口来访问当前在 Slimline Open Firmware(SLOF)提示符下等待的虚拟机控制台时，用户输入会被忽略，虚拟机仍无响应。要临时解决这个问题，请在连接虚拟机时使用 **nc -C** 选项，或者使用 telnet 接口。

(BZ#1974622)

在某些情况下，在 virt-manager 中将中介设备附加到虚拟机会失败

virt-manager 应用程序目前能够检测中介设备，但不能识别该设备是否处于活跃状态。因此，尝试使用 **virt-manager** 将不活跃的中介设备附加到正在运行的虚拟机(VM)会失败。同样，尝试使用不活跃的中介设备创建新虚拟机会失败，并显示 **device not found** 错误。

要临时解决这个问题，请使用 **virsh nodedev-start** 或 **mdevctl start** 命令来激活中介设备，然后再在 **virt-manager** 中使用它。

(BZ#2026985)

RHEL 9 虚拟机无法在 POWER8 兼容模式下启动

目前，如果虚拟机也使用类似如下的 CPU 配置，启动将 RHEL 9 作为客户机操作系统来运行的虚拟机(VM)会失败：

```
<cpu mode="host-model">
  <model>power8</model>
</cpu>
```

要临时解决这个问题，请不要在 RHEL 9 虚拟机中使用 POWER8 兼容性模式。

另外，请注意，在 POWER8 主机上无法运行 RHEL 9 虚拟机。

(BZ#2035158)

在主机上重启 OVS 服务可能会阻止在其上运行的虚拟机的网络连接

当 Open vSwitch (OVS)服务在主机上重启时或崩溃时，在此主机上运行的虚拟机(VM)无法恢复网络设备的状态。因此，虚拟机可能无法完全接收数据包。

此问题只会影响在 **virtio** 网络堆栈中使用压缩 **virtqueue** 格式的系统。

要临时解决这个问题，使用 **virtio** 网络设备定义中的 **packed=off** 参数来禁用压缩的 **virtqueue**。当禁用压缩的 **virtqueue** 时，网络设备的状态在某些情况下可以从 RAM 中恢复。

(BZ#1792683)

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)

在 virtiofs 上不会自动清除 SUID 和 SGID

当您运行带有 **killpriv_v2** 功能的 **virtiofsd** 服务时，您的系统在执行某些文件系统操作后可能不能自动清除 SUID 和 SGID 权限。因此，不能清除权限可能会导致潜在的安全威胁。要临时解决这个问题，请输入以下命令禁用 **killpriv_v2** 功能：

■

```
# virtiofsd -o no_killpriv_v2
```

(BZ#1966475)

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 主机上以 CPU 主机直通模式引导虚拟机(VM)时，不会显示 **TOPOEXT** CPU 功能标志。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)

11.18. 云环境中的 RHEL

在 VMware 主机上的 RHEL 虚拟机中设置静态 IP 无法正常工作

目前，当使用 RHEL 作为 VMware 主机上虚拟机(VM)的客户机操作系统时，DatasourceOVF 功能无法正常工作。因此，如果您使用 **cloud-init** 实用程序将虚拟机的网络设置为静态 IP，然后重启虚拟机，则虚拟机的网络将更改为 DHCP。

(BZ#1750862)

kdump 有时不会在 Azure 和 Hyper-V 上启动

在托管在 Microsoft Azure 或 Hyper-V hypervisor 上的 RHEL 8 客户机操作系统中，启动 **kdump** 内核在某些情况下会在启用执行后通知程序时失败。

要临时解决这个问题，请禁用 crash kexec post notifiers：

```
# echo N > /sys/module/kernel/parameters/crash_kexec_post_notifiers
```

(BZ#1865745)

当使用多个客户机磁盘引导 Hyper-V 虚拟机时，SCSI 主机地址有时会改变

目前，当在 Hyper-V hypervisor 上引导 RHEL 8 虚拟机时，主机部分的主机、*Bus*、*Target*、*Lun* (HBTL) SCSI 地址在某些情况下发生了变化。因此，使用虚拟机中的 HBTL SCSI 识别或者设备节点设置的自动任务无法持续工作。如果虚拟机有多个磁盘，或者磁盘大小不同，会出现这种情况。

要临时解决这个问题，使用以下方法之一修改 kickstart 文件：

方法 1：对 SCSI 设备使用持久性标识符。

您可以使用以下 powershell 脚本来确定具体设备标识符：

```
# Output what the /dev/disk/by-id/<value> for the specified hyper-v virtual disk.
# Takes a single parameter which is the virtual disk file.
# Note: kickstart syntax works with and without the /dev/ prefix.
param (
    [Parameter(Mandatory=$true)][string]$virtualdisk
)

$what = Get-VHD -Path $virtualdisk
$part = $what.DiskIdentifier.ToLower().split('-')

$p = $part[0]
$s0 = $p[6] + $p[7] + $p[4] + $p[5] + $p[2] + $p[3] + $p[0] + $p[1]
```



```
$p = $part[1]
$s1 = $p[2] + $p[3] + $p[0] + $p[1]

[string]::format("/dev/disk/by-id/wwn-0x60022480{0}{1}{2}", $s0, $s1, $part[4])
```

您可以在 hyper-v 主机上使用此脚本，如下所示：

```
PS C:\Users\Public\Documents\Hyper-V\Virtual hard disks> .\by-id.ps1 .\Testing_8\disk_3_8.vhdx
/dev/disk/by-id/wwn-0x60022480e00bc367d7fd902e8bf0d3b4
PS C:\Users\Public\Documents\Hyper-V\Virtual hard disks> .\by-id.ps1 .\Testing_8\disk_3_9.vhdx
/dev/disk/by-id/wwn-0x600224807270e09717645b1890f8a9a2
```

之后，可在 kickstart 文件中使用磁盘值，如下所示：

```
part / --fstype=xfs --grow --asprimary --size=8192 --ondisk=/dev/disk/by-id/wwn-
0x600224807270e09717645b1890f8a9a2
part /home --fstype="xfs" --grow --ondisk=/dev/disk/by-id/wwn-
0x60022480e00bc367d7fd902e8bf0d3b4
```

因为这些值特定于每个虚拟磁盘，因此需要为每个虚拟机实例进行配置。因此，使用 **%include** 语法将磁盘信息放在单独的文件中可能会很有用。

方法 2：按大小设置设备选择。

根据大小配置磁盘选择的 kickstart 文件必须包含类似如下的行：

```
...

# Disk partitioning information is supplied in a file to kick start
%include /tmp/disks

...

# Partition information is created during install using the %pre section
%pre --interpreter /bin/bash --log /tmp/ks_pre.log

# Dump whole SCSI/IDE disks out sorted from smallest to largest ouputting
# just the name
disks=(`lsblk -n -o NAME -l -b -x SIZE -d -l 8,3`) || exit 1

# We are assuming we have 3 disks which will be used
# and we will create some variables to represent
d0=${disks[0]}
d1=${disks[1]}
d2=${disks[2]}

echo "part /home --fstype="xfs" --ondisk=$d2 --grow" >> /tmp/disks
echo "part swap --fstype="swap" --ondisk=$d0 --size=4096" >> /tmp/disks
echo "part / --fstype="xfs" --ondisk=$d1 --grow" >> /tmp/disks
echo "part /boot --fstype="xfs" --ondisk=$d1 --size=1024" >> /tmp/disks

%end
```

(BZ#1906870)

如果 RHEL 实例是由 **cloud-init** 提供的，且使用 NFSv3 挂载条目配置的，则其在 Azure 上无法引导

目前，如果 VM 是由 **cloud-init** 工具提供的，且虚拟机的客户机操作系统在 **/etc/fstab** 文件中有 NFSv3 挂载条目，则在 Microsoft Azure 云平台上引导 RHEL 虚拟机(VM)会失败。

(BZ#2081114)

11.19. 支持性

getattachment 命令无法一次下载多个附件

redhat-support-tool 命令提供了用于下载附件的 **getattachment** 子命令。但是，**getattachment** 目前只能下载单个附件，无法下载多个附件。

作为临时解决方案，您可以通过在 **getattachment** 子命令中一个一个地传递每个附件的问题单号和 UUID 来下载多个附件。

(BZ#2064575)

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)

当在 IBM Power Systems, Little Endian 上运行 sos report 时会超时

当在具有带有数百或数千个 CPU 的 IBM Power Systems, Little Endian 上运行 **sos report** 命令时，处理器插件会在收集 **/sys/devices/system/cpu** 目录的大量内容时达到默认的 300 秒超时时间。作为临时解决方案，请相应地增加插件的超时时间：

- 对于一次性设置，请运行：

```
# sos report -k processor.timeout=1800
```

- 对于永久性更改，请编辑 **/etc/sos/sos.conf** 文件的 **[plugin_options]** 部分：

```
[plugin_options]
# Specify any plugin options and their values here. These options take the form
# plugin_name.option_name = value
#rpm.rpmva = off
processor.timeout = 1800
```

示例值设为 1800。特定的超时值高度依赖于特定的系统。要适当地设置插件超时，您可以首先通过运行以下命令来估算收集一个没有超时的插件所需的时间：

```
# time sos report -o processor -k processor.timeout=0 --batch --build
```

(BZ#2011413)

11.20. 容器

在较旧的容器镜像中运行 systemd 无法正常工作

在较旧的容器镜像（如 **centos:7**）中运行 systemd 将无法正常工作：

```
$ podman run --rm -ti centos:7 /usr/lib/systemd/systemd
Storing signatures
Failed to mount cgroup at /sys/fs/cgroup/systemd: Operation not permitted
[!!!!!!] Failed to mount API filesystems, freezing.
```

要临时解决这个问题，请使用以下命令：

```
# mkdir /sys/fs/cgroup/systemd
# mount none -t cgroup -o none,name=systemd /sys/fs/cgroup/systemd
# podman run --runtime /usr/bin/crun --annotation=run.oci.systemd.force_cgroup_v1=/sys/fs/cgroup -
-rm -ti centos:7 /usr/lib/systemd/systemd
```

(JIRA:RHELPLAN-96940)

第 12 章 国际化

12.1. 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-hangul, libhangul

12.2. 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](#)。
- 很多 **glibc** 区域已经与 Unicode 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#2052528 , BZ#2057063 , BZ#2057066 , BZ#2062679 , BZ#1817505
NetworkManager	BZ#2097270 , BZ#2082000 , BZ#2063109 , BZ#1943153 , BZ#1920398 , BZ#2132754
SLOF	BZ#1910848
accel-config	BZ#1843266
anaconda	BZ#1899494 , BZ#1970726 , BZ#2029101 , BZ#2050140 , BZ#1914955 , BZ#1929105 , BZ#2126506
ansible-collection-microsoft-sql	BZ#2066338
ansible-collection-redhat-rhel_mgmt	BZ#2112435
ansible-freeipa	BZ#2076554
apr	BZ#1819607
authselect	BZ#1892761
bacula	BZ#2089399
brltty	BZ#2008197
chrony	BZ#2062356
clevis	BZ#2107081
cloud-init	BZ#1750862
cockpit-appstream	BZ#2030836
cockpit	BZ#2056786 , BZ#1666722
coreutils	BZ#2030661
corosync-qdevice	BZ#1784200

组件	票证
crash-ptdump-command	BZ#1838927
crash	BZ#1906482
createrepo_c	BZ#1973588
cronie	BZ#1832510
crypto-policies	BZ#1919155 , BZ#1660839
cups-filters	BZ#2064606
device-mapper-multipath	BZ#2065477 , BZ#2011699
distribution	BZ#2063772 , BZ#1657927
dnf	BZ#2060815 , BZ#1986657
dotnet7.0	BZ#2112096
dyninst	BZ#2057676
ec2-images	BZ#1862930
edk2	BZ#1741615 , BZ#1935497
fapolicyd	BZ#2070639 , BZ#2100087 , BZ#2054741
fence-agents	BZ#1775847
firewalld	BZ#1871860
freeradius	BZ#1958979
frr	BZ#1714984
gcc-toolset-12-annobin	BZ#2077447
gcc-toolset-12-binutils	BZ#2077448
gcc-toolset-12-gcc	BZ#2077276
gcc-toolset-12-gdb	BZ#2077492

组件	票证
gdb	BZ#1853140
glibc	BZ#1888660, BZ#1982608, BZ#2065588, BZ#1961109, BZ#2089247, BZ#2091553, BZ#2104907, BZ#2033684, BZ#2096189, BZ#2077835
gnome-control-center	BZ#2079139
gnome-shell-extensions	BZ#1717947
gnome-software	BZ#1668760
gnutls	BZ#1628553
golang	BZ#2075162
grub2	BZ#2074762, BZ#1583445
grubby	BZ#1978226, BZ#1900829
initscripts	BZ#1875485
ipa	BZ#2059396, BZ#2022028, BZ#782917, BZ#2062379, BZ#1924707, BZ#2120572, BZ#2122919, BZ#1664719, BZ#1664718, BZ#2101770
ipmitool	BZ#1873614
iptables	BZ#2058444
kdump-anaconda-addon	BZ#2086100
内核	BZ#2084242, BZ#2062870, BZ#2068429, BZ#2101938, JIRA:RHELPLAN-121252, BZ#2096127, BZ#2054656, BZ#1868526, BZ#1694705, BZ#1730502, BZ#1609288, BZ#1602962, BZ#1865745, BZ#1906870, BZ#1924016, BZ#1942888, BZ#1812577, BZ#1910358, BZ#1930576, BZ#2046396, BZ#1793389, BZ#1654962, BZ#1940674, BZ#1920086, BZ#1971506, BZ#2059262, BZ#2050411, BZ#2106341, BZ#2130159, BZ#1605216, BZ#1519039, BZ#1627455, BZ#1501618, BZ#1633143, BZ#1814836, BZ#1696451, BZ#1348508, BZ#1837187, BZ#1904496, BZ#1660337, BZ#1905243, BZ#1878207, BZ#1665295, BZ#1871863, BZ#1569610, BZ#1794513
kexec-tools	BZ#2006000
kmod-kvdo	BZ#2109047
kmod	BZ#2103605

组件	票证
krb5	BZ#2026462 , BZ#2125182 , BZ#1877991
libdnf	BZ#2088149
libgnome-keyring	BZ#1607766
libguestfs	BZ#1554735
libpfm	BZ#2067218
libreswan	BZ#1989050
libselinux-python-2.8-module	BZ#1666328
libva	BZ#2099907
libvirt	BZ#1664592 , BZ#1332758 , BZ#2067126 , BZ#1528684
libvpd	BZ#2051319
llvm-toolset	BZ#2061042 , BZ#2088315
lsupd	BZ#2051316
lvm2	BZ#1496229 , BZ#1768536
make43	BZ#2083419
mariadb-java-client	BZ#2043212
mariadb	BZ#1944653 , BZ#1942330
maven	BZ#2083114
mercurial	BZ#2089849
mesa	BZ#1886147
motif	BZ#2060571
nfs-utils	BZ#1946283 , BZ#2087187 , BZ#2081114 , BZ#1592011
nispor	BZ#1848817

组件	票证
nodejs	BZ#2083073
nss_nis	BZ#1803161
nss	BZ#1817533 , BZ#1645153 , BZ#2097837
open-vm-tools	BZ#2061193
opencryptoki	BZ#2043845
opencv	BZ#2104776 , BZ#1886310
openmpi	BZ#1866402
opensc	BZ#1947025
openssh	BZ#2044354
openssl	BZ#1810911
osbuild-composer	BZ#2065734
oscap-anaconda-addon	BZ#2075508 , BZ#1843932 , BZ#1665082
pacemaker	BZ#2036815 , BZ#2059638 , BZ#1182956 , BZ#1724310
papi	BZ#2037426, BZ#2071558 , BZ#2037427 , BZ#2037417
pcs	BZ#1786964 , BZ#1954099 , BZ#2023845 , BZ#1950551 , BZ#1909904 , BZ#1874624 , BZ#1619620 , BZ#1847102 , BZ#1851335
pki-core	BZ#1729215 , BZ#2134093 , BZ#1628987
podman	BZ#2097708 , JIRA:RHELPLAN-77238
postfix	BZ#1711885
powerpc-utils	BZ#2078514, BZ#2051330
ppc64-diag	BZ#2051313
procps-ng	BZ#2111915
pykickstart	BZ#1637872

组件	票证
qemu-kvm	BZ#2043830 , BZ#2020133 , BZ#1740002 , BZ#1719687 , BZ#1966475 , BZ#1792683 , BZ#1651994
rear	BZ#2072978 , BZ#2115918 , BZ#2077404 , BZ#2021935 , BZ#2035872 , BZ#1925531 , BZ#1868421 , BZ#2083301
redhat-support-tool	BZ#2064575 , BZ#1802026
redis	BZ#1999873
restore	BZ#1997366
rhel-system-roles	BZ#2064067 , BZ#2115884 , BZ#2060377 , BZ#2072749 , BZ#2083378 , BZ#2100285 , BZ#2060378 , BZ#2083426 , BZ#2100298 , BZ#2101607 , BZ#2115161 , BZ#2109997 , BZ#2065339 , BZ#2086869 , BZ#1996731 , BZ#2065670 , BZ#2043009 , BZ#2112143 , BZ#2066876 , BZ#2071011 , BZ#2075116 , BZ#2079008 , BZ#2086934 , BZ#2086935 , BZ#2093437 , BZ#2079114 , BZ#2056480 , BZ#2065215 , BZ#2065216 , BZ#2065218 , BZ#2100297 , BZ#2100939 , BZ#2100979 , BZ#2115159 , BZ#2115160 , BZ#2115162 , BZ#2021685 , BZ#2006081
rpm	BZ#1688849
rsync	BZ#2139118
rsyslog	BZ#1962318 , BZ#1679512 , JIRA:RHELPLAN-10431
rust-toolset	BZ#2075344
s390utils	BZ#1660911
samba	BZ#2077468 , BZ#2009213 , JIRA:RHELPLAN-13195 , Jira:RHELDPCS-16612
sblim-wbemcli	BZ#2075807
scap-security-guide	BZ#2064696 , BZ#2075384 , BZ#2077531 , BZ#2078974 , BZ#2083109 , BZ#2109602 , BZ#2070564 , BZ#2058203 , BZ#1967947 , BZ#2032403 , BZ#2112937 , BZ#2028428 , BZ#1858866 , BZ#1750755 , BZ#2038977 , BZ#2119356
selinux-policy	BZ#1461914
scs	BZ#2011413

组件	票证
spice	BZ#1849563
sssd	BZ#2065692 , BZ#2056483 , BZ#1947671
systemtap	BZ#2057565
ubi8-container	BZ#2120378
udica	BZ#1763210
unbound	BZ#2027735
vdo	BZ#1949163
virt-manager	BZ#2026985
vulkan-loader	BZ#2012639
wayland	BZ#1673073
weldr-client	BZ#2033192
xmlstarlet	BZ#1882020
xorg-x11-server	BZ#1698565
xorg-x11-xtrans-devel	BZ#2075132

组件	票证
其他	<p>JIRA:RHELPLAN-109067, JIRA:RHELPLAN-115603, BZ#2020301, BZ#2125545, BZ#2128016, JIRA:RHELPLAN-121982, JIRA:RHELPLAN-118463, JIRA:RHELPLAN-100037, BZ#1497089, JIRA:RHELPLAN-121981, JIRA:RHELPLAN-121983, JIRA:RHELPLAN-121980, BZ#2049492, JIRA:RHELPLAN-98420, JIRA:RHELPLAN-100039, JIRA:RHELPLAN-123369, JIRA:RHELPLAN-130379, JIRA:RHELPLAN-130376, JIRA:RHELPLAN-122735, BZ#2070793, BZ#2122716, JIRA:RHELPLAN-135602, JIRA:RHELPLAN-136150, BZ#2139821, BZ#2025814, BZ#2077770, BZ#1777138, BZ#1640697, BZ#1697896, BZ#1961722, BZ#1659609, BZ#1687900, BZ#1757877, BZ#1741436, JIRA:RHELPLAN-59111, JIRA:RHELPLAN-27987, JIRA:RHELPLAN-34199, JIRA:RHELPLAN-57914, JIRA:RHELPLAN-96940, BZ#1974622, BZ#2028361, BZ#2041997, BZ#2035158, JIRA:RHELPLAN-109613, BZ#2126777, BZ#1690207, JIRA:RHELPLAN-1212, BZ#1559616, BZ#1889737, JIRA:RHELPLAN-14047, BZ#1769727, JIRA:RHELPLAN-27394, JIRA:RHELPLAN-27737, BZ#1906489, JIRA:RHELPLAN-75165, JIRA:RHELPLAN-118470, JIRA:RHELPLAN-122316, BZ#1642765, JIRA:RHELPLAN-10304, BZ#1646541, BZ#1647725, BZ#1932222, BZ#1686057, BZ#1748980, JIRA:RHELPLAN-71200, BZ#1827628, JIRA:RHELPLAN-45858, BZ#1871025, BZ#1871953, BZ#1874892, BZ#1916296, JIRA:RHELPLAN-100400, BZ#1926114, BZ#1904251, BZ#2011208, JIRA:RHELPLAN-59825, BZ#1920624, JIRA:RHELPLAN-70700, BZ#1929173, JIRA:RHELPLAN-85066, BZ#2006665, JIRA:RHELPLAN-98983, BZ#2009113, BZ#1958250, BZ#2038929, BZ#2029338, BZ#2061288, BZ#2060759, BZ#2055826, BZ#2059626, JIRA:RHELPLAN-133171, BZ#2142499</p>

附录 B. 修改历史记录

0.2-8

2024 年 5 月 9 日星期四, Brian Angelica (bangelic@redhat.com)

- 在 [BZROX0207](#) 中更新了技术预览。

0.2-7

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)。

0.2-6

2024 年 2 月 29 日星期四, Gabriela Fialová(gfialova@redhat.com)

- 添加了一个已弃用的功能 [JIRA:RHELDOCS-17641](#) (网络)。

0.2-5

2024 年 2 月 13 日星期二, Lucie Vařáková (lvarakova@redhat.com)

- 添加了一个已弃用的功能 [JIRA:RHELDOCS-17573](#) (身份管理)

0.2-4

2024 年 2 月 2 日星期五, Lucie Vařáková (lvarakova@redhat.com)

- 添加了一个已知问题 [BZ#1834716](#) (安全)。

0.2-3

2023 年 11 月 10 日星期五, Gabriela Fialová(gfialova@redhat.com)

- 更新了对 RHEL 文档提供反馈的模块。

0.2-2

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)。

0.2-1

2023 年 10 月 9 日, Lucie Vařáková (lvarakova@redhat.com)

- 更新了一个已知问题 [BZ#2169382](#) (网络)。

0.2-0

2023 年 9 月 8 日星期五, Lucie Vařáková(lvarakova@redhat.com)

- 添加了一个已弃用的功能发行注记 [JIRA:RHELDOCS-16612](#) (Samba)。
- 更新了 [对 Red Hat 文档提供反馈](#) 部分。

0.1-9

2023 年 8 月 24 日星期四, Lucie Vařáková (lvarakova@redhat.com)

- 添加了一个已知问题 [BZ#2014500](#) (内核)

- 添加了一个已知问题 [BZ#2214508](#) (内核)。

0.1-8

2023 年 8 月 4 日星期五, Lenka Špačková (lspackova@redhat.com)

- 修复了 [BZ#2225332](#) 部分。

0.1-7

2023 年 8 月 1 日星期二, Lenka Špačková (lspackova@redhat.com)

- 添加了已弃用的功能 [BZ#2225332](#)。
- 改进了抽象。

0.1-6

2023 年 7 月 17 日星期一, Gabriela Fialová(gfialova@redhat.com)

- 修复了 [BZ#2072749](#) (系统角色) 中的一个错误。

0.1-5

2023 年 6 月 29 日星期四, Marc Muehlfeld (mmuehlfeld@redhat.com)

- 添加了一个技术预览 [BZ#1570255](#)(内核)。

0.1-4

2023 年 6 月 16 日星期五, Lucie Vařáková (lvarakova@redhat.com)

- 添加了一个已知问题 [BZ#2214235](#) (内核)。

0.1-3

2023 年 6 月 15 日星期四, Lucie Vařáková (lvarakova@redhat.com)

- 添加了一个增强 [BZ#2070347](#) (引导加载程序)。

0.1-3

2023 年 5 月 18 日, Gabriela Fialová(gfialova@redhat.com)

- 添加了一个改进 [BZ#2083077](#) (文件系统和存储)。

0.1-2

2023 年 5 月 10 日星期三, Jaroslav Klech (jklech@redhat.com)

- 添加了一个已知问题 [BZ#2169382](#)(网络)。

0.1-1

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)。

0.1-0

2023 年 4 月 18 日星期二, Lenka Špačková (lspackova@redhat.com)

- 添加了异步更新 [BZ#2178087](#) 的一个增强 (动态编程语言、Web 和数据库服务器)。

0.0-9

2023 年 4 月 13 日星期四, Gabriela Fialová (gfialova@redhat.com)

- 修复了 DF 和 KI 中的 2 个坏掉的链接。

0.0-8

Feb 17 2023, Lucie Vařáková (lvarakova@redhat.com)

- 添加了一个已知问题 [BZ#2158783](#) (文件系统和存储)。

0.0-7

2023 年 2 月 14 日, Marc Muehlfeld (mmuehlfeld@redhat.com)

- 添加了一个增强 [BZ#2144898](#) (网络)。

0.0-6

Feb 08 2023, Lucie Vařáková (lvarakova@redhat.com)

- 添加了 bug 修复 [BZ#2046396](#) (内核) 和 [BZ#2069379](#) (身份管理)。
- 其他小更新。

0.0-5

2023 年 1 月 24 日, Lucie Vařáková(lvarakova@redhat.com)

- 添加了一个已知问题 [BZ#2115791](#) (云环境中的RHEL)。

0.0-4

2023 年 1 月 18 日, Lucie Vařáková(lvarakova@redhat.com)

- 添加了一个已知问题 [BZ#1920086](#) (内核)。

0.0-3

2022 年 12 月 7 日, Lucie Vařáková (lvarakova@redhat.com)

- 将 **nodejs:18** 模块流 [BZ#2083073](#) 从技术预览移到完全支持的功能(动态编程语言、web 和数据库服务器)。
- 添加了一个已知问题 [BZ#2132754](#) (网络)。

0.0-2

2022 年 11 月 23 日, Gabriela Fialová(gfialova@redhat.com)

- 目录服务器 RNs 的发布和 RHEL 8.7 RNs 的后续重新发布。

0.0-1

2022 年 11 月 9 日, Lucie Vařáková (lvarakova@redhat.com)

- Red Hat Enterprise Linux 8.7 发行注记的发布。

0.0-0

2022 年 9 月 28 日, Lucie Vařáková (lvarakova@redhat.com)

- 发布 Red Hat Enterprise Linux 8.7 Beta 发行注记。