



# Red Hat Enterprise Linux 9

## 9.0 发行注记

Red Hat Enterprise Linux 9.0 发行注记



# Red Hat Enterprise Linux 9 9.0 发行注记

---

Red Hat Enterprise Linux 9.0 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<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

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

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

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

Node.js<sup>®</sup> 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<sup>®</sup> 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 9.0 中实现的改进和附加组件的高级信息，记录了本版本中已知的问题，以及重要的 bug 修复、技术预览、弃用的功能和其他详情。有关如何安装 Red Hat Enterprise Linux 的详情，请参考 Installation 部分。

# 目录

对红帽文档提供反馈 .....	5
<b>第 1 章 概述 .....</b>	<b>6</b>
1.1. RHEL 9.0 的主要变化	6
1.2. 原位升级	9
1.3. 红帽客户门户网站 LABS	10
1.4. 其它资源	10
<b>第 2 章 构架 .....</b>	<b>12</b>
<b>第 3 章 RHEL 9 发布的内容 .....</b>	<b>13</b>
3.1. 安装	13
3.2. 软件仓库	13
3.3. 应用程序流	13
3.4. 使用 YUM/DNF 的软件包管理	14
<b>第 4 章 新功能 .....</b>	<b>15</b>
4.1. 安装程序和镜像创建	15
4.2. RHEL FOR EDGE	16
4.3. 订阅管理	17
4.4. 软件管理	18
4.5. SHELL 和命令行工具	19
4.6. 基础架构服务	22
4.7. 安全性	24
4.8. 网络	33
4.9. 内核	36
4.10. 引导装载程序	42
4.11. 文件系统和存储	42
4.12. 高可用性和集群	44
4.13. 动态编程语言、网页和数据库服务器	47
4.14. 编译器和开发工具	53
4.15. 身份管理	61
4.16. DESKTOP	66
4.17. 图形基础结构	70
4.18. WEB 控制台	70
4.19. RED HAT ENTERPRISE LINUX 系统角色	71
4.20. 虚拟化	76
4.21. 云环境中的 RHEL	77
4.22. 支持性	79
4.23. 容器	79
<b>第 5 章 BUG 修复 .....</b>	<b>83</b>
5.1. 安装程序和镜像创建	83
5.2. 订阅管理	83
5.3. 软件管理	84
5.4. SHELL 和命令行工具	84
5.5. 安全性	84
5.6. 网络	86
5.7. 内核	86
5.8. 文件系统和存储	87
5.9. 高可用性和集群	87
5.10. 编译器和开发工具	87

5.11. 身份管理	87
5.12. RED HAT ENTERPRISE LINUX 系统角色	88
5.13. 虚拟化	92
5.14. 容器	92
<b>第 6 章 技术预览</b>	<b>93</b>
6.1. RHEL FOR EDGE	93
6.2. SHELL 和命令行工具	93
6.3. 网络	93
6.4. 内核	94
6.5. 文件系统和存储	94
6.6. 编译器和开发工具	95
6.7. 身份管理	95
6.8. DESKTOP	97
6.9. WEB 控制台	98
6.10. 虚拟化	98
6.11. 容器	99
<b>第 7 章 过时的功能</b>	<b>100</b>
7.1. 安装程序和镜像创建	100
7.2. SHELL 和命令行工具	100
7.3. 安全性	101
7.4. 网络	102
7.5. 内核	103
7.6. 文件系统和存储	103
7.7. 动态编程语言、网页和数据库服务器	103
7.8. 身份管理	104
7.9. 图形基础结构	104
7.10. RED HAT ENTERPRISE LINUX 系统角色	105
7.11. 虚拟化	105
7.12. 容器	106
7.13. 已弃用的软件包	106
<b>第 8 章 已知问题</b>	<b>108</b>
8.1. 安装程序和镜像创建	108
8.2. 订阅管理	110
8.3. 软件管理	110
8.4. SHELL 和命令行工具	111
8.5. 基础架构服务	112
8.6. 安全性	112
8.7. 网络	116
8.8. 内核	117
8.9. 引导装载程序	118
8.10. 文件系统和存储	119
8.11. 动态编程语言、网页和数据库服务器	119
8.12. 编译器和开发工具	120
8.13. 身份管理	120
8.14. DESKTOP	123
8.15. 图形基础结构	123
8.16. WEB 控制台	124
8.17. 虚拟化	125
8.18. 云环境中的 RHEL	126
8.19. 支持性	127
8.20. 容器	128

---

附录 A. 按组件划分的问题单列表 .....	130
附录 B. 致谢 .....	138
附录 C. 修改历史记录 .....	139



---

## 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 9.0 的主要变化

### 安全性

使用 **SHA-1** 消息摘要用于加密目的在 RHEL 9 中已被弃用。SHA-1 生成的摘要不被视为是安全的，因为已发现多个基于哈希进行的安全攻击。RHEL 核心加密组件不再默认使用 SHA-1 创建签名。RHEL 9 中的应用程序已更新，以避免在与安全相关的用例中使用 SHA-1。

其中一个例外是，仍然可以使用 SHA-1 创建 HMAC-SHA1 消息验证代码和 Universal Unique Identifier(UUID)值，因为这些用例目前不存在安全风险。另外，为了保持一些重要的互操作性和兼容性，SHA-1 还会在一些有限的情况下使用，例如 Kerberos 和 WPA-2。如需了解更多详细信息，请参阅[使用不符合 FIPS 140-3 的加密系统的 RHEL 应用程序列表](#)。

有关仍需要 SHA-1 的系统的兼容性问题的解决方案，请查看以下 KCS 文章：

- [从 RHEL 9 SSH 到 RHEL 6 系统无法正常工作](#)
- [使用 SHA-1 签名的软件包无法安装或升级](#)
- [与不支持"server-sig-algs"扩展的 SSH 服务器和客户端的连接失败](#)

**OpenSSL** 现在提供的版本为 3.0.1，它添加了一个供应商 (provide) 概念、一个新的版本控制方案、改进的 HTTP(S)客户端、支持新的协议、格式和算法以及许多其他改进。

对系统范围的**加密策略**进行了调整，以提供最新的安全默认值。

**OpenSSH** 在 8.7p1 版本中发布，与版本 8.0p1 相比（在 RHEL 8.5 中发布），它提供了很多改进、错误修复和安全改进。

SFTP 协议取代了以前在 **OpenSSH** 中使用的 SCP/RCP 协议。SFTP 提供了更可预测的文件名处理方式，远程端的 shell 不再需要 **glob(3)** 模式扩展。

**SELinux** 性能显著提高，包括将 SELinux 策略加载到内核、内存开销和其他参数的时间。如需更多信息，请参阅验证 SELinux 博客文章的[性能和空间效率](#)。

RHEL 9 提供了上游版本 1.1 中的 **fapolicyd** 架构。在其他改进中，您现在可以使用新的 **rules.d/** 和 **trust.d/** 目录、**fagenrules** 脚本和 **fapolicyd-cli** 命令的新选项。

SCAP 安全指南(SSG)软件包在 0.1.60 版本中提供，它引入了 delta 定制、更新了安全配置集并包括一些其他改进。

如需更多信息，请参阅 [第 4.7 节“安全性”](#)。

在 DEFAULT 加密策略中对使用 SHA-1 进行签名进行了限制。除了 HMAC 外，TLS、DTLS、SSH、IKEv2、DNSSEC 和 Kerberos 协议中不再允许使用 SHA-1。

如果您需要使用 SHA-1 来验证现有或第三方加密签名，您可以输入以下命令启用它：

```
# update-crypto-policies --set DEFAULT:SHA1
```

或者，您可以将系统范围的加密策略切换到 **LEGACY** 策略。请注意，**LEGACY** 也启用了其他一些不安全的算法。

Cyrus SASL 现在使用 GDBM 而不是 Berkeley DB，而网络安全服务(NSS)库不再支持信任数据库的 DBM 文件格式。

在内核中删除了通过 `/etc/selinux/config` 文件中的 **SELINUX=disabled** 选项禁用 SELinux 的支持。当您只通过 `/etc/selinux/config` 禁用 SELinux 时，系统会在启用 SELinux 的情况下启动，但不会载入策略。如果您需要禁用 SELinux，请在内核命令行中添加 **selinux=0** 参数。

有关 RHEL 9 和 RHEL 8 之间的安全性区别的更多信息，请参阅 *使用 RHEL 9 的注意事项* 中的 [安全](#) 部分。

## 网络

您可以使用新的 MultiPath TCP 守护进程(mptcpd)配置 MultiPath TCP(MPTCP)端点，而无需使用 **iproute2** 实用程序。要使 MPTCP 子流和端点持久，请使用 NetworkManager 分配器脚本。

默认情况下，NetworkManager 现在使用密钥文件存储新的连接配置集。请注意，**ifcfg** 格式仍被支持。

有关本发行版本中引入的功能以及现有功能更改的更多信息，请参阅 [新功能 - 网络](#)。

WireGuard VPN 技术现在作为技术预览提供。详情请查看 [技术预览 - 网络](#)。

**teamd** 服务和 **libteam** 库已弃用。作为替换，配置绑定而不是网络组。

**iptables-nft** 和 **ipset** 已被弃用。这些软件包包括了相关的工具，如 **iptables**、**ip6tables**、**ebtables** 和 **arptables**。使用 **nftables** 框架配置防火墙规则。

有关已弃用功能的更多信息，请参阅 [已弃用功能 - 网络](#)。

**network-scripts** 软件包已被删除。使用 NetworkManager 配置网络连接。有关不再是 RHEL 功能一部分的更多信息，请参阅 *采用 RHEL 9 时的考虑* 文档中的 [网络](#) 部分。

## 动态编程语言、网页和数据库服务器

RHEL 9.0 提供以下动态编程语言：

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

RHEL 9.0 包括以下版本控制系统：

- Git 2.31
- Subversion 1.14

以下 web 服务器随 RHEL 9.0 一起发布：

- Apache HTTP Server 2.4.51
- nginx 1.20

以下代理缓存服务器可用：

- Varnish Cache 6.6
- Squid 5.2

RHEL 9.0 提供以下数据库服务器：

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

如需更多信息，请参阅 [第 4.13 节 “动态编程语言、网页和数据库服务器”](#)。

## 编译器和开发工具

### 系统工具链

RHEL 9.0 提供了以下系统工具链组件：

- GCC 11.2.1
- glibc 2.34
- binutils 2.35.2

RHEL 9 系统工具链组件包括对 POWER10 的支持。

### 性能工具和调试器

RHEL 9.0 提供了以下性能工具和调试器：

- GDB 10.2
- Valgrind 3.18.1
- SystemTap 4.6
- Dyninst 11.0.0
- elfutils 0.186

### 性能监控工具

RHEL 9.0 有以下性能监控工具：

- PCP 5.3.5
- Grafana 7.5.11

### 编译器工具集

RHEL 9.0 提供以下编译器工具集：

- LLVM Toolset 13.0.1
- Rust Toolset 1.58.1
- Go Toolset 1.17.7

具体更改请查看 [第 4.14 节 “编译器和开发工具”](#)。

## RHEL 9 中的 Java 实现

RHEL 9 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 9.0 提供以下 Java 工具：

- **Maven 3.6**
- **Ant 1.10**

如需更多信息，请参阅 [第 4.14 节“编译器和开发工具”](#)。

### Desktop

GNOME 环境已从 GNOME 3.28 更新至具有许多新功能的 GNOME 40。

**X.org** 显示服务器已弃用，并将在以后的主 RHEL 发行版本中删除。现在，在大多数情形中，默认桌面会话都是 **Wayland** 会话。

当使用 NVIDIA 驱动程序时，如果驱动程序配置支持 Wayland，则桌面会话现在默认选择 Wayland 显示协议。在以前的 RHEL 版本中，NVIDIA 驱动程序总是禁用 Wayland。

**PipeWire** 服务现在管理所有音频输出和输入。在常用用例中，**PipeWire** 代替了 **PulseAudio** 服务，并在专业用例中代替了 **JACK** 服务。

如需更多信息，请参阅 [第 4.16 节“Desktop”](#)。

### 虚拟化

在 RHEL 9 中，**libvirt** 库使用 modular 守护进程来处理您主机上的单个虚拟化驱动程序集。这样便可对涉及虚拟化驱动程序的各种任务进行微调，如资源负载优化和监控。

QEMU 模拟器现在使用 Clang 编译器构建。这可让 RHEL 9 KVM 管理程序使用许多高级安全和调试功能。其中一个功能是 SafeStack，它可以使在 RHEL 9 上托管的虚拟机(VM)显著减小了针对基于面向返回编程(ROP)的攻击风险。

另外，vTPM (Virtual Trusted Platform Module) 现已被完全支持。通过使用 vTPM，您可以为虚拟机添加一个 TPM 虚拟加密处理器，然后可用于生成、存储和管理加密密钥。

最后，**virtiofs** 功能已经实现，可用于更有效地在 RHEL 9 主机及其虚拟机间共享文件。

有关这个版本中引入的虚拟化功能的更多信息，请参阅 [第 4.20 节“虚拟化”](#)。

## 1.2. 原位升级

### 从 RHEL 8 原位升级到 RHEL 9

- 在以下构架中，从 RHEL 8.6 升级到 RHEL 9.0：
  - 64 位 Intel
  - 64 位 AMD
  - 64-bit ARM

- IBM POWER 9(little endian)
- IBM Z 架构，不包括 z13
- 在使用 SAP HANA 的系统上，从 RHEL 8.6 升级到 RHEL 9.0

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 8 升级到 RHEL 9](#)。

有关在具有 SAP 环境的系统上执行原位升级的说明，请参阅[如何将 SAP 环境从 RHEL 8 原位升级到 RHEL 9](#)。

### 从 RHEL 7 原位升级到 RHEL 9

无法执行从 RHEL 7 直接升级到 RHEL 9 的原位升级。但是，您可以执行从 RHEL 7 原位升级到 RHEL 8，然后再执行到 RHEL 9 的第二个原位升级。如需更多信息，请参阅[从 RHEL 7 升级到 RHEL 8](#)。

## 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](#)
- [Kickstart Generator](#)
- [Red Hat Product Certificates](#)
- [Red Hat CVE Checker](#)
- [Kernel Oops Analyzer](#)
- [Red Hat Code Browser](#)
- [VNC 配置器](#)
- [Red Hat OpenShift Container Platform Update Graph](#)
- [Red Hat Satellite Upgrade Helper](#)
- [JVM Options Configuration Tool](#)
- [Load Balancer Configuration Tool](#)
- [Red Hat OpenShift Data Foundation Supportability and Interoperability Checker](#)
- [Ansible Automation Platform Upgrade Assistant](#)
- [每个池计算器的 Ceph 放置组 \(PG\)](#)

## 1.4. 其它资源

与其他版本系统相比，Red Hat Enterprise Linux 9 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 9 提供 **软件包列表**，包括许可证和应用程序兼容性等级。

**应用程序兼容性等级** 的信息包括在 [Red Hat Enterprise Linux 9 : 应用程序兼容性指南](#) 文档。

**RHEL 8 和 RHEL 9 的主要区别**（包括删除的功能）包括在 [使用 RHEL 9 时的注意事项](#)。

有关如何执行从 **RHEL 8 到 RHEL 9 的原位升级** 的说明，请参考 [从 RHEL 8 升级到 RHEL 9](#) 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 9.0 带有内核版本 5.14.0，它支持满足最低版本要求的以下架构：

- AMD 和 Intel 64 位体系架构 (x86-64-v2)
- 64 位 ARM 架构(ARMv8.0-A)
- IBM Power Systems, Little Endian(POWER9)
- 64 位 IBM Z (z14)

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](#)。

## 第 3 章 RHEL 9 发布的内容

### 3.1. 安装

Red Hat Enterprise Linux 9 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：

- 安装 ISO：一个包含 BaseOS 和 AppStream 软件仓库的完整的安装镜像，允许您在没有其它软件仓库的情况下完成安装。在[产品下载页面](#)中，**安装 ISO** 被称为 **Binary DVD**。



#### 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：用于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。软件仓库是安装 ISO 镜像的一部分。您还可以在安装过程中注册红帽 CDN 或 Satellite，以使用来自红帽 CDN 或 Satellite 的最新 BaseOS 和 AppStream 内容。

有关下载 ISO 镜像、创建安装介质以及完成 RHEL 安装的说明，请参阅[执行标准的 RHEL 9 安装](#)文档。有关自动 Kickstart 安装和其他高级主题，请参阅[执行高级 RHEL 9 安装](#)文档。

### 3.2. 软件仓库

Red Hat Enterprise Linux 9 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。如需更多信息，请参阅[覆盖范围](#)详情文档。

AppStream 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。

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 9 软件仓库及其提供的软件包的更多信息，请参阅[软件包清单](#)。

### 3.3. 应用程序流

用户空间组件的多个版本会以 Application Streams（应用程序流）的形式提供，其更新频率会比核心操作系统软件包的更新频率更快。这为自定义 RHEL 提供了更大的灵活性，而不影响平台或特定部署的基本稳定性。

应用程序流以 RPM 格式提供，可以是一个模块（RPM 格式的一个扩展），软件集合（Software Collections），或 Flatpaks。

每个 Application Stream 组件都有其特定的生命周期，可能和 RHEL 9 的生命周期相同或更短。有关 RHEL 生命周期信息，请查看 [Red Hat Enterprise Linux 生命周期](#)。

RHEL 9 改进了应用程序流的使用体验，它提供了初始的应用程序流版本，可以使用传统的 `dnf install` 命令作为 RPM 软件包进行安装。



### 注意

某些 RPM 格式的初始应用程序流的生命周期比 Red Hat Enterprise Linux 9 的生命周期短。

一些额外的 Application Stream 版本将作为模块发布，并在以后的 RHEL 9 次要发行本中带有较短的生命周期。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。

始终决定要安装哪个版本的应用程序流，并确保首先查看 [Red Hat Enterprise Linux 应用程序流生命周期](#)。

需要快速更新的内容（例如备用编译器和容器工具）会在滚动流中提供，且不会并行提供替代版本。滚动流可以打包为 RPM 或模块。

有关 RHEL 9 中可用的 Application Streams 及其应用程序兼容性级别的详情，请查看 [软件包清单](#)。应用程序兼容性等级的信息包括在 [Red Hat Enterprise Linux 9：应用程序兼容性指南](#) 文档。

## 3.4. 使用 YUM/DNF 的软件包管理

在 Red Hat Enterprise Linux 9 中，使用 **DNF** 确保软件安装。红帽继续支持使用 **yum** 术语，以便与以前的 RHEL 主版本保持一致。如果您键入 **dnf** 而不是 **yum**，则命令按预期运行，因为它们都是兼容性的别名。

虽然 RHEL 8 和 RHEL 9 基于 **DNF**，但它们与 RHEL 7 中使用的 **YUM** 兼容。

如需更多信息，请参阅使用 [DNF 工具管理软件](#)。

## 第 4 章 新功能

这部分论述了 Red Hat Enterprise Linux 9.0 中引入的新功能和主要增强。

### 4.1. 安装程序和镜像创建

#### Anaconda 通过 Satellite 的 Kickstart 安装支持用于机器置备的 rhsm

在以前的版本中，机器置备取决于 Red Hat Satellite 上 Kickstart 安装的自定义 `%post` 脚本。此 `%post` 脚本导入自定义 Satellite 自签名证书、注册机器、附加订阅，并安装了存储库中的软件包。

有了 RHEL 9，Satellite 支持已被添加到用于机器置备的 `rhsm` 命令中。现在，您可以对所有置备任务使用 `rhsm`，如注册系统、附加 RHEL 订阅，以及从 Satellite 实例安装。

(BZ#1951709)

#### RHEL 支持 localhost 作为静态主机名

从 RHEL 9 开始，在 `/etc/hostname` 中将 `localhost` 设为静态主机名是有效的。在这种情况下，NetworkManager 不会尝试通过 DHCP 或反向 DNS 查找获取临时主机名。

(BZ#2190045)

#### 在标准安装后禁用了许可证、系统和用户设置配置屏幕

在以前的版本中，RHEL 用户在 `gnome-initial-setup` 和登录屏幕之前配置许可、系统(订阅管理器)和用户设置。在这个版本中，初始设置页面已被默认禁用，以改进用户体验。

如果需要运行初始设置以使用户创建或许可证显示，请根据要求安装以下软件包。

1. 安装初始安装软件包。

```
# dnf install initial-setup initial-setup-gui
```

2. 在下次重启系统时启用初始设置。

```
# systemctl enable initial-setup
```

3. 重启系统以查看初始设置。

对于 Kickstart 安装，在 `packages` 部分添加 `initial-setup-gui` 并启用 `initial-setup` 服务。

```
firstboot --enable
%packages
@^graphical-server-environment
initial-setup-gui
%end
```

(BZ#1878583)

#### Anaconda 会自动为互动安装激活网络

在以前的版本中，当执行交互式安装，而不需要有 Kickstart 或引导选项激活的网络时，用户必须在网络会话中手动激活网络。在这个版本中，Anaconda 会自动激活网络，无需用户访问网络 spoke 并手动激活它。



### 注意

在这个版本中，不会更改 Kickstart 安装的安装体验，并使用 **ip=** 引导选项安装。

([BZ#1978264](#))

### 镜像构建器现在支持文件系统配置

在这个版本中，您可以在蓝图中指定自定义文件系统配置，您可以使用所需的磁盘布局创建镜像。因此，通过具有非默认布局，您可以从安全基准中受益，与现有设置、性能以及防止磁盘不足错误的一致性。

要自定义蓝图中的文件系统配置，请设置以下自定义：

```
[[customizations.filesystem]]
mountpoint = "MOUNTPOINT"
size = MINIMUM-PARTITION-SIZE
```



### 注意

在蓝图中添加了文件系统自定义后，该文件系统将被转换为 LVM 分区。

([BZ#2011448](#))

### 用于锁定 root 账户和允许使用密码进行 root SSH 登陆的新选项

在 RHEL 图形安装中，已对 root 密码配置页面添加了以下新选项：

- 锁定 root 帐户：使用这个选项锁定对计算机的 root 访问权限。
- 允许使用密码进行 root SSH 登录：使用这个选项启用基于密码的身份验证。

要启用 **基于密码的 SSH root 登录**，请在启动安装过程前将以下行添加到 Kickstart 文件中。

```
%post
echo "PermitRootLogin yes" > /etc/ssh/sshd_config.d/01-permitrootlogin.conf
%end
```

([BZ#1940653](#))

### 镜像构建器现在支持创建可引导安装程序镜像

在这个版本中，您可以使用镜像构建器创建由 **tarball** 文件组成的可引导 ISO 镜像，其中包含一个根文件系统。因此，您可以使用可引导 ISO 镜像将 **tarball** 文件系统安装到裸机系统中。

([BZ#2019318](#))

## 4.2. RHEL FOR EDGE

### RHEL for Edge 现在默认支持 Greenboot 内置健康检查

在这个版本中，RHEL for Edge **Greenboot** 包括了带有 **watchdog** 功能的内置健康检查，以确保在重启时硬件不会挂起或冻结。因此，您可以从以下功能中获益：

- 使用 **watchdogs** 硬件的用户可以轻松地实现内置健康检查

- 一组默认健康检查，为内置操作系统组件提供值
- **watchdog** 现在作为默认预设存在，可以轻松地启用或禁用此功能
- 基于已可用的健康检查，创建自定义健康检查。

([BZ#2083036](#))

### RHEL 9 提供 rpm-ostree v2022.2

RHEL 9 带有 **rpm-ostree** 版本 v2022.2，它提供多个程序错误修复和增强。主要变更包括：

- 现在，可以使用新的 **--append-if-missing** 和 **--delete-if-present** kargs 标志以幂等方式更新内核参数。
- 现在，所有存储库查询都默认禁用 DNF 的 **Count Me** 功能，且只被对应的 **rpm-ostree-countme.timer** 和 **rpm-ostree-countme.service** 单元触发。请参阅 [countme](#)。
- 后处理逻辑现在可以处理 **user.ima** IMA 扩展属性。找到 **xattr** 扩展属性时，系统会自动将它转换为最终 **OSTree** 软件包内容的 **security.ima**。
- **treefile** 文件中有一个新的 **repo-packages** 字段。您可以使用它来将一组软件包固定到特定的软件仓库。

([BZ#1961324](#))

### RHEL 9 提供 OSTree v2021.2

RHEL 9 与 **OSTree** 软件包版本 v2021.2 一起发布，提供多个程序错误修复和增强。主要变更包括：

- 用于编写文件的新 API（在新的 **ostree-rs-ext** 项目中使用），以改进 tarball 中的导入。
- **rofiles-fuse** 命令现在处理 **xattrs** 扩展属性。备注：**rofiles-fuse** 被视为已弃用，请参阅 [#2281](#)。
- 对 **introspection** API 和测试的改进。

([BZ#1961254](#))

### rpm-ostree rebase 工具支持从 RHEL 8 升级到 RHEL 9

在这个版本中，您可以使用 **rpm-ostree rebase** 工具将 RHEL 8 系统升级到 RHEL 9。它完全支持在最新的 RHEL 8 更新和最新的 RHEL 9 更新间升级 RHEL for Edge 的默认软件包集合。

([BZ#2082306](#))

## 4.3. 订阅管理

### 在 subscription-manager syspurpose 下合并的系统目的命令

在以前的版本中，有两个不同的命令来设置系统目的的属性：**syspurpose** 和 **subscription-manager**。要在一个模块下统一所有系统目的的属性，**subscription-manager** 中的所有 **addons**, **role**, **service-level**, 和 **usage** 命令都已移至新的子模块 **subscription-manager syspurpose**。

新子模块之外的现有 **subscription-manager** 命令已弃用。在 RHEL 9 中删除了提供 **syspurpose** 命令行工具的独立软件包(**python3-syspurpose**)。

在这个版本中，可以使用 `subscription-manager` 的单一命令来查看、设置和更新所有系统目的属性。这会将所有现有系统目的命令替换为其同等版本可用的新子命令。例如，`subscription-manager role --set SystemRole` 变成 `subscription-manager syspurpose role --set SystemRole` 等等。

有关新命令、选项和其他属性的完整信息，请参阅 `subscription-manager` man page 中的 **SYSPURPOSE OPTIONS** 部分。

(BZ#1898563)

## 4.4. 软件管理

### RHEL 9 提供 RPM 4.16

RHEL 9 提供 RPM 版本 4.16。与版本 4.14 相比，重要的程序错误修复和增强包括：

- 新的 SPEC 功能，最重要的是：
  - 快速基于宏的依赖关系生成器
  - 允许生成动态构建依赖项的 `%generate_buildrequires` 部分
  - 元（未排序）依赖项
  - 增加了软件包构建的并行性
  - 表达式中的原生版本比较
  - 尖号 (^) 操作符，与波形符 (~) 相反
  - `%elif`、`%elifos` 和 `%elifarch` 语句
  - 可选的自动补丁和源编号
  - `%autopatch` 现在接受补丁范围
  - `%patchlist` 和 `%sourcelist` 部分
  - 构建时强制进行标头数据的 UTF-8 验证
- rpm 数据库现在基于 `sqlite` 库。为迁移和查询目的保留了对 `BerkeleyDB` 数据库的只读支持。
- 一个新的 `rpm-plugin-audit` 插件，用于发出交易的审计日志事件，之前内置在 RPM 自身中

(JIRA:RHELPLAN-80734)

### 新的RPM插件会通知fapolicyd关于 RPM事务过程中的更改

这个 `rpm` 软件包更新引进了新的 RPM 插件，该插件将 `fapolicyd` 框架与 RPM 数据库集成。插件通知了 RPM 事务期间已安装和更改的文件的 `fapolicyd`。因此，`fapolicyd` 现在支持完整性检查。

请注意，RPM 插件了替换 DNF 插件，因为它的功能不仅限于 YUM 事务，同时也包括了 RPM 的更改。

(BZ#1942549)

### RPM 现在支持 EdDSA 公钥算法

在这个版本中，`rpm` 命令支持使用 EdDSA 公钥算法签名密钥。因此，现在可使用 EdDSA 生成的密钥签名和验证软件包。

请注意，现在支持使用 EdDSA 签名密钥，RSA 仍然是 GnuPG 中的默认公钥算法。

([BZ#1962234](#))

### RPM 现在支持 Zstandard (zstd) 压缩算法

在这个版本中，默认的 RPM 压缩算法已切换到 Zstandard (**zstd**)。因此，用户可以受益于更快的软件包安装，这在大型环境中特别明显。

([JIRA:RHELPLAN-117903](#))

### 新的 DNF 选项 `exclude_from_weak_autodetect` 和 `exclude_from_weak`

在这个版本中，默认的 DNF 行为是不安装不需要的弱依赖项。要修改此行为，请使用以下新选项：

- **`exclude_from_weak_autodetect`**  
如果启用，则 **`exclude_from_weak_autodetect`** 选项会自动检测您系统中安装的弱依赖关系软件包 (Recommends: 或 Supplements:)。因此，这些弱依赖项不会作为弱依赖项进行安装，但如果进行拉取，则会作为常规依赖项进行安装。默认值为 **true**。
- **`exclude_from_weak`**  
如果启用，**`exclude_from_weak`** 选项可防止将软件包安装为弱依赖项 (Recommends: 或 Supplements:)。您可以使用软件包名称或 glob 指定软件包，使用逗号分隔它们。默认值为 **[]**。

([BZ#2005305](#))

### RHEL 9 提供 `libmodulemd 2.13.0`

RHEL 9 带有 `libmodulemd` 软件包 2.13.0 版本。与版本 2.9.4 相比，重要的程序错误修复和增强包括：

- 添加了对从模块中分离软件包的支持。
- 添加了对使用 `modulemd-validator` 工具的新 `--type` 选项验证 `modulemd-packager-v3` 文档的支持。
- Fortified 解析整数。
- 修复了多个 `modulemd-validator` 问题。

([BZ#1984403](#))

## 4.5. SHELL 和命令行工具

### 默认情况下，在 `bash` 中会启用括号粘贴 (bracketed paste)

`bash` `readline` 库版本 8.1 现已正式发布，它会默认启用括号粘贴模式。当您将文本粘贴到终端时，`bash` 会突出显示文本，您必须按 **enter** 键来执行粘贴命令。括号粘贴模式是默认设置，它可以避免意外执行恶意命令。

要为特定用户禁用括号粘贴模式，请将以下行添加到 `~/.inputrc` 中：

```
set enable-bracketed-paste off
```

要为所有用户禁用括号粘贴模式，请将以下行添加到 `/etc/inputrc` 中：

```
set enable-bracketed-paste off
```

当您禁用了括号粘贴模式时，命令会在粘贴时直接执行，而无需按 **Enter** 键进行确认。

([BZ#2079078](#))

### RHEL 9 包括 **powerpc-utils 1.3.9**

RHEL 9 提供 **powerpc-utils** 软件包版本 1.3.9。与版本 1.3.8 相比，重要的程序错误修复和增强包括：

- 将 **drmgr** 中的日志大小增加到 1 MB。
- 修复了引导时 **HCIND** 阵列大小的问题。
- 在 **hcnmgr** 的 HNV 连接中实施了 **autoconnect-slaves**。
- 改进了 **hcnmgr** 中的 HNV 绑定列表连接。
- 在 **hcnmgr** 中使用来自 **util-linux** 的 **hexdump**。
- **hcn-init.service** 使用 NetworkManager 开始。
- 修复了在 **pathname** 中针对多路径的逻辑 FC 查找问题。
- 修复了使用 **pathname** 中的分区进行逻辑查找的问题。
- 修复了大于 5 个路径的多路径设备的引导列表。
- 在 **pathname** 的 **l2of\_vd()** 中添加了缺少的子字符串提取 **devpart**。
- 引入了 **lpamumascore**。
- 修复了 **drmgr** 中的 **索引操作** 所移除的问题。
- 将 **SYS\_PATH** 的定义从 **l2of\_vs()** 移动到 **pathname** 中的 **l2of\_scsi()**。
- 添加了 **-x** 选项，以便在 **partstat** 中增强安全性。
- 修复了 **lparstat** man page 中的 **nroff** 警告和错误。
- 在 **drmgr** 中实施基于 NUMA 的 LMB 移除。
- 在 **hcnmgr** 中修复了与 **udev** 重命名的 **ofpathname** 竞争。
- 使用 **NetworkManager nmcli** 检查 **hcnmgr** 中的绑定接口状态。
- 当 HNV 不存在时，使用 **NetworkManager nmcli** 清除绑定接口。

([BZ#1873868](#))

### RHEL 9 带有 **opal-prd 6.7.1**

与之前可用的版本 6.6.3 相比，**opal-prd** 软件包版本 6.7.1 提供了以下显著的程序错误修复和增强：

- 修复了因为 **xscom OPAL** 调用导致的 **xscom** 错误日志问题。
- 修复了使用 **DEBUG** 构建的可能死锁。
- 如果 **fast-reboot** 在 **core/platform** 中失败，会回调到 **full\_reboot**。
- 修复了 **core/cpu** 中的 **next\_ungarded\_primary**。

- 改进了速率限制计时器请求以及自助服务引擎(SBE)中的计时器状态。

(BZ#1869560)

### RHEL 9 提供 **Isvcpd 1.7.12**

RHEL 9 与 **Isvcpd** 软件包版本 1.7.12 一起发布。与版本 1.7.11 相比，重要的程序错误修复和增强包括：

- 在 **sysvcpd** 中添加了 UUID 属性。
- 改进了 **NVMe** 固件版本。
- 修复了 PCI 设备厂商解析逻辑。
- 在 **Isvcpd** 配置文件中添加 **recommend** 子句。

(BZ#1869564)

### **ppc64-diag** 版本 2.7.7 可用

RHEL 9 提供了 **ppc64-diag** 软件包版本 2.7.7。与 2.7.6 版本相比，重要的程序错误修复和增强包括：

- 改进了单元测试案例。
- 在 **sysvcpd** 中添加了 UUID 属性。
- **rtas\_errd** 服务不在 Linux 容器中运行。
- **systemd** 服务文件中不再提供过时的日志记录选项。

(BZ#1869567)

### RHEL 9 包括 **Fetchmail 6.4.24**

RHEL 9 提供了 **fetchmail** 软件包 6.4.24。**fetchmail** 是一个远程邮件检索和转发实用程序。

如需更多信息，请参阅：

- `/usr/share/doc/fetchmail/NEWS` 文件，
- **fetchmail(1)** 手册页，
- 如果需要更改配置，请参阅 `/usr/share/doc/fetchmail/README.SSL` 以获得与 SSL 相关的信息。

(BZ#1999276)

### RHEL 9 包括 **Eigen 3.4**

RHEL 9 包括 **eigen3** 软件包版本 3.4。**Eigen 3.4** 是一个用于线性算法的 C++ 模板库，现在支持 `POWER10 matrix multiplication assist` 指令。

因此，**Eigen 3.4** 用户可以在 POWER10 系统上执行优化的线性算法计算。

(BZ#2032423)

### RHEL 9 引进了 **cdrskin** 软件包

RHEL 9 引进了 **cdrskin** 软件包，用于在 CD、DVD 或者 BD 中刻录数据。**cdrskin** 软件包提供了来自 **wodim** 软件包中的 **cdrecord** 可执行文件的替代，该软件包在 RHEL 9 中不可用。

**cdrskin** 软件包包括：

- 在光驱介质上对数据进行清空、格式化和刻录。
- CD 上可以有多个 session。
- 在可写的 DVD+RW、DVD-RW、DVD-RAM、BD-RE 上模拟 ISO-9660 多 session。

**cdrskin** 软件包还提供 **cdrecord** 命令，作为指向 **cdrskin** 二进制文件的符号链接，因此您不必在用户脚本中进行任何更改。有关完整的功能集合，请参阅 **cdrskin(1)** 手册页。

([BZ#2015861](#))

## RHEL 9 发行版本支持 redhat.rhel\_mgmt Ansible 集合

这个版本提供对智能平台管理接口(IPMI)Ansible 模块的支持。**IPMI** 是一组管理接口与基板管理控制器(BMC)设备通信的规范。**IPMI** 模块 - **ipmi\_power** 和 **ipmi\_boot** - 在 **redhat.rhel\_mgmt** Collection 中提供，您可以通过安装 **ansible-collection-redhat-rhel\_mgmt** 软件包来访问它。

([BZ#2023381](#))

## RHEL 9 引进了 util-linux-core 软件包

除了 **util-linux** 软件包外，RHEL 9 还为对安装的软件包的大小有要求的环境（如 buildroot、某些特定容器以及引导镜像）提供了 **util-linux-core** 子软件包。

**util-linux-core** 子软件包是 **util-linux** 实用程序的一个子集，它只包括引导 Linux 系统所必需的程序（如 **mount** 实用程序）。

**util-linux-core** 子软件包不包含任何外部依赖项。例如，因为需要依赖于 PAM，它并不包括 **login** 实用程序。

对于标准用例，如进行安装，需要使用标准的 **util-linux** 软件包。**util-linux** 软件包依赖于 **util-linux-core**，这意味着如果安装了 **util-linux**，则会自动安装 **util-linux-core**。

([BZ#2079313](#))

## 更新的 systemd-udev 将一致的网络设备名称分配给 InfiniBand 接口

RHEL 9 中引入的 **systemd** 软件包的新版本包含更新的 **systemd-udev** 设备管理器。设备管理器将 InfiniBand 接口的默认名称更改为 **systemd-udev** 选择的一致性名称。

您可以按照 [重命名 IPoIB 设备](#) 流程为命名 InfiniBand 接口定义自定义命名规则。

有关命名方案的详情，请查看 **systemd.net-naming-scheme (7)** 手册页。

([BZ#2136937](#))

## 4.6. 基础架构服务

### S-nail 替换 mailx

**s-nail** 邮件处理系统已经替换 **mailx** 实用程序。**s-nail** 实用程序与 **mailx** 兼容，并添加了许多新功能。**mailx** 软件包不再在上游社区维护。

(BZ#1940863)

## Tuned 2.18 现在可用

RHEL 9 带有 Tuned 版本 2.18。2.16 版本的显著变化包括：

- **net** 插件：添加了对 **txqueuelen** 调优功能的支持。
- **disk** 插件：添加了对 NVMe 磁盘调优功能的支持。
- **tuned-gui** 程序错误修复。

(BZ#2003838)

## RHEL 9 提供了 mod\_security\_crs 3.3

RHEL 9 提供了 **mod\_security\_crs** 软件包版本 3.3。重要的程序错误修复和增强包括：

- 引进了 **libinjection**。
- 阻止备份文件的文件名以 `~` 结尾。
- 添加了新的 **LDAP** 注入和 **HTTP** 分割规则。
- 向受限扩展添加了 **.swp**。
- 添加了用于攻击类别的通用模式枚举和分类(CAPEC)标签。
- 添加了检测 **Nuclei**、**WFuzz** 和 **ffuf** 安全漏洞扫描程序的支持。
- 改进了小写的变量（**modsec3** 行为修复）
- 添加了对 Unix RCE 绕过技术（通过未初始化的变量、字符串串联和 globbing patterns）进行检测的支持。
- 删除了过时的规则标签：**WASCTC**、**OWASP\_TOP\_10**、**OWASP\_AppSensor/RE1**，和 **OWASP\_CRS/FOO/BAR.OWASP\_CRS** 和 **attack-type** 仍然包含在 **mod\_security\_crs** 软件包中。
- **crs-setup.conf** 变量 **tx.allowed\_request\_content\_type** 的格式已更改为与其它变量一致。如果变量被覆盖，请参阅 **crs-setup.conf** 文件中的示例。

(BZ#1947962)

## RHEL 9 提供 chrony 4.1

RHEL 9 提供 **chrony** 版本 4.1。与版本 3.5 相比，重要的程序错误修复和增强包括：

- 添加了对 Network Time Security (NTS) 身份验证的支持。如需更多信息，请参阅 [chrony 中的网络时间协议\(NTS\)概述](#)。
- 默认情况下，Authenticated Network Time Protocol(NTP)源可以通过未验证的 NTP 源信任。要恢复到原始的行为，在 **chrony.conf** 文件中添加 **autselectmode ignore** 参数。
- 支持使用 **RIPEDM** 密钥进行身份验证 - **RMD128,RMD160,RMD256,RMD320** 不再可用。
- 对 NTPv4 数据包中长期非标准 MAC 的支持不再可用。如果您使用 **chrony 2.x,non-MD5/SHA1** 密钥，您需要使用 **version 3** 选项配置 **chrony**。

另外，以下内容与 RHEL 8 中的 **chrony** 版本不同：

- **seccomp** 过滤器被默认启用（**-F 2** 在 `/etc/sysconfig/chronyd` 中设置）。**seccomp** 过滤器与 **mailnchange** 指令有冲突。如果在 `/etc/chrony.conf` 中有 **mailnchange** 指令，请从 `/etc/sysconfig/chronyd` 中删除 **-F 2** 设置。

(BZ#1961131)

## 4.7. 安全性

### 系统范围的 **crypto-policies** 现在更为安全

在这个版本中，系统范围的加密策略已被调整，以提供最新的安全默认值：

- 在所有策略中禁用 TLS 1.0、TLS 1.1、DTLS 1.0、RC4、Camellia、DSA、3DES 和 FFcontainerruntime-1024。
- 在 LEGACY 中增加了最小 RSA 密钥大小和最小 Diffie-Hellman 参数的大小。
- 使用 SHA-1 禁用 TLS 和 SSH 算法，在 Hash-based Message Authentication Codes (HMACs) 中使用 SHA-1 除外。

如果您的场景需要启用一些禁用的算法和密码，请使用自定义策略或子策略。

(BZ#1937651)

### RHEL 9 提供 OpenSSL 3.0.1

RHEL 9 提供了上游版本 3.0.1 的 **openssl** 软件包，其中包括很多改进和程序错误修复。最显著的变化包括：

- 添加了新的供应商概念。供应商是算法的集合，您可以为不同的应用程序选择不同的供应商。
- 以以下格式引入了新版本方案：`<major>.<minor>.<patch>`。
- 添加了对证书管理协议（CMP、RFC 4210）、证书请求消息格式(CRMF)和 HTTP 传输(RFC 6712)的支持。
- 引入了支持 GET 和 POST、重定向、普通和 ASN.1- 编码的内容、代理和超时的 HTTP(S)客户端。
- 添加了新的 Key Derivation Function API(EVP\_KDF)和消息身份验证代码 API(EVP\_MAC)。
- 使用 **enable-ktls** 配置选项编译对 Linux Kernel TLS(KTLS)的支持。
- 添加了 CADES-BES 签名验证支持。
- 向 CMS API 添加了 CADES-BES 签名方案和属性支持(RFC 5126)。
- 添加了对新算法的支持，例如：
  - KDF 算法"SINGLE STEP"和"SSH"
  - MAC 算法"GMAC"和"KMAC"。
  - KEM 算法"RSASVE"。
  - 密码算法 "AES-SIV"

- 添加了使用 AES\_GCM 的 AuthEnvelopedData 内容类型结构(RFC 5083)。
- PKCS #12 使用 **PKCS12\_create()** 功能创建的默认算法改为更现代的 PBKDF2 和基于 AES 的算法。
- 添加了一个新的通用 trace API。

(BZ#1990814)

## Openssl 现在包含了供应商

包括在 RHEL 9 中的版本 3.0.1 的 OpenSSL 工具包中增加了一个供应商 (provider) 的概念。供应商是算法的集合，您可以为不同的应用程序选择不同的供应商。openssl 目前包括以下供应商：**base**, **default**, **fips**, **legacy**, 和 **null**。

默认情况下，OpenSSL 载入并激活 **default** 供应商，其中包括常用的算法，如 RSA、DSA、DH、CAMELLIA、SHA-1 和 SHA-2。

当在内核中设置 FIPS 标志时，OpenSSL 会自动加载 FIPS 供应商，并只使用 FIPS 批准的算法。因此，您不必手动将 OpenSSL 切换到 FIPS 模式。

要更改为系统级别上的不同供应商，请编辑 **openssl.cnf** 配置文件。例如，如果您的场景需要使用传统供应商，请取消注释对应的部分。



### 警告

明确激活提供程序会覆盖默认提供程序的隐式激活，并可能使系统能够远程访问，例如 OpenSSH 套件。

有关每个供应商中包含的算法的详情，请查看相关的 man page。例如，**legacy** 供应商的 **OSSL\_PROVIDER-legacy(7)** man page。

(BZ#2010291)

## OpenSSL 随机位生成器现在支持 CPACF

这个 **openssl** 软件包版本引进了对 OpenSSL NIST SP800-90A-based deterministic random bit generator (DRBG) 中的 Cryptographic Functions (CPACF) 的支持。

(BZ#1871147)

**openssl-spkac** 现在可以创建使用 SHA-1 和 SHA-256 签名的 SPKAC 文件。

**openssl-spkac** 工具现在可以创建使用与 MD5 不同的哈希签名的 Netscape 签名的公钥和质询 (SPKAC) 文件。现在，您可以创建和验证使用 SHA-1 和 SHA-256 哈希签名的 SPKAC 文件。

(BZ#1970388)

## RHEL 9 提供 openCryptoki 3.17.0

RHEL 9 带有 **openCryptoki** 版本 3.17.0。与 3.16.0 相比，重要的程序错误修复和增强包括：

- **p11sak** 实用程序添加了列出密钥的新功能。

- **opencryotki** 现在支持：
  - OpenSSL 3.0.
  - 事件通知。
  - ICA 令牌中的软件回退。
- 在启用了硬件加密适配器时，WebSphere Application Server 不会再出现无法启动的问题。

RHEL 9 包括了带有额外补丁的 OpenSSL，它们特定于 RHEL。如果系统处于 Federal Information Processing Standards(FIPS)模式，OpenSSL 会自动加载 FIPS 供应商和基础供应商，并强制应用程序使用 FIPS 供应商。因此，RHEL 9 中的 **openCryptoki** 的行为与上游社区版本中的不同：

- 依赖 OpenSSL 的加密操作（软令牌和 ICA 令牌软件回退）的令牌现在只支持 FIPS 批准的机制（未批准但被列为可用的机制不被支持）。
- **openCryptoki** 支持两种不同的令牌数据格式：旧数据格式（使用非FIPS 批准的算法（如 DES 和 SHA1）以及新的数据格式（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

为了在 RHEL 9 上使用 **openCryptoki**，请在为系统启用 FIPS 模式前将令牌迁移为使用新数据格式。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#1869533)

## GnuTLS 的版本为 3.7.3

在 RHEL 9 中，**gnutls** 软件包在上游版本 3.7.3 中提供。与之前的版本相比，这提供了很多改进和程序错误修复，最重要的是：

- 引入了 FIPS 140-3 明确指示符的 API。
- 强化用于导出 PKCS#12 文件的默认值。
- 修复了早期数据的时间（零往返数据，0-RTT）交换。
- 在签署证书签名请求(CSR)时，**certutil** 工具不再从证书颁发机构(CA)发布点继承 CRL(Certificate Revocation List)分布点。

(BZ#2033220)

## RHEL 9 提供 NSS 3.71

RHEL 9 提供网络安全服务(NSS)库版本 3.71 提供。主要变更包括：

- 对旧的 DBM 数据库格式的支持已被完全删除。NSS 仅支持 RHEL 9 中的 SQLite 数据库格式。

- PKCS #12 加密密码现在使用带有 PBKDF2 和 SHA-256 算法的 AES-128-CBC（而不是 PBE-SHA1-RC2-40 和 PBE-SHA1-2DES）。

(BZ#2008320)

## 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 位改为 1023 位。这意味着 NSS 不再执行以下功能：

- 生成大于 1023 位的 RSA 密钥。
- 使用少于 1023 位的 RSA 密钥进行签名或验证 RSA 签名。
- 使用超过 1023 位的 RSA 密钥加密或解密值。

(BZ#2099438)

## OpenSSH 中的最小 RSA 密钥位长度选项

意外使用短的 RSA 密钥可能会使系统容易受到攻击。有了这个更新，您可以为 OpenSSH 服务器和客户端设置 RSA 密钥最小位长度。要定义最小 RSA 密钥长度，请在 `/etc/ssh/sshd_config` 文件中为 OpenSSH 服务器，在 `/etc/ssh/ssh_config` 文件中为 OpenSSH 客户端使用新的 `RSAMinSize` 选项。

(BZ#2119694)

## OpenSSH 的版本为 8.7p1

RHEL 9 包括 OpenSSH 版本 8.7p1。与 OpenSSH 版本 8.0p1 相比，这个版本提供了很多改进和程序错误修复，它在 RHEL 8.5 中发布，重点包括：

### 新功能

- 支持使用 SFTP 协议作为之前使用 SCP/RCP 协议的替换。SFTP 提供了更可预测的文件名处理方式，远程端的 shell 不再需要 glob(3) 模式扩展。  
默认启用 SFTP 支持。如果您的环境无法使用 SFTP 或存在不兼容的情况，您可以使用 `-O` 标志来强制使用原始 SCP/RCP 协议。
- **LogVerbose** 配置指令，允许按 file/function/line pattern 列表来强制限制最大的调试日志记录。
- 使用新的 **sshd\_config PerSourceMaxStartups** 和 **PerSourceNetBlockSize** 指令，基于客户端地址的速率限制。这比全局 **MaxStartups** 限制提供更精细的控制。
- **Host basedAcceptedAlgorithms** 关键字现在根据签名算法过滤，而不是根据密钥类型进行过滤。
- **sshd** 守护进程中的 **Include sshd\_config** 关键字，允许使用 **glob** 模式包含其他配置文件。
- 支持由 FIDO 联盟指定的 Universal 2nd Factor (U2F) 硬件身份验证器。U2F/FIDO 是广泛用于网站身份验证的双重认证硬件的、成本低的开放标准。在 OpenSSH 中，FIDO 设备由新的公钥类型 **ecdsa-sk** 和 **ed25519-sk** 以及相应的证书类型提供支持。
- 支持需要每个用途的 FIDO 密钥。您可以使用 **ssh-keygen** 和新的 **verify-required** 选项生成这些密钥。当使用 PIN-required 密钥时，系统会提示用户完成签名操作。

- **authorized\_keys** 文件现在支持一个新的 **verify-required** 选项。这个选项需要 FIDO 签名，以断言用户在签名前存在的令牌验证。FIDO 协议支持多种验证方法，OpenSSH 目前仅支持 PIN 验证。
- 添加了对验证 FIDO **Webauthn** 签名的支持。**webauthn** 是在 Web 浏览器中使用 FIDO 密钥的标准。这些签名是普通 FIDO 签名的一种稍有不同格式，因此需要明确支持。

## bug 修复

- **ClientAliveCountMax=0** 关键字的明确语义。现在，它完全禁用连接终止，而不是之前的行为在第一个存活度测试后立即终止连接，而不考虑成功。

## 安全性

- 修复了在 XMSS 密钥类型的私钥解析代码中可利用的整数溢出错误。这个密钥类型仍是实验性的，默认情况下不会编译它。便携式 OpenSSH 中没有面向用户的 `autoconf` 选项以启用它。
- 为 RAM 中剩余的私钥添加保护，以防规范和内存侧通道攻击，如 Spectre、Mltdown 和 Rambleed。当私钥没有与来自随机数据的非对称密钥一起使用时加密它们（当前为 16 KB）。

([BZ#1952957](#))

## 在 OpenSSH 中默认禁用区域设置转发

在小镜像（如容器和虚拟机）中使用 **C.UTF-8** 本地设置，这与传统的 **en\_US.UTF-8** 设置相比，可减少其大小并提高性能。

大多数发行版本都会默认发送本地设置（locale）环境变量，并在服务器端接受它们。但是，这意味着，从使用 **C** 或 **C.UTF-8** 以外的本地设置的客户端 SSH 到没有 **glibc-langpack-en** 或 **glibc-all-langpacks** 软件包的服务器时，用户使用体验会有所降低。特别是，UTF-8 格式的输出无法正常工作，有些工具无法正常工作或频繁发出警告信息。

在这个版本中，OpenSSH 默认关闭本地设置转发功能。即使客户端使用最小安装（仅支持少量的本地设置）连接到服务器，也可以保留本地设置。

([BZ#2002734](#))

## OpenSSH 支持 U2F/FIDO 安全密钥

在以前的版本中，保存在硬件中的 OpenSSH 密钥只通过 PKCS #11 标准支持，这限制了在 SSH 中使用其他安全密钥。支持 U2F/FIDO 安全密钥由上游开发，现在已在 RHEL 9 中实施。这会提高独立于 PKCS #11 接口的 SSH 中的安全密钥的可用性。

([BZ#1821501](#))

## libreswan 的版本为 4.6

在 RHEL 9 中提供了上游的 Libreswan 版本 4.6。这个版本提供了很多程序错误修复和增强，特别是对使用 IKEv2（Internet Key Exchange version 2）的被标签的 IPsec 的改进。

([BZ#2017355](#))

## libreswan 默认不接受 IKEv1 软件包

因为 IKEv2（Key Exchange v2）现已广泛部署，所以 Libreswan 不再支持 IKEv1 数据包。IKEv2 提供更安全的环境，并对安全攻击更具有防御性。如果您需要使用 IKEv1，可以在 `/etc/ipsec.conf` 配置文件中添加 **ikev1-policy=accept** 选项来启用它。

([BZ#2039877](#))

## RHEL 9 提供 stunnel 5.62

RHEL 9 带有 **stunnel** 软件包版本 5.62。重要的程序错误修复和增强包括：

- 在 FIPS 模式的系统中，**stunnel** 现在总是使用 FIPS 模式。
- **NO\_TLSv1.1**, **NO\_TLSv1.2**, 和 **NO\_TLSv1.3** 选项现在已分别被重新命名为 **NO\_TLSv1\_1**, **NO\_TLSv1\_2**, 和 **NO\_TLSv1\_3**。
- 新的服务级 **sessionResume** 选项启用和禁用会话恢复。
- 现在，在使用 **protocol** 选项的 **stunnel** 客户端中支持 LDAP。
- 现在提供了 Bash 补全脚本。

([BZ#2039299](#))

## RHEL 9 提供 nettle 3.7.3

RHEL 9 提供了 **nettle** 软件包 3.7.3 版本，提供多个程序错误修复和增强。主要变化如下：

- 支持新的算法和模式，例如 **Ed448**、**SHAKE256**、**AES-XTS**、**SIV-CMAC**。
- 为现有算法添加特定于架构的优化。

([BZ#1986712](#))

## RHEL 9 提供 p11-kit 0.24

RHEL 9 提供带有 0.24 版本的 **p11-kit** 软件包。这个版本提供多个程序错误修复和增强。值得注意的是，存储不受信任的证书颁发机构的子目录已重命名为 **blocklist**。

([BZ#1966680](#))

## cyrus-sasl 现在使用 GDBM 而不是 Berkeley DB

**cyrus-sasl** 软件包构建时没有 **libdb** 依赖项，**sasldb** 插件使用 GDBM 数据库格式而不是 Berkeley DB。要迁移以旧 Berkeley DB 格式存储的现有简单身份验证和安全层(SASL)数据库，请使用 **cyrusbdb2current** 工具，语法如下：

```
cyrusbdb2current <sasldb_path> <new_path>
```

([BZ#1947971](#))

## RHEL 9 中的 SELinux 策略与当前内核保持同步

SELinux 策略包括新的权限、类和功能，也属于内核的一部分。因此，SELinux 可以使用内核提供的完整潜力。具体来说，SELinux 对授予权限提供更好的粒度，其具有后续的安全优势。这也启用了使用 MLS SELinux 策略运行的系统，因为 MLS 策略会在系统包含策略中没有权限时阻止一些系统启动。

([BZ#1941810](#), [BZ#1954145](#))

## 默认 SELinux 策略不允许使用文本重新定位库的命令

**selinuxuser\_execmod** 布尔值现在默认关闭，以提高已安装系统的安全性。因此，SELinux 用户无法使用需要文本重新定位的库输入命令，除非库文件具有 **textrel\_shlib\_t** 标签。

([BZ#2055822](#))

## OpenSCAP 在版本 1.3.6 中提供

RHEL 9 在 1.3.6 版本中包括 OpenSCAP，它提供程序错误修复和增强，最重要的是：

- 您可以使用 **--local-files** 选项提供远程 SCAP 源数据流组件的本地副本，而不是在扫描过程中下载它们
- OpenSCAP 接受多个 **--rule** 参数，以便在命令行上选择多个规则。
- 您可以使用 **--skip-rule** 选项跳过某些规则的评估。
- 您可以使用 **OSCAP\_PROBE\_MEMORY\_USAGE\_RATIO** 环境变量来限制 OpenSCAP 探测所使用的内存。
- OpenSCAP 现在支持 OSBuild Blueprint 作为补救类型。

([BZ#2041782](#))

## OSCAP Anaconda 附加组件现在支持新的附加组件名称

在这个改进中，您可以使用新的 **com\_redhat\_oscaped** 附加组件名称，而不是在 **OSCAP Anaconda 附加组件** 的 Kickstart 文件中旧的 **org\_fedora\_oscaped** 附加组件名称。例如，Kickstart 部分的结构如下：

```
%addon com_redhat_oscaped
  content-type = scap-security-guide
%end
```

OSCAP Anaconda 附加组件目前与旧附加组件名称一致，但在以后的主 RHEL 版本中将删除对传统附加名称的支持。

([BZ#1893753](#))

## CVE OVAL feed 现在会被压缩

在这个版本中，红帽以压缩的形式提供 CVE OVAL feed。它们不再作为 XML 文件提供，而是使用 **bzip2** 格式。RHEL9 源的位置也已更新，以反映这个变化。请注意，第三方 SCAP 扫描程序可能会遇到扫描规则以使用压缩源的问题，因为引用压缩内容没有标准化。

([BZ#2028435](#))

## 提供的 SCAP 安全指南版本为 0.1.60

RHEL 9 包括 **scap-security-guide** 0.1.60 版本的软件包。此版本提供程序错误修正和增强，最重要的是：

- 规则强化 PAM 堆栈现在使用 **authselect** 作为配置工具。
- SCAP 安全指南现在为 STIG 配置集提供了一个 delta 定制文件。此定制文件定义了一个配置集，它代表了 DISA 的自动化 STIG 和 SSG 自动化内容之间的区别。

([BZ#2014561](#))

## RHEL 9.0 支持的 SCAP 安全指南配置集

使用 RHEL 9.0 中包含的 SCAP 安全指南合规性配置集，您可以从发出的组织将系统强化到建议。因此，您可以使用关联的补救和 SCAP 配置集根据所需的强化级别配置和自动化 RHEL 9 系统合规性。

配置文件名称	配置文件 ID	策略版本
法国信息系统安全局(ANSSI)BP-028 增强级	<b>xccdf_org.ssgproject.content_profile_anssi_bp28_enhanced</b>	1.2
法国信息系统安全全部(ANSSI)BP-028 高级别	<b>xccdf_org.ssgproject.content_profile_anssi_bp28_high</b>	1.2
法国信息系统安全局(ANSSI)BP-028 中级	<b>xccdf_org.ssgproject.content_profile_anssi_bp28_intermediary</b>	1.2
法国信息系统安全局(ANSSI)BP-028 最低级	<b>xccdf_org.ssgproject.content_profile_anssi_bp28_minimal</b>	1.2
[DRAFT] CIS Red Hat Enterprise Linux 9 基准 (第 2 级 - 服务器)	<b>xccdf_org.ssgproject.content_profile_cis</b>	草案 <sup>[a]</sup>
[DRAFT] CIS Red Hat Enterprise Linux 9 基准 (第 1 级 - 服务器)	<b>xccdf_org.ssgproject.content_profile_cis_server_l1</b>	DRAFT <sup>[a]</sup>
[DRAFT] CIS Red Hat Enterprise Linux 9 Benchmark for Level 1 - Workstation	<b>xccdf_org.ssgproject.content_profile_cis_workstation_l1</b>	DRAFT <sup>[a]</sup>
[DRAFT] CIS Red Hat Enterprise Linux 9 Benchmark for Level 2 - Workstation	<b>xccdf_org.ssgproject.content_profile_cis_workstation_l2</b>	DRAFT <sup>[a]</sup>
[DRAFT] 在非保障信息系统和机构中未分类的信息(NIST 800-171)	<b>xccdf_org.ssgproject.content_profile_cui</b>	r2
澳大利亚网络安全中心(ACSC)要点 8	<b>xccdf_org.ssgproject.content_profile_e8</b>	未版本化
健康保险可移植性和责任法案(HIPAA)	<b>xccdf_org.ssgproject.content_profile_hipaa</b>	未版本化
澳大利亚网络安全中心(ACSC)ISM 官方	<b>xccdf_org.ssgproject.content_profile_ism_o</b>	未版本化
[DRAFT] 常规目的操作系统的保护配置文件	<b>xccdf_org.ssgproject.content_profile_ospp</b>	4.2.1
适用于 Red Hat Enterprise Linux 9 的 PCI-DSS v3.2.1 控制基本行	<b>xccdf_org.ssgproject.content_profile_pci-dss</b>	3.2.1

配置文件名称	配置文件 ID	策略版本
[DRAFT] DISA STIG for Red Hat Enterprise Linux 9	<b>xccdf_org.ssgproject.content_profile_stig</b>	草案 <sup>[b]</sup>
[DRAFT] DISA STIG with GUI for Red Hat Enterprise Linux 9	<b>xccdf_org.ssgproject.content_profile_stig_gui</b>	DRAFT <sup>[b]</sup>
<p>[a] CIS 尚未发布 RHEL 9 的官方基准</p> <p>[b] DISA 尚未发布 RHEL 9 的官方基准</p>		



### 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#2045341](#), [BZ#2045349](#), [BZ#2045361](#), [BZ#2045368](#), [BZ#2045374](#), [BZ#2045381](#), [BZ#2045386](#), [BZ#2045393](#), [BZ#2045403](#))

## RHEL 9 提供 **fapolicyd** 1.1

RHEL 9 提供 **fapolicyd** 软件包版本 1.1。最显著的改进包括：

- 包含允许和拒绝执行规则的文件的 `/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。详情请查看新的 **fagenrules(8)** 手册页。
- 除了用于将 RPM 数据库之外的文件标记为可信的 `/etc/fapolicyd/fapolicyd.trust` 文件外，您现在可以使用新的 `/etc/fapolicyd/trust.d` 目录，它支持将可信文件列表分成多个文件。您还可以通过使用 **fapolicyd-cli -f** 子命令及 **--trust-file** 指令，为这些文件添加一个文件项。如需更多信息，请参阅 **fapolicyd-cli(1)** 和 **fapolicyd.trust(13)** 手册页。
- **fapolicyd** 信任数据库现在支持文件名中的空格。
- 现在，当将文件添加到信任数据库时，**fapolicyd** 会保存可执行文件的正确路径。

([BZ#2032408](#))

## rsyslog 包含用于高性能操作和 CEF 的 **mmfields** 模块

rsyslog 现在包含 **rsyslog-mmfields** 子软件包，它提供 **mmfields** 模块。这是使用 property replacer 字段提取的替代方式，但与属性替换器不同，所有字段都会一次性提取并存储在结构化的数据部分中。因此，您可以使用 **mmfields** 特别用于处理基于字段的日志格式，如 Common Event Format(CEF)，并且如果您需要大量字段或重复使用特定字段，可以使用 **mmfields**。在这些情况下，**mmfields** 的性能比现有 Rsyslog 功能要提高性能。

([BZ#2027971](#))

## 单独 **rsyslog-logrotate** 软件包中包含的 **logrotate**

**logrotate** 配置与主 **rsyslog** 软件包分隔到新的 **rsyslog-logrotate** 软件包中。这在某些最小环境中非常有用，例如不需要日志轮转的情况，以防止安装不必要的依赖项。

([BZ#1992155](#))

### sudo 支持 Python 插件

通过 **sudo** 程序版本 1.9（包括在 RHEL 9 中），您可以在 Python 中编写 **sudo** 插件。这样可以更轻松地增强 **sudo** 以更精确地适合特定的场景。

如需更多信息，请参阅 **sudo\_plugin\_python(8)** man page。

([BZ#1981278](#))

### libseccomp 的版本为 2.5.2

RHEL 9.0 提供了上游版本 2.5.2 **libseccomp** 软件包。与之前的版本相比，这个版本提供了很多程序错误修复和增强，最重要的是：

- 将 Linux 的 syscall 表更新为 **v5.14-rc7** 版本。
- 在 Python 绑定中添加 **get\_notify\_fd()** 函数来获取通知文件描述符。
- 将所有架构的多路系统调用处理整合到一个位置。
- 为 PowerPC(PPC)和 MIPS 架构添加了多路系统调用支持。
- 更改内核中的 **SECCOMP\_IOCTL\_NOTIF\_ID\_VALID** 操作的含义。
- **libseccomp** 文件描述符通知逻辑已被修改来支持内核之前和现在对 **SECCOMP\_IOCTL\_NOTIF\_ID\_VALID** 的使用。
- 修复了一个错误，**seccomp\_load()** 只能调用一次。
- 对通知 **fd** 的处理进行了更改，在过滤有一个 **\_NOTIFY** 操作时只请求一个通知 **fd**。
- 向 **seccomp\_add\_rule(3)** manpage 中添加了与 **SCMP\_ACT\_NOTIFY** 相关的内容。
- 阐明维护器的 GPG 密钥。

([BZ#2019887](#))

### Clevis 现在支持 SHA-256

在这个版本中，Clevis 框架支持 **SHA-256** 算法作为 JSON Web 密钥(JWK)指纹的默认哈希值，如 **RFC 7638** 的建议。因为旧的 thumbprints(SHA-1)仍被支持，所以您仍可以解密之前加密的数据。

([BZ#1956760](#))

## 4.8. 网络

### diag 模块现在包括在内核中

现在，内核镜像中会包括 **diag** 模块。在这个版本中，当使用 **ss** 命令时，不再需要动态加载 **diag** 模块。无论内核模块中的客户策略是什么，都可更好地调试网络问题。内核中包含的模块：

```
CONFIG_INET_DIAG
CONFIG_INET_RAW_DIAG
```

```
CONFIG_INET_TCP_DIAG
CONFIG_INET_UDP_DIAG
CONFIG_INET_MPTCP_DIAG
CONFIG_NETLINK_DIAG
CONFIG_PACKET_DIAG
CONFIG_UNIX_DIAG
```

(BZ#1948340)

## 新的内核和 IPv4 相关的网络 `sysctl` 内核参数

与之前的 RHEL 版本相比，RHEL 9.0 内核提供以下新的内核和 IPv4 网络 `sysctl` 参数：

- `net.core.devconf_inherit_init_net`
- `net.core.gro_normal_batch`
- `net.core.high_order_alloc_disable`
- `net.core.netdev_unregister_timeout_secs`
- `net.ipv4.fib_multipath_hash_fields`
- `net.ipv4.fib_notify_on_flag_change`
- `net.ipv4.fib_sync_mem`
- `net.ipv4.icmp_echo_enable_probe`
- `net.ipv4.ip_autobind_reuse`
- `net.ipv4.nexthop_compat_mode`
- `net.ipv4.raw_l3mdev_accept`
- `net.ipv4.tcp_comp_sack_slack_ns`
- `net.ipv4.tcp_migrate_req`
- `net.ipv4.tcp_mtu_probe_floor`
- `net.ipv4.tcp_no_ssthresh_metrics_save`
- `net.ipv4.tcp_reflect_tos`

有关这些参数的详情，请安装 `kernel-doc` 软件包并查看以下文件：

- `/usr/share/doc/kernel-doc-<version>/Documentation/admin-guide/sysctl/net.rst`
- `/usr/share/doc/kernel-doc-<version>/Documentation/networking/ip-sysctl.rst`

(BZ#2068532)

## 更改了在跨区传输数据包时 `firewalld` 的行为

在基于区域的防火墙中，数据包只输入一个区。隐式数据包传输是概念违规，并允许意外流量或服务。在 Red Hat Enterprise Linux 9 中，`firewalld` 服务不再允许两个不同区间的隐式数据包传输。

有关此更改的更多信息，请参阅 [Changed behavior in firewalld when transmitting packets between zones](#) 知识库文档。

([BZ#202921](#))

## 默认启用区域内转发

**firewalld** 区域内转发功能允许 firewalld 区域内接口间或源间的转发流量。从 RHEL 9.0 开始，这个功能已被默认启用。使用 **firewall-cmd** 工具的 **--add-forward** 选项，为特定区域启用区域内转发。**firewall-cmd --list-all** 命令显示是否对区域启用或禁用了区域内转发：

```
# firewall-cmd --list-all
public (active)
...
forward: no
```

([BZ#2089193](#))

## 使 Nmstate 更包含

红帽承诺使用适当的语言。因此，**nmstate** API 中的术语 **slave** 已被术语 **port** 取代。

([BZ#1969941](#))

## NetworkManager 支持 IBM Z 上的 rd.znet\_ifname 内核选项中设置的接口名称

在这个版本中，在 IBM Z 平台上，NetworkManager 现在会在从网络安装或引导 Red Hat Enterprise Linux 时解释 **rd.znet** 和 **rd.znet\_ifname** 内核命令行选项。因此，可以指定由子频道标识的网络接口名称，而不是默认接口。

([BZ#1980387](#))

## hostapd 软件包已添加到 RHEL 9.0

在这个版本中，RHEL 提供了 **hostapd** 软件包。但是，红帽支持 **hostapd** 只用来在以太网网络中将 RHEL 主机设置为 802.1X 验证器。对 Wi-Fi 网络中的 Wi-Fi 接入点或验证器等其他场景不支持。

有关使用 FreeRADIUS 后端将 RHEL 配置为 802.1X authenticator 的详情，请参阅[为使用 FreeRADIUS 后端的主机 apd 设置 802.1x 网络身份验证服务](#)。

([BZ#2019830](#))

## 提供版本 1.18.2 的 ModemManager

RHEL 9.0 在上游版本 1.18.2 中提供 **ModemManager** 软件包。与之前的版本相比，这个版本包括程序错误修正和增强，最重要的是：

- 改进了对具有 5G 功能的设备的功能和模式处理
- 其他设备支持

([BZ#1996716](#))

## NetworkManager 允许更改绑定端口的 queue\_id

绑定中的 NetworkManager 端口现在支持 **queue\_id** 参数。假设 **eth1** 是绑定接口的端口，您可以使用以下内容为绑定端口启用 **queue\_id**：

```
# nmcli connection modify eth1 bond-port.queue-id 1
# nmcli connection up eth1
```

任何需要使用这个选项的网络接口都应配置为多个调用，直到为所有接口设置正确的优先级。如需更多信息，请参阅 **kernel-doc** 软件包提供的 `/usr/share/docs/kernel-doc-<version>/Documentation/networking/bonding.rst` 文件。

([BZ#1949127](#))

### 支持使用最新 NetworkManager 配置 blackhole, prohibit 和 unreachable 路由类型

内核除了支持常见的单播、广播和本地路由类型，还支持其他多种路由类型。另外，用户现在可以配置 NetworkManager 连接配置集中的 **blackhole**, **prohibit** 和 **unreachable** 的静态路由类型。当激活配置集时，NetworkManager 会添加一个配置集。

([BZ#2060013](#))

### RoCE Express Adapters 现在使用改进的接口命名方案

有了这个增强，RDMA over Converged Ethernet(RoCE)Express 适配器使用可预测的接口命名方案和 z-system 连接器上的外设通信接口(zPCI)。在这个命名方案中，RHEL 使用用户标识符(UID)或功能标识符(FID)来生成唯一名称。如果没有可用的唯一 UID，RHEL 使用 FID 来设置命名方案。

([BZ#2091653](#))

## 4.9. 内核

### RHEL 9.0 中的内核版本

Red Hat Enterprise Linux 9.0 带有内核版本 5.14.0-70。

([BZ#2077836](#))

### 默认情况下，Red Hat 在所有 RHEL 版本中只为特权用户启用 eBPF

扩展 Berkeley Packet Filter(eBPF)是一个复杂的技术，其允许用户执行 Linux 内核中的自定义代码。由于它的性质，eBPF 代码需要通过验证器和其他安全机制。存在常见的漏洞和暴露(CVE)实例，其中，这个代码中的 bug 可能会被误用于未经授权的操作。为了减少这种风险，红帽默认在所有 RHEL 版本中只为特权用户启用 eBPF。可以使用 `kernel.command-line` 参数 **unprivileged\_bpf\_disabled=0** 为非特权用户启用 eBPF。

但请注意：

- 应用 **unprivileged\_bpf\_disabled=0** 可使您的内核失去红帽的支持，并使您的系统面临安全风险。
- 红帽敦促您对待具有 **CAP\_BPF** 功能的进程，就如同该功能等同于 **CAP\_SYS\_ADMIN**。
- 设置 **unprivileged\_bpf\_disabled=0** 不足以让非特权用户执行许多 BPF 程序，因为大多数 BPF 程序类型的加载需要额外的功能（通常为 **CAP\_SYS\_ADMIN** 或 **CAP\_PERFMON**）。

有关如何应用内核命令行参数的详情，请参考 [配置内核命令行参数](#)。

([BZ#2091643](#))

### 红帽只为次版本保护内核符号

红帽保证，在您使用受保护的内核符号编译内核模块时，内核模块将继续在延长更新支持(EUS)版本中的所有更新中载入，。RHEL 9 的次版本之间没有内核应用程序二进制接口(ABI)保证。

([BZ#2059183](#))

## RHEL 9 Beta 内核使用受信任的 SecureBoot 证书签名

在以前的版本中，RHEL Beta 版本需要用户使用 Machine Owner Key(MOK)工具注册独立的 Beta 公钥。从 RHEL 9 Beta 版开始，内核与受信任的 SecureBoot 证书签名，因此用户不再需要注册一个单独的 Beta 公钥以便在启用了 UEFI 安全引导的系统上使用 beta 版本。

([BZ#2002499](#))

## RHEL 9 中默认启用 cgroup-v2

控制组版本 2(**cgroup-v2**)功能实施单一层次结构模型，以简化控制组的管理。此外，它确保一个进程一次只能是一个控制组的成员。与 **systemd** 的深度集成提高了在 RHEL 系统上配置资源控制时的最终用户体验。

新功能的开发主要针对 **cgroup-v2**，其具有 **cgroup-v1** 缺少的一些功能。类似地，**cgroup-v1** 还包含 **cgroup-v2** 中缺少的一些传统功能。此外，控制接口也不同。因此，直接依赖 **cgroup-v1** 的第三方软件在 **cgroup-v2** 环境中可能无法正常运行。

要使用 **cgroup-v1**，您需要在内核命令行中添加以下参数：

```
systemd.unified_cgroup_hierarchy=0
systemd.legacy_systemd_cgroup_controller
```



### 注意

内核中完全启用了 **cgroup-v1** 和 **cgroup-v2**。从内核的角度来看，没有默认的控制组版本，并且由 **systemd** 决定在启动时挂载。

([BZ#1953515](#))

## 可能会影响第三方内核模块的内核更改

Linux 分发自 5.9 之前内核版本，支持导出 GPL 功能，作为非 GPL 功能。因此，用户可以通过 **shim** 机制将专有功能链接到 GPL 内核功能。在这个版本中，RHEL 内核融合了上游更改，这些更改提高了 RHEL 通过重新调整 **shim** 来强制实施 GPL 的能力。



### 重要

合作伙伴和独立软件供应商(ISV)应利用早期版本的 RHEL 9 测试他们的内核模块，以确保其符合 GPL。

([BZ#1960556](#))

## 64 位 ARM 架构在 RHEL 9 中有 4 KB 页面大小

红帽已经为 Red Hat Enterprise Linux 9 的 64 位 ARM 架构选择了 4 KB 页面大小。这个大小对大多数基于 ARM 的系统的工作负载和内存数量良好。要有效地使用大型页面大小，请使用巨页选项来处理大量内存或带有大型数据集的工作负载。

有关巨页的更多信息，请参阅[监控和管理系统状态和性能](#)。

(BZ#1978382)

## strace 程序现在可以正确地显示不匹配的 SELinux 上下文

**strace** 的现有 `--secontext` 选项已使用 `mismatch` 参数进行了扩展。这个参数可让您只打印预期的上下文以及实际不匹配的上下文。输出用双感叹号(!!)分隔，第一个是实际上下文，然后是预期上下文。在下面的示例中，`full,mismatch` 参数打印预期的完整上下文以及实际的上下文，因为上下文的用户部分不匹配。但是，在使用单独的 `mismatch` 时，它只检查上下文的类型部分。预期的上下文不会打印，因为上下文的类型部分匹配。

```
[...]
$ strace --secontext=full,mismatch -e statx stat /home/user/file
statx(AT_FDCWD, "/home/user/file"
[system_u:object_r:user_home_t:s0!!unconfined_u:object_r:user_home_t:s0], ...

$ strace --secontext=mismatch -e statx stat /home/user/file
statx(AT_FDCWD, "/home/user/file" [user_home_t:s0], ...
```

SELinux 上下文不匹配通常会导致与 SELinux 相关的访问控制问题。系统调用 traces 中打印的不匹配可显著加快 SELinux 上下文正确性的检查。系统调用 traces 也可以解释有关访问控制检查的特定内核行为。

(BZ#2038965)

## perf-top 现在可以按特定列排序

在这个版本中，对 **perf-top** 系统性能分析工具的更新，您可以根据任意事件列对示例进行排序。在以前的版本中，当一个组中的多个事件被抽样时，事件按照第一列进行排序。要排序示例，请使用 `--group-sort-idx` 命令行选项，然后按数字键，按匹配的数据列列表进行排序。请注意列编号从 **0** 开始。

(BZ#1851933)

## 新软件包：jigawatts

checkpoint/Restore In Userspace(CRIU)是一个 Linux 工具，它允许检查识别和恢复进程。**jigawatts** 软件包包含一个 Java 库，旨在提高 Java 应用程序可用性的 CRIU 机制。

(BZ#1972029)

## trace-cmd reset 命令有新的行为

在以前的版本中，`trace-cmd reset` 命令会将 `tracing_on` 配置重置为 0。`trace-cmd reset` 的新行为是将 `tracing_on` 重置为默认值 1。

(BZ#1933980)

## RHEL 9 支持扩展 Berkeley Packet 过滤器

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

eBPF 字节码首先加载到内核。然后，使用即时编译对原机器代码进行验证并转换为原机器代码。最后，虚拟机会执行代码。

红帽提供大量使用 eBPF 虚拟机的组件。在 RHEL 9 中，这些组件包括：

- **BPF Compiler Collection (BCC)** 软件包，提供用于使用 eBPF 的 I/O 分析、联网和监控 Linux 操作系统的工具。

- BCC 库，它允许开发与 BCC 工具软件包中相似的工具。
- **bpftrace** 追踪语言。
- **libbpf** 软件包，对于 **bpf** 的开发以及与 **bpf** 相关的应用程序（如 **bpftrace**）至关重要。
  - **libbpf** 库中的 **XDP** 和 **AF\_XDP** API 部分不被支持，并可能在以后的发行版本中删除。
- **eBPF for Traffic Control (tc)** 功能，可在内核网络数据路径中启用可编程数据包处理。
- **eXpress Data Path (XDP)** 功能在内核网络堆栈处理它们前提供对接收的数据包的访问。红帽仅在通过 **libxdp** 库使用时才支持 **XDP**。
- **xdp-tools** 软件包，其中包含 **XDP** 功能的用户空间支持工具，在 AMD64 和 Intel64 CPU 构架中被支持。**xdp-tools** 软件包包括：
  - **libxdp** 库。
  - 载入 **XDP** 程序的 **xdp-loader** 工具。
  - 用于数据包过滤的 **xdp-filter** 示例程序。
  - 用于从启用了 **XDP** 的网络接口捕获数据包的 **xdpdump** 工具。目前仅在 AMD64 和 Intel64 CPU 构架中支持 **xdpdump** 工具。它可用于其他架构，但为技术预览。
- 用于连接 **eXpress Data Path (XDP)** 路径到用户空间的 **AF\_XDP** 套接字

([BZ#2070506](#))

## RHEL 9 提供 **crash** 工具版本 8.0.0

RHEL 9 提供 **crash** 工具版本 8.0.0。程序错误修复和显著改进包括：

- 在 **add-symbol-file** 命令中添加新的 **offset** 参数。这一支持有助于将 **kaslr\_offset** 设置为 **gdb**。
- 将 **gdb-7.6** 升级到 **gdb-10.2**。

([BZ#1896647](#))

## **makedumpfile** 现在支持改进的 **zstd** 压缩功能

在这个版本中，**makedumpfile** 包含了 **Zstandard(zstd)** 压缩功能，它提供了高压缩率。这种改进有助于在大型内存系统中特别使用。

与之前的压缩率相比，**zstd** 压缩功能在 **vmcore** 转储大小和压缩时间之间有很好的平衡。现在，改进的压缩机制会使用可接受的压缩时间创建一个较小的 **vmcore** 文件。

请注意，良好的压缩率还取决于系统的使用方式以及 RAM 中存储的数据类型。

([BZ#1988894](#))

## Intel Xeon 可扩展服务器处理器上启用了 **numatop**

**numatop** 是一个跟踪和分析 NUMA 系统上运行的进程和线程的行为，并显示可以识别 NUMA 相关性能瓶颈的指标。

**numatop** 使用 Intel 性能计数器抽样技术，并将性能数据与 Linux 系统 **runtime** 相关联，以在生产系统中提供分析。

(BZ#1874125)

### kexec\_file\_load 已添加为 RHEL 9 的默认选项

这个更新为 64 位 ARM 架构添加了 **kexec\_file\_load** 系统调用。它为 **kdump** 提供内核化 **kexec** 装载程序。在以前的版本中，当启用了安全引导选项时，内核会阻止加载未签名的内核镜像。**kdump** 机制首先会尝试检测是否启用了安全引导，然后选择要运行的引导接口。因此，在启用了安全引导并指定了 **kexec\_file\_load()** 时，未签名的内核无法加载。

在这个版本中解决了这个问题，未签名的内核在上述场景中可以正常工作。

(BZ#1895232)

### makedumpfile 现在包含改进的选项，以获得预计 vmcore 大小

在这个版本中，**makedumpfile** 程序包括以下选项，可帮助打印当前运行的内核转储大小的估算：

- **--dry-run** 执行其他选项指定的所有操作，但不会写入输出文件。
- **--show-stats** 会显示报告信息。这是为提供给 **--message** 选项的级别中启用位 4 的替代选择。

以下示例显示了 **--dry-run** 和 **--show-stats** 用法：

```
$ makedumpfile --dry-run --show-stats -l --message-level 7 -d 31 /proc/kcore dump.dummy
```

请注意，转储文件大小可能会根据 **panic** 的系统状态而有所不同，选项提供的估算可能与实际状态不同。

(BZ#1958452)

### kexec-tools 软件包现在支持 RHEL 9 的默认 crashkernel 内存保留值

**kexec-tools** 软件包现在维护默认的 **crashkernel** 内存保留值。**kdump** 服务使用默认值为每个内核保留 **crashkernel** 内存。通过这个实现，当系统的可用内存少于 4GB 时，对 **kdump** 的内存分配有所改进。

查询默认的 **crashkernel** 值：

```
$ kdumpctl get-default-crashkernel
```

如果系统上默认 **crashkernel** 值保留的内存不够，请提高 **crashkernel** 参数。

请注意，RHEL 9 及更新的版本中不再支持引导命令行中的 **crashkernel=auto** 选项。

如需更多信息，请参阅 `/usr/share/doc/kexec-tools/crashkernel-howto.txt` 文件。

(BZ#2034490)

### RHEL 9 支持内核调度

借助内核调度功能，用户可以防止不应相互信任的任务共享相同的 CPU 内核。类似地，用户可以定义可共享 CPU 内核的任务组。

可以指定这些组：

- 通过减少一些跨严重多线程(SMT)攻击来提高安全性
- 隔离需要整个内核的任务。例如，对于实时环境中的任务，或依赖特定处理器功能的任务，如单指令、多数据(SID)处理

如需更多信息，请参阅 [Core Scheduling](#)。

(JIRA:RHELPLAN-100497)

### 使用非限制 iommu 模式作为默认在 64 位 ARM 架构上提高了性能

在这个版本中，64 位 ARM 架构默认使用 lazy 直接内存访问(DMA)域进行系统内存管理单元(SMMU)。虽然这可以显著提高性能，但它在一个地址被取消映射和在 SMMU 上进行 TLB (Translation Lookaside Buffer) 刷新之间会有一个时间窗口。在以前的版本中，64 位 ARM 架构将严格的 DMA 域配置为默认值，这会因为 4KB 页大小而导致性能下降。

如果您需要使用严格的 DMA 域模式，请使用内核命令行指定 `iommu.strict=1` 模式。请注意，使用严格的 DMA 域可能会导致 64 位 ARM 架构的性能下降。

(BZ#2050415)

### kernel-rt 源树已更新至 RHEL 9.0 树

**kernel-rt** 源已更新为使用最新的 Red Hat Enterprise Linux 内核源树。实时补丁集也更新至最新的上游版本 v5.15-rt19。这些更新提供了很多程序错误修正和增强。

(BZ#2002474)

### 在 hv\_24x7 和 hv\_gpci PMU 中支持 CPU 热插拔

在这个版本中，PMU 计数器可以正确地响应 CPU 的热插操作。因此，如果 `hv_gpci` 事件计数器在禁用的 CPU 上运行，则计数会重定向到另一个 CPU。

(BZ#1844416)

### POWERPC hv\_24x7 事件的指标现在可用

POWERPC **hv\_24x7** 嵌套事件的指标现在可用于 `perf`。通过聚合多个事件，这些指标可以更好地了解从 `perf` 计数器获取的值，以及 CPU 如何处理工作负载。

(BZ#1780258)

### RHEL 9 中引入了 IRDMA 驱动程序

IRDMA 驱动程序在支持 RDMA 的 Intel® 网络设备上启用 RDMA 功能。这个驱动程序支持的设备有：

- Intel® Ethernet Controller E810
- Intel® Ethernet Network Adapter X722

RHEL 9 为 X722 Internet Wide-area RDMA 协议(iWARP)设备提供更新的 Intel® 以太网协议驱动程序 (IRDMA)设备。RHEL 9 还引进了新的 E810 设备，它通过融合以太网支持 iWARP 和 RDMA(RoCEv2)。IRDMA 模块取代了 X722 的传统 i40iw 模块，并扩展为 i40iw 定义的应用程序二进制接口(ABI)。这个变化与旧的 X722 RDMA-Core 供应商(libi40iw)向后兼容。

- X722 设备只支持 iWARP 和一组有限的配置参数。
- E810 设备支持以下一组 RDMA 和拥塞管理功能：
  - iWARP 和 RoCEv2 RDMA 传输
  - 优先流控制(PFC)
  - 显式拥塞通知(ECN)

(BZ#1874195)

## 内核 bonding 模块的新参数：lACP\_active

RHEL 9 引入了 **bonding** 内核模块的 **lACP\_active** 参数。这个参数指定是否以指定间隔发送链路聚合控制协议数据单元(LACPDU)帧。这些选项包括：

- **on** (默认) - 允许发送 LACPDU 帧以及配置的 **lACP\_rate** 参数
- **off** - LACPDU 帧以 "speak when spoken to" 模式执行

请注意，当初始化或未绑定端口时，LACPDU 状态帧仍然被发送。

(BZ#1951951)

## 4.10. 引导装载程序

### 引导装载程序配置文件跨 CPU 架构统一

GRUB 引导装载程序的配置文件现在保存在所有支持的 CPU 架构的 **/boot/grub2/** 目录中。GRUB 之前在 UEFI 系统上用作主配置文件的 **/boot/efi/EFI/redhat/grub.cfg** 文件现在只加载 **/boot/grub2/grub.cfg** 文件。

此更改简化了 GRUB 配置文件的布局，改进了用户体验，并提供以下显著优点：

- 您可以使用 EFI 或旧 BIOS 引导相同的安装。
- 您可以将相同的文档和命令用于所有架构。
- GRUB 配置工具更加强大，因为它们不再依赖于符号链接，而且不必处理平台特定的情况。
- GRUB 配置文件的使用与 CoreOS Assembler(COSA)和 OSBuild 生成的镜像一致。
- GRUB 配置文件的使用与其他 Linux 发行版一致。

(JIRA:RHELPLAN-101246)

## 4.11. 文件系统和存储

### Samba 工具中的选项已被重命名和删除，以获得一致的用户体验

Samba 工具已被改进，来提供一致的命令行界面。这些改进包括重命名和删除的选项。因此，为了避免更新后出现问题，请查看使用 Samba 工具脚本文本，并在需要时更新它们。

Samba 4.15 在 Samba 工具中引进了以下更改：

- 在以前的版本中，Samba 命令行工具会悄悄忽略未知选项。为防止意外行为，工具现在一致拒绝未知选项。
- 现在，几个命令行选项有一个对应的 **smb.conf** 变量来控制它们的默认值。请参阅工具的手册页来识别命令行选项是否有 **smb.conf** 变量名。
- 默认情况下，Samba 工具现在记录到标准错误(**stderr**)。使用 **--debug-stdout** 选项更改此行为。
- **--client-protection=off|sign|encrypt** 选项已添加到通用解析程序中。
- 在所有工具中已重命名了以下选项：

- **--Kerberos** 变为 **--use-kerberos=required|desired|off**
- **--krb5-ccache** 变为 **--use-krb5-ccache=CCACHE**
- **--scope** 变为 **--netbios-scope=SCOPE**
- **--use-ccache** 变为 **--use-winbind-ccache**
- 以下选项已从所有工具中删除：
  - **-e** 和 **--encrypt**
  - 从 **--use-winbind-ccache** 中删除了 **-c**
  - 从 **--netbios-scope** 中删除了 **-i**
  - **-S** 和 **--signing**
- 要避免重复选项，某些选项已从以下工具中删除或重命名了：
  - **ndrdump:-l** 对于 **--load-dso** 不再可用
  - **net:-l** 对于 **--long** 不再可用
  - **sharesec:-V** 对于 **--viewsddl** 不再可用
  - **smbcquotas:--user** 已重命名为 **--quota-user**
  - **nmbd:--log-stdout** 已重命名为 **--debug-stdout**
  - **smbd:--log-stdout** 已重命名为 **--debug-stdout**
  - **winbindd:--log-stdout** 已重命名为 **--debug-stdout**

([BZ#2065646](#))

### RHEL 9 中 NFS 客户端和服务器的更改

- RHEL 9.0 NFS 服务器和客户端不再支持不安全的 GSS Kerberos 5 加密类型 **des-cbc-crc**。
- NFS 客户端不再支持使用 UDP 传输挂载文件系统。

([BZ#1952863](#))

### 现在，使用版本 1802 创建 GFS2 文件系统

RHEL 9 中的 GFS2 文件系统采用格式版本 1802 创建。这可启用以下功能：

- **trusted** 命名空间的扩展属性 ("trusted.\* xattrs") 可被 **gfs2** 和 **gfs2-utils** 识别。
- **rgrplvb** 选项默认为活动状态。这允许 **allowgfs2** 将更新的资源组数据附加到 DLM 锁定请求，因此获取锁定的节点不需要从磁盘更新资源组信息。这在某些情况下提高了性能。

使用新格式版本创建的文件系统将无法被挂载到以前的 RHEL 版本以及 **fsck.gfs2** 工具的旧版本下，将无法对其进行检查。

用户可以运行带有 **-o format=1801** 选项的 **mkfs.gfs2** 命令，创建采用较旧版本的文件系统。

用户可以在卸载的文件系统中通过运行 **tunefs2 -r 1802 device** 来升级旧文件系统的格式版本。不支持降级格式版本。

(BZ#1616432)

### RHEL 9 提供 **nvml** 软件包版本 1.10.1

RHEL 9.0 将 **nvml** 软件包更新至 1.10.1 版本。在这个版本中，添加了功能并修复在电源丢失时潜在的数据崩溃错误。

(BZ#1874208)

### 添加了对 **exFAT** 文件系统的支持

RHEL 9.0 支持扩展文件分配表(exFAT)文件系统。现在，您可以挂载、格式化并通常使用这个文件系统，这通常在闪存内存中使用。

(BZ#1943423)

### **rpcctl** 命令现在显示 SunRPC 连接信息

在这个版本中，您可以使用 **rpcctl** 命令显示 SunRPC **sysfs** 文件中有关系统的 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#2059245)

### 为 LVM 限制一组设备

默认情况下，RHEL 9 中的 LVM 只使用您明确选择的设备。使用新命令 **lvmdesires** 和 **vgimportdesires** 来选择特定的设备。使用 **pvcreate**、**vgcreate** 和 **vgextend** 命令间接为 **lvmdesires** 选择新设备（如果尚未选中）。LVM 忽略附加到系统的设备，直到您使用其中一个命令选择它们。**lvmdesires** 命令将所选设备列表保存在设备文件 **/etc/lvm/devices/system.devices** 中。当您启用新设备文件功能时，**lvmdesires.conf** 过滤或任何其他命令行配置过滤器都无法正常工作。如果您删除或禁用设备文件，LVM 会将过滤器应用到所有附加的设备。有关这个功能的详情请参考 **lvmdesires(8)** man page。

(BZ#1749513)

### 现在完全支持使用 **nvme\_tcp.ko** 的 NVMe/TCP 主机

现在完全支持使用 **nvme\_tcp.ko** 内核模块的 TCP/IP 网络(NVMe/TCP)存储 Nonvolatile Memory Express(NVMe/TCP)存储。带有 **nvmet\_tcp.ko** 模块的 NVMe/TCP 目标在 RHEL 9.0 中处于 Unmaintained 状态。

(BZ#2054441)

### **multipathd** 现在支持检测 **FPIN-Li** 事件

当您为 **marginal\_pathgroups** 配置选项添加了新值 **fpin** 时，您使 **multipathd** 能够监控 Link Integrity Fabric Performance Impact Notification(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，并依赖光纤通道结构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 事件的光纤通道结构上检测边缘路径变得更为强大。

(BZ#2053642)

## 4.12. 高可用性和集群

对于新创建的集群，**resource-stickiness** 资源 meta-attribute 现在默认为 1 而不是 0。

在以前的版本中，**resource-stickiness** 资源 meta-attribute 的默认值对于新创建的集群具有默认值 0。这个 meta-attribute 现在默认为 1。

如果粘性为 0，集群可以根据需要移动资源，以便在节点间平衡资源。这可能会导致资源在不相关的资源启动或停止时移动。对于正的粘性，资源更倾向于保持原处，只有在其他情况超出粘性时才会移动。这可能会导致新添加的节点在没有管理员干预的情况下无法获取分配给它们的任何资源。这两种方法都有潜在的意外行为，但大多数用户更喜欢使用某种粘性。此 meta-attribute 的默认值已更改为 1 以反映这种首选项。

只有新创建的集群会受到这个更改的影响，因此现有集群的行为不会更改。优先选择其集群的旧行为的用户可以从资源默认值中删除 **resource-stickiness** 条目。

(BZ#1850145)

### 新的 LVM 卷组标志来控制自动激活

LVM 卷组现在支持 **setautoactivation** 标记，它控制在启动时是否自动激活您从卷组中创建逻辑卷。当创建一个由 Pacemaker 在集群中管理的卷组时，使用 **vgcreate --setautoactivation n** 命令将这个标志设置为 **n**，以便卷组防止可能的数据崩溃。如果您在 Pacemaker 集群中使用现有卷组，请使用 **vgchange --setautoactivation n** 设置标志。

(BZ#1899214)

### 新的 pcs resource status display 命令

**pcs resource status** 和 **pcs stonith status** 命令现在支持以下选项：

- 您可以使用 **pcs resource status node=node\_id** 命令显示在特定节点上配置的资源状态，以及 **pcs stonith status node=node\_id** 命令。您可以使用这些命令显示集群和远程节点上的资源状态。
- 您可以使用 **pcs resource status resource\_id** 和 **pcs stonith status resource\_id** 命令显示单个资源的状态。
- 您可以使用 **pcs resource status tag\_id** 和 **pcs stonith status tag\_id** 命令显示所有资源的状态。

(BZ#1290830, BZ#1285269)

### 针对 pcs resource secure-disable 命令的新减少输出显示选项

**pcs resource safe-disable** 和 **pcs resource disable --safe** 命令在错误报告后打印一个非常长的模拟结果。现在，您可以为这些命令指定 **--brief** 选项，以只打印错误信息。现在，错误报告总是包含受影响资源的资源 ID。

(BZ#1909901)

### 新的 pcs 命令更新 SCSI 隔离设备而不重启所有其他资源

使用 **pcs stonith update** 命令更新 SCSI 隔离设备会导致重启运行于 stonith 资源所在的同一节点上运行的所有资源。新的 **pcs stonith update-scsi-devices** 命令允许您在不重启其他群集资源的情况下更新 SCSI 设备。

(BZ#1872378)

### 在集群节点子集中为隔离配置 watchdog-only SBD

在以前的版本中，要使用仅限 watchdog 的 SBD 配置，集群中的所有节点都必须使用 SBD。这会阻止在一个集群中使用 SBD，有些节点支持它，但其他节点（通常是远程节点）需要某种其他形式的隔离。用户现在可以使用新的 **fence\_watchdog** 代理配置仅限 watchdog 的 SBD 设置，允许集群配置，其中只有 watchdog-only SBD 进行隔离类型，其他节点使用其他隔离类型。集群只能有一个这样的设备，它必须命名为 **watchdog**。

([BZ#1443666](#))

### 内部错误的详细 Pacemaker 状态显示

如果由于某种原因，Pacemaker 无法执行资源或隔离代理，例如没有安装代理，或者有一个内部超时，Pacemaker 状态现在会显示内部错误的详细退出原因。

([BZ#1470834](#))

### pcmk\_delay\_base 参数现在可以为不同的节点使用不同的值

在配置隔离设备时，您现在可以使用 **pcmk\_delay\_base** 参数为不同的节点指定不同的值。这允许在双节点集群中使用单个隔离设备，每个节点有不同的延迟。这有助于防止每个节点同时尝试隔离其他节点的情况。要为不同的节点指定不同的值，您可以使用与 **pcmk\_host\_map** 类似的语法将主机名映射为该节点的延迟值。例如，在隔离 node1 时，node1:0;node2:10s 将不会使用延迟，在隔离 node2 时，会有 10 秒的延迟。

([BZ#1082146](#))

### 支持 pcmk\_host\_map 值中的特殊字符

**pcmk\_host\_map** 属性现在在值前面使用反斜杠(\)来支持 **pcmk\_host\_map** 值中的特殊字符。例如，您可以指定 **pcmk\_host\_map="node3:plug\ 1"**，来在主机别名中包含一个空格。

([BZ#1376538](#))

### OpenShift 的新隔离代理

**fence\_kubevirt** 隔离代理现在可用于 Red Hat OpenShift Virtualization 上的 RHEL 高可用性。有关 **fence\_kubevirt** 代理的详情，请查看 **fence\_kubevirt(8)** 手册页。

([BZ#1977588](#))

### 现在完全支持 pcs cluster setup 命令的本地模式版本

默认情况下，**pcs cluster setup** 命令会自动将所有配置文件与集群节点同步。**pcs cluster setup** 命令现在完全支持 **--corosync-conf** 选项。指定这个选项可将命令切换到 **本地** 模式。在这个模式中，**pcs** 命令行界面会创建一个 **corosync.conf** 文件，并将其只保存到本地节点的指定文件中，而无需与其它节点通信。这可让您在脚本中创建 **corosync.conf** 文件，并使用脚本处理该文件。

([BZ#2008558](#))

### 自动删除位置约束以下资源移动

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。默认情况下，在资源被移动后，命令创建的位置约束会被自动删除。这不一定将资源重新移到原始节点；此时可以在哪里运行这些资源取决于您最初配置的资源。如果您要移动资源并保留生成的约束，请使用 **pcs resource move-with-contraint** 命令。

([BZ#2008575](#))

### pcs 支持 OCF Resource Agent API 1.1 标准

**pcs** 命令行界面现在支持 OCF 1.1 资源和 STONITH 代理。作为此支持的实施的一部分，任何代理的元数据都必须符合 OCF 模式，代理是否为 OCF 1.0 还是 OCF 1.1 代理。如果代理的元数据不符合 OCF 架构，**pcs** 会考虑代理无效，除非指定了 **--force** 选项，否则不会创建或更新代理的资源。**pcsd** Web UI 和 **pcs** 命令用于列出代理，现在从列表中省略带有无效元数据的代理。

([BZ#2018969](#))

### **pcs** 现在接受 **Promoted** 和 **Unpromoted** 作为角色名称

**pcs** 命令行界面现在接受 **Promoted** 和 **Unpromoted** anywhere 角色，在 Pacemaker 配置中指定。这些角色名称是与之之前的 RHEL 版本中 **Master** 和 **Slave** Pacemaker 角色的功能，它们是在配置显示和帮助页面中可见的角色名称。

([BZ#2009455](#))

### **pcsd** Web UI 更新版本

**pcsd** Web UI（用于创建和配置 Pacemaker/Corosync 集群的图形用户界面）已更新。更新的 Web UI 提供改进的用户体验以及通过其他红帽 Web 应用程序中使用的 PatternFly 框架构建的标准化接口。

([BZ#1996067](#))

## 4.13. 动态编程语言、网页和数据库服务器

### RHEL 9 中的 Python

**Python 3.9** 是 RHEL 9 中的默认 **Python** 实现。**Python 3.9** 在 BaseOS 存储库中的非模块化 **python3** RPM 软件包中分发，通常默认安装。**Python 3.9** 将支持 RHEL 9 的整个生命周期。

其他版本的 **Python 3** 将以 RPM 软件包的形式发布，并通过 AppStream 软件仓库提供较短的生命周期，并可并行安装。

**python** 命令(`/usr/bin/python`)和其他 **Python**-related 命令（如 **pip**）位于未指定版本的表单中，并指向默认的 **Python 3.9** 版本。

**Python 2** 不随 RHEL 9 提供。

有关 RHEL 9 中的 **Python** 的更多信息，请参阅 [Python 简介](#)。

([BZ#1941595](#), [JIRA:RHELPLAN-80598](#))

### **Node.js 16** 在 RHEL 9 中可用

RHEL 9 提供了一个 Long Term Support(LTS)版本 16 的 **Node.js**，它是一个软件开发平台，用于使用 JavaScript 编程语言构建快速、可扩展的网络应用程序。

与 **Node.js 14** 相比，**Node.js 16** 中的显著变化包括：

- **V8** 引擎已升级至版本 9.4。
- **npm** 软件包管理器已升级至 8.3.1 版本。
- 新的 **Timers Promises** API 提供了一组替代的计时器功能，用于返回 **Promise** 对象。
- **Node.js** 现在与 **OpenSSL 3.0** 兼容。
- **Node.js** 现在提供一个新的实验性 **Web Streams** API 和实验性 ECMAScript 模块(ESM)加载程序 hook API。

**Node.js 16** 是此 Application Stream 的初始版本，您可以轻松地作为 RPM 软件包安装。**Node.js 16** 生命周期比 RHEL 9 的生命周期短。详情请查看 [Red Hat Enterprise Linux 应用程序流生命周期](#) 文档。其它 **Node.js** 版本将作为模块提供，其在将来的 RHEL 9 次要发行版中的生命周期也较短。

([BZ#1953491](#))

## RHEL 9 提供 Ruby 3.0

RHEL 9 使用 **Ruby 3.0.3** 提供，它比 **Ruby 2.7** 提供了很多性能改进、程序错误和安全修复以及新功能。

主要改进包括：

- 并发和并行功能：
  - **Ractor** 作为一个提供线程安全并行执行的 Actor-model 抽象作为实验功能提供。
  - **Fiber Scheduler** 作为一种实验功能引进。**Fiber Scheduler** 截获阻塞操作，在不更改现有代码的情况下启用轻量级并发操作。
- 静态分析功能：
  - 引进了 **RBS** 语言，它描述了 **Ruby** 程序的结构。**rbs** gem 已被添加来解析使用 **RBS** 编写的类型定义。
  - 引进了 **TypeProf** 实用工具，它是 **Ruby** 代码的类型分析工具。
- 与 **case/in** 表达式的模式匹配不再是实验性。
- 多行模式匹配（一个实验功能）已被重新设计。
- **find pattern** 已被添加为实验功能。

改进的性能：

- 将长代码粘贴到 **Interactive Ruby Shell(IRB)** 的速度现在明显加快。
- **measure** 命令已添加到 **IRB** 中以进行测量。

其他显著变化包括：

- 关键字参数现在与其他参数分开。
- 用户安装的 gems 的默认目录现在是 **\$HOME/.local/share/gem/**，除非 **\$HOME/.gem/** 目录已存在。

**Ruby 3.0** 是此 Application Stream 的初始版本，您可以轻松地作为 RPM 软件包安装。额外的 **Ruby** 版本将会作为模块提供，在以后的 RHEL 9 的次版本中会带有较短的生命周期。

([JIRA:RHELPLAN-80758](#))

## RHEL 9 引入了 Perl 5.32

RHEL 9 包括 **Perl 5.32**，它提供很多程序错误修复和增强 5.30。

主要改进包括：

- **Perl** 现在支持 Unicode 版本 13.0。
- **qr** quote-like 运算符已被改进。

- **POSIX::mblen()**, **mbtowc**, 和 **wctomb** 功能现在可以在转换状态区域上工作，在 C99 上执行 thread-safe on C99，在具有 locale thread-safety 的平台上执行，长度参数现在是可选的。
- 新的实验性 **isa infix** 操作符用于测试一个给定的对象是一个给定类的实例，还是从其中衍生出来的类。
- alpha assertions 不再是实验性。
- 脚本运行不再是实验性的。
- 功能检查现在可以更快。
- 现在，**Perl** 可以在优化前转储编译的模式。

**Perl 5.32** 是此 Application Stream 的初始版本，您可以轻松地作为 RPM 软件包安装。在以后的 RHEL 9 版本中，会将额外的 **Perl** 版本作为模块提供，其生命周期会短。

(JIRA:RHELPLAN-80759)

## RHEL 9 包括 PHP 8.0

RHEL 9 提供了 **PHP 8.0**，与版本 7.4 相比，这个版本提供了多个程序错误修复和增强。

主要改进包括：

- 新的命名参数是独立于顺序且自我记录的参数，并可让您只指定所需的参数。
- 新的属性允许您将结构化元数据与 PHP 的原生语法搭配使用。
- 新的 union 类型可让您使用在运行时验证的原生 union 类型声明，而不是运行时验证的 PHPDoc 注解。
- 现在，如果参数验证失败，内部功能会更加一致地引发 Error 异常而不是警告。
- 新的 Just-In-Time 编译引擎可显著提高应用程序性能。
- PHP 的 **Xdebug** 调试和生产力扩展已更新至版本 3。与 **Xdebug 2** 相比，这个版本对功能和配置进行了重大更改。

**PHP 8.0** 是此 Application Stream 的初始版本，您可以轻松地作为 RPM 软件包安装。在以后的 RHEL 9 版本中，会将其他 **PHP** 版本作为模块提供更短的生命周期。

如需更多信息，请参阅[使用 PHP 脚本语言](#)。

(BZ#1949319)

## RHEL 9 提供 Git 2.31 和 Git LFS 2.13

RHEL 9 提供了 **Git 2.31**，它比 RHEL 8 中的版本 2.27 提供了很多改进和性能。主要变更包括：

- **git status** 命令现在报告稀疏签出的状态。
- 现在，您可以将 **--add-file** 选项与 **git archive** 命令一起使用，从树状标识符将未跟踪的文件包含在快照中。
- 您可以使用 **clone.defaultremotename** 配置变量自定义源远程存储库的 nickname。

- 您可以配置 **git format-patch** 命令创建的输出文件名的最大长度。在以前的版本中，长度限制为 64 字节。
- 删除了对已弃用的 PCRE1 库的支持。

另外，**Git 大文件存储(LFS)** 扩展版本 2.13 现已正式发布。RHEL 8 中发布的版本 2.11 的增强包括：

- **Git LFS** 现在支持 SHA-256 软件仓库。
- **Git LFS** 现在支持 **socks5h** 协议。
- **git lfs install** 和 **git lfs uninstall** 命令提供了一个新的 **--worktree** 选项。
- **git lfs migrate import** 命令提供了一个新的 **--above** 参数。

(BZ#1956345, BZ#1952517)

## RHEL 9 中的 Subversion 1.14

RHEL 9 由 **Subversion 1.14** 提供。**Subversion 1.14** 是此应用程序流的初始版本，您可以将其作为 RPM 软件包轻松安装。在以后的 RHEL 9 次要发行本中，会将其他 **Subversion** 版本作为模块提供更短的生命周期。

(JIRA:RHELPLAN-82578)

## Apache HTTP 服务器中的显著变化

RHEL 9.0 提供 Apache HTTP 服务器的版本 2.4.51。2.4.37 版本的显著变化包括：

- Apache HTTP 服务器控制接口(**apachectl**)：
  - 现在，**apachectl status** 输出禁用了 **systemctl** pager。
  - 现在，如果您传递了附加参数，则 **apachectl** 命令会失败，而不是发出警告。
  - **apachectl graceful-stop** 命令现在会立即返回。
  - **apachectl configtest** 命令现在在不更改 SELinux 上下文的情况下执行 **httpd -t** 命令。
  - RHEL 中的 **apachectl(8)** man page 现在完全指明了与上游 **apachectl** 之间的差异。
- Apache eXtenSion 工具(**pxs**)：
  - 构建 **httpd** 软件包时，**/usr/bin/apxs** 命令不再使用或公开编译器选择的标志。现在，您可以使用 **/usr/lib64/httpd/build/vendor-apxs** 命令应用与构建 **httpd** 相同的编译器标志。要使用 **vendor-apxs** 命令，您必须首先安装 **redhat-rpm-config** 软件包。
- Apache 模块：
  - **mod\_lua** 模块现在在一个单独的软件包中提供。
  - Apache HTTP 服务器的新 **mod\_jk** 连接器是利用 Apache JServ 协议(AJP)将 Web 服务器与 Apache Tomcat 和其他后端连接的模块。
  - 新的 **mod\_proxy\_cluster** 模块提供基于 httpd 的负载均衡器，它使用通信通道将请求从负载均衡器转发到一组应用服务器节点。应用程序服务器节点使用此连接来传输服务器端负载，通过一组称为 Mod-Cluster Management Protocol(MCMP)的 HTTP 方法将系和生命周期事

件回发到负载均衡器。这种附加反馈通道允许 **mod\_proxy\_cluster** 提供一定程度的智能和粒度，在其他负载均衡解决方案中未找到。此模块要求在后端服务器中安装 **ModCluster** 客户端才能成功通信。

- 配置语法更改：
  - 在由 **mod\_access\_compat** 模块提供的已弃用的 **Allow** 指令中，注释（# 字符）现在会触发语法错误，而不是静默忽略。
- 其他更改：
  - 内核线程 ID 现在直接在错误信息中使用，从而使它们准确且更简洁。
  - 多个小幅改进和漏洞修复。
  - 模块作者可使用多个新接口。

从 RHEL 8 开始，**httpd** 模块 API 没有向后兼容的更改。

Apache HTTP Server 2.4 是此 Application Stream 的初始版本，您可以将其作为 RPM 软件包轻松安装。

如需更多信息，请参阅[设置 Apache HTTP web 服务器](#)。

(JIRA:RHELPLAN-68364, BZ#1931976, JIRA:RHELPLAN-80725)

## RHEL 9 提供了 nginx 1.20

RHEL 9 包括 **nginx 1.20** web 和代理服务器。与版本 1.18 相比，这个版本提供了很多程序错误修复、安全修复、新功能及改进。

新特性：

- **nginx** 现在支持使用在线证书状态协议(OCSP)的客户端 SSL 证书验证。
- **nginx** 现在支持基于最小可用空间量清除缓存。此支持作为 **proxy\_cache\_path** 指令的 **min\_free** 参数实施。
- 添加了一个新的 **ngx\_stream\_set\_module** 模块，它可让您为变量设置值。
- 添加了一个新的 **nginx-mod-devel** 软件包，它提供所有必要的文件，包括 RPM 宏和 **nginx** 源代码，以便为 **nginx** 构建外部动态模块。

增强的指令：

- 现在，可以使用多个新指令，如 **ssl\_conf\_command** 和 **ssl\_reject\_handshake**。
- **proxy\_cookie\_flags** 指令现在支持变量。

改进了 HTTP/2:

- **ngx\_http\_v2** 模块现在包含 **lingering\_close**、**lingering\_time**、**lingering\_timeout** 指令。
- 处理 HTTP/2 中的连接已与 HTTP/1.x 一致。从 **nginx 1.20** 中，使用 **keepalive\_timeout** 和 **keepalive\_requests** 指令，而不是移除的 **http2\_recv\_timeout**、**http2\_idle\_timeout** 和 **http2\_max\_requests** 指令。

**Nginx 1.20** 是此 Application Stream 的初始版本，您可以将其作为 RPM 软件包轻松安装。额外的 **nginx** 版本将会作为模块提供，在以后的 RHEL 9 次版本中会带有较短的生命周期。

如需更多信息，请参阅[设置和配置 NGINX](#)。

([BZ#1953639](#)、[BZ#1991720](#))

## RHEL 9 中的 Varnish Cache 6.6

RHEL 9 包括 **Varnish Cache 6.6**，这是一个高性能 HTTP 反向代理。

自 6.0 版本开始的主要变化包括：

- 提高了日志处理工具的性能，如 **varnishlog**
- 改进了统计的准确性
- 在缓存查询中有多个优化
- 各种配置更改
- 修复了很多改进和程序错误修复

**Varnish Cache 6** 是此 Application Stream 的初始版本，您可以将其作为 RPM 软件包轻松安装。

([BZ#1984185](#))

## RHEL 9 引入了 Squid 5

RHEL 9 通过 **Squid 5.2** 提供，这是一个用于 Web 客户端的高性能代理缓存服务器，支持 FTP、Gopher 和 HTTP 数据对象。与版本 4 相比，这个版本提供了很多程序错误修复、安全修复、新功能及改进。

新特性：

- **Squid** 通过使用 Happy Eyeballs(HE)算法提高责任。
  - **Squid** 现在会在很快请求转发时使用接收的 IP 地址，需要它而无需等待所有潜在的转发目的地完全解析。
  - 可用的新指令包括：**happy\_eyeballs\_connect\_gap**、**happy\_eyeballs\_connect\_limit**，和 **happy\_eyeballs\_connect\_timeout** 指令。
  - **dns\_v4\_first** 指令已被删除。
- **Squid** 现在在 Content Delivery Networks(CDN)中使用 **CDN-Loop** 标头作为循环检测的来源。
- **Squid** 引入了对 SSL bumping 的对等支持。
- 提供了新的互联网内容适配器协议(ICAP)尾随程序功能，它允许 ICAP 代理在邮件正文后可靠地发送消息元数据。

对配置选项的更改：

- **mark\_client\_packet** 配置选项已替换了 **clientside\_mark**。
- **shared\_transient\_entries\_limit** 配置选项已替换 **collapsed\_forwarding\_shared\_entries\_limit**。

**Squid 5** 是此 Application Stream 的初始版本，您可以将其作为 RPM 软件包轻松安装。

如需更多信息，请参阅[配置 Squid 缓存代理服务器](#)。

([BZ#1990517](#))

### RHEL 9 中的 MariaDB 10.5

RHEL 9 提供 **MariaDB 10.5**。**MariaDB 10.5** 是此 Application Stream 的初始版本，您可以轻松地作为 RPM 软件包安装。在以后的 RHEL 9 次要发行本中，其他 **MariaDB** 版本将会作为模块提供较短的生命周期。

如需更多信息，请参阅[使用 MariaDB](#)。

([BZ#1971248](#))

### RHEL 9 包括 MySQL 8.0

RHEL 9 提供 **MySQL 8.0**。**MySQL 8.0** 是此 Application Stream 的初始版本，您可以轻松地作为 RPM 软件包安装。**MySQL 8.0** 生命周期比 RHEL 9 的生命周期短。详情请查看 [Red Hat Enterprise Linux 应用程序流生命周期](#) 文档。

有关用法的详情，请参考 [使用 MySQL](#)。

(JIRA:RHELPLAN-78673)

### RHEL 9 提供 PostgreSQL 13

RHEL 9 提供 **PostgreSQL 13**。**PostgreSQL 13** 是此 Application Stream 的初始版本，您可以轻松地作为 RPM 软件包安装。在以后的 RHEL 9 次要发行本中，其他 **PostgreSQL** 版本将会作为模块提供更短的生命周期。

如需更多信息，请参阅[使用 PostgreSQL](#)。

(JIRA:RHELPLAN-78675)

### RHEL 9 中的 Redis 6.2

RHEL 9 由 **Redis 6.2** 提供，它比 RHEL 8 提供的 6.0 版本提供了很多程序错误和安全修复和增强。

值得注意的是，**Red Hat** 服务器配置文件现在位于专用目录中：`/etc/redis/redis.conf` 和 `/etc/redis/sentinel.conf`。在 RHEL 8 版本中，这些文件的位置分别是 `/etc/redis.conf` 和 `/etc/redis-sentinel.conf`。

**Redis 6** 是此 Application Stream 的初始版本，您可以轻松地作为 RPM 软件包安装。

([BZ#1959756](#))

### 新软件包：perl-Module-Signature

RHEL 9 引进了 **perl-Module-Signature** Perl 模块。使用这个新模块，您可以为 **cpan** 启用签名检查来缓解 CVE-2020-16156。如需更多信息，请参阅[如何在 perl-CPAN 中缓解 perl-App-cpanminus 和 CVE-2020-16156 中的 CVE-2020-16154](#)。

([BZ#2039361](#))

## 4.14. 编译器和开发工具

### RHEL 9 支持 IBM POWER10 处理器

从 Linux 内核中，通过系统工具链（GCC、binutils、glibc）Red Hat Enterprise Linux 9 已被更新，使其包含对 IBM 的最新 POWER 处理器( POWER10)的支持。RHEL 9 适用于 POWER10 上的工作负载，在以后的版本中有所改进。

(BZ#2027596)

## GCC 11.2.1 可用

RHEL 9 由 GCC 版本 11.2.1 提供。重要的程序错误修复和增强包括：

### 一般改进

- GCC 现在默认为 DWARF 版本 5 调试格式。
- 诊断中显示的列号默认为实际列号并尊重多列字符。
- 直接线性代码向向量化时，会考虑整个功能。
- 如果每个条件表达式中包含比较同一变量，则可以将一系列条件表达式转换为 switch 语句。
- 流程优化改进：
  - 新的 IPA-modref pass 由 **-fipa-modref** 选项控制，跟踪函数调用的副作用并改进点分析的精度。
  - 通过 **-fipa-icf** 选项控制相同的代码是明显的，以增加统一功能的数量并减少编译内存使用。
- 链路优化改进：
  - 链接时间优化(LTO) 可以让编译器通过在链接时使用其中间表示，在程序的所有转换单元中执行各种优化。如需更多信息，请参阅 [链接时间优化](#)。
  - 改进了连接过程中的内存分配，以减少峰值内存用量。
- 在 IDE 中使用新的 **GCC\_EXTRA\_DIAGNOSTIC\_OUTPUT** 环境变量，您可以在不调整构建标志的情况下请求机器可读的"fix-it hints"。
- 由 **-fanalyzer** 选项运行的静态分析器会显著提高，提供很多程序错误修复和增强。

### 特定于语言的改进

#### C family

- C 和 C++ 编译器支持 OpenMP 构造和 OpenMP 5.0 规范的"非恢复循环"例程。
- 属性：
  - 新的 **no\_stack\_protector** 属性标记不应通过堆栈保护来检测的功能(**-fstack-protector**)。
  - 改进的 **malloc** 属性可用于识别分配器和放大器 API 对。
- 新警告：
  - **-Wsizeof-array-div** 通过 **-Wall** 选项启用，它会警告两个 **sizeof** 操作符的不同（当第一个应用到一个数组，而 divisor 与数组的大小不同时）。
  - **-Wstringop-overread** 默认启用，警告字符串对字符串功能的调用，尝试将以参数形式传递给它们的阵列结束。

- 增强的警告：
  - **-Wfree-nonheap-object** 检测到更多调用实例来利用没有从动态内存分配功能返回的指针来取消分配函数。
  - **-Wmaybe-uninitialized** 诊断将指针和引用未初始化的内存传递给使用 **const-qualified** 参数的功能。
  - **-Wuninitialized** 检测到从未初始化动态分配的内存中读取。

## C

- 通过 **-std=c2x** 和 **-std=gnu2x** 选项支持来自即将发布的 C2X 版本中的几个新功能。例如：
  - 支持 标准属性。
  - 支持 **\_\_has\_c\_attribute** preprocessor operator。
  - 标签可能会在 compound 语句的末尾出现。

## C++

- 默认模式改为 **-std=gnu++17**。
- 现在，C++ 库 **libstdc++** 改进了 C++17 支持。
- 实施了几个新的 C++20 功能。请注意，C++20 支持是实验性的。有关功能的更多信息，请参阅 [C++20 语言功能](#)。
- C++ 前端对一些即将出现的 C++23 草案功能提供了实验性支持。
- 新警告：
  - **-Wctad-maybe-unsupported**（默认禁用），提醒在类型为 deduction 指南中的执行类模板参数。
  - **-Wrange-loop-construct**（由 **-Wall** 启用），当基于范围循环时警告是创建不必要的且资源低效的副本。
  - **-Wmismatched-new-delete**（由 **-Wall** 启用），警告对 Operator 的调用，并发出从不匹配形式的 Operator 返回的指针，或从其他不匹配分配功能返回。
  - **-Wvexing-parse**（默认启用）会警告大多数 vexing 解析规则：当声明与变量定义类似时，C++ 语言需要被解释为函数声明。

## 特定于架构的改进

### 64 位 ARM 架构

- Armv8-R 架构通过 **-march=armv8-r** 选项进行支持。
- GCC 可自动执行增加、减法、乘法、乘法和减去复杂数字上的操作。

### AMD 和 Intel 64 位构架

- 添加了对 Intel AVX-VNNI 的新 ISA 扩展支持。**-mavxvnni** 编译器切换控制 AVX-VNNI 内部。
- 新的 **-march=znver3** 选项支持基于 znver3 内核的 AMD CPU。

- [the x86-64 psABI supplement](#) 中定义的两个微架构级别通过新的 `-march=x86-64-v2`、`-march=x86-64-v3` 和 `-march=x86-64-v4` 选项支持。

## IBM Z 架构

- GCC 11.2.1 默认为 IBM z14 处理器。

## IBM Power 系统

- GCC 11.2.1 默认为 IBM POWER9 处理器。
- GCC 编译器现在支持使用新的 `-mcpu=power10` 命令行选项的 POWER10 指令

([BZ#1986836](#), [BZ#1870016](#), [BZ#1870025](#), [BZ#1870028](#), [BZ#2019811](#), [BZ#2047296](#))

## 用于捕获 glibc 优化数据的新命令

新的 `ld.so --list-diagnostics` 命令可以捕获影响 `glibc` 优化决策的数据，如在一个机器可读的文件中的 IFUNC 选择和 `glibc-hwcaps` 配置。

([BZ#2023422](#))

## 对 binutils 的显著变化

RHEL 9 对 `binutils` 包括以下更改：

- `binutils` 现在支持 Intel 的 AMX/TMUL 指令集，从而提高了可利用此新功能的应用程序的性能。
- `assembler`、`linker` 和其它二进制工具现在支持 POWER10 指令。

([BZ#2030554](#), [BZ#1870021](#))

## `sched_getcpu` 实现现在可以（可选）使用 `rseq`（可重启序列）来在 64 位 ARM 架构和其他架构上提高性能

以前 64 位 ARM 架构上的 `sched_getcpu` 实现使用 `getcpu` 系统调用，其在大多数并行算法中的使用效率太慢。其他架构使用 `vDSO`（虚拟动态共享对象）加速来临时解决这个问题。使用 `rseq` 实现 `sched_getcpu` 大幅提高了 64 位 ARM 架构上的性能。其他架构有略微改进。

要将 `sched_getcpu` 配置为使用 `rseq`，请设置 `GLIBC_TUNABLES=glibc.pthread.rseq=1` 环境变量：

```
# GLIBC_TUNABLES=glibc.pthread.rseq=1
# export GLIBC_TUNABLES
```

([BZ#2024347](#))

## 更新了性能工具和调试器

RHEL 9.0 提供了以下性能工具和调试器：

- GDB 10.2
- Valgrind 3.18.1
- SystemTap 4.6
- Dyninst 11.0.0

- elfutils 0.186

(BZ#2019806)

### DAWR 功能在 IBM POWER10 上的 GDB 中进行了改进

RHEL 9 提供了 GDB 10.2，它改进了 DAWR 功能。在 IBM POWER10 处理器上为 GDB 启用新的硬件观察点功能。例如，添加了一组新的 DAWR/DAWRX 寄存器。

(BZ#1870029)

### GDB 支持 IBM POWER10 的新前缀指令

GDB 10.2 完全支持 POWER10 上的 Power ISA 3.1 前缀的说明，它包括了八个字节的前缀指令。在 RHEL 8.4 中，GDB 只支持四字节说明。

(BZ#1870031)

### RHEL 9 提供了 boost 1.75.0

RHEL 9 提供了 **boost** 软件包版本 1.75.0。与 1.67.0 版本相比，重要的程序错误修复和增强包括：

- **Boost.Signals** 库已被删除，并替换为只包括标头的 **Boost.Signals2** 组件。
- **boost-jam** 软件包中的 **bjam** 工具已被 **boost-b2** 软件包中的 **b2** 替代。
- 新的库：
  - **Boost.Contracts**
  - **Boost.HOF**
  - **Boost.YAP**
  - **Boost.Safe Numerics**
  - **Boost.Outcome**
  - **Boost.Histogram**
  - **Boost.Variant2**
  - **Boost.Nowide**
  - **Boost.StaticString**
  - **Boost.STL\_Interfaces**
  - **Boost.JSON**
  - **Boost.LEAF**
  - **Boost.PFR**

(BZ#1957950)

### RHEL 9 提供 LLVM Toolset 13.0.1

RHEL 9 提供 LLVM Toolset 版本 13.0.1 提供。与版本 12.0.1 相比，重要的程序错误修复和增强包括：

- clang 现在支持带有声明属性 `[[clang::musttail]]`（在 C++ 中）和 `__attribute__((musttail))`（在 C 中）的保证 tial 调用。
- clang 现在支持 `-Wreserved-identifier` 警告，它会在代码中使用保留标识符时警告开发人员。
- clang 的 `-Wshadow` 标志现在还检查影子结构的绑定。
- Clang 的 `-Wextra` 现在也意味着 `-Wnull-pointer-subtraction`。
- clang 现在支持带有声明属性 `[[clang::musttail]]`（在 C++ 中）和 `__attribute__((musttail))`（在 C 中）的保证 tial 调用。

在 RHEL 9 中，您可以作为 RPM 软件包轻松地安装 `llvm-toolset`。

(BZ#2001107)

### CMake 3.20.2 中的显著变化

RHEL 9 提供了 CMake 3.20.2。要在需要版本 3.20.2 或 less 的项目中使用 CMake，请使用命令 `cmake_minimum_required`（版本 3.20.2）。

主要变更包括：

- 现在，可以使用目标属性 `CXX_STANDARD`, `CUDA_STANDARD`, `OBJCXX_STANDARD` 或通过使用编译功能中的 `cxx_std_23` 元功能功能来指定 C++23 的编译器模式。
- CUDA 语言支持现在允许 NVIDIA CUDA 编译器是符号链接。
- Intel oneAPI NextGen LLVM 编译器现在支持 `IntelLLVM` 编译器 ID。
- cmake 现在通过与 Android NDK 的工具链文件合并来促进 Android 的跨编译。
- 运行 `cmake(1)` 以生成项目构建系统时，现在拒绝以连字符开头的未知命令行参数。

有关新功能和已弃用功能的更多信息，请参阅 [CMake 发行注记](#)。

(BZ#1957948)

### RHEL 9 提供 Go 1.17.7

RHEL 9 与 Go Toolset 版本 1.17.7 一同发布。与 1.16.7 版本相比，重要的程序错误修复和增强包括：

- 添加了将片段转换为数组指针的选项。
- 添加了对 `//go:build` 行的支持。
- 在 amd64 上功能调用性能的改进。
- 在堆栈追踪中更明确地格式化函数参数。
- 包括闭包的函数可以是 `inline` 模式。
- 对 x509 证书进行解析的过程减少了资源消耗。

在 RHEL 9 中，您可以轻松地安装 `go-toolset` 作为 RPM 软件包。

(BZ#2014087)

### OpenSSL 3 支持 go FIPS 模式

现在，在使用 Go FIPS 模式时，您可以使用 OpenSSL 3 库。

([BZ#1984110](#))

### RHEL 9 提供 Rust Toolset 1.58.1

RHEL 9 提供了 Rust Toolset 版本 1.58.1。与 1.54.0 版本相比，重要的程序错误修复和增强包括：

- Rust 编译器现在支持语言的 2021 版，其特点是闭包中的分离捕获、用于数组的 **Intolterator**、一个新的 Cargo 功能解析器等等。
- 添加了对新自定义配置文件的 Cargo 支持。
- Cargo 去掉了重复的编译器错误。
- 添加了新的开放范围模式。
- 以格式字符串形式添加了捕获的标识符。

如需更多信息，请参阅 [Rust 1.55](#)[Rust 1.56](#)[Rust 1.57](#)[Rust 1.58](#)

在 RHEL 9 中，您可以轻松地安装 **rust-toolset** 作为 RPM 软件包。

([BZ#2002885](#))

### RHEL 9 提供 pcp 软件包版本 5.3.5

RHEL 9 带有 Performance Co-Pilot(**pcp**)软件包 5.3.5 版。自 5.3.1 版本开始，添加了新的 **pcp-pmda-bpf** 子软件包，从 **eBPF** 程序提供使用 BPF CO-RE (**libbpf** 和 **BTF**) 的性能数据。

([BZ#1991764](#))

### 用于访问 PCP 中的 SQL 服务器指标的 Active Directory 身份验证

在这个版本中，系统管理员可以配置 **pmdamssql(1)**，以使用 Active Directory(AD)身份验证安全地连接到 SQL Server 指标。

([BZ#1847808](#))

### 新的 pcp-ss PCP 工具现在可用

**pcp-ss** PCP 实用程序报告 **pmdasockets(1)** PMDA 收集的套接字统计。命令与许多 **ss** 命令行选项和报告格式兼容。它还在实时模式下提供本地或远程监控的优势，以及之前记录的 PCP 归档重新显示历史。

([BZ#1981223](#))

### RHEL 9 提供 grafana 7.5.11

RHEL 9 带有 **grafana** 软件包版本 7.5.11。与版本 7.5.9 相比的主要变化包括：

- 添加了一个新的**准备时间序列**转换，以向后兼容不支持新数据帧格式的面板。
- 更新了密码恢复功能，以使用 HMAC-SHA-256 而不是 SHA-1 生成密码重置令牌。

([BZ#1993215](#))

### RHEL 9 提供 grafana-pcp 3.2.0

RHEL 9 提供 **grafana-pcp** 软件包版本 3.2.0。与版本 3.1.0 相比，重要的程序错误修复和增强包括：

- 为 PCP Redis 添加了一个新的 MS SQL 服务器仪表盘。
- 在 PCP Vector eBPF/BCC Overview 仪表盘中添加了空直方存储桶。
- 修复了 PCP Redis 的 **metric ()** 函数没有返回所有指标名称的错误。

([BZ#1993156](#))

### 通过在 grafana-pcp 中 Vector 数据源的中央 pmproxy 来访问远程主机

在一些环境中网络策略不允许从仪表板查看器浏览器中直接连接到受监控的主机的连接。在这个版本中，可以自定义 **hostspect** 以连接到中央 **pmproxy**，后者将请求转发到单个主机。

([BZ#1845592](#))

### 新软件包：ansible-pcp

**ansible-pcp** 软件包包含 Performance Co-Pilot (PCP) 的角色和相关软件，如 Redis 和 Grafana，用来实现 **metrics** RHEL 系统角色。

([BZ#1957566](#))

### RHEL 9 提供 python-jsonpointer 2.0

RHEL 9 提供 **python-jsonpointer** 软件包版本 2.0。

与版本 1.9 相比的主要变化包括：

- Python 版本 2.6 和 3.3 已被弃用。
- **python-jsonpointer** 模块现在会自动检查无效转义序列的指针。
- 现在，您可以在命令行中以参数形式编写指针。
- 指针不能以 URL 编码的格式提交。

([BZ#1980256](#))

### .NET 6.0 可用

RHEL 9 带有 .NET 版本 6.0。主要改进包括：

- 支持 64 位 Arm(aarch64)
- 支持 IBM Z 和 LinuxONE(s390x)

如需更多信息，请参阅 [.NET 6.0 RPM 软件包](#) 和 [.NET 6.0 容器发行注记](#)。

.NET 6.0 是此应用程序流的初始版本，您可以作为一个 RPM 软件包轻松地安装它。.NET 6.0 的生命周期比 RHEL 9 的生命周期短。详情请查看 [Red Hat Enterprise Linux 应用程序流生命周期文档](#)。

([BZ#1986211](#))

### RHEL 9 中的 Java 实现

RHEL 9 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](#)。

(BZ#2021262)

## RHEL 9 中的 Java 工具

RHEL 9 AppStream 存储库包括以下 Java 工具：

- **Maven 3.6.3**，一种软件项目管理和理解工具。
- **Ant 1.10.9**，一种用于编译、汇编、测试和运行 Java 应用程序的 Java 库和命令行工具。

**Maven 3.6** 和 **Ant 1.10** 是这些 Application Streams 的初始版本，您可以作为非模块化 RPM 软件包轻松安装。

(BZ#1951482)

## CRB 存储库中提供的 SWIG 4.0

CodeReady Linux Builder(CRB)存储库中提供了 Simplified Wrapper 和 Interface Generator(SWIG)版本 4.0。此发行版本添加了对 **PHP 8** 的支持。

在 RHEL 9 中，您可以轻松地安装 **SWIG** 作为 RPM 软件包。

请注意，不支持 CodeReady Linux Builder 存储库中包含的软件包。

(BZ#1943580)

## 4.15. 身份管理

### 目录服务器不再使用全局更改日志

在这个版本中，Directory 服务器 changelog 已集成到主数据库中。在以前的版本中，Directory 服务器使用全局更改日志。但是，如果目录使用多个数据库，这可能会导致问题。现在，每个后缀在与常规数据库文件相同的目录中都有自己的 changelog。

(BZ#1805717)

### Ansible-freeipa 现在可在带有所有依赖项的 AppStream 存储库中

以前，在 RHEL 8 中，安装 **ansible-freeipa** 软件包之前，您必须首先启用 Ansible 存储库并安装 **ansible** 软件包。在 RHEL 8.6 和 RHEL 9 中，您可以在没有任何初始步骤的情况下安装 **ansible-freeipa**。安装 **ansible-freeipa** 会自动安装 **ansible-core** 软件包，它是更基本的 **ansible** 版本，作为依赖项。**ansible-freeipa** 和 **ansible-core** 都位于 **rhel-9-for-x86\_64-appstream-rpms** 存储库中。

RHEL 8.6 和 RHEL 9 中的 Ansible **-freeipa** 包含了它在 RHEL 8 中包含的所有模块。

(JIRA:RHELPLAN-100359)

### IdM 现在支持 automountlocation、automountmap 和 automountkey Ansible 模块

在这个版本中，**ansible-freeipa** 软件包包含 **ipaautomountlocation**、**ipaautomountmap** 和 **ipaautomountkey** 模块。您可以使用这些模块为登录到 IdM 位置的 IdM 客户端自动挂载目录。请注意，目前只支持直接映射。

(JIRA:RHELPLAN-79161)

### shadow-utils 中提供了管理 subID 范围的支持

在以前的版本中，**shadow-utils** 配置来自 **/etc/subuid** 和 **/etc/subgid** 文件的 subID 范围。在这个版本中，**/etc/nsswitch.conf** 文件中提供了 subID 范围配置，方法是在 **subid** 字段中设置值。如需更多信息，请参阅 **man subuid** 和 **man subgid**。另外，通过此次更新，**shadow-utils** 插件的 SSSD 实施可用，它提供来自 IPA 服务器的 subID 范围。要使用此功能，请将 **subid: sss** 值添加到 **/etc/nsswitch.conf** 文件。此解决方案在容器化环境中可能很有用，以促进无根容器。

请注意，如果 **/etc/nsswitch.conf** 文件由 **authselect** 工具配置，您必须遵循 **authselect** 文档中介绍的步骤。否则，您可以手动修改 **/etc/nsswitch.conf** 文件。

(BZ#1859252)

### IdM 中提供管理 subID 范围的支持

借助此次更新，您可以管理身份管理中用户的 ID 子范围。您可以使用 **ipa** CLI 工具或 IdM WebUI 界面向用户分配自动配置的 subID 范围，这可能在容器化环境中很有用。

(BZ#1952028)

### 身份管理安装软件包已进行演示

在以前的版本中，在 RHEL 8 中，IdM 软件包作为模块发布，需要启用流并安装与所需安装对应的配置集。IdM 安装软件包在 RHEL 9 中进行了演示，因此您可以使用以下 **dnf** 命令安装 IdM 服务器软件包：

对于没有集成 DNS 服务的服务器：

```
# dnf install ipa-server
```

对于具有集成 DNS 服务的服务器：

```
# dnf install ipa-server ipa-server-dns
```

(BZ#2080875)

### 传统的 RHEL ansible-freeipa 存储库的替代方案：Ansible Automation Hub

有了这个更新，您可以从 Ansible Automation Hub(AAH)下载 **ansible-freeipa** 模块，而不是从标准的 RHEL 存储库下载它们。通过使用 AAH，您可以从此存储库中 **ansible-freeipa** 模块的快速更新中受益。

在 AAH 中，**ansible-freeipa** 角色和模块以集合的形式发布。请注意，您需要 Ansible Automation Platform(AAP)订阅来访问 AAH 门户中的内容。您还需要 **ansible** 版本 2.9 或更高版本。

**redhat.rhel\_idm** 集合与传统的 **ansible-freeipa** 软件包具有相同的内容。但是，集合形式使用完全限定的集合名称(FQCN)，其中包含一个命名空间和集合名称。例如，**redhat.rhel\_idm.ipadnsconfig** 模块对应于 RHEL 存储库提供的 **ansible-freeipa** 中的 **ipadnsconfig** 模块。命名空间和集合名称的组合可确保对象是唯一的，并可无冲突地共享。

(JIRA:RHELPLAN-103147)

**Ansible-freeipa** 模块现在可以在 IdM 客户端上远程执行

在以前的版本中，**ansible-freeipa** 模块只能在 IdM 服务器上执行。这要求您的 Ansible 管理员具有对 IdM 服务器的 **SSH** 访问权限，从而导致潜在的安全威胁。有了这个更新，您可以在 IdM 客户端的系统上远程执行 **ansible-freeipa** 模块。因此，您可以以更安全的方法管理 IdM 配置和实体。

要在 IdM 客户端上执行 **ansible-freeipa** 模块，请选择以下选项之一：

- 对 IdM 客户端主机设置 playbook 的 **hosts** 变量。
- 将 **ipa\_context: client** 行添加到使用 **ansible-freeipa** 模块的 playbook 任务中。

您还可将 **ipa\_context** 变量设置为 IdM 服务器上的 **client**。但是，服务器上下文通常提供更好的性能。如果没有设置 **ipa\_context**，**ansible-freeipa** 会检查其是否在服务器或客户端上运行，并相应地设置上下文。请注意，在 IdM 客户端主机上执行将 **context** 设为 **server** 的 **ansible-freeipa** 模块会导致 **missing libraries** 的错误。

(JIRA:RHELPLAN-103146)

### **ipadnsconfig** 模块现在需要 **action: member** 来排除一个全局的转发者

在这个版本中，在 Identity Management (IdM) 中使用 **ansible-freeipa ipadnsconfig** 模块排除全局转发者需要在使用 **state: absent** 选项外还使用 **action: member** 选项。如果您只使用 playbook 中的 **state: absent**，而没有使用 **action: member**，则 playbook 将失败。因此，要删除所有全局转发器，您必须在 playbook 中单独指定它们。相反，**state: present** 选项不需要 **action: member**。

(BZ#2046325)

### AD 用户的自动私人组群支持集中配置

现在，您可以集中定义 IdM 客户端中 SSSD 的兼容版本如何从可信 Active Directory 域管理用户的专用组。在这个版本中，您可以为处理 AD 用户的 ID 范围明确设置 SSSD 的 **auto\_private\_groups** 选项的值。

如果没有显式设置 **auto\_private\_groups** 选项，它将使用默认值：

- 对于 **ipa-ad-trust-posix** ID 范围，默认值为 **false**。SSSD 始终使用 AD 条目的 **uidNumber** 和 **gidNumber**。AD 中必须存在具有 **gidNumber** 的组。
- 对于 **ipa-ad-trust** ID 范围，默认值为 **true**。SSSD 从 SID 映射 **uidNumber**，**gidNumber** 始终被设置为相同的值，并且始终映射私有组。

您还可以将 **auto\_private\_groups** 设置为第三个设置：**hybrid**。使用这个设置时，如果用户条目的 GID 等于 UID，则 SSSD 会映射私有组，但没有具有此 GID 的组。如果 UID 和 GID 有所不同，则该 GID 号的组必须存在。

此功能对于希望停止为用户私有组维护单独的组对象，但也希望保留现有用户专用组的管理人员非常有用。

(BZ#1957736)

### BIND 的可自定义日志记录设置

在这个版本中，您可以在 **/etc/named/ipa-logging-ext.conf** 配置文件中为身份管理服务服务器的 BIND DNS 服务器组件配置日志设置。

(BZ#1966101)

### 检索 IdM keytab 时自动发现 IdM 服务器

在这个版本中，在使用 `ipa-getkeytab` 命令检索 Kerberos keytab 时，不再需要指定 IdM 服务器主机名。如果没有指定服务器主机名，会使用 DNS 发现来查找 IdM 服务器。如果没有找到服务器，命令会返回 `/etc/ipa/default.conf` 配置文件中指定的 `host` 值。

(BZ#1988383)

## RHEL 9 提供 Samba 4.15.5

RHEL 9 提供了 Samba 4.15.5，它比版本 4.14 提供程序错误修复和增强：

- [Samba 工具中的选项已被重命名和删除，以获得一致的用户体验](#)
- 现在默认启用服务器多通道支持。
- 仅被 Windows 技术预览使用的 `SMB2_22`、`SMB2_24` 和 `SMB3_10` dialects 已被删除。

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#2013578)

## 使用日志分析器工具跟踪客户端请求

系统安全服务守护进程(SSSD)现在包含一个日志解析工具，它跟踪来自开始的请求，以在多个 SSSD 组件的日志文件上结束。

日志分析器工具可让您更轻松地了解 SSSD 调试日志，以帮助您对 SSSD 中的任何问题进行故障排除。例如，您可以提取并打印与 SSSD 进程间的特定客户端请求相关的 SSSD 日志。要运行分析器工具，请使用 `sssctl analyze` 命令。

(JIRA:RHELPLAN-97899)

## SSSD 现在默认记录后端

在这个版本中，SSSD 会将详细的调试日志保存在内存缓冲区中，并在出现故障时将其附加到日志文件中。默认情况下，以下错误级别会触发回溯：

- 0 级：严重失败
- 1 级：关键失败
- 2 级：严重失败

您可以通过在 `sssd.conf` 配置文件的对应部分设置 `debug_level` 选项来修改每个 SSSD 进程的行为：

- 如果将调试级别设置为 0，则只有级别 0 事件触发回溯。
- 如果将调试级别设置为 1，则级别 0 和 1 会触发回溯追踪。
- 如果将调试级别设置为 2 或更高版本，则事件在级别 0 到 2 时触发回溯追踪。

您可以通过在 `sssd.conf` 的对应部分将 `debug_backtrace_enabled` 选项设置为 `false` 来禁用 SSSD 进程。

[sssd]

```
debug_backtrace_enabled = true
debug_level=0
...

[nss]
debug_backtrace_enabled = false
...

[domain/idm.example.com]
debug_backtrace_enabled = true
debug_level=2
...
...
```

(BZ#1949149)

### SSSD 默认的 SSH 散列值现在与 OpenSSH 设置保持一致

`ssh_hash_known_hosts` 的默认值已更改为 `false`。它现在与 OpenSSH 设置保持一致，默认情况下不对主机名进行哈希处理。

但是，如果您需要继续哈希主机名，请将 `ssh_hash_known_hosts = True` 添加到 `/etc/sss/sss.conf` 配置文件的 `[ssh]` 部分。

(BZ#2014249)

### 目录服务器 12.0 基于上游版本 2.0.14

目录服务器 12.0 基于上游版本 2.0.14，它提供很多程序错误修复和增强。如需显著变化的完整列表，请在更新前阅读上游发行注记：

- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-14.html>
- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-13.html>
- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-12.html>
- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-11.html>
- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-10.html>
- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-9.html>
- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-8.html>
- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-7.html>
- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-6.html>
- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-5.html>
- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-4.html>
- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-3.html>
- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-2.html>

- <https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-1.html>

(BZ#2024693)

### 目录服务器现在在 tmpfs 文件系统中存储数据库的内存映射文件

在目录服务器中，`nsslapd-db-home-directory` 参数定义数据库内存映射文件的位置。此增强将参数的默认值从 `/var/lib/dirsrv/slapd-instance_name/db/` 改为 `/dev/shm/`。因此，在 `tmpfs` 文件系统中存储内部数据库，目录服务器的性能就会提高。

(BZ#2088414)

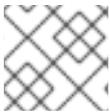
### FreeRADIUS 支持现已重新设计

在 RHEL 9 中，现有的 FreeRADIUS 产品现已简化，并与身份管理(IdM)的战略方向更加紧密地保持一致。为了为 IdM 用户提供最佳支持，红帽正在通过 FreeRADIUS 加强对这些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

以下模块不再被支持：

- MySQL、PostgreSQL、SQLite 和 unixODBC 数据库连接器
- `Perl` 语言模块
- REST API 模块



#### 注意

作为基本软件包的一部分提供的 PAM 身份验证和其他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删除模块的替代品，例如在 Fedora 项目中。

另外，现在对 `freeradius` 软件包的支持范围仅限于以下用例：

- 将 FreeRADIUS 作为无线身份验证提供者与 IdM 一起使用，来作为身份验证的后端源。身份验证通过 `krb5` 和 LDAP 身份验证软件包或在主 FreeRADIUS 软件包中作为 PAM 身份验证进行。
- 使用 FreeRADIUS，通过 `Python 3` 身份验证软件包为 IdM 中的身份验证提供真实源。

(JIRA:RHELDOS-17553)

## 4.16. DESKTOP

### GNOME 更新至版本 40

GNOME 环境现已从 GNOME 3.28 更新至 GNOME 40，有许多新功能。

GNOME 40 包含新的改进的**活动概述**设计。这为概览提供了更加一致的外观，并为导航系统和启动应用程序提供更好的体验。现在，工作区可以水平排列，窗口概览和应用程序网格可垂直访问。

GNOME 的其他改进包括：

- GNOME 的性能和资源使用率显著提高。

- 已刷新了可视化风格（包括用户界面、图标和桌面）。
- GNOME 应用程序不再使用应用程序菜单，该菜单位于顶部面板中。功能现在位于应用程序窗口的主菜单中。
- **Settings** 应用已重新设计。
- 改进了屏幕共享和远程桌面会话。
- 如果您使用专有的 NVIDIA 驱动程序，现在可以使用离散 GPU 启动应用程序：
  - a. 打开概述。
  - b. 右键单击短划线中的应用图标。
  - c. 选择菜单中的 **Launch on Discrete GPU** 项。
- **Power Off / Log Out** 菜单现在包含 **Suspend** 选项和一个新的 **Restart** 选项，它可在您保存 **Alt** 时重启系统进入引导装载程序菜单。
- 现在，flatpak 应用程序可以自动更新了。
- 现在，您可以使用拖放将概述中的应用图标分组到文件夹中。
- 现在，**Terminal** 应用支持从右到左和双向文本。
- **Pointer Location** 访问功能现在可以在 Wayland 中正常工作。启用该功能后，按 **Ctrl** 突出显示屏幕上的指针位置。
- GNOME shell 扩展现在由 **扩展** 应用程序管理，而不是 **软件**。**扩展** 应用程序处理更新扩展、配置扩展首选项以及删除或禁用扩展。
- 通知弹出现在包含一个 **Do Not Disturb** 按钮。按钮启用后，不会在屏幕上显示通知。
- 要求密码的系统对话框现在可以选择显示密码文本。
- **软件应用程序** 现在会自动检测计量网络，如移动数据网络。当当前网络被计量时，**软件** 会暂停更新以减少数据使用量。
- 现在，每个连接的显示都可以在 Wayland 会话中使用不同的刷新率。
- 部分显示缩放可作为实验性选项提供。它包含多个预配置的分值。  
要启用实验性部分扩展，请在启用了实验功能列表中添加 **scale-monitor-framebuffer** 值：

```
$ dconf write \
  /org/gnome/mutter/experimental-features \
  ["scale-monitor-framebuffer"]
```

因此，可以在 **Settings** 的 **Display** 面板中访问部分扩展选项。

有关 GNOME 中更改的更多详细信息，请参阅[发行注记](#)中的 3.30 到 40.0 版本。

(JIRA:RHELPLAN-101240)

**PipeWire** 现在是默认的音频服务

**Pipewire** 服务现在管理所有音频输出和输入。**Pipewire** 在一般用例中取代了 **PulseAudio** 服务，并在专业用例中使用 **JACK** 服务。系统现在从使用 **PulseAudio**、**JACK** 或 **ALSA** 框架的应用程序重定向音频到 **Pipewire**。

与之前的解决方案相比，**Pipewire** 的优点包括：

- 面向消费者和专业用户的统一解决方案
- 灵活的模块化架构
- 高性能和低延迟，类似于 **JACK** 服务
- 音频客户端之间的隔离以获得更高的安全性

您无需再为使用它的应用程序配置 **JACK** 服务。所有 **JACK** 应用程序现在都可以在默认的 RHEL 配置中工作。

**pulseaudio** 仍可在 RHEL 中提供，您可以启用它而不是 **PipeWire**。详情请参阅 [从 PipeWire 切换到 PulseAudio](#)。

(JIRA:RHELPLAN-101241)

## GNOME 中提供了电源配置集

现在，您可以在 GNOME 环境中的设置界面的 **Power** 面板中在多个电源配置集间进行切换。电源配置文件针对所选目标优化了各种系统设置。

可用的电源配置集如下：

### 性能

针对高系统性能进行优化，并缩短冷却时间。这个配置集仅在特定的所选系统配置中可用。

### balanced

提供标准系统性能和功耗。这是默认配置文件。

### Power Saver

提高电池的使用时间，降低系统性能。这个配置集会在电池量低时自动激活。

重新启动系统后，您的电源配置集配置会保留。

**power** 配置文件功能可从 **power-profiles-daemon** 软件包获得，该软件包默认安装。

(JIRA:RHELPLAN-101242)

## 现在，语言支持由 langpacks 提供

现在可从 **langpacks** 软件包中获得对各种语言的支持。您可以使用以下软件包名称自定义要安装的语言支持级别，其中 **code** 是该语言的简短 ISO 代码，例如 **es** 代表西班牙语：

### langpacks-core-code

提供基本语言支持，包括：

- **glibc** 区域设置
- 默认字体
- 默认输入法（如果语言需要）

### langpacks-core-font-code

仅提供语言的默认字体。

### langpacks-code

提供完整的语言支持，除了基本语言支持外，还包括以下内容：

- 翻译
- 拼写检查器字典
- 其他字体

(JIRA:RHELPLAN-101247)

### 轻量级、单一应用程序环境

对于仅存在单个应用程序的图形用例，现在提供了一个轻量级用户界面(UI)。

您可以在单一应用程序会话中启动 GNOME，也称为 kiosk 模式。在这个会话中，GNOME 只显示一个您配置的应用程序的完整屏幕窗口。

单应用程序会话的资源密集型明显低于标准的 GNOME 会话。

如需更多信息，请参阅[限制会话到单个应用程序](#)。

(JIRA:RHELPLAN-102552)

### 登录和桌面会话中的安全分类横幅

现在，您可以配置一个分类横幅来说明系统的整体安全分类级别。对于用户必须了解登录的系统的的安全分类级别的部署，这非常有用。

分类横幅可能出现在以下上下文中，具体取决于您的配置：

- 在运行的会话中
- 在锁定屏幕中
- 在登录屏幕中

分类横幅可以是可以被忽略的通知模式，也可以是一个永久的横幅。

如需更多信息，请参阅[显示系统安全分类](#)。

(BZ#2031186)

### 默认墙纸添加了红帽徽标

现在，默认的 RHEL 墙纸会显示红帽徽标。徽标位于屏幕的左上角。

要禁用徽标，请禁用背景徽标 GNOME Shell 扩展。

(BZ#2057150)

### Firefox 现在在 PKCS#12 文件中使用更强大的加密

Firefox Web 浏览器使用 PKCS#12 文件来建立客户端验证证书。在以前的版本中，Firefox 使用旧算法加密这些文件：

- PBE-SHA1-RC2-40 来加密 PKCS#12 文件中的证书
- PBE-SHA1-3DES 用于加密 PKCS#12 文件中的密钥

在这个版本中，Firefox 默认使用更强大的算法加密文件：

- 带有 PBKDF2 的 AES-256-CBC 用于加密 PKCS#12 文件中的证书
- 带有 PBKDF2 的 AES-128-CBC 用于加密 PKCS#12 文件中的密钥

在这个版本中，PKCS#12 文件与联邦信息处理标准(FIPS)兼容。

Firefox 作为非默认选项，支持传统的加密算法。

([BZ#1764205](#))

## 4.17. 图形基础结构

### Wayland 会话现在默认带有 NVIDIA 驱动程序

当使用 NVIDIA 驱动程序时，如果驱动程序配置支持 Wayland，则桌面会话现在默认选择 Wayland 显示协议。在以前的 RHEL 版本中，NVIDIA 驱动程序总是禁用 Wayland。

要在您的系统中使用 NVIDIA 驱动程序启用 Wayland，请在内核命令行中添加以下选项：

- `nvidia-drm.modeset=1`
- `NVreg_PreserveVideoMemoryAllocations=1`

请注意，从 RHEL 8.0 开始，带有其他图形驱动程序的默认显示协议是 Wayland。

目前，带有 NVIDIA 驱动程序的 Wayland 会话仍然不完整，并有一些已知的问题。红帽正在积极和 NVIDIA 合作来解决 GPU 堆栈中的这些问题。

有关使用 NVIDIA 驱动程序进行 Wayland 的一些限制，请参阅 [已知问题部分](#)。

([JIRA:RHELPLAN-119000](#))

## 4.18. WEB 控制台

### 通过 web 控制台为 sudo 和 SSH 进行智能卡验证

在以前的版本中，无法使用智能卡验证来获取 sudo 权限或在 web 控制台中使用 SSH。在这个版本中，身份管理用户可以使用智能卡获得 sudo 权限，或者使用 SSH 连接到其他主机。



#### 注意

只能使用一个智能卡进行验证并获得 sudo 权限。不支持为 sudo 使用一个单独的智能卡。

([JIRA:RHELPLAN-95126](#))

### 在 web 控制台中没有重启的内核安全补丁

此 web 控制台更新允许用户在不强制重启的情况下使用 `kpatch` 框架应用内核安全补丁。管理员也可以自动订阅未来的内核到实时补丁流。

(JIRA:RHELPLAN-95056)

### RHEL web 控制台默认提供 Insights 注册

在这个版本中，当您使用 Red Hat Enterprise Linux web 控制台注册 RHEL 系统时，默认会选择 **Connect this system to Red Hat Insights**. 选项。如果您不想连接 Insights 服务，请取消这个选择。

([BZ#2049441](#))

### Cockpit 现在支持使用现有 TLS 证书

在这个版本中，证书没有严格的文件权限要求（如 **root:cockpit-ws 0640**），因此可与其他服务共享。

(JIRA:RHELPLAN-103855)

## 4.19. RED HAT ENTERPRISE LINUX 系统角色

### Networking 系统角色现在支持 SAE

在 Wi-Fi 保护的访问版本 3(WPA3)网络中，等同(SAE)方法的同步身份验证可确保不会传输加密密钥。有了这个增强，Networking RHEL 系统角色支持 SAE。因此，管理员现在可以使用 Networking 系统角色来配置到 Wi-Fi 网络（其使用 WPA-SAE）的连接。

([BZ#1993304](#))

### Networking 系统角色现在支持 owe

Networking RHEL 系统角色现在支持 Opportunistic Wireless Encryption (owe)。**owe** 是一个无线验证密钥管理类型，它在 Wi-Fi 客户端和访问点之间使用加密，并保护 Wi-Fi 客户端免受嗅探攻击。要使用 owe，将无线验证密钥管理类型 **key\_mgmt** 字段设置为 **owe**。

([BZ#1993377](#))

### Firewall 系统角色现在支持设置防火墙默认区域

zones 代表一种更透明管理传入流量的概念。这些区域连接到联网接口或者分配一系列源地址。每个区域的防火墙规则可以独立管理，使管理员能够定义复杂的防火墙设置并将其应用到流量。此功能允许设置用作分配接口的默认区域，与 **firewall-cmd --set-default-zone zone-name** 相同。

([BZ#2022461](#))

### Storage RHEL 系统角色现在支持 LVM VDO 卷

有了这个增强，您可以使用 Storage 系统角色来管理逻辑卷管理器卷(LVM)虚拟数据优化器(VDO)卷。LVM 文件系统管理 VDO 卷并使用这个功能，现在可以在 LVM 卷上压缩和重复数据删除。因此，VDO 有助于优化存储卷的使用。

([BZ#1978488](#))

### Storage 系统角色支持以百分比表示卷大小

此增强添加了对存储 RHEL 系统角色以池总大小的百分比表示 LVM 卷大小的支持。您可以将 LVM 卷大小指定为池/VG 大小的百分比，例如：除文件系统可读大小（如 10g、50 GiB）之外，还有 50%。

([BZ#1984583](#))

### Storage 系统角色支持缓存卷

此增强添加了对存储 RHEL 系统角色创建和管理缓存的 LVM 逻辑卷的支持。LVM 缓存可用于提高较慢的逻辑卷的性能，方法是将 LV 的数据的子集临时存储在较小的、更快的设备上，例如 SSD。

([BZ#2016517](#))

### 在 Firewall 角色中添加或删除源

在这个版本中，您可以使用 **source** 参数在防火墙设置配置中添加或删除源。

([BZ#2021667](#))

### Microsoft SQL Server Management 的新 Ansible 角色

新的 **microsoft.sql.server** 角色旨在帮助 IT 和数据库管理员自动处理 Red Hat Enterprise Linux 上 SQL Server 的设置、配置和性能调优的过程。

([BZ#2013853](#))

### Microsoft SQL 系统角色现在支持对断开连接的或 Satellite 订阅的自定义存储库

在以前的版本中，在断开连接的环境中，用户需要从自定义服务器拉取软件包或需要指向 Satellite 或 Capsule 的 Satellite 用户没有 **microsoft.sql.server** 的支持。在这个版本中，您可以通过提供 **mssql\_rpm\_key**、**mssql\_server\_repository** 和 **mssql\_client\_repository** 变量来解决它，您可以用来自定义存储库以从中下载软件包。如果没有提供 URL，**mssql** 角色将使用官方 Microsoft 服务器来下载 RPM。

([BZ#2064648](#))

### MSSQL 角色会在其受管配置文件中始终使用 "Ansible\_managed" 注释

MSSQL 角色生成 **/var/opt/mssql/mssql.conf** 配置文件。在这个版本中，MSSQL 角色使用 Ansible 标准 **ansible\_managed** 变量将 "Ansible managed" 注释插入到配置文件。注释指示不应直接编辑配置文件，因为 MSSQL 角色将覆盖该文件。因此，配置文件包含一个声明，表示配置文件由 Ansible 管理。

([BZ#2064690](#))

### RHEL 系统角色的 Ansible Core 支持

在 RHEL 9 GA 发行版中，提供了 Ansible Core，它的支持范围有限，以便启用 RHEL 支持的自动化用例。Ansible Core 替换了 Ansible Engine，它在以前的 RHEL 版本中提供的是一个独立的存储库中。Ansible Core 在 RHEL 的 AppStream 存储库中提供。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](#)。

如果您需要 Ansible Engine 支持，或者需要支持非 RHEL 自动化用例，请[创建一个红帽支持问题单](#)。

([JIRA:RHELPLAN-103540](#))

### 支持在一个 elasticsearch 输出字典中配置多个 elasticsearch 主机

在以前的版本中，**server\_host** 参数用于为单个主机取字符串值。此功能增强将其调整为底层的 **rsyslog omelasticsearch** 规格，因此它现在也取一个字符串列表来支持多个主机。因此，它会被调整为主机，符合底层 **rsyslog omelasticsearch** 的规格。因此，用户可以在一个 **elasticsearch** 输出字典中配置多个 **elasticsearch** 主机。

([BZ#1986460](#))

### RHEL 系统角色现在支持 VPN 管理

在以前的版本中，在 Linux 中设置安全且正确配置的 IPsec 隧道和虚拟专用网络(VPN)解决方案很难。有了这个增强，您可以使用 VPN RHEL 系统角色更轻松地在大量主机之间设置和配置用于主机到主机和网

格连接的 VPN 隧道。因此，您在 RHEL 系统角色项目中有一个用于 VPN 和 IPsec 隧道配置的一致且稳定的配置接口。

([BZ#2019341](#))

### SSHD RHEL 系统角色现在支持非排他配置片断

借助此功能，您可以通过不同的角色和 playbook 配置 SSHD，而不必使用命名空间重写以前的配置。命名空间类似于置入目录，并为 SSHD 定义非专用配置片断。因此，如果您需要只配置一小部分配置，而不是整个配置文件，您可以使用来自不同角色的 SSHD RHEL 系统角色。

([BZ#1978752](#))

### Network Time Security (NTS)选项添加到 timesync RHEL 系统角色中

**NTS** 选项被添加到 Timesync RHEL 系统角色中，以便在客户端服务器上启用 **NTS**。NTS 是为网络时间协议(NTP)指定的新安全机制。NTS 可以安全同步 NTP 客户端，无需特定于客户端，并可扩展到大量客户端。**NTS** 选项只支持版本 4.0 及之后的版本中的 **chrony** NTP 供应商。

([BZ#1978753](#))

### 支持 HA Cluster RHEL 系统角色

High Availability Cluster(HA Cluster)角色现已获得全面支持。以下是值得注意的配置：

- 配置隔离设备、资源、资源组和包括元数据属性和资源操作的资源克隆
- 配置资源位置约束、资源托管约束、资源顺序约束和资源票据约束
- 配置集群属性
- 配置集群节点、自定义集群名称和节点名称
- 配置多链接集群
- 配置集群在引导时是否自动启动

运行该角色会删除会删除角色不支持或者运行角色时未指定的任何配置。

HA 集群系统角色目前不支持 SBD。

([BZ#2054401](#))

### 支持 Elasticsearch 的 Rsyslog 用户名和密码验证

此更新将 Elasticsearch 用户名和密码参数添加到 Logging 系统角色中。因此，您可以启用 Rsyslog 使用用户名和密码进行 Elasticsearch 验证。

([BZ#1990490](#))

### 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#2031555](#))

## 添加了对为 LVM 指定 `raid_level` 的支持

RHEL 9.0 支持使用 `lvraid` 功能将逻辑卷管理(LVM)卷分组到 RAID 中。

([BZ#2016518](#))

## 证书角色在其 `hook` 脚本中一致使用 "Ansible\_managed" 注释

在这个版本中，证书角色生成 `pre-scripts` 和 `post-scripts` 以支持提供程序，该角色使用 Ansible 标准 "ansible\_managed" 变量插入 "Ansible managed" 注释：

- `/etc/certmonger/pre-scripts/script_name.sh`
- `/etc/certmonger/post-scripts/script_name.sh`

注释指示不应直接编辑脚本文件，因为证书角色可覆盖该文件。因此，配置文件包含一个声明，表示配置文件由 Ansible 管理。

([BZ#2054364](#))

## 新的选项 `auto_gateway` 控制默认的路由行为

在以前的版本中，`DEFROUTE` 参数无法通过配置文件进行配置，只能通过命名每个路由来手动进行配置。这个更新在 `ip` 配置部分中为连接添加了一个新的 `auto_gateway` 选项，您可以使用它来控制默认的路由行为。您可以使用以下方法配置 `auto_gateway`：

- 如果设置为 `true`，则默认网关设置适用于默认路由。
- 如果设置为 `false`，则删除默认路由。
- 如果未指定，`network` 角色将使用所选 `network_provider` 的默认行为。

([BZ#1978773](#))

## 支持 `network` 系统角色中添加的所有绑定选项

这个更新提供了对 `network` RHEL 系统角色的所有绑定选项的支持。因此，它可让您灵活地控制绑定接口上的网络传输。因此，您可以通过为该接口指定多个选项来控制绑定接口上的网络传输。

([BZ#2054435](#))

## NetworkManager 支持使用其 PCI 地址指定网卡

在以前的版本中，在设置连接配置文件的过程中，只允许 NetworkManager 使用其名称或 MAC 地址来指定网卡。在本例中，设备名称不稳定，并且 MAC 地址需要清单来维护使用 MAC 地址的记录。现在，您可以根据连接配置文件中的 PCI 地址来指定网卡。

([BZ#1999162](#))

## Network 系统角色现在直接管理 Ansible 的配置文件

在这个版本中，`network` 角色在 `/etc/sysconfig/network-scripts` 中生成 `ifcfg` 文件。然后，它会使用标准的 `ansible_managed` 变量插入注释 "Ansible managed"。此注释表示 `ifcfg` 文件无法直接编辑，因为 `network` 角色可能会覆盖它。处理 `ifcfg` 文件以添加 "Ansible managed" 注释的主要不同之处是，`network` 角色使用 `initscripts` 软件包，NetworkManager 使用 `nm` 软件包。

([BZ#2057657](#))

## RHEL 系统角色的 Ansible Core 支持

在 RHEL 9.0 中，提供了 Ansible Core，其支持范围有限，以便启用 RHEL 支持的自动化用例。Ansible Core 替换了之前在其它存储库中提供的 Ansible Engine。Ansible Core 在 RHEL 的 AppStream 存储库中提供。有关支持的用例的详情，请参阅 [RHEL 9 和 RHEL 8.6 及更新的 AppStream 软件仓库中包含的 Ansible Core 软件包的范围支持](#)。用户必须手动将他们的系统从 Ansible Engine 迁移到 Ansible Core。

([BZ#2012298](#))

### 现在支持 Cockpit 系统角色

在这个版本中，您可以在系统中安装和配置 web 控制台。因此，您可以自动管理 Web 控制台。

([BZ#2021028](#))

### Terminal session recording 系统角色在其管理的配置文件中使用 "Ansible managed" 注释

Terminal 会话记录角色会生成 2 配置文件：

- `/etc/sss/conf.d/sss-session-recording.conf`
- `/etc/tlog/tlog-rec-session.conf`

在这个版本中，Terminal 会话记录角色使用标准的 Ansible 变量 `ansible_managed`，将 "Ansible managed" 注释插入到配置文件中。注释指示不应直接编辑配置文件，因为 Terminal 会话记录角色会覆盖该文件。因此，配置文件包含一个声明，表示配置文件由 Ansible 管理。

([BZ#2054367](#))

### VPN 角色在其受管配置文件中以统一的方式使用 "Ansible\_managed" 注释

VPN 角色生成以下配置文件：

- `/etc/ipsec.d/mesh.conf`
- `/etc/ipsec.d/policies/clear`
- `/etc/ipsec.d/policies/private`
- `/etc/ipsec.d/policies/private-or-clear`

在这个版本中，VPN 角色使用 Ansible 标准 `ansible_managed` 变量将 "Ansible managed" 注释插入到配置文件。注释表示不应直接编辑配置文件，因为 VPN 角色可以覆盖该文件。因此，配置文件包含一个声明，表示配置文件由 Ansible 管理。

([BZ#2054369](#))

### Postfix 角色在其受管配置文件中以统一的方式使用 "Ansible\_managed" 注释

Postfix 角色生成 `/etc/postfix/main.cf` 配置文件。在这个版本中，Postfix 角色使用 Ansible 标准 `ansible_managed` 变量将 "Ansible managed" 注释插入到配置文件。这个注释指示，不应直接编辑配置文件，因为 Postfixrole 可以覆盖该文件。因此，配置文件包含一个声明，表示配置文件由 Ansible 管理。

([BZ#2057662](#))

### 已在 RHEL 9 中添加了 Firewall RHEL 系统角色

有了此增强，`rhel-system-roles.firewall` RHEL 系统角色被添加到 `rhel-system-roles` 软件包中。因此，管理员可以为受管节点自动执行其防火墙设置。

([BZ#2021665](#))

## SSH 客户端 RHEL 系统角色现在在 OpenSSH 8.7 中支持新的配置选项

在这个版本中，OpenSSH 更新至最新版本，它提供了新的配置选项，可在 SSH 客户端角色中配置新主机。

([BZ#2029427](#))

## 4.20. 虚拟化

### RHEL web 控制台新的虚拟化功能

在这个版本中，RHEL web 控制台在 Virtual Machines 页面中包含新功能。您现在可以：

- 重命名虚拟机
- 使用云镜像身份验证创建虚拟机
- 为虚拟机添加和删除 USB 和 PCI 设备
- 指定网络接口模型
- 在主机及其虚拟机间共享和取消共享文件

([JIRA:RHELPLAN-102009](#))

### QEMU 使用 Clang

QEMU 模拟器现在使用 Clang 编译器构建。这可让 RHEL 9 KVM 管理程序使用诸多高级安全和调试功能，并使未来功能开发更加高效。

([BZ#1940132](#))

### 虚拟机的 SafeStack

在 AMD64 和 Intel 64 硬件(x86\_64)的 RHEL 9 中，QEMU 模拟器可以使用 SafeStack，它是一种基于编译器的堆栈保护功能。SafeStack 可减少攻击者利用基于堆栈的缓冲区溢出更改堆栈中的返回指针并创建面向返回编程(ROP)攻击的能力。因此，在 RHEL 9 中托管的虚拟机对基于 ROP 的漏洞进行更加安全。

([BZ#1939509](#))

### Intel 64、AMD64 和 IBM Z 的 virtiofs 完全支持

现在，Intel 64、AMD64 和 IBM Z 构架中完全支持 virtio 文件系统(**virtiofs**)。使用 **virtiofs**，您可以在主机系统及其虚拟机之间高效地共享文件。

([JIRA:RHELPLAN-64576](#))

### KVM 客户端支持 AMD EPYC 7003 系列处理器

现在，在 KVM 管理程序和内核代码中添加了对 AMD EPYC 7003 系列处理器（也称为 **AMD Milan**）的支持，以及 libvirt API。这可让 KVM 虚拟机使用 AMD EPYC 7003 系列处理器。

([JIRA:RHELPLAN-65223](#))

### qemu-kvm 现在支持额外的机器类型

添加了一组基于 RHEL 9 的新机器类型，以供虚拟机(VM)使用。要获取主机上所有目前支持的机器类型，请使用 `/usr/libexec/qemu-kvm -M help` 命令。

另外，所有基于 RHEL 7.5.0 或更早版本的机器类型现在都不支持。它们还包括 **pc-i440fx-rhel7.5.0** 和更早的机器类型，它们在以前的 RHEL 主版本中是默认的。因此，在 RHEL 9 上试图启动此类机器类型的虚拟机会失败，并显示 **unsupported configuration** 错误。如果您在将主机升级到 RHEL 9 后遇到这个问题，请参阅 [红帽知识库](#)。

(JIRA:RHELPLAN-75866)

### 现在 IBM Z 上的虚拟化 CLI 支持中介设备

使用 **virt-install** 或 **virt-xml**，您现在可以将介质设备附加到虚拟机，如 **vfio-ap** 和 **vfio-ccw**。例如，这可以更灵活地管理 IBM Z 主机上的 DASD 存储设备和加密协处理器。另外，您可以使用 **virt-install** 创建一个使用现有 DASD 中介设备作为其主磁盘的虚拟机。有关这样做的步骤，请参阅 RHEL 9 指南中的配置和管理虚拟化。

(BZ#1995131)

### 模块 **libvirt** 守护进程

在 RHEL 9 中，**libvirt** 库使用 modular 守护进程来处理您主机上的单个虚拟化驱动程序集。例如，**virtqemud** 守护进程处理 QEMU 驱动程序。这样便可对涉及虚拟化驱动程序的各种任务进行微调，如资源负载优化和监控。

另外，单一 **libvirt** 守护进程 **libvirtd** 已被弃用。但是，如果您从 RHEL 8 升级到 RHEL 9，您的主机仍将使用 **libvirtd**，您可以在 RHEL 9 中继续使用它。但是，红帽建议改为切换到模块 **libvirt** 守护进程。

(JIRA:RHELPLAN-113994)

### 支持 Windows 11 和 Windows Server 2022 虚拟机

RHEL 9 支持使用 Windows 11 和 Windows Server 2022 作为 KVM 虚拟机上的客户机操作系统。

(BZ#2036856, BZ#2004161)

### **ksmtuned** 现在独立于 **qemu-kvm** 进行分发

要减少 KVM hypervisor 的足迹，**ksmtuned** 工具不再是 **qemu-kvm** 的依赖项。因此，如果您需要配置内核相同的页面合并(KSM)，您必须手动安装 **ksmtuned** 软件包。

(BZ#2069501, [BZ#1971678](#), [BZ#1972158](#))

### 新功能 : vTPM

RHEL 9 完全支持 Virtual Trusted Platform 模块(vTPM)。使用 vTPM，您可以将 TPM 虚拟加密处理器添加到 RHEL 9 KVM 管理程序中运行的虚拟机(VM)。这样，可以使用虚拟机生成、存储和管理加密密钥。

(JIRA:RHELPLAN-98617)

### Intel Atom P59 系列处理器的虚拟化支持

在这个版本中，RHEL 9 中的虚拟化增加了对 Intel Atom P59 系列处理器的支持，以前称为 Snow Ridge。因此，在 RHEL 9 中托管的虚拟机现在可以使用 **Snowridge** CPU 模型并使用处理器提供的新功能。

(BZ#1874187)

## 4.21. 云环境中的 RHEL

### RHEL 9 提供 WALinuxAgent 2.3.0.2

RHEL 9 与 Windows Azure Linux 代理(**WALinuxAgent**)软件包 2.3.0.2 一同发布。与版本 2.2.49 相比，重要的程序错误修复和增强包括：

- 添加了对 RequiredFeatures 和 GoalStateAggregateStatus API 的支持。
- 添加了扩展清单的回退位置。
- 在创建异常时，添加了缺少的对 str.format() 的调用。

([BZ#1972101](#))

## Azure 上的 RHEL 现在支持 MANA

在 Microsoft Azure 上运行的 RHEL 9 虚拟机现在可以使用 Microsoft Azure Network Adapter(MANA)。

([BZ#1957818](#))

## cloud-init 支持 VMware GuestInfo 数据源

在这个版本中，**cloud-init** 实用程序能够读取 VMware guestinfo 数据的数据源。因此，使用 **cloud-init** 在 VMware vSphere 上设置 RHEL 9 虚拟机现在更为高效且可靠。

([BZ#2040090](#))

## RHEL 9 虚拟机现在在 Azure 上的特定 ARM64 主机上被支持

现在，在基于 Ampere Altra ARM 处理器上运行的 Microsoft Azure hypervisor 支持使用 RHEL 9 作为客户机操作系统的虚拟机。

([BZ#1949613](#))

## cloud-init 支持 Microsoft Azure 上的用户数据

为 **cloud-init** 工具引进了 **--user-data** 选项。使用此选项时，您可以在 Azure 中设置 RHEL 9 虚拟机时，从 Azure 实例元数据服务(IMDS)传递脚本和元数据。

([BZ#2042351](#))

## cloud-init 的新 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，有了这种更改，默认的 **cloud-init** 配置已更新。因此，如果您有一个本地修改，请确保 **/etc/cloud/cloud.cfg** 文件包含 **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。如果发生这种情况，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)

## 4.22. 支持性

### sos 报告 现在提供了一个估算模式运行

这个 **sos 报告** 更新添加了 `--estimate-only` 选项，您可以使用它来估算从 RHEL 服务器收集 **sos 报告** 所需的磁盘空间。运行 `sos report --estimate-only` 命令：

- 对 **sos 报告** 执行一个试运行
- 连续地模拟所有插件，并估计其磁盘大小。

请注意，最终的磁盘空间估计是非常近似的。因此，建议您将估计值加倍。

(BZ#2011537)

## 4.23. 容器

### Podman 现在支持安全简短名称

现在可以在 `[aliases]` 表中的 `registries.conf` 文件中配置镜像的短名称别名。简短名称模式为：

- **Enforcing**:如果在镜像拉取过程中找不到匹配的别名，则 Podman 会提示用户选择一个非限定 registry。如果成功拉取了所选镜像，Podman 会自动在 `$HOME/.cache/containers/short-name-aliases.conf` 文件（rootless 用户）和 `/var/cache/containers/short-name-aliases.conf` 中记录一个新的短名称别名。如果无法提示用户（例如，stdin 或 stdout 而不是 TTY），则 Podman 会失败。请注意，如果两者都指定了相同的别名，则 `short-name-aliases.conf` 文件优先于 `registries.conf` 文件。
- **Permissive**:与 enforcing 模式类似，但如果用户无法提示，Podman 不会失败。相反，Podman 会按照指定顺序搜索所有非限定 registry。请注意，没有记录别名。

例如：

```
unqualified-search-registries=["registry.fedoraproject.org", "quay.io"]

[aliases]

"fedora"="registry.fedoraproject.org/fedora"
```

(JIRA:RHELPLAN-74542)

### container-tools 模块的更改

**container-tools** 模块包含 Podman、Buildah、Skopeo 和 runc 工具。滚动流，由 RHEL 8 中的 `container-tools:rhel8` 流表示，在 RHEL 9 中命名为 `container-tools:latest`。与 RHEL 8 类似，容器工具的稳定版本也会以数字化的流（如 3.0）提供。

有关 Container Tools Application Stream 的更多信息，请参阅 [Container Tools AppStream - Content Availability](#)。

(JIRA:RHELPLAN-73678)

**containers-common 软件包现在可用**

**containers-common** 软件包已添加到 **container-tools:latest** 模块中。**containers-common** 软件包包含常见配置文件和容器工具生态系统文档，如 Podman、Buildah 和 Skopeo。

(JIRA:RHELPLAN-77549)

**使用新软件包更新容器镜像**

例如，要使用最新的软件包更新 **registry.access.redhat.com/rhel9** 容器镜像，请使用以下命令：

```
# podman run -it registry.access.redhat.com/rhel9
# dnf update -y && rm -rf /var/cache/dnf
```

要安装特定的 **<package>**，请输入：

```
# dnf install <package>
```

如需更多信息，请参阅 [将软件添加到正在运行的 UBI 容器中](#)。

请注意，对于 RHEL 9，在镜像中更新或安装新软件包需要您在授权主机上运行。您可以使用 Red Hat Enterprise Linux Developer 订阅进行单独订阅，以免免费获得授权存储库的访问权限。

如需更多信息，请参阅 [No-cost Red Hat Enterprise Linux 独立开发人员订阅：常见问题](#)。

(JIRA:RHELPLAN-84168)

**container-tools meta-package 已更新**

包含 Podman、Buildah、Skopeo 和 runc 工具的 **container-tools** RPM 元软件包可用。与之前的版本相比，这个版本提供了程序错误修正和增强的列表。

(JIRA:RHELPLAN-118914)

**podman-py 软件包现在可用**

**podman-py** 软件包已添加到 **container-tools:3.0 stable** 模块流和 **container-tools:latest** 模块中。**podman-py** 软件包是要使用的 Podman RESTful API 的绑定库。

(BZ#1975462)

**控制组群版本 2 现已正式发布**

之前版本的控制组 1(cgroups v1)会导致各种应用程序的性能问题。最新版本的控制组 2(cgroups v2)可让系统管理员限制任何应用程序的资源，而不造成性能问题。

这个新版本的控制组群(cgroup v2)可在 RHEL 8 中启用，在 RHEL 9 中被默认启用。

(JIRA:RHELPLAN-73697)

**container-tools meta-package 现已正式发布**

RHEL 9 提供了 **container-tools** RPM meta-package 包括 Podman、Buildah、Skopeo、CRIU、Udica 和所有必需库。RHEL 9 不提供 stable 流。要获得对 Podman 的稳定访问、Buildah、Skopeo 等，请使用 RHEL EUS 订阅。

要安装 **container-tools** meta-package, 请输入 :

```
# dnf install container-tools
```

(BZ#2000871)

### 现在, 内核中的原生覆盖文件系统支持

覆盖文件系统支持现在可从内核 5.11 获得。非 root 用户即使在无根用户 (作为用户) 时, 也具有原生的覆盖性能。因此, 这个增强为希望在不需绑定挂载的情况下使用 overlayfs 的非 root 用户提供更好的性能。

(JIRA:RHELPLAN-99892)

### NFS 存储现在可用

现在, 如果您的文件系统支持 xattr, 则可以使用 NFS 文件系统作为容器和镜像的后端存储。

(JIRA:RHELPLAN-74543)

### container-tools meta-package 已更新

**container-tools** meta-package 包括 Podman、Buildah、Skopeo、CRIU、Udica 以及所有必需的库。与之前的版本相比, 这个版本提供了程序错误修正和增强的列表。

主要变更包括 :

- 由于网络堆栈的变化, Podman v3 和更早版本所创建的容器在 Podman v4.0 中不可用
- 原生覆盖文件系统可作为无根用户使用
- 现在容器中支持 NFS 存储
- 控制组群版本 2(cgroup v2)被默认启用
- 除非所有容器被销毁并重新创建, 否则不支持从 Podman v4 降级到 v3

有关 Podman 中显著变化的更多信息, 请参阅[上游发行注记](#)。

(JIRA:RHELPLAN-99889)

### crun 容器运行时现在为默认

**crun** 容器运行时现在是默认的运行时。**crun** 容器运行时支持注解, 允许容器访问非根用户的额外组。这对设置 setgid 或用户只能访问组的目录中的卷挂载有用。**crun** 和 **runc** 运行时都完全支持 **cgroup v2**。

(JIRA:RHELPLAN-99890)

### 控制组群版本 2 现已正式发布

以前的控制组群版本 cgroup 版本 1(cgroup v1)会导致各种应用程序出现性能问题。最新版本的控制组 cgroup 版本 2(cgroup v2)可让系统管理员为任何应用程序限制资源, 而不会造成性能问题。

在 RHEL 9 中, cgroup v2 被默认启用。

(JIRA:RHELPLAN-75322)

### 通用基础镜像现在包括在 Docker Hub 中

在以前的版本中，通用基础镜像只包括在 Red Hat 容器目录中。在这个版本中，Docker Hub 中也提供了通用基础镜像作为 Verified [Publisher](#) 镜像。

(JIRA:RHELPLAN-100032)

### openssl 容器镜像现在可用

**openssl** 镜像提供了一个 **openssl** 命令行工具，用于使用 OpenSSL 加密库的不同功能。使用 OpenSSL 库，您可以生成私钥、创建证书签名请求(CSR)并显示证书信息。

**openssl** 容器镜像位于这些存储库中：

- [registry.redhat.io/rhel9/openssl](https://registry.redhat.io/rhel9/openssl)
- [registry.access.redhat.com/ubi9/openssl](https://registry.access.redhat.com/ubi9/openssl)

(JIRA:RHELPLAN-100034)

### Netavark 网络堆栈现在可用

Netavark 堆栈是容器的网络配置工具。在 RHEL 9 中，Netavark 堆栈被完全支持并启用。

此网络堆栈具有以下功能：

- 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

(JIRA:RHELPLAN-101141)

### Podman 现在支持使用 YAML 文件自动构建和自动运行的 pod

**podman play kube** 命令会使用 YAML 文件自动构建并运行带有 pod 中多个容器的多个 pod。

(JIRA:RHELPLAN-108830)

### Podman 现在可以从 IdM 提供子UID 和 subGID 范围

subUID 和 subGID 范围现在可以由 IdM 管理。现在，您可以在单个中央存储中定义范围，而不是将相同的 `/etc/subuid` 和 `/etc/subgid` 文件部署到每一主机中。您必须修改 `/etc/nsswitch.conf` 文件，将 **sss** 添加到 services map 行：**services: files sss**。

如需了解更多详细信息，请参阅 IdM 文档中的 [手动管理子ID 范围](#) 部分。

(JIRA:RHELPLAN-100020)

## 第 5 章 BUG 修复

这部分论述了 Red Hat Enterprise Linux 9.0 中修复的错误，它们对用户有严重影响。

### 5.1. 安装程序和镜像创建

#### **--leavebootorder** 不再更改引导顺序

在以前的版本中，在 UEFI 系统中使用 **--leavebootorder** 用于 bootloader kickstart 命令无法正常工作，并更改了引导顺序。这会导致安装程序在 UEFI 引导菜单中安装的系统列表的顶部添加 RHEL。

在这个版本中解决了这个问题，使用 **--leavebootorder** 不再更改引导装载程序中的引导顺序。UEFI 系统 RHEL 支持 **--leavebootorder**。

([BZ#2025953](#))

#### Anaconda 在运行 **%post** 脚本前设置静态主机名

在以前的版本中，当 Anaconda 被设置安装程序环境主机名到 kickstart 配置中的值时 (**network --hostname**)，它会用来设置临时主机名。在 **%post** 脚本运行期间执行的一些操作（如网络设备激活）会导致主机名重置为通过反向 **dns** 获取的值。

在这个版本中，Anaconda 会在 kickstart **%post** 脚本运行时将安装程序环境的静态主机名设置为稳定。

([BZ#2009403](#))

#### 用户现在可以在 RHEL for Edge Installer 蓝图中指定用户帐户

在以前的版本中，在没有升级的边缘提交中定义的用户帐户（如添加 rpm 软件包）在蓝图上执行更新可能会导致用户被应用后锁定。它导致用户在升级现有系统时重新定义用户帐户。这个问题已被解决，允许用户在安装时在 RHEL for Edge Installer 蓝图中指定用户帐户，这会在安装时创建用户，而不是将用户作为 **ostree** 提交的一部分。

([BZ#2060575](#))

#### 引导菜单中删除了基本图形模式

在以前的版本中，**基本图形** 模式用于在带有不支持图形卡的硬件中安装 RHEL，或者解决阻止启动图形界面的图形驱动程序问题。在这个版本中，在**基本图形**模式中安装的选项已从安装程序引导菜单中删除。在不支持的硬件中使用 VNC 安装选项，或解决驱动程序错误。

有关使用 VNC 安装的详情，请参考[使用 VNC 执行远程 RHEL 安装](#)部分。

([BZ#1961092](#))

### 5.2. 订阅管理

#### **virt-who** 现在可以在 Hyper-V 主机中正常工作

在以前的版本中，当使用 **virt-who** 在 Hyper-V hypervisor 中设置 RHEL 9 虚拟机(VM)时，**virt-who** 无法正确与虚拟机监控程序通信，设置会失败。这是因为 **openssl** 软件包中的已弃用的加密方法。

在这个版本中，Hyper-V 的 **virt-who** 验证模式已被修改，并使用 **virt-who** 在 Hyper-V 中设置 RHEL 9 虚拟机现在可以正常工作。请注意，这还需要管理程序使用基本的验证模式。要启用这个模式，请使用以下命令：

```
winrm set winrm/config/service/auth '{@Basic="true"}'  
winrm set winrm/config/service '{@AllowUnencrypted="true"}
```

([BZ#2008215](#))

## 5.3. 软件管理

### 在模块化存储库上运行 `createrepo_c --update` 现在会在存储库中保留模块元数据

在以前的版本中，当在没有原始模块化元数据的现有模块存储库上运行 `createrepo_c --update` 命令时，默认策略将删除包括此存储库中模块元数据的所有额外元数据，从而导致存储库出问题。要保留元数据，需要运行带有 `--keep-all-metadata` 选项的 `createrepo_c --update` 命令。

有了此更新，您可以运行不带任何选项的 `createrepo_c --update`，来在模块化存储库上保留模块元数据。

要删除其他元数据，您可以使用新的 `--discard-additional-metadata` 选项。

([BZ#2055032](#))

## 5.4. SHELL 和命令行工具

### RHEL 9 提供 `libservicelog 1.1.19`

RHEL 9 提供 `libservicelog` 版本 1.1.19。主要程序错误修复包括：

- 修复了输出对齐问题。
- 修复了 `servicelog_open()` 失败中的 `segfault`。

([BZ#1869568](#))

## 5.5. 安全性

### 当为 FIPS 模式，会在 `libgcrypt` 中启用硬件优化

在以前的版本中，联邦信息处理标准(FIPS 140-2)不允许使用硬件优化。因此，在之前的 RHEL 版本中，当使用 FIPS 模式时，操作在 `libgcrypt` 软件包中被禁用。RHEL 9 在 FIPS 模式中启用硬件优化，因此，所有加密操作都会更快地执行。

([BZ#1990059](#))

### `crypto-policies` 现在可以禁用 ChaCha20 密码

在以前的版本中，`crypto-policies` 软件包使用错误关键字在 OpenSSL 中禁用 ChaCha20 密码。因此，您无法通过 `crypto-policies` 在 OpenSSL 中为 TLS 1.2 协议禁用 ChaCha20。在这个版本中，使用 `-CHACHA20` 关键字而不是 `-CHACHA20-POLY1305`。现在，您可以使用加密策略在 OpenSSL 中为 TLS 1.2 和 TLS 1.3 禁用 ChaCha20 密码。

([BZ#2004207](#))

### 当使用 FIPS 模式安装时，64 位 IBM Z 系统不再无法引导

在以前的版本中，带有 `--no-bootcfg` 选项的 `fips-mode-setup` 命令不会执行 `zipl` 工具。因为 `fips-mode-setup` 会重新生成初始 RAM 磁盘(`initrd`)，因此生成的系统需要更新 `zipl` 内部状态才能引导，因此在以 FIPS 模式安装时将 64 位 IBM Z 系统置于无法引导的状态。在这个版本中，即使使用了 `--no-`

**bootcfg, fips-mode-setup** 也会在 64 位 IBM Z 系统中执行 **zipl**, 新安装的系统可以引导成功。

(BZ#2013195)

### **GNUTLS\_NO\_EXPLICIT\_INIT** 不再禁用隐式库初始化

在以前的版本中, **GNUTLS\_NO\_EXPLICIT\_INIT** 环境变量禁用了隐式库初始化。在 RHEL 9 中, **GNUTLS\_NO\_IMPLICIT\_INIT** 变量禁用隐式库初始化。

(BZ#1999639)

### 基于 **openssl** 的应用程序现在可以使用 **Turkish** 区域正常工作

因为 **OpenSSL** 库使用不区分大小写的字符串比较功能, 因此基于 **OpenSSL** 的应用程序无法使用 **Turkish** 区域正常工作, 因此忽略的检查会导致应用程序使用这个区域设置崩溃。在这个版本中, 提供一个补丁以使用可移植的操作系统接口(POSIX)区域进行不区分大小写的字符串比较。因此, 基于 **OpenSSL** 的应用程序 (如 **curl**) 可以与 **Turkish** 区域正常工作。

(BZ#2071631)

### **kdump** 不再会因为 **SELinux** 权限崩溃

**kdump** 崩溃恢复服务需要额外的 **SELinux** 权限才能正确启动。因此, **SELinux** 会阻止 **kdump** 工作。**kdump** 报告它无法正常工作, 并且对 **Access Vector Cache(AVC)** 拒绝进行审核。在这个版本中, 所需权限被添加到 **selinux-policy** 中, 因此 **kdump** 可以正常工作, 且不会对 **AVC** 拒绝进行审核。

(BZ#1932752)

### **usbguard-selinux** 软件包不再依赖于 **usbguard**

在以前的版本中, **usbguard-selinux** 软件包依赖于 **usbguard** 软件包。这与这些软件包的其它依赖关系相结合, 这会导致安装 **usbguard** 时存在文件冲突。因此, 这会阻止在某些系统上安装 **usbguard**。在这个版本中, **usbguard-selinux** 不再依赖于 **usbguard**, 因此 **dnf** 可以正确安装 **usbguard**。

(BZ#1986785)

### **dnf install** 和 **dnf update** 现在可以在 **SELinux** 中使用 **fapolicyd**

**fapolicyd-selinux** 软件包包含 **fapolicyd** 的 **SELinux** 规则, 不包含监视所有文件和目录的权限。因此, **fapolicyd-dnf-plugin** 无法正常工作, 从而导致任何 **dnf install** 和 **dnf update** 命令使系统无限期停止响应。在这个版本中, 监视任何文件类型的权限被添加到 **fapolicyd-selinux** 中。因此, **fapolicyd-dnf-plugin** 可以正常工作, 并且 **dnf install** 和 **dnf update** 命令可以正常工作。

(BZ#1932225)

### 现在, **Ambient** 能力可以被正确应用到非 **root** 用户

作为一种安全措施, 将 **UID** (用户标识符) 从 **root** 更改为非 **root** 会使 **permitted**, **effective**, 和 **ambient** 能力集无效。

但是, **pam\_cap.so** 模块无法设置 **ambient** 能力, 因为能力需要同时位于 **permitted** 和 **inheritable** 集才可以处于 **ambient** 中。此外, 在更改 **UID** (例如使用 **setuid** 实用程序) 后, **permitted** 集会变为无效, 因此不能设置 **ambient** 能力。

要解决这个问题, **pam\_cap.so** 模块现在支持 **keepcaps** 选项, 该选项允许在将 **UID** 从 **root** 更改为非 **root** 后保留其允许的能力。**pam\_cap.so** 模块现在支持 **defer** 选项, 该选项会导致 **pam\_cap.so** 在回调中的 **pam\_cap.so** 中重新应用到 **pam\_end()** 的功能。更改 **UID** 后, 其他应用程序可使用此回调。

因此，如果 **su** 和 **login** 工具被更新并符合 PAM，您现在可以使用 **pam\_cap.so** 和 **keepcaps** 和 **defer** 选项为非 root 用户设置 ambient 能力。

([BZ#2037215](#))

### usbguard-notifier 不再将太多的错误消息记录到 Journal 中

在以前的版本中，**usbguard-notifier** 服务没有连接到 **usbguard-daemon** IPC 接口的进程间通信(IPC)权限。因此，**usbguard-notifier** 无法连接到接口，它会将对应的错误消息写入 Journal。由于 **usbguard-notifier** 使用 **--wait** 选项启动，因此 **usbguard-notifier** 会在连接失败后尝试每秒连接到 IPC 接口，默认情况下日志会很快包含大量与这个行为相关的消息。

在这个版本中，**usbguard-notifier** 默认不以 **--wait** 开始。现在，该服务只会以 1 秒的间隔尝试连接到守护进程三次。因此，日志包含最多三个相关的错误消息。

([BZ#2009226](#))

## 5.6. 网络

### WiFi 和 802.1x 以太网连接配置集现在被正确连接

在以前的版本中，很多 Wifi 和 802.1x 以太网连接配置集无法连接。这个程序漏洞现已解决。现在，所有配置集都可以正常工作。使用旧加密算法的配置集仍可以正常工作，但您需要手动启用 OpenSSL 旧供应商。例如，当使用带有 MS-CHAPv2 和 RC4 和 TKIP 的 RC4 时，这是必需的。

([BZ#1975718](#))

### Afterburn 不再在 /etc/hostname 中设置超长的主机名

RHEL 主机名的最大长度为 64 个字符。但是，某些云提供商使用完全限定域名(FQDN)作为主机名，最多可包含 255 个字符。在以前的版本中，**afterburn-hostname** 服务将此类主机名直接写入 **/etc/hostname** 文件。**systemd** 服务将主机名截断为 64 个字符，NetworkManager 从截断后的值派生出错误的 DNS 搜索域。在这个版本中，**afterburn-hostname** 会在第一个点或 64 个字符处（取决于哪个在前）进行截断。因此，NetworkManager 不再在 **/etc/resolv.conf** 中设置无效的 DNS 搜索域。

([BZ#2008521](#))

## 5.7. 内核

### modprobe 按预期加载树外内核模块

**/etc/depmod.d/dist.conf** 配置文件为 **depmod** 程序提供搜索顺序。根据搜索顺序，**depmod** 创建 **modules.dep.bin** 文件。此文件列出了 **modprobe** 实用程序用来加载和卸载内核模块并同时解析模块依赖项的模块依赖项。在以前的版本中，**/etc/depmod.d/dist.conf** 缺失。因此，**modprobe** 无法加载一些树外的内核模块。这个版本包括 **/etc/depmod.d/dist.conf** 配置文件，该文件修复了搜索顺序。因此，**modprobe** 会如预期加载树外内核模块。

([BZ#1985100](#))

### alsa-lib 现在可以正确处理使用 UCM 的音频设备

**alsa-lib** 软件包中的错误导致内部用例管理器(UCM)标识符错误解析。因此，有些使用 UCM 配置的音频设备没有被检测到，或者它们无法正常工作。当系统使用 **pipewire** 声音服务时，这个问题更为频繁。在 RHEL 9 的新版本中，更新 **alsa-lib** 库解决了这个问题。

([BZ#2015863](#))

## 5.8. 文件系统和存储

### 保护 uevents 不再导致多路径设备重新载入失败

在以前的版本中，当重新扫描 只读 路径设备时，内核会发送两个写保护 uevents - 一个设备被设为 读/写，以下设备被设为 只读。因此，在路径设备上检测到 读/写 uevent 时，**multipathd** 会尝试重新载入多路径设备，这会导致重新载入错误信息。有了这个更新，**multipathd** 会在重新载入读/写设备前，检查所有路径是否被设置为 读/写。因此，当重新扫描 只读 设备时，**multipathd** 不再尝试重新载入 读/写。

(BZ#2017979)

### device-mapper-multipath 被更新到版本 0.8.7

**device-mapper-multipath** 软件包已升级至 0.8.7 版本，提供多个程序错误修复和增强。主要变更包括：

- 修复了 **multipath** 和 **kpartx** 命令中的内存泄漏的问题。
- 修复了 **multipathd.socket** 单元文件中的重复触发器错误。
- 改进了更多设备的自动配置，如 DELL SC 系列数组、EMC Invista 和 Symmetrix 数组（如果是其他设备）。

(BZ#2017592)

## 5.9. 高可用性和集群

### Pacemaker 属性管理器可以正确地决定远程节点属性，阻止取消隔离循环

在以前的版本中，在节点上 Pacemaker 的控制器可能会选择 Designated Controller(DC)，然后其属性管理器发现一个已经主动的远程节点是远程的。发生这种情况时，节点的调度程序不会看到任何远程节点的节点属性。如果集群使用了 un fencing，则可能会导致无法发生循环。在这个版本中，属性管理器可以通过额外的事件来远程了解远程节点，包括启动时的初始属性同步。因此，无论哪个节点被选择使用 DC，都不会发生取消隔离循环。

(BZ#1975388)

## 5.10. 编译器和开发工具

### -Wsequence-point 警告行为修复

在以前的版本中，当使用 GCC 编译 C++ 程序时，**-Wsequence-point** 警告选项会尝试警告非常长的表达式，因此可能导致量化行为，因此会显著编译时间。在这个版本中，**-Wsequence-point** 不会试图警告非常大的表达式，因此不会增加编译时间。

(BZ#1481850)

## 5.11. 身份管理

### 使用 OpenSSL 传统供应商的 MS-CHAP 验证

在以前的版本中，使用 MS-CHAP 的 FreeRADIUS 身份验证机制会失败，因为它们依赖于 MD4 hash 功能，而 MD4 已在 RHEL 9 中弃用。在这个版本中，如果您启用了 OpenSSL 旧供应商，您可以使用 MS-CHAP 或 MS-CHAPv2 验证 FreeRADIUS 用户。

如果您使用默认 OpenSSL 供应商、MS-CHAP 和 MS-CHAPv2 验证失败，并显示以下出错信息，显示这个修复：

Couldn't init MD4 algorithm. Enable OpenSSL legacy provider.

(BZ#1978216)

### 运行 `sudo` 命令不再导出 `KRB5CCNAME` 环境变量

在以前的版本中，在运行 `sudo` 命令后，环境变量 `KRB5CCNAME` 指向原始用户的 Kerberos 凭证缓存，这些信息可能无法被目标用户访问。因此，与 Kerberos 相关的操作可能会失败，因为这个缓存无法访问。在这个版本中，运行 `sudo` 命令不再设置 `KRB5CCNAME` 环境变量，目标用户可以使用它们的默认 Kerberos 凭证缓存。

(BZ#1879869)

### SSSD 可以正确地评估 `/etc/krb5.conf` 中 Kerberos keytab 名称的默认设置

在以前的版本中，如果您为 `krb5.keytab` 文件定义了一个非标准位置，SSSD 不会使用此位置，并使用默认的 `/etc/krb5.keytab` 位置。因此，当您试图登录系统时，登录会失败，因为 `/etc/krb5.keytab` 不包含条目。

在这个版本中，SSSD 会评估 `/etc/krb5.conf` 中的 `default_keytab_name` 变量，并使用此变量指定的位置。如果未设置 `default_keytab_name` 变量，SSSD 仅使用默认的 `/etc/krb5.keytab` 位置。

(BZ#1737489)

### 现在，使用 PBKDF2 算法以 FIPS 模式验证目录服务器可以正常工作

当目录服务器在联邦信息处理标准(FIPS)模式下运行时，`K11_ExtractKeyValue ()` 函数不可用。因此，在更新之前，启用了 FIPS 模式时，使用基于密码的身份验证功能 2(PBKDF2)算法无法向服务器进行身份验证。有了这个更新，目录服务器使用 `PK11_Decrypt ()` 函数来获取密码哈希数据。因此，使用 PBKDF2 算法哈希的密码验证现在可以正常工作。

(BZ#1779685)

## 5.12. RED HAT ENTERPRISE LINUX 系统角色

### 如果 IPv6 被禁用了，则 Networking 系统角色不再无法设置 DNS 搜索域

在以前的版本中，如果禁用了 IPv6 协议，`libnm` 库的 `nm_connection_verify ()` 函数不会忽略 DNS 搜索域。因此，当您使用 Networking RHEL 系统角色并设置 `dns_search` 和 `ipv6_disabled: true` 时，系统角色失败并显示以下错误：

```
nm-connection-error-quark: ipv6.dns-search: this property is not allowed for 'method=ignore' (7)
```

有了这个更新，如果禁用了 IPv6，`nm_connection_verify ()` 函数会忽略 DNS 搜索域。因此，您可以按预期使用 `dns_search`，即使禁用了 IPv6。

(BZ#2004899)

### Postfix 角色 README 不再使用普通角色名称

在以前的版本中，`/usr/share/ansible/roles/rhel-system-roles.postfix/README.md` 中提供的示例使用了角色名称的普通版本 (`postfix`)，而不是使用 `rhel-system-roles.postfix`。因此，用户会因为参考了文档而错误地使用普通角色名称而不是完全限定域名(FQDN)。在这个版本中解决了这个问题，文档包含了 FQDN、`rhel-system-roles.postfix` 的示例，让用户能够正确地写入 playbook。

(BZ#1958964)

## Postfix RHEL 系统角色 README.md 不再在"Role Variables"部分下缺失变量

在以前的版本中，Postfix RHEL 系统角色变量（如 **postfix\_check**、**postfix\_backup**、**postfix\_backup\_multiple**）在"Role Variables"部分中不可用。因此，用户无法查阅 Postfix 角色文档。在这个版本中，在 Postfix README 部分中添加了角色变量文档。角色变量记录在 **rhel-system-roles** 提供的 **doc/usr/share/doc/rhel-system-roles/postfix/README.md** 文档中。

(BZ#1978734)

## 运行相同输出时角色任务不再改变

在以前的版本中，在再次运行同一输入时，一些角色任务会报告为 **CHANGED**，即使没有更改也是如此。因此，该角色并不具有幂等性。要解决这个问题，请执行以下操作：

- 在应用配置变量前检查配置变量是否有所变化。您可以使用 **--check** 选项进行此验证。
- 不要向配置文件中添加 **Last Modified: \$date** 标头。

因此，角色任务是幂等的。

(BZ#1978760)

## logging\_purge\_confs 选项可以正确地删除不必要的配置文件

将 **logging\_purge\_confs** 选项设为 **true** 时，它应删除不必要的日志记录配置文件。但是，即使 **logging\_purge\_confs** 被设置为 **true**，也不会从配置目录中删除不必要的配置文件。现在，这个问题已被解决，选项已被重新定义：如果 **logging\_purge\_confs** 设为 **true**，Rsyslog 从没有属于任何 rpm 软件包的 **rsyslog.d** 目录中删除文件。这包括之前运行日志记录角色的配置文件。**logging\_purge\_confs** 的默认值为 **false**。

(BZ#2039106)

## 使用 Metrics 角色的 playbook 可以在多个运行中成功完成，即使 Grafana admin 密码已更改

在以前的版本中，在使用 **metrics\_graph\_service: yes** 布尔值运行 Metrics 角色后更改 Grafana **admin** 用户密码会导致 Metrics 角色后续运行失败。这会导致使用 Metrics 角色失败 playbook，受影响的系统只部分设置性能分析。现在，Metrics 角色会在 Grafana **deployment** API 可用时使用，不再需要了解用户名或密码来执行必要的配置操作。因此，使用 Metrics 角色的 playbook 会在多个运行中成功完成，即使管理员更改了 Grafana **admin** 密码。

(BZ#2041632)

## 现在，Metrics 角色配置会正确跟随符号链接

安装 **mssql\_pcp** 软件包后，**mssql.conf** 文件位于 **/etc/pcp/mssql/** 中，是符号链接 **/var/lib/pcp/pmdas/mssql/mssql.conf** 的目标。但是，Metrics 角色覆盖符号链接，而不是按照其配置 **mssql.conf**。因此，运行 Metrics 角色会更改到常规文件的符号链接，因此配置仅影响 **/var/lib/pcp/pmdas/mssql.conf** 文件。这会导致符号链接失败，主配置文件 **/etc/pcp/mssql.conf** 不受配置的影响。这个问题现已解决，在 Metrics 角色中添加了 **follow: yes** 选项来跟踪链接。因此，Metrics 角色会保留符号链接并正确配置主配置文件。

(BZ#2058777)

## timesync 角色不再无法找到请求的服务 ptp4l

在以前的版本中，在一些 RHEL 版本中，Ansible **service\_facts** 模块报告服务事实错误。因此，**timesync** 角色会报告一个错误，试图停止 **ptp4l** 服务。在这个版本中，Ansible **service\_facts** 模块检查任务的返回值，以停止 **timesync** 服务。如果返回的值为 **failed**，但错误消息是 **Could not find the**

**requested service NAME:**, 则模块会假定成功。现在, **timesync** 角色会在没有任何错误的情况下运行, 如 **Could not find the requested service ptp4l**。

(BZ#2058645)

### kernel\_settings configobj 在受管主机上可用

在以前的版本中, **kernel\_settings** 角色没有在受管主机上安装 **python3-configobj** 软件包。因此, 角色会返回一个错误, 表示无法找到 **configobj** Python 模块。在这个版本中, 角色可确保受管主机上存在 **python3-configobj** 软件包, **kernel\_settings** 角色可以正常工作。

(BZ#2058756)

### Terminal Session Recording 角色 tlog-rec-session 现在被 SSSD 正确覆盖

在以前的版本中, Terminal Session Recording RHEL 系统角色依赖系统安全服务守护进程(SSSD)文件提供者和启用的 **authselect** 选项 **with-files-domain**, 来在 **nsswitch.conf** 文件中设置正确的 **passwd** 条目。在 RHEL 9.0 中, SSSD 默认不会隐式启用文件供应商, 因此 SSSD 的 **tlog-rec-session** shell 覆盖无法正常工作。在这个版本中, Terminal Session Recording 角色更新 **nsswitch.conf** 以确保 SSSD 正确覆盖 **tlog-rec-session**。

(BZ#2071804)

### SSHD 系统角色可以在 FIPS 模式下管理系统

在以前的版本中, SSHD 系统角色在调用时无法创建 **not allowed** 的 HostKey 类型。因此, SSHD 系统角色无法在联邦信息处理标准(FIPS)模式下管理 RHEL 8 和旧的系统。有了此更新, SSHD 系统角色检测 FIPS 模式, 并正确调整默认的 HostKey 列表。因此, 系统角色可以使用默认的 HostKey 配置在 FIPS 模式中管理 RHEL 系统。

(BZ#2029634)

### SSHD 系统角色使用正确的模板文件

在以前的版本中, SSHD 系统角色使用错误的模板文件。因此, 生成的 **sshd\_config** 文件不包含 **ansible\_managed** 注释。在这个版本中, 系统角色使用正确的模板文件, **sshd\_config** 包含正确的 **ansible\_managed** 注释。

(BZ#2044408)

### Kdump RHEL 系统角色可以重启导, 或者表示需要重启

在以前的版本中, Kdump RHEL 系统角色忽略没有为崩溃内核保留内存的受管节点。因此, 角色使用 "Success" 状态完成, 即使它没有正确配置系统。在 RHEL 9 的这个版本中, 这个问题已被解决。如果受管节点没有为崩溃内核保留任何内存, 则 Kdump RHEL 系统角色失败, 建议用户将 **kdump\_reboot\_ok** 变量设置为 **true**, 以便在受管节点上正确配置 **kdump** 服务。

(BZ#2029602)

### Networking 系统角色中的 nm 提供者现在可以正确地管理网桥

在以前的版本中, 如果您使用 **initscripts** 提供者, 则 Networking 系统角色会创建一个 **ifcfg** 文件, 该文件配置了 NetworkManager, 来将网桥接口标记为 **unmanaged**。另外, NetworkManager 无法检测后续的 **initscript** 操作。例如, 如果在 **down** 和 **absent** 操作后没有重新载入连接, 则 **initscript** 提供程序的 **down** 和 **absent** 操作不会更改 NetworkManager 对这个接口的未受管状态的理解。有了此修复, Networking 系统角色使用 **NM.Client.reload\_connections\_async ()** 函数在具有 NetworkManager 1.18 的受管主机上重新载入 NetworkManager。因此, 在将提供程序从 **initscript** 切换到 **nm** 时, NetworkManager 会管理网桥接口。

(BZ#2038957)

### 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#2064391)

### Logging 系统角色不再多次调用任务

在以前的版本中，日志记录角色会多次调用只应该调用一次的任务。因此，额外的任务调用会减慢角色执行速度。在这个版本中，日志记录角色被修改为只调用一次任务，从而提高了日志记录角色性能。

(BZ#2004303)

### RHEL 系统角色现在处理生成的文件中的多行 ansible\_managed 注释

在以前的版本中，一些 RHEL 系统角色使用 `# {{ ansible_managed }}` 生成一些文件。因此，如果客户有自定义多行 **ansible\_managed** 设置，则文件会被错误地生成。在这个版本中，所有系统角色在生成文件时都使用了与 `{{ ansible_managed | comment }}` 等效的，以便 **ansible\_managed** 字符串始终被正确注释，包括多行 **ansible\_managed** 值。因此，生成的文件具有正确的多行 **ansible\_managed** 值。

(BZ#2006230)

### 现在，当 target 更改时，Firewall 系统角色现在立即重新载入防火墙

在以前的版本中，当 **target** 参数已改变时，Firewall 系统角色不重新载入防火墙。在这个版本中，防火墙角色会在 **target** 更改时重新载入防火墙，因此 **target** 更改将立即可用，供后续操作使用。

(BZ#2057164)

### Certificate 系统角色中的 group 选项不再使组无法访问证书

在以前的版本中，当为证书设置组时，**mode** 不会设置为允许组读取权限。因此，组成员无法读取证书角色发布的证书。在这个版本中，组设置可确保文件模式包含组读取权限。因此，组成员可以访问证书角色为组发布的证书。

(BZ#2021025)

### Logging 角色不再缺少 immark 模块间隔值的引号

在以前的版本中，**immark** 模块的 **interval** 字段值没有正确加引号，因为 **immark** 模块没有正确配置。在这个版本中，**interval** 值会被正确加引号。现在，**immark** 模块可以正常工作。

(BZ#2021676)

### /etc/tuned/kernel\_settings/tuned.conf 文件有一个正确的 ansible\_managed 标头

在以前的版本中，**kernel\_settings** RHEL 系统角色对 `/etc/tuned/kernel_settings/tuned.conf` 文件中的 **ansible\_managed** 标头有一个硬编码的值。因此，用户无法提供自己的自定义 **ansible\_managed** 标头。在这个版本中，这个问题已被解决，**kernel\_settings** 使用用户的 **ansible\_managed** 设置更新了 `/etc/tuned/kernel_settings/tuned.conf` 的标头。因此，`/etc/tuned/kernel_settings/tuned.conf` 有一个适当的 **ansible\_managed** 标头。

(BZ#2047506)

### VPN 系统角色过滤插件 vpn\_ipaddr 现在转换为 FQCN（完全限定集合名）

在以前的版本中，从旧角色格式转换为集合格式的转变没有将过滤器插件 `vpn_ipaddr` 转换为 FQCN(Fully Qualified Collection Name) `redhat.rhel_system_roles.vpn_ipaddr`。因此，VPN 角色无法通过短名称查找插件并报告错误。在这个版本中，转换脚本已被修改，过滤器会在集合中转换为 FQCN 格式。现在，VPN 角色在不发出错误的情况下运行。

(BZ#2050341)

### **kdump.service 的作业不再失败**

在以前的版本中，用来配置内核崩溃大小的 `Kdump` 角色代码没有为 RHEL9 更新，这需要使用 `kdumpctl reset-crashkernel`。因此，`kdump.service` 无法启动并发出错误。在这个版本中，`kdump.service` 角色使用 `kdumpctl reset-crashkernel` 来配置崩溃内核大小。现在，`kdump.service` 角色可以成功启动 `kdump` 服务，内核崩溃大小会被正确配置。

(BZ#2050419)

## 5.13. 虚拟化

### **热拔挂载的虚拟磁盘不再会导致客户端内核在 IBM Z 上崩溃**

在以前的版本中，当将挂载的磁盘从 IBM Z 硬件的正在运行的虚拟机(VM)中分离时，VM 内核会在以下情况下崩溃：

- 这个磁盘已附加有目标总线类型 `scsi` 并挂载到客户机中。
- 热拔磁盘设备后，相应的 SCSI 控制器也被热拔。

在这个版本中，底层代码已被修复，上面描述的崩溃不再发生。

(BZ#1997541)

## 5.14. 容器

### **UBI 9-Beta 容器可以在 RHEL 7 和 8 主机上运行**

在以前的版本中，UBI 9-Beta 容器镜像在 `containers-common` 软件包中设置了不正确的 `seccomp` 配置集。因此，容器无法处理导致失败的一些系统调用。在这个版本中，这个问题已被解决。

(BZ#2019901)

## 第 6 章 技术预览

这部分列出了 Red Hat Enterprise Linux 9 中的所有技术预览。

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](#)。

### 6.1. RHEL FOR EDGE

#### FDO 流程作为技术预览提供

FDO 过程可用于自动置备和加入 RHEL for Edge 镜像，作为技术预览提供。在这个版本中，您可以构建 RHEL for Edge Simplified Installer 镜像，将其提供给 RHEL for Edge 镜像，并使用 FDO (FIDO 设备装载) 进程自动提供和装载边缘设备，与网络上连接的其他设备和系统交换数据。因此，FIDO 设备加载协议在制造阶段执行设备初始化，然后进行后期绑定以实际使用该设备。

(BZ#1989930)

### 6.2. 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 环境中在 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 生成内核和初始 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#2046653)

#### GIMP 在 RHEL 9 中作为技术预览提供

GNU Image Manipulation Program(GIMP)2.99.8 现在作为技术预览在 RHEL 9 中提供。**gimp** 软件包版本 2.99.8 是一个预发行版本，它有一组改进，但只能保证稳定性。发布官方 GIMP 3 后，将作为此预发布版本的更新，在 RHEL 9 中引入。

在 RHEL 9 中，您可以作为 RPM 软件包轻松安装 **gimp**。

(BZ#2047161)

### 6.3. 网络

#### WireGuard VPN 作为技术预览提供

WireGuard（红帽作为技术预览提供）是一个在 Linux 内核中运行的高性能 VPN 解决方案。它使用现代加密，比其他 VPN 解决方案更容易配置。此外，因为 WireGuard 较小的代码基础，减少了受攻击的风险，因此提高了安全性。

详情请查看[设置 WireGuard VPN](#)。

(BZ#1613522)

### 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)

### systemd-resolved 服务作为技术预览提供

**systemd-resolved** 服务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 stub 解析器、链接本地多播名称解析 (LLMNR) 以及多播 DNS 解析器和响应程序。

请注意, **systemd-resolved** 是一个不受支持的技术预览。

(BZ#2020529)

## 6.4. 内核

### 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 集成的加速器，包括共享工作队列 ID(pasid)提交和共享虚拟内存(SVM)。

(BZ#2030412)

### 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 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#1874182)

### Soft-iWARP 驱动程序作为技术预览提供

软硬件硬件(siw)是一种软件，互联网是 RDMA 协议(iWARP)，适用于 Linux 的内核驱动程序。soft-iWARP 通过 TCP/IP 网络堆栈实施 iWARP 协议套件。这个协议套件在软件中完全实现，不需要特定的远程直接内存访问(RDMA)硬件。Soft-iWARP 使具有标准以太网适配器的系统连接到 iWARP 适配器或安装了 Soft-iWARP 的其他系统。

(BZ#2023416)

## 6.5. 文件系统和存储

### DAX 现在作为技术预览供 ext4 和 XFS 使用

在 RHEL 9 中，DAX 文件系统作为技术预览提供。DAX 提供了将持久内存直接映射到其地址空间的方法。要使用 DAX，系统必须有某种可用的持久性内存，通常使用一个或多个非线性内存模块(NVDIMM)，必须在 NVDIMM 上创建 DAX 兼容文件系统。另外，该文件系统必须使用 **dax** 挂载选项挂载。然后，在 **dax** 挂载的文件系统中的文件 **mmap** 会把存储直接映射到应用程序的地址空间中。

(BZ#1995338)

### Stratis 作为技术预览提供

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

- 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](#)。

(BZ#2041558)

### NVMe-oF Discovery Service 功能作为技术预览

NVMe-oF Discovery Service 功能（在 NVMeexpress.org 技术 Proposals(TP)8013 和 8014 中）作为技术预览提供。要预览这些功能，请使用 **nvme-cli 2.0** 软件包，并将主机附加到实现 TP-8013 或 TP-8014 的 NVMe-oF 目标设备。有关 TP-8013 和 TP-8014 的更多信息，请参阅 <https://nvmeexpress.org/developers/nvme-specification/> 网站中的 NVM Express 2.0 Ratified TPs。

(BZ#2021672)

## 6.6. 编译器和开发工具

### jmc-core 和 owasp-java-encoder 作为技术预览

RHEL 9 带有 **jmc-core** 和 **owasp-java-encoder** 软件包作为技术预览功能。

**JMC-core** 是一个为 Java Development Kit(JDK)Mission Control 提供核心 API 的库，包括用于解析和编写 JDK Flight Recording 文件的库，以及通过 Java 发现协议(JDP)发现的 Java 虚拟机(JVM)发现库。

**owasp-java-encoder** 软件包提供了 Java 的高性能低后台上下文组。

(BZ#1980981)

## 6.7. 身份管理

### 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 协议安全性的 DNS 扩展。托管在 IdM 服务器上的 DNS 区可以使用 DNSSEC 自动签名。加密密钥是自动生成和轮转的。

建议那些决定使用 DNSSEC 保护 DNS 区的用户读取并遵循这些文档：

- [DNSSEC 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#2084180](#))

## 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#2084166](#))

## 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#2084181)

## 6.8. DESKTOP

### GNOME 用于 64 位 ARM 架构，作为一个技术预览

GNOME 桌面环境可用于 64 位 ARM 架构，作为技术预览。

现在，您可以使用 VNC 连接到 64 位 ARM 服务器上的桌面会话。因此，您可以使用图形应用程序管理服务。

64 位 ARM 提供了有限的图形应用程序集合。例如：

- Firefox Web 浏览器
- Red Hat 订阅管理器 (**subscription-manager-cockpit**)
- 防火墙配置(**firewall-config**)
- 磁盘用量分析器(**baobab**)

使用 Firefox，您可以连接到服务器上的 Cockpit 服务。

某些应用程序，如 LibreOffice，只提供命令行界面，其图形界面被禁用。

(JIRA:RHELPLAN-27394)

### 用于 IBM Z 架构的 GNOME 作为技术预览提供

对于 IBM Z 架构，GNOME 桌面环境作为技术预览。

现在，您可以使用 VNC 连接到 IBM Z 服务器上的桌面会话。因此，您可以使用图形应用程序管理服务。

IBM Z 上提供了一组有限的图形应用程序。例如：

- Firefox Web 浏览器
- Red Hat 订阅管理器 (**subscription-manager-cockpit**)
- 防火墙配置(**firewall-config**)
- 磁盘用量分析器(**baobab**)

使用 Firefox，您可以连接到服务器上的 Cockpit 服务。

某些应用程序，如 LibreOffice，只提供命令行界面，其图形界面被禁用。

(JIRA:RHELPLAN-27737)

## 6.9. WEB 控制台

### Stratis 在 RHEL web 控制台中作为技术预览提供

有了这个更新，Red Hat Enterprise Linux web 控制台提供了将 Stratis 存储作为技术预览管理的能力。

要了解更多有关 Stratis 的信息，请参阅 [什么是 Stratis](#)。

(JIRA:RHELPLAN-122345)

## 6.10. 虚拟化

### 创建嵌套虚拟机

对于运行在 Intel、AMD64 和 IBM Z 主机上的 RHEL 9 KVM 虚拟机，嵌套的 KVM 虚拟化作为技术预览提供。有了这个功能，运行在物理 RHEL 9 主机上的 RHEL 7、RHEL 8 或 RHEL 9 虚拟机可以充当 hypervisor，并托管自己的虚拟机。

(JIRA:RHELDPCS-17040)

### 用于 KVM 虚拟机的 AMD SEV 和 SEV-ES

作为技术预览，RHEL 9 为使用 KVM 管理程序的 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 9 包括 SEV 和 SEV-ES 加密，但不包括 SEV 和 SEV-ES 安全测试。

(JIRA:RHELPLAN-65217)

### 虚拟化现在在 ARM 64 上可用

作为技术预览，现在可以使用 ARM 64 CPU 在系统中创建 KVM 虚拟机。

(JIRA:RHELPLAN-103993)

### virtio-mem 现在包括在 AMD64 和 Intel 64 中

作为技术预览，RHEL 9 在 AMD64 和 Intel 64 系统中引入了 **virtio-mem** 功能。使用 **virtio-mem** 可让虚拟机(VM)动态添加或删除主机内存。

要使用 **virtio-mem**，请在虚拟机 XML 配置中定义 **virtio-mem** 内存设备，并使用 **virsh update-memory-device** 命令请求 VM 运行期间内存设备大小更改。要查看此类内存设备向正在运行的虚拟机公开的当前内存大小，请查看虚拟机的 XML 配置。

(BZ#2014487)

### Intel vGPU 作为技术预览提供

---

作为技术预览，可以将物理 Intel GPU 设备划分为多个虚拟设备，称为 **介质设备**。然后可将这些介质设备分配给多个虚拟机(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，这个功能已弃用，并将在以后的 RHEL 发行版本中完全删除。

(JIRA:RHELDPCS-17050)

## 6.11. 容器

### **podman-machine 命令不受支持**

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

(JIRA:RHELDPCS-16861)

## 第 7 章 过时的功能

这部分提供在 Red Hat Enterprise Linux 9 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 9 中，已弃用的功能的支持状态不会改变。有关支持长度的详情，请查看 [Red Hat Enterprise Linux 生命周期](#) 和 [Red Hat Enterprise Linux 应用程序流生命周期](#)。

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

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

有关 RHEL 8 中存在但已在 RHEL 9 中删除的功能的信息，请参阅 [使用 RHEL 9 的注意事项](#)。

### 7.1. 安装程序和镜像创建

#### 弃用的 Kickstart 命令

以下 Kickstart 命令已弃用：

- **timezone --ntpservers**
- **timezone --nontp**
- **logging --level**
- **%packages --excludeWeakdeps**
- **%packages --instLangs**
- **%anaconda**
- **pwpolicy**

请注意，当只列出具体的选项时，基础命令及其其它选项仍可用且未被弃用。在 Kickstart 文件中使用已弃用的命令会在日志中显示警告信息。您可以使用 **inst.ksstrict** 引导选项将已弃用的命令警告转换为错误。

(BZ#1899167)

### 7.2. SHELL 和命令行工具

#### 在 ReaR 配置文件中设置 TMPDIR 变量已弃用

使用语句（如 **export TMPDIR=...**）在 **/etc/rear/local.conf** 或 **/etc/rear/site.conf** ReaR 配置文件中设置 **TMPDIR** 环境变量，无法正常工作，且已弃用。

要为 ReaR 临时文件指定一个自定义目录，请在执行 ReaR 前在 shell 环境中导出变量。例如，执行 **export TMPDIR=...** 语句，然后在同一 shell 会话或脚本中执行 **rear** 命令。

[Jira:RHELDOCS-18049](#)

## 7.3. 安全性

### 对于加密目的，SHA-1 已被弃用

使用 SHA-1 消息摘要用于加密目的在 RHEL 9 中已被弃用。SHA-1 生成的摘要不被视为是安全的，因为已发现多个基于哈希进行的安全攻击。RHEL 核心加密组件不再默认使用 SHA-1 创建签名。RHEL 9 中的应用程序已更新，以避免在与安全相关的用例中使用 SHA-1。

其中一个例外是，仍然可以使用 SHA-1 创建 HMAC-SHA1 消息验证代码和 Universal Unique Identifier(UUID)值，因为这些用例目前不存在安全风险。另外，为了保持一些重要的互操作性和兼容性，SHA-1 还会在一些有限的情况下使用，例如 Kerberos 和 WPA-2。详情请查看 [RHEL 9 安全强化文档中的使用与 FIPS 140-3 不兼容的加密系统的 RHEL 应用程序列表](#)。

如果您需要使用 SHA-1 来验证现有或第三方加密签名，您可以输入以下命令启用它：

```
# update-crypto-policies --set DEFAULT:SHA1
```

或者，您可以将系统范围的加密策略切换到 **LEGACY** 策略。请注意，**LEGACY** 也启用了其他一些不安全的算法。

(JIRA:RHELPLAN-110763)

### 在 RHEL 9 中弃用 SCP

安全复制协议(SCP)已弃用，因为它有已知的安全漏洞。SCP API 仍可用于 RHEL 9 生命周期，但使用它可以降低系统安全性。

- 在 **scp** 实用程序中，默认情况下，SCP 被 SSH 文件传输协议(SFTP)替代。
- OpenSSH 套件在 RHEL 9 中不使用 SCP。
- SCP 在 **libssh** 库中已弃用。

(JIRA:RHELPLAN-99136)

### SASL 中的 digest-MD5 已被弃用

Simple Authentication Security Layer(SASL)框架中的 Digest-MD5 身份验证机制已弃用，并可能在以后的主发行版本中从 **cyrus-sasl** 软件包中删除。

(BZ#1995600)

### OpenSSL 弃用 MD2, MD4, MDC2, Whirlpool, RIPEMD160, Blowfish, CAST, DES, IDEA, RC2, RC4, RC5, SEED, and PBKDF1

OpenSSL 项目已弃用了一组加密算法，因为它们不安全，不常用，或两者都不安全。红帽还不建议使用这些算法，RHEL 9 则为其提供迁移加密数据以使用新的算法。对于系统的安全性，用户不得依赖于这些算法。

在 OpenSSL 中，以下算法的实现已移至旧的供应商：MD2, MD4, MDC2, Whirlpool, RIPEMD160, Blowfish, CAST, DES, IDEA, RC2, RC4, RC5, SEED, 和 PBKDF1。

有关如何载入旧供应商的说明，请参阅 `/etc/pki/tls/openssl.cnf` 配置文件，并启用对已弃用算法的支持。

(BZ#1975836)

### `/etc/system-fips` 现已弃用

支持通过 `/etc/system-fips` 文件指定 FIPS 模式，该文件将不会包含在将来的 RHEL 版本中。要在 FIPS 模式中安装 RHEL，请在系统安装过程中将 `fips=1` 参数添加到内核命令行。您可以使用 `fips-mode-setup --check` 命令检查 RHEL 是否以 FIPS 模式运行。

(JIRA:RHELPLAN-103232)

### libcrypt.so.1 现已弃用

`libcrypt.so.1` 库现已弃用，它可能会在以后的 RHEL 版本中删除。

(BZ#2034569)

### 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#2054740)

## 7.4. 网络

### ipset 和 iptables-nft 已被弃用

RHEL 中弃用了 `ipset` 和 `iptables-nft` 软件包。`iptables-nft` 软件包包含不同的工具，如 `iptables`、`ip6tables`、`ebtables` 和 `arptables`。这些工具将不再获得新功能，我们不建议将其用于新部署。作为替换，最好使用 `nftables` 软件包提供的 `nft` 命令行工具。现有设置应该会在可能的情况下迁移到 `nft`。

当您载入 `iptables`、`ip6tables`、`ebtables`、`arptables`、`nft_compat`，或 `ipset` 模块时，模块会将以下警告记录到 `/var/log/messages` 文件：

```
Warning: <module_name> - this driver is not recommended for new deployments. It continues to be supported in this RHEL release, but it is likely to be removed in the next major release. Driver updates and fixes will be limited to critical issues. Please contact Red Hat Support for additional information.
```

有关迁移到 `nftables` 的更多信息，请参阅[从 iptables 迁移到 nftables](#)，以及 `iptables-translate(8)` 和 `ip6tables-translate(8)` man page。

(BZ#1945151)

### RHEL 9 中已弃用网络团队 (Network teams)

`teamd` 服务和 `libteam` 库在 Red Hat Enterprise Linux 9 中已弃用，并将在下一个主发行版本中删除。作为替换，配置绑定而不是网络组。

红帽注重于基于内核的绑定操作，以避免维护具有类似功能的两个功能：绑定和团队 (team)。绑定代码具有较高的客户采用率，非常可靠，具有活跃的社区开发。因此，绑定代码会收到功能增强和更新。

有关如何将团队迁移到绑定的详情，请参阅[将网络组配置迁移到网络绑定](#)。

(BZ#1935544)

### ifcfg 格式的 NetworkManager 连接配置文件已弃用

在 RHEL 9.0 及更高版本中，**ifcfg** 格式的连接配置文件已弃用。下一个主要 RHEL 发行版本将删除对这个格式的支持。但是，在 RHEL 9 中，如果修改了配置文件，NetworkManager 仍然会使用这个格式处理和更新现有的配置文件。

默认情况下，NetworkManager 现在在 `/etc/NetworkManager/system-connections/` 目录中以 **keyfile** 格式存储连接配置文件。与 **ifcfg** 格式不同，**keyfile** 格式支持 NetworkManager 提供的所有连接设置。有关 **keyfile** 格式以及如何迁移配置文件的详情，请参考 [keyfile 格式的 NetworkManager 连接配置文件](#)。

(BZ#1894877)

### firewalld 中的 iptables 后端已弃用

在 RHEL 9 中，**iptables** 框架已弃用。因此，**firewalld** 中的 **iptables** 后端和 **direct interface** 也被弃用。您可以使用 **firewalld** 中的原生功能，而不是 **direct interface** 来配置所需的规则。

(BZ#2089200)

## 7.5. 内核

### 在 RHEL 9 中弃用 ATM 封装

异步传输模式(ATM)封装为 ATM Adaptation Layer 5(AAL-5)提供第 2 层 (Point-to-Point 协议、以太网) 或第 3 层 (IP) 连接。从 RHEL 7 开始，红帽尚未为 ATM NIC 驱动程序提供支持。RHEL 9 中丢弃对 ATM 实施的支持。这些协议目前仅在芯片组中使用，该协议支持 ADSL 技术，并由制造商逐步淘汰。因此，Red Hat Enterprise Linux 9 中已弃用 ATM 封装。

如需更多信息，请参阅 [PPP Over AAL5](#), [Multiprotocol Encapsulation over ATM Adaptation Layer 5](#), 和 [Classical IP and ARP over ATM](#)。

(BZ#2058153)

### v4l/dvb 电视和视频捕获设备不再被支持

在 RHEL 9 中，红帽不再支持 **Video4Linux (v4l)**和 **Linux DVB (DVB)**设备，这些设备由各种电视调节卡和各种视频捕获卡组成，红帽不再提供相关的驱动程序。

(BZ#2074598)

## 7.6. 文件系统和存储

### lvm2-activation-generator 及其生成的服务在 RHEL 9.0 中删除

**lvm2-activation-generator** 程序及其生成的服务 **lvm2-activation**、**lvm2-activation-early**、**lvm2-activation-net** 已在 RHEL 9.0 中删除。**lvm.conf event\_activation** 设置用于激活服务将不再起作用。自动激活卷组的唯一方法是基于事件激活。

(BZ#2038183)

## 7.7. 动态编程语言、网页和数据库服务器

### libdb 已被弃用

RHEL 8 和 RHEL 9 目前提供 Berkeley DB(**libdb**)版本 5.3.28，该版本根据 LGPLv2 许可证发布。上游 Berkeley DB 版本 6 在 AGPLv3 许可证下提供，该许可证更严格。

从 RHEL 9 开始，**libdb** 软件包已弃用，可能不会在以后的 RHEL 版本中可用。

另外，在 RHEL 9 中，加密算法已从 **libdb** 中删除，从 RHEL 9 中删除了多个 **libdb** 依赖项。

建议 **libdb** 用户迁移到其他键值数据库。如需更多信息，请参阅 [RHEL 中已弃用的 Berkeley DB\(libdb\)](#) 的 知识库文章。

(BZ#1927780, [BZ#1974657](#), JIRA:RHELPLAN-80695)

## 7.8. 身份管理

### OpenDNSSec 中的 SHA-1 现已弃用

OpenDNSSEC 支持使用 **SHA-1** 算法导出数字签名和身份验证记录。不再支持使用 **SHA-1** 算法。在 RHEL 9 发行版本中，OpenDNSSec 中的 **SHA-1** 已被弃用，并可能在以后的次版本中删除。另外，OpenDNSSec 支持仅限于与红帽身份管理的集成。OpenDNSSEC 不支持独立。

([BZ#1979521](#))

### SSSD 隐式文件供应商域默认禁用

SSSD 隐式 **文件** 供应商域，从 **/etc/shadow** 和 **/etc/** groups 等本地文件检索用户信息，现已默认禁用。

使用 SSSD 从本地文件检索用户和组信息：

1. 配置 SSSD.选择以下选项之一：

- a. 使用 **sssd.conf** 配置文件中的 **id\_provider=files** 选项明确配置本地域。

```
[domain/local]
id_provider=files
...
```

- b. 通过在 **sssd.conf** 配置文件中设置 **enable\_files\_domain=true** 来启用 **文件** 供应商。

```
[sssd]
enable_files_domain = true
```

2. 配置名称服务切换。

```
# authselect enable-feature with-files-provider
```

(JIRA:RHELPLAN-100639)

### SMB1 协议在 Samba 中被弃用

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

为提高安全性，在 Samba 服务器和客户端工具中默认禁用 SMB1。

Jira:RHELDPCS-16612

## 7.9. 图形基础结构

### x.org Server 现已弃用

X.org 显示服务器已弃用，并将在以后的主 RHEL 发行版本中删除。现在，在大多数情形中，默认桌面会话都是 **Wayland** 会话。

X11 协议仍完全支持使用 XWayland 后端。因此，需要 X11 的应用程序可以在 Wayland 会话中运行。

红帽正在努力解决 Wayland 会话中的剩余问题。有关 Wayland 中的未解决的问题，请参阅 [已知问题部分](#)。

您可以将用户会话切回到 X.org 后端。如需更多信息，请参阅 [选择 GNOME 环境和显示协议](#)。

(JIRA:RHELPLAN-121048)

### Motif 已被弃用

Motif 小部件工具包已在 RHEL 中弃用，因为上游 Motif 社区中的开发不活跃。

以下 Motif 软件包已弃用，包括其开发和调试变体：

- **motif**
- **openmotif**
- **openmotif21**
- **openmotif22**

另外，删除了 **motif-static** 软件包。

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)

## 7.10. RED HAT ENTERPRISE LINUX 系统角色

在 RHEL 9 节点上配置 team 时，**networking** 系统角色显示一条弃用的警告

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，会显示一条有关其弃用的警告。

([BZ#1999770](#))

## 7.11. 虚拟化

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](#))

对虚拟机快照的支持有限

目前，只有使用 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 监控可能会变得被阻止，这会影响某些工作负载的虚拟机监控程序性能。

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。但是，一个新的虚拟机快照机制正在开发中，计划在以后的 RHEL 9 次要发行本中完全实施。

(JIRA:RHELPLAN-15509, [BZ#1621944](#))

## 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)

## libvirtd 已被弃用

单体 **libvirt** 守护进程 **libvirtd** 已在 RHEL 9 中弃用，并将在以后的 RHEL 主发行版本中删除。请注意，您仍然可以使用 **libvirtd** 在虚拟机监控程序上管理虚拟化，但红帽建议您切换到新引入的模块化 **libvirt** 守护进程。具体说明和详情，请参阅 [RHEL 9 配置和管理虚拟化](#) 文档。

(JIRA:RHELPLAN-113995)

## 虚拟软盘驱动程序已弃用

用于控制虚拟软盘设备的 **isa-fdc** 驱动程序现已弃用，并将在以后的 RHEL 发行版本中不被支持。因此，为了确保与迁移的虚拟机(VM)兼容，红帽不建议在 RHEL 9 上托管的虚拟机中使用软盘磁盘设备。

(BZ#1965079)

## qcow2-v2 镜像格式已弃用

在 RHEL 9 中，虚拟磁盘镜像的 qcow2-v2 格式已弃用，并将在以后的 RHEL 主发行版本中不被支持。另外，RHEL 9 Image Builder 无法以 qcow2-v2 格式创建磁盘镜像。

红帽强烈建议您使用 qcow2-v3，而不是 qcow2-v2。要将 qcow2-v2 镜像转换为更新的格式版本，请使用 **qemu-img amend** 命令。

(BZ#1951814)

## 7.12. 容器

### 不支持在 RHEL 7 主机上运行 RHEL 9 容器

不支持在 RHEL 7 主机上运行 RHEL 9 容器。它可能可以正常工作，但却没有保证。

如需更多信息，请参阅 [Red Hat Enterprise Linux Container Compatibility Matrix](#) 。

(JIRA:RHELPLAN-100087)

### Podman 中的 SHA1 哈希算法已弃用

Podman 不再支持用于生成无根网络命名空间文件名的 SHA1 算法。因此，在从 [RHBA-2022:5951](#) 公告更新到 Podman 4.1.1 之前启动的无根容器，如果它们加入到网络（不仅仅使用 **slirp4netns**），则必须重启，以确保升级后可以连接到启动的容器。

(BZ#2069279)

### rhel9/pause 已被弃用

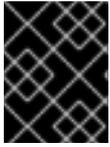
**rhel9/pause** 容器镜像已被弃用。

(BZ#2106816)

## 7.13. 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 8 和 RHEL 9 之间软件包的更改，请参阅 [使用 RHEL 9 文档中的软件包的更改](#)。



### 重要

在 RHEL 9 中，已弃用软件包的支持状态保持不变。有关支持长度的更多信息，请参阅 [Red Hat Enterprise Linux 生命周期](#) 和 [Red Hat Enterprise Linux 应用程序流生命周期](#)。

RHEL 9 中已弃用以下软件包：

- iptables-devel
- iptables-libs
- iptables-nft
- iptables-nft-services
- iptables-utils
- libdb
- mcpp
- python3-pytz

## 第 8 章 已知问题

这部分论述了 Red Hat Enterprise Linux 9.0 中已知的问题。

### 8.1. 安装程序和镜像创建

#### 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)

#### 当使用第三方工具创建的 USB 引导安装时，不会检测到本地介质安装源

当从使用第三方工具创建的 USB 引导 RHEL 安装时，安装程序无法检测 **Local Media** 安装源（只检测到 *Red Hat CDN*）。

出现这个问题的原因是，默认的引导选项 **inst.stage2=** 会尝试搜索 **iso9660** 镜像格式。但是，第三方工具可能会创建具有不同格式的 ISO 镜像。

作为临时解决方案，请使用以下解决方案之一：

- 当引导安装时，点 **Tab** 键编辑内核命令行，并将引导选项 **inst.stage2=** 改为 **inst.repo=**。
- 要在 Windows 中创建可引导 USB 设备，使用 Fedora Media Writer。
- 使用 Rufus 等第三方工具创建可引导 USB 设备时，首先在 Linux 系统上重新生成 RHEL ISO 镜像，然后使用第三方工具创建可引导 USB 设备。

有关执行任何指定临时解决方案的步骤的更多信息，请参阅[安装 RHEL 8.3 过程中不会自动探测到安装介质](#)

(BZ#1877697)

#### 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)

#### 在 Anaconda 作为应用程序运行的系统中意外 SELinux 策略

当在已安装的系统中作为应用程序运行 Anaconda（例如，使用 **-image anaconda** 选项对镜像文件执行另一个安装）时，系统将无法在安装过程中修改 SELinux 类型和属性。因此，某些 SELinux 策略元素可能会更改运行 Anaconda 的系统。要临时解决这个问题，请不要在生产环境系统上运行 Anaconda，并在临时虚拟机中执行它。因此，生产系统上的 SELinux 策略不会被修改。作为系统安装过程的一部分，运行 **anaconda**，如从 **boot.iso** 或 **dvd.iso** 安装不受此问题的影响。

(BZ#2050140)

## 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](#))

## 最小 RHEL 安装不再包含 `s390utils-base` 软件包

在 RHEL 8.4 及之后的版本中，`s390utils-base` 软件包被分成 `s390utils-core` 软件包，以及一个辅助 `s390utils-base` 软件包。因此，将 RHEL 安装设置为 `minimal-environment` 只会安装必要的 `s390utils-core` 软件包，而不是辅助 `s390utils-base` 软件包。要临时解决这个问题，请在完成 RHEL 安装后手动安装 `s390utils-base` 软件包，或使用 kickstart 文件明确安装 `s390utils-base`。

([BZ#1932480](#))

## 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。

备注：在执行临时解决方案前，备份磁盘上可用的数据。`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](#))

## Anaconda 无法验证管理员用户帐户是否存在

在使用图形用户界面安装 RHEL 时，Anaconda 无法验证管理员帐户是否已创建。因此，用户可以在没有管理员用户帐户的情况下安装系统。

要临时解决这个问题，请确保配置管理员用户帐户或 root 密码已设置，且 root 帐户被解锁。因此，用户可以在安装的系统中执行管理任务。

([BZ#2047713](#))

## 在出现不成功的 CHAP 验证尝试后，Anaconda 无法使用 `no authentication` 方法登录 iSCSI 服务器

当您使用 CHAP 身份验证添加 iSCSI 磁盘时，如果因为凭证不正确而导致登录失败，使用 `no authentication` 方法尝试重新登录也将失败。要解决这个问题，请先关闭当前会话，再使用 `no authentication` 方法登录。

([BZ#1983602](#))

## 新的 XFS 功能可防止使用比版本 5.10 更早的固件引导 PowerNV IBM POWER 系统

PowerNV IBM POWER 系统使用 Linux 内核进行固件，并使用 Petitboot 作为 GRUB 的替代。这会导致固件内核挂载 **/boot**，Petitboot 读取 GRUB 配置和引导 RHEL。

RHEL 9 内核为 XFS 文件系统引入了 **bigtime=1** 和 **inobtcount=1** 功能，而使用比版本 5.10 旧固件的内核不理解。

要临时解决这个问题，您可以为 **/boot** 使用另一个文件系统，例如 **ext4**。

(BZ#1997832)

### 当 PReP 大小为 4 或 8 MiB 时，无法安装 RHEL

如果 PowerPC Reference Platform(PReP)分区与使用 4 kiB 扇区的磁盘上的 4 MiB 或 8 MiB 不同，RHEL 安装程序无法安装引导装载程序。因此，您无法在磁盘中安装 RHEL。

要临时解决这个问题，请确保 PReP 分区的大小为 4 MiB 或 8 MiB，且大小没有舍入到另一个值。现在，安装程序可以在磁盘中安装 RHEL。

(BZ#2026579)

### 新的 XFS 功能防止具有比版本 5.10 更旧的固件内核的 PowerNV IBM POWER 系统启动

PowerNV IBM POWER 系统使用 Linux 内核进行固件，并使用 Petitboot 作为 GRUB 的替代。这会导致固件内核挂载 **/boot**，Petitboot 读取 GRUB 配置和引导 RHEL。

RHEL 9 内核向 XFS 文件系统中引入了 **bigtime=1** 和 **inobtcount=1** 功能，具有早于版本 5.10 固件的系统不会理解。因此，Anaconda 会阻止安装，并报以下错误信息：

您的固件不支持 **/boot** 文件系统上的 XFS 文件系统功能。系统将无法启动。请升级固件或更改文件系统类型。

作为临时解决方案，请为 **/boot** 使用其它文件系统，例如 **ext4**。

(BZ#2008792)

## 8.2. 订阅管理

### 在 FIPS 模式中时，**virt-who** 无法连接到 ESX 服务器

当在 RHEL 9 系统中使用 **virt-who** 程序时，**virt-who** 无法连接到 ESX 服务器。因此，**virt-who** 不会报告任何 ESX 服务器，即使为它们配置，并记录以下出错信息：

```
ValueError: [digital envelope routines] unsupported
```

要临时解决这个问题，请执行以下操作之一：

- 不要将您用来运行 **virt-who** 的 RHEL 9 系统设置为 FIPS 模式。
- 请勿将您用于运行 **virt-who** 的 RHEL 系统升级到 9.0 版本。

(BZ#2054504)

## 8.3. 软件管理

### 安装过程有时将变为无响应

安装 RHEL 时，安装过程有时会变得无响应。**/tmp/packaging.log** 文件在末尾显示以下消息：

```
10:20:56,416 DDEBUG dnf: RPM transaction over.
```

要解决这个问题，重启安装过程。

([BZ#2073510](#))

## 8.4. SHELL 和命令行工具

如果在配置文件中设置了 **TMPDIR** 变量，则在恢复过程中，**ReaR** 失败

在 `/etc/rear/local.conf` 或 `/etc/rear/site.conf` **ReaR** 配置文件中设置和导出 **TMPDIR** 无法工作，且已弃用。

**ReaR** 默认配置文件 `/usr/share/rear/conf/default.conf` 包含以下说明：

```
# To have a specific working area directory prefix for Relax-and-Recover
# specify in /etc/rear/local.conf something like
#
# export TMPDIR="/prefix/for/rear/working/directory"
#
# where /prefix/for/rear/working/directory must already exist.
# This is useful for example when there is not sufficient free space
# in /tmp or $TMPDIR for the ISO image or even the backup archive.
```

上述说明无法正常工作，因为 **TMPDIR** 变量在救援环境中具有相同的值，如果 **TMPDIR** 变量中指定的目录在救援镜像中不存在，则这是不正确的。

因此，在 `/etc/rear/local.conf` 文件中设置和导出 **TMPDIR** 在救援镜像引导时导致以下错误：

```
mktemp: failed to create file via template '/prefix/for/rear/working/directory/tmp.XXXXXXXXXX': No
such file or directory
cp: missing destination file operand after '/etc/rear/mappings/mac'
Try 'cp --help' for more information.
No network interface mapping is specified in /etc/rear/mappings/mac
```

或者，在运行 **rear recover** 时导致以下错误，并稍后中止：

```
ERROR: Could not create build area
```

要临时解决这个问题，如果您想要有一个自定义临时目录，请在执行 **ReaR** 环境前，通过在 shell 环境中导出变量来为 **ReaR** 临时文件指定一个自定义目录。例如，执行 **export TMPDIR=...** 语句，然后在同一 shell 会话或脚本中执行 **rear** 命令。因此，在上述配置中，恢复成功。

[Jira:RHEL-24847](#)

### 使用 **ifcfg** 文件重命名网络接口失败

在 RHEL 9 中，默认情况下不会安装 **initscripts** 软件包。因此，使用 **ifcfg** 文件重命名网络接口会失败。要解决这个问题，红帽建议您使用 **udev** 规则或链接文件来重命名接口。详情请查看 [Consistent 网络接口设备命名](#) 和 **systemd.link(5)** man page。

如果您无法使用推荐的解决方案之一，请安装 **initscripts** 软件包。

([BZ#2018112](#))

## RHEL 9 中不默认安装 chkconfig 软件包

RHEL 9 中不默认安装 **chkconfig** 软件包（更新和查询系统服务运行级别信息）。

要管理服务，请使用 **systemctl** 命令或手动安装 **chkconfig** 软件包。

有关 **systemd** 的更多信息，请参阅[管理 systemd](#)。有关如何使用 **systemctl** 实用程序的步骤，请参阅[使用 systemctl 管理系统服务](#)。

(BZ#2053598)

## 8.5. 基础架构服务

### bind 和 unbound 都禁用基于 SHA-1- 的签名验证

**bind** 和 **unbound** 组件禁用所有 RSA/SHA1（算法 5）和 RSASHA1-NSEC3-SHA1（算法号 7）签名，且签名的 SHA-1 用法在 DEFAULT 系统范围的加密策略中受到限制。

因此，某些 DNSSEC 记录使用 SHA-1、RSA/SHA1 和 RSASHA1-NSEC3-SHA1 摘要算法无法验证在 Red Hat Enterprise Linux 9 中，受影响的域名会存在安全漏洞。

要临时解决这个问题，升级到不同的签名算法，如 RSA/SHA-256 或 elliptic curve 键。

有关受影响和存在安全漏洞的顶级域的信息和列表，请参阅[使用 RSASHA1 签名的 DNSSEC 记录失败来验证](#) 解决方案。

(BZ#2070495)

### 如果在多个区域中使用相同的可写区域文件，named 无法启动

BIND 不允许在多个区域中具有相同的可写区域文件。因此，如果配置包含多个区域，它们共享到可由 **named** 服务修改的文件的路径，则 **named** 无法启动。要临时解决这个问题，请使用 **in-view** 子句在多个视图间共享一个区域，并确保为不同的区使用不同的路径。例如，在路径中包含视图名称。

请注意，可写区域文件通常用于允许由 DNSSEC 维护的动态更新、从属区域或区域的区域。

(BZ#1984982)

### 设置控制台 keymap 在最小安装上需要 libxkbcommon 库

在 RHEL 9 中，某些 **systemd** 库依赖项已从动态链接转换为动态加载，因此您的系统在运行时可打开并使用库（当它们可用时）。有了这个更改，除非您安装必要的库，否则无法使用依赖于此类库的功能。这也会影响在最小安装的系统上设置键盘布局。因此，**localectl --no-convert set-x11-keymap gb** 命令会失败。

要临时解决这个问题，请安装 **libxkbcommon** 库：

```
# dnf install libxkbcommon
```

(BZ#2214130)

## 8.6. 安全性

### openssl 不检测 PKCS #11 令牌是否支持创建原始 RSA 或 RSA-PSS 签名

TLS 1.3 协议需要支持 RSA-PSS 签名。如果 PKCS #11 令牌不支持原始 RSA 或 RSA-PSS 签名，如果 **PKCS #11** 令牌持有密钥，使用 **OpenSSL** 库的服务器应用程序无法使用 **RSA** 密钥。因此，在上述场景中 TLS 通信会失败。

要临时解决这个问题，请配置服务器和客户端以使用 TLS 版本 1.2 作为可用最高 TLS 协议版本。

(BZ#1681178)

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

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

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

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

因此，可以在描述的场景中建立 TLS 连接。

(BZ#1685470)

### FIPS 批准的加密在 FIPS 模式中可以正常工作

经 FIPS 批准的加密可在 OpenSSL 工具包中正常工作，无论系统设置是什么。因此，您可以使用系统以 FIPS 模式运行时禁用的加密算法和密码，例如：

- 使用 RSA 密钥交换的 TLS 密码套件可以正常工作。
- 尽管使用 PKCS #1 和 SSLv23 补丁，或者使用比 2048 位更短的密钥，对公钥的加密算法进行基于 RSA 的加密算法。

(BZ#2053289)

### OpenSSL 不能在 FIPS 模式下使用引擎

引擎 API 在 OpenSSL 3.0 中已弃用，其与 OpenSSL Federal Information Processing Standards(FIPS)实现和其他兼容 FIPS 的实现不兼容。因此，OpenSSL 无法在 FIPS 模式下运行引擎。对这个问题没有临时解决方案。

(BZ#2087253)

### PSK 密码suites 无法用于 FUTURE 加密策略

预共享密钥(PSK)密码组合不能被识别为执行完美的转发保密(PFS)密钥交换方法。因此，**ECDHE-PSK** 和 **DHE-PSK** 加密套件无法与配置为 **SECLEVEL=3** 的 OpenSSL 一起工作，例如使用 **FUTURE** 加密策略。作为临时解决方案，您可以为使用 PSK 密码的应用程序设置限制较低的加密策略或设置较低的安全级别(**SECLEVEL**)。

(BZ#2060044)

### GnuPG 错误地允许使用 SHA-1 签名，即使通过 crypto-policies 禁止使用 SHA-1 签名

无论系统范围的加密策略中定义的设置如何，GNU Privacy Guard(GnuPG)加密软件可以创建和验证使用 SHA-1 算法的签名。因此，您可以在 **DEFAULT** 加密策略中将 SHA-1 用于加密目的，这与这个不安全算法的系统范围弃用没有一致的。

要临时解决这个问题，请不要使用涉及 SHA-1 的 GnuPG 选项。因此，您将使用非安全 SHA-1 签名来防止 GnuPG 降低默认系统安全性。

([BZ#2070722](#))

### 有些 OpenSSH 操作不使用 FIPS 批准的接口

OpenSSH 使用的 OpenSSL 加密库提供两个接口：legacy 和 modern。由于对 OpenSSL 内部的更改，只有现代接口使用加密算法的 FIPS 认证的实现。因为 OpenSSH 对某些操作使用旧的接口，因此它不会遵守 FIPS 要求。

([BZ#2087121](#))

### gpg-agent 在 FIPS 模式下不作为 SSH 代理工作

在将密钥添加到 `ssh-agent` 程序时，`gpg-agent` 工具会创建 MD5 指纹，即使 FIPS 模式禁用了 MD5 摘要。因此，`ssh-add` 程序无法将密钥添加到身份验证代理中。

要临时解决这个问题，请在不使用 `gpg-agent --daemon --enable-ssh-support` 命令的情况下创建 `~/.gnupg/sshcontrol` 文件。例如，您可以将 `<FINGERPRINT> 0` 格式的 `gpg --list-keys` 命令的输出粘贴到 `~/.gnupg/sshcontrol`。因此，`gpg-agent` 作为 SSH 验证代理工作。

([BZ#2073567](#))

### SELinux `staff_u` 用户可能会错误地切换到 `unconfined_r`

启用 `secure_mode` 布尔值后，`staff_u` 用户可能会错误地切换到 `unconfined_r` 角色。因此，`staff_u` 用户可以执行影响系统安全性的特权操作。

([BZ#2021529](#))

### 默认 SELinux 策略允许无限制的可执行文件使其堆栈可执行

SELinux 策略中的 `selinuxuser_execstack` 布尔值的默认状态是 `on`，这意味着无限制的可执行文件可以使其堆栈为可执行。可执行文件不应该使用这个选项，这通常代表开发的可执行代码的质量较差，或可能存在安全攻击的风险。但是，由于需要与其他工具、软件包和第三方产品保持兼容，红帽无法更改默认策略中的这个布尔值。如果您的环境没有此类兼容性问题，请使用 `setsebool -P selinuxuser_execstack off` 命令在您的本地策略中将这个布尔值设置为 `off`。

([BZ#2064274](#))

### 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](#))

### STIG 配置集中的 SSH 超时规则配置了不正确的选项

对 OpenSSH 的更新会影响以下 Defense Information Systems Agency Security Technical Implementation Guide (DISA STIG) 配置集中的规则：

- DISA STIG for RHEL 9 ([xccdf\\_org.ssgproject.content\\_profile\\_stig](#))
- DISA STIG with GUI for RHEL 9 ([xccdf\\_org.ssgproject.content\\_profile\\_stig\\_gui](#))

在每个配置集中，以下两条规则会受到影响：

Title: Set SSH Client Alive Count Max to zero  
 CCE Identifier: CCE-90271-8  
 Rule ID: xccdf\_org.ssgproject.content\_rule\_sshd\_set\_keepalive\_0

Title: Set SSH Idle Timeout Interval  
 CCE Identifier: CCE-90811-1  
 Rule ID: xccdf\_org.ssgproject.content\_rule\_sshd\_set\_idle\_timeout

当应用到 SSH 服务器时，每个规则都会配置一个选项（**ClientAliveCountMax** 和 **ClientAliveInterval**），它们的行为与之前的不同。因此，当 OpenSSH 达到这些规则配置的超时时间，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 9 和 DISA STIG with GUI for RHEL 9 配置集中临时删除，直到开发出解决方案为止。

([BZ#2038978](#))

### fagenrules --load 无法正常工作

**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。因此，**fagenrules --load** 命令无法正常工作，规则更新需要手动重启 **fapolicyd**。要临时解决这个问题，当规则中有任何更改时会重启 **fapolicyd** 服务，因此 **fagenrules --load** 可以正常工作。

([BZ#2070655](#))

### Ansible 补救需要额外的集合

用 **ansible-core** 软件包替换 Ansible Engine 时，RHEL 订阅提供的 Ansible 模块的列表会减少。因此，运行使用包含在 **scap-security-guide** 软件包中的 Ansible 内容的补救需要来自 **rhc-worker-playbook** 软件包的集合。

对于 Ansible 补救，请执行以下步骤：

1. 安装所需的软件包：

```
# dnf install -y ansible-core scap-security-guide rhc-worker-playbook
```

2. 进入到 **/usr/share/scap-security-guide/ansible** 目录：`# cd /usr/share/scap-security-guide/ansible`

3. 运行使用环境变量的相关 Ansible playbook，这些变量定义了到额外 Ansible 集合的路径：

```
# ANSIBLE_COLLECTIONS_PATH=/usr/share/rhc-worker-playbook/ansible/collections/ansible_collections/ ansible-playbook -c local -i localhost, rhel9-playbook-cis_server_11.yml
```

将 **cis\_server\_11** 替换为您要修复系统的配置文件的 ID。

因此，Ansible 内容会被正确处理。



#### 注意

对 **rhc-worker-playbook** 中提供的集合的支持仅限于启用 **scap-security-guide** 中提供的 Ansible 内容。

([BZ#2105162](#))

## 8.7. 网络

### nm-cloud-setup 服务从接口中删除手动配置辅助 IP 地址

根据从云环境收到的信息，**nm-cloud-setup** 服务会配置网络接口。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#2151040](#))

### 内核命令行中的空 **rd.znet** 选项会导致网络配置失败

内核中的 **rd.znet** 选项没有任何参数，如网络类型或子频道无法配置网络。要临时解决这个问题，可完全从命令行中删除 **rd.znet** 选项，或者指定相关的网络类型、子频道和其他相关选项。有关这些选项的详情请参考 **dracut.cmdline(7)** man page。

([BZ#1931284](#))

### 更新会话密钥失败会导致连接中断

内核传输层安全(kTLS)协议不支持更新会话密钥，这些密钥由对称密码使用。因此，用户无法更新密钥，从而导致连接中断。要临时解决这个问题，请禁用 kTLS。因此，解决这一问题，可以成功更新会话密钥。

([BZ#2013650](#))

### 默认情况下不安装 **initscripts** 软件包

默认情况下，不会安装 **initscripts** 软件包。因此，**ifup** 和 **ifdown** 工具不可用。一个替代的方法是，可以使用 **nmcli connection up** 和 **nmcli connection down** 命令来启用和禁用连接。如果这个替代方法无法正常工作，请报告这个问题并安装 **NetworkManager-initscripts-updown** 软件包，该软件包为 **ifup** 和 **ifdown** 工具提供了一个 NetworkManager 解决方案。

([BZ#2082303](#))

### 在 Alibaba Cloud 中启动 **nm-cloud-setup** 服务后，实例的主要 IP 地址会改变

在 Alibaba Cloud 中启动实例后，**nmn-cloud-setup** 服务会将主 IP 地址分配给实例。但是，如果您为实例分配多个辅助 IP 地址并启动 **nm-cloud-setup** 服务，则前一个主 IP 地址被替换为已经分配的辅助 IP 地址之一。返回的元数据列表会验证这个情况。要临时解决这个问题，请手动配置辅助 IP 地址以避免主 IP 地址更改。因此，实例会保留 IP 地址，主 IP 地址不会改变。

([BZ#2079849](#))

## 8.8. 内核

### kdump 无法在 RHEL 9 内核中启动

RHEL 9 内核没有默认配置的 `crashkernel=auto` 参数。因此，`kdump` 服务默认无法启动。

要临时解决这个问题，请将 `crashkernel=` 选项配置为所需的值。

例如，要使用 `grubby` 工具保留 256 MB 内存，请输入以下命令：

```
# grubby --args crashkernel=256M --update-kernel ALL
```

因此，RHEL 9 内核会启动 `kdump`，并使用配置的内存大小值转储 `vmcore` 文件。

(BZ#1894783)

### kdump 机制无法捕获 LUKS 加密目标中的 vmcore

当在使用 Linux Unified Key Setup(LUKS)加密分区的系统中运行 `kdump` 时，系统需要特定的可用内存。当可用内存小于所需内存量时，`systemd-cryptsetup` 服务将无法挂载分区。因此，第二个内核无法捕获 LUKS 加密目标上的崩溃转储文件(`vmcore`)。

使用 `kdumpctl estimate` 命令，您可以查询推荐的 `crashkernel` 值，这是推荐的 `kdump` 所需的内存大小。

要临时解决这个问题，请按照以下步骤为 LUKS 加密目标上的 `kdump` 配置所需的内存：

1. 输出估计的 `crashkernel` 值：

```
# kdumpctl estimate
```

2. 通过增大 `crashkernel` 值来配置所需的内存量：

```
# grubby --args=crashkernel=652M --update-kernel=ALL
```

3. 重启系统以使更改生效。

```
# reboot
```

因此，`kdump` 在带有 LUKS 加密分区的系统上可以正常工作。

(BZ#2017401)

### 在引导时分配崩溃内核内存会失败

在某些 Ampere Altra 系统上，在可用内存低于 1 GB 时，在引导过程中，为 `kdump` 使用分配崩溃内核内存会失败。因此，`kdumpctl` 命令无法启动 `kdump` 服务，因为所需内存大于可用内存。

作为临时解决方案，将 `crashkernel` 参数的值减少至少 240 MB 以满足大小要求，例如 `crashkernel=240M`。因此，`kdump` 的崩溃内核内存分配不会在 Ampere Altra 系统上失败。

(BZ#2065013)

### KTLS 不支持将 TLS 1.3 卸载到 NIC

内核传输层安全(kTLS)不支持将 TLS 1.3 卸载到 NIC。因此，即使 NIC 支持 TLS 卸载，软件加密也会与 TLS 1.3 一起使用。要临时解决这个问题，如果需要卸载，禁用 TLS 1.3。因此，您只能卸载 TLS 1.2。当使用 TLS 1.3 时，性能较低，因为无法卸载 TLS 1.3。

(BZ#2000616)

### 启用安全引导的 FADump 可能会导致 GRUB 内存不足(OOM)

在安全引导环境中，GRUB 和 PowerVM 一起分配一个 512 MB 内存区域，称为 Real Mode Area (RMA)，用于引导内存。区域在引导组件之间划分，如果任何一个组件超过其分配，则会发生内存不足故障。

通常，默认安装的 **initramfs** 文件系统和 **vmlinuz** 符号表在限制内，以避免出现这样的故障。但是，如果在系统中启用了固件辅助转储(FADump)，则默认的 **initramfs** 大小可以增加并超过 95 MB。因此，每个系统重启都会导致 GRUB OOM 状态。

要避免这个问题，请不要将安全引导和 FADump 一起使用。有关如何临时解决这个问题的更多信息和方法，请参阅链接：<https://www.ibm.com/support/pages/node/6846531>。

(BZ#2149172)

### 安全引导中的系统无法运行动态 LPAR 操作

如果满足这些条件中的任何一，则用户无法从硬件管理控制台(HMC)运行动态逻辑分区(DLPAR)操作：

- 启用安全引导功能，隐式在完整性模式下启用内核 **lockdown** 机制。
- 内核 **lockdown** 机制在完整性或机密模式下被手动启用。

在 RHEL 9 中，内核 **lockdown** 会完全阻止 Run Time Abstraction 服务(RTAS)访问可通过 **/dev/mem** 字符设备文件访问的系统内存。多个 RTAS 调用需要对 **/dev/mem** 有写访问才能正常工作。因此，RTAS 调用无法正确执行，用户会看到以下错误消息：

```
HSCL2957 Either there is currently no RMC connection between the management console and the
partition <LPAR name> or the partition does not support dynamic partitioning operations. Verify the
network setup on the management console and the partition and ensure that any firewall
authentication between the management console and the partition has occurred. Run the
management console diagrmc command to identify problems that might be causing no RMC
connection.
```

(BZ#2083106)

### dkms 使用在 64 位 ARM CPU 上正确编译的驱动程序对程序失败提供不正确的警告

动态内核模块支持(**dkms**)工具无法识别适用于 4 KB 和 64 KB 页大小内核的 64 位 ARM CPU 的内核头文件。因此，当执行了内核更新，且 **kernel-64k-devel** 软件包未安装时，**dkms** 会提供一条有关为什么程序在正确编译的驱动程序上失败的错误警告。要临时解决这个问题，请安装 **kernel-headers** 软件包，其包含用于两种类型的 ARM CPU 架构的头文件，且不特定于 **dkms** 及其要求。

(JIRA:RHEL-25967)

## 8.9. 引导装载程序

### 新内核丢失了以前的命令行选项

GRUB 引导装载程序不会将之前配置的内核命令行选项应用到新内核。因此，当您升级内核软件包时，因为缺少选项，系统在重启后可能会改变。

要临时解决这个问题，请在每次内核升级后手动添加所有自定义内核命令行选项。因此，内核会如预期应用自定义选项，直到下一个内核升级为止。

(BZ#1969362)

## 8.10. 文件系统和存储

### NVMe/TCP 不支持设备映射器多路径

使用带有 **nvme-tcp** 驱动程序的设备映射器多路径可能会导致 Call Trace 警告和系统不稳定。要临时解决这个问题，NVMe/TCP 用户必须启用原生 NVMe 多路径，且不能在 NVMe 中使用 **device-mapper-multipath** 工具。

默认情况下，RHEL 9 中启用了原生 NVMe 多路径。如需更多信息，请参阅[在 NVMe 设备上启用多路径](#)。

(BZ#2033080)

### blk-availability systemd 服务取消激活复杂设备堆栈

在 **systemd** 中，默认块取消激活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，在关闭过程中可能无法删除虚拟设备，从而导致错误信息被记录。要临时解决这个问题，请执行以下命令取消激活复杂块设备堆栈：

```
# systemctl enable --now blk-availability.service
```

因此，在关闭过程中可以正确地取消激活复杂的虚拟设备堆栈，且不会生成错误消息。

(BZ#2011699)

### supported\_speeds 的无效 sysfs 值

**qla2xxx** 驱动程序报告 20Gb/s，而不是预期的 64Gb/s 作为 **sysfs supported\_speeds** 属性中支持的端口速度之一：

```
$ cat /sys/class/fc_host/host12/supported_speeds
16 Gbit, 32 Gbit, 20 Gbit
```

因此，如果 HBA 支持 64Gb/s 链接速度，**sysfs supported\_speeds** 值不正确。这只会影响 **sysfs** 的 **supported\_speeds** 值，端口在预期的协商链接率中运行。

(BZ#2069758)

### 无法从 AMD EPYC 系统上的 Broadcom initiator 连接到 NVMe 命名空间

默认情况下，RHEL 内核在基于 AMD 的平台上启用 IOMMU。因此，当您在带有 AMD 处理器的服务器中使用启用了 IOMMU 的平台时，您可能会遇到 NVMe I/O 问题，如 I/O 因传输长度不匹配而失败。

要临时解决这个问题，使用内核命令行选项 **iommu=pt** 在 passthrough 模式中添加 IOMMU。现在，您可以在 AMD EPYC 系统中从 Broadcom initiator 连接到 NVMe 命名空间。

(BZ#2073541)

## 8.11. 动态编程语言、网页和数据库服务器

### MySQL 和 MariaDB 中的 --ssl-fips-mode 选项不会改变 FIPS 模式

RHEL 中 **MySQL** 和 **MariaDB** 中的 `--ssl-fips-mode` 选项与上游中的工作方式不同。

在 RHEL 9 中，如果您使用 `--ssl-fips-mode` 作为 `mysqld` 或 `mariadb` 守护进程的参数，或者在 **MySQL** 或 **MariaDB** 服务器配置文件中 使用 `ssl-fips-mode`，则 `--ssl-fips-mode` 不会更改这些数据库服务器的 FIPS 模式。

相反：

- 如果将 `--ssl-fips-mode` 设为 **ON**，则 `mysqld` 或 `mariadb` 服务器守护进程不会启动。
- 如果您在启用了 FIPS 的系统上将 `--ssl-fips-mode` 设为 **OFF**，则 `mysqld` 或 `mariadb` 服务器守护进程仍然在 FIPS 模式下运行。

这是预期的，因为应该为整个 RHEL 系统启用或禁用 FIPS 模式，而不是为特定组件。

因此，不要在 RHEL 中的 **MySQL** 或 **MariaDB** 中使用 `--ssl-fips-mode` 选项。相反，请确保在整个 RHEL 系统上启用 FIPS 模式：

- 最好安装启用了 FIPS 模式的 RHEL。在安装过程中启用 FIPS 模式可确保系统使用 FIPS 批准的算法生成所有的密钥，并持续监控测试。有关在 FIPS 模式下安装 RHEL 的详情，请参考 [在 FIPS 模式下安装系统](#)。
- 或者，您可以按照 [将系统切换到 FIPS 模式](#) 中的流程，为整个 RHEL 系统切换 FIPS 模式。

([BZ#1991500](#))

## 8.12. 编译器和开发工具

### 某些基于符号的探测无法在 64 位 ARM 架构上的 SystemTap 中工作

内核配置禁用 **SystemTap** 所需要的某些功能。因此，一些基于符号的探测无法在 64 位 ARM 架构上工作。因此，受影响的 **SystemTap** 脚本可能无法运行，或可能不会收集对所需探测点的点击。

请注意，随着 [RHBA-2022:5259](#) 公告的发布，已针对其余架构对此 bug 进行了修复。

([BZ#2083727](#))

## 8.13. 身份管理

### RHEL 9 Kerberos 客户端无法针对 Heimdal KDC 使用 PKINIT 验证用户

在 RHEL 9 Kerberos 客户端中 IdM 用户的 PKINIT 验证过程中，RHEL 9 或者更早的 Kerberos 分发中心 (KDC) 使用 SHA-1 备份签名算法，因为 Kerberos 客户端不支持 `supportedCMSTypes` 字段。但是，RHEL 9 中弃用了 SHA-1 算法，因此用户身份验证会失败。

要临时解决这个问题，使用以下命令在 RHEL 9 客户端上启用对 SHA-1 算法的支持：

```
# update-crypto-policies --set DEFAULT:SHA1
```

因此，PKINIT 身份验证在 Kerberos 客户端和 Heimdal KDC 之间正常工作。

有关支持的备份签名算法的详情，请参阅 [CMS Algorithm 标识符的 Kerberos 加密类型](#)。

另请参阅 [如果 RHEL 9 Kerberos 代理与非 RHEL 9 Kerberos 代理进行通信，则用户的 PKINIT 身份验证会失败](#)。

(BZ#2068935)

**如果 RHEL 9 Kerberos 代理与非 RHEL 9 Kerberos 代理进行通信，则用户的 PKINIT 身份验证会失败**

如果 RHEL 9 Kerberos 代理与您环境中的其他非 RHEL 9 Kerberos 代理进行交互，则用户的初始验证 (PKINIT) 验证的公钥加密会失败。要临时解决这个问题，请执行以下操作之一：

- 将 RHEL 9 代理的加密策略设为 **DEFAULT:SHA1** 以允许 SHA-1 签名的验证：

```
# update-crypto-policies --set DEFAULT:SHA1
```

- 更新非 RHEL 9 代理，以确保它不会使用 SHA-1 算法为 CMS 数据签名。为此，将 Kerberos 软件包更新为使用 SHA-256 而不是 SHA-1 的版本：
  - CentOS 9 流: krb5-1.19.1-15
  - RHEL 8.7: krb5-1.18.2-17
  - RHEL 7.9: krb5-1.15.1-53
  - Fedora Rawhide/36: krb5-1.19.2-7
  - Fedora 35/34 : krb5-1.19.2-3

无论没打补丁的代理是 Kerberos 客户端还是 Kerberos 发布中心(KDC)，您都必须执行这些操作中的一个。

因此，用户的 PKINIT 身份验证可以正常工作。

请注意，对于其他操作系统，是 krb5-1.20 版本确保代理使用 SHA-256 而不是 SHA-1 为 CMS 数据签名。

另请参阅 [必须在 RHEL 9 客户端上设置 DEFAULT:SHA1 子策略，以便 PKINIT 能够对旧的 RHEL KDC 和 AD KDC 工作。](#)

(BZ#2077450)

**必须在 RHEL 9 客户端上设置 DEFAULT:SHA1 子策略，以便 PKINIT 能够对旧的 RHEL KDC 和 AD KDC 工作**

RHEL 9 中已弃用了 SHA-1 摘要算法，对初始验证的公共密钥加密的 CMS 消息现在使用更强大的 SHA-256 算法进行签名。

从 RHEL 7.9 和 RHEL 8.7 开始，虽然默认使用 SHA-256，但 RHEL 7.8 和 RHEL 8.6 及其早期版本上的旧 Kerberos 密钥分发中心(KDC)仍使用 SHA-1 摘要算法来签名 CMS 信息。活动目录(AD) KDC 也是如此。

因此，RHEL 9 Kerberos 客户端无法使用 PKINIT 针对以下情况验证用户：

- 在 RHEL 7.8 及更早版本上运行的 KDC
- 在 RHEL 8.6 和更早版本上运行的 KDC
- AD KDC

要临时解决这个问题，使用以下命令在 RHEL 9 系统上启用对 SHA-1 算法的支持：

■

```
# update-crypto-policies --set DEFAULT:SHA1
```

另请参阅 [RHEL 9 Kerberos 客户端无法针对 Heimdal KDC 使用 PKINIT 验证用户](#)。

([BZ#2060798](#))

## AD 信任的 FIPS 支持需要 AD-SUPPORT 加密子策略

Active Directory(AD)使用 AES SHA-1 HMAC 加密类型，默认情况下在 RHEL 9 上不允许 FIPS 模式。如果要使用带有 AD 信任的 RHEL 9 IdM 主机，请在安装 IdM 软件前支持 AES SHA-1 HMAC 加密类型。

由于 FIPS 合规性是一个涉及技术和机构协议的进程，请在启用 **AD-SUPPORT** 子策略前参考 FIPS 审核员，以允许技术测量结果支持 AES SHA-1 HMAC 加密类型，然后安装 RHEL IdM：

```
# update-crypto-policies --set FIPS:AD-SUPPORT
```

([BZ#2057471](#))

## 当以引用模式启动时，目录服务器会意外终止

由于一个程序错误，全局引用模式无法在 Directory Server 中工作。如果您以 **dirsrv** 用户身份使用 **refer** 选项启动 **ns-slapd** 进程，则目录服务器会忽略端口设置并意外终止。尝试以 **root** 用户身份运行进程会更改 SELinux 标签，并可以防止服务以后以正常模式启动。没有可用的临时解决方案。

([BZ#2053204](#))

## 在 Directory 服务器中为后缀配置引用失败

如果您在 Directory 服务器中设置了后端引用，使用 **dsconf <instance\_name> backend suffix set --state referral** 命令设置后端状态会失败，并显示以下错误：

```
Error: 103 - 9 - 53 - Server is unwilling to perform - [] - need to set nsslapd-referral before moving to referral state
```

因此，为后缀配置引用会失败。要临时解决这个问题：

1. 手动设置 **nsslapd-referral** 参数：

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com

dn: cn=dc\3Dexample\2Cdc\3Dcom,cn=mapping tree,cn=config
changetype: modify
add: nsslapd-referral
nsslapd-referral: ldap://remote_server:389/dc=example,dc=com
```

2. 设置后端状态：

```
# dsconf <instance_name> backend suffix set --state referral
```

因此，您可以使用临时解决方案为后缀配置引用。

([BZ#2063140](#))

## dsconf 实用程序没有为 entryUUID 插件创建修复任务的选项

**dsconf** 实用程序没有提供为 **entryUUID** 插件创建修复任务的选项。因此，管理员无法使用 **dsconf** 创建任务来自动将 **entryUUID** 属性添加到现有条目。作为临时解决方案，请手动创建任务：

```
# ldapadd -D "cn=Directory Manager" -W -H ldap://server.example.com -x
dn: cn=entryuuid_fixup__<time_stamp__>,cn=entryuuid task,cn=tasks,cn=config
objectClass: top
objectClass: extensibleObject
basedn: __<fixup base tree>__
cn: entryuuid_fixup__<time_stamp>__
filter: __<filtered_entry>__
```

创建了任务后，Directory 服务器会修复缺少或无效 **entryUUID** 属性的条目。

([BZ#2047175](#))

### 将对 **ldap\_id\_use\_start\_tls** 选项使用默认值时的潜在风险

当不使用 TLS 进行身份查找的情况下使用 **ldap://** 时，可能会对攻击向量构成风险。特别是中间人(MITM)攻击，例如，其可以通过更改LDAP 搜索中返回的对象的 UID 或 GID 来允许攻击者冒充用户。

目前，用于强制 TLS 的 **ldap\_id\_use\_start\_tls** SSSD 配置选项，默认为 **false**。确保您的设置可在可信环境中操作，并决定是否可以对 **id\_provider = ldap** 使用未加密的通信。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** 选项设为 **true** 来强制使用 TLS。计划在以后的 RHEL 版本中对默认的行为会有更改。

([JIRA:RHELPLAN-155168](#))

## 8.14. DESKTOP

### 升级到 RHEL 9 后将禁用 Firefox 附加组件

如果您从 RHEL 8 升级到 RHEL 9，则之前在 Firefox 中启用的所有附加组件都会被禁用。

要临时解决这个问题，请手动重新安装或更新附加组件。因此，附加组件会如预期启用。

([BZ#2013247](#))

### 升级到 RHEL 9 后，VNC 没有运行

从 RHEL 8 升级到 RHEL 9 后，VNC 服务器无法启动，即使之前启用它。

要临时解决这个问题，在系统升级后手动启用 **vncserver** 服务：

```
# systemctl enable --now vncserver@:port-number
```

现在，每个系统引导后都会启用 VNC 并按预期启动。

([BZ#2060308](#))

## 8.15. 图形基础结构

### Matrox G200e 显示在 VGA 上没有任何输出

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. 在内核命令行中添加 `module_blacklist=mgag200` 选项。

因此，RHEL 会正常引导并如期显示图形输出，但最大分辨率限制为 16 位颜色深度的 1024x768。

(BZ#1960467)

### x.org 配置工具无法在 Wayland 下工作

用于操作屏幕的 x.org 实用程序无法在 Wayland 会话中工作。值得注意的是，`xrandr` 实用程序无法在 Wayland 下工作，因为其处理、解析、轮转和布局的不同方法。

(JIRA:RHELPLAN-121049)

### NVIDIA 驱动程序可能会恢复到 X.org

在某些情况下，专有 NVIDIA 驱动程序会禁用 Wayland 显示协议并恢复到 X.org 显示服务器：

- 如果 NVIDIA 驱动程序的版本低于 470。
- 如果系统是使用混合图形的笔记本电脑。
- 如果您还没有启用所需的 NVIDIA 驱动程序选项。

另外，启用 Wayland，但如果 NVIDIA 驱动程序的版本低于 510，则桌面会话默认使用 X.org。

(JIRA:RHELPLAN-119001)

### 使用 NVIDIA 在 Wayland 上无法使用 night Light

当您的系统上启用了专有 NVIDIA 驱动程序时，Wayland 会话将无法使用 GNOME 的 Night Light 功能。NVIDIA 驱动程序目前不支持 Night Light。

(JIRA:RHELPLAN-119852)

## 8.16. WEB 控制台

### 使用 Web 控制台删除 USB 主机设备无法正常工作

当您把 USB 设备附加到虚拟机时，USB 设备的设备号和总线号可能会在传递给虚拟机后发生更改。因此，使用 Web 控制台删除这样的设备会失败，因为设备和总线号不正确。要解决这个问题，从虚拟机的 XML 配置中删除 `<hostdev>` 部分 USB 设备。

(JIRA:RHELPLAN-109067)

### 使用 Web 控制台附加多个主机设备无法正常工作

当您选择使用 web 控制台附加到虚拟机(VM)的多个设备时，只会附加一个设备，且其它设备会被忽略。要临时解决这个问题，请一次只附加一个设备。

(JIRA:RHELPLAN-115603)

## 8.17. 虚拟化

### 在某些情况下，通过 https 安装虚拟机失败

目前，当尝试通过 https 连接从 ISO 源安装客户机操作系统时，**virt-install** 工具会失败 - 例如使用 **virt-install --cdrom https://example/path/to/image.iso**。所描述的操作意外终止并显示 **internal error: process exited while connecting to monitor** 错误，而不是创建虚拟机(VM)

要临时解决这个问题，请在主机上安装 **qemu-kvm-block-curl** 来启用 https 协议支持。或者，使用不同的连接协议或不同的安装源。

(BZ#2014229)

### 在虚拟机中使用 NVIDIA 驱动程序会禁用 Wayland

目前，NVIDIA 驱动程序与 Wayland 图形会话不兼容。因此，使用 NVIDIA 驱动程序的 RHEL 客户机操作系统会自动禁用 Wayland 并加载 Xorg 会话。这主要在以下情况下发生：

- 当您通过 NVIDIA GPU 设备传递给 RHEL 虚拟机(VM)
- 当您为 RHEL 虚拟机分配 NVIDIA vGPU mediated 设备

(JIRA:RHELPLAN-117234)

### 在 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，载这些系统上可能无法提供"Milan"CPU 类型。另外，载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，在主机 BIOS 中手动打开 **erms** 和 **fsrm**。

(BZ#2077767)

### 虚拟机中的网络流量性能可能会降低

在某些情况下，RHEL 9.0 虚拟机(VM)在处理高级别网络流量时会降低性能。

(BZ#1945040)

### 禁用 AVX 会导致虚拟机无法引导

在使用具有高级向量扩展(AVX)支持的 CPU 的主机上，尝试引导明确禁用 AVX 的虚拟机当前会失败，并触发虚拟机中的内核 panic。

(BZ#2005173)

### 在 Windows 虚拟机上不分配 IP 地址故障转移 virtio NIC

目前，当使用故障转移 virtio NIC 启动 Windows 虚拟机时，虚拟机无法为 NIC 分配 IP 地址。因此，NIC 无法设置网络连接。目前，没有临时解决方案。

(BZ#1969724)

### 带有故障切换设置的 hostdev 接口在热拔后无法进行热插

从正在运行的虚拟机(VM)中删除带有故障切换配置的 **hostdev** 网络接口后，该接口目前无法重新连接到同一正在运行的虚拟机。

([BZ#2052424](#))

### 带有故障切换 VF 的虚拟机实时复制迁移失败

目前，如果虚拟机使用启用了虚拟功能(VF)故障转移功能的设备，则试图对一个正在运行的虚拟机(VM)进行 post-copy 迁移会失败。要临时解决这个问题，请使用标准迁移类型，而不要使用 post-copy 迁移方式。

([BZ#1817965](#), [BZ#1789206](#))

## 8.18. 云环境中的 RHEL

### SR-IOV 在 Azure 上的 ARM 64 RHEL 9 虚拟机中执行子优化

目前，SR-IOV 网络设备在 Microsoft Azure 平台上运行的 ARM 64 RHEL 9 虚拟机中已明显低于预期的延迟。

([BZ#2068432](#))

### 使用控制台代理的 XenServer 7 上的 RHEL 9 虚拟机无法使用鼠标

在使用控制台代理的 XenServer 7 平台中运行 RHEL 9 虚拟机(VM)时，无法在 VM GUI 中使用鼠标。要临时解决这个问题，请禁用虚拟机中的 Wayland Compositor 协议，如下所示：

1. 打开 `/etc/gdm/custom.conf` 文件。
2. 取消注释 `WaylandEnable=false` 行。
3. 保存该文件。

此外，请注意，红帽不支持将 XenServer 用作运行 RHEL 虚拟机的平台，并且不建议在生产环境中使用带有 RHEL 的 XenServer。

([BZ#2019593](#))

### 在 Nutanix AHV 中使用 LVM 克隆或恢复 RHEL 9 虚拟机会导致非 root 分区消失

当在 Nutanix AHV 虚拟机监控程序上托管的虚拟机中运行 RHEL 9 客户机操作系统时，从快照中恢复虚拟机或克隆虚拟机目前会导致虚拟机中的非 root 分区在虚拟机中使用逻辑卷管理(LVM)时消失。因此，会出现以下问题：

- 从快照恢复虚拟机后，虚拟机无法引导，而是进入紧急模式。
- 通过克隆创建的虚拟机无法引导，而是进入紧急模式。

要临时解决这个问题，在虚拟机的紧急模式下执行以下操作：

1. 删除 LVM 系统设备文件：`rm /etc/lvm/devices/system.devices`
2. 重新创建 LVM 设备设置：`vgimportdevices -a`
3. 重启虚拟机

这样，克隆或恢复的虚拟机可以正确引导。

(BZ#2059545)

### 附加到 Hyper-V 虚拟机的网络适配器的 SR-IOV 功能可能无法工作

目前，当把带有单根 I/O 虚拟化(SR-IOV)的网络适配器附加到在 Microsoft Hyper-V hypervisor 上运行的 RHEL 9 虚拟机(VM)时，在某些情况下 SR-IOV 功能无法正常工作。

要临时解决这个问题，在虚拟机配置中禁用 SR-IOV，然后再次启用它。

1. 在 Hyper-V Manager 窗口中，右键点击虚拟机。
2. 在上下文菜单中，进入到 **Settings/Network Adapter/Hardware Acceleration**。
3. 取消选中 **启用 SR-IOV**。
4. 点应用。
5. 重复步骤 1 和 2 以再次进入到 **Enable SR-IOV** 选项。
6. 选择**启用 SR-IOV**。
7. 点应用。

(BZ#2030922)

### 在 ESXi 上自定义 RHEL 9 客户机有时会导致网络问题

目前，在 VMware ESXi hypervisor 中自定义 RHEL 9 客户机操作系统无法正常工作。因此，如果客户机使用这样的密钥文件，它有不正确的网络设置，如 IP 地址或网关。

有关详情和临时解决方案说明，请参阅 [VMware 知识库](#)。

(BZ#2037657)

## 8.19. 支持性

### 在 IBM Power Systems Little Endian 上运行 `sos report` 时超时

在带有数百或数千 CPU 的 IBM Power Systems Little Endian 上运行 `sos report` 命令时，当收集 `/sys/devices/system/cpu` 目录中的大量内容时，处理器插件会达到其默认的超时时间（300 秒）。作为临时解决方案，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
```

- 对于永久更改，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。特定的超时时值高度取决于特定的系统。要相应地设置插件超时，您可以首先通过运行以下命令估算收集没有超时的插件所需的时间：

-

```
# time sos report -o processor -k processor.timeout=0 --batch --build
```

(BZ#1869561)

## 8.20. 容器

### 使用 Beta GPG 密钥签名的容器镜像无法拉取

目前，当您试图拉取 RHEL 9 Beta 容器镜像时，**podman** 会退出并显示错误消息：**Error:Source image rejected:None of the signatures were accepted**。镜像因为当前构建被配置为不信任 RHEL Beta GPG 密钥而无法拉取。

作为临时解决方案，请确保红帽 Beta GPG 密钥保存在您的本地系统中，并使用针对适当的 beta 命名空间更新 **podman image trust set** 命令的现有信任范围。

如果您没有本地存储 Beta GPG 密钥，您可以通过运行以下命令来拉取它：

```
sudo wget -O /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-beta
https://www.redhat.com/security/data/f21541eb.txt
```

要将 Beta GPG 密钥作为信任添加到命名空间中，请使用以下命令之一：

```
$ sudo podman image trust set -f /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-beta
registry.access.redhat.com/namespace
```

和

```
$ sudo podman image trust set -f /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-beta
registry.redhat.io/namespace
```

将 *namespace* 替换为 *ubi9-beta* 或 *rhel9-beta*。

(BZ#2020026)

### Podman 无法拉取容器 "X509: certificate signed by unknown authority"

如果您的内部注册中心由自己的 CA 证书签名，则必须将证书导入到主机上。否则，会发生错误：

```
x509: certificate signed by unknown authority
```

在主机上导入 CA 证书：

```
# cd /etc/pki/ca-trust/source/anchors/
[anchors]# curl -O <your_certificate>.crt

[anchors]# update-ca-trust
```

然后，您可以从内部注册中心拉取容器镜像。

(BZ#2027576)

### 在较旧的容器镜像中运行 systemd 无法正常工作

在较旧的容器镜像中运行 systemd，如 **centos:7**，它无法正常工作：

-

```
$ 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)

### podman system connection add 和 podman image scp 失败

Podman 为 RSA 密钥交换使用 SHA-1 哈希。在使用 RSA 密钥的机器间，常规的 SSH 连接可以正常工作，但使用相同 RSA 密钥的 **podman system connection add** 和 **podman image scp** 无法正常工作，这是因为在 RHEL 9 的密钥交换中不接受 SHA-1 哈希：

```
$ podman system connection add --identity ~/.ssh/id_rsa test_connection
$REMOTE_SSH_MACHINE
Error: failed to connect: ssh: handshake failed: ssh: unable to authenticate, attempted methods [none
publickey], no supported methods remain
```

要临时解决这个问题，使用 ED25519 密钥：

1. 连接到远程机器：

```
$ ssh -i ~/.ssh/id_ed25519 $REMOTE_SSH_MACHINE
```

2. 为 Podman 服务记录 ssh 目标：

```
$ podman system connection add --identity ~/.ssh/id_ed25519 test_connection
$REMOTE_SSH_MACHINE
```

3. 验证 ssh 目标是否已记录：

```
$ podman system connection list
```

请注意，随着 [RHBA-2022:5951](#) 公告的发布，这个问题已被解决。

(JIRA:RHELPLAN-121180)

## 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
<b>389-ds-base</b>	<a href="#">BZ#2024693</a> , <a href="#">BZ#1805717</a> , <a href="#">BZ#1779685</a> , <a href="#">BZ#2053204</a> , <a href="#">BZ#2063140</a> , <a href="#">BZ#2047175</a>
<b>ModemManager</b>	<a href="#">BZ#1996716</a>
<b>NetworkManager</b>	<a href="#">BZ#1980387</a> , <a href="#">BZ#1949127</a> , <a href="#">BZ#2060013</a> , <a href="#">BZ#1931284</a> , <a href="#">BZ#1894877</a> , <a href="#">BZ#2079849</a>
<b>RHCOS</b>	<a href="#">BZ#2008521</a>
<b>WALinuxAgent</b>	<a href="#">BZ#1972101</a>
<b>alsa-lib</b>	<a href="#">BZ#2015863</a>
<b>anaconda</b>	<a href="#">BZ#1951709</a> , <a href="#">BZ#1978264</a> , <a href="#">BZ#2025953</a> , <a href="#">BZ#2009403</a> , <a href="#">BZ#2050140</a> , <a href="#">BZ#1877697</a> , <a href="#">BZ#1914955</a> , <a href="#">BZ#1929105</a> , <a href="#">BZ#1983602</a> , <a href="#">BZ#1997832</a> , <a href="#">BZ#2008792</a>
<b>ansible-collection-microsoft-sql</b>	<a href="#">BZ#2064648</a> , <a href="#">BZ#2064690</a>
<b>ansible-collection-redhat-rhel_mgmt</b>	<a href="#">BZ#2023381</a>
<b>ansible-pcp</b>	<a href="#">BZ#1957566</a>
<b>bash</b>	<a href="#">BZ#2079078</a>
<b>bind</b>	<a href="#">BZ#1984982</a>
<b>binutils</b>	<a href="#">BZ#2030554</a>
<b>boost</b>	<a href="#">BZ#1957950</a>
<b>chrony</b>	<a href="#">BZ#1961131</a>
<b>clevis</b>	<a href="#">BZ#1956760</a>
<b>cloud-init</b>	<a href="#">BZ#2040090</a> , <a href="#">BZ#2042351</a>
<b>cmake</b>	<a href="#">BZ#1957948</a>

组件	票证
<b>container-tools</b>	BZ#2000871
<b>containers-common</b>	<a href="#">BZ#2019901</a>
<b>crash</b>	BZ#1896647
<b>createrepo_c</b>	<a href="#">BZ#2055032</a>
<b>crypto-policies</b>	<a href="#">BZ#2004207</a> , BZ#2013195
<b>cyrus-sasl</b>	<a href="#">BZ#1947971</a> , BZ#1995600
<b>device-mapper-multipath</b>	BZ#2017979, <a href="#">BZ#2017592</a> , BZ#2011699
<b>distribution</b>	BZ#1878583
<b>dnf</b>	<a href="#">BZ#2005305</a> , <a href="#">BZ#2073510</a>
<b>dotnet6.0</b>	BZ#1986211
<b>edk2</b>	BZ#1935497
<b>eigen3</b>	<a href="#">BZ#2032423</a>
<b>fapolicyd</b>	<a href="#">BZ#2032408</a> , BZ#1932225, <a href="#">BZ#2054740</a> , <a href="#">BZ#2070655</a>
<b>fence-agents</b>	<a href="#">BZ#1977588</a>
<b>fetchmail</b>	BZ#1999276
<b>fido-device-onboard</b>	BZ#1989930
<b>firefox</b>	<a href="#">BZ#1764205</a> , <a href="#">BZ#2013247</a>
<b>firewalld</b>	<a href="#">BZ#2029211</a>
<b>freeradius</b>	<a href="#">BZ#1978216</a>
<b>gcc</b>	<a href="#">BZ#1986836</a> , BZ#1481850
<b>gdb</b>	BZ#1870029, BZ#1870031
<b>gfs2-utils</b>	BZ#1616432

组件	票证
<b>gimp</b>	<a href="#">BZ#2047161</a>
<b>git</b>	<a href="#">BZ#1956345</a>
<b>glibc</b>	<a href="#">BZ#2023422</a> , <a href="#">BZ#2024347</a>
<b>gnome-shell-extension-background-logo</b>	<a href="#">BZ#2057150</a>
<b>gnome-shell-extensions</b>	<a href="#">BZ#2031186</a>
<b>gnupg2</b>	<a href="#">BZ#2070722</a> , <a href="#">BZ#2073567</a>
<b>gnutls</b>	<a href="#">BZ#2033220</a> , <a href="#">BZ#1999639</a>
<b>golang</b>	<a href="#">BZ#2014087</a> , <a href="#">BZ#1984110</a>
<b>grafana-pcp</b>	<a href="#">BZ#1993156</a> , <a href="#">BZ#1845592</a>
<b>grafana</b>	<a href="#">BZ#1993215</a>
<b>grub2</b>	<a href="#">BZ#2026579</a>
<b>grubby</b>	<a href="#">BZ#1969362</a>
<b>hostapd</b>	<a href="#">BZ#2019830</a>
<b>ipa</b>	<a href="#">BZ#1952028</a> , <a href="#">BZ#1957736</a> , <a href="#">BZ#1966101</a> , <a href="#">BZ#1988383</a> , <a href="#">BZ#2084180</a> , <a href="#">BZ#2084166</a> , <a href="#">BZ#2057471</a>
<b>iptables</b>	<a href="#">BZ#1945151</a>
<b>javapackages-tools</b>	<a href="#">BZ#1951482</a>
<b>jigawatts</b>	<a href="#">BZ#1972029</a>
<b>jmc-core</b>	<a href="#">BZ#1980981</a>
<b>kdump-anaconda-addon</b>	<a href="#">BZ#1894783</a> , <a href="#">BZ#2017401</a>
<b>kernel-rt</b>	<a href="#">BZ#2002474</a>

组件	票证
内核	BZ#1844416, BZ#1851933, BZ#1780258, BZ#1874195, <a href="#">BZ#1953515</a> , <a href="#">BZ#1960556</a> , BZ#1948340, <a href="#">BZ#1952863</a> , BZ#1978382, <a href="#">BZ#1957818</a> , <a href="#">BZ#2002499</a> , BZ#2050415, <a href="#">BZ#1951951</a> , BZ#1949613, BZ#2036856, BZ#2034490, BZ#1943423, <a href="#">BZ#2054441</a> , BZ#2046472, BZ#2068432, BZ#1997541, BZ#1613522, BZ#1874182, BZ#1995338, BZ#1570255, BZ#2023416, BZ#2021672, BZ#2019593, BZ#2000616, BZ#2013650, BZ#2033080, BZ#2069758, BZ#2059545, BZ#2030922, <a href="#">BZ#1945040</a> , BZ#2073541, BZ#1960467, BZ#2005173
kexec-tools	BZ#1988894, BZ#1895232, <a href="#">BZ#1958452</a> , <a href="#">BZ#2065013</a>
kmod	<a href="#">BZ#1985100</a>
krb5	<a href="#">BZ#2060798</a> , <a href="#">BZ#2068935</a> , <a href="#">BZ#2077450</a>
libburn	<a href="#">BZ#2015861</a>
libcap	<a href="#">BZ#2037215</a>
libgcrypt	<a href="#">BZ#1990059</a>
libmodulemd	<a href="#">BZ#1984403</a>
libreswan	BZ#2017355, <a href="#">BZ#2039877</a>
libseccomp	<a href="#">BZ#2019887</a>
libservicelog	BZ#1869568
libvirt	<a href="#">BZ#2014487</a>
libxcrypt	<a href="#">BZ#2034569</a>
llvm-toolset	BZ#2001107
lorax-templates-rhel	<a href="#">BZ#1961092</a>
lsvpd	BZ#1869564
lvm2	<a href="#">BZ#1899214</a> , <a href="#">BZ#1749513</a> , <a href="#">BZ#2038183</a>
mariadb	<a href="#">BZ#1971248</a>
mod_security_crs	<a href="#">BZ#1947962</a>
nettle	BZ#1986712

组件	票证
<b>nfs-utils</b>	BZ#2059245
<b>nginx</b>	<a href="#">BZ#1953639</a>
<b>nmstate</b>	<a href="#">BZ#1969941</a>
<b>nodejs</b>	<a href="#">BZ#1953491</a>
<b>nss</b>	<a href="#">BZ#2008320</a> , <a href="#">BZ#2099438</a>
<b>numatop</b>	BZ#1874125
<b>nvml</b>	BZ#1874208
<b>opal-prd</b>	BZ#1869560
<b>open-vm-tools</b>	BZ#2037657
<b>opencryptoki</b>	BZ#1869533
<b>openscap</b>	<a href="#">BZ#2041782</a>
<b>openssh</b>	<a href="#">BZ#1952957</a> , <a href="#">BZ#2002734</a> , <a href="#">BZ#1821501</a> , <a href="#">BZ#2087121</a>
<b>openssl</b>	<a href="#">BZ#1990814</a> , <a href="#">BZ#1871147</a> , <a href="#">BZ#1970388</a> , <a href="#">BZ#1975836</a> , <a href="#">BZ#1681178</a> , <a href="#">BZ#1685470</a> , <a href="#">BZ#2053289</a> , <a href="#">BZ#2087253</a> , <a href="#">BZ#2060044</a> , <a href="#">BZ#2071631</a>
<b>osbuild-composer</b>	<a href="#">BZ#2060575</a>
<b>oscap-anaconda-addon</b>	BZ#1893753
<b>ostree</b>	<a href="#">BZ#1961254</a>
<b>p11-kit</b>	BZ#1966680
<b>pacemaker</b>	<a href="#">BZ#1850145</a> , <a href="#">BZ#1443666</a> , <a href="#">BZ#1470834</a> , <a href="#">BZ#1082146</a> , <a href="#">BZ#1376538</a> , <a href="#">BZ#1975388</a>
<b>pcp</b>	<a href="#">BZ#1991764</a> , <a href="#">BZ#1847808</a> , <a href="#">BZ#1981223</a>
<b>pcs</b>	<a href="#">BZ#1290830</a> , <a href="#">BZ#1909901</a> , <a href="#">BZ#1872378</a> , <a href="#">BZ#2018969</a> , <a href="#">BZ#1996067</a>
<b>perl-Module-Signature</b>	BZ#2039361
<b>php</b>	<a href="#">BZ#1949319</a>

组件	票证
<b>pki-core</b>	BZ#2084181
<b>podman</b>	JIRA:RHELPLAN-77549, JIRA:RHELPLAN-75322, JIRA:RHELPLAN-108830, <a href="#">BZ#2027576</a>
<b>powerpc-utils</b>	BZ#1873868
<b>ppc64-diag</b>	BZ#1869567
<b>python-jsonpointer</b>	<a href="#">BZ#1980256</a>
<b>python-podman</b>	<a href="#">BZ#1975462</a>
<b>qemu-kvm</b>	BZ#1940132, BZ#1939509, JIRA:RHELPLAN-75866, BZ#1874187, <a href="#">BZ#1965079</a> , <a href="#">BZ#1951814</a> , <a href="#">BZ#2014229</a> , <a href="#">BZ#2052424</a> , <a href="#">BZ#1817965</a>
<b>redis</b>	<a href="#">BZ#1959756</a>
<b>rhel-system-roles</b>	<a href="#">BZ#1993304</a> , <a href="#">BZ#1993377</a> , <a href="#">BZ#2022461</a> , <a href="#">BZ#1978488</a> , <a href="#">BZ#1984583</a> , <a href="#">BZ#2016517</a> , <a href="#">BZ#2021667</a> , <a href="#">BZ#1986460</a> , <a href="#">BZ#1978752</a> , <a href="#">BZ#1978753</a> , <a href="#">BZ#1990490</a> , <a href="#">BZ#2031555</a> , <a href="#">BZ#2016518</a> , <a href="#">BZ#2054364</a> , <a href="#">BZ#1978773</a> , <a href="#">BZ#2054435</a> , <a href="#">BZ#1999162</a> , <a href="#">BZ#2057657</a> , <a href="#">BZ#2012298</a> , <a href="#">BZ#2021028</a> , <a href="#">BZ#2054367</a> , <a href="#">BZ#2054369</a> , <a href="#">BZ#2057662</a> , <a href="#">BZ#2021665</a> , <a href="#">BZ#2029427</a> , <a href="#">BZ#2004899</a> , <a href="#">BZ#1958964</a> , <a href="#">BZ#1978734</a> , <a href="#">BZ#1978760</a> , <a href="#">BZ#2039106</a> , <a href="#">BZ#2041632</a> , <a href="#">BZ#2058777</a> , <a href="#">BZ#2058645</a> , <a href="#">BZ#2058756</a> , <a href="#">BZ#2071804</a> , <a href="#">BZ#2029634</a> , <a href="#">BZ#2044408</a> , <a href="#">BZ#2029602</a> , <a href="#">BZ#2038957</a> , <a href="#">BZ#2064391</a> , <a href="#">BZ#2004303</a> , <a href="#">BZ#2006230</a> , <a href="#">BZ#2057164</a> , <a href="#">BZ#2021025</a> , <a href="#">BZ#2021676</a> , <a href="#">BZ#2047506</a> , <a href="#">BZ#2050341</a> , <a href="#">BZ#2050419</a> , <a href="#">BZ#1999770</a>
<b>rpm-ostree</b>	<a href="#">BZ#1961324</a>
<b>rpm</b>	BZ#1942549, <a href="#">BZ#1962234</a>
<b>rsyslog</b>	<a href="#">BZ#2027971</a> , <a href="#">BZ#1992155</a>
<b>rust-toolset</b>	BZ#2002885
<b>s390utils</b>	BZ#1932480
<b>samba</b>	<a href="#">BZ#2013578</a> , Jira:RHELDOCS-16612
<b>scap-security-guide</b>	<a href="#">BZ#2028435</a> , <a href="#">BZ#2014561</a> , <a href="#">BZ#2045341</a> , <a href="#">BZ#2038978</a>
<b>selinux-policy</b>	<a href="#">BZ#2055822</a> , <a href="#">BZ#1932752</a> , <a href="#">BZ#2021529</a> , <a href="#">BZ#2064274</a>
<b>shadow-utils</b>	<a href="#">BZ#1859252</a>

组件	票证
<b>scs</b>	BZ#2011537, BZ#1869561
<b>squid</b>	<a href="#">BZ#1990517</a>
<b>sssd</b>	<a href="#">BZ#1949149</a> , <a href="#">BZ#2014249</a> , BZ#1879869, BZ#1737489
<b>strace</b>	<a href="#">BZ#2038965</a>
<b>stratisd</b>	<a href="#">BZ#2041558</a>
<b>stunnel</b>	<a href="#">BZ#2039299</a>
<b>subscription-manager</b>	BZ#1898563, <a href="#">BZ#2049441</a>
<b>sudo</b>	<a href="#">BZ#1981278</a>
<b>swig</b>	BZ#1943580
<b>systemd</b>	BZ#2018112
<b>systemtap</b>	BZ#2083727
<b>tigervnc</b>	<a href="#">BZ#2060308</a>
<b>trace-cmd</b>	BZ#1933980
<b>tuned</b>	<a href="#">BZ#2003838</a>
<b>unbound</b>	<a href="#">BZ#2070495</a>
<b>usbguard</b>	<a href="#">BZ#1986785</a> , <a href="#">BZ#2009226</a>
<b>varnish</b>	<a href="#">BZ#1984185</a>
<b>virt-manager</b>	BZ#1995131
<b>virt-who</b>	<a href="#">BZ#2008215</a> , <a href="#">BZ#2054504</a>
<b>virtio-win</b>	<a href="#">BZ#1969724</a>
<b>wpa_supplicant</b>	<a href="#">BZ#1975718</a>

组件	票证
其他	<p> <a href="#">BZ#2077836</a>, BZ#2019806, BZ#1937651, <a href="#">BZ#2010291</a>, BZ#1941810, BZ#2091643, BZ#1941595, JIRA:RHELPLAN-80758, JIRA:RHELPLAN-80759, JIRA:RHELPLAN-82578, JIRA:RHELPLAN-68364, JIRA:RHELPLAN-78673, JIRA:RHELPLAN-78675, BZ#1940863, <a href="#">BZ#2079313</a>, JIRA:RHELPLAN-100497, BZ#2068532, <a href="#">BZ#2089193</a>, JIRA:RHELPLAN-102009, <a href="#">BZ#2065646</a>, <a href="#">BZ#2088414</a>, JIRA:RHELPLAN-80734, <a href="#">BZ#2013853</a>, JIRA:RHELPLAN-103540, <a href="#">BZ#2019341</a>, <a href="#">BZ#2008558</a>, <a href="#">BZ#2008575</a>, <a href="#">BZ#2009455</a>, JIRA:RHELPLAN-74542, JIRA:RHELPLAN-73678, JIRA:RHELPLAN-84168, JIRA:RHELPLAN-73697, JIRA:RHELPLAN-95126, <a href="#">BZ#2080875</a>, JIRA:RHELPLAN-97899, JIRA:RHELPLAN-100359, JIRA:RHELPLAN-103147, JIRA:RHELPLAN-103146, JIRA:RHELPLAN-79161, <a href="#">BZ#2046325</a>, BZ#2021262, JIRA:RHELPLAN-64576, JIRA:RHELPLAN-65223, <a href="#">BZ#2083036</a>, <a href="#">BZ#2011448</a>, <a href="#">BZ#2019318</a>, JIRA:RHELPLAN-101240, JIRA:RHELPLAN-101241, JIRA:RHELPLAN-101242, JIRA:RHELPLAN-101246, JIRA:RHELPLAN-101247, JIRA:RHELPLAN-102552, JIRA:RHELPLAN-99892, BZ#2027596, JIRA:RHELPLAN-119000, BZ#1940653, JIRA:RHELPLAN-95056, <a href="#">BZ#2054401</a>, JIRA:RHELPLAN-113994, <a href="#">BZ#2059183</a>, JIRA:RHELPLAN-74543, JIRA:RHELPLAN-99889, JIRA:RHELPLAN-99890, JIRA:RHELPLAN-100032, JIRA:RHELPLAN-100034, JIRA:RHELPLAN-101141, JIRA:RHELPLAN-100020, BZ#2069501, <a href="#">BZ#2070506</a>, JIRA:RHELPLAN-117903, JIRA:RHELPLAN-98617, JIRA:RHELPLAN-103855, <a href="#">BZ#2091653</a>, <a href="#">BZ#2082306</a>, JIRA:RHELPLAN-65217, <a href="#">BZ#2020529</a>, <a href="#">BZ#2030412</a>, BZ#2046653, JIRA:RHELPLAN-103993, JIRA:RHELPLAN-122345, BZ#1927780, JIRA:RHELPLAN-110763, BZ#1935544, <a href="#">BZ#2089200</a>, JIRA:RHELPLAN-15509, JIRA:RHELPLAN-99136, JIRA:RHELPLAN-103232, BZ#1899167, <a href="#">BZ#1979521</a>, JIRA:RHELPLAN-100087, JIRA:RHELPLAN-100639, JIRA:RHELPLAN-10304, <a href="#">BZ#2058153</a>, JIRA:RHELPLAN-113995, JIRA:RHELPLAN-121048, JIRA:RHELPLAN-98983, BZ#1640697, BZ#1697896, <a href="#">BZ#2020026</a>, <a href="#">BZ#2047713</a>, JIRA:RHELPLAN-109067, JIRA:RHELPLAN-115603, JIRA:RHELPLAN-96940, JIRA:RHELPLAN-117234, JIRA:RHELPLAN-119001, JIRA:RHELPLAN-119852, BZ#2077767, BZ#2053598, JIRA:RHELPLAN-121180, <a href="#">BZ#2082303</a>, JIRA:RHELPLAN-121049 </p>

## 附录 B. 致谢

感谢以下红帽员工在 RHEL 9 就绪挑战中提供了反馈：

- Buland Singh
- Pradeep Jagtap
- Omkar Andhekar
- Ju Ke
- Suresh Jagtap
- Prijesh Patel
- Nikhil Suryawanshi
- Amit Yadav
- Pranav Lawate
- John Pittman

## 附录 C. 修改历史记录

### 0.1-30

2024 年 6 月 11 日星期二, Brian Angelica ([bangelic@redhat.com](mailto:bangelic@redhat.com))

- 添加已弃用的功能 [RHELDOCS-18049](#) (Shells 和命令行工具)。

### 0.1-29

2024 年 6 月 11 日星期二, Brian Angelica ([bangelic@redhat.com](mailto:bangelic@redhat.com))

- 添加了一个已知问题 [JIRA:RHEL-24847](#) (Shells 和命令行工具)。

### 0.1-28

2024 年 3 月 14 日星期四, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个已知问题 [JIRA:RHEL-25967](#) (内核)

### 0.1-27

2024 年 2 月 14 日星期三, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个改进 [JIRA:RHELDOCS-17553](#) (身份管理)

### 0.1-26

2024 年 2 月 1 日星期四, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个 KI [BZ#1834716](#) (安全)

### 0.1-25

2023 年 11 月 13 日星期一, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个技术预览 [JIRA:RHELDOCS-17040](#) (虚拟化)

### 0.1-24

2023 年 11 月 10 日星期五, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 更新了对 RHEL 文档提供反馈的模块。

### 0.1-23

2023 年 11 月 10 日星期五, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个技术预览 [JIRA:RHELDOCS-17050](#) (虚拟化)。

### 0.1-22

2023 年 10 月 13 日星期五, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个技术预览 [JIRA:RHELDOCS-16861](#) (容器)。

### 0.1-21

2023 年 9 月 8 日, Marc Muehlfeld ([mmuehlfeld@redhat.com](mailto:mmuehlfeld@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", 以在 JIRA 中反映 RHEL。

#### 0.1-20

2023 年 8 月 17 日, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个 Enh [BZ#2136937](#) (Plumbers)。

#### 0.1-19

2023 年 8 月 7 日, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个 KI [BZ#2214130](#) (CS)。

#### 0.1-18

2023 年 8 月 2 日, Marc Muehlfeld ([mmuehlfeld@redhat.com](mailto:mmuehlfeld@redhat.com))

- 更新了一个已弃用的功能发行注记 [BZ#1894877](#) (NetworkManager)。

#### 0.1-17

2023 年 6 月 19 日星期一, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 修复了 KI [BZ#2068935](#) (IdM)中的拼写错误。

#### 0.1-16

2023 年 5 月 18 日, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个改进 [BZ#2053642](#) (文件系统和存储)。

#### 0.1-15

2023 年 5 月 17 日, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 使用有关 EOL 的信息更新 `deprecated-packages.adoc`。

#### 0.1-14

2023 年 5 月 11 日, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个改进 [BZ#2190045](#) (安装程序)。

#### 0.1-13

2023 年 4 月 27 日, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个已知问题 [JIRA:RHELPLAN-155168](#) (身份管理)。

#### 0.1-12

2023 年 4 月 13 日, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 修复了新特性中的一个中断的链接 [JIRA:RHELPLAN-84168](#) (容器)。

#### 0.1-11

2023 年 3 月 1 日星期三, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- [BZ#2091643](#) (内核)的修改后的文档文本。

**0.1-10**

2023 年 2 月 20 日星期一, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 将信息添加到有关 SAP 环境的 "In-place upgrade from RHEL 8 to RHEL 9" 中。

**0.1-9**

2023 年 1 月 18 日星期三, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个已知问题文档文本 [BZ#2083106](#) (内核)。

**0.1-8**

2023 年 1 月 17 日星期二, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 更新了一个技术预览文档文本 [BZ#2084181](#) (身份管理)。

**0.1-7**

2023 年 1 月 16 日星期一, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个已知问题文档文本 [BZ#2149172](#) (内核)。

**0.1-6**

2022 年 12 月 22 日星期四, Gabi Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 更新了一个已知问题文档文本 [BZ#1960467](#) (图形基础架构)。

**0.1-5**

2022 年 12 月 8 日星期四, Marc Muehlfeld ([mmuehlfeld@redhat.com](mailto:mmuehlfeld@redhat.com))

- 添加了一个已知问题 [BZ#2151040](#) (网络)。

**0.1-4**

2022 年 11 月 15 日星期二, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 更新了 [原位升级](#) 部分。

**0.1-3**

2022 年 9 月 23 日星期五, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个弃用的功能 [BZ#2074598](#) (内核)。

**0.1-2**

2022 年 9 月 21 日, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 删除了一个已知问题 [BZ#2060798](#) (身份管理)。
- 添加了一个 bug 修复 [BZ#2060798](#) (身份管理)。

**0.1-1**

2022 年 9 月 12 日星期一, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 更新了 `proc_providing-feedback-on-red-hat-documentation.adoc`。

- 添加了一个改进 [BZ#2119694](#) (安全性)。

#### 0.1-0

2022 年 8 月 22 日星期一, Lenka Špačková([lspackova@redhat.com](mailto:lspackova@redhat.com))

- 添加了已弃用的功能 [BZ#2069279](#) 和 [BZ#2106816](#) (容器)。
- 使用有关 z-stream 修复 (容器) 的信息更新了 [JIRA-RHELPLAN-121180](#)。

#### 0.0-9

2022 年 8 月 10 日星期三, Lenka Špačková([lspackova@redhat.com](mailto:lspackova@redhat.com))

- 添加了一个已知问题 [BZ#1991500](#) (动态编程语言、web 和数据库服务器)。

#### 0.0-8

2022 年 8 月 4 日星期四, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个增强 [JIRA-RHELPLAN-118914](#) (容器)。
- 添加了一个已知问题 [BZ#2105162](#) (安全)。
- 添加了一个已知问题 [BZ#1960467](#) (图形基础设施)。

#### 0.0-7

2022 年 7 月 28 日星期四, Lenka Špačková ([lspackova@redhat.com](mailto:lspackova@redhat.com))

- 添加了一个增强 [BZ#2099438](#) (安全)。
- 添加了一个已知问题 [BZ#2087253](#) (安全)。
- 在[分发](#) 中扩展了有关应用程序流的信息。

#### 0.0-6

2022 年 7 月 11 日星期一, Lenka Špačková([lspackova@redhat.com](mailto:lspackova@redhat.com))

- 添加了一个已知问题 [BZ#2077450](#)。
- 添加了一个增强 [BZ#2091653](#)。
- 添加了一个 bug 修复 [BZ#2006230](#)。

#### 0.0-5

2022 年 6 月 29 日星期三, Lenka Špačková([lspackova@redhat.com](mailto:lspackova@redhat.com))

- 添加了已知问题 [BZ#2087121](#), [BZ#2073567](#), [BZ#2083727](#) 和 [BZ#2005173](#)。
- 添加了一个增强 [BZ#2091643](#)。

#### 0.0-4

2022 年 6 月 1 日星期三, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 添加了一个已知问题 [BZ#2027576](#)。

**0.0-3**

2022 年 5 月 24 日星期二, Gabriela Fialová([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 更新了前 10 个流行的客户门户网站实验室的列表。
- 添加并重新发布了已弃用的功能 [BZ#2089200](#) (网络)。

**0.0-2**

2022 年 5 月 18 日星期三, Gabriela Fialová ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 发布 Red Hat Enterprise Linux 9.0 发行注记。

**0.0-1**

Wed Nov 03, 2021, Lenka Špačková ([lspackova@redhat.com](mailto:lspackova@redhat.com))

- 发布 Red Hat Enterprise Linux 9.0 Beta 发行注记。