



# Red Hat Enterprise Linux 8

## 8.7 リリースノート

Red Hat Enterprise Linux 8.7 リリースノート



# Red Hat Enterprise Linux 8 8.7 リリースノート

---

Red Hat Enterprise Linux 8.7 リリースノート

## 法律上の通知

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.

## 概要

このリリースノートでは、Red Hat Enterprise Linux 8.7 での改良点および実装された追加機能の概要、このリリースにおける既知の問題などを説明します。また、重要なバグ修正、テクニカルレビュー、非推奨機能などの詳細も説明します。Red Hat Enterprise Linux のインストールは、Installation を参照してください。

## 目次

多様性を受け入れるオープンソースの強化 .....	5
RED HAT ドキュメントへのフィードバック (英語のみ) .....	6
<b>第1章 概要</b> .....	<b>7</b>
1.1. RHEL 8.7 における主な変更点	7
1.2. インプレースアップグレードおよび OS 移行	10
1.3. RED HAT CUSTOMER PORTAL LABS	11
1.4. 関連情報	11
<b>第2章 アーキテクチャー</b> .....	<b>13</b>
<b>第3章 RHEL 8 のコンテンツの配布</b> .....	<b>14</b>
3.1. インストール	14
3.2. リポジトリ	14
3.3. アプリケーションストリーム	15
3.4. YUM/DNF を使用したパッケージ管理	15
<b>第4章 新機能</b> .....	<b>16</b>
4.1. インストーラーおよびイメージの作成	16
4.2. シェルおよびコマンドラインツール	17
4.3. インフラストラクチャーサービス	20
4.4. セキュリティー	21
4.5. ネットワーキング	23
4.6. カーネル	25
4.7. ブートローダー	26
4.8. ファイルシステムおよびストレージ	26
4.9. 高可用性およびクラスター	27
4.10. 動的プログラミング言語、WEB サーバー、およびデータベースサーバー	29
4.11. コンパイラーおよび開発ツール	32
4.12. IDENTITY MANAGEMENT	40
4.13. グラフィックインフラストラクチャー	43
4.14. WEB コンソール	44
4.15. RED HAT ENTERPRISE LINUX システムロール	45
4.16. 仮想化	50
4.17. クラウド環境の RHEL	51
4.18. コンテナ	52
<b>第5章 外部カーネルパラメーターへの重要な変更</b> .....	<b>55</b>
新しいカーネルパラメーター	55
更新されたカーネルパラメーター	57
<b>第6章 デバイスドライバー</b> .....	<b>60</b>
6.1. 新しいドライバー	60
6.2. 更新されたドライバー	61
<b>第7章 利用可能な BPF 機能</b> .....	<b>62</b>
<b>第8章 バグ修正</b> .....	<b>76</b>
8.1. インストーラーおよびイメージの作成	76
8.2. ソフトウェア管理	77
8.3. シェルおよびコマンドラインツール	77
8.4. インフラストラクチャーサービス	79
8.5. セキュリティー	79

8.6. ネットワーキング	80
8.7. カーネル	81
8.8. ブートローダー	81
8.9. 高可用性およびクラスター	82
8.10. 動的プログラミング言語、WEB サーバー、およびデータベースサーバー	82
8.11. コンパイラーおよび開発ツール	82
8.12. IDENTITY MANAGEMENT	84
8.13. デスクトップ	85
8.14. グラフィックインフラストラクチャー	85
8.15. WEB コンソール	85
8.16. RED HAT ENTERPRISE LINUX システムロール	86
8.17. 仮想化	88
8.18. クラウド環境の RHEL	88
8.19. コンテナ	89
<b>第9章 テクノロジーレビュー</b>	<b>91</b>
9.1. シェルおよびコマンドラインツール	91
9.2. ネットワーキング	91
9.3. カーネル	93
9.4. ファイルシステムおよびストレージ	94
9.5. 高可用性およびクラスター	96
9.6. IDENTITY MANAGEMENT	97
9.7. デスクトップ	100
9.8. グラフィックインフラストラクチャー	100
9.9. 仮想化	100
9.10. クラウド環境の RHEL	102
9.11. コンテナ	102
<b>第10章 非推奨になった機能</b>	<b>104</b>
10.1. インストーラーおよびイメージの作成	104
10.2. ソフトウェア管理	105
10.3. シェルおよびコマンドラインツール	105
10.4. セキュリティー	106
10.5. ネットワーキング	108
10.6. カーネル	110
10.7. ブートローダー	111
10.8. ファイルシステムおよびストレージ	111
10.9. 高可用性およびクラスター	113
10.10. 動的プログラミング言語、WEB サーバー、およびデータベースサーバー	113
10.11. コンパイラーおよび開発ツール	113
10.12. IDENTITY MANAGEMENT	114
10.13. デスクトップ	117
10.14. グラフィックインフラストラクチャー	117
10.15. WEB コンソール	118
10.16. RED HAT ENTERPRISE LINUX システムロール	118
10.17. 仮想化	119
10.18. コンテナ	120
10.19. 非推奨のパッケージ	121
10.20. 非推奨のデバイスおよび非保守のデバイス	158
<b>第11章 既知の問題</b>	<b>162</b>
11.1. インストーラーおよびイメージの作成	162
11.2. サブスクリプションの管理	164
11.3. ソフトウェア管理	164

---

11.4. シェルおよびコマンドラインツール	165
11.5. インフラストラクチャーサービス	166
11.6. セキュリティー	167
11.7. ネットワーク	172
11.8. カーネル	174
11.9. ブートローダー	180
11.10. ファイルシステムおよびストレージ	180
11.11. 動的プログラミング言語、WEB サーバー、およびデータベースサーバー	182
11.12. IDENTITY MANAGEMENT	183
11.13. デスクトップ	187
11.14. グラフィックインフラストラクチャー	187
11.15. WEB コンソール	189
11.16. RED HAT ENTERPRISE LINUX システムロール	189
11.17. 仮想化	190
11.18. クラウド環境の RHEL	193
11.19. サポート性	195
11.20. コンテナ	196
<b>第12章 国際化</b> .....	<b>198</b>
12.1. RED HAT ENTERPRISE LINUX 8 の多言語	198
12.2. RHEL 8 における国際化の主な変更点	198
<b>付録A コンポーネント別のチケットリスト</b> .....	<b>200</b>
<b>付録B 改訂履歴</b> .....	<b>208</b>



## 多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みは膨大な作業を要するため、今後の複数のリリースで段階的に用語の置き換えを実施して参ります。詳細は、[Red Hat CTO である Chris Wright のメッセージ](#) をご覧ください。

## RED HAT ドキュメントへのフィードバック (英語のみ)

Red Hat ドキュメントに関するご意見やご感想をお寄せください。また、改善点があればお知らせください。

### Jira からのフィードバック送信 (アカウントが必要)

1. [Jira](#) の Web サイトにログインします。
2. 上部のナビゲーションバーで **Create** をクリックします。
3. **Summary** フィールドにわかりやすいタイトルを入力します。
4. **Description** フィールドに、ドキュメントの改善に関するご意見を記入してください。ドキュメントの該当部分へのリンクも追加してください。
5. ダイアログの下部にある **Create** をクリックします。

# 第1章 概要

## 1.1. RHEL 8.7 における主な変更点

### インストーラーおよびイメージの作成

以下は、RHEL 8.7-GA の Image Builder に関する重要事項です。

- オンプレミスの Image Builder は以下をサポートするようになりました。
  - GCP へのイメージのアップロード
  - `/boot` パーティションのカスタマイズ
  - コンテナイメージのレジストリーへの直接プッシュ
  - ユーザーによるイメージ作成プロセス中のブループリントのカスタマイズ

詳細は、「[インストーラーおよびイメージの作成](#)」を参照してください。

### セキュリティ

**scap-security-guide** (SSG) パッケージで利用可能な Red Hat Enterprise Linux 8 プロファイル用の DISA STIG は、DISA のコンテンツとの整合性が向上しました。これにより、SSG 修復後に DISA コンテンツに対する調査結果が少なくなります。

**scap-security-guide** (SSG) パッケージで利用可能な Center for Internet Security (CIS) プロファイルが CIS Red Hat Enterprise Linux 8 Benchmark バージョン 2.0.0 と連携するようになりました。このバージョンのベンチマークでは、新しい要件が追加され、関連性がなくなった要件が削除され、いくつかの既存の要件が並べ替えられました。この更新は、関連するルールの参照とそれぞれのプロファイルの精度に影響を与えます。

システム設定と **clevis-luks-systemd** サブパッケージの変更により、Clevis 暗号化クライアントは、デプロイメントプロセス中に **systemctl enable clevis-luks-askpass.path** コマンドを使用せずに、起動プロセスの後半にマウントされる LUKS 暗号化ボリュームもロック解除できるようになりました。

詳細は、[新機能 - セキュリティ](#) を参照してください。

### シェルおよびコマンドラインツール

RHEL 8.7 では、新しいパッケージ **xmlstarlet** が導入されました。XMLStarlet を使用すると、XML ファイルを解析、変換、クエリー、検証、および編集できます。

次のコマンドラインツールが RHEL 8.7 で更新されました。

- **openssl** をバージョン 3.1.0 に更新
- **powerpc-utils** をバージョン 1.3.10 に更新
- **libva** をバージョン 2.13.0 に更新

詳細については、[新機能 - シェルおよびコマンドラインツール](#) を参照してください。

### インフラストラクチャーサービス

RHEL 8.7 では、次のインフラストラクチャーサービスツールが更新されました。

- **chrony** をバージョン 4.2 に更新
- **unbound** をバージョン 1.16.2 に更新

詳細については、[新機能 - インフラストラクチャーサービス](#) を参照してください。

### 動的プログラミング言語、Web サーバー、およびデータベースサーバー

以下のコンポーネントの後続のバージョンが、新しいモジュールストリームとして利用できるようになりました。

- Ruby 3.1
- Mercurial 6.2
- Node.js 18

さらに、Redis 6 はバージョン 6.2.7 にアップグレードされました。

詳細は、[新機能 - 動的プログラミング言語、Web サーバー、およびデータベースサーバー](#) を参照してください。

### コンパイラおよび開発ツール

#### パフォーマンスツールとデバッガーの更新

RHEL 8.7 では、以下のパフォーマンスツールおよびデバッガーが更新されました。

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

#### 更新されたパフォーマンスモニタリングツール

RHEL 8.7 では、以下のパフォーマンス監視ツールが更新されました。

- PCP 5.3.7
- Grafana 7.5.13

#### 更新されたコンパイラツールセット

RHEL 8.7 では、以下のコンパイラツールセットが更新されました。

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

詳しくは [新機能 - コンパイラおよび開発ツール](#) をご覧ください。

### RHEL 8 の Java 実装

RHEL 8 AppStream リポジトリには、以下が含まれます。

- **java-17-openjdk** パッケージ。OpenJDK 17 Java Runtime Environment および OpenJDK 17 Java Software Development Kit を提供します。
- **java-11-openjdk** パッケージ。OpenJDK 11 Java Runtime Environment および OpenJDK 11 Java Software Development Kit を提供します。

- **java-1.8.0-openjdk** パッケージ。OpenJDK 8 Java Runtime Environment および OpenJDK 8 Java Software Development Kit を提供します。

詳細は、[OpenJDK のドキュメント](#) を参照してください。

## Java ツール

RHEL 8.7 では、新しいモジュールストリームとして **Maven 3.8** が導入されています。

詳細については、[New features - Compilers and development tools](#) を参照してください。

## Identity Management

RHEL 8.7 の Identity Management (IdM) にはテクノロジープレビューが導入されており、OAuth 2 Device Authorization Grant フローをサポートする外部 ID プロバイダー (IdP) にユーザー認証を委任できます。これらのユーザーが SSSD で認証されると、外部 IdP で認証と許可が完了すると、Kerberos チケットを使用して RHEL IdMSingle Sign-On 機能を受け取ります。

詳細は、[テクノロジープレビュー - Identity Management](#) を参照してください。

## Red Hat Enterprise Linux システムロール

8.7 RHEL システムロールの注目すべき新機能:

- RHEL システムロールは、ファクト収集が無効になっている Playbook でも利用できるようになりました。
- **ha\_cluster** ロールは、SBD フェンシング、Corosync 設定、およびバンドルリソースの設定をサポートするようになりました。
- **network** ロールは、ルーティングルールのネットワークオプションの設定と **nmstate API** を使用したネットワーク設定のサポートするようになり、ユーザーは IPoIB 機能を使用して接続を作成できるようになりました。
- **microsoft.sql.server** ロールには、高可用性クラスターの設定を制御する変数、ファイアウォールポートを自動的に管理するための変数、管理対象ノードで **mssql\_tls\_cert** および **mssql\_tls\_private\_key** 値を検索する変数など、新しい変数があります。
- **logging** ロールは、さまざまな新しいオプションをサポートします。たとえば、ファイル入力の **startmsg.regex** と **endmsg.regex**、または **template**、**severity**、および **facility** オプションです。
- **storage** ロールには、シンプロビジョニングされたボリュームのサポートが追加され、ロールはデフォルトで詳細レベルが低くなりました。
- **sshd** ロールは、ドロップインディレクトリーの include ディレクティブを検証し、`/etc/ssh/sshd_config` を通じてロールを管理できるようになりました。
- **metrics** ロールは、postfix パフォーマンスデータをエクスポートできるようになりました。
- **postfix** ロールには、以前の設定を上書きする新しいオプションが追加されました。
- **firewall** ロールは、masquerade または icmp\_block\_inversion を設定する際に state パラメーターを必要としません。**firewall** ロールでは、absent および present 状態を使用してサービスを追加、更新、または削除できるようになりました。また、このロールは Ansible ファクトを提供し、PCI デバイス ID を使用してインターフェイスの追加や削除も可能になりました。**firewall** ロールには、以前の設定を上書きする新しいオプションがあります。
- **selinux** ロールには、**seuser** および **selevel** パラメーターの設定が含まれるようになりました。

## 1.2. インプレースアップグレードおよび OS 移行

### RHEL 7 から RHEL 8 へのインプレースアップグレード

現在、考えられるインプレースアップグレードパスは以下のとおりです。

- 64 ビット Intel、IBM POWER 8 (little endian)、IBM Z アーキテクチャーでの RHEL 7.9 から RHEL 8.4 および RHEL 8.6 へのアップグレード。
- カーネルバージョン 4.14 を必要とするアーキテクチャー (IBM POWER 9 (リトルエンディアン) および IBM Z (Structure A)) での RHEL 7.6 から RHEL 8.8.4 のアップグレード。これは、これらのアーキテクチャーの最終のインプレースアップグレードパスです。
- 64 ビット Intel アーキテクチャーの SAP HANA 搭載システム上における、RHEL 7.9 から RHEL 8.2 および RHEL 8.6 へのアップグレード。

RHEL 8.6 へのアップグレード後もシステムがサポートされるようにするには、最新の RHEL 8.7 バージョンに更新するか、RHEL 8.6 Extended Update Support (EUS) リポジトリが有効になっていることを確認します。

詳細は、[Supported in-place upgrade paths for Red Hat Enterprise Linux](#) を参照してください。

インプレースアップグレードの実行方法は、[RHEL 7 から RHEL 8 へのアップグレード](#) を参照してください。

SAP 環境があるシステムでインプレースアップグレードを実行する手順については、[How to in-place upgrade SAP environments from RHEL 7 to RHEL 8](#) を参照してください。



#### 注記

IBM POWER 9 (リトルエンディアン) および IBM Z (structure A) アーキテクチャー用の RHEL 7.6 のインプレースアップグレードを成功させるには、特定の Leapp データを手動でダウンロードする必要があります。詳細は、ナレッジベースの記事 [Leapp data snapshots for an in-place upgrade](#) を参照してください。

主な機能拡張は、次のとおりです。

- Red Hat Update Infrastructure (RHUI) を使用して、Microsoft Azure で SAP Apps システムのインプレースアップグレードが可能になりました。
- Red Hat Update Infrastructure (RHUI) を使用する Google Cloud Platform でインプレースアップグレードが可能になりました。

### RHEL 6 から RHEL 8 へのインプレースアップグレード

RHEL 6.10 から RHEL 8 にアップグレードするには、[RHEL 6 から RHEL 8 へのアップグレード](#) の手順に従います。

### RHEL 8 から RHEL 9 へのインプレースアップグレード

Leapp ユーティリティーを使用して RHEL 8 から RHEL 9 へのインプレースアップグレードを行う方法は、[RHEL 8 から RHEL 9 へのアップグレード](#) を参照してください。RHEL 8 と RHEL 9 の主な相違点は、[RHEL 9 の導入における検討事項](#) を参照してください。

### 別の Linux ディストリビューションから RHEL への移行

CentOS Linux 8 または Oracle Linux 8 を使用している場合は、Red Hat がサポートする

**Convert2RHEL** ユーティリティーを使用してオペレーティングシステムを RHEL 8 に変換できます。詳細は、[RPM ベースの Linux ディストリビューションから RHEL への変換](#) を参照してください。

CentOS Linux または Oracle Linux の旧バージョン (バージョン 6 または 7) を使用している場合は、お使いのオペレーティングシステムを RHEL に移行してから、RHEL 8 へのインプレースアップグレードを実行できます。CentOS Linux 6 および Oracle Linux 6 変換は、サポート対象外の **Convert2RHEL** ユーティリティーを使用することに注意してください。サポートされていない変換の詳細については、[How to perform an unsupported conversion from a RHEL-derived Linux distribution to RHEL](#) を参照してください。

Red Hat が他の Linux ディストリビューションから RHEL への移行は、[Convert2RHEL サポートポリシー](#) を参照してください。

### 1.3. RED HAT CUSTOMER PORTAL LABS

Red Hat Customer Portal Labs は、カスタマーポータル内のセクションにあるツールセットで、<https://access.redhat.com/labs/> から入手できます。Red Hat Customer Portal Labs のアプリケーションは、パフォーマンスの向上、問題の迅速なトラブルシューティング、セキュリティ問題の特定、複雑なアプリケーションの迅速なデプロイメントおよび設定に役立ちます。最も一般的なアプリケーションには、以下のものがあります。

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

### 1.4. 関連情報

- 他のバージョンと比較した Red Hat Enterprise Linux 8.0 の **機能および制限** は、Red Hat ナレッジベースの記事 [Red Hat Enterprise Linux technology capabilities and limits](#) を参照してください。
- Red Hat Enterprise Linux の **ライフサイクル** に関する情報は [Red Hat Enterprise Linux のライフサイクル](#) を参照してください。

- RHEL 8 の [パッケージリスト](#) は、[パッケージマニフェスト](#) を参照してください。
- 削除された機能を含む主な RHEL 7 と RHEL 8 の相違点は、[RHEL 8 の導入における考慮事項](#) で説明されています。
- RHEL 7 から RHEL 8 へのインプレースアップグレードを実行する方法は、[Upgrading from RHEL 7 to RHEL 8](#) を参照してください。
- すべての RHEL サブスクリプションで、既知の技術問題の特定、検証、および解決をプロアクティブに行う Red Hat Insights サービスが利用できるようになりました。Red Hat Insights クライアントをインストールし、システムをサービスに登録する方法は、[Red Hat Insights を使い始める](#) ページを参照してください。

## 第2章 アーキテクチャー

Red Hat Enterprise Linux 8.7 には、カーネルバージョン 4.18.0-425 が同梱されており、以下のアーキテクチャーに対応します。

- AMD アーキテクチャーおよび Intel 64 ビットアーキテクチャー
- 64 ビット ARM アーキテクチャー
- IBM Power Systems (リトルエンディアン)
- 64 ビット IBM Z

各アーキテクチャーに適切なサブスクリプションを購入してください。詳細は [Get Started with Red Hat Enterprise Linux - additional architectures](#) を参照してください。利用可能なサブスクリプションのリストは、カスタマーポータル [サブスクリプションの使用状況](#) を参照してください。

## 第3章 RHEL 8 のコンテンツの配布

### 3.1. インストール

Red Hat Enterprise Linux 8 は、ISO イメージを使用してインストールします。AMD64、Intel 64 ビット、64 ビット ARM、IBM Power Systems、IBM Z アーキテクチャーで、以下の 2 種類のインストールメディアが利用できます。

- Binary DVD ISO - BaseOS リポジトリおよび AppStream リポジトリが含まれ、リポジトリを追加しなくてもインストールを完了できる完全インストールイメージです。



#### 注記

インストール用 ISO イメージのサイズは複数 GB であるため、光学メディア形式には適合しない場合があります。インストール ISO イメージを使用して起動可能なインストールメディアを作成する場合は、USB キーまたは USB ハードドライブを使用することが推奨されます。Image Builder ツールを使用すれば、RHEL イメージをカスタマイズできます。Image Builder の詳細は [RHEL システムイメージのカスタマイズの作成](#) を参照してください。

- Boot ISO - インストールプログラムを起動するのに使用する最小限の ISO ブートイメージです。このオプションでは、ソフトウェアパッケージをインストールするのに、BaseOS リポジトリおよび AppStream リポジトリにアクセスする必要があります。リポジトリは、Binary DVD ISO イメージに含まれます。

ISO イメージのダウンロード、インストールメディアの作成、および RHEL インストールの完了の手順については [標準的な RHEL 8 インストールの実行](#) ドキュメントを参照してください。自動化したキックスタートインストールなどの高度なトピックは [高度な RHEL 8 インストールの実行](#) を参照してください。

ベース RHEL インストールで RPM によって作成されるユーザーとグループのリスト、およびこのリストを取得する手順は、[ベース RHEL インストールのすべてのユーザーとグループはどのようなものですか?](#) を参照してください。ナレッジベースの記事。

### 3.2. リポジトリ

Red Hat Enterprise Linux 8 は、2 つのメインリポジトリで配布されています。

- BaseOS
- AppStream

基本的な RHEL インストールにはどちらのリポジトリも必要で、すべての RHEL サブスクリプションで利用できます。

BaseOS リポジトリのコンテンツは、すべてのインストールのベースとなる、基本的な OS 機能のコアセットを提供します。このコンテンツは RPM 形式で提供されており、RHEL の以前のリリースと同様のサポート条件が適用されます。BaseOS から配布されるパッケージのリストは [パッケージマニフェスト](#) を参照してください。

アプリケーションストリーム (AppStream) リポジトリのコンテンツには、さまざまなワークロードとユースケースに対応するために、ユーザー空間アプリケーション、ランタイム言語、およびデータベースが含まれています。Application Streams は、[モジュール](#) と呼ばれる RPM 形式への拡張、または

Software Collections として通常の RPM 形式で利用できます。AppStream で利用可能なパッケージのリストは、[パッケージマニフェスト](#) を参照してください。

また、CodeReady Linux Builder リポジトリーは、すべての RHEL サブスクリプションで利用できます。このリポジトリーは、開発者向けの追加パッケージを提供します。CodeReady Linux Builder リポジトリーに含まれるパッケージには対応しません。

RHEL 8 リポジトリーの詳細は [パッケージマニフェスト](#) を参照してください。

### 3.3. アプリケーションストリーム

Red Hat Enterprise Linux 8 では、アプリケーションストリームの概念が導入されました。ユーザー空間コンポーネントのバージョンが複数配信され、オペレーティングシステムのコアパッケージよりも頻繁に更新されるようになりました。これによりプラットフォームや特定のデプロイメントの基本的な安定性に影響を及ぼすことなく、Red Hat Enterprise Linux をカスタマイズする柔軟性が向上します。

アプリケーションストリームとして使用できるコンポーネントは、モジュールまたは RPM パッケージとしてパッケージ化され、RHEL 8 の AppStream リポジトリーを介して配信されます。各 Application Stream コンポーネントには、RHEL 8 と同じか、より短いライフサイクルが指定されています。詳細は [Red Hat Enterprise Linux のライフサイクル](#) を参照してください。

モジュールは、論理ユニット (アプリケーション、言語スタック、データベース、またはツールセット) を表すパッケージの集まりです。これらのパッケージはまとめてビルドされ、テストされ、そしてリリースされます。

モジュールストリームは、アプリケーションストリームコンポーネントのバージョンを表します。たとえば、**postgresql:10** のデフォルトのストリーム以外に、**postgresql** モジュールでは、PostgreSQL データベースサーバーの複数のストリーム (バージョン) を利用できます。システムにインストールできるモジュールストリームは1つだけです。複数のコンテナで異なるバージョンを使用できます。

詳細なモジュールコマンドは [ユーザー空間コンポーネントのインストール、管理、および削除](#) を参照してください。AppStream で利用可能なモジュールのリストは、[Package manifest](#) を参照してください。

### 3.4. YUM/DNF を使用したパッケージ管理

Red Hat Enterprise Linux 8 へのソフトウェアのインストールは、**DNF** テクノロジーをベースとした **YUM** ツールにより行われます。以前のメジャーバージョンの RHEL との一貫性を保つために、**yum** の用語の使用が意図的に準拠しています。ただし、**yum** の代わりに **dnf** を呼び出すと、**yum** は互換性のために **dnf** のエイリアスであるため、コマンドが期待どおりに動作します。

詳細は、以下のドキュメントを参照してください。

- [ユーザー空間コンポーネントのインストール、管理、および削除](#)
- [RHEL 8 の導入における検討事項](#)

## 第4章 新機能

ここでは、Red Hat Enterprise Linux 8.7 に追加された新機能および主要な機能拡張を説明します。

### 4.1. インストーラーおよびイメージの作成

#### インストーラーで自動 FCP SCSI LUN スキャンがサポートされました

インストーラーは、IBM Z システムに FCP SCSI LUN をアタッチする際に、自動 LUN スキャンを使用できるようになりました。自動 LUN スキャンは、`zfcplib.allow_lun_scan` カーネルモジュールパラメーターで無効にされていない場合、NPIV モードで動作する FCP デバイスで使用できます。これは、デフォルトで有効になっています。これは、指定されたデバイスバス ID を持つ FCP デバイ스에接続されたストレージエリアネットワークで見つかったすべての SCSI デバイスへのアクセスを提供します。WWPN と FCP LUN を指定する必要はなくなり、FCP デバイスバス ID のみ指定する必要があります。

(BZ#1497089)

#### Image Builder オンプレミスが /boot パーティションのカスタマイズをサポートするようになりました

Image Builder オンプレミスバージョンは、カスタムの `/boot` マウントポイントパーティションサイズを使用したイメージのビルドをサポートするようになりました。ブループリントのカスタマイズで `/boot` マウントポイントパーティションのサイズを指定し、デフォルトのブートパーティションサイズが小さすぎる場合に `/boot` パーティションのサイズを増やすことができます。以下に例を示します。

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

(JIRA:RHELPLAN-130379)

#### Image Builder オンプレミスで GCP へのイメージのアップロードがサポートされました

この機能強化により、Image Builder CLI を使用して `gce` イメージを作成し、イメージのアップロードに使用するユーザーまたはサービスアカウントのクレデンシャルを指定できます。その結果、Image Builder はイメージを作成し、指定した GCP 環境に `gce` イメージを直接アップロードします。

(BZ#2049492)

#### Image Builder オンプレミス CLI で、レジストリーへのコンテナイメージの直接プッシュがサポートされました

今回の機能強化により、Image Builder CLI を使用して、ビルド後に RHEL for Edge Container イメージをコンテナレジストリーに直接プッシュできます。Container Image をビルドするには、以下を実行します。

1. アップロードプロバイダーを設定し、必要に応じてクレデンシャルを追加します。
2. コンテナイメージをビルドし、コンテナレジストリーとリポジトリを `composer-cli` に引数として渡します。  
イメージの準備ができれば、設定したコンテナレジストリーで使用できます。

(JIRA:RHELPLAN-130376)

#### Image Builder オンプレミスのユーザーは、イメージ作成プロセス中にブループリントをカスタマイズできるようになりました

今回の更新では、**Edit Blueprint** ページが削除され、Image Builder サービスと **cockpit-composer** の Image Builder アプリのユーザーエクスペリエンスが統合されました。ユーザーは、ブループリントを作成し、イメージ作成プロセス中にパッケージの追加やユーザーの追加などのカスタマイズを行うことが可能になりました。ブループリントのバージョン管理も削除されたため、ブループリントのバージョンは1つだけ(現在のバージョン)になります。ユーザーは、作成済みのイメージから古いブループリントバージョンにアクセスできます。

(JIRA:RHELPLAN-122735)

## 4.2. シェルおよびコマンドラインツール

### Cronie に、選択した範囲内での時間のランダム化に対するサポートが追加されました

**Cronie** ユーティリティーは、cronjob の実行で ~ (範囲内でのランダム化) Operator をサポートするようになりました。その結果、選択した範囲内のランダムな時間に cronjob を開始できます。

(BZ#1832510)

### 新しいパッケージ: xmlstarlet

XMLStarlet は、XML ファイルの解析、変換、クエリー、検証、および編集を行うための一連のコマンドラインユーティリティーです。新しい **xmlstarlet** パッケージは、**grep**、**sed**、**awk**、**diff**、**patch**、**join** などのプレーンテキストファイルに対して UNIX コマンドを使用する場合と同様の方法で使用できる、シェルコマンドの単純なセットを提供します。

(BZ#1882020)

### ReaR は、リカバリー前後にコマンドを実行する新しい変数を追加します

この機能強化により、ReaR にはリカバリー前後に実行されるコマンドの自動化を容易にする2つの新しい変数が導入されます。

- **PRE\_RECOVERY\_COMMANDS** ではコマンド配列を使用できます。これらのコマンドは、リカバリー開始前に実行されます。
- **POST\_RECOVERY\_COMMANDS** ではコマンド配列を使用できます。これらのコマンドは、リカバリー完了後に実行されます。

これらの変数は、**PRE\_RECOVERY\_SCRIPT** および **POST\_RECOVERY\_SCRIPT** に代わるものですが、次の違いがあります。

- 以前の **PRE\_RECOVERY\_SCRIPT** および **POST\_RECOVERY\_SCRIPT** 変数では、単一のシェルコマンドを使用できます。これらの変数に複数のコマンドを渡すには、コマンドをセミコロンの区切る必要があります。
- 新しい **PRE\_RECOVERY\_COMMANDS** および **POST\_RECOVERY\_COMMANDS** 変数ではコマンド配列が使用でき、配列の各要素が個別のコマンドとして実行されます。

その結果、リカバリー前後にレスキューシステムで実行する複数のコマンドを提供することが容易になり、エラーが発生しにくくなりました。

詳細については、**default.conf** ファイルを参照してください。

(BZ#2035872)

### libva はバージョン 2.13.0 にリベースされました

ビデオアクセラレーション API の **libva** ライブラリーがバージョン 2.13.0 に更新されました。主な改善点と新機能は次のとおりです。

- 2つの新しい FourCC ビデオコーディング形式: **X2R10G10B10** および **X2B10G10R10** は、10ビット RGB 形式 (アルファを除く) でビデオをキャプチャ、処理、および表示します。
- **iris** および **crocus** DRI ドライバーの VA-API ドライバーマッピング。
- 出力バッファの同期のための **vaSyncBuffer** 関数。
- サーフェスとバッファをコピーするための **vaCopy** インターフェイス。
- デジタル著作権管理 (DRM) で保護されたビデオ用の LibVA Protected Content API。
- ビデオ処理の 3DLUT フィルター。入力カラーを新しい出力値にマップします。

([BZ#2099907](#))

### powerpc-utils がバージョン 1.3.10 にリベースされました

PowerPC プラットフォーム用のさまざまなユーティリティを提供する **powerpc-utils** パッケージがバージョン 1.3.10 に更新されました。以下は、主な改善点です。

- **ppc64\_cpu** ツールのエネルギーと周波数に関する Power Architecture Platform Reference (PAPR) 情報を解析する機能が追加されました。
- 最大設定システムで **lparstat -E** コマンドが失敗した場合に、拡張エラーメッセージを表示するように **lparstat** ユーティリティを改善しました。**lparstat** コマンドは、論理パーティション関連の情報を報告します。
- 報告された **lparstat** コマンドにおけるレガシー形式のオンラインメモリーが修正されました。
- NX GZIP アクセラレータのサービス品質クレジット (QoS) を動的に変更する **acc** コマンドのサポートが追加されました。
- **printf()** および **sprintf()** 呼び出しの書式指定子が改善されました。
- HMC ツールをハイブリッド仮想ネットワークに提供する **hcnmgr** ユーティリティには、以下の機能拡張が含まれます。
  - ハイブリッドネットワーク仮想化 **HNV FEATURE** リストに **wicked** 機能が追加されました。**hcnmgr** ユーティリティは、**wicked** ハイブリッドネットワーク仮想化 (HNV) をサポートし、ボンディングに **wicked** 関数を使用します。
  - **hcnmgr** は、後のクリーンアップのために **hcnid** 状態を維持します。
  - **hcnmgr** は、NetworkManager (NM) **nmcli** コードを除外します。
  - NM HNV **primary slave** 設定が修正されました。
  - **hcnmgr** は、仮想ネットワークインターフェイスコントローラー (vNIC) をバックアップデバイスとしてサポートします。
- **bootlist** の無効な 16 進数のシステムメッセージを修正しました。
- **-l** フラグが、**bootlist** コマンドの **-p** デリミタ値として **kpartx** ユーティリティに含まれました。

- IO スロットを一覧表示する際のメモリーリークを防ぐために、**sslot** ユーティリティーが修正されました。
- **lsslot** ユーティリティーに、最新の PCIe (Peripheral Component Interconnect Express) スロットタイプの DRC タイプ記述文字列が追加されました。
- **errinjct** ツールの RTAS への無効な設定アドレスが修正されました。
- **ofpathname** ユーティリティーで、non-volatile memory over fabrics (NVMf) デバイスのサポートが追加されました。このユーティリティーは、論理デバイス名からオープンファームウェアデバイスパス、およびその逆に変換するメカニズムを提供します。
- **ofpathname** ユーティリティーの非対称名前空間アクセス (ANA) モードでの不揮発性メモリー (NVMe) に対するサポートが修正されました。
- 設定ファイルとして **smt.state** ファイルをインストールしました。

(BZ#2051330)

### opencryptoki がバージョン 3.18.0 にリベースされました

Public-Key Cryptography Standard (PKCS) #11 の実装である **opencryptoki** パッケージがバージョン 3.18.0 に更新されました。以下は、主な改善点です。

- デフォルトは連邦情報処理標準 (FIPS) 準拠のトークンデータ形式 (tokversion = 3.12) です。
- グローバルポリシーでメカニズムとキーの使用を制限するためのサポートが追加されました。
- メカニズム使用状況の統計カウンターのサポートが追加されました。
- **ICA/EP11** トークンが **libica** ライブラリーバージョン 4 をサポートするようになりました。
- **p11sak** ツールを使用すると、公開鍵と秘密鍵に異なる属性を設定できます。
- **C\_GetMechanismList** は、EP11 トークンで **CKR\_BUFFER\_TOO\_SMALL** を返しません。

**openCryptoki** は、2 つの異なるトークンデータ形式をサポートしています。

- FIPS で承認されていないアルゴリズム (DES や SHA1 など) を使用する以前のデータ形式
- FIPS で承認されたアルゴリズムのみを使用する新しいデータ形式

FIPS プロバイダーは FIPS で承認されたアルゴリズムのみ使用を許可しているため、古いデータ形式は機能しなくなりました。



#### 重要

**openCryptoki** を RHEL 8 で機能させるには、システムで FIPS モードを有効にする前に、トークンを移行して新しいデータ形式を使用します。**openCryptoki 3.17** では以前のデータ形式がデフォルトのままであるため、これが重要です。システムが FIPS 対応に変更されると、以前びトークンデータ形式を使用する既存の **openCryptoki** インストールは機能しなくなります。

**openCryptoki** で提供される **pkcstok\_migrate** ユーティリティーを使用して、トークンを新しいデータ形式に移行できます。移行中は、**pkcstok\_migrate** は FIPS で承認されていないアルゴリズムを使用することに注意してください。したがって、システムで FIPS モードを有効にする前に、このツールを使用します。詳細は、[FIPS 準拠への移行 -pkcstok\\_migrate ユーティリティー](#) を参照してください。

(BZ#2043845)

Redfish モジュールが `redhat.rhel_mgmt` Ansible コレクションに含まれるようになりました。

`redhat.rhel_mgmt` Ansible コレクションには、次のモジュールが含まれるようになりました。

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

これによりユーザーは、Redfish モジュールを使用してサーバーのヘルスステータスの取得、ハードウェアとファームウェアのインベントリに関する情報の取得、電源管理、BIOS 設定の変更、帯域外 (OOB) コントローラーの設定、ハードウェア RAID の設定、ファームウェアの更新を実行し、自動化管理の恩恵を受けることができます。

(BZ#2112435)

`sysctl` が `systemd` ディレクトリーの順序と一致するようになりました

`sysctl` ユーティリティーの設定ディレクトリーの順序が、`systemd-sysctl` ディレクトリーの順序と同期されるようになりました。設定ディレクトリーは、実行時にカーネルパラメーターを調べて変更します。`/etc/sysctl.d` ディレクトリー内の設定ファイルは、`/run/sysctl.d` 内の設定ファイルよりも優先度が高く、`sysctl` と `systemd` の間でファイルの優先順位が乱れることはもうありません。

(BZ#2111915)

## 4.3. インフラストラクチャーサービス

`chrony` がバージョン 4.2 にリベースされました

`chrony` スイートがバージョン 4.2 に更新されました。バージョン 4.1 からの注目すべき機能強化は次のとおりです。

- サーバーインターリーブモードが改善され、信頼性が向上し、単一のアドレストランスレーター (Network Address Translation - NAT) の背後で複数のクライアントをサポートするようになりました。
- Network Time Protocol Version 4 (NTPv4) 拡張フィールドの実験的支持が追加され、時刻同期の安定性と推定誤差の精度が向上しました。`extfield F323` オプションを使用して、プロトコル NTPv4 の機能を拡張するこのフィールドを有効にできます。
- Precision Time Protocol (PTP) を介した NTP 転送の実験的支持が追加され、タイムスタンプが PTP パケットに制限されているネットワークインターフェイスカード (NIC) で完全なハードウェアタイムスタンプが有効になりました。`ptpport 319` ディレクティブを使用して、NTP over PTP を有効にできます。

(BZ#2062356)

`unbound` がバージョン 1.16.2 にリベースされました

`unbound` コンポーネントがバージョン 1.16.2 に更新されました。`unbound` は、検証、再帰、およびキャッシング DNS リゾルバーです。以下は、主な改善点です。

- **RFC 8976** をサポートする ZONEMD ゾーン検証により、受信者はデータのインテグリティと発信元の信頼性についてゾーンの内容を検証できるようになりました。

- **unbound** を使用すると、永続的な TCP 接続を設定できるようになりました。
- SVCB タイプおよび HTTPS タイプと、DNS **draft-ietf-dnsop-svcb-https** ドキュメントを介したサービスバインディングとパラメーター仕様に基づく処理が追加されました。
- **unbound** は、暗号化ポリシーからデフォルトの TLS 暗号を取得します。
- **RFC8375** に従い、専用ドメイン **home.arpa** を使用できます。このドメインは、住宅ホームネットワークでの非専用の使用に指定されています。
- **unbound** は、スタブゾーンまたはフォワードゾーンの **tcp-upstream** クエリーの選択的有効化をサポートするようになりました。
- **aggressive-nsec** オプションのデフォルトが **yes** になりました。
- **ratelimit** ロジックが更新されました。
- Unbound 応答ポリシーゾーン (RPZ) **nxdomain** 応答によってクエリーがブロックされた場合に、新しい **rpz-signal-nxdomain-ra** オプションを使用して **RA** フラグを解除できます。
- **RFC8914** に準拠した拡張 DNS エラー (EDE) の基本サポートにより、追加のエラー情報を利用できます。

([BZ#2027735](#))

## 4.4. セキュリティー

### NSS が 1023 ビット未満の RSA 鍵に対応しなくなる

Network Security Services (NSS) ライブラリーの更新により、すべての RSA 操作の最小鍵サイズが 128 から 1023 ビットに変更されます。つまり、NSS は以下の機能を実行しなくなります。

- RSA 鍵の生成は 1023 ビット未満です。
- 1023 ビット未満の RSA 鍵で RSA に署名するか、署名を検証します。
- 1023 ビットより短い RSA キーで値を暗号化または復号化します。

([BZ#2097837](#))

### SCAP セキュリティーガイドが 0.1.63 にリベースされました

SCAP セキュリティーガイド (SSG) パッケージは、アップストリームバージョン 0.1.63 にリベースされました。このバージョンは、さまざまな拡張機能とバグ修正を提供します。特に、次のようなものがあります。

- **sysctl**、**grub2**、**pam\_pwquality**、およびビルド時のカーネル設定の新しいコンプライアンスルールが追加されました。
- PAM スタックを強化するルールは、設定ツールとして **authselect** を使用するようになりました。注記: この変更により、PAM スタックが他の方法で編集された場合、PAM スタックを強化するルールは適用されません。

([BZ#2070564](#))

### CIS RHEL 8 ベンチマーク 2.0.0 に準拠した SSG CIS プロファイル

SCAP Security Guide (SSG) には、Center for Internet Security (CIS) プロファイルを CIS Red Hat

Enterprise Linux 8 Benchmark バージョン 2.0.0 に合わせる変更が含まれるようになりました。このバージョンのベンチマークでは、新しい要件が追加され、関連性がなくなった要件が削除され、いくつかの既存の要件が並べ替えられました。この更新は、関連するルールの参照とそれぞれのプロファイルの精度に影響を与えます。

(BZ#2058203)

**RHEL 8 STIG プロファイルが DISA STIG コンテンツにより適切に調整されるようになりました。**

**scap-security-guide** (SSG) パッケージで利用可能な DISA STIG for Red Hat Enterprise Linux 8 プロファイル (**xccdf\_org.ssgproject.content\_profile\_stig**) を使用して、アメリカ国防情報システム局 (DISA) による Security Technical Implementation Guides (STIG) に従ってシステムを評価できます。SSG のコンテンツを使用してシステムを修復できますが、DISA STIG 自動コンテンツを使用してシステムを評価する必要がある場合があります。今回の更新により、DISA STIG RHEL 8 プロファイルは、DISA のコンテンツとの整合性が向上しました。これにより、SSG 修復後に DISA コンテンツに対する調査結果が少なくなります。

以下のルールの評価は引き続き異なる点に注意してください。

- SV-230264r627750\_rule - CCE-80790-9 (**ensure\_gpgcheck\_globally\_activated**)
- SV-230349r833388\_rule - CCE-82266-8 (**configure\_bashrc\_exec\_tmux**)
- SV-230311r833305\_rule - CCE-82215-5 (**sysctl\_kernel\_core\_pattern**)
- SV-230546r833361\_rule - CCE-80953-3 (**sysctl\_kernel\_yama\_ptrace\_scope**)
- SV-230287r743951\_rule - CCE-82424-3 (**file\_permissions\_sshd\_private\_key**)
- SV-230364r627750\_rule - CCE-82472-2 (**accounts\_password\_set\_min\_life\_existing**)
- SV-230343r743981\_rule - CCE-86107-0 (**account\_passwords\_pam\_faillock\_audit**)

(BZ#1967947)

**/tmp および /var/tmp パーティションが存在しない場合に、マウントオプションの SSG ルールが誤って失敗することがなくなりました**

以前は、**/tmp** および **/var/tmp** パーティションのマウントオプションに関する SCAP セキュリティーガイド (SSG) ルールは、そのようなパーティションがシステムに存在しない場合、**fail** の結果を誤って報告していました。

この機能強化により、これらのルールが失敗するのではなく、適用されなくなります。現在、ルールは、パーティションが存在し、システムに正しいマウントオプションがない場合にのみ失敗します。

これらのマウントオプションが特定のポリシーに不可欠である場合、そのようなパーティションの存在を規定するルールがプロファイルに存在し、その1つのルールが失敗する必要があります。

(BZ#2032403)

**STIG セキュリティープロファイルがバージョン V1R7 に更新されました。**

SCAP セキュリティーガイドの **DISA STIG for Red Hat Enterprise Linux 8** プロファイルが更新され、最新バージョンの **V1R7** に合わせて更新されました。

このプロファイルはより安定し、DISA (Defense Information Systems Agency) が提供する RHEL 8 STIG (Security Technical Implementation Guide) のマニュアルベンチマークにより適切に調整されるようになりました。この反復により、**sysctl** コンテンツを新しい STIG に合わせて更新が行われます。

古バージョンが有効でなくなったため、このプロファイルの現行バージョンのみを使用する必要があります。



#### 警告

自動修正によりシステムが機能しなくなる場合があります。まずテスト環境で修復を実行してください。

(BZ#2112937)

#### clevis-luks-askpass がデフォルトで有効になりました

`/lib/systemd/system-preset/90-default.preset` ファイルには **enable clevis-luks-askpass.path** 設定オプションが含まれます。**clevis-systemd** サブパッケージをインストールすると、必ず **clevis-luks-askpass.path** ユニットファイルが有効になります。これにより、Clevis 暗号化クライアントは、起動プロセスの後半でマウントされる LUKS 暗号化ボリュームもロック解除できます。この更新の前に、管理者は **systemctl enable clevis-luks-askpass.path** コマンドを使用して、Clevis がそのようなボリュームのロックを解除できるようにする必要があります。

(BZ#2107081)

#### Rsyslog エラーファイルの最大サイズオプションを追加しました

新しい **action.errorfile.maxsize** オプションを使用すると、Rsyslog ログ処理システムのエラーファイルの最大バイト数を指定できます。エラーファイルが指定されたサイズに達すると、Rsyslog は追加のエラーやその他のデータを書き込むことができなくなります。これにより、エラーファイルが原因でファイルシステムがいっぱいになり、ホストが使用できなくなるのを防ぐことができます。

(BZ#1962318)

#### fapolicyd が 1.1.3 にリベースされました。

**fapolicyd** パッケージがバージョン 1.1.3 にアップグレードされました。主な改善点とバグ修正は次のとおりです。

- ルールに、サブジェクトの親 PID (プロセス ID) と一致する新しいサブジェクト PPID 属性を含めることができるようになりました。
- OpenSSL ライブラリーは、ハッシュ計算用の暗号化エンジンとして Libcrypt ライブラリーに取って代わりました。
- **fagenrules --load** コマンドが正しく機能するようになりました。

(BZ#2100087)

## 4.5. ネットワーキング

#### 大規模な iptables ルールセットの保存速度が改善されました

この機能強化により、**iptables-save** ユーティリティーが最適化され、大きなルールセットを保存する際のオーバーヘッドが削減されます。**/etc/protocols** ファイルからエントリーを読み取る際のユーティリティーが改善され、不要な場合に拡張共有オブジェクトファイルを検索しなくなりました。その結果、ストレージアクセスの遅延が大きい環境で大規模なルールセットを保存する場合の **iptables-save** の実行時間が大幅に改善されました。

(BZ#2058444)

## NetworkManager がバージョン 1.40 にリベース

**NetworkManager** パッケージがアップストリームバージョン 1.40 にアップグレードされ、以前のバージョンに対するバグ修正や機能強化が数多く追加されました。

- **/run/NetworkManager/devices/** ディレクトリーのデバイス状態ファイルに、現在のリースの DHCP オプションを含む新しいセクション **dhcp4** および **dhcp6** が追加されました。
- NetworkManager は、接続の **ipv6.mtu** プロパティーでの IPv6 最大転送単位 (MTU) の設定をサポートしています。
- NetworkManager は **nm.debug** カーネルコマンドラインオプションを使用して、デバッグログを有効にします。
- キャリア検出が改善されました。
- デバイスで MAC アドレスが変更された場合、NetworkManager は接続のために DHCP クライアントを再起動するようになりました。
- 特定のチャンネルを選択しない限り、Wifi ホットスポットは安定したランダムチャンネル番号を使用するようになりました。
- **wifi.key-mgmt** プロパティーを **wpa-psk** に設定し、ネットワークインターフェイスが Protected Management Frames (PMF) をサポートしていない場合、NetworkManager は Wi-Fi Protected Access 3 (WPA3) 移行モードを無効にするようになりました。移行モードは、このシナリオの特定のセットアップで問題を引き起こしました。WPA3 移行モードを明示的に有効にするには、**wifi.key-mgmt** を **sae** に設定します。
- NetworkManager は、DHCP サーバーから受信した非常に長いホスト名を最初のドットまたは 64 文字に短縮するようになりました。

主な変更の詳細は、[アップストリームのリリースノート](#) を参照してください。

(BZ#2063109)

## Microsoft Azure で毎起動時に cloud-init がネットワーク設定を更新するようになりました

仮想マシンがオフラインのときに管理者がネットワークインターフェイスの設定を更新しても、Microsoft Azure はインスタンス ID を変更しません。今回の機能強化により、仮想マシンの起動時に **cloud-init** サービスが常にネットワーク設定を更新し、Microsoft Azure 上の RHEL が最新のネットワーク設定を使用するようになります。

結果として、追加の検索ドメインなどのインターフェイスの設定を手動で設定すると、仮想マシンの再起動時に **cloud-init** がそれらをオーバーライドする可能性があります。詳細と回避策は、[cloud-init-22.1-5 が起動ごとにネットワーク設定を更新する](#) の解決策を参照してください。

(BZ#2144898)

NetworkManger は DHCP リース情報を **/run/NetworkManager/devices/** ディレクトリーに保存するようになりました

NetworkManager は、DHCP サーバーからのリース情報を `/run/NetworkManager/devices/` ディレクトリーに保存するようになりました。以前は、ファイルベースの API は利用できず、この情報は `nmcli -f all devices show DEVICE` コマンドの出力でのみ表示されました。この機能強化により、他のユーティリティーやスクリプトは、`nmcli` を呼び出さずに DHCP オプションにアクセスできます。

(BZ#1943153)

## 4.6. カーネル

### RHEL 8.7 のカーネルバージョン

Red Hat Enterprise Linux 8.7 には、カーネルバージョン 4.18.0-425 が同梱されています。

(BZ#2125545)

### SSBD と STIBP のデフォルトの軽減策が変更されました

`spec_store_bypass_disable` (SSBD) および `spectre_v2_user` (STIBP) ブートパラメーターのデフォルトの軽減策が、`seccomp` モードから `prctl` に変更されました。今回の更新により、`seccomp` の制御下にあるコンテナとアプリケーションのパフォーマンスが向上します。

(BZ#2101938)

### デバッグカーネルバリエーションで `vmcore` ダンプファイルが正しく生成される

今回の更新により、現在のカーネルがデバッグバリエーションである場合、`kdump` メカニズムはキャプチャカーネルと同じバージョンの非デバッグカーネルを使用するようになりました。非デバッグカーネルをキャプチャカーネルとして使用することで、`kdump` はデバッグバリエーションよりも少ないメモリーを消費します。その結果、`kdump` は `vmcore` ファイルを正しく生成し、クラッシュしたカーネルのメモリーの内容をキャプチャします。

(BZ#2006000)

### Intel E800 デバイスが iWARP および RoCE プロトコルをサポートするように

この機能拡張により、`enable_iwarp` および `enable_roce` devlink パラメーターを使用して、iWARP または RoCE プロトコルのサポートをオンまたはオフにできるようになりました。この必須機能を使用すると、いずれかのプロトコルを使用してデバイスを設定できます。Intel E800 デバイスは、同じポートで両方のプロトコルを同時にサポートしません。

特定の E800 デバイスの iWARP プロトコルを有効または無効にするには、まずカードの PCI の場所を確認します。

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

次に、プロトコルを有効または無効にします。devlink コマンドの引数として、カードの最初のポートに `pci/0000:44:00.0` を使用し、2 番目のポートに `pci/0000:44:00.1` を使用できます。

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

特定の E800 デバイスの RoCE プロトコルを有効または無効にするには、上記のようにカードの PCI の場所を確認します。次に、以下のコマンドの 1 つを使用します。

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

(BZ#2096127)

## 4.7. ブートローダー

### GRUB は新しい鍵で署名されています

セキュリティ上の理由から、GRUB は新しい鍵で署名されるようになりました。結果として、セキュアブート機能が有効になっている IBM POWER のリトルエンディアンバリエントで RHEL を使用している場合は、ブートできるようにファームウェアをバージョン FW1010.30 (またはそれ以降) または FW1020 に更新する必要があります。

(BZ#2074762)

### IBM POWER で仮想マシンを起動すると、設定可能なディスクアクセスが再試行する

IBM POWER アーキテクチャーで論理パーティション (**lpar**) 仮想マシン (VM) を起動すると、GRUB ブートローダーがリモートディスクへのアクセスを再試行する回数を設定できるようになりました。再試行回数を下げると、特定の状況で起動が遅くなる可能性があります。

以前は、起動時にディスクアクセスが失敗したときに GRUB がディスクへのアクセスを 20 回試行していました。これにより、低速なストレージエリアネットワーク (SAN) ディスクに接続されている **lpar** システムで、Live Partition Mobility (LPM) 移行を実行すると、問題が発生しました。その結果、20 回の再試行が完了するまで、起動に非常に長い時間がかかっていた可能性があります。

今回の更新で、**ofdisk\_retries** GRUB オプションを使用して、ディスクアクセスの再試行回数を減らすように設定できるようになりました。詳細は、[IBM POWER で仮想マシンを起動する際のディスクアクセスの再試行設定](#) を参照してください。

その結果、**lpar** ブートは LPM の後、POWER で遅くならず、障害が発生したディスクなしで **lpar** システムが起動できるようになりました。

(BZ#2070347)

## 4.8. ファイルシステムおよびストレージ

### nfsrahead が RHEL 8 に追加されました

**nfsrahead** ツールの導入により、それを使用して NFS マウントの **readahead** 値を変更できるため、NFS 読み取りパフォーマンスに影響を与えることができます。

(BZ#1946283)

### rpcctl コマンドが SunRPC 接続情報を表示するようになりました。

今回の更新で、**rpcctl** コマンドを使用して、システムの SunRPC オブジェクトに関する SunRPC **sysfs** ファイルで収集された情報を表示するようになりました。**sysfs** ファイルシステムを介して、SunRPC ネットワークレイヤーのオブジェクトを表示、削除、および設定できます。

(BZ#2087187)

**multipath.conf** は、DM Multipath でプロトコル固有の設定オーバーライドを含めることができるようになりました

さまざまなプロトコルを介してマルチパスデバイスのパスにアクセスできます。さまざまなプロトコルがさまざまな最適な設定を持つ可能性があるため、以前は、プロトコルごとのオプションを使用せずに Device Mapper Multipath 機能ですべてのプロトコルに最適な設定を設定することはできませんでした。この機能強化により、プロトコル固有の設定オーバーライドを **multipath.conf** ファイルに含めることができます。その結果、プロトコルごとにマルチパスデバイスパスを設定できるようになり、複数のプロトコルを介してアクセスできるマルチパスデバイスを正しく設定できるようになりました。

(BZ#2065477)

#### multipathd が、FPIN-Li イベントの検出をサポートするようになりました

**marginal\_pathgroups** 設定オプションに新しい値 **fpin** を追加すると、**multipathd** を有効化して Link Integrity Fabric Performance Impact Notification (PFIN-Li) イベントを監視し、リンク整合性の問題があるパスをマージナルパスグループに移動します。**fpin** 値を設定すると、**multipathd** は既存のマージナルパス検出方法をオーバーライドし、ファイバーチャネルファブリックに依存してリンクの整合性の問題を特定します。

この機能強化により、**multipathd** メソッドは、PFIN-Li イベントを発行できるファイバーチャネルファブリック上のマージナルパスの検出においてより堅牢になります。

(BZ#2083077)

## 4.9. 高可用性およびクラスター

**pcs** コマンドラインは、システムの再起動を必要としないマルチパス SCSI デバイスの更新をサポートします

**pcs stonith update-scsi-devices** コマンドを使用して、マルチパス SCSI デバイスを更新できるようになりました。このコマンドは、同じノードで実行されている他のクラスターリソースを再起動することなく、SCSI デバイスを更新します。

(BZ#2023845)

#### クラスター UUID をサポートします

クラスターのセットアップ中に、**pcs** コマンドがすべてのクラスターの UUID を生成するようになりました。クラスター名は一意的なクラスター識別子ではないため、複数のクラスターを管理する場合に、クラスター UUID を使用して同じ名前のクラスターを識別できます。

**pcs cluster config [show]** コマンドを使用して、現在のクラスター UUID を表示できます。**pcs cluster config uuid generate** コマンドを使用して、UUID を既存のクラスターに追加するか、すでに UUID が存在する場合は再生成できます。

(BZ#1950551)

#### multiple-active リソースパラメーターが stop\_unexpected の値を受け入れるようになりました

**multiple-active** リソースパラメーターは、リソースが複数のノードでアクティブであってはならないときにアクティブである場合の回復動作を決定します。デフォルトでは、リソースが本来あるべき場所で正常に実行されている場合でも、この状況ではリソースを完全に再起動する必要があります。今回の更新で、**multiple-active** リソースパラメーターは **stop\_unexpected** の値を受け入れます。これにより、multiple-active リソースの予期しないインスタンスのみを停止するように指定できます。ユーザーは、サービスとそのリソースエージェントが、完全に再起動しなくても追加のアクティブインスタンスで機能することを確認する必要があります。

(BZ#2036815)

## 新しい `allow-unhealthy-node` Pacemaker リソースメタ属性

Pacemaker は、`allow-unhealthy-node` リソースメタ属性をサポートするようになりました。このメタ属性が `true` に設定されている場合、ノードの正常性が低下したことでリソースがノードから強制的に切り離されることはありません。正常性リソースにこの属性が設定されている場合、クラスターはノードの正常性が回復したかどうかを自動的に検出し、リソースをノードに戻すことができます。

([BZ#2059638](#))

## Red Hat OpenStack Platform で高可用性がサポートされました

Red Hat OpenStack Platform で高可用性クラスターを設定できるようになりました。この機能をサポートするために、Red Hat は次の新しいクラスターエージェントを提供します。

- **fence\_openstack**: OpenStack 上の HA クラスターのフェンスエージェント
- **openstack-info**: OpenStack 上の HA クラスターに必要な **openstack-info** クローンリソースを設定するためのリソースエージェント
- **openstack-virtual-ip**: 仮想 IP アドレスリソースを設定するリソースエージェント
- **openstack-floating-ip**: フローティング IP アドレスリソースを設定するリソースエージェント
- **openstack-cinder-volume**: ブロックストレージリソースを設定するリソースエージェント

([BZ#1182956](#))

## Pacemaker は、システムグループのアクセス制御リスト (ACL) の指定をサポートするようになりました

これまで Pacemaker では個々のユーザーに ACL を指定できましたが、ローカルポリシーを使用してシステムグループに ACL を指定し、そのグループ内のすべてのユーザーに適用する方が簡単で適している場合があります。`pcs acl group` コマンドは以前のリリースに存在していましたが、効果がありませんでした。これで、ユーザーはこのコマンドを使用してシステムグループの ACL を指定できるようになりました。

([BZ#1724310](#))

## 設定済みフェンスデバイスを再作成する `pcs` コマンドを表示する新しい `pcs stonith config` コマンドオプションが追加されました

`pcs stonith config` コマンドで、`--output-format=cmd` オプションが使用可能になりました。このオプションを指定すると、別のシステムで設定済みのフェンスデバイスを再作成するために使用できる `pcs` コマンドが表示されます。

([BZ#1909904](#))

## 設定済みリソースを再作成する `pcs` コマンドを表示する新しい `pcs resource config` コマンドオプションが追加されました

`pcs resource config` コマンドで、`--output-format=cmd` オプションが使用可能になりました。このオプションを指定すると、別のシステムで設定済みリソースを再作成するために使用できる `pcs` コマンドが表示されます。

([BZ#1874624](#))

## 4.10. 動的プログラミング言語、WEB サーバー、およびデータベースサーバー

**nodejs:18** モジュールストリームに完全に対応しました。

以前はテクノロジープレビューとして利用できた **nodejs:18** モジュールストリームは、[RHSA-2022:8833](#) アドバイザリーのリリースで完全に対応しています。**nodejs:18** モジュールストリームでは、LTS (Long Term Support) バージョンの **Node.js 18.12** が提供されるようになりました。

RHEL 8.5 に含まれる **Node.js 18** は、RHEL 8.7 以降で利用可能な **Node.js 16** 上で、多くの新機能と、バグおよびセキュリティーの修正を提供します。

主な変更点は、以下のとおりです。

- **V8** エンジンがバージョン 10.2 にアップグレードされました。
- **npm** パッケージマネージャーがバージョン 8.18.0 にアップグレードされました。
- **Node.js** が実験的な新しい **fetch** API を提供するようになりました。
- **Node.js** は、新しい実験的な **node:test** モジュールを提供するようになりました。これにより、Test Anything Protocol (TAP) 形式で結果を報告するテストの作成が容易になります。
- **Node.js** は、IPv4 よりも IPv6 アドレスを優先するようになりました。

**nodejs:18** モジュールストリームをインストールするには、以下を使用します。

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

**nodejs:16** ストリームからアップグレードする場合は、[皇族のストリームへの切り替え](#) を参照してください。

(BZ#2083073)

**nodejs:18** がバージョン 18.14 にリベースされ、**npm** がバージョン 9 にリベースされました。

[RHSA-2023:1583](#) でリリースされた **Node.js 18.14** には、**npm** のバージョン 8 からバージョン 9 への SemVer メジャーアップグレードが含まれています。この更新はメンテナンス上の理由から必要であり、**npm** 設定の調整が必要になる場合があります。

特に、特定のレジストリーに範囲を限定しない認証関連の設定はサポートされなくなりました。この変更はセキュリティー上の理由から行われました。スコープ指定されていない認証設定を使用した場合、指定されたトークンは **.npmrc** ファイルにリストされているすべてのレジストリーに送信されます。

スコープなしの認証トークンを使用する場合は、レジストリースコープ付きトークンを生成して **.npmrc** ファイルに指定します。

**.npmrc** ファイル内に `//registry.npmjs.org/:_auth` など、`_auth` を使用する設定行がある場合は、それらを `//registry.npmjs.org/:_authToken=${NPM_TOKEN}` に置き換え、生成したスコープ付きトークンを指定します。

変更の完全なリストについては、[アップストリームの変更ログ](#) を参照してください。

(BZ#2178087)

新しいモジュールストリーム: **ruby:3.1**

RHEL 8.7 では、新しい **ruby:3.1.2** モジュールストリームに **Ruby 3.1** が導入されました。このバージョンでは、RHEL 8.5 で配布される **Ruby 3.0** に対するパフォーマンスの向上、バグおよびセキュリティ修正、および新機能が数多く追加されました。

主な機能拡張は、次のとおりです。

- **Interactive Ruby** (IRB) ユーティリティーは、オートコンプリート機能とドキュメントダイアログを提供するようになりました。
- **lib/debug.rb** を置き換える新しい **debug** gem は、パフォーマンスを改善し、リモートデバッグとマルチプロセス/マルチスレッドデバッグをサポートします。
- **error\_highlight** gem は、バックトレースでエラーの詳細な場所を提供するようになりました。
- ハッシュリテラルデータ型とキーワード引数の値を省略できるようになりました。
- ピン Operator (^) がパターンマッチングで式を受け入れるようになりました。
- 1行のパターンマッチングで括弧を省略できるようになりました。
- 新しい実験的なインプロセス Just-in-Time (JIT) コンパイラーである YJIT が、AMD および Intel 64 ビットアーキテクチャーで利用可能になりました。
- **TypeProf For IDE** ユーティリティーが導入されました。これは、IDE 内の **Ruby** コード用の実験的な静的型分析ツールです。

Method Based Just-in-Time Compiler (MJIT) では、以下の通りパフォーマンスが改善されました。

- **Rails** のようなワークロードでは、デフォルトの最大 JIT キャッシュ値が 100 から 10000 に増加しました。
- クラスイベントの **TracePoint** が有効になっている場合、JIT を使用してコンパイルされたコードがキャンセルされなくなりました。

その他の主な変更点は次の通りです。

- **tracer.rb** ファイルが削除されました。
- バージョン 4.0 以降、**Psych** YAML パーサーはデフォルトで **safe\_load** メソッドを使用しません。

**ruby:3.1** モジュールストリームをインストールするには、以下を使用します。

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

以前の **ruby** モジュールストリームからアップグレードするには、[Switching to a later stream](#) を参照してください。

(BZ#2063772)

### 新しいモジュールストリーム: **mercurial:6.2**

RHEL 8.7 は、新しいモジュールストリームとして **Mercurial 6.2** を追加します。このバージョンでは、RHEL 8.0 以降に利用可能な **Mercurial 4.8** よりも多くのバグ修正、機能強化、およびパフォーマンスの改善が提供されます。

主な変更点は、以下のとおりです。

- **Mercurial 6.2** は **Python 3.6** 以降をサポートします
- **Mercurial** は **Python 2** をサポートしなくなりました
- **hg purge** および **hg clean** コマンドで新しい **-i** オプションが提供されるようになりました。これにより、追跡されていないファイルの代わりに無視されたファイルを削除できます。
- **hg diff** および **hg extdiff** コマンドは、**--from <revision>** および **--to <revision>** 引数をサポートするようになりました
- 新しい内部マージユーティリティー **internal:mergediff** が利用可能になりました
- Zstandard (ZSTD) 圧縮は、利用可能な場合、新しいリポジトリにデフォルトで使用されるようになりました
- 必要な拡張機能が見つからない場合に **Mercurial** が起動しないようにする、必要な拡張機能を指定する新しい方法が利用可能になりました。

さらに、**hg** コマンドの C ラッパーを提供する新しい **mercurial-chg** ユーティリティーが利用可能になりました。**chg** コマンドを使用すると、**Mercurial** コマンドサーバーのバックグラウンドプロセスが作成され、C プログラムがそのバックグラウンドプロセスに接続して **Mercurial** コマンドを実行します。その結果、パフォーマンスが大幅に向上します。

**mercurial:6.2** モジュールストリームをインストールするには、次を使用します。

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

**mercurial:4.8** ストリームからアップグレードする場合は、[新しいストリームへの切り替え](#)を参照してください。

(BZ#2089849)

### **mariadb-java-client** がバージョン 2.7.1 にリベースされました

Java で開発されたアプリケーション用の **MariaDB** コネクターを提供する **mariadb-java-client** パッケージが、バージョン 2.7.1 に更新されました。

この更新では、サービスに次の変更が導入されています。

- クライアント認証プラグインがサービスとして定義されるようになりました。その結果、新しいクライアント認証プラグインを簡単に追加できます。ドライバーには、**MySQL** との互換性のために **caching\_sha2\_password** および **sha256\_password** プラグインが含まれています。
- 資格情報プラグインが資格情報を提供できるようになりました。ドライバーには、**AWS IAM**、**Environment**、および **Property** の 3 つのデフォルトプラグインが含まれています。
- SSL ファクトリーサービスにより、カスタム SSL 実装を使用できるようになりました。たとえば、新しい **HostnameVerifier** 実装を作成できます。

その他の主な変更点は次の通りです。

- **enabledSslProtocolSuites** オプションには、デフォルトで TLSv1.2 が含まれるようになりました。

(BZ#2043212)

### バージョン 6.2.7 にリベースされた **redis**

**redis:6** モジュールストリームを提供する高度なキー値ストアである **Redis 6** は、バージョン 6.2.7 に更新されました。この更新では、RHEL 8.4 以降に利用可能なバージョン 6.0 に対するバグ修正、セキュリティ修正、および改善が提供されます。

(BZ#1999873)

### httpd 設定の LimitRequestBody ディレクティブのデフォルトが新しくなりました

[CVE-2022-29404](#) を修正するために、Apache HTTP Server の **LimitRequestBody** ディレクティブのデフォルト値が **0** (無制限) から 1 GiB に変更されました。

**LimitRequestBody** の値が **httpd** 設定ファイルで明示的に指定されていないシステムでは、**httpd** パッケージを更新すると、**LimitRequestBody** がデフォルト値の 1 GiB に設定されます。その結果、HTTP 要求本文の合計サイズがデフォルトの制限である 1 GiB を超える場合、**httpd** は **413 Request Entity Too Large** エラーコードを返します。

HTTP 要求メッセージ本文の新しいデフォルトの許容サイズがユースケースに不十分な場合は、それぞれのコンテキスト (サーバー、ディレクトリーごと、ファイルごと、または場所ごと) 内で **httpd** 設定ファイルを更新し、適切な制限をバイト単位で設定します。たとえば、新しい制限を 2 GiB に設定するには、次を使用します。

```
LimitRequestBody 2147483648
```

すでに **LimitRequestBody** ディレクティブで明示的な値を使用するように設定されているシステムは、この変更の影響を受けません。

(BZ#2128016)

## 4.11. コンパイラーおよび開発ツール

### 新しい GCC Toolset 12

GCC Toolset 12 は最新バージョンの開発ツールを提供するコンパイラーツールセットです。このツールセットは、**AppStream** リポジトリにおいて、Software Collection の形式で、Application Streams として利用できます。

GCC コンパイラーがバージョン 12.1.1 に更新され、アップストリームの GCC で利用可能なバグ修正および機能拡張が数多く追加されました。

以下のツールおよびバージョンは、GCC Toolset 12 で利用できます。

ツール	バージョン
GCC	12.1.1
GDB	11.2
binutils	2.35
dwz	0.14
annobin	10.76

GCC Toolset 12 をインストールするには、root で以下のコマンドを実行します。

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

GCC Toolset 12 のツールを実行するには、以下のコマンドを実行します。

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

GCC Toolset バージョン 12 のツールバージョンが、このようなツールのシステムバージョンをオーバーライドするシェルセッションを実行するには、次のコマンドを実行します。

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

詳細は、[GCC ツールセット](#) を参照してください。

(BZ#2077276)

### GCC Toolset 12: Annobin がバージョン 10.76 にリベースされました。

GCC Toolset 12 では、Annobin パッケージがバージョン 10.76 に更新されました。

主なバグ修正と機能拡張は、以下のとおりです。

- annocheck の新しいコマンドラインオプションは、別の方法でデバッグ情報を見つけることができない場合、**debuginfod** サービスの使用を避けるように指示します。**debuginfod** を使用すると annocheck で多くの情報が提供されますが、**debuginfod** サーバーが使用できない場合、annocheck のパフォーマンスが大幅に低下する可能性があります。
- Annobin ソースは、必要に応じて設定および作成するのではなく、**meson** および **ninja** を使用してビルドできるようになりました。
- Annocheck は、Rust 1.18 コンパイラーによってビルドされたバイナリーをサポートするようになりました。

さらに、GCC Toolset 12 バージョンの Annobin で次の既知の問題が報告されています。

状況によっては、次のようなエラーメッセージでコンパイルが失敗する可能性があります。

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

この問題を回避するには、プラグインディレクトリーに **annobin.so** から **gcc-annobin.so** へのシンボリックリンクを作成します。

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

**architecture** は、使用しているアーキテクチャーに置き換えます。

- **aarch64**
- **i686**
- **ppc64le**

- **s390x**
- **x86\_64**

(BZ#2077447)

## GCC Toolset 12: binutils がバージョン 2.38 にリベースされました

GCC Toolset 12 では、**elfutils** パッケージがバージョン 2.38 に更新されました。

主なバグ修正と機能拡張は、以下のとおりです。

- **binutils** パッケージのすべてのツールで、マルチバイト文字の存在を表示または警告するオプションがサポートされるようになりました。
- **readelf** および **objdump** ツールは、デフォルトで **debuginfo** ファイルへのリンクを自動的にたどるようになりました。この動作は、**readelf** の **--debug-dump=no-follow-links** オプションまたは **objdump** の **--dwarf=no-follow-links** オプションを使用して無効にすることができます。

(BZ#2077448)

## GCC 12 以降は `_FORTIFY_SOURCE` レベル 3 をサポートします

この機能拡張により、ユーザーは、GCC バージョン 12 以降でビルドする場合に、コンパイラーコマンドラインで **-D\_FORTIFY\_SOURCE=3** を使用してアプリケーションをビルドできます。`_FORTIFY_SOURCE` レベル 3 では、ソースコード強化の範囲が改善されるため、コンパイラーコマンドラインで **-D\_FORTIFY\_SOURCE=3** を指定してビルドされたアプリケーションのセキュリティが向上します。これは、GCC バージョン 12 以降および Clang バージョン 9.0 以降でサポートされており、`__builtin_dynamic_object_size` ビルトインが含まれています。

(BZ#2033684)

## DNS スタブリゾルバーオプションが `no-aaaa` オプションをサポートするようになりました

今回の機能拡張により、**glibc** は `/etc/resolv.conf` の `no-aaaa` スタブリゾルバーオプションと `RES_OPTIONS` 環境変数を認識するようになりました。このオプションが有効な場合、AAAA クエリーはネットワーク経由で送信されません。システム管理者は、診断目的で AAAA DNS ルックアップを無効にすることができます。たとえば、IPv4 のみのネットワークでの余分なルックアップが DNS の問題に影響しないように除外できます。

(BZ#2096189)

## **glibc** で IBM Z シリーズ z16 のサポートを追加

**glibc** の **IBM z16** プラットフォームで **s390** 命令セットがサポートされるようになりました。**IBM z16** は、`HWCAP_S390_VXRS_PDE2` と `HWCAP_S390_NNPA` という 2 つの追加のハードウェア機能を提供します。その結果、アプリケーションはこれらの機能を使用して、最適化されたライブラリーと機能を提供できるようになりました。

(BZ#2077835)

## 新しい **make-latest** パッケージ

今回の機能拡張により、**make** ユーティリティーの最新バージョンを含む **make-latest** パッケージが導入されました。以前は、GCC Toolset を通じて最新の **make** バージョンを提供していました。これで、**make-latest** パッケージを個別にインストールし、`scl enable make43/bin/bash` で最新バージョンを実行できます (**make43** バージョンが最新の場合)。

(BZ#2083419)

## GCC Toolset 12: GDB がバージョン 11.2 にリベースされました。

GCC Toolset 12 では、GDB パッケージがバージョン 11.2 に更新されました。

主なバグ修正と機能拡張は、以下のとおりです。

- Aarch64 MTE の新しいサポート。 **memory-tag** プレフィックスが付いた新しいコマンドを参照してください。
- **-break-insert** および **-dprintf-insert** の **--qualified** オプション。このオプションは、全スコープで検索するのではなく、ユーザーのイベントの場所と完全に一致するものを探します。たとえば、**break --qualified foo** は、グローバルスコープで **foo** という名前のシンボルを探します。**--qualified** を指定しないと、GDB はすべてのスコープでその名前のシンボルを検索します。
- **--force-condition**: 現在無効になっていても、指定された条件が定義されます。
- **-break-condition --force**: MI コマンドと同様。
- **-file-list-exec-source-files** は、出力を制限するオプションの **REGEXP** を受け入れます。
- **.gdbinit** 検索パスには config ディレクトリーが含まれます。順序は次のとおりです。
  - a. **\$XDG\_CONFIG\_HOME/gdb/gdbinit**
  - b. **\$HOME/.config/gdb/gdbinit**
  - c. **\$HOME/.gdbinit**
- **~/.config/gdb/gdbinit** または **~/.gdbinit** のサポート。
- **-eix** および **-eix** 早期初期化ファイルオプション。

ターミナルユーザーインターフェイス (TUI):

- ターミナルユーザーインターフェイス (TUI) ウィンドウ内でのマウス操作がサポートされません。
- フォーカスされたウィンドウで機能しないキーの組み合わせが GDB に渡されるようになりました。

新しいコマンド:

- **show print memory-tag-violations**
- **set print memory-tag-violations**
- **memory-tag show-logical-tag**
- **memory-tag with-logical-tag**
- **memory-tag show-allocation-tag**
- **memory-tag check**
- **show startup-quietly** および **set startup-quietly**: GDB スクリプトで **-q** または **-quiet** を指定する方法。早期初期化ファイルでのみ有効です。

- **show print type hex** および **set print type hex**: ストラクチャーメンバーのサイズまたはオフセットを 10 進法ではなく 16 進法で出力するように GDB に指示します。
- **show python ignore-environment** および **set python ignore-environment**: 有効にすると、GDB の Python インタープリターは、Python 実行可能ファイルに **-E** を渡すのと同じように、Python 環境変数を無視します。早期初期化ファイルでのみ有効です。
- **show python dont-write-bytecode** および **set python dont-write-bytecode: off** の場合、これらのコマンドは、Python 実行可能ファイルに **-B** を渡すのと同様に、GDB の Python インタープリターがインポートされたモジュールのバイトコードコンパイル済みオブジェクトを書き込むことを抑制します。早期初期化ファイルでのみ有効です。

変更したコマンド:

- **break LOCATION if CONDITION: CONDITION** が無効な場合、GDB はブレークポイントの設定を拒否します。 **-force-condition** オプションはこれをオーバーライドします。
- **CONDITION -force N COND**: 前のコマンドと同じ。
- **Inferior ID**: ID が省略されている場合、このコマンドは現在の下位関連情報を出力します。それ以外の場合は、変更されません。
- **ptype[/FLAGS] TYPE | EXPRESSION**: **/x** フラグを使用して、struct メンバーのサイズとオフセットを出力するときに 16 進法での表記を使用します。 **/d** フラグを使用して同じことを行いますが、10 進法を使用します。
- **info sources**: 出力が再構築されました。

Python API:

- 下位オブジェクトには、読み取り専用の **connection\_num** 属性が含まれています。
- 新しい **gdb.Frame.level()** メソッド。
- 新しい **gdb.PendingFrame.level()** メソッド。
- **gdb.Stop** の代わりに **gdb.BreakpointEvent** が出力されます。

(BZ#2077492)

### libpfm が AMD Zen 2 および Zen 3 プロセッサをサポートするようになりました

この機能強化により、ユーザーは **libpfm** を使用して AMD Zen 2 および Zen 3 パフォーマンス監視ハードウェアにアクセスできるようになりました。

(BZ#2067218)

### papi が AMD Zen 2 および Zen 3 プロセッサをサポートするようになりました

この機能強化により、ユーザーは **papi** を使用して AMD Zen 2 および Zen 3 パフォーマンス監視ハードウェアにアクセスできるようになりました。

(BZ#2071558)

### ARM プロセッサのハードウェア識別の改善

この機能強化により、**papi\_avail** ユーティリティは、さまざまな ARM ベンダーのベンダー文字列とコード情報を正しく報告するようになりました。このユーティリティを使用する

と、`PAPI_get_hardware_info()` 関数を使用して、**aarch64** アーキテクチャーで、**ARM** 以外のメーカーが製造したプロセッサを識別できます。その結果、開発者は必要なアーキテクチャーに合わせてコードを調整できます。

(BZ#2037427)

### 富士通 A64FX イベントマッピングの更新

**PAPI** ライブラリーは、Fujitsu A64FX プロセッサ用に更新されました。ユーザーは、プログラムのパフォーマンスを分析するために使用できる `papi_avail` の出力で追加のプリセットを使用できるようになりました。

これらには、**IDL** イベントプリセットが含まれます。

#### **PAPI\_BRU\_IDL**

ブランチユニットのアイドル

#### **PAPI\_FXU\_IDL**

整数ユニットのアイドル

#### **PAPI\_FPU\_IDL**

浮動小数点ユニットのアイドル

#### **PAPI\_LSU\_IDL**

ロードストアユニットアイドル

(BZ#2037417)

### **dyninst** パッケージ化がバージョン 12.1 にリベースされました

**dyninst** パッケージがバージョン 12.1 にリベースされました。主なバグ修正と機能拡張は、以下のとおりです。

- **glibc-2.35** の複数名前空間に対する初期サポート
- DWARF 並列解析の並列処理性を修正
- **CUDA** および **CDNA2** GPU バイナリーのサポートを改善
- IBM POWER Systems (リトルエンディアン) レジスターアクセスのサポートを改善
- PIE バイナリーのサポートを改善
- キャッチブロックの解析を修正
- 64 ビット Arm (**aarch64**) 浮動小数点レジスターへのアクセスを修正

(BZ#2057676)

### **systemtap** パッケージがバージョン 4.7 にリベース

**systemtap** パッケージがバージョン 4.7 にリベースされました。主なバグ修正と機能拡張は、以下のとおりです。

- xMOK キーを使用してモジュールに手動で署名するための新しい `--sign-module` オプション。SecureBoot で使用します。
- アノテーション付きソースコードのシステム全体のプロファイルを作成するための新しい `stap-profile-annotate` ツール。

- 関数のエントリーとリターンをプローブするための新しい一般的な Python タップセット。
- ユーザー空間にある可能性がある文字列のカーネル空間プローブの `$ foo $` の拡張処理。
- 非キャプチャグループの正規表現言語の拡張。
- 最近追加されたいくつかのカーネルシステムコールのタップセットサポートの追加。

(BZ#2057565)

## Rust Toolset がバージョン 1.62.1 にリベースされました

Rust Toolset が、バージョン 1.62.1 に更新されました。主な変更点は、以下のとおりです。

- 分割代入では、代入の左側にある既存の変数にパターンを代入できます。たとえば、タプル代入は変数にスワップできます: `(a, b) = (b, a);`
- `core::arch::asm` マクロを使用して、インラインアセンブリーが 64 ビット x86 および 64 ビット ARM でサポートされるようになりました。詳細については、[リファレンスのインラインアセンブリー](https://doc.rust-lang.org/reference/inline-assembly.html) の章 `/usr/share/doc/rust/html/reference/inline-assembly.html` (オンライン: <https://doc.rust-lang.org/reference/inline-assembly.html>) を参照してください。
- 列挙は、明示的にアノテーションが付けられた `#[default]` バリエーションを使用して `Default` トレイトを派生できるようになりました。
- `Mutex`、`CondVar`、および `RwLock` は、`pthread` ではなくカスタム `futex` ベースの実装を使用するようになり、Rust 言語保証によって新たな最適化が可能になりました。
- Rust は、新しく安定化された `Termination` トレイトを実装するユーザー定義型を含む、`main` からのカスタム終了コードをサポートするようになりました。
- Cargo は、依存関係機能のより詳細な制御をサポートしています。`dep:` プレフィックスは、それを機能として公開することなく、オプションの依存関係を参照できます。また、`?` は、`package-name?/feature-name` のように、その依存関係が他の場所でも有効になっている場合にのみ依存関係の機能を有効にします。
- Cargo には、依存関係を `Cargo.toml` に追加するための新しい `cargo add` サブコマンドがあります。
- 詳細については、一連のアップストリームリリース発表を参照してください。
  - [Announcing Rust 1.59.0](#)
  - [Announcing Rust 1.60.0](#)
  - [Announcing Rust 1.61.0](#)
  - [Announcing Rust 1.62.0](#)
  - [Announcing Rust 1.62.1](#)

(BZ#2075344)

## LLVM Toolset がバージョン 14.0.6 にリベース

LLVM Toolset はバージョン 14.0.6 にリベースされました。主な変更点は、以下のとおりです。

- 64 ビット x86 では、`AVX512-FP16` 命令のサポートが追加されました。

- Armv9-A、Armv9.1-A、および Armv9.2-A アーキテクチャーのサポートが追加されました。
- PowerPC では、IBM double-double 形式を表す `__ibm128` 型が追加されました。これは `__attribute__((mode(IF)))` としても使用できます。

clang の変更:

- **C++2b** の `if constexpr` が実装されました。
- 64 ビット x86 では、**AVX512-FP16** 命令のサポートが追加されました。
- 実験段階の OpenCL C 3.0 および OpenCL 2021 の **C++** サポートが完了しました。
- **-E -P** プリプロセッサの出力は、必ず空白行を省略します。これは、GCC の動作に一致します。以前は、最大 8 つの連続する空白行が出力に表示される可能性がありました。
- C89 だけでなく、**C99** 以降の標準で **-Wdeclaration-after-statement** をサポートします。これは、GCC の動作に一致します。注目すべきユースケースは、宣言とコードの混在を禁止するスタイルガイドのサポートですが、さらに新しい C 標準に移行したいと考えています。

詳細については、[LLVM Toolset](#) および [Clang](#) アップストリームのリリースノートを参照してください。

(BZ#2061042)

## Go Toolset がバージョン 1.18.2 にリベース

Go Toolset はバージョン 1.18.2 にリベースされました。

主な変更点は、以下のとおりです。

- 以前のバージョンの Go との下位互換性を維持しながら generics を導入。
- 新しいファジングライブラリー。
- 新しい `debug/buildinfo` および `net/netip` パッケージ。
- `go get` ツールがパッケージをビルドまたはインストールしなくなりました。現在は、`go.mod` の依存関係のみを処理します。
- メインモジュールの `go.mod` ファイルで `go 1.17` 以降が指定されている場合、追加の引数なしで `go mod download` コマンドを使用すると、メインモジュールの `go.mod` ファイルで明示的に必要なモジュールのソースコードのみがダウンロードされます。推移的な依存関係のソースコードもダウンロードするには、`go mod download all` コマンドを使用します。
- `go mod vendor` サブコマンドが、出力ディレクトリーを設定する `-o` オプションをサポートするようになりました。
- `go mod tidy` コマンドが、インポートされた各パッケージを提供しているビルドリスト内のモジュールが 1 つだけであることを確認するためにソースコードが必要なモジュールの追加のチェックサムを `go.sum` ファイルに保持するようになりました。この変更は、メインモジュールの `go.mod` ファイルの Go バージョンを条件に含まれません。

(BZ#2075162)

## LLVM ゴールドプラグインが IBM Z アーキテクチャーで利用可能になりました

この機能拡張により、ユーザーは IBM Z (**s390x**) アーキテクチャーで **clang** および **ld.bfd** を使用して LTO ビルドを作成できます。**s390x** アーキテクチャーは、**ld.bfd** および LTO とのリンクをサポートするようになりました。

(BZ#2088315)

### 新しいモジュールストリーム: **maven:3.8**。

RHEL 8.7 では、新しいモジュールストリームとして **Maven 3.8** が導入されています。

**maven:3.8** モジュールストリームをインストールするには、次を使用します。

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

**maven:3.6** ストリームからアップグレードする場合は、[後続のストリームへの切り替え](#) を参照してください。

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

### **.NET** バージョン 7.0 が利用可能

Red Hat Enterprise Linux 8.7 には **.NET** バージョン 7.0 が同梱されています。以下は、主な改善点です。

- IBM Power のサポート (**ppc64le**)

詳細は、[.NET 7.0 RPM パッケージリリースノート](#) および [.NET 7.0 コンテナリリースノート](#) を参照してください。

(BZ#2112096)

## 4.12. IDENTITY MANAGEMENT

### SSSD が SID リクエストのメモリーキャッシングをサポートするようになりました

この機能拡張により、SSSD は SID 要求のメモリーキャッシングをサポートするようになりました。これには、SID による GID および UID ルックアップと、その逆が含まれます。メモリーキャッシングにより、たとえば、Samba サーバーとの間で大量のファイルをコピーする場合などに、パフォーマンスが向上します。

(JIRA:RHELPLAN-123369)

### IdM は、Windows Server 2022 での AD Trust の設定をサポートするようになりました

この機能拡張により、Identity Management (IdM) ドメインと、Windows Server 2022 を実行するドメインコントローラーを使用する Active Directory フォレストとの間でフォレスト間の信頼を確立できます。

([BZ#2122716](#))

### IdM は、ユーザーパスワードの有効期限が切れた後に許可される LDAP バインド数の制限をサポートするようになりました

今回の機能拡張により、Identity Management (IdM) ユーザーのパスワードの有効期限が切れたときに許可される LDAP バインドの数を設定できます。

-1

IdM は、ユーザーがパスワードをリセットする必要がある前に、ユーザーに無制限の LDAP バインドを許可します。これはデフォルト値で、以前の動作と一致します。

0

この値は、パスワードの有効期限が切れると、すべての LDAP バインドを無効にします。適用された場合、ユーザーは自分のパスワードをすぐにリセットする必要があります。

#### 1-MAXINT

有効期限が切れると、入力された値どおりの数のバインドが許可されます。

この値は、グローバルパスワードポリシーとグループポリシーで設定できます。

数はサーバーごとに保存されることに注意してください。

ユーザーが自分のパスワードをリセットするには、現在の期限切れパスワードにバインドする必要があります。ユーザーが有効期限後に許可されるバインドをすべて使い果たした場合、管理者がパスワードをリセットする必要があります。

(BZ#782917)

**IdM は、名前検索中に、指定された名前が信頼された AD ドメインのユーザーまたはグループであるかどうかを示すようになりました**

今回の更新により、新しい `getorigbyusername()` および `getorigbygroupname()` 呼び出しが、SID ベースのルックアップ用のユーティリティライブラリーである `libsss_nss_idmap` に追加されました。この追加により、Identity Management (IdM) が Active Directory (AD) ドメインと信頼関係にある場合に、ユーザーとグループのルックアップがより堅牢になります。ユーザーまたはグループのルックアップを実行するときに、IdM は、指定された名前が信頼できるドメインのユーザーまたはグループに属しているかどうかを表示できるようになりました。

(BZ#2062379)

**新しい `ipasmartcard_server` および `ipasmartcard_client` ロールが追加されました**

今回の更新により、`ansible-freeipa` パッケージは、スマートカード認証用に Identity Management (IdM) サーバーとクライアントを設定するための Ansible ロールを提供します。`ipasmartcard_server` および `ipasmartcard_client` ロールは、統合を自動化および簡素化するために `ipa-advise` スクリプトを置き換えます。他の `ansible-freeipa` ロールと同じインベントリと命名スキームが使用されます。

(BZ#2076554)

**samba がバージョン 4.16.1 にリベースされました。**

`samba` パッケージがアップストリームバージョン 4.16.1 にアップグレードされ、以前のバージョンに対するバグ修正や機能強化が追加されました。

- デフォルトでは、`smbd` プロセスは新しい `samba-dcerpcd` プロセスをオンデマンドで自動的に開始し、分散コンピューティング環境/リモートプロシージャコール (DCERPC) を提供します。Samba 4.16 以降では、DCERPC を使用するために必ず `samba-dcerpcd` が必要であることに注意してください。`/etc/samba/smb.conf` ファイルの `[global]` セクションで `rpc start on demand helpers` 設定を無効にする場合、スタンドアロンモードで `samba-dcerpcd` を実行する `systemd` サービスユニットを作成する必要があります。
- Cluster Trivial Database (CTDB) の `recovery master` のロール名が、`leader` に変更されました。その結果、次のとおり `ctdb` サブコマンドの名前が変更されました。
  - `recmaster` から `leader` に変更
  - `setrecmasterrole` から `setleaderrole` に変更

- CTDB **recovery lock** 設定の名前が **cluster lock** に変更されました。
- CTDB は、リーダーブロードキャストと関連するタイムアウトを使用して、選択が必要かどうかを判断するようになりました。

Samba 4.11 以降はサーバーメッセージブロックバージョン 1 (SMB1) プロトコルが非推奨となり、今後のリリースで削除されることに注意してください。

Samba を起動する前にデータベースファイルがバックアップされます。**smbd**、**nmbd**、または **winbind** サービスが起動すると、Samba が **tdb** データベースファイルを自動的に更新します。Red Hat は、**tdb** データベースファイルのダウングレードをサポートしていないことに留意してください。

Samba を更新したら、**testparm** ユーティリティーを使用して **/etc/samba/smb.conf** ファイルを確認します。

更新する前に、[upstream release notes](#) で重要な変更の詳細を確認してください。

([BZ#2077468](#))

### SSSD が Windows Server 2022 との直接統合をサポートするようになりました

この機能拡張により、SSSD を使用して、Windows Server 2022 を実行するドメインコントローラーを使用する Active Directory フォレストと RHEL システムを直接統合できます。

([BZ#2070793](#))

### Directory Server が Auto Membership プラグインタスクのキャンセルに対応

以前は、ディレクトリーサーバーの設定が複雑な場合 (大規模なグループ、複雑なルール、他のプラグインとの対話など)、Auto Membership プラグインタスクが原因でサーバーの CPU 使用率が高くなる可能性があります。この機能強化により、Auto Membership プラグインタスクをキャンセルできます。その結果、パフォーマンスの問題は発生しなくなりました。

([BZ#2052528](#))

### Directory Server は、Idapdelete 使用時に再帰的な削除操作をサポートするようになりました

この機能拡張により、Directory Server は **Tree Delete Control 1.2.840.113556.1.4.805** OpenLDAP コントロールをサポートするようになりました。その結果、**Idapdelete** ユーティリティーを使用して、親エントリーのサブエントリーを再帰的に削除できます。

([BZ#2057063](#))

### Directory Server インストール時における基本的な複製オプションの設定に対応

この機能強化により、インスタンスのインストール時に **.inf** ファイルを使用して、認証情報や変更履歴のトリミングなどの基本的なレプリケーションオプションを設定することができるようになりました。

([BZ#2057066](#))

### Directory Server でレプリケーション changelog のトリミングがデフォルトで有効化

以前は、Directory Server はデフォルトでレプリケーション **changelog** ファイルを自動的にトリミングするように設定されていませんでした。その結果、**changelog** ファイルが非常に大きくなる可能性があります。今回の更新により、Directory Server はデフォルトで7日より古い変更ログエントリーをトリミングするように設定され、**changelog** ファイルが過度に大きくなるのを防ぎます。

([BZ#2062679](#))

## pki パッケージの名前を idm-pki に変更

以下の **pki** パッケージの名前が **idm-pki** に変更され、IDM パッケージと Red Hat Certificate System パッケージを区別しやすくなりました。

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

**pki-core** は変更されません (これには **pki-core-debuginfo** と **pki-core-debugsource** も含まれます)。

([BZ#2139821](#))

## 4.13. グラフィックインフラストラクチャー

### Vulkan パッケージは 64 ビット IBM POWER で利用可能です

Vulkan 3D グラフィックス API のサポートを提供するパッケージが、リトルエンディアン 64 ビット IBM POWER アーキテクチャー (**ppc64le**) で利用できるようになりました。

- **vulkan-headers**
- **vulkan-loader**
- **vulkan-loader-devel**
- **vulkan-tools**

これらのパッケージを使用すると、Vulkan レンダリングエンジンを使用するソフトウェアを実行できます。

以前は、これらのパッケージは AMD64 および Intel 64 アーキテクチャーでのみ利用可能でした。

([BZ#2012639](#))

### 新しい AMD GPU のサポート

今回のリリースで、AMD Ryzen 6000 Series CPU の複数の AMD Radeon RX 6000 Series GPU と統合グラフィックスのサポートが追加されました。

以下の AMD Radeon RX 6000 Series GPU モデルがサポートされるようになりました。

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

AMD Ryzen 6000 シリーズには、以下の CPU モデルで確認された統合 GPU が含まれています。

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

(JIRA:RHELPLAN-135602)

#### **force\_probe オプションは第 12 世代 Intel Core GPU で不要に**

このリリースより前は、`i915.alpha_support=1` または `i915.force_probe=*` カーネルオプションを設定して、以前は Alder Lake-S および Alder Lake-P として知られていた第 12 世代 Intel Core GPU のサポートを有効にする必要がありました。

このリリースでは、オプションを設定する必要がなくなり、これらの GPU の完全なサポートがデフォルトで有効になっています。

(JIRA:RHELPLAN-136150)

## **4.14. WEB コンソール**

**RHEL Web コンソールは、OS のダウンロードの仮想マシンワークフローのオプションとして RHEL を備えています。**

この機能拡張により、RHEL Web コンソールは、デフォルトの OS のダウンロードワークフローを使用した RHEL 仮想マシン (VM) のインストールをサポートするようになりました。その結果、Web コンソール内で直接 RHEL OS を仮想マシンとしてダウンロードおよびインストールできます。

(JIRA:RHELPLAN-121982)

**カーネルパッチを個別にインストールするための RHEL Web コンソールの新しいボタン**

今回の更新により、RHEL Web コンソールに **kpatch 更新のインストール** ボタンが表示されます。これを使用して、他の更新のインストールやシステムの再起動なしに、カーネルのパッチのみをインストールできます。

(JIRA:RHELPLAN-121981)

### 診断レポートページに新機能が追加されました

更新された Web コンソールの診断レポート (**sos** レポート) ページでは、次のことができるようになりました。

- レポートにラベルを付ける
- レポートをパスワードで暗号化する
- レポート内の個人データを非表示にする

さらに、以前に生成されたレポートの一覧を表示し、それらをダウンロードまたは削除できます。

(JIRA:RHELPLAN-121983)

### Web コンソール UI からの暗号化ポリシーのセットアップ

今回の更新により、さまざまな暗号化ポリシーレベルを RHEL Web コンソールのユーザーインターフェイス (UI) から直接変更できるようになりました。UI の **概要** ページの **設定** フィールドから、暗号化ポリシーの設定オプションにアクセスできます。

設定を変更できるようにするには、管理者アクセスがアクティブになっている必要があることに注意してください。

(JIRA:RHELPLAN-121980)

### Web コンソールの更新進行状況ページで、自動再起動オプションがサポートされるようになりました

更新の進行状況ページに、**完了後に再起動する** スイッチが追加されました。これにより、更新のインストール後にシステムが自動的に再起動します。

(BZ#2056786)

## 4.15. RED HAT ENTERPRISE LINUX システムロール

### ha\_cluster RHEL システムロールは、SBD フェンシングと Corosync 設定をサポートするようになりました

**ha\_cluster** システムロールは、次の機能をサポートするようになりました。

#### SBD フェンシング

フェンシングは、HA クラスタ設定の重要な部分です。SBD は、フェンシングが必要な場合にノードが確実に自己終了する手段を提供します。SBD フェンシングは、従来のフェンシングメカニズムが不可能な環境で特に役立ちます。**ha\_cluster** システムロールを使用して SBD フェンシングを設定できるようになりました。

#### Corosync 設定

**ha\_cluster** システムロールは、トランスポート、圧縮、暗号化、リンク、トーテム、クォーラムなどの Corosync 設定をサポートするようになりました。これらの設定は、デフォルト設定が適切でない場合に、クラスタ設定をお客様のニーズと環境に一致させるために必要です。

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

ユーザーは、**network RHEL** システムロールを使用して、**IPoIB** 機能で接続を作成できます。

**network RHEL** システムロールの **infiniband** 接続タイプは、Internet Protocol over Infiniband (IPoIB) 機能をサポートするようになりました。この機能を有効にするには、**infiniband** の **p\_key** オプションに値を定義します。**p\_key** を指定する場合は、**network\_connections** 変数の **interface\_name** オプションを未設定にする必要があることに注意してください。**network RHEL** システムロールの以前の実装では、**infiniband** 接続タイプの **p\_key** 値と **interface\_name** オプションが適切に検証されませんでした。したがって、IPoIB 機能はこれまで機能しませんでした。詳細については、`/usr/share/doc/rhel-system-roles/network/` ディレクトリーにある README ファイルを参照してください。

([BZ#2086869](#))

**network RHEL** システムロールは、ルーティングルールのネットワーク設定を設定するようになりました

以前は、パケットの宛先アドレスフィールドに基づいてパケットをルーティングできましたが、ソースルーティングおよびその他のポリシールーティングルールを定義できませんでした。今回の機能拡張により、**network RHEL** システムロールはルーティングルールをサポートし、ユーザーがパケット転送またはルート選択を制御できるようになります。

([BZ#1996731](#))

**Networking** システムロールは、管理対象設定ファイルで **Ansible managed** コメントを使用します

**initscripts** プロバイダーを使用する場合、**Networking** システムロールにより、コメント付きの **ifcfg** ファイルが `/etc/sysconfig/network-scripts` ディレクトリーに生成されるようになりました。**Networking** ロールは、Ansible 標準の **ansible\_managed** 変数を使用して **Ansible managed** コメントを挿入します。コメントは、**ifcfg** ファイルが Ansible によって管理されていることを宣言し、**Networking** ロールによってファイルが上書きされるため、**ifcfg** ファイルを直接編集してはならないことを示しています。プロバイダーが **initscripts** の場合、**Ansible managed** コメントが追加されます。**nm** (NetworkManager) プロバイダーで **Networking** ロールを使用する場合、**ifcfg** ファイルは **Networking** ロールではなく **NetworkManager** によって管理されます。

([BZ#2065670](#))

新しい **previous:replaced** 設定により、**firewall** システムロールがファイアウォール設定をデフォルトにリセットできるようになります

各マシンで既存のファイアウォール設定が異なる一連のマシンを管理するシステム管理者は、**firewall** ロールの **previous: replaced** 設定を使用して、すべてのマシンで同じファイアウォール設定を設定できるようになりました。**previous: replaced** 設定では、既存のすべてのファイアウォール設定を消去し、それらを一貫した設定に置き換えることができます。

([BZ#2043009](#))

強化された **Microsoft SQL Server RHEL** システムロール

次の新しい変数が、**microsoft.sql.server** RHEL システムロールで使用できるようになりました。

- 高可用性クラスターの設定を制御する **mssql\_ha\_** 接頭辞を持つ変数。
- 管理対象ノードで **mssql\_tls\_cert** および **mssql\_tls\_private\_key** の値を検索するための **mssql\_tls\_remote\_src** 変数。デフォルトの **false** 設定のままにすると、ロールは制御ノードでこれらのファイルを検索します。

- ファイアウォールポートを自動的に管理するための `mssql_manage_firewall` 変数。この変数が `false` に設定されている場合は、ファイアウォールポートを手動で有効にする必要がありません。
- `mssql_pre_input_sql_file` 変数と `mssql_post_input_sql_file` 変数を使用して、SQL スクリプトをロールの実行前または実行後に実行するかどうかを制御します。これらの新しい変数は、SQL スクリプトの実行時間に影響を与えることができなかった以前の `mssql_input_sql_file` 変数に取って代わります。

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

**logging RHEL システムロールは、ファイル入力の `startmsg.regex` および `endmsg.regex` オプションをサポートします。**

この機能拡張により、正規表現を使用して、ファイルからのログメッセージをフィルタリングできるようになりました。オプション `startmsg_regex` と `endmsg_regex` がファイルの入力に含まれるようになりました。`startmsg_regex` はメッセージの開始部分に一致する正規表現を表し、`endmsg_regex` はメッセージの最後の部分に一致する正規表現を表します。その結果、日時、優先度、重大度などのプロパティに基づいてメッセージをフィルタリングできるようになりました。

([BZ#2112143](#))

**シンプロビジョニングされたボリュームが、storage RHEL システムロールでサポートされません。**

**storage RHEL システムロールは、シンプロビジョニングされた LVM 論理ボリュームを作成し、管理できるようになりました。**シンプロビジョニングされた LV は書き込み時に割り当てられます。これにより、後で必要に応じてシンプロビジョニングされた LV に提供される物理ストレージとしてボリュームを作成する際の柔軟性が向上します。LVM シンプロビジョニングでは、シン LV とそのスナップショットに共通するデータブロックが共有されるため、より効率的なスナップショットを作成できます。

([BZ#2066876](#))

**logging RHEL システムロールは、`template`、`severity`、および `facility` オプションをサポートするようになりました。**

**logging RHEL システムロールは、ファイル入力に対する新しい便利な `severity` と `facility` オプション、およびファイルと転送出力に対する新しい `template` オプションを備えています。**`template` オプションでは、`traditional` パラメーターを使用して従来の時刻形式を指定し、`syslog` パラメーターを使用して syslog プロトコル 23 形式を指定し、`modern` パラメーターを使用してモダンスタイル形式を指定します。そうすることで、**logging** ロールを使用して、`severity` と `facility` でフィルタリングしたり、`template` で出力形式を指定したりできるようになりました。

([BZ#2075116](#))

**RHEL システムロールは、ファクト収集が無効になっている Playbook でも利用できるようになりました。**

Ansible ファクト収集は、パフォーマンスまたはその他の理由により、環境内で無効になっている場合があります。以前は、このような設定で RHEL システムロールを使用することはできませんでした。今回の更新により、システムは設定内の `ANSIBLE_GATHERING=explicit` パラメーターと Playbook 内の `gather_facts: false` パラメーターを検出し、`setup` モジュールを使用して、指定されたロールに必要なファクトのみを収集します (ファクトキャッシュから取得できない場合)。



## 注記

パフォーマンスのために Ansible ファクト収集を無効にしている場合は、代わりに Ansible ファクトキャッシングを有効にできます。これにより、ソースからそれらを取得する際にパフォーマンスに影響が及びません。

(BZ#2079008)

**sshd RHEL システムロールは、ドロップインディレクトリーの include ディレクティブを検証します。**

RHEL 9 の **sshd** RHEL システムロールは、drop-in ディレクトリー内のファイルのみを管理しますが、以前はディレクトリーがメインの **sshd\_config** ファイルから含まれていることを確認していませんでした。今回の更新で、ロールは、**sshd\_config** に drop-in ディレクトリーの include ディレクティブが含まれていることを確認します。その結果、ロールは提供された設定をより確実に適用します。

(BZ#2086934)

**sshd RHEL システムロールは、`/etc/ssh/sshd_config` から管理できます。**

RHEL 9 管理対象ノードに適用される **sshd** RHEL システムロールは、SSH 設定を drop-in ディレクトリー (デフォルトでは `/etc/ssh/sshd_config.d/00-ansible_system_role.conf`) に配置します。以前は、`/etc/ssh/sshd_config` ファイルを変更すると、`00-ansible_system_role.conf` のデフォルト値が上書きされていました。今回の更新により、`00-ansible_system_role.conf` のシステムデフォルト値を保持しながら、`00-ansible_system_role.conf` の代わりに `/etc/ssh/sshd_config` を使用して SSHD を管理できるようになりました。

(BZ#2086935)

**masquerade または `icmp_block_inversion` を設定する場合、`firewall` RHEL システムロールは `state` パラメーターを必要としません。**

カスタムファイアウォールゾーンを設定する場合、変数 **masquerade** と **icmp\_block\_inversion** はブール値で設定します。**true** の値は **state: present** を意味し、**false** の値は **state: Absent** を意味します。したがって、**masquerade** または **icmp\_block\_inversion** を設定する場合、**state** パラメーターは必要ありません。

(BZ#2093437)

**metrics ロールは、`postfix` パフォーマンスデータをエクスポートできるようになりました。**

記録と詳細なパフォーマンス分析のために、**metrics** ロールで新しい **metrics\_from\_postfix** ブール変数を使用できるようになりました。今回の機能強化により、変数を設定すると、システムで **pmdapostfix** メトリックエージェントが有効になり、**postfix** に関する統計が利用可能になります。

(BZ#2079114)

**デフォルトで、`storage` システムロールの詳細度が低くなりました**

**storage** ロール出力の詳細レベルがデフォルトで低くなりました。今回の更新により、ユーザーは **storage** ロール出力の詳細レベルを上げて、Ansible 詳細レベル 1 以上を使用している場合にのみデバッグ出力を生成できるようになりました。

(BZ#2056480)

**metrics システムロールは、ヘッダーに適切な `ansible_managed` コメントを含むファイルを生成するようになりました**

以前は、**metrics** ロールは、ロールによって生成されたファイルに **ansible\_managed** ヘッダーコメントを追加しませんでした。この修正により、**metrics** ロールは生成するファイルに **ansible\_managed** ヘッダーコメントを追加するようになりました。その結果、ユーザーは **metrics** ロールによって生成されたファイルを簡単に識別できます。

(BZ#2065215)

**postfix** システムロールは、ヘッダーに適切な **ansible\_managed** コメントを含むファイルを生成するようになりました

以前は、**postfix** ロールは、ロールによって生成されたファイルに **ansible\_managed** ヘッダーコメントを追加しませんでした。この修正により、**postfix** ロールは生成するファイルに **ansible\_managed** ヘッダーコメントを追加するようになりました。その結果、ユーザーは **postfix** ロールによって生成されたファイルを簡単に識別できます。

(BZ#2065216)

以前の設定をオーバーライドする **postfix** RHEL システムロールの新しいオプションが追加されました

グループ内で **postfix** 設定が異なるシステムグループを管理していると、設定を一貫させる必要があるバイアがあります。この機能拡張により、**postfix\_conf** ディクショナリー内で **previous: replaced** オプションを指定して、既存の設定を削除し、クリーンな **postfix** インストール上に目的の設定を適用できます。その結果、既存の **postfix** 設定を消去して、管理対象のすべてのシステムで一貫性を確保できます。

(BZ#2065218)

**firewall** RHEL システムロールの **absent** および **present** 状態を使用して、サービスを追加、更新、削除できるようになりました

この機能拡張により、**present** の状態を使用してポート、モジュール、プロトコル、サービス、および宛先アドレスを追加したり、**absent** 状態を使用してそれらを削除したりできます。**firewall** RHEL システムロールで **absent** と **present** の状態を使用するには、**permanent** オプションを **true** に設定する必要があることに注意してください。**permanent** オプションを **true** に設定すると、状態の設定は変更されるまで適用され、ロールをリロードしても影響を受けません。

(BZ#2100297)

**firewall** システムロールは、PCI デバイス ID を使用してゾーンにインターフェイスを追加または削除できます

**firewall** システムロールは、PCI デバイス ID を使用して、ネットワークインターフェイスをゾーンに割り当てたり、ゾーンから削除したりできるようになりました。以前は、インターフェイス名ではなく PCI デバイス ID しか分からない場合、ユーザーはまず対応するインターフェイス名を識別し、**firewall** システムロールを使用する必要がありました。今回の更新により、**firewall** システムロールは PCI デバイス ID を使用して、ゾーン内のネットワークインターフェイスを管理できるようになりました。

(BZ#2100939)

**network** RHEL System Role は、**nmstate** API を使用したネットワーク設定をサポートします

今回の更新により、**network** RHEL システムロールは、**nmstate** API を介したネットワーク設定をサポートします。ユーザーは、接続プロファイルを作成する代わりに、必要なネットワーク状態の設定をネットワークインターフェイスに直接適用できるようになりました。この機能により、ネットワークの部分的な設定も可能になります。これには次の利点があります。

- ネットワーク設定の複雑さの軽減

- 信頼できるネットワーク状態の変更適用方法
- ネットワーク設定全体の追跡が不要

(BZ#2100979)

### カスタムリスニングポートを設定するための新しい cockpit システムロール変数

**cockpit** システムロールには、デフォルトの 9090 ポート以外のカスタムリスニングポートを設定できる **cockpit\_port** 変数が導入されています。カスタムのリスニングポートを設定する場合は、Web コンソールがそのポートでリスンできるように SELinux ポリシーを調整する必要があることに注意してください。

(BZ#2115159)

### firewall RHEL システムロールは、Ansible ファクトを提供できます

今回の機能拡張により、引数なしで playbook に **firewall:** 変数を含めることで、すべてのシステムから **firewall** RHEL システムロールの Ansible ファクトを収集できるようになりました。Ansible ファクトのより詳細なバージョンを収集するには、**detailed: true** 引数を使用します。以下はその例です。

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

(BZ#2115160)

### selinux RHEL システムロールに seuser と selevel の設定を追加しました

SELinux コンテキストファイルシステムのマッピング設定時に、**seuser** および **selevel** パラメーターを設定する必要がある場合があります。今回の更新により、**selinux\_fcontext** でオプション引数 **seuser** および **selevel** を使用し、SELinux コンテキストファイルシステムマッピングで SELinux ユーザーとレベルを指定できるようになりました。

(BZ#2115162)

## 4.16. 仮想化

### ap-check が RHEL 8 で利用可能になりました

**mdevctl** ツールは、新しい **ap-check** サポートユーティリティを提供するようになりました。**mdevctl** を使用して、**matrix** および **vfio-ap** デバイスだけでなく、仮想マシンへのパススルー使用が許可されている暗号化アダプターとドメインを永続的に設定できます。**mdevctl** を使用すると、IPL のたびにこれらのアダプター、ドメイン、デバイスを再設定する必要がなくなります。さらに、**mdevctl** は、ディストリビューターがそれらを再設定する他の方法を発明するのを防ぎます。

**vfio-ap** デバイスに対して **mdevctl** コマンドを呼び出すと、新しい **ap-check** サポートユーティリティが **mdevctl** コマンドの一部として呼び出され、**vfio-ap** デバイス設定に対して追加の有効性チェックが実行されます。

さらに、**chzdev** ツールは、**vfio-ap** デバイスで使用できる AP リソースを決定する、システム全体の補助プロセッサ (AP) マスク設定を管理する機能を提供するようになりました。**chzdev** を使用すると、関連付けられた **udev** ルールを生成することで、これらの設定を永続化できます。**lszdev** を使用して、システム全体の AP マスク設定を照会できるようになりました。

(BZ#1660911)

## IBM Z 上の一部の VM は、896 バイトを超えるカーネルコマンドラインで起動できるようになりました

以前は、RHEL 8 IBM Z ホストでの仮想マシン (VM) の起動は、VM のカーネルコマンドラインが 896 バイトより長い場合、常に失敗していました。今回の更新により、QEMU エミュレーターは 896 バイトを超えるカーネルコマンドラインを処理できるようになりました。その結果、VM カーネルがサポートしている場合、非常に長いカーネルコマンドラインで VM の QEMU ダイレクトカーネルブートを使用できるようになりました。具体的には、896 バイトを超えるコマンドラインを使用するには、VM で Linux カーネルバージョン 5.16-rc1 以降を使用する必要があります。

(BZ#2043830)

### 複数のスレッドを使用した VM メモリーの事前割り当て

次の例のように、ドメイン XML 設定で仮想マシン (VM) メモリー割り当て用に複数の CPU スレッドを定義できるようになりました。

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

これにより、VM の起動時にメモリーページの割り当てに複数のスレッドが使用されるようになります。その結果、特に VM に大量の RAM が割り当てられ、hugepage によってバックアップされている場合、複数の割り当てスレッドが設定されている VM の起動が大幅に高速化されます。

(BZ#2067126)

### ESXi ハイパーバイザーと SEV-ES が完全にサポートされるようになりました

AMD Secure Encrypted Virtualization-Encrypted State (SEV-ES) を有効にして、VMware の ESXi ハイパーバイザー (バージョン 7.0.2 以降) で RHEL 仮想マシンのセキュリティーを確保できるようになりました。この機能は、以前は RHEL 8.4 でテクノロジープレビューとして導入されていました。今回、完全にサポートされるようになりました。

(BZ#1904496)

### IBM Z でのセキュアな実行がリモート認証をサポートするようになりました

IBM Z アーキテクチャーの Secure Execution 機能は、リモート認証をサポートするようになりました。**pvattest** ユーティリティーは、Secure Execution が有効になっている仮想マシン (VM) の整合性を検証するリモート認証要求を作成できます。

さらに、Secure Execution VM で Guest Interruption State Area (GISA) メカニズムが有効になり、ホストオペレーティングシステムを完全にバイパスすることで、割り込みを仮想マシンに直接配信できるようになりました。

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

## 4.17. クラウド環境の RHEL

### RHEL 仮想マシンが Ampere Altra アーキテクチャーでサポートされるようになりました

今回の更新により、RHEL オペレーティングシステムの実行が、Ampere® Altra® アーキテクチャーに基づくプロセッサを搭載した Azure Virtual Machines でサポートされるようになりました。

(JIRA:RHELPLAN-121252)

## open-vm-tools が 12.0.5 にリベースされました

**open-vm-tools** パッケージがバージョン 12.0.5 にアップグレードされ、多数のバグ修正および新機能が追加されました。最も注目すべきは、Salt Minion ツールをゲスト OS 変数で管理するためのサポートが追加されたことです。

(BZ#2061193)

## cloud-init の新しい SSH モジュール

今回の更新で、**cloud-init** ユーティリティーに SSH モジュールが追加され、インスタンスの作成時にホストキーが自動的に生成されるようになりました。

この変更により、デフォルトの **cloud-init** 設定が更新されました。したがって、ローカルの変更があった場合は、`/etc/cloud/cloud.cfg` に `"ssh_genkeytypes: ['rsa', 'ecdsa', 'ed25519']"` 行が含まれていることを確認してください。

そうしないと、**sshd** サービスを起動できないイメージが **cloud-init** によって作成されます。この問題が発生した場合は、次の手順に従って問題を回避してください。

1. `/etc/cloud/cloud.cfg` ファイルに次の行が含まれていることを確認します。

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

2. `/etc/ssh/ssh_host_*` ファイルがインスタンスに存在するかどうかを確認します。
3. `/etc/ssh/ssh_host_*` ファイルが存在しない場合は、次のコマンドを使用してホストキーを生成します。

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

4. **sshd** サービスを再起動します。

```
systemctl restart sshd
```

(BZ#2115791)

## 4.18. コンテナ

### Container Tools パッケージが更新されました

Podman、Buildah、Skopeo、crun、および runc ツールを含む Container Tools パッケージが利用可能になりました。今回の更新で、以前のバージョンに対するバグ修正および機能拡張のリストが追加されました。

主な変更点は、以下のとおりです。

- **podman pod create** コマンドが、CPU とメモリーの制限の設定をサポートするようになりました。Pod 内のすべてのコンテナに制限を設定できますが、Pod 内の個々のコンテナには独自の制限を設定できます。
- **podman pod clone** コマンドは、既存の Pod のコピーを作成します。
- **podman play kube** コマンドは、**BlockDevice** および **CharDevice** ボリュームを使用したセキュリティコンテキスト設定をサポートするようになりました。

- **podman play kube** によって作成された Pod は、**podman-kube@<service>.service** (たとえば、**systemctl --user start podman-play-kube@\$(systemd-escape my.yaml).service** など) を使用して、systemd ユニットファイルによって管理できるようになりました。サービス。
- **podman push** および **podman push manifest** コマンドが sigstore 署名をサポートするようになりました。
- **podman network --opt isolate** コマンドを使用して、Podman ネットワークを分離できるようになりました。

Podman がバージョン 4.2 にアップグレードされました。注目すべき変更点の詳細については、[アップストリームリリースノート](#) を参照してください。

(JIRA:RHELPLAN-118463)

### GitLab Runner が Podman を使用して RHEL で利用できるようになりました

GitLab Runner 15.1 以降では、GitLab Runner Docker Executor でコンテナランタイムとして Podman を使用できます。詳細については、[GitLab のリリースノート](#) を参照してください。

(JIRA:RHELPLAN-100037)

### Podman が --health-on-failure オプションをサポートするようになりました

**podman run** および **podman create** コマンドで **--health-on-failure** オプションがサポートされるようになり、コンテナのステータスが異常になったときに実行するアクションを決定できるようになりました。

**--health-on-failure** オプションは、次の 4 つのアクションをサポートします。

- **none**: アクションを実行しません。これがデフォルトのアクションです。
- **kill**: コンテナを強制終了します。
- **restart**: コンテナを再起動します。
- **stop**: コンテナを停止します。



#### 注記

**restart** アクションを **--restart** オプションと組み合わせないでください。systemd ユニット内で実行する場合は、systemd の再起動ポリシーを利用する代わりに **kill** または **stop** アクションを使用することを検討してください。

(BZ#2097708)

### Netavark ネットワークスタックが利用可能になりました。

Podman 4.0 以降で利用可能な新しいネットワークスタックは、Netavark ネットワークセットアップツールと Aardvark DNS サーバーの 2 つのツールで設定されます。RHEL 8 では、以前はテクノロジープレビューとして利用できた Netavark スタックが完全にサポートされるようになりました。

このネットワークスタックには、次の機能があります。

- JSON 設定ファイルを使用したコンテナネットワークの設定

- ブリッジおよび MACVLAN インターフェイスを含むネットワークインターフェイスの作成、管理、および削除
- ネットワークアドレス変換 (NAT) やポートマッピングルールなどのファイアウォールの設定
- IPv4 および IPv6 (IPv4 and IPv6)
- 複数ネットワークのコンテナ機能の向上
- [aardvark-dns project](#) を使用したコンテナ DNS 解決



#### 注記

同じバージョンの Netavark スタックと Aardvark 権限のある DNS サーバーを使用する必要があります。

(JIRA:RHELPLAN-100039)

## 第5章 外部カーネルパラメーターへの重要な変更

本章では、システム管理者向けに、Red Hat Enterprise Linux 8.7 に同梱されるカーネルにおける重要な変更の概要について説明します。変更には、たとえば、**proc** エントリー、**sysctl** および **sysfs** のデフォルト値、ブートパラメーター、カーネル設定オプション、または重要な動作の変更などが含まれます。

### 新しいカーネルパラメーター

#### `idxd.tc_override = [HW]`

このパラメーターを **<bool>** 形式で使用すると、デバイスのデフォルトのトラフィッククラス設定をオーバーライドできます。

デフォルト値は **false (0)** に設定されます。

#### `kvm.eager_page_split = [KVM,X86]`

このパラメーターを使用すると、KVM がダーティロギング中にすべてのヒュージページをプロアクティブに分割するかどうかを制御できます。Eager page splitting は、巨大なページを遅延して分割するために必要な書き込み保護障害とメモリー管理ユニット (MMU) ロックの競合を排除することで、vCPU 実行の中断を減らします。

書き込みをほとんど実行しない、または VM メモリーの小さな領域にのみ書き込む VM ワークロードは、熱心なページ分割を無効にして、巨大なページを引き続き読み取りに使用できるようにすることでメリットが得られます。

積極的なページ分割の動作は、**KVM\_DIRTY\_LOG\_INITIALLY\_SET** オプションが有効か無効かによって異なります。

- 無効にすると、その **memslot** でダーティロギングが有効になっているときに、**memslot** 内のすべての huge ページが積極的に分割されます。
- 有効にすると、**KVM\_CLEAR\_DIRTY ioctl()** システムコール中に、ページがクリアされている場合にのみ、熱心なページ分割が実行されます。  
イーガーページ分割は現在、2次元ページング (TDP) MMU によってマップされた Huge Page の分割のみをサポートしています。

デフォルト値は **Y (on)** に設定されています。

#### `kvm.nx_huge_pages_recovery_period_ms = [KVM]`

このパラメーターを使用すると、KVM が 4 KiB ページを huge ページにザッピングする期間を制御できます。

- 値がゼロ以外の **N** の場合、KVM はページの一部を **N** ミリ秒ごとにザッピングします。
- 値が **0** の場合、KVM は比率に基づいて期間を選択し、ページが平均1時間後にザッピングされるようにします。  
デフォルト値は **0** に設定されています。

#### `mmio_stale_data = [X86,INTEL]`

このパラメーターを使用して、プロセッサメモリーにマップされた I/O (MMIO) の古いデータの脆弱性の緩和を制御できます。

Processor MMIO Stale Data は、MMIO 操作後にデータを公開できる脆弱性のクラスです。公開されたデータは、メタデータサーバー (MDS) および Transactional Asynchronous Abort (TAA) の影響を受ける同じ CPU バッファで開始または終了する可能性があります。そのため、MDS や TAA と同様に、影響を受ける CPU バッファをクリアすることで軽減できます。

利用可能なオプションは以下のとおりです。

- **full**: 脆弱な CPU で軽減策を有効にします
- **full,nosmt**: 緩和策を有効にし、脆弱な CPU で SMT を無効にします。
- **off**: 無条件に緩和を無効にします  
MDS または TAA の影響を受けるマシンでは、アクティブな MDS または TAA の軽減策によって **mmio\_stale\_data=off** を防ぐことができます。これらの脆弱性は同じメカニズムで軽減されるからです。したがって、この軽減策を無効にするには、**mds=off** と **tsx\_async\_abort=off** も指定する必要があります。

このオプションを指定しないことは、**mmio\_stale\_data=full** と同等です。

詳細については、[Documentation/admin-guide/hw-vuln/processor\\_mmio\\_stale\\_data.rst](#) を参照してください。

#### **rcutree.rcu\_delay\_page\_cache\_fill\_msec = [KNL]**

このパラメーターを使用すると、メモリー不足の状態に応じて、ページキャッシュのリフィルの遅延をミリ秒単位で設定できます。許可される値の範囲は **0:100000** です。

#### **rcuscale.kfree\_rcu\_test\_double = [KNL]**

このパラメーターを使用すると、**kfree\_rcu()** 関数の二重引数バリエーションをテストできます。このパラメーターが **rcuscale.kfree\_rcu\_test\_single** と同じ値である場合は、単一引数と二重引数の両方のバリエーションがテストされます。

#### **rcuscale.kfree\_rcu\_test\_single = [KNL]**

このパラメーターを使用すると、**kfree\_rcu()** 関数の単一引数バリエーションをテストできます。このパラメーターが **rcuscale.kfree\_rcu\_test\_double** と同じ値である場合は、単一引数と二重引数の両方のバリエーションがテストされます。

#### **retbleed = [X86]**

このパラメーターを使用して、リターン命令による任意の投機的コード実行 (RETbleed) 脆弱性の緩和を制御できます。利用可能なオプションは以下のとおりです。

- **off**: 緩和なし
- **auto**: 軽減策を自動的に選択します
- **auto,nosmt**: 緩和策を自動的に選択し、必要に応じて完全な緩和策として SMT を無効にします (STIBP を使用しない Zen1 以前のみ)。
- **ibpb**: 基本ブロック境界での短い投機ウィンドウも軽減します。安全で最高のパフォーマンスへの影響。
- **unret**: トレーニングされていないリターンサンクを強制的に有効にします。AMD f15h-f17h ベースのシステムでのみ有効です。
- **unret,nosmt**: **unret** オプションと同様に、STIBP が利用できない場合に SMT を無効にします。  
**auto** オプションを選択すると、実行時に CPU に応じて緩和方法が選択されます。

このオプションを指定しないことは、**retbleed=auto** と同等です。

#### **s390\_iommu\_aperture = [KNL,S390]**

このパラメーターを使用すると、DMA および IOMMU API を介してアクセスできるデバイスごとの DMA アドレス空間のサイズを、メインメモリーのサイズの小数として指定できます。

- デフォルト値は **1** に設定されています。これは、ハードウェアでサポートされている場合、インストールされている物理メモリーと同じ数の DMA アドレスを同時に使用できることを意味し、すべてのメモリーを一度にマップします。
- 値を **2** にすると、すべてのメモリーを 2 回マップできます。
- **0** の値は、ハードウェアによって与えられるもの以外の制限を課しませんが、テーブルのメモリー使用量が大幅に増加します。

## 更新されたカーネルパラメーター

### acpi\_sleep = [HW,ACPI]

フォーマット: {s3\_bios、s3\_mode、s3\_beeep、s4\_hwsig、s4\_nohwsig、old\_ordering、nonvs、sci\_force\_enable、nobl}

- **s3\_bios** および **s3\_mode** の詳細については、[Documentation/power/video.rst](#) を参照してください。
- **s3\_beeep** はデバッグ用です。カーネルのリアルモードエントリーポイントが呼び出されるとすぐに、PC のスピーカーからビープ音が鳴ります。
- **s4\_hwsig** により、カーネルは休止状態からの再開中に ACPI ハードウェア署名をチェックし、変更されている場合は再開を適切に拒否します。デフォルトの動作は、**s4\_hwsig** オプションが有効になっていない限り、再開を許可し、署名が変更されたときに単に警告することです。
- **s4\_nohwsig** は、再開中に ACPI ハードウェア署名が使用されたり、警告されたりするのを防ぎます。**old\_ordering** は、デバイスを低電力状態にすることに関して、**\_PTS** 制御メソッドの ACPI 1.0 順序付けを強制します。デフォルトでは、**\_PTS** の ACPI 2.0 順序が使用されます。
- **nonvs** は、カーネルがサスペンド、ハイバネーション、およびレジューム中に ACPI NVS メモリーを保存および復元するのを防ぎます。
- **sci\_force\_enable** により、カーネルは S1/S3 からの再開時に直接 **SCI\_EN** を設定します。この動作は ACPI 仕様と反していますが、一部の破損したシステムはそれなしでは機能しません。
- **nobl** は、システムのサスペンドとレジュームに関して何らかの点で正しく動作しないことが知られているシステムの内部拒否リストを無視します。このオプションを賢く使用してください。詳細については、[Documentation/power/video.rst](#) を参照してください。

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

このパラメーターを使用すると、次のように物理メモリー領域を上から割り当てることができます。

- システムに 4 GB を超える RAM がインストールされている場合、物理メモリー領域は 4 GB を超える可能性があります。
- システムに 4 GB 未満の RAM がインストールされている場合、物理メモリー領域は 4 GB 未満に割り当てられます (利用可能な場合)。  
**crashkernel=X** パラメーターが指定されている場合、このパラメーターは無視されます。

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

**crashkernel=X,high** を渡すと、カーネルは 4 GB を超える物理メモリー領域を割り当てることができます。これにより、ある程度の低メモリー (たとえば、**swiotlb** は少なくとも 64M+32K 低メモリーが必要) と、32 ビットデバイスの DMA バッファが使い果たされないようにするのに十分な余分な低メモリーを必要とするシステムで、2 番目のカーネルクラッシュが発生します。カーネルは、4 GB 未満に少なくとも 256 M を自動的に割り当てようとします。このパラメーターを使用すると、代わりに 2 番目のカーネルに 4 GB 未満の範囲を指定できます。

- **0**: 低割り当てを無効にします。**crashkernel=X,high** が使用されていない場合、または予約済みメモリーが 4 GB 未満の場合は無視されます。

#### **kvm.nx\_huge\_pages\_recovery\_ratio = [KVM]**

このパラメーターを使用すると、4KiB ページを定期的に巨大なページにザッピングする回数を制御できます。

- **0** はリカバリーを無効にします
- **N** KVM は、4KiB ページの **1/N** を周期ごとに消去します。デフォルトは **60** に設定されます。

#### **module.sig\_enforce = norid [S390]**

このパラメーターを使用すると、RID フィールドを無視して、PCI 機能ごとに 1 つの PCI ドメインの使用を強制できます。

#### **rcu\_nocbs[=cpu-list] = [KNL]**

オプションの引数は CPU リストです。

**CONFIG\_RCU\_NOCB\_CPU=y** でビルドされたカーネルでは、no-callback CPU モードを有効にできます。これにより、そのような CPU コールバックが softirq コンテキストで呼び出されるのを防ぎます。そのような CPU の RCU コールバックの呼び出しは、代わりに、その目的のために作成された **rcuox/N kthreads** にオフロードされます。ここで、**x** は RCU-preempt を表す **p**、RCU-sched を表す **s**、猶予期間を仲介する **kthread** を表す **g** です。**N** は CPU 番号です。これにより、オフロードされた CPU の OS ジッターが減少し、HPC およびリアルタイムのワークロードに役立ちます。また、非対称マルチプロセッサのエネルギー効率も改善できます。

- **cpulist** が引数として渡された場合、指定された CPU のリストはブートからコールバックなしモードに設定されます。
- **=** 記号と **cpulist** 引数を省略した場合、起動時に CPU がコールバックなしモードに設定されることはありませんが、**cpuset** を使用して実行時にモードを切り替えることができます。

#### **spectre\_v2\_user = [X86]**

このパラメーターを使用すると、ユーザー空間タスク間の Spectre バリエーション 2 (間接分岐スペキュレーション) の脆弱性の緩和を制御できます。

- **auto**: カーネルは、利用可能な CPU 機能と脆弱性に応じて緩和策を選択します。
- デフォルトの軽減策は **prctl** に設定されています。
- このオプションを指定しないことは、**spectre\_v2\_user=auto** と同等です。

#### **spec\_store\_bypass\_disable = [X86]**

このパラメーターを使用すると、投機的ストアバイパス (SSB) の脆弱性を軽減するための最適化を使用するかどうかを制御できます。

- このオプションを指定しないことは、**spec\_store\_bypass\_disable=auto** と同等です。

- デフォルトの軽減策は **prctl** に設定されています。

## 新しい sysctl パラメーター

### perf\_user\_access = [ARM64]

このパラメーターを使用すると、パフォーマンスイベントカウンターを読み取るためのユーザー空間アクセスを制御できます。

- **1** に設定すると、ユーザー空間はパフォーマンスモニターカウンターレジスターを直接読み取ることができます。
- デフォルトは **0** に設定されており、これは **access disabled** であることを意味します。詳細については、[Documentation/arm64/perf.rst](#) を参照してください。

### force\_cgroup\_v2\_swappiness

このパラメーターを使用すると、**cgroupsV1** でのみ使用可能な cgroup ごとの swappiness 値を非推奨にすることができます。**systemd** の設計上の選択により、すべてのシステムおよびユーザープロセスのほとんどは **cgroup** 内で実行されます。さらに、これらの **cgroup** swappiness 値のデフォルトは **60** です。これにより、システムの swappiness 値がシステムのスワップ動作にほとんど影響を与えないという望ましくない結果が生じる可能性があります。

**cgroup** ごとの swappiness 機能を使用する場合は、**force\_cgroup\_v2\_swappiness=1** でシステムを設定して、システム全体でより一貫した swappiness 動作を持たせることができます。

これは RHEL 固有の機能であることに注意してください。

## 第6章 デバイスドライバー

### 6.1. 新しいドライバー

#### ネットワークドライバー

- Maxlinear イーサネット GPY ドライバー (**mxl-gpy**)
- Realtek 802.11ax ワイヤレス 8852A ドライバー (**rtw89\_8852a**)
- Realtek 802.11ax ワイヤレス 8852AE ドライバー (**rtw89\_8852ae**)

#### グラフィックドライバーとその他のドライバー

- MHI ホストインターフェイス (**mhi**)
- Modem Host Interface (MHI) PCI コントローラードライバー **spectre\_v2\_user=auto**
- IDXD driver dsa\_bus\_type driver (**idxd\_bus**)
- AMD PassThru DMA ドライバー (**ptdma**)
- Cirrus Logic DSP サポート (**cs\_dsp**)
- DRM DisplayPort ヘルパー (**drm\_dp\_helper**)
- DRM Buddy Allocator (**drm\_buddy**)
- DRM SHMEM メモリ-マネジメントヘルパー (**drm\_shmem\_helper**)
- bochs dispi インターフェイスを使用する DRM ドライバー (**bochs**)
- Intel® PMT Class ドライバー (**pmt\_class**)
- Intel® PMT Crashlog ドライバー (**pmt\_crashlog**)
- Intel® PMT Telemetry ドライバー (**pmt\_telemetry**)
- Intel® スピードセレクトインターフェイスドライバー (**isst\_if\_common**)
- Intel® スピード選択インターフェイス mailbox ドライバー (**isst\_if\_mbox\_msr**)
- Intel® speed select interface pci mailbox driver (**isst\_if\_mbox\_pci**)
- Intel® speed select interface mmio driver (**isst\_if\_mmio**)
- Intel® Software Defined Silicon ドライバー (**intel\_sdsi**)
- Intel® Extended Capabilities aux バスドライバー (**intel\_vsec**)
- ISH ISHTP eclite クライアント OPERATION ドライバー (**ishtp\_eclite**)
- シリアルマルチインスタンス化擬似デバイスドライバー (**serial-multi-instantiate**)
- AMD® SPI Master Controller Driver (**spi-amd**)

## 6.2. 更新されたドライバー

### ネットワークドライバー

- VMware vmxnet3 virtual NIC ドライバー (**vmxnet3**) がバージョン 1.7.0.0-k に更新されました。
- Intel® PRO/1000 Network Driver (**e1000e**) がバージョン 4.18.0-425.3.1 に更新されました。
- Intel® Ethernet Switch Host Interface Driver (**fm10k**) がバージョン 4.18.0-425.3.1 に更新されました。
- Intel® Ethernet Connection XL710 Network Driver (**i40e**) がバージョン 4.18.0-425.3.1 に更新されました。
- Intel® Ethernet Adaptive Virtual Function Network Driver (**iavf**) がバージョン 4.18.0-425.3.1 に更新されました。
- Intel® Gigabit Ethernet Network Driver (**igb**) がバージョン 4.18.0-425.3.1 に更新されました。
- Intel® Gigabit Virtual Function Network Driver (**igbvf**) がバージョン 4.18.0-425.3.1 に更新されました。
- Intel® 2.5G Ethernet Linux Driver (**igc**) がバージョン 4.18.0-425.3.1 に更新されました。
- Intel® 10 Gigabit PCI Express Network Driver (**ixgbe**) がバージョン 4.18.0-425.3.1 に更新されました。
- Intel® 10 Gigabit Virtual Function Network Driver (**ixgbev**) がバージョン 4.18.0-425.3.1 に更新されました。
- Mellanox 第 5 世代ネットワークアダプター (ConnectX シリーズ) コアドライバー (**mlx5\_core**) がバージョン 4.18.0-425.3.1 に更新されました。

### ストレージドライバー

- Emulex LightPulse Fibre Channel SCSI ドライバー (**lpfc**) がバージョン 14.0.0.15 に更新されました。
- MPI3 Storage Controller Device Driver (**mpi3mr**) がバージョン 8.0.0.69.0 に更新されました。
- LSI MPT Fusion SAS 3.0 デバイスドライバー (**mpt3sas**) がバージョン 42.100.00.00 に更新されました。
- QLogic Fibre Channel HBA ドライバー (**qla2xxx**) がバージョン 10.02.07.400-k に更新されました。
- Microchip Smart Family Controller (**smartpqi**) のドライバーがバージョン 2.1.18-045 に更新されました。

### グラフィックおよびその他ドライバーの更新

- VMware SVGA デバイス (**vmwgfx**) のスタンドアロン drm ドライバーがバージョン 2.20.0.0 に更新されました。

## 第7章 利用可能な BPF 機能

この章では、Red Hat Enterprise Linux 8 のこのマイナーバージョンのカーネルで利用可能な **Berkeley Packet Filter (BPF)** 機能の完全なリストを提供します。表には次のリストが含まれます。

- システム設定とその他のオプション
- 利用可能なプログラムの種類とサポートされているヘルパー
- 利用可能なマップの種類

この章には、**bpftool feature** コマンドの自動生成された出力が含まれています。

表7.1 システム設定とその他のオプション

オプション	値
unprivileged_bpf_disabled	1 (特権ユーザーに限定された bpf() syscall、リカバリーなし)
JIT コンパイラー	1 (有効)
JIT コンパイラーの強化	1 (権限のないユーザーに対して有効)
JIT コンパイラー kallsyms エクスポート	1 (ルートで有効)
非特権ユーザーの JIT のメモリー制限	264241152
CONFIG_BPF	y
CONFIG_BPF_SYSCALL	y
CONFIG_HAVE_EBPF_JIT	y
CONFIG_BPF_JIT	y
CONFIG_BPF_JIT_ALWAYS_ON	y
CONFIG_DEBUG_INFO_BTF	y
CONFIG_DEBUG_INFO_BTF_MODULES	n
CONFIG_CGROUPS	y
CONFIG_CGROUP_BPF	y
CONFIG_CGROUP_NET_CLASSID	y
CONFIG_SOCK_CGROUP_DATA	y

オプション	値
CONFIG_BPF_EVENTS	y
CONFIG_KPROBE_EVENTS	y
CONFIG_UPROBE_EVENTS	y
CONFIG_TRACING	y
CONFIG_FTRACE_SYSCALLS	y
CONFIG_FUNCTION_ERROR_INJECTION	y
CONFIG_BPF_KPROBE_OVERRIDE	y
CONFIG_NET	y
CONFIG_XDP_SOCKETS	y
CONFIG_LWTUNNEL_BPF	y
CONFIG_NET_ACT_BPF	m
CONFIG_NET_CLS_BPF	m
CONFIG_NET_CLS_ACT	y
CONFIG_NET_SCH_INGRESS	m
CONFIG_XFRM	y
CONFIG_IP_ROUTE_CLASSID	y
CONFIG_IPV6_SEG6_BPF	n
CONFIG_BPF_LIRC_MODE2	n
CONFIG_BPF_STREAM_PARSER	y
CONFIG_NETFILTER_XT_MATCH_BPF	m
CONFIG_BPFILTER	n
CONFIG_BPFILTER_UMH	n

オプション	値
CONFIG_TEST_BPF	m
CONFIG_HZ	1000
bpf() syscall	available
大きなプログラムサイズの制限	available

表7.2 利用可能なプログラムの種類とサポートされているヘルパー

プログラムの種類	利用可能なヘルパー
socket_filter	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_skb_load_bytes, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_skb_load_bytes_relative, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
kprobe	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_override_return, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

プログラムの種類	利用可能なヘルパー
sched_cls	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_l3_csum_replace, bpf_l4_csum_replace, bpf_tail_call, bpf_clone_redirect, bpf_get_cgroup_classid, bpf_skb_vlan_push, bpf_skb_vlan_pop, bpf_skb_get_tunnel_key, bpf_skb_set_tunnel_key, bpf_redirect, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_get_tunnel_opt, bpf_skb_set_tunnel_opt, bpf_skb_change_proto, bpf_skb_change_type, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_change_tail, bpf_skb_pull_data, bpf_csum_update, bpf_set_hash_invalid, bpf_get_numa_node_id, bpf_skb_change_head, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_set_hash, bpf_skb_adjust_room, bpf_skb_get_xfrm_state, bpf_skb_load_bytes_relative, bpf_fib_lookup, bpf_skb_cgroup_id, bpf_skb_ancestor_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sk_fullsock, bpf_tcp_sock, bpf_skb_ecn_set_ce, bpf_get_listener_sock, bpf_skc_lookup_tcp, bpf_tcp_check_syncookie, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_tcp_gen_syncookie, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_sk_assign, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_csum_level, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_skb_cgroup_classid, bpf_redirect_neigh, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_redirect_peer, bpf_ktime_get_coarse_ns, bpf_check_mtu, bpf_for_each_map_elem, bpf_snprintf
sched_act	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_l3_csum_replace, bpf_l4_csum_replace, bpf_tail_call, bpf_clone_redirect, bpf_get_cgroup_classid, bpf_skb_vlan_push, bpf_skb_vlan_pop, bpf_skb_get_tunnel_key, bpf_skb_set_tunnel_key, bpf_redirect, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_get_tunnel_opt, bpf_skb_set_tunnel_opt, bpf_skb_change_proto, bpf_skb_change_type, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_change_tail, bpf_skb_pull_data, bpf_csum_update, bpf_set_hash_invalid, bpf_get_numa_node_id, bpf_skb_change_head, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_set_hash, bpf_skb_adjust_room, bpf_skb_get_xfrm_state, bpf_skb_load_bytes_relative, bpf_fib_lookup, bpf_skb_cgroup_id, bpf_skb_ancestor_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sk_fullsock, bpf_tcp_sock, bpf_skb_ecn_set_ce, bpf_get_listener_sock, bpf_skc_lookup_tcp, bpf_tcp_check_syncookie, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_tcp_gen_syncookie, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_sk_assign, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_csum_level, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_skb_cgroup_classid, bpf_redirect_neigh, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_redirect_peer, bpf_ktime_get_coarse_ns, bpf_check_mtu, bpf_for_each_map_elem, bpf_snprintf

プログラムの種類	利用可能なヘルパー
tracepoint	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
xdp	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_redirect, bpf_perf_event_output, bpf_csum_diff, bpf_get_current_task, bpf_get_numa_node_id, bpf_xdp_adjust_head, bpf_redirect_map, bpf_xdp_adjust_meta, bpf_xdp_adjust_tail, bpf_fib_lookup, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_skc_lookup_tcp, bpf_tcp_check_syncookie, bpf_tcp_gen_syncookie, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_check_mtu, bpf_for_each_map_elem, bpf_snprintf
perf_event	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_perf_prog_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_read_branch_records, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

プログラムの種類	利用可能なヘルパー
cgroup_skb	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_skb_load_bytes, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_skb_load_bytes_relative, bpf_skb_cgroup_id, bpf_get_local_storage, bpf_skb_ancestor_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sk_fullsock, bpf_tcp_sock, bpf_skb_ecn_set_ce, bpf_get_listener_sock, bpf_skc_lookup_tcp, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_sk_cgroup_id, bpf_sk_ancestor_cgroup_id, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
cgroup_sock	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_get_cgroup_classid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sk_storage_get, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_netns_cookie, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
lwt_in	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_cgroup_classid, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_pull_data, bpf_get_numa_node_id, bpf_lwt_push_encap, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

プログラムの種類	利用可能なヘルパー
lwt_out	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_cgroup_classid, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_pull_data, bpf_get_numa_node_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
lwt_xmit	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_l3_csum_replace, bpf_l4_csum_replace, bpf_tail_call, bpf_clone_redirect, bpf_get_cgroup_classid, bpf_skb_get_tunnel_key, bpf_skb_set_tunnel_key, bpf_redirect, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_get_tunnel_opt, bpf_skb_set_tunnel_opt, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_change_tail, bpf_skb_pull_data, bpf_csum_update, bpf_set_hash_invalid, bpf_get_numa_node_id, bpf_skb_change_head, bpf_lwt_push_encap, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_csum_level, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
sock_ops	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_setsockopt, bpf_sock_map_update, bpf_getsockopt, bpf_sock_ops_cb_flags_set, bpf_sock_hash_update, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_tcp_sock, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_load_hdr_opt, bpf_store_hdr_opt, bpf_reserve_hdr_opt, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

プログラムの種類	利用可能なヘルパー
sk_skb	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_tail_call, bpf_perf_event_output, bpf_skb_load_bytes, bpf_get_current_task, bpf_skb_change_tail, bpf_skb_pull_data, bpf_get_numa_node_id, bpf_skb_change_head, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_skb_adjust_room, bpf_sk_redirect_map, bpf_sk_redirect_hash, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_skc_lookup_tcp, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
cgroup_device	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_uid_gid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
sk_msg	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_cgroup_classid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_msg_redirect_map, bpf_msg_apply_bytes, bpf_msg_cork_bytes, bpf_msg_pull_data, bpf_msg_redirect_hash, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_msg_push_data, bpf_msg_pop_data, bpf_spin_lock, bpf_spin_unlock, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

プログラムの種類	利用可能なヘルパー
raw_tracepoint	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
cgroup_sock_addr	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_get_cgroup_classid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_setsockopt, bpf_getsockopt, bpf_bind, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_skc_lookup_tcp, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_netns_cookie, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
lwt_seg6local	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_cgroup_classid, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_pull_data, bpf_get_numa_node_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
lirc_mode2	サポート対象外

プログラムの種類	利用可能なヘルパー
sk_reuseport	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_skb_load_bytes, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_skb_load_bytes_relative, bpf_sk_select_reuseport, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
flow_dissector	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_skb_load_bytes, bpf_get_current_task, bpf_get_numa_node_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
cgroup_sysctl	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_uid_gid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sysctl_get_name, bpf_sysctl_get_current_value, bpf_sysctl_get_new_value, bpf_sysctl_set_new_value, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
raw_tracepoint_wri table	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

プログラムの種類	利用可能なヘルパー
cgroup_sockopt	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_uid_gid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_tcp_sock, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
tracing	サポート対象外

プログラムの種類	利用可能なヘルパー
struct_ops	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_l3_csum_replace, bpf_l4_csum_replace, bpf_tail_call, bpf_clone_redirect, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_get_cgroup_classid, bpf_skb_vlan_push, bpf_skb_vlan_pop, bpf_skb_get_tunnel_key, bpf_skb_set_tunnel_key, bpf_perf_event_read, bpf_redirect, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_get_stackid, bpf_csum_diff, bpf_skb_get_tunnel_opt, bpf_skb_set_tunnel_opt, bpf_skb_change_proto, bpf_skb_change_type, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_skb_change_tail, bpf_skb_pull_data, bpf_csum_update, bpf_set_hash_invalid, bpf_get_numa_node_id, bpf_skb_change_head, bpf_xdp_adjust_head, bpf_probe_read_str, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_set_hash, bpf_setsockopt, bpf_skb_adjust_room, bpf_redirect_map, bpf_sk_redirect_map, bpf_sock_map_update, bpf_xdp_adjust_meta, bpf_perf_event_read_value, bpf_perf_prog_read_value, bpf_getsockopt, bpf_override_return, bpf_sock_ops_cb_flags_set, bpf_msg_redirect_map, bpf_msg_apply_bytes, bpf_msg_cork_bytes, bpf_msg_pull_data, bpf_bind, bpf_xdp_adjust_tail, bpf_skb_get_xfrm_state, bpf_get_stack, bpf_skb_load_bytes_relative, bpf_fib_lookup, bpf_sock_hash_update, bpf_msg_redirect_hash, bpf_sk_redirect_hash, bpf_lwt_push_encap, bpf_lwt_seg6_store_bytes, bpf_lwt_seg6_adjust_srh, bpf_lwt_seg6_action, bpf_rc_repeat, bpf_rc_keydown, bpf_skb_cgroup_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_sk_select_reuseport, bpf_skb_ancestor_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_msg_push_data, bpf_msg_pop_data, bpf_rc_pointer_rel, bpf_spin_lock, bpf_spin_unlock, bpf_sk_fullsock, bpf_tcp_sock, bpf_skb_ecn_set_ce, bpf_get_listener_sock, bpf_skc_lookup_tcp, bpf_tcp_check_syncookie, bpf_sysctl_get_name, bpf_sysctl_get_current_value, bpf_sysctl_get_new_value, bpf_sysctl_set_new_value, bpf_strtol, bpf_strtoul, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_send_signal, bpf_tcp_gen_syncookie, bpf_skb_output, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_tcp_send_ack, bpf_send_signal_thread, bpf_jiffies64, bpf_read_branch_records, bpf_get_ns_current_pid_tgid, bpf_xdp_output, bpf_get_netns_cookie, bpf_get_current_ancestor_cgroup_id, bpf_sk_assign, bpf_ktime_get_boot_ns, bpf_seq_printf, bpf_seq_write, bpf_sk_cgroup_id, bpf_sk_ancestor_cgroup_id, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_csum_level, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_get_task_stack, bpf_load_hdr_opt, bpf_store_hdr_opt, bpf_reserve_hdr_opt, bpf_inode_storage_get, bpf_inode_storage_delete, bpf_d_path, bpf_copy_from_user, bpf_snprintf_btf, bpf_seq_printf_btf, bpf_skb_cgroup_classid, bpf_redirect_neigh, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_redirect_peer, bpf_task_storage_get, bpf_task_storage_delete, bpf_get_current_task_btf, bpf_bprm_opts_set, bpf_ktime_get_coarse_ns, bpf_ima_inode_hash, bpf_sock_from_file, bpf_check_mtu, bpf_for_each_map_elem, bpf_snprintf, bpf_sys_bpf, bpf_btf_find_by_name_kind, bpf_sys_close
ext	サポート対象外
lsm	サポート対象外

プログラムの種類	利用可能なヘルパー
sk_lookup	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_sk_assign, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

表7.3 利用可能なマップの種類

マップの種類	Available
ハッシュ	はい
array	はい
prog_array	はい
perf_event_array	はい
percpu_hash	はい
percpu_array	はい
stack_trace	はい
cgroup_array	はい
lru_hash	はい
lru_percpu_hash	はい
lpm_trie	はい
array_of_maps	はい
hash_of_maps	はい
devmap	はい
sockmap	はい

マップの種類	Available
cpumap	はい
xskmap	はい
sockhash	はい
cgroup_storage	はい
reuseport_sockarray	はい
percpu_cgroup_storage	はい
queue	はい
stack	はい
sk_storage	はい
devmap_hash	はい
struct_ops	いいえ
ringbuf	はい
inode_storage	はい
task_storage	いいえ

## 第8章 バグ修正

ここでは、ユーザーに大きな影響を及ぼしていた Red Hat Enterprise Linux 8.7 のバグで修正されたものを説明します。

### 8.1. インストーラーおよびイメージの作成

#### インストーラーは以前のバージョンのパッケージをインストールしなくなりました

以前は、インストーラーはインストールプロセス中に DNF 設定ファイルを正しくロードしませんでした。その結果、インストーラーは、RPM トランザクションで選択したパッケージの以前のバージョンをインストールすることがありました。

このバグは修正され、最新バージョンのパッケージのみがインストールリポジトリからインストールされるようになりました。パッケージの最新バージョンをインストールできない場合、インストールは想定どおり失敗します。

(BZ#1899494)

#### stage2 でネットワーク設定を変更しても Anaconda のインストールは成功します

以前は、`rd.live.ram` ブート引数を使用すると、Anaconda は `initramfs` でメモリーにインストールイメージをフェッチするために使用される NFS マウントポイントをアンマウントしませんでした。その結果、stage2 でネットワーク設定が変更された場合、インストールプロセスが応答しなくなるか、タイムアウトエラーで失敗する可能性があります。

この問題を修正するために、インストールイメージをメモリーにフェッチするために使用される NFS マウントポイントは、`switchroot` の前に `initramfs` でアンマウントされます。その結果、インストールプロセスは中断することなく完了します。

(BZ#1970726)

#### インストーラーは、インストール中に、暗号化されたデバイスのキックスタートファイルにないパスフレーズを要求します。

以前は、グラフィカルモードでインストーラーを実行するときに、キックスタートファイルでパスフレーズが指定されていない場合、インストーラーは暗号化されたデバイスのパスフレーズの入力を求めませんでした。その結果、キックスタートファイルで指定されたパーティショニングがインストール中に適用されませんでした。

今回の更新により、インストール中に表示され、不足しているパスフレーズを要求するダイアログウィンドウが追加されます。その結果、インストーラーは、キックスタートファイルで指定されたパーティショニングスキームを適切に適用します。

(BZ#2029101)

#### 条件付き依存関係を含むブループリントのパッケージのイメージが正常にビルドされるようになりました

以前は、Web コンソールを使用して、`ipa-client`、`cockpit`、`podman` などの条件付き依存関係を含むパッケージでブループリントをカスタマイズすると、依存関係がないためにビルドが失敗していました。その結果、`dep-solve` パッケージ中に条件付き依存関係が満たされませんでした。この問題は現在修正されており、条件付き依存関係を `dep-solve` するときにビルドが失敗することはなくなりました。

(BZ#2065734)

## 8.2. ソフトウェア管理

**DNF は、Reason Change Action タイプのアイテムを含むトランザクションを正しくロールバックするようになりました**

以前は、**Reason Change** Action タイプの項目を含むトランザクションで **dnf history rollback** コマンドを実行すると失敗しました。今回の更新で問題が修正され、**dnf 履歴のロールバック** が期待どおりに機能するようになりました。

([BZ#2060815](#))

## 8.3. シェルおよびコマンドラインツール

**パラメーターのない cmx 操作により CIM クライアントがクラッシュしなくなりました**

**cmx** 操作はメソッドを呼び出し、XML を返します。パラメーターは呼び出されたメソッドの名前を指定します。以前は、追加のパラメーターなしで **cmx** 操作を実行すると、コマンドライン **sblim-wbemcli** Common Information Model (CIM) クライアントがクラッシュしていました。今回の更新により、**cmx** 操作には、呼び出されたメソッドの名前を定義するパラメーターが必要になりました。このパラメーターなしで **cmx** 操作を呼び出すとエラーメッセージが表示され、CIM クライアントはクラッシュしなくなりました。

([BZ#2075807](#))

**opencv ライブラリーの cvSaveImage 関数がユーザーアプリケーションを終了させなくなりました。**

以前は、**opencv** ライブラリーは **cvSaveImage** 関数を正しく使用できませんでした。その結果、ユーザーアプリケーションが予期せず終了しました。今回の更新により、**cvSaveImage** 関数はイメージデータをディスクに書き込み、ユーザーアプリケーションを終了しなくなりました。

([BZ#2104776](#))

**/etc/fstab で UUID を更新しない場合、ReaR はエラーメッセージの表示に失敗しなくなりました**

これまで ReaR では、Universally Unique Identifier (UUID) が異なる場合に **/etc/fstab** の UUID を新しく作成されたパーティションの UUID と一致するように更新できなかった場合、リカバリー中にエラーメッセージを表示しませんでした。これは、レスキューイメージがバックアップと同期していない場合に発生する可能性があります。今回の更新により、復元された基本システムファイルが再作成されたシステムと一致しない場合、リカバリー中にエラーメッセージが表示されます。

([BZ#2072978](#))

**PXE 出力メソッドを使用する ReaR は、出力ファイルを rsync OUTPUT\_URL 内に出力ファイルを保存する際に失敗しなくなりました**

RHEL 8.5 では、**OUTPUT=PXE** および **BACKUP=RSYNC** オプションを使用した **OUTPUT\_URL** 変数の処理が削除されていました。その結果、**OUTPUT\_URL** に rsync の場所を使用すると、ReaR は **initrd** およびカーネルファイルをこの場所にコピーできず、それらは **BACKUP\_URL** で指定された場所にアップロードされました。今回の更新により、RHEL 8.4 以前のリリースの動作が復元されました。ReaR は、rsync を使用して、指定された **OUTPUT\_URL** の宛先に必要なファイルを作成します。

([BZ#2115918](#))

**ReaR は、NetBackup バージョン 9 を使用したシステムの復元をサポートするようになりました**

以前は、NetBackup バージョン 9 以降で NetBackup (NBU) メソッドを使用してシステムを復元すると、ライブラリーやその他のファイルが見つからないために失敗していました。今回の更新により、**NBU\_LD\_LIBRARY\_PATH** 変数に必要なライブラリーパスが含まれ、レスキューシステムに必要なファイルが組み込まれるようになり、ReaR は NetBackup メソッドを使用できるようになりました。

(BZ#2077404)

### ReaR では、シンボリックリンクターゲットの欠落に関する誤ったエラーメッセージが表示されなくなりました

以前のリリースでは、ReaR はレスキューイメージの作成時に、**/usr/lib/modules/** の **build** および **source** シンボリックリンクのターゲットがないことを示す誤ったエラーメッセージを表示していました。この状況は安全性に問題がないため、エラーメッセージを無視しても問題はありませんでした。今回の更新により、この状況で ReaR は、シンボリックリンクターゲットが欠落していることを示す誤ったエラーメッセージを報告しなくなりました。

(BZ#2021935)

### SR-IOV デバイスのフォールバックが正常に完了するようになりました

以前は、**hcnmgr** スクリプトが **primary** 属性ではなく誤った **active\_slave** 属性を使用していたため、Single Root I/O Virtualization (SR-IOV) デバイスはデバイスのフェイルオーバー後にフォールバックしませんでした。今回の更新により、**hcnmgr** スクリプトは正しい属性を使用し、SR-IOV デバイスのフォールバックが正常に完了します。

(BZ#2078514)

### ppc64-diag がバージョン 2.7.8 にリベースされました

プラットフォーム診断用の **ppc64-diag** パッケージがバージョン 2.7.8 に更新されました。主な改善点とバグ修正は次のとおりです。

- **libvpd** ユーティリティーバージョン 2.2.9 以降を使用するようにビルド依存関係を更新
- サポート対象外のプラットフォームでの **extract\_opal\_dump** エラーメッセージを修正
- **GCC-8.5** および **GCC-11** コンパイラーでのビルド警告を修正

(BZ#2051313)

### lsvpd がバージョン 1.7.14 にリベースされました

ハードウェアインベントリシステムを設定するためのコマンドを提供する **lsvpd** パッケージがバージョン 1.7.14 に更新されました。今回の更新により、**lsvpd** ユーティリティーは、**vpdupdate** コマンドの実行時にデータベースファイルが破損するのを防ぎます。

(BZ#2051316)

### libvpd がバージョン 2.2.9 にリベースされました

Vital Product Data (VPD) にアクセスするためのクラスを含む **libvpd** パッケージがバージョン 2.2.9 に更新されました。主な改善点とバグ修正は次のとおりです。

- データベースのロックを修正
- **libtool** ユーティリティーのバージョン情報を更新

(BZ#2051319)

## 8.4. インフラストラクチャーサービス

### RHEL 8 のプリンターテストページのレイアウトが変更されました

以前は、宛先ドキュメント形式が PDF の場合、印刷テストページが印刷されませんでした。この更新プログラムでは、より広範なプリンターセットで動作する新しいテストページレイアウトが導入されています。テストページには、プリンターまたはテストページの印刷ジョブに関する情報は含まれていないことに注意してください。

([BZ#2064606](#))

### frr バイナリーファイルとスクリプトの場所が新しくなりました

以前は、動的ルーティングスタックを管理するための **frr** パッケージには、バイナリーファイルとスクリプトが **/usr/lib/frr** ディレクトリーに含まれていたため、新しいターゲット SELinux ポリシーを適用するときに特定の問題が発生していました。その結果、SELinux は拒否メッセージをアクセスベクター キャッシュ (AVC) に記録し、**frr** が適切に起動するのを妨げていました。

今回の更新により、**/usr/libexec/frr** が **frr** バイナリーファイルとスクリプトの新しい場所になります。その結果、SELinux は **/usr/libexec/frr** にあるバイナリーとスクリプト、および **/usr/lib64/frr** にあるその他の **frr** ライブラリーに個別にルールを適用し、拒否メッセージを生成しなくなりました。

([BZ#1714984](#), [BZ#1941765](#))

## 8.5. セキュリティー

### OpenSCAP の修復により、**/etc/tmux.conf** の正しいアクセス許可が設定されます

以前は、SCAP ルール **configure\_tmux\_lock\_after\_time** を修正するときに、**umask (600)** に関する権限で **/etc/tmux.conf** ファイルが作成されました。これにより、通常のユーザーが **/etc/tmux.conf** を読み取れなくなりました。通常のユーザーがログインすると、エラーメッセージが表示され、タイムアウトが発生してログインするまで数分間待たなければなりませんでした。今回の更新では、**configure\_tmux\_lock\_after\_time** ルールの修正により、**/etc/tmux.conf** の特定の権限が **644** に設定されます。その結果、通常のユーザーはエラーメッセージやログインの遅延に遭遇しなくなりました。

([BZ#2064696](#))

### Rsyslog の SCAP ルールは **.conf** ファイルを正しく識別します

以前は、ルールシステムログファイルに正しいアクセス許可があることを確認する (**xccdf\_org.ssgproject.content\_rule\_rsyslog\_files\_permissions**) は、Rsyslog インクルードステートメントの glob 式を展開しませんでした。その結果、ルールは関連するすべての設定ファイルを解析しませんでした。また、一部のログファイルの権限がチェックされませんでした。今回の更新により、ルールは glob 式を正しく展開して、解析する必要がある **.conf** ファイルを識別します。その結果、ルールは必要な **.conf** ファイルを正しく処理し、設定されたすべてのログファイルに正しい権限が付与されるようになりました。

([BZ#2075384](#))

### chronyd のルールでは、明示的な **chrony** ユーザー設定は必要ありません

RHEL は、デフォルトで **chrony** ユーザーの下で **chronyd** を実行します。以前は、**chronyd** サービス設定ユーザーのチェックと修復が必要以上に厳格でした。厳しすぎるチェックは、誤検知と過剰な修復につながりました。このバージョンでは、ルール

**xccdf\_org.ssgproject.content\_rule\_chronyd\_run\_as\_chrony\_user** のチェックと修復が更新され、最小限の正しい設定と従来の明示的な正しい設定パスの両方がパスされます。その結果、ルールはデフォルトの RHEL の動作を尊重し、明示的な **chrony** ユーザー設定を必要としません。

(BZ#2077531)

### **rsyslog\_remote\_loghost** に追加された警告

SCAP ルール **xccdf\_org.ssgproject.content\_rule\_rsyslog\_remote\_loghost** により、Rsyslog デモーンがログメッセージをリモートログホストに送信するように設定されます。ただし、ルールは TCP キューを設定しません。その結果、TCP キューが設定されていないとシステムがハングし、リモートログホストが使用できなくなります。この更新により、TCP キューの設定方法を説明する警告メッセージが追加されました。このルールの使用中にシステムがハングした場合は、警告を読み、システムを適切に設定してください。

(BZ#2078974)

### **sudo\_custom\_logfile** の修復は、カスタム **sudo** ログファイルに対して機能します

以前は、SCAP セキュリティガイドルール **xccdf\_org.ssgproject.content\_sudo\_custom\_logfile** の修正が、**/var/log/sudo.log** とは異なるパスを持つカスタム **sudo** ログファイルに対して機能しませんでした。今回の更新でルールが修正され、予想されるパスと一致しないカスタム **sudo** ログファイルがシステムにある場合に適切に修復できるようになりました。

(BZ#2083109)

### **firewalld\_sshd\_port\_enabled** の修復が正しく機能するようになりました

以前は、SCAP ルール **xccdf\_org.ssgproject.content\_rule\_firewalld\_sshd\_port\_enabled** の Bash 修正で、ネットワークインターフェイスのリストが正しく処理されませんでした。さらに、設定ファイルの名前が必要とは異なるものでした。この更新により、修復が修正されました。その結果、修復によってすべてのネットワークインターフェイスが正しく処理され、設定ファイルには予測可能な名前が付けられます。

(BZ#2109602)

### **fagenrules --load** が正常に動作するようになりました

以前は、**fapolicyd** サービスはシグナルのハングアップ (SIGHUP) を正しく処理しませんでした。その結果、**fapolicyd** は SIGHUP シグナルを受信した後に終了し、**fagenrules --load** コマンドが正しく機能しませんでした。この更新では、その問題が修正されました。その結果、**fagenrules --load** が正常に機能し、ルール更新時に **fapolicyd** を手動で再起動する必要がなくなりました。

(BZ#2070639)

## 8.6. ネットワーキング

### **NetworkManager** ユーティリティーは、さまざまなソースからの IPv6 アドレスの正しい順序付けを強制します

通常、IPv6 アドレスの順序は、ソースアドレス選択の優先度に影響します。たとえば、発信 TCP 接続を確立する場合などです。以前は、**manual**、**dhcpv6**、および **autoconf6** メソッドで追加された IPv6 アドレスの相対的な優先度は、正しくありませんでした。今回の更新で問題が修正され、順序付けの優先順位が次のロジックを反映するようになりました: **manual** > **dhcpv6** > **autoconf6**。ただし、**ipv6.addresses** 設定でのアドレスの順序は変更されておらず、最後に追加されたアドレスが引き続き最高の優先度になります。

(BZ#2097270)

## 非対称ルーティングが正しく機能するようになりました

RHEL 8 の以前のマイナーバージョンには、場合によっては接続追跡が失敗する原因となる変更が含まれていました。その結果、非対称ルーティングが正しく機能していませんでした。このリリースは、RHEL 8.6 で導入された変更を元に戻します。その結果、非対称ルーティングは正しく機能します。

(BZ#2062870)

## 8.7. カーネル

### CgroupV1 memory.swappiness を非推奨にする新しい機能により、一貫したスワップ動作が可能になります

CgroupV1 には、指定された cgroup のスワップ動作を制御する cgroup ごとの **swappiness** 値の **memory.swappiness** が含まれています。

ただし、**systemd** プロセスは **cgroup** 内で実行され、**sysctl** の **swappiness** 値は **swap** ヒューリスティックにほとんど影響しません。このような cgroup は **sysctl** または **tuned** 設定の値を無視し、システムで実行されているプロセスにはデフォルトの **swappiness** 値 **60** が割り当てられます。その結果、メモリー負荷が高く、ページの再利用が行われる場合、割り当てられた **swappiness** 値よりも早い、またはより積極的なスワッピングが発生する可能性があります。

今回の更新では、新しい **sysctl** 変数 **/proc/sys/vm/force\_cgroupv2\_swappiness** が導入され、デフォルト値は **0** です。 **1** に設定すると、**memory.swappiness** 値は非推奨になり、すべての cgroup ごとの **swappiness** 値は、**/proc/sys/vm/swappiness** ファイル内のシステム全体の **swappiness** 値を反映します。その結果、cgroup のメモリースワッピング動作がより一貫したものになります。

(BZ#2084242)

### 暗号化されたデバイスのパスフレーズを入力した後に Anaconda が失敗しなくなりました

以前は、インストールの準備時に **kdump** が無効になっていて、ユーザーが暗号化されたディスクのパーティショニングを選択した場合、暗号化されたデバイスのパスフレーズを入力した後、Anaconda インストーラーがトレースバックで失敗していました。

今回の更新で問題が修正され、ユーザーは **kdump** を有効にして暗号化されたディスクパーティションを作成する必要がなくなりました。

(BZ#2086100)

### v1 モードの net\_prio または net\_cls コントローラーが正しく動作するようになりました

以前は、**cgroup-v2** 環境では、v1 モードで **net\_prio** または **net\_cls** コントローラーのいずれかを使用すると、ソケットデータの階層追跡が無効になっていました。その結果、ソケットデータトラッキングコントローラーの **cgroup-v2** 階層がアクティブではなく、**dmesg** コマンドが次のメッセージを報告しました。

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

この更新により、再起動後に **cgroup-v2** が正しくアクティブになります。

(BZ#2046396)

## 8.8. ブートローダー

## grubby が引数を将来のカーネルに渡すようになりました

新しいバージョンのカーネルをインストールすると、**grubby** ツールは、以前のカーネルバージョンからのカーネルコマンドライン引数を渡しませんでした。その結果、GRUB ブートローダーがユーザー設定を無視していました。今回の修正により、新しいカーネルバージョンのインストール後も、ユーザー設定が永続化されるようになりました。

(BZ#1978226)

## 8.9. 高可用性およびクラスター

### pcs が新しい Booth チケットの作成時に mode オプションを認識するようになりました

以前は、新しい Booth チケットを追加するときにユーザーが **mode** オプションを指定すると、**pcs** がエラー **invalid booth ticket option 'mode'** を報告していました。今回の修正により、Booth チケットの作成時に **mode** オプションを指定できるようになりました。

(BZ#1786964)

### pcs が stonith-watchdog-timeout の値を検証するようになりました

以前は、**stonith-watchdog-timeout** プロパティを SBD 設定と互換性のない値に設定できました。これにより、フェンスループが発生したり、アクションが終了していても、クラスターがフェンシングアクションが成功したとみなす可能性がありました。この修正により、不正な設定を防ぐために、**pcs** 設定時に **stonith-watchdog-property** の値を検証するようになりました。

(BZ#1954099)

## 8.10. 動的プログラミング言語、WEB サーバー、およびデータベースサーバー

### OQGraph プラグインが有効になっている場合、MariaDB 10.5 は存在しないテーブルの削除について警告するようになりました

以前は、**OQGraph** ストレージエンジンプラグインが **MariaDB 10.5** サーバーにロードされたときに、**MariaDB** は存在しないテーブルの削除について警告しませんでした。特に、ユーザーが **DROP TABLE** コマンドまたは **DROP TABLE IF EXISTS** SQL コマンドを使用して存在しないテーブルをドロップしようとするとき、**MariaDB** はエラーメッセージを返したり警告をログに記録したりしません。このバグは修正され、説明されているシナリオで警告が表示されるようになりました。

(BZ#1944653)

## 8.11. コンパイラーおよび開発ツール

### fork ハンドラーコールバックから pthread\_atfork または dclose を呼び出すときに、アプリケーションがデッドロックしなくなりました

以前は、**glibc** が内部ロックを取得している間に、アプリケーションが **pthread\_atfork** ハンドラーコールバックを呼び出していました。その結果、フォークハンドラーを登録したり、フォークハンドラーから **dclose** を呼び出したりすると、アプリケーションがデッドロックする可能性がありました。

fork ハンドラーの実行中に内部データ構造を保護するために、別の Synchronization メカニズムが使用されるようになりました。その結果、フォークハンドラーコールバックから **pthread\_atfork** または **dclose** を呼び出すときに、アプリケーションがデッドロックしなくなりました。

(BZ#1888660)

**ディレクトリーのみが想定されている場合、Makefile のワイルドカード関数はシンボリックリンクを返さなくなりました**

以前は、**glob()** で使用される **GLOB\_ONLYDIR** ヒントが、シンボリックリンクを特定の XFS ファイルシステムのディレクトリーとして誤って報告していました。**glob()** を使用する場合、**make** はヒントが実際にディレクトリーであることを確認せず、その結果、ディレクトリーのみが予期される場合に、Makefile のワイルドカード関数がシンボリックリンクを返しました。

バグは修正され、Makefile のワイルドカード関数は、ディレクトリーのみが必要な場合にシンボリックリンクを返さなくなりました。

(BZ#1982608)

**popen() がマルチスレッドプロセスをクラッシュさせなくなりました**

以前は、**popen()** の不具合により、マルチスレッドプロセスからインターフェイスを使用すると、アプリケーションがクラッシュしました。今回の更新でバグが修正され、マルチスレッドプロセスが **popen()** を使用してもクラッシュしなくなりました。

(BZ#2065588)

**一部の IBM 文字セットの 0xBC コードポイントのマッピングは U+00AF MACRON になりました**

以前は、**IBM256**、**IBM277**、**IBM278**、**IBM280**、**IBM284**、**IBM297**、および **IBM424** 文字セットは、**EBCDIC** コードポイント **0xBC** を Unicode 文字 **U+203E OVERLINE** としてエンコードしていました。その結果、**glibc** が提供する **iconv** プログラムを使用する場合、**0xBC** コードポイントを含む文字セットのテキストの変換は、**ISO-8859-1** などの非 Unicode 文字セットでは失敗しました。これは、**U+203E OVERLINE** 文字をエンコードできなかったためです。

今回の更新で、バグが修正されました。その結果、**IBM277**、**IBM278**、**IBM280**、**IBM284**、および **IBM297** 文字セットでの入力は、すべての場合に **ISO-8859-1** に変換できます。**IBM256** および **IBM424** 文字セットの場合、入力テキストに **0xBC** コードポイントが含まれ、それぞれの出力が **U+00AF MACRON** である場合に、変換が失敗しなくなりました。

(BZ#1961109)

**tempnam 関数は getrandom を使用して、生成されたファイル名のランダム性を高めるようになりました**

以前は、Red Hat Enterprise Linux 8.4 以降の **tempnam** 関数は、パスの選択に時間由来のランダム性を使用していました。その結果、**tempnam** 関数は、立て続けに繰り返し呼び出されると、可能なファイル名の完全なセットを生成しませんでした。このバグは、**getrandom** 関数を使用して生成されるファイル名のランダム性を高める新しい実装によって修正されました。その結果、**tempnam** 関数はより明確なファイル名を生成するようになりました。

(BZ#2089247)

**POWER9 向けに最適化された strncpy 関数が誤った結果を返さなくなりました**

以前は、POWER9 の **strncpy** 関数は、埋め込み用の NUL バイトのソースとして正しいレジスターを使用していませんでした。その結果、出力バッファーには、NUL パディングではなく、初期化されていないレジスタコンテンツが含まれていました。今回の更新で、**strncpy** 関数が修正され、出力バッファーの末尾が NUL バイトで正しくパディングされるようになりました。

(BZ#2091553)

## en\_US@ampm ロケールが locale -a によって正しくリストされるようになりました

以前は、**locale -a** コマンドの出力の **en\_US@ampm** のリストに欠陥がありました。その結果、**locale -a** によって出力された名前/エイリアスを使用してこのロケールを設定しようとすると、**setlocale** API が失敗しました。今回の更新により、**en\_US@ampm** が正しくリストされ、**locale -a** によって出力されるすべてのロケールで **setlocale** の呼び出しが成功するようになりました。

(BZ#2104907)

## イベントのユニットマスクがすべて papi\_xml\_event\_info 出力に含まれるようになりました

以前は、**papi\_xml\_event\_info** のイベントユニットマスク情報のテストが不完全でした。場合によっては、イベントのユニットマスクが **papi\_xml\_event\_info** 出力に含まれていませんでした。このバグは修正され、その結果、**papi\_xml\_event\_command** はイベントのすべてのユニットマスクを出力するようになりました。

(BZ#2037426)

## 8.12. IDENTITY MANAGEMENT

### デフォルトで /var/log/messages に記録されなくなったデバッグメッセージ

以前は、**ipa-dnskeysyncd** および **ipa-ods-exporter** デーモンがデフォルトですべてのデバッグメッセージを **/var/log/messages** に記録していたため、ログファイルが大幅に増大していました。必要に応じて、**/etc/ipa/dns.conf** ファイルで **debug=True** を設定することにより、デバッグログレベルを設定できるようになりました。詳細は、**default.conf(5)** の man ページを参照してください。

(BZ#2059396)

### ユーザーアカウントの保持

以前は、**ipa user-del --preserve user\_login** コマンドを実行してユーザーアカウントを保存すると、出力に **Deleted user "user\_login"** というメッセージが誤って返されていました。このメッセージは、ユーザーが削除され、期待どおりに保持されていないことを誤って示しています。今回の更新で、出力は **Preserved user "user\_login"** を返すようになりました。

(BZ#2022028)

### 4 GB を超える Kerberos データベースの転送

以前は、**kprop** サービスと **kproptd** コマンドは、Kerberos KDC データベースのサイズを格納するときには 32 ビット値を使用していました。その結果、データベースサイズが 4 GB を超えると、プライマリ Kerberos サーバーからレプリカサーバーへの Kerberos データベースダンプファイルの転送が失敗しました。

今回の更新で Kerberos が変更され、4 GB を超える KDC データベースを転送できるようになりました。

(BZ#2026462)

### LDAP グループのメンバーリスト内の読み取り不能オブジェクトの処理

この更新の前は、SSSD が LDAP グループのメンバーリスト内の読み取り不能オブジェクトを一貫して処理していなかったため、読み取り不能オブジェクトが原因でエラーが発生したり、特定の状況で読み取り不能オブジェクトが無視されたりしていました。

今回の更新により、SSSD には、この動作を変更するための新しいオプション

**ldap\_ignore\_unreadable\_references** が追加されました。**ldap\_ignore\_unreadable\_references** オプ

ションが **false** に設定されている場合には、読み取り不能オブジェクトによってエラーが発生し、**true** に設定されている場合、読み取り不能オブジェクトは無視されます。デフォルトは **false** に設定されており、元の一貫性のない動作のために、更新後に一部のグループルックアップが失敗する場合があります。この場合、`/etc/sss/sss.conf` ファイルの対応する `[domain/name of the domain]` セクションで `ldap_ignore_unreadable_references = True` と指定します。

これにより、読み取り不能オブジェクトを一貫した方法で処理でき、新しい `ldap_ignore_unreadable_references` オプションを使用して動作を調整できます。

(BZ#2069379)

## 8.13. デスクトップ

### 機内モードスイッチは常に表示されます

以前は、機内モードを有効にすると、**Settings** アプリケーションの **Wi-Fi** セクションにある **Airplane Mode** スイッチが消えていました。今回の更新で問題が修正され、状態に関係なく、**Settings** には常に **Airplane Mode** スイッチが表示されます。

(BZ#2079139)

## 8.14. グラフィックインフラストラクチャー

### Motif アプリケーションのホットキーが正しい項目をアクティブにする

以前は、Motif ツールキットを使用するアプリケーションで、メニューホットキーが間違ったメニュー項目をアクティブにしていました。サブメニューが開いているときに、その項目に関連付けられたホットキーを押すと、アプリケーションは代わりに親メニューの項目をアクティブにしました。

今回の更新で問題が修正され、ホットキーが正しいサブメニュー項目をアクティブにするようになりました。

(BZ#2060571)

### デスクトップが無効な IPv6 および `DisallowTCP=false` で起動に失敗しなくなりました

以前は、次の状況でログイン後に X11 デスクトップセッションを開始できませんでした。

- システムで IPv6 ネットワーキングが無効になっています。
- `DisallowTCP=false` オプションが GDM 設定で有効になっていました。

今回の更新で問題が修正され、説明されている設定で期待どおりに X11 セッションにログインできるようになりました。

(BZ#2075132)

## 8.15. WEB コンソール

### Web コンソールを使用した USB ホストデバイスの削除が期待どおりに機能するようになりました

以前は、USB デバイスを仮想マシン (VM) に接続すると、USB デバイスのデバイス番号とバス番号が仮想マシンに渡された後に変更されていました。その結果、Web コンソールを使用してこのようなデバイスを削除すると、デバイスとバス番号の相関が正しくないため失敗していました。今回の更新で問題が修正され、Web コンソールを使用して USB ホストデバイスを削除できるようになりました。

(JIRA:RHELPLAN-109067)

Web コンソールを使用した複数のホストデバイスの接続が想定通りに機能するようになりました

以前は、Web コンソールを使用して仮想マシンに接続する複数のデバイスを選択すると、1つのデバイスのみが接続され、残りは無視されていました。今回の更新で問題が修正され、Web コンソールを使用して複数のホストデバイスを同時に接続できるようになりました。

(JIRA:RHELPLAN-115603)

## 8.16. RED HAT ENTERPRISE LINUX システムロール

正しいボンディングモードの **active-backup** をサポートするようにタイプミスを修正する

以前は、**active-backup** ボンディングモードを指定する際に InfiniBand ポートをサポートする際に、タイプミス (**active\_backup**) がありました。このタイプミスが原因で、接続は InfiniBand ボンディングポートの正しいボンディングモードをサポートできませんでした。この更新では、ボンディングモードを **active-backup** に変更することで、タイプミスを修正しています。これで、接続は InfiniBand ボンディングポートを正常にサポートします。

(BZ#2064067)

**IPRouteUtils.get\_route\_tables\_mapping()** 関数が空白シーケンスを受け入れるようになりました。

以前は、**/etc/iproute2/rt\_tables** などの **iproute2** ルーティングテーブルデータベースのパarser は、**254 main** の形式でファイルにエントリーをアサートしていたため、数値 ID と名前は1つのスペースのみで区切られていました。その結果、parser はルートテーブル名とテーブル ID 間のすべてのマッピングをキャッシュできませんでした。これにより、ユーザーはルートテーブル名を定義して静的ルートをルートテーブルに追加できませんでした。今回の更新で、parser はテーブル ID とテーブル名の上に空白シーケンスを受け入れるようになりました。その結果、parser はルートテーブル名とテーブル ID 間のすべてのマッピングをキャッシュし、ユーザーはルートテーブル名を定義して静的ルートをルートテーブルに追加できます。

(BZ#2115884)

**metrics** による設定 RHEL システムロールはシンボリックリンクを正しくたどりま

**mssql pcp** パッケージがインストールされると、**mssql.conf** ファイルは **/etc/pcp/mssql/** に配置され、シンボリックリンク **/var/lib/pcp/pmdas/mssql/mssql.conf** のターゲットになります。ただし、以前の **metrics** ロールはシンボリックリンクをたどって、**mssql.conf** を設定する代わりにシンボリックリンクを上書きしていました。その結果、**metrics** ロールを実行すると、シンボリックリンクが通常のファイルに変更され、**/var/lib/pcp/pmdas/mssql/mssql.conf** ファイルのみ設定の影響を受けました。これによりシンボリックリンクが失敗し、メインの設定ファイル **/etc/pcp/mssql/mssql.conf** は設定の影響を受けませんでした。この問題は修正され、シンボリックリンクをたどる **follow: yes** オプションが **metrics** ロールに追加されました。その結果、**metrics** ロールはシンボリックリンクを保持し、メイン設定ファイルを正しく設定します。

(BZ#2060377)

**tlog** RHEL System Roles が SSSD によって正しくオーバーレイされるようになりました

以前は、**tlog** RHEL System Role は、システムセキュリティーサービスデーモン (SSSD) ファイルプロバイダーと有効な **authselect** オプション **with-files-domain** に依存して、**nsswitch.conf** ファイルに正しい **passwd** エントリーを設定していました。今回の修正により、**tlog** ロールは **nsswitch.conf** を更新して、**tlog-rec-session** が SSSD によって正しくオーバーレイされるようになりました。

(BZ#2072749)

### ボリュームの `mount_options` パラメーターがボリュームに対して有効になりました

以前は、ボリュームの有効なパラメーターの一覧からパラメーターが誤って削除されていました。そのため、ユーザーはボリュームに `mount_options` パラメーターを設定できませんでした。今回のバグ修正により、`mount_options` パラメーターが有効なパラメーターの一覧に戻され、エラーをキャッチするようにコードがリファクタリングされました。その結果、`storage` RHEL システムロールは、ボリュームの `mount_options` パラメーターを設定できます。

(BZ#2083378)

### `metrics` RHEL システムロールの README およびドキュメントでは、特定のバージョンの RHEL においてロールがサポートする Redis および Grafana バージョンが明確に指定されるようになりました。

以前のバージョンでは、サポートされていないバージョンの Redis および Grafana で `metrics` ロールを使用しようとすると、ロールは失敗していました。今回の更新により、ロールによって、サポートされているバージョンの Redis および Grafana についてのドキュメントが明確になりました。そのため、サポート対象外のバージョンの Redis および Grafana をサポート対象外のプラットフォームで使用することを回避できます。

(BZ#2100285)

### `kernel_settings` システムロールが `python3-configobj` を正しくインストールするようになりました

以前は、`kernel_settings` ロールは `python3-configobj` パッケージが見つからないというエラーを返していました。マネージドホストに `python3-configobj` がインストールされていないため、ロールはパッケージを見つけることができませんでした。今回の更新で、ロールはマネージドホストに `python3-configobj` をインストールし、正しく機能するようになりました。

(BZ#2060378)

### `storage` RHEL システムロールが、LVM ボリュームの `striped` および `raid0` レベルを正しくサポートするようになりました。

以前は、`storage` RHEL システムロールは、LVM ボリュームに対応していないため、RAID レベルの `striped` および `raid0` を誤って報告していました。これは修正され、ロールは LVM でサポートされているすべての RAID レベルの LVM ボリュームを正しく作成できるようになりました。これには、`raid0`、`raid1`、`raid4`、`raid5`、`raid6`、`raid10`、`striped`、および `mirror` が含まれます。

(BZ#2083426)

### `metrics` RHEL システムロールは、設定の更新後に `pmie` および `pmlogger` サービスを自動的に再起動します。

以前は、`pmie` および `pmlogger` サービスは、設定が変更され、ハンドラーの実行を待機した後、再起動しませんでした。これにより、他の `metrics` サービスでエラーが発生し、`pmie` と `pmlogger` の設定を実行時の動作と一致させる必要がありました。今回の更新により、ロールは設定の更新直後に `pmie` と `pmlogger` を再起動し、それらの設定は依存メトリックサービスの実行時の動作と一致し、正しく機能します。

(BZ#2100298)

### `forward_port` パラメーターが、`string` オプションと `dict` オプションの両方を受け入れるようになりました。

以前は、**firewall** RHEL システムロールで、**forward\_port** パラメーターは **string** オプションのみを受け入れていました。しかし、ロールのドキュメントでは、**string** および **dict** の両オプションがサポートされているとしていました。そのため、ドキュメントを読み込んだユーザーはエラーを経験していました。このバグは、**forward\_port** で両方のオプションを受け入れるようにすることで修正されました。その結果、ユーザーはドキュメントに従いポート転送を設定できるようになりました。

(BZ#2101607)

**追加の Dracut コマンドラインパラメーターを指定する際に、nbde\_client システムロールが適切な間隔を使用するように**

Dracut フレームワークでは、カーネルコマンドラインパラメーターなどの追加のパラメーターを指定する場合には、適切な間隔が必要です。パラメーターが適切な間隔で指定されていない場合、Dracut は指定された追加パラメーターをカーネルコマンドラインに追加しない場合があります。今回の更新で、**nbde\_client** システムロールは、アドオン Dracut 設定ファイルの作成時に適切な間隔を使用するようになりました。その結果、ロールは Dracut コマンドラインパラメーターを正しく設定します。

(BZ#2115161)

**ssh および sshd RHEL システムロールの最小 RSA 鍵ビット長オプションが追加されました**

誤って短い RSA 鍵を使用すると、システムが攻撃に対してより脆弱になる可能性があります。今回の更新で、**ssh** および **sshd** RHEL システムロールの **RSAMinSize** オプションを使用して、OpenSSH クライアントおよびサーバーに RSA 鍵の最小ビット長を設定できるようになりました。

(BZ#2109997)

**NBDE クライアントシステムのロールは静的 IP アドレスをサポートします**

以前のバージョンの RHEL では、静的 IP アドレスを使用してシステムを再起動し、ネットワークバウンドディスク暗号化 (NBDE) クライアントシステムロールを使用して設定すると、システムの IP アドレスが変更されていました。この変更により、NBDE クライアントシステムロールにより静的 IP アドレスを持つシステムがサポートされ、再起動後にその IP アドレスは変更されません。

デフォルトでは、NBDE ロールは起動時に DHCP を使用し、システムの起動時に設定済みの静的 IP に切り替えることに注意してください。

(BZ#2071011)

## 8.17. 仮想化

**フェイルオーバー VF を使用した仮想マシンのコピー前のライブ移行が正しく機能するようになりました**

以前のリリースでは、VM が仮想機能 (VF) フェイルオーバー機能が有効になっているデバイスを使用している場合に、実行中の仮想マシン (VM) のコピー前移行の試行に失敗しました。今回の更新で問題が修正され、説明したシナリオでの仮想マシンの移行が正しく機能するようになりました。

(BZ#2054656)

## 8.18. クラウド環境の RHEL

**インスタンスは、Alibaba Cloud で nm-cloud-setup サービスを開始した後もプライマリー IP アドレスを保持するようになりました。**

以前は、Alibaba Cloud でインスタンスを起動すると、nm-cloud-setup サービスは、複数の IPv4 アドレスがある場合、誤った IP アドレスをプライマリー IP アドレスとして設定していました。そのため、

これは発信接続の IPv4 ソースアドレスの選択に影響を与えていました。今回の更新で、セカンダリー IP アドレスを手動で設定すると、NetworkManager パッケージは primary-ip-address メタデータからプライマリー IP アドレスを取得し、プライマリー IP アドレスとセカンダリー IP アドレスの両方を正しく設定するようになりました。

(BZ#2082000)

### SR-IOV は、Azure 上の ARM 64 RHEL 8 仮想マシンで準最適に動作しなくなりました

これまで、SR-IOV ネットワーキングデバイスは、Microsoft Azure プラットフォームで実行されている ARM 64 RHEL 8 仮想マシンで想定されるよりも、全体でははるかに低く、レイテンシーは高くなっていました。この問題は修正され、影響を受ける仮想マシンが期待どおりに動作するようになりました。

(BZ#2068429)

### cloud-init を使用して AWS で RHEL 8 仮想マシンを起動しても、予想より時間がかからなくなった

以前のリリースでは、Amazon Web Services (AWS) で **cloud-init** サービスを使用して RHEL 8 の EC2 インスタンスを初期化するには、非常に長い時間がかかっていました。RHEL 8 の Amazon Machine Image (AMI) が更新され、問題の修正が含まれるようになり、RHEL 8 の EC2 インスタンスの初期化が正しく機能するようになりました。

ただし、独自の RHEL 8 イメージをカスタマイズしてアップロードするときに、初期化が依然として遅くなることがあります。この問題を回避するには、イメージを AWS にアップロードする前に、VM の作成に使用しているイメージから **/etc/resolv.conf** ファイルを削除します。

(BZ#1862930)

## 8.19. コンテナ

### リポジトリ ID が一致しないために DNF と YUM が失敗することがなくなりました

以前は、DNF および YUM リポジトリ ID は、DNF または YUM が期待する形式と一致しませんでした。たとえば、次の例を実行すると、エラーが発生しました。

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

今回の更新で、この問題が修正されています。接尾辞 **--debug-rpms** がすべてのデバグリポジトリ名に追加され (例: **ubi-8-appstream-debug-rpms**)、接尾辞 **-rpms** がすべての UBI リポジトリ名に追加されました (例: **ubi-8-appstream-rpms**)。

詳細については、[ユニバーサルベースイメージ \(UBI\): イメージ、リポジトリ、パッケージ、およびソースコード](#) を参照してください。

(BZ#2120378)

### ベータ版 GPG キーで署名されたコンテナイメージがプルできるように

以前は、RHEL ベータコンテナイメージをプルすると、Podman は次のエラーメッセージで失敗していました。**Error: Source image rejected: None of the signatures were accepted** (エラー: ソースイメージが拒否されました: 署名が受け入れられませんでした)。現在のビルドでは、RHEL ベータ版の GPG キーをデフォルトで信頼しないように設定されているため、イメージのプルに失敗してしま

た。今回の更新により、`/etc/containers/policy.json` ファイルは、信頼できるキーを含むファイルのリストを受け入れる新しい **keyPaths** フィールドをサポートします。このため、GA および Beta GPG キーで署名されたコンテナイメージがデフォルト設定で受け入れられるようになりました。

([BZ#2020301](#))

## 第9章 テクノロジープレビュー

ここでは、Red Hat Enterprise Linux 8.7 で利用可能なすべてのテクノロジープレビュー機能の一覧を提示します。

テクノロジープレビューに対する Red Hat のサポート範囲の詳細は、[テクノロジープレビューのサポート範囲](#) を参照してください。

### 9.1. シェルおよびコマンドラインツール

**ReaR は、64 ビット IBM Z アーキテクチャーでテクノロジープレビューとして利用できます。**

Basic Relax and Recover (ReaR) 機能が、64 ビットの IBM Z アーキテクチャーでテクノロジープレビューとして利用できるようになりました。IBM Z では、z/VM 環境でのみ ReaR レスキューイメージを作成できます。論理パーティション (LPAR) のバックアップおよび復元はテストされていません。

現在利用できる出力方法は、Initial Program Load (IPL) のみです。IPL は、**ziPL** ブートローダーで利用できるカーネルと初期 ramdisk (initrd) を生成します。



#### 警告

現在、レスキュープロセスは、システムに接続したすべての DASD (Direct Attached Storage Devices) を再フォーマットします。システムストレージデバイスに貴重なデータが存在する場合は、システムの復旧を行わないでください。これには、レスキュー環境で起動するのに使用された **ziPL** ブートローダー、ReaR カーネル、および initrd で準備されたデバイスも含まれます。必ずコピーを保管してください。

詳細は、[64 ビット IBM Z アーキテクチャーで ReaR レスキューイメージの使用](#) を参照してください。

(BZ#1868421)

### 9.2. ネットワーキング

**KTLS がテクノロジープレビューとして利用可能に**

RHEL は、テクノロジープレビューとして KTLS (Kernel Transport Layer Security) を提供します。KTLS は、AES-GCM 暗号化のカーネルで対称暗号化アルゴリズムまたは複号アルゴリズムを使用して TLS レコードを処理します。KTLS には、この機能を提供するネットワークインターフェイスコントローラー (NIC) に TLS レコード暗号化をオフロードするインターフェイスも含まれています。

(BZ#1570255)

**AF\_XDP がテクノロジープレビューとして利用可能に**

**AF\_XDP (Address Family eXpress Data Path)** ソケットは、高性能パケット処理用に設計されています。さらに処理するために、**XDP** を取り入れ、プログラムにより選択されたパケットの効率的なリダイレクトをユーザー空間アプリケーションに付与します。

(BZ#1633143)

## テクノロジープレビューとして利用できる XDP 機能

Red Hat は、以下の eXpress Data Path (XDP) 機能をサポート対象外のテクノロジープレビューとして提供します。

- AMD および Intel 64 ビット以外のアーキテクチャーで XDP プログラムを読み込む。 **libxdp** ライブラリーは、AMD および Intel 64 ビット以外のアーキテクチャーでは使用できません。
- XDP ハードウェアオフロード。

([BZ#1889737](#))

## TC のマルチプロトコルラベルスイッチがテクノロジープレビューとして利用可能に

Multi-protocol Label Switching (MPLS) は、エンタープライズネットワーク全体でトラフィックフローをルーティングするカーネル内データ転送メカニズムです。MPLS ネットワークでは、パケットを受信するルーターは、パケットに割り当てられたラベルに基づいて、パケットの追加のルートを決めます。ラベルを使用すると、MPLS ネットワークは特定の特性を持つパケットを処理する機能があります。たとえば、特定ポートから受信したパケットの管理や、特定のタイプのトラフィックを一貫した方法で伝送する **tc filters** を追加できます。

パケットがエンタープライズネットワークに入ると、MPLS ルーターは、パケット上で複数の操作を実行します。ラベルの追加には **push**、**swap** (ラベルの更新)、ラベルの削除の **pop** などが含まれます。MPLS では、RHEL の 1 つまたは複数のラベルに基づいて、アクションをローカルに定義できます。ルーターを設定し、トラフィック制御 (**tc**) フィルターを設定して、**label**、**traffic class**、**bottom of stack**、**time to live** などの MPLS ラベルスタックエントリー (**lse**) 要素に基づいて、パケットに対して適切なアクションを実行するように設定することができます。

たとえば、次のコマンドは、フィルターを **enp0s1** ネットワークインターフェイスに追加して、最初のラベル **12323** と 2 番目のラベル **45832** を持つ着信パケットと一致させます。一致するパケットでは、以下のアクションが実行されます。

- 最初の MPLS TTL はデクリメントされます (TTL が 0 に達するとパケットがドロップされません)。
- 最初の MPLS ラベルが **549386** に変更
- 作成されるパケットは **enp0s2** 経由で送信されます。宛先 MAC アドレス **00:00:5E:00:53:01**、およびソース MAC アドレス **00:00:5E:00:53:02**。

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

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

## systemd-resolved サービスがテクノロジープレビューとして利用できるようになりました。

**systemd-resolved** サービスは、ローカルアプリケーションに名前解決を提供します。このサービスは、DNS スタブリゾルバー、LLMNR (Link-Local Multicast Name Resolution)、およびマルチキャスト DNS リゾルバーとレスポンスのキャッシュと検証を実装します。

**systemd** パッケージが **systemd-resolved** を提供している場合でも、このサービスはサポートされていないテクノロジープレビューであることに注意してください。

(BZ#1906489)

### 9.3. カーネル

**kexec fast reboot 機能は、テクノロジープレビューとしてご利用いただけます。**

**kexec fast reboot 機能は、引き続きテクノロジープレビューとして利用できます。** **kexec** の高速再起動では、カーネルが最初に Basic Input/Output System (BIOS) を経由せずに、2 番目のカーネルを直接起動できるため、起動プロセスが大幅に高速化されます。この機能を使用するには、以下を実行します。

1. **kexec** カーネルを手動で読み込みます。
2. オペレーティングシステムを再起動します。

(BZ#1769727)

**accel-config パッケージがテクノロジープレビューとして利用可能になりました。**

**accel-config** パッケージが、テクノロジープレビューとして、Intel **EM64T** および **AMD64** アーキテクチャーで利用可能になりました。このパッケージは、Linux カーネルでデータストリーミング (DSA) サブシステムを制御し、設定するのに役立ちます。また、**sysfs** (pseudo-filesystem) を介してデバイスを設定し、設定を **json** 形式で保存および読み込みます。

(BZ#1843266)

**SGX がテクノロジープレビューとして利用可能になりました。**

**Software Guard Extensions (SGX)** は、ソフトウェアコードおよび公開および修正からのデータを保護する Intel® テクノロジーです。RHEL カーネルは、SGX v1 および v1.5 の機能を部分的に提供します。バージョン 1 では、**Flexible Launch Control** メカニズムを使用するプラットフォームが SGX テクノロジーを使用できるようにします。

(BZ#1660337)

**eBPF がテクノロジープレビューとして利用可能になりました。**

**eBPF (extended Berkeley Packet Filter)** は、限られた一連の関数にアクセスできる制限付きサンドボックス環境において、カーネル領域でのコード実行を可能にするカーネル内の仮想マシンです。

仮想マシンには、さまざまな種類のマップの作成を可能にする、新しいシステムコール **bpf()** が含まれ、特別なアセンブリーのコードでプログラムをロードすることも可能です。そして、このコードはカーネルにロードされ、実行時コンパイラーでネイティブマシンコードに変換されます。**bpf()** は、root ユーザーなど、**CAP\_SYS\_ADMIN** が付与されているユーザーのみが利用できます。詳細は、man ページの **bpf(2)** を参照してください。

ロードしたプログラムは、データを受信して処理するために、さまざまなポイント (ソケット、トレースポイント、パケット受信) に割り当てることができます。

**eBPF** 仮想マシンを使用する Red Hat には、多くのコンポーネントが同梱されています。各コンポーネントは異なる開発フェーズにあります。特定のコンポーネントがサポート対象と示されていない限り、すべてのコンポーネントはテクノロジープレビューとして提供されます。

現在、以下の主要 **eBPF** コンポーネントが、テクノロジープレビューとして利用可能です。

- **AF\_XDP**。これは、eXpress Data Path (XDP)パスを、パケット処理のパフォーマンスを優先するアプリケーションのユーザー空間に接続するためのソケットです。

(BZ#1559616)

### カーネルの Intel データストリーミングタブレットドライバーがテクノロジープレビューとして利用可能になる

カーネルの Intel データストリーミングアクセラレータードライバー (IDX) は、現在テクノロジープレビューとして利用できます。これは Intel CPU 統合アクセラレーターで、プロセスアドレス空間 ID (pasid) の送信および共有仮想メモリー (SVM) の共有ワークキューが含まれます。

(BZ#1837187)

### テクノロジープレビューとして利用できる Soft-RoCE

Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) は、RDMA over Ethernet を実装するネットワークプロトコルです。Soft-RoCE は、RoCE v1 および RoCE v2 の 2 つのプロトコルバージョンを維持する RoCE のソフトウェア実装です。Soft-RoCE ドライバーの **rdma\_rxe** は、RHEL 8 ではサポートされていないテクノロジープレビューとして利用できます。

(BZ#1605216)

### stmmac ドライバーがテクノロジープレビューとして利用可能に

Red Hat は、チップ (SoC) での Intel® Elkhart Lake システムの **stmmac** の使用をサポート対象外のテクノロジープレビューとして提供します。

(BZ#1905243)

## 9.4. ファイルシステムおよびストレージ

### ファイルシステム DAX が、テクノロジープレビューとして ext4 および XFS で利用可能に

Red Hat Enterprise Linux 8 では、ファイルシステムの DAX がテクノロジープレビューとして利用できます。DAX は、永続メモリーをそのアドレス空間に直接マッピングする手段をアプリケーションに提供します。DAX を使用するには、システムで利用可能な永続メモリーの形式が必要になります。通常は、NVDIMM (Non-Volatile Dual In-line Memory Module) の形式で、DAX 機能を提供するファイルシステムを NVDIMM に作成する必要があります。また、ファイルシステムは **dax** マウントオプションでマウントする必要があります。これにより、**dax** をマウントしたファイルシステムのファイルの **mmap** が、アプリケーションのアドレス空間にストレージを直接マッピングされます。

(BZ#1627455)

### OverlayFS

OverlayFS は、ユニオンファイルシステムのタイプです。これにより、あるファイルシステムを別のファイルシステムに重ねることができます。変更は上位のファイルシステムに記録され、下位のファイルシステムは変更しません。これにより、ベースイメージが読み取り専用メディアにあるコンテナや DVD-ROM などのファイルシステムイメージを、複数のユーザーが共有できるようになります。

OverlayFS は、ほとんどの状況で引き続きテクノロジープレビューになります。したがって、カーネルは、この技術がアクティブになると警告を記録します。

以下の制限下で、対応しているコンテナエンジン (**podman**、**cri-o**、または **buildah**) とともに使用すると、OverlayFS に完全対応となります。

- OverlayFS は、コンテナエンジングラフィックドライバーとしての使用、または圧縮された **kdump**

initramfs などのその他の特殊なユースケースとしての使用のみサポートされています。その使用は主にコンテナ COW コンテンツでサポートされており、永続ストレージではサポートされていません。非 OverlayFS ボリュームに永続ストレージを配置する必要があります。デフォルトのコンテナエンジン設定のみを使用できます。つまり、あるレベルのオーバーレイ、1つの下位ディレクトリー、および下位と上位の両方のレベルが同じファイルシステムにあります。

- 下層ファイルシステムとして使用に対応しているのは現在 XFS のみです。

また、OverlayFS の使用には、以下のルールと制限が適用されます。

- OverlayFS カーネル ABI とユーザー空間の動作については安定しているとみなされていないため、今後の更新で変更が加えられる可能性があります。
- OverlayFS は、POSIX 標準の制限セットを提供します。OverlayFS を使用してアプリケーションをデプロイする前に、アプリケーションを十分にテストしてください。以下のケースは、POSIX に準拠していません。
  - **O\_RDONLY** で開いているファイルが少ない場合は、ファイルの読み取り時に **st\_atime** の更新を受け取りません。
  - **O\_RDONLY** で開いてから、**MAP\_SHARED** でマッピングした下位ファイルは、後続の変更と一貫性がありません。
  - 完全に準拠した **st\_ino** 値または **d\_ino** 値は、RHEL 8 ではデフォルトで有効になっていませんが、モジュールオプションまたはマウントオプションを使用して、この値の完全な POSIX コンプライアンスを有効にできます。一貫した inode 番号を付けるには、**xino=on** マウントオプションを使用します。

**redirect\_dir=on** オプションおよび **index=on** オプションを使用して、POSIX コンプライアンスを向上させることもできます。この2つのオプションにより、上位レイヤーの形式は、このオプションなしでオーバーレイと互換性がありません。つまり、**redirect\_dir=on** または **index=on** でオーバーレイを作成し、オーバーレイをアンマウントしてから、このオプションなしでオーバーレイをマウントすると、予期しない結果またはエラーが発生することがあります。

- 既存の XFS ファイルシステムがオーバーレイとして使用できるかどうかを確認するには、次のコマンドを実行して、**ftype=1** オプションが有効になっているかどうかを確認します。

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

- SELinux セキュリティーラベルは、OverlayFS で対応するすべてのコンテナエンジンでデフォルトで有効になっています。
- このリリースの既知の問題は、OverlayFS に関連しています。詳細は [Linux カーネルドキュメントの Non-standard behavior](#) を参照してください。

OverlayFS の詳細は、[Linux カーネルのドキュメント](#) を参照してください。

(BZ#1690207)

## Straits がテクノロジーレビューとして利用可能になりました。

Stratis は、新しいローカルストレージマネージャーです。ユーザーへの追加機能を備えたストレージプールに、管理されるファイルシステムを提供します。

Stratis を使用すると、次のようなストレージタスクをより簡単に実行できます。

- スナップショットおよびシンプロビジョニングを管理する
- 必要に応じてファイルシステムのサイズを自動的に大きくする
- ファイルシステムを維持する

Stratis ストレージを管理するには、バックグラウンドサービス **stratisd** と通信する **stratis** ユーティリティを使用します。

Stratis はテクノロジープレビューとして提供されます。

詳細については、Stratis のドキュメント ([Stratis ファイルシステムの設定](#)) を参照してください。

RHEL 8.3 は Stratis をバージョン 2.1.0 に更新した。詳細は、[Stratis 2.1.0 リリースノート](#) を参照してください。

(JIRA:RHELPLAN-1212)

## テクノロジープレビューとして、IdM ドメインメンバーで Samba サーバーを設定できるようになりました。

今回の更新で、Identity Management (IdM) ドメインメンバーに Samba サーバーを設定できるようになりました。同じ名前パッケージに含まれる新しい **ipa-client-samba** ユーティリティは、Samba 固有の Kerberos サービスプリンシパルを IdM に追加し、IdM クライアントを準備します。たとえば、ユーティリティは、**sss** ID マッピングバックエンドの ID マッピング設定で `/etc/samba/smb.conf` を作成します。その結果、管理者が IdM ドメインメンバーに Samba を設定できるようになりました。

IdM 信頼コントローラーが Global Catalog Service をサポートしないため、AD が登録した Windows ホストは Windows で IdM ユーザーおよびグループを見つけることができません。さらに、IdM 信頼コントローラーは、Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) プロトコルを使用する IdM グループの解決をサポートしません。これにより、AD ユーザーは、IdM クライアントから Samba の共有およびプリンターにしかアクセスできません。

詳細は、[IdM ドメインメンバーでの Samba の設定](#) を参照してください。

(JIRA:RHELPLAN-13195)

## NVMe/TCP ホストはテクノロジープレビューとして利用可能です

TCP/IP ネットワーク (NVMe/TCP) および対応する **nvme-tcp.ko** カーネルモジュールへのアクセスおよび共有がテクノロジープレビューとして追加されました。ホストとしての NVMe/TCP の使用は、**nvme-cli** パッケージによって提供されるツールを使用して管理できます。NVMe/TCP ホストテクノロジープレビュー機能はテスト目的としてのみ同梱されており、現時点ではフルサポートの予定はありません。

(BZ#1696451)

## 9.5. 高可用性およびクラスター

### Pacemaker の podman バンドルがテクノロジープレビューとして利用可能になりました。

Pacemaker コンテナバンドルは、テクノロジープレビューとして利用できるコンテナバンドル機能を使用して、Podman で動作するようになりました。この機能はテクノロジープレビューとして利用できますが、例外が1つあります。Red Hat は、Red Hat Openstack 用の Pacemaker バンドルの使用に完全対応します。

(BZ#1619620)

## テクノロジープレビューとして利用可能な `corosync-qdevice` のヒューリスティック

ヒューリスティックは、起動、クラスターメンバーシップの変更、`corosync-qnetd` への正常な接続でローカルに実行され、任意で定期的に行われる一連のコマンドです。すべてのコマンドが時間どおりに正常に終了すると (返されるエラーコードがゼロである場合)、ヒューリスティックは渡されますが、それ以外の場合は失敗します。ヒューリスティックの結果は `corosync-qnetd` に送信され、クォーラムとなるべきパーティションを判断するための計算に使用されます。

(BZ#1784200)

## 新しい `fence-agents-heuristics-ping` フェンスエージェント

Pacemaker は、テクノロジープレビューとして `fence_heuristics_ping` エージェントを提供するようになりました。このエージェントの目的は、実際にはフェンシングを行わず、フェンシングレベルの動作を新しい方法で活用する実験的なフェンスエージェントのクラスを開くことです。

ヒューリスティックエージェントが、実際のフェンシングを行うフェンスエージェントと同じフェンシングレベルで設定されていて、そのエージェントよりも順番が前に設定されているとします。その場合、フェンシングを行うエージェントで `off` 操作を行う前に、ヒューリスティックエージェントで、この操作を行います。このヒューリスティックエージェントが `off` アクションに対して失敗する場合、このフェンシングレベルが成功しないのはすでに明らかです。そのため、Pacemaker フェンシングは、フェンシングを行うエージェントで `off` 操作を行うステップをスキップします。ヒューリスティックエージェントはこの動作を利用して、特定の条件下で、実際のフェンシングを行うエージェントがフェンシングできないようにできます。

サービスを適切に引き継ぐことができないことを事前に把握できる場合は、ノードがピアをフェンシングする意味がないのであれば、ユーザーは特に 2 ノードクラスターでこのエージェントを使用できます。たとえば、ネットワークアップリンクに到達してサービスがクライアントに到達できない場合は、ノードがサービスを引き継ぐ意味はありません。これは、ルーターへの ping が検出できる状況が考えられます。

(BZ#1775847)

## リソース移動後の場所の制約の自動削除がテクノロジープレビューとして利用可能

`pcs resource move` コマンドを実行すると、現在実行しているノードでリソースが実行されないように、制約がリソースに追加されます。`pcs resource move` コマンドの `--autodelete` オプションが、テクノロジープレビューとして利用可能になりました。このオプションを指定すると、リソースを移動すると、コマンドが作成する場所の制約が自動的に削除されます。

(BZ#1847102)

## 9.6. IDENTITY MANAGEMENT

Identity Management JSON-RPC API がテクノロジープレビューとして利用可能になりました。

Identity Management (IdM) では API が利用できます。API を表示するために、IdM は、テクノロジープレビューとして API ブラウザーも提供します。

以前では、複数のバージョンの API コマンドを有効にするために、IdM API が拡張されました。これらの機能拡張により、互換性のない方法でコマンドの動作が変更することがありました。IdM API を変更しても、既存のツールおよびスクリプトを引き続き使用できるようになりました。これにより、以下が可能になります。

- 管理者は、管理しているクライアント以外のサーバーで、IdM の以前のバージョンもしくは最近のバージョンを使用できます。

- サーバーで IdM のバージョンを変更しても、開発者は特定バージョンの IdM コールを使用できます。

すべてのケースでサーバーとの通信が可能になります。たとえば、ある機能向けの新オプションが新しいバージョンに追加されていて、通信の一方の側でこれを使用していたとしても、特に問題はありませ

API の使用法は [Identity Management API を使用して IdM サーバーに接続する \(テクノロジープレビュー\)](#) を参照してください。

(BZ#1664719)

## DNSSEC が IdM でテクノロジープレビューとして利用可能

統合 DNS のある Identity Management (IdM) サーバーは、DNS プロトコルのセキュリティーを強化する DNS に対する拡張セットである DNS Security Extensions (DNSSEC) を実装するようになりました。IdM サーバーでホストされる DNS ゾーンは、DNSSEC を使用して自動的に署名できます。暗号鍵は、自動的に生成およびローテートされます。

DNSSEC で DNS ゾーンを保護する場合は、以下のドキュメントを参照することが推奨されます。

- [DNSSEC Operational Practices, Version 2](#)
- [Secure Domain Name System \(DNS\) Deployment Guide](#)
- [DNSSEC Key Rollover Timing Considerations](#)

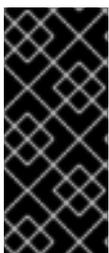
統合 DNS のある IdM サーバーは、DNSSEC を使用して、他の DNS サーバーから取得した DNS 回答を検証することに注意してください。これが、推奨される命名方法に従って設定されていない DNS ゾーンの可用性に影響を与える可能性があります。

(BZ#1664718)

## ACME はテクノロジープレビューとしてご利用いただけます。

Automated Certificate Management Environment (ACME) サービスが、テクノロジープレビューとして Identity Management (IdM) で利用可能になりました。ACME は、自動化識別子の検証および証明書の発行に使用するプロトコルです。この目的は、証明書の有効期間を短縮し、証明書のライフサイクル管理での手動プロセスを回避することにより、セキュリティーを向上させることです。

RHEL では、ACME サービスは Red Hat Certificate System (RHCS) PKI ACME レスポンダーを使用します。RHCS ACME サブシステムは、IdM デプロイメントのすべての認証局 (CA) サーバーに自動的にデプロイされますが、管理者が有効にするまでリクエストに対応しません。RHCS は、ACME 証明書を発行する際に `acmeIPAServerCert` プロファイルを使用します。発行された証明書の有効期間は 90 日です。ACME サービスの有効化または無効化は、IdM デプロイメント全体に影響します。



### 重要

ACME は、すべてのサーバーが RHEL 8.4 以降を実行している IdM デプロイメントでのみ有効にすることが推奨されます。以前の RHEL バージョンには ACME サービスが含まれていないため、バージョンが混在するデプロイメントで問題が発生する可能性があります。たとえば、ACME のない CA サーバーは、異なる DNS サブジェクト代替名 (SAN) を使用しているため、クライアント接続が失敗する可能性があります。



### 警告

現在、RHCS は期限切れの証明書を削除しません。ACME 証明書は 90 日後に期限切れになるため、期限切れの証明書が蓄積され、パフォーマンスに影響を及ぼす可能性があります。

- IdM デプロイメント全体で ACME を有効にするには、**ipa-acme-manage enable** コマンドを使用します。

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

- IdM デプロイメント全体で ACME を無効にするには、**ipa-acme-manage disable** コマンドを使用します。

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

- ACME サービスがインストールされ、有効または無効であるかを確認するには、**ipa-acme-manage status** コマンドを使用します。

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

(BZ#1628987)

## RHEL IdM では、ユーザー認証をテクノロジープレビューとして外部 ID プロバイダーに委任できる

RHEL IdM では、OAuth 2 デバイス認証フローをサポートする外部 ID プロバイダー (IdP) にユーザーを関連付けることができるようになりました。これらのユーザーは、RHEL 8.7 で利用可能な SSSD バージョンで認証すると、外部 IdP で認証と承認を実行した後、Kerberos チケットを使用して RHEL IdMSingle Sign-On 機能を受け取ります。

主な変更には以下のものがあります。

- **ipa idp-\*** コマンドによる外部 IdP への参照の追加、変更、および削除
- **ipa user-mod --user-auth-type=idp** コマンドを使用したユーザーの IdP 認証の有効化

追加情報は、[外部 ID プロバイダーを使用した IdM への認証](#) を参照してください。

(BZ#2101770)

## sssd-idp サブパッケージがテクノロジープレビューとして利用可能

SSSD の **sssd-idp** サブパッケージには、Identity Management (IdM) サーバーに対して OAuth2 認証を実行するクライアント側のコンポーネントである **oidc\_child** プラグインおよび **krb5 idp** プラグインが含まれます。この機能は、RHEL 8.7 以降、および RHEL 9.1 以降の IdM サーバーでのみ使用できます。

([BZ#2065692](#))

## SSSD の内部 krb5 idp プラグインがテクノロジープレビューとして利用可能

SSSD krb5 **idp** プラグインを使用すると、OAuth2 プロトコルを使用して外部アイデンティティプロバイダー (IdP) に対して認証できます。この機能は、RHEL 8.7 以降、および RHEL 9.1 以降の IdM サーバーでのみ使用できます。

([BZ#2056483](#))

## 9.7. デスクトップ

### 64 ビット ARM アーキテクチャーの GNOME がテクノロジープレビューとして利用できるようになりました。

GNOME デスクトップ環境がテクノロジープレビューとして、64 ビット ARM アーキテクチャーで利用可能になりました。これにより、管理者は VNC セッションを使用して、グラフィカルユーザーインターフェイス (GUI) からサーバーをリモートで設定し、管理できます。

そのため、64 ビット ARM アーキテクチャーで新しい管理アプリケーションが利用できるようになりました。たとえば、Disk Usage Analyzer (**baobab**)、Firewall Configuration (**firewall-config**)、Red Hat Subscription Manager (**subscription-manager-cockpit**)、または Firefox Web ブラウザーなどです。Firefox を使用すると、管理者はローカルの Cockpit デーモンにリモートで接続できます。

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

### IBM Z の GNOME デスクトップがテクノロジープレビューとして利用可能に

Firefox Web ブラウザーを含む GNOME デスクトップが、IBM Z アーキテクチャーでテクノロジープレビューとして利用できるようになりました。VNC を使用して GNOME を実行するリモートグラフィカルセッションに接続し、IBM Z サーバーを設定および管理できるようになりました。

([JIRA:RHELPLAN-27737](#))

## 9.8. グラフィックインフラストラクチャー

### 64 ビット ARM アーキテクチャーで VNC リモートコンソールがテクノロジープレビューとして利用可能に

64 ビットの ARM アーキテクチャーでは、Virtual Network Computing (VNC) リモートコンソールがテクノロジープレビューとして利用できます。グラフィックススタックの残りの部分は、現在、64 ビット ARM アーキテクチャーでは検証されていません。

([BZ#1698565](#))

## 9.9. 仮想化

### KVM 仮想マシンの AMD SEV および SEV-ES

テクノロジープレビューとして、RHEL 8 に、KVM ハイパーバイザーを使用する AMD EPYC ホストマシン用のセキュア暗号化仮想化 (SEV) 機能が同梱されます。仮想マシンで有効になっている場合は、SEV が仮想マシンのメモリーを暗号化して、ホストから仮想マシンへのアクセスを防ぎます。これにより、仮想マシンのセキュリティが向上します。

さらに、強化された SEV (Encrypted State) バージョンの SEV (SEV-ES) もテクノロジープレビューとして提供されます。SEV-ES は、仮想マシンの実行が停止すると、すべての CPU レジスターの内容を

暗号化します。これにより、ホストが仮想マシンの CPU レジスターを変更したり、そこから情報を読み取ったりできなくなります。

SEV および SEV-ES は、第 2 世代の AMD EPYC CPU (コードネーム Rome) 以降でのみ機能することに注意してください。また、RHEL 8 には SEV および SEV-ES の暗号化が含まれますが、SEV および SEV-ES のセキュリティー証明は含まれません。

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

## Intel vGPU

テクノロジープレビューとして、物理 Intel GPU デバイスを、**仲介デバイス** と呼ばれる複数の仮想デバイスに分割できるようになりました。この仲介デバイスは、仮想 GPU として複数の仮想マシンに割り当てることができます。これにより、この仮想マシンが、1つの物理 Intel GPU のパフォーマンスを共有します。

選択した Intel GPU のみが vGPU 機能と互換性があることに注意してください。

さらに、Intel vGPU が操作する VNC コンソールを有効にすることもできます。これを有効にすると、ユーザーは仮想マシンの VNC コンソールに接続し、Intel vGPU がホストする仮想マシンのデスクトップを確認できます。ただし、これは現在 RHEL ゲストオペレーティングシステムでのみ動作します。

(BZ#1528684)

## 入れ子仮想マシンの作成

入れ子 KVM 仮想化は、RHEL 8 で Intel、AMD64、IBM POWER および IBM Z システムホストで実行している KVM 仮想マシン用のテクノロジープレビューとして提供されます。この機能を使用すると、物理 RHEL 8 ホストで実行中の RHEL 7 または RHEL 8 仮想マシンがハイパーバイザーとして機能し、独自の仮想マシンをホストできます。

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

## テクノロジープレビュー:一部の Intel ネットワークアダプターが、Hyper-V の RHEL ゲストに SR-IOV を提供するようになりました

テクノロジープレビューとして、Hyper-V ハイパーバイザーで実行している Red Hat Enterprise Linux のゲストオペレーティングシステムは、**ixgbevf** および **ixgbevf** ドライバーがサポートする Intel ネットワークアダプターに、シングルルート I/O 仮想化 (SR-IOV) 機能を使用することができるようになりました。この機能は、以下の条件が満たされると有効になります。

- ネットワークインターフェイスコントローラー (NIC) に対して SR-IOV サポートが有効になっている
- 仮想 NIC の SR-IOV サポートが有効になっている
- 仮想スイッチの SR-IOV サポートが有効になっている
- NIC からの VF (Virtual Function) が仮想マシンに割り当てられている

この機能は現在、Microsoft Windows Server 2016 以降で提供されています。

(BZ#1348508)

## virtiofs を使用したホストと仮想マシン間でのファイルの共有

RHEL 8 では、テクノロジープレビューとして virtio ファイルシステム (**virtiofs**) が追加されました。**virtiofs** を使用すると、ホストシステムと仮想マシン (VM) との間で、ファイルを効率的に共有できます。

(BZ#1741615)

## RHEL 8 Hyper-V 仮想マシンで KVM 仮想化が利用可能に

ネストされた KVM 仮想化は、テクノロジープレビューとして、Microsoft Hyper-V ハイパーバイザーで使用できるようになりました。これにより、Hyper-V ホストで実行している RHEL 8 ゲストシステムで仮想マシンを作成できます。

この機能は、現在 Intel および AMD システムでのみ有効です。また、ネストされた仮想化は、Hyper-V でデフォルトで有効になっていない場合があります。これを有効にするには、以下の Microsoft ドキュメントを参照してください。

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

(BZ#1519039)

## 9.10. クラウド環境の RHEL

### RHEL Confidential VMs がテクノロジープレビューとして Azure で利用可能になりました

更新された RHEL カーネルを使用すると、Microsoft Azure で機密仮想マシン (VM) をテクノロジープレビューとして作成して実行できるようになりました。ただし、Azure での起動中に RHEL 機密 VM イメージを暗号化することはまだできません。

(JIRA:RHELPLAN-122316)

## 9.11. コンテナ

### Toolbox がテクノロジープレビューとして利用可能になりました。

以前は、Toolbox ユーティリティーは RHEL CoreOS [coreos/toolbox](#) に基づいていました。このリリースでは、Toolbox は [containers/toolbox](#) に置き換えられました。

(JIRA:RHELPLAN-77238)

### sigstore 署名がテクノロジープレビューとして利用可能になりました

Podman 4.2 以降では、コンテナイメージ署名の sigstore 形式を使用できます。sigstore 署名はコンテナイメージと共にコンテナレジストリーに格納されるため、イメージ署名を格納するために別の署名サーバーを用意する必要はありません。

(JIRA:RHELPLAN-75165)

イメージに署名するための複数の信頼できる GPG キーの機能は、テクノロジープレビューとして利用できます。

`/etc/containers/policy.json` ファイルは、信頼できるキーを含むファイルのリストを受け入れる新しい `keyPaths` フィールドをサポートします。このため、GA および Beta GPG キーで署名されたコンテナイメージがデフォルト設定で受け入れられるようになりました。

以下に例を示します。

```
"registry.redhat.io": [
  {
    "type": "signedBy",
    "keyType": "GPGKeys",
    "keyPaths": ["/etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release", "/etc/pki/rpm-gpg/RPM-GPG-
```

```
KEY-redhat-beta"]
    }
]
```

(JIRA:RHELPLAN-118470)

**podman-machine** コマンドはサポート対象外です。

仮想マシンを管理するための **podman-machine** コマンドは、テクノロジープレビューとしてのみ利用可能です。代わりに、コマンドラインから直接 Podman を実行してください。

(JIRA:RHELDOCS-16861)

## 第10章 非推奨になった機能

ここでは、Red Hat Enterprise Linux 8 で **非推奨** となった機能の概要を説明します。

非推奨の機能は、本製品の今後のメジャーリリースではサポートされない可能性が高く、新たに実装することは推奨されません。特定のメジャーリリースにおける非推奨機能の最新情報は、そのメジャーリリースの最新版のリリースノートを参照してください。

非推奨機能のサポートステータスは、Red Hat Enterprise Linux 8 では変更されません。サポート期間の詳細は、[Red Hat Enterprise Linux Life Cycle](#) および [Red Hat Enterprise Linux Application Streams Life Cycle](#) を参照してください。

現行および今後のメジャーリリースでは、非推奨のハードウェアコンポーネントの新規実装は推奨されません。ハードウェアドライバーの更新は、セキュリティと重大な修正のみに行われます。Red Hat では、このようなハードウェアの早期交換を推奨します。

パッケージが非推奨となり、使用の継続が推奨されない場合があります。製品からパッケージが削除されることもあります。その場合には、製品のドキュメントで、非推奨となったパッケージと同様、同一、またはより高度な機能を提供する最近のパッケージが指定され、詳しい推奨事項が記載されます。

RHEL 7 で使用され、RHEL 8 で **削除された** 機能の詳細は [RHEL 8 の導入における検討事項](#) を参照してください。

### 10.1. インストーラーおよびイメージの作成

複数のキックスタートコマンドおよびオプションが非推奨になりました。

RHEL 8 キックスタートファイルで以下のコマンドとオプションを使用すると、ログに警告が表示されます。

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

特定のオプションだけがリスト表示されている場合は、基本コマンドおよびその他のオプションは引き続き利用でき、非推奨ではありません。

キックスタートの詳細および変更点は、RHEL 8 の導入における検討事項の [キックスタートの変更](#) を参照してください。

(BZ#1642765)

**キックスタートコマンド `ignoredisk` の `--interactive` オプションが非推奨になりました。**

Red Hat Enterprise Linux の将来のリリースで `--interactive` オプションを使用すると、致命的なインストールエラーが発生します。このオプションを削除するには、キックスタートファイルを変更することが推奨されます。

(BZ#1637872)

**キックスタートの `autostep` コマンドが非推奨に**

`autostep` コマンドが非推奨になりました。このコマンドに関連するセクションは、[RHEL 8 のドキュメント](#) から削除されました。

(BZ#1904251)

## 10.2. ソフトウェア管理

**`rpmbuild --sign` が非推奨になりました。**

`rpmbuild --sign` コマンドは、RHEL 8.1 以降非推奨になりました。Red Hat Enterprise Linux の今後のリリースでこのコマンドを実行すると、エラーが発生します。代わりに `rpmsign` コマンドを使用することが推奨されます。

(BZ#1688849)

## 10.3. シェルおよびコマンドラインツール

**OpenEXR コンポーネントが非推奨になりました。**

OpenEXR コンポーネントが非推奨になりました。そのため、EXR イメージ形式のサポートは `imagecodecs` モジュールから削除されました。

(BZ#1886310)

**`dump` からの `dump` ユーティリティーが非推奨になりました。**

ファイルシステムのバックアップに使用される `dump` ユーティリティーが非推奨になり、RHEL 9 では使用できなくなります。

RHEL 9 では、使用方法に基づいて、`tar`、`dd`、または `bacula` のバックアップユーティリティーを使用することが推奨されています。これにより、`ext2`、`ext3`、および `ext4` のファイルシステムで完全に安全なバックアップが提供されます。

`dump` パッケージの `restore` ユーティリティーは、RHEL 9 で引き続き利用可能で、サポートされており、`restore` パッケージとして利用できます。

(BZ#1997366)

**ABRT ツールは非推奨になりました**

アプリケーションのクラッシュを検出して報告するための自動バグ報告ツール (ABRT) は、RHEL8 で非推奨になりました。代わりに、**systemd-coredump** ツールを使用して、プログラムのクラッシュ後に自動的に生成されるファイルであるコアダンプをログに記録して保存します。

(BZ#2055826)

### ReaR crontab は非推奨になりました

**rear** パッケージの `/etc/cron.d/rear` は RHEL 8 で非推奨になり、RHEL 9 では使用できなくなります。crontab は、ディスクレイアウトが変更されたかどうかを毎晩チェックし、変更が発生した場合は **rear mkrescue** コマンドを実行します。

この機能が必要な場合は、RHEL 9 にアップグレードした後、ReaR の定期的な実行を手動で設定してください。

(BZ#2083301)

### Bacula の SQLite データベースバックエンドは廃止されました

Bacula バックアップシステムは、複数のデータベースバックエンド (PostgreSQL、MySQL、および SQLite) をサポートしていました。SQLite バックエンドは廃止され、RHEL の今後のリリースではサポートされなくなります。代わりに、他のバックエンド (PostgreSQL または MySQL) のいずれかに移行し、新しい展開では SQLite バックエンドを使用しないでください。

(BZ#2089399)

### hidepid=n マウントオプションが、RHEL 8 systemd で未サポート

マウントオプションの **hidepid=n** は、`/proc/[pid]` ディレクトリーの情報にアクセスできるユーザーを制御しますが、RHEL 8 で提供されている **systemd** インフラストラクチャーと互換性がありません。

また、このオプションを使用すると、**systemd** が起動する特定のサービスで SELinux の AVC 拒否メッセージが生成され、その他の操作が完了しないようにする場合があります。

詳細は、関連するナレッジベースソリューションを参照してください。[RHEL7 および RHEL8 では、hidepid=2 での /proc のマウントが推奨されますか。](#)

(BZ#2038929)

### /usr/lib/udev/rename\_device ユーティリティーは非推奨になる

ネットワークインターフェイスの名前を変更するための **udev** ヘルパーユーティリティー `/usr/lib/udev/rename_device` は非推奨になる

(BZ#1875485)

### raw コマンドは廃止されました

**raw** (`/usr/bin/raw`) コマンドは廃止されました。Red Hat Enterprise Linux の今後のリリースでこのコマンドを実行すると、エラーが発生します。

(JIRA:RHELPLAN-133171)

## 10.4. セキュリティー

### NSS SEED 暗号が非推奨になりました。

Mozilla Network Security Services (NSS) ライブラリーでは、今後のリリースで SEED 暗号化を使用す

る TLS 暗号スイートのサポートがなくなります。NSS がサポートを削除した際に SEED 暗号に依存するデプロイメントを円滑に移行させるため、Red Hat は、他の暗号スイートのサポートを有効にすることを推奨します。

RHEL では、SEED 暗号はデフォルトですでに無効にされています。

(BZ#1817533)

### TLS 1.0 および TLS 1.1 が非推奨になりました。

TLS 1.0 プロトコルおよび TLS 1.1 プロトコルは、システム全体の暗号化ポリシーレベル **DEFAULT** で無効になります。たとえば、Firefox Web ブラウザーのビデオ会議アプリケーションで、非推奨のプロトコルを使用する必要がある場合は、システム全体の暗号化ポリシーを **LEGACY** レベルに変更してください。

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

詳細は、Red Hat カスタマーポータルナレッジベースの記事 [Strong crypto defaults in RHEL 8 and deprecation of weak crypto algorithms](#) および man ページの [update-crypto-policies\(8\)](#) を参照してください。

(BZ#1660839)

### RHEL 8 で DSA が非推奨になりました。

デジタル署名アルゴリズム (DSA) は、Red Hat Enterprise Linux 8 では非推奨であると考えられています。DSA キーに依存する認証メカニズムはデフォルト設定では機能しません。**OpenSSH** クライアントは、**LEGACY** のシステム全体の暗号化ポリシーレベルでも DSA ホストキーを許可しません。

(BZ#1646541)

### NSS で SSL2 Client Hello が非推奨に

**TLS** (Transport Layer Security) プロトコルバージョン 1.2 以前は、**SSL** (Secure Sockets Layer) プロトコルバージョン 2 と後方互換性がある形式の **Client Hello** メッセージを使用してネゴシエーションを開始できます。**NSS** (Network Security Services) ライブラリーでのこの機能への対応は非推奨となっており、デフォルトで無効になっています。

この機能への対応が必要なアプリケーションを有効にするには、新しい API の **SSL\_ENABLE\_V2\_COMPATIBLE\_HELLO** を使用する必要があります。この機能への対応は、Red Hat Enterprise Linux 8 の将来のリリースから完全に削除される可能性があります。

(BZ#1645153)

### TPM 1.2 が非推奨になりました。

Trusted Platform Module (TPM) のセキュアな暗号化プロセッサの標準バージョンが 2016 年にバージョン 2.0 に更新されました。TPM 2.0 は TPM 1.2 に対する多くの改良を提供しますが、以前のバージョンと後方互換性はありません。TPM 1.2 は RHEL 8 で非推奨となり、次のメジャーリリースで削除される可能性があります。

(BZ#1657927)

### crypto-policies から派生したプロパティーが非推奨に

カスタムポリシーにおける **crypto-policies** ディレクティブのスキープの導入により、**tls\_cipher**、**ssh\_cipher**、**ssh\_group**、**ike\_protocol**、および **sha1\_in\_dnssec** の派生プロパティーが非推奨になりました。さらに、スキープを指定しない **protocol** プロパティーの使用も非推奨

になりました。推奨される代替は、**crypto-policies(7)** の man ページを参照してください。

(BZ#2011208)

**/etc/selinux/config** を使用して SELinux を無効にするランタイムが非推奨になりました。

**/etc/selinux/config** ファイルの **SELINUX=disabled** オプションを使用して SELinux を無効にするランタイムが非推奨になりました。RHEL 9 では、**/etc/selinux/config** でのみ SELinux を無効にすると、システムは SELinux が有効化されますが、ポリシーが読み込まれずに開始します。

SELinux を完全に無効にする必要がある場合には、Red Hat は、**selinux=0** パラメーターをカーネルコマンドラインに追加して SELinux を無効にすることを推奨します。これは、[SELinux の使用](#) タイトルの [システムの起動時に SELinux モードの変更](#) セクションで説明されています。

(BZ#1932222)

**selinux-policy** から **ipa** SELinux モジュールが削除されました。

**ipa** SELinux はメンテナンスされなくなったため、**selinux-policy** から削除されました。この機能は、**ipa-selinux** サブパッケージに含まれるようになりました。

ローカルの SELinux ポリシーで、**ipa** モジュールからタイプやインターフェイスを使用する必要がある場合は、**ipa-selinux** をインストールします。

(BZ#1461914)

**fapolicyd.rules** が非推奨に

実行ルールの許可と拒否を含むファイルの **/etc/fapolicyd/rules.d/** ディレクトリーは、**/etc/fapolicyd/fapolicyd.rules** ファイルを置き換えます。**fagenrules** スクリプトは、このディレクトリー内のすべてのコンポーネントルールファイルを **/etc/fapolicyd/compiled.rules** ファイルにマージするようになりました。**/etc/fapolicyd/fapolicyd.trust** のルールは引き続き **fapolicyd** フレームワークによって処理されますが、下位互換性を確保するためにのみ使用されます。

(BZ#2054741)

## 10.5. ネットワーキング

**RHEL 8 でネットワークスクリプトが非推奨に**

Red Hat Enterprise Linux 8 では、ネットワークスクリプトが非推奨になっており、デフォルトでは提供されなくなりました。基本的なインストールでは、**nmcli** ツール経由で、NetworkManager サービスを呼び出す **ifup** スクリプトおよび **ifdown** スクリプトの新しいバージョンが提供されます。Red Hat Enterprise Linux 8 で **ifup** スクリプトおよび **ifdown** スクリプトを実行する場合は、NetworkManager を実行する必要があります。

**/sbin/ifup-local**、**ifdown-pre-local**、および **ifdown-local** の各スクリプトでは、カスタムコマンドが実行されません。

このスクリプトが必要な場合は、次のコマンドを使用すれば、システムに非推奨のネットワークスクリプトをインストールできます。

```
# yum install network-scripts
```

**ifup** スクリプトと **ifdown** スクリプトが、インストールされている従来のネットワークスクリプトにリンクされます。

従来のネットワークスクリプトを呼び出すと、そのスクリプトが非推奨であることを示す警告が表示されます。

(BZ#1647725)

### dropwatch ツールが非推奨に

**dropwatch** ツールが非推奨になりました。このツールは今後のリリースではサポートされませんので、新規デプロイメントには推奨できません。このパッケージの代わりに、Red Hat は **perf** コマンドラインツールを使用することを推奨します。

**perf** コマンドラインツールの使用方法の詳細は、Red Hat カスタマーポータル [Getting started with Perf](#) セクションまたは **perf** の man ページを参照してください。

(BZ#1929173)

### cgdcboxd パッケージが非推奨に

コントロールグループデータセンターブリッジ交換デーモン (**cgdcboxd**) は、データセンターのブリッジ (DCB) のネットリンクイベントをモニターし **net\_prio control** グループサブシステムを管理するサービスです。RHEL 8.5 以降では、**cgdcboxd** パッケージは非推奨となり、次の RHEL メジャーリリースで削除されます。

(BZ#2006665)

### xinetd が非推奨に

**xinetd** サービスが非推奨になり、RHEL 9 では削除される予定です。代わりに **systemd** を使用します。詳細は、[xinetd サービスを systemd に変換する方法](#) を参照してください。

(BZ#2009113)

### WEP Wi-Fi 接続方法が非推奨になりました。

安全でない WEP (wired equivalent privacy) の Wi-Fi 接続方法は、RHEL 8 では非推奨となり、RHEL 9.0 では削除されます。安全な Wi-Fi 接続には、Wi-Fi Protected Access 3 (WPA3) または WPA2 の接続方法を使用します。

(BZ#2029338)

### サポートされていない xt\_u32 モジュールが非推奨になりました。

サポートされていない **xt\_u32** を使用すると、**iptables** のユーザーはパケットヘッダーまたはペイロード内の任意の 32 ビットにマッチできます。RHEL 8.6 以降、**xt\_u32** モジュールが非推奨になり、RHEL 9 では削除されます。

**xt\_u32** を使用する場合は、**nftable** パケットフィルタリングフレームワークに移行します。たとえば、最初にファイアウォールを、個々のルールを段階的に置き換えるために、ネイティブで **iptables** を使用するように変更し、その後に **iptables-translate** と付属のユーティリティを使用して **nftable** に移行します。**nftable** にネイティブマッチが存在しない場合は、**nftable** の raw ペイロードマッチング機能を使用します。詳細は、**nft(8)** man ページの **raw ペイロード表現** セクションを参照してください。

(BZ#2061288)

### スレーブ という用語は、nmstate API では非推奨です。

Red Hat では、意識的な言語の使用に取り組んでいます。この取り組みの詳細は、[オープンソースをより包括的にする](#) を参照してください。したがって、**スレーブ** という用語は Nmstate API では非推奨です。**nmstatectl** を使用する場合は、**port** という用語を使用します。

(JIRA:RHELDPCS-17641)

## 10.6. カーネル

**カーネルライブパッチが、すべての RHEL マイナーリリースに対応するようになりました。**

RHEL 8.1以降、カーネルライブパッチは、影響度が重大および重要な Common Vulnerabilities and Exposures (CVE) を修正するために、Extended Update Support (EUS) ポリシーの対象となる RHEL の一部のマイナーリリースストリームに提供されています。同時にカバーされるカーネルとユースケースの最大数に対応するため、各ライブパッチのサポート期間は、カーネルのマイナー、メジャー、および zStream の各バージョンで 12 カ月から 6 カ月に短縮されました。これは、カーネルライブパッチがリリースされると、過去 6 カ月間に配信されたすべてのマイナーリリースとスケジュール済みのエラータカーネルが含まれます。

この機能の詳細は、[Applying patches with kernel live patching](#) を参照してください。

利用可能なカーネルライブパッチの詳細は、[Kernel Live Patch life cycles](#) を参照してください。

(BZ#1958250)

### **crash-ptdump-command パッケージは非推奨です**

クラッシュユーティリティの **ptdump** 拡張モジュールである **crash-ptdump-command** パッケージは非推奨であり、将来の RHEL リリースでは利用できなくなる可能性があります。 **ptdump** コマンドは、Single Range Output モードで作業している場合、ログバッファの取得に失敗し、Table of Physical Addresses (ToPA) モードでのみ機能します。 **crash-ptdump-command** は現在、アップストリームに維持されていません

(BZ#1838927)

**ディスクレスブートを使用した RHEL for Real Time 8 のインストールが非推奨になりました。**

ディスクレスブートにより、複数のシステムがネットワーク経由で root ファイルシステムを共有できます。メリットはありますが、ディスクレスブートでは、リアルタイムのワークロードでネットワークレイテンシーが発生する可能性が高くなります。RHEL for Real Time 8 の将来のマイナー更新では、ディスクレスブート機能はサポートされなくなります。

(BZ#1748980)

**Linux firewire サブシステムおよび関連するユーザー空間コンポーネントは、RHEL 8 では非推奨になりました。**

**firewire** サブシステムは、IEEE 1394 バスでリソースを使用し、維持するインターフェイスを提供します。RHEL 9 では、**firewire** は、**kernel** パッケージで対応しなくなります。 **firewire** には、**libavc1394**、**libdc1394**、**libraw1394** パッケージで提供されるユーザー空間コンポーネントが複数含まれることに注意してください。これらのパッケージも非推奨になります。

(BZ#1871863)

### **rdma\_rxe Soft-RoCE ドライバーが非推奨に**

Remote Software Direct Memory Access over Converged Ethernet (Soft-RoCE) は RXE としても知られており、RDMA (Remote Direct Memory Access) をエミュレートする機能です。RHEL 8 では、Soft-RoCE 機能が、サポートされていないテクノロジープレビューとして利用できます。ただし、安定性の問題により、この機能は非推奨になり、RHEL 9 では削除されます。

(BZ#1878207)

## 10.7. ブートローダー

### kernelopts 環境変数は非推奨になる

RHEL 8 では、GRUB2 ブートローダーを使用するシステムのカーネルコマンドラインパラメーターが **kernelopts** 環境変数で定義されていました。変数は、カーネルブートエントリーごとに `/boot/grub2/grubenv` ファイルに保存されました。ただし、**kernelopts** を使用してカーネルコマンドラインパラメーターを保存することは堅牢ではありませんでした。したがって、RHEL の将来のメジャー更新では **kernelopts** が削除され、代わりにカーネルコマンドラインパラメーターが Boot Loader Specification (BLS) スニペットに格納されます。

([BZ#2060759](#))

## 10.8. ファイルシステムおよびストレージ

### async 以外の VDO 書き込みモードが非推奨に

VDO は、RHEL 8 で複数の書き込みモードに対応します。

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

RHEL 8.4 以降、以下の書き込みモードが非推奨になりました。

#### sync

VDO レイヤー上のデバイスは、VDO が同期されているかどうかを認識できないため、デバイスは VDO **sync** モードを利用できません。

#### async-unsafe

VDO は、ACID (Atomicity, Consistency, Isolation, and Durability) に準拠する **async** モードの回避策としてこの書き込みモードを追加しました。Red Hat は、ほとんどのユースケースで **async-unsafe** を推奨せず、それに依存するユーザーを認識しません。

#### auto

この書き込みモードは、他の書き込みモードのいずれかのみを選択します。VDO が1つの書き込みモードのみに対応している場合は、不要になりました。

この書き込みモードは、今後の RHEL メジャーリリースで削除されます。

推奨される VDO 書き込みモードが **async** になりました。

VDO 書き込みモードの詳細は、[VDO 書き込みモードの選択](#) を参照してください。

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

### NFSv3 over UDP が無効になりました。

NFS サーバーは、デフォルトで UDP (User Datagram Protocol) ソケットを開いたり、リッスンしなくなりました。バージョン 4 では TCP (Transmission Control Protocol) が必要なため、この変更は NFS バージョン 3 にのみ影響を及ぼします。

RHEL 8 では、NFS over UDP に対応しなくなりました。

(BZ#1592011)

### **cramfs が非推奨になりました。**

ユーザーの不足により、**cramfs** カーネルモジュールが非推奨になりました。代替策として **squashfs** が推奨されます。

(BZ#1794513)

### **VDO マネージャーが非推奨に**

python ベースの VDO 管理ソフトウェアは非推奨となり、RHEL 9 から削除される予定です。RHEL 9 では、LVM-VDO 統合に置き換えられます。そのため、**lvcreate** コマンドを使用して VDO ボリュームを作成することが推奨されます。

VDO 管理ソフトウェアを使用して作成した既存のボリュームは、**lvms2** パッケージが提供する **/usr/sbin/lvm\_import\_vdo** スクリプトを使用して変換できます。LVM-VDO 実装の詳細は、[RHEL での論理ボリュームの重複排除および圧縮](#) を参照してください。

(BZ#1949163)

### **elevator カーネルコマンドラインパラメーターが非推奨になりました。**

カーネルコマンドラインパラメーターの **elevator** は、すべてのデバイスのディスクスケジューラーを設定するために、以前の RHEL リリースで使用されていました。RHEL 8 では、このパラメーターが非推奨になりました。

アップストリームの Linux カーネルでは、**elevator** パラメーターに対応しなくなりましたが、互換性のために RHEL 8 でも引き続き利用できます。

カーネルは、デバイスのタイプに基づいてデフォルトのディスクスケジューラーを選択することに注意してください。これは通常、最適な設定です。別のスケジューラーが必要な場合は、**udev** ルールまたは TuneD サービスを使用して設定することが推奨されます。選択したデバイスを一致させ、それらのデバイスのスケジューラーのみを切り替えます。

詳しい情報は、[ディスクスケジューラーの設定](#) を参照してください。

(BZ#1665295)

### **LVM mirror が非推奨化されました。**

LVM **mirror** セグメントタイプが非推奨になりました。**mirror** のサポートは、RHEL の今後のメジャーリリースで削除されます。

Red Hat は、セグメントタイプが **mirror** ではなく、**raid1** の LVM RAID 1 デバイスを使用することを推奨します。**raid1** のセグメントタイプは、デフォルトの RAID 設定タイプで、**mirror** の代わりに、推奨のソリューションとしてこのタイプが使用されます。

**mirror** デバイスを **raid1** に変換するには、[ミラーリングされた LVM デバイスの RAID1 論理ボリュームへの変換](#) を参照してください。

LVM **mirror** には既知の問題が複数あります。詳細は、[ファイルシステムおよびストレージの既知の問題](#) を参照してください。

(BZ#1827628)

### **peripety が非推奨に**

**peripety** パッケージは、RHEL 8.3 以降で非推奨になりました。

Peripety ストレージイベント通知デーモンは、システムストレージログを構造化されたストレージイベントに解析します。ストレージの問題を調査するのに役立ちます。

(BZ#1871953)

## 10.9. 高可用性およびクラスター

**clufter** ツールに対応する **pcs** コマンドが非推奨になりました。

クラスター設定フォーマットを分析する **clufter** ツールに対応する **pcs** コマンドが非推奨になりました。これらのコマンドにより、コマンドが非推奨になり、コマンドに関連するセクションが **pcs** ヘルプ表示と、**pcs(8)** man ページから削除されていることを示す警告が出力されるようになりました。

以下のコマンドが非推奨になりました。

- **pcs config import-cman**: CMAN / RHEL6 HA クラスター設定のインポート
- **pcs config export**: クラスター設定を、同じクラスターを再作成する **pcs** コマンドのリストにエクスポート

(BZ#1851335)

## 10.10. 動的プログラミング言語、WEB サーバー、およびデータベースサーバー

Apache HTTP サーバーで使用するために PHP に提供されている **mod\_php** モジュールが非推奨になりました。

RHEL 8 の Apache HTTP サーバーで使用するために PHP に付属している **mod\_php** モジュールは利用可能ですが、デフォルト設定では有効になっていません。このモジュールは RHEL 9 では使用できなくなりました。

RHEL 8 以降、PHP スクリプトはデフォルトで FastCGI Process Manager (**php-fpm**) を使用して実行されます。詳細は、[Apache HTTP サーバーでの PHP の使用](#) を参照してください。

(BZ#2225332)

## 10.11. コンパイラーおよび開発ツール

**libdwarf** が非推奨に

RHEL 8 では、**libdwarf** ライブラリーが非推奨になりました。ライブラリーは、将来のメジャーリリースでサポートされない可能性があります。代わりに、ELF/DWARF ファイルを処理するアプリケーションに **elfutils** および **libdw** ライブラリーを使用してください。

**libdwarf-tools dwarfdump** プログラムの代替は、**binutils readelf** プログラムまたは **elfutils eu-readelf** プログラムになります。どちらも **--debug-dump** フラグを渡すことで使用されます。

(BZ#1920624)

**gdb.i686** パッケージが非推奨に

RHEL 8.1 では、別のパッケージの依存関係の問題が原因で、32 ビットバージョンの GNU Debugger(GDB) **gdb.i686** が同梱されていました。RHEL 8 は 32 ビットハードウェアに対応していないため、RHEL 8.4 以降、**gdb.i686** パッケージは非推奨になりました。64 ビットバージョンの GDB

**(gdb.x86\_64)** は、32 ビットアプリケーションをデバッグできます。

**gdb.i686** を使用する場合は、以下の重要な問題に注意してください。

- **gdb.i686** パッケージは更新されなくなりました。代わりに **gdb.x86\_64** をインストールする必要があります。
- **gdb.i686** をインストールしている場合は、**gdb.x86\_64** をインストールすると、**yum** が **package gdb-8.2-14.el8.x86\_64 obsoletes gdb < 8.2-14.el8 provided by gdb-8.2-12.el8.i686** を報告します。これは想定される状況です。**gdb.i686** をアンインストールするか、**--allowerase** オプションを **dnf** に渡して **gdb.i686** を削除し、**gdb.x86\_64** をインストールします。
- ユーザーは、64 ビットシステム (つまり、**libc.so.6()(64-bit)** パッケージのある) に **gdb.i686** パッケージをインストールすることができなくなります。

(BZ#1853140)

## 10.12. IDENTITY MANAGEMENT

### openssh-ldap が非推奨に

**openssh-ldap** サブパッケージは、Red Hat Enterprise Linux 8 で非推奨になり、RHEL 9 で削除されます。**openssh-ldap** サブパッケージはアップストリームでは維持されないため、Red Hat は **SSSD** と **sss\_ssh\_authorizedkeys** ヘルパーを使用することを推奨しています。これは、他の IdM ソリューションよりも適切に統合でき、安全です。

デフォルトでは、**ldap** および **ipa** プロバイダーはユーザーオブジェクトの **sshPublicKey** LDAP 属性を読み取ります (利用可能な場合)。AD (Active Directory) には公開鍵を保存するためのデフォルトの LDAP 属性がないため、**ad** プロバイダーまたは IdM の信頼されるドメインのデフォルト SSSD 設定を使用して AD から SSH 公開鍵を取得することはできません。

**sss\_ssh\_authorizedkeys** ヘルパーが SSSD から鍵を取得できるようにするには、**sssd.conf** ファイルの **services** オプションに **ssh** を追加して **ssh** レスポンダーを有効にします。詳細は man ページの **sssd.conf(5)** を参照してください。

**sshd** が **sss\_ssh\_authorizedkeys** を使用できるようにするには、man ページの **sss\_ssh\_authorizedkeys(1)** に記載されているように、**AuthorizedKeysCommand** **/usr/bin/sss\_ssh\_authorizedkeys** および **AuthorizedKeysCommandUser nobody** オプションを **/etc/ssh/sshd\_config** ファイルに追加します。

(BZ#1871025)

### DES および 3DES 暗号化タイプが削除されました。

RHEL 7 以降、セキュリティ上の理由から、データ暗号化標準 (DES) アルゴリズムが非推奨になり、デフォルトで無効化になりました。Kerberos パッケージの最近のリベースで、RHEL 8 からシングル DES (DES) およびトリプル DES (3DES) の暗号化タイプが削除されました。

DES または 3DES の暗号化のみを使用するようにサービスまたはユーザーが設定されている場合、以下のようなサービスの中断が発生する可能性があります。

- Kerberos 認証エラー
- **unknown enctype** 暗号化エラー

- DES で暗号化されたデータベースマスターキー (**K/M**) を使用した KDC (Kerberos Distribution Center) が起動しない

アップグレードを準備するには、以下の操作を実施します。

1. KDC が **krb5check** オープンソース Python スクリプトで DES または 3DES 暗号化を使用しているかどうかを確認します。GitHub の [krb5check](#) を参照してください。
2. Kerberos プリンシパルで DES または 3DES 暗号化を使用している場合は、Advanced Encryption Standard (AES) などのサポート対象の暗号化タイプでキーを変更します。キー変更の手順については、MIT Kerberos ドキュメントの [Retiring DES](#) を参照してください。
3. アップグレードの前に以下の Kerberos オプションを一時的に設定して、DES および 3DES からの独立性をテストします。
  - a. KDC の `/var/kerberos/krb5kdc/kdc.conf` で、**supported\_encyptypes** を設定し、**des** または **des3** は含まれません。
  - b. すべてのホストについて、`/etc/krb5.conf` および `/etc/krb5.conf.d` のすべてのファイルで、**allow\_weak\_crypto** を **false** に設定します。デフォルトは false です。
  - c. すべてのホストについて、`/etc/krb5.conf` および `/etc/krb5.conf.d` のすべてのファイルで、**permitted\_encyptypes**、**default\_tgs\_encyptypes**、**default\_tkt\_encyptypes** を設定します。また、**des** または **des3** は含めません。
4. 前の手順で Kerberos 設定をテストしてサービスが中断されない場合は、サービスを削除してアップグレードします。最新の Kerberos パッケージにアップグレードした後は、この設定は必要ありません。

(BZ#1877991)

### ctdb サービスのスタンドアロン使用が非推奨になりました。

RHEL 8.4 以降、以下の条件がすべて適用されている場合に限り、**ctdb** クラスタ Samba サービスを使用することが推奨されます。

- **ctdb** サービスは、resource-agent **ctdb** を使用して **pacemaker** リソースとして管理されます。
- **ctdb** サービスは、Red Hat Gluster Storage 製品または GFS2 ファイルシステムが提供する GlusterFS ファイルシステムのいずれかが含まれるストレージボリュームを使用します。

**ctdb** サービスのスタンドアロンユースケースは非推奨となり、Red Hat Enterprise Linux の次期メジャーリリースには含まれません。Samba のサポートポリシーの詳細は、ナレッジベースの記事 [Support Policies for RHEL Resilient Storage - ctdb General Policies](#) を参照してください。

(BZ#1916296)

### Samba を PDC または BDC として実行することは非推奨になりました。

管理者が Samba を NT4 のようなプライマリドメインコントローラー (PDC) として実行し、バックアップドメインコントローラー (BDC) を実行できるようにする従来のドメインコントローラーモードが非推奨になりました。これらのモードを設定するためのコードおよび設定は、今後の Samba リリースで削除されます。

RHEL 8 の Samba バージョンが PDC モードおよび BDC モードを提供している限り、Red Hat は、NT4 ドメインに対応する Windows バージョンを使用する既存のインストールでのみ、これらのモードをサポートします。Red Hat は、新規の Samba NT4 ドメインのセットアップを推奨しません。なぜな

ら、Microsoft のオペレーティングシステム (Windows 7 以降) および Windows Server 2008 R2 は、NT4 ドメインをサポートしないからです。

PDC を使用して Linux ユーザーのみを認証する場合、Red Hat は、RHEL サブスクリプションに含まれる [Red Hat Identity Management \(IdM\)](#) への移行を推奨します。ただし、Windows システムを IdM ドメインに参加させることはできません。Red Hat は、引き続き IdM が使用する PDC 機能のサポートを継続することに注意してください。

Red Hat は、Samba を AD ドメインコントローラー (DC) として実行することはサポートしていません。

([BZ#1926114](#))

### WinSync による IdM との間接的な AD 統合が非推奨に

WinSync は、さまざまな機能制限のため、RHEL 8 では積極的に開発されなくなりました。

- WinSync は、1つの Active Directory (AD) ドメインのみをサポートします。
- パスワードの同期には、AD ドメインコントローラーに追加のソフトウェアをインストールする必要があります。

リソースとセキュリティの分離を強化したより強固なソリューションとして、レッドハットは Active Directory との間接的な統合に [フォレスト間の信頼](#) を使用することを推奨しています。[間接的な統合](#) のドキュメントを参照してください。

(JIRA:RHELPLAN-100400)

### SSSD バージョンの libwbclient が削除される

**libwbclient** パッケージの SSSD 実装は、RHEL 8.4 で非推奨になりました。最新バージョンの Samba で使用できないため、**libwbclient** の SSSD 実装が削除されています。

([BZ#1947671](#))

### SMB1 プロトコルは Samba では非推奨に

Samba 4.11 以降、安全でない Server Message Block バージョン 1 (SMB1) プロトコルは非推奨となり、今後のリリースでは削除される予定です。

セキュリティを向上させるために、デフォルトでは、Samba サーバーおよびクライアントユーティリティで SMB1 が無効になっています。

(Jira:RHELDPCS-16612)

### FreeRADIUS のサポートは限定的です

RHEL 8 では、FreeRADIUS サービスの一部として、次の外部認証モジュールが非推奨になりました。

- MySQL、PostgreSQL、SQLite、および unixODBC データベースコネクター
- **Perl** 言語モジュール
- REST API モジュール



## 注記

ベースパッケージの一部として提供される PAM 認証モジュールおよびその他の認証モジュールは影響を受けません。

廃止されたモジュールの代替は、Fedora プロジェクトなどのコミュニティでサポートされているパッケージで見つけることができます。

さらに、**freeradius** パッケージのサポート範囲は、将来の RHEL リリースでは次のユースケースに限定されます。

- FreeRADIUS をワイヤレス認証プロバイダーとして使用し、Identity Management (IdM) を認証のバックエンドソースとして使用します。認証は、**krb5** および LDAP 認証パッケージを使用して、またはメインの FreeRADIUS パッケージの PAM 認証として行われます。
- FreeRADIUS を使用して、Python 3 認証パッケージで IdM の認証用に信頼できる情報源を提供します。

これらの廃止とは対照的に、Red Hat は FreeRADIUS による次の外部認証モジュールのサポートを強化します。

- **krb5** および LDAP に基づく認証
- **Python 3** 認証

これらのインテグレーションオプションに重点を置くことは、Red Hat IdM の戦略的方向性に一致します。

(Jira:RHELDPCS-17573)

## 10.13. デスクトップ

**libgnome-keyring** ライブラリーが非推奨になりました。

**libgnome-keyring** ライブラリーがアップストリームで維持されず、RHEL に必要な暗号化ポリシーに従っていないため、**libsecret** ライブラリーが **libgnome-keyring** ライブラリーを引き継ぎ、**libgnome-keyring** は非推奨となりました。新しい **libsecret** ライブラリーは、必要なセキュリティー標準に準拠する代替ライブラリーです。

(BZ#1607766)

## 10.14. グラフィックインフラストラクチャー

**AGP グラフィックカードがサポートされなくなりました。**

AGP (Accelerated Graphics Port) バスを使用するグラフィックカードは、Red Hat Enterprise Linux 8 ではサポートされていません。推奨される代替として、PCI-Express バスを備えたグラフィックスカードを使用してください。

(BZ#1569610)

**Motif は非推奨になりました**

アップストリームの Motif コミュニティーでの開発は非アクティブであるため、Motif ウィジェットツールキットは RHEL で非推奨になりました。

開発バリエーションおよびデバッグバリエーションを含む、以下の Motif パッケージが非推奨になりました。

- **motif**
- **openmotif**
- **openmotif21**
- **openmotif22**

さらに、**motif-static** パッケージが削除されました。

Red Hat は、GTK ツールキットを代替として使用することを推奨します。GTK は Motif と比較してメンテナンス性が高く、新機能を提供します。

(JIRA:RHELPLAN-98983)

## 10.15. WEB コンソール

Web コンソールは、不完全な翻訳への対応を終了しました。

RHEL Web コンソールは、コンソールの翻訳可能な文字列の翻訳率が 50 % 未満の言語に対する翻訳提供を廃止しました。ブラウザがこのような言語に翻訳を要求すると、ユーザーインターフェイスは英語になります。

(BZ#1666722)

## 10.16. RED HAT ENTERPRISE LINUX システムロール

RHEL 9 ノードでチームを設定すると、**networking** システムロールが非推奨の警告を表示しません。

ネットワークチーム機能は、RHEL 9 では非推奨になりました。その結果、RHEL 8 コントローラーの **networking** RHEL システムロールを使用して RHEL 9 ノードにネットワークチームを設定すると、非推奨に関する警告が表示されます。

(BZ#2021685)

**Ansible Engine は非推奨になりました**

以前のバージョンの RHEL8 は、サポートの範囲が限定された Ansible Engine リポジトリへのアクセスを提供し、RHEL System Roles や Insights 救済策などのサポートされた RHEL Automation ユースケースを有効にしました。Ansible Engine は非推奨になり、Ansible Engine 2.9 は 2023 年 9 月 29 日以降サポートされなくなります。サポートされているユースケースの詳細については、[RHEL 9 および RHEL 8.6 以降の AppStream リポジトリに含まれる Ansible Core パッケージのサポート対象範囲](#) を参照してください。

ユーザーは、システムを Ansible Engine から Ansible Core に手動で移行する必要があります。そのためには、以下の手順に従います。

### 手順

1. システムが RHEL 8.7 を実行しているかどうかを確認します。

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

2. Ansible Engine 2.9 をインストールします。

```
# yum remove ansible
```

3. **ansible-2-for-rhel-8-x86\_64-rpms** リポジトリを無効にします。

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

4. RHEL 8 AppStream リポジトリから Ansible Core パッケージをインストールします。

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

詳細については、[RHEL8.6 以降での Ansible の使用](#) を参照してください。

([BZ#2006081](#))

### geoipupdate パッケージが非推奨に

**geoipupdate** パッケージにはサードパーティーのサブスクリプションが必要で、プロプライエタリーコンテンツもダウンロードします。したがって、**geoipupdate** パッケージは非推奨となり、次の RHEL メジャーバージョンで削除されます。

([BZ#1874892](#))

## 10.17. 仮想化

### virsh iface-\* コマンドが非推奨になりました。

**virsh iface-start**、**virsh iface-destroy** などの **virsh iface-\*** コマンドは非推奨になり、将来のメジャーバージョンの RHEL では削除される予定です。また、このようなコマンドは設定の依存関係により頻繁に失敗します。

したがって、ホストネットワーク接続の設定および管理には **virsh iface-\*** コマンドを使用しないことが推奨されます。代わりに、NetworkManager プログラムと、関連する管理アプリケーション (**nmcli** など) を使用します。

([BZ#1664592](#))

### virt-manager が非推奨になりました。

Virtual Machine Manager アプリケーション (**virt-manager**) は非推奨になっています。RHEL Web コンソール (**Cockpit**) は、後続のリリースで置き換えられる予定です。したがって、GUI で仮想化を管理する場合は、Web コンソールを使用することが推奨されます。ただし、**virt-manager** で利用可能な機能によっては、RHEL Web コンソールで利用できない場合があります。

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

### 仮想マシンスナップショットのサポートが限定されました

仮想マシンのスナップショットの作成は、現在、UEFI ファームウェアを使用していない仮想マシンのみでサポートされています。さらに、スナップショット操作中に QEMU モニターがブロックされる可能性があり、これは特定のワークロードのハイパーバイザーのパフォーマンスに悪影響を及ぼします。

また、現在の仮想マシンスナップショットの作成メカニズムは非推奨となり、Red Hat は実稼働環境での仮想マシンスナップショットの使用を推奨していないことにも注意してください。

(BZ#1686057)

### Cirrus VGA 仮想 GPU タイプが非推奨に

Red Hat Enterprise Linux の将来のメジャー更新では、KVM 仮想マシンで **Cirrus VGA** GPU デバイスに対応しなくなります。したがって、Red Hat は **Cirrus VGA** の代わりに **stdvga** または **virtio-vga** デバイスの使用を推奨します。

(BZ#1651994)

### IBM POWER 上の KVM が非推奨に

IBM POWER ハードウェアでの KVM 仮想化の使用は非推奨になりました。その結果、IBM POWER の KVM は、RHEL 8 でも引き続きサポートされますが、RHEL の今後のメジャーリリースではサポートされなくなります。

(JIRA:RHELPLAN-71200)

### SHA1 ベースの署名を使用した SecureBoot イメージ検証が非推奨に

UEFI (PE/COFF) 実行ファイルでの SHA1 ベースの署名を使用した SecureBoot イメージ検証の実行は非推奨になりました。代わりに、Red Hat は、SHA2 アルゴリズムまたはそれ以降に基づく署名を使用することを推奨します。

(BZ#1935497)

### SPICE を使用したスマートカードリーダーの仮想マシンへの接続が非推奨となりました

RHEL 8 では、SPICE リモートディスプレイプロトコルが非推奨になりました。スマートカードリーダーを仮想マシンに割り当てる唯一の推奨される方法は、SPICE プロトコルに依存するため、仮想マシンでのスマートカードの使用も RHEL 8 で非推奨になりました。

RHEL の将来のメジャーバージョンでは、スマートカードリーダーを仮想マシンに割り当てる機能は、サードパーティーのリモート可視化ソリューションでのみサポートされる予定です。

(BZ#2059626)

### SPICE が非推奨になりました

SPICE リモートディスプレイプロトコルが非推奨になりました。その結果、SPICE は RHEL 8 でも引き続きサポートされますが、Red Hat はリモートディスプレイストリーミングに代替ソリューションを使用することを推奨しています。

- リモートコンソールへのアクセスには、VNC プロトコルを使用します。
- 高度なリモートディスプレイ機能には、RDP、HP RGS、または Mechdyne TGX などのサードパーティーツールを使用します。

SPICE で使用される **QXL** グラフィックスデバイスも非推奨になっていることに注意してください。

(BZ#1849563)

## 10.18. コンテナ

### Podman varlink ベースの API v1.0 が削除されました

Podman varlink ベースの API v1.0 は、以前のリリースの RHEL 8 で非推奨となりました。Podman v2.0 には、新しい Podman v2.0 RESTful API が導入されました。Podman v3.0 のリリースでは、varlink ベースの API v1.0 が完全に削除されました。

(JIRA:RHELPLAN-45858)

### container-tools:1.0 が非推奨に

**container-tools:1.0** モジュールは非推奨となり、セキュリティ更新を受信しなくなります。**container-tools:2.0** や **container-tools:3.0** などの新しいサポートされる安定したモジュールストリームを使用することが推奨されます。

(JIRA:RHELPLAN-59825)

### container-tools:2.0 モジュールは非推奨になりました

container-tools:2.0 モジュールは非推奨となり、セキュリティ更新を受信しなくなります。**container-tools:3.0** など、サポートされている新しい安定したモジュールストリームの使用を推奨します。

(JIRA:RHELPLAN-85066)

### GIMP 以外の Flatpak イメージは廃止されました

**rhel8/firefox-flatpak**、**rhel8/thunderbird-flatpak**、**rhel8/inkscape-flatpak**、および **rhel8/libreoffice-flatpak** RHEL 8 Flatpak アプリケーションは廃止され、RHEL 9 バージョンに置き換えられました。RHEL 9 にはまだ代替品がないため、**rhel8/gimp-flatpak** Flatpak アプリケーションは非推奨ではありません。

(BZ#2142499)

## 10.19. 非推奨のパッケージ

このセクションでは、非推奨となり、将来バージョンの Red Hat Enterprise Linux には含まれない可能性があるパッケージのリストを示します。

RHEL 7 と RHEL 8 との間でパッケージを変更する場合は、**RHEL 8 の導入における考慮事項** ドキュメントの [パッケージの変更](#) を参照してください。

以下のパッケージは非推奨となり、RHEL 8 のライフサイクルが終了するまでサポートされ続けます。

- 389-ds-base-legacy-tools
- abrt
- abrt-addon-ccpp
- abrt-addon-kerneloops
- abrt-addon-pstoreoops
- abrt-addon-vmcore
- abrt-addon-xorg
- abrt-cli
- abrt-console-notification
- abrt-dbus
- abrt-desktop

- abrt-gui
- abrt-gui-libs
- abrt-libs
- abrt-tui
- adobe-source-sans-pro-fonts
- adwaita-qt
- alsa-plugins-pulseaudio
- amanda
- amanda-client
- amanda-libs
- amanda-server
- ant-contrib
- antlr3
- antlr32
- aopalliance
- apache-commons-collections
- apache-commons-compress
- apache-commons-exec
- apache-commons-jxpath
- apache-commons-parent
- apache-ivy
- apache-parent
- apache-resource-bundles
- apache-sshd
- apiguardian
- aspnetcore-runtime-3.0
- aspnetcore-runtime-3.1
- aspnetcore-runtime-5.0
- aspnetcore-targeting-pack-3.0

- aspnetcore-targeting-pack-3.1
- aspnetcore-targeting-pack-5.0
- assertj-core
- authd
- auto
- autoconf213
- autogen
- autogen-libopts
- awscli
- base64coder
- batik
- batik-css
- batik-util
- bea-stax
- bea-stax-api
- bind-export-devel
- bind-export-libs
- bind-libs-lite
- bind-pkcs11
- bind-pkcs11-devel
- bind-pkcs11-libs
- bind-pkcs11-utils
- bind-sdb
- bind-sdb
- bind-sdb-chroot
- bluez-hid2hci
- boost-jam
- boost-signals
- bouncycastle

- bpg-algeti-fonts
- bpg-chveulebrivi-fonts
- bpg-classic-fonts
- bpg-courier-fonts
- bpg-courier-s-fonts
- bpg-dedaena-block-fonts
- bpg-dejavu-sans-fonts
- bpg-elite-fonts
- bpg-excelsior-caps-fonts
- bpg-excelsior-condenced-fonts
- bpg-excelsior-fonts
- bpg-fonts-common
- bpg-glaho-fonts
- bpg-gorda-fonts
- bpg-ingiri-fonts
- bpg-irubaqidze-fonts
- bpg-mikhail-stephan-fonts
- bpg-mrgvlovani-caps-fonts
- bpg-mrgvlovani-fonts
- bpg-nateli-caps-fonts
- bpg-nateli-condenced-fonts
- bpg-nateli-fonts
- bpg-nino-medium-cond-fonts
- bpg-nino-medium-fonts
- bpg-sans-fonts
- bpg-sans-medium-fonts
- bpg-sans-modern-fonts
- bpg-sans-regular-fonts
- bpg-serif-fonts

- `bpg-serif-modern-fonts`
- `bpg-ucnobi-fonts`
- `brlapi-java`
- `bsh`
- `buildnumber-maven-plugin`
- `byaccj`
- `call0n`
- `cbi-plugins`
- `cdparanoia`
- `cdparanoia-devel`
- `cdparanoia-libs`
- `cdrdao`
- `cmirror`
- `codehaus-parent`
- `codemodel`
- `compat-exiv2-026`
- `compat-guile18`
- `compat-hwloc1`
- `compat-libpthread-nonshared`
- `compat-libtiff3`
- `compat-openssl10`
- `compat-sap-c++-11`
- `compat-sap-c++-10`
- `compat-sap-c++-9`
- `createrepo_c-devel`
- `ctags`
- `ctags-etags`
- `custodia`
- `cyrus-imapd-vzic`

- dbus-c++
- dbus-c++-devel
- dbus-c++-glib
- dbxtool
- dhcp-libs
- directory-maven-plugin
- directory-maven-plugin-javadoc
- dirsplit
- dleyna-connector-dbus
- dleyna-core
- dleyna-renderer
- dleyna-server
- dnssec-trigger
- dnssec-trigger-panel
- dotnet-apphost-pack-3.0
- dotnet-apphost-pack-3.1
- dotnet-apphost-pack-5.0
- dotnet-host-fxr-2.1
- dotnet-host-fxr-2.1
- dotnet-hostfxr-3.0
- dotnet-hostfxr-3.1
- dotnet-hostfxr-5.0
- dotnet-runtime-2.1
- dotnet-runtime-3.0
- dotnet-runtime-3.1
- dotnet-runtime-5.0
- dotnet-sdk-2.1
- dotnet-sdk-2.1.5xx
- dotnet-sdk-3.0

- dotnet-sdk-3.1
- dotnet-sdk-5.0
- dotnet-targeting-pack-3.0
- dotnet-targeting-pack-3.1
- dotnet-targeting-pack-5.0
- dotnet-templates-3.0
- dotnet-templates-3.1
- dotnet-templates-5.0
- dotnet5.0-build-reference-packages
- dptfextract
- drpm
- drpm-devel
- dump
- dvd+rw-tools
- dyninst-static
- eclipse-ecf
- eclipse-ecf-core
- eclipse-ecf-runtime
- eclipse-emf
- eclipse-emf-core
- eclipse-emf-runtime
- eclipse-emf-xsd
- eclipse-equinox-osi
- eclipse-jdt
- eclipse-license
- eclipse-p2-discovery
- eclipse-pde
- eclipse-platform
- eclipse-swt

- ed25519-java
- ee4j-parent
- elfutils-devel-static
- elfutils-libelf-devel-static
- enca
- enca-devel
- environment-modules-compat
- evince-browser-plugin
- exec-maven-plugin
- farstream02
- felix-gogo-command
- felix-gogo-runtime
- felix-gogo-shell
- felix-scr
- felix-osgi-compendium
- felix-osgi-core
- felix-osgi-foundation
- felix-parent
- file-roller
- fipscheck
- fipscheck-devel
- fipscheck-lib
- firewire
- fonts-tweak-tool
- forge-parent
- freeradius-mysql
- freeradius-perl
- freeradius-postgresql
- freeradius-rest

- freeradius-sqlite
- freeradius-unixODBC
- fuse-sshfs
- fusesource-pom
- future
- gamin
- gamin-devel
- gavl
- gcc-toolset-10
- gcc-toolset-10-annobin
- gcc-toolset-10-binutils
- gcc-toolset-10-binutils-devel
- gcc-toolset-10-build
- gcc-toolset-10-dwz
- gcc-toolset-10-dyninst
- gcc-toolset-10-dyninst-devel
- gcc-toolset-10-elfutils
- gcc-toolset-10-elfutils-debuginfod-client
- gcc-toolset-10-elfutils-debuginfod-client-devel
- gcc-toolset-10-elfutils-devel
- gcc-toolset-10-elfutils-libelf
- gcc-toolset-10-elfutils-libelf-devel
- gcc-toolset-10-elfutils-libs
- gcc-toolset-10-gcc
- gcc-toolset-10-gcc-c++
- gcc-toolset-10-gcc-gdb-plugin
- gcc-toolset-10-gcc-gfortran
- gcc-toolset-10-gdb
- gcc-toolset-10-gdb-doc

- gcc-toolset-10-gdb-gdbserver
- gcc-toolset-10-libasan-devel
- gcc-toolset-10-libatomic-devel
- gcc-toolset-10-libitm-devel
- gcc-toolset-10-liblsan-devel
- gcc-toolset-10-libquadmath-devel
- gcc-toolset-10-libstdc++-devel
- gcc-toolset-10-libstdc++-docs
- gcc-toolset-10-libtsan-devel
- gcc-toolset-10-libubsan-devel
- gcc-toolset-10-ltrace
- gcc-toolset-10-make
- gcc-toolset-10-make-devel
- gcc-toolset-10-perftools
- gcc-toolset-10-runtime
- gcc-toolset-10-strace
- gcc-toolset-10-systemtap
- gcc-toolset-10-systemtap-client
- gcc-toolset-10-systemtap-devel
- gcc-toolset-10-systemtap-initscript
- gcc-toolset-10-systemtap-runtime
- gcc-toolset-10-systemtap-sdt-devel
- gcc-toolset-10-systemtap-server
- gcc-toolset-10-toolchain
- gcc-toolset-10-valgrind
- gcc-toolset-10-valgrind-devel
- gcc-toolset-9
- gcc-toolset-9-annobin
- gcc-toolset-9-build

- gcc-toolset-9-perftools
- gcc-toolset-9-runtime
- gcc-toolset-9-toolchain
- gcc-toolset-11-make-devel
- GConf2
- GConf2-devel
- gegl
- genisoimage
- genwqe-tools
- genwqe-vpd
- genwqe-zlib
- genwqe-zlib-devel
- geoipupdate
- geronimo-annotation
- geronimo-jms
- geronimo-jpa
- geronimo-parent-poms
- gfbgraph
- gflags
- gflags-devel
- glassfish-annotation-api
- glassfish-el
- glassfish-fastinfoset
- glassfish-jaxb-core
- glassfish-jaxb-txw2
- glassfish-jsp
- glassfish-jsp-api
- glassfish-legal
- glassfish-master-pom

- glassfish-servlet-api
- glew-devel
- glib2-fam
- glog
- glog-devel
- gmock
- gmock-devel
- gnome-abrt
- gnome-boxes
- gnome-menus-devel
- gnome-online-miners
- gnome-shell-extension-disable-screenshield
- gnome-shell-extension-horizontal-workspaces
- gnome-shell-extension-no-hot-corner
- gnome-shell-extension-window-grouper
- gnome-themes-standard
- gnu-free-fonts-common
- gnu-free-mono-fonts
- gnu-free-sans-fonts
- gnu-free-serif-fonts
- gnupg2-smime
- gnuplot
- gnuplot-common
- gobject-introspection-devel
- google-gson
- google-noto-sans-syriac-eastern-fonts
- google-noto-sans-syriac-estrangela-fonts
- google-noto-sans-syriac-western-fonts
- google-noto-sans-tibetan-fonts

- google-noto-sans-ui-fonts
- gphoto2
- gsl-devel
- gssntlmssp
- gtest
- gtest-devel
- gtkmm24
- gtkmm24-devel
- gtkmm24-docs
- gtksourceview3
- gtksourceview3-devel
- gtkspell
- gtkspell-devel
- gtkspell3
- guile
- gutenprint-gimp
- gutenprint-libs-ui
- gvfs-afc
- gvfs-afp
- gvfs-archive
- hamcrest-core
- hawtjni
- hawtjni
- hawtjni-runtime
- HdrHistogram
- HdrHistogram-javadoc
- highlight-gui
- hivex-devel
- hostname

- hplip-gui
- httpcomponents-project
- hwloc-plugins
- hyphen-fo
- hyphen-grc
- hyphen-hsb
- hyphen-ia
- hyphen-is
- hyphen-ku
- hyphen-mi
- hyphen-mn
- hyphen-sa
- hyphen-tk
- ibus-sayura
- icedax
- icu4j
- idm-console-framework
- iptables
- ipython
- isl
- isl-devel
- isorelax
- istack-commons-runtime
- istack-commons-tools
- iwl3945-firmware
- iwl4965-firmware
- iwl6000-firmware
- jacoco
- jaf

- jaf-javadoc
- jakarta-oro
- janino
- jansi-native
- jarjar
- java-1.8.0-ibm
- java-1.8.0-ibm-demo
- java-1.8.0-ibm-devel
- java-1.8.0-ibm-headless
- java-1.8.0-ibm-jdbc
- java-1.8.0-ibm-plugin
- java-1.8.0-ibm-src
- java-1.8.0-ibm-webstart
- java-1.8.0-openjdk-accessibility
- java-1.8.0-openjdk-accessibility-slowdebug
- java\_cup
- java-atk-wrapper
- javacc
- javacc-maven-plugin
- javaewah
- javaparser
- javapoet
- javassist
- javassist-javadoc
- jaxen
- jboss-annotations-1.2-api
- jboss-interceptors-1.2-api
- jboss-logmanager
- jboss-parent

- jctools
- jdepend
- jdependency
- jdom
- jdom2
- jetty
- jetty-continuation
- jetty-http
- jetty-io
- jetty-security
- jetty-server
- jetty-servlet
- jetty-util
- jffi
- jflex
- jgit
- jline
- jmc
- jnr-netdb
- jolokia-jvm-agent
- js-uglify
- jsch
- json\_simple
- jss-javadoc
- jtidy
- junit5
- jvnet-parent
- jzlib
- kernel-cross-headers

- ksc
- kurdit-unikurd-web-fonts
- kyotocabinet-libs
- ldapjdk-javadoc
- lensfun
- lensfun-devel
- lftp-scripts
- libaec
- libaec-devel
- libappindicator-gtk3
- libappindicator-gtk3-devel
- libatomic-static
- libavc1394
- libblocksruntime
- libcacard
- libcacard-devel
- libcgroup
- libcgroup-tools
- libchamplain
- libchamplain-devel
- libchamplain-gtk
- libcroco
- libcroco-devel
- libcxl
- libcxl-devel
- libdap
- libdap-devel
- libdazzle-devel
- libdbusmenu

- libdbusmenu-devel
- libdbusmenu-doc
- libdbusmenu-gtk3
- libdbusmenu-gtk3-devel
- libdc1394
- libdnet
- libdnet-devel
- libdv
- libdwarf
- libdwarf-devel
- libdwarf-static
- libdwarf-tools
- libeasyfc
- libeasyfc-gobject
- libepubgen-devel
- libertas-sd8686-firmware
- libertas-usb8388-firmware
- libertas-usb8388-olpc-firmware
- libgdither
- libGLEW
- libgovirt
- libguestfs-benchmarking
- libguestfs-devel
- libguestfs-gfs2
- libguestfs-gobject
- libguestfs-gobject-devel
- libguestfs-java
- libguestfs-java-devel
- libguestfs-javadoc

- libguestfs-man-pages-ja
- libguestfs-man-pages-uk
- libguestfs-tools
- libguestfs-tools-c
- libhugetlbfs
- libhugetlbfs-devel
- libhugetlbfs-utils
- libIDL
- libIDL-devel
- libidn
- libiec61883
- libindicator-gtk3
- libindicator-gtk3-devel
- libiscsi-devel
- libjose-devel
- libkkc
- libkkc-common
- libkkc-data
- libldb-devel
- liblogging
- libluksmeta-devel
- libmalaga
- libmcpp
- libmemcached
- libmemcached-libs
- libmetalink
- libmodulemd1
- libmongocrypt
- libmtp-devel

- libmusicbrainz5
- libmusicbrainz5-devel
- libnbd-devel
- liboauth
- liboauth-devel
- libpfm-static
- libpng12
- libpurple
- libpurple-devel
- libraw1394
- libreport-plugin-mailx
- libreport-plugin-rhtsupport
- libreport-plugin-ureport
- libreport-rhel
- libreport-rhel-bugzilla
- librpmem
- librpmem-debug
- librpmem-devel
- libsass
- libsass-devel
- libselinux-python
- libsqlite3x
- libtalloc-devel
- libtar
- libtdb-devel
- libtevent-devel
- libtpms-devel
- libunwind
- libusal

- libvarlink
- libverto-libevent
- libvirt-admin
- libvirt-bash-completion
- libvirt-daemon-driver-storage-gluster
- libvirt-daemon-driver-storage-iscsi-direct
- libvirt-devel
- libvirt-docs
- libvirt-gconfig
- libvirt-gobject
- libvirt-lock-sanlock
- libvirt-wireshark
- libvmem
- libvmem-debug
- libvmem-devel
- libvmmalloc
- libvmmalloc-debug
- libvmmalloc-devel
- libvncserver
- libwinpr-devel
- libwmf
- libwmf-devel
- libwmf-lite
- libXNVCtrl
- libyami
- log4j12
- log4j12-javadoc
- lohit-malayalam-fonts
- lohit-nepali-fonts

- `lorax-composer`
- `lua-guestfs`
- `lucene`
- `lucene-analysis`
- `lucene-analyzers-smartcn`
- `lucene-queries`
- `lucene-queryparser`
- `lucene-sandbox`
- `lz4-java`
- `lz4-java-javadoc`
- `mailman`
- `mailx`
- `make-devel`
- `malaga`
- `malaga-suomi-voikko`
- `marisa`
- `maven-antrun-plugin`
- `maven-assembly-plugin`
- `maven-clean-plugin`
- `maven-dependency-analyzer`
- `maven-dependency-plugin`
- `maven-doxia`
- `maven-doxia-sitetools`
- `maven-install-plugin`
- `maven-invoker`
- `maven-invoker-plugin`
- `maven-parent`
- `maven-plugins-pom`
- `maven-reporting-api`

- maven-reporting-impl
- maven-resolver-api
- maven-resolver-connector-basic
- maven-resolver-impl
- maven-resolver-spi
- maven-resolver-transport-wagon
- maven-resolver-util
- maven-scm
- maven-script-interpreter
- maven-shade-plugin
- maven-shared
- maven-verifier
- maven-wagon-file
- maven-wagon-http
- maven-wagon-http-shared
- maven-wagon-provider-api
- maven2
- meanwhile
- mercurial
- mercurial-hgk
- metis
- metis-devel
- mingw32-bzip2
- mingw32-bzip2-static
- mingw32-cairo
- mingw32-expat
- mingw32-fontconfig
- mingw32-freetype
- mingw32-freetype-static

- mingw32-gstreamer1
- mingw32-harfbuzz
- mingw32-harfbuzz-static
- mingw32-icu
- mingw32-libjpeg-turbo
- mingw32-libjpeg-turbo-static
- mingw32-libpng
- mingw32-libpng-static
- mingw32-libtiff
- mingw32-libtiff-static
- mingw32-openssl
- mingw32-readline
- mingw32-sqlite
- mingw32-sqlite-static
- mingw64-adwaita-icon-theme
- mingw64-bzip2
- mingw64-bzip2-static
- mingw64-cairo
- mingw64-expat
- mingw64-fontconfig
- mingw64-freetype
- mingw64-freetype-static
- mingw64-gstreamer1
- mingw64-harfbuzz
- mingw64-harfbuzz-static
- mingw64-icu
- mingw64-libjpeg-turbo
- mingw64-libjpeg-turbo-static
- mingw64-libpng

- mingw64-libpng-static
- mingw64-libtiff
- mingw64-libtiff-static
- mingw64-nettle
- mingw64-openssl
- mingw64-readline
- mingw64-sqlite
- mingw64-sqlite-static
- modello
- mojo-parent
- mongo-c-driver
- mousetweaks
- mozjs52
- mozjs52-devel
- mozjs60
- mozjs60-devel
- mozvoikko
- msv-javadoc
- msv-manual
- munge-maven-plugin
- mythes-mi
- mythes-ne
- nafees-web-naskh-fonts
- nbd
- nbdkit-devel
- nbdkit-example-plugins
- nbdkit-gzip-plugin
- nbdkit-plugin-python-common
- nbdkit-plugin-vddk

- ncompress
- ncurses-compat-libs
- net-tools
- netcf
- netcf-devel
- netcf-libs
- network-scripts
- network-scripts-ppp
- nkf
- nodejs-devel
- nodejs-packaging
- nss\_nis
- nss-pam-ldapd
- objectweb-asm
- objectweb-asm-javadoc
- objectweb-pom
- ocaml-bisect-ppx
- ocaml-camlp4
- ocaml-camlp4-devel
- ocaml-lwt
- ocaml-mmap
- ocaml-ocplib-endian
- ocaml-ounit
- ocaml-result
- ocaml-seq
- opencryptoki-tpmtok
- opencv-contrib
- opencv-core
- opencv-devel

- openhpi
- openhpi-libs
- OpenIPMI-perl
- openssh-cavs
- openssh-ldap
- openssl-ibmpkcs11
- opentest4j
- os-maven-plugin
- pakchois
- pandoc
- paps-libs
- paranamer
- parfait
- parfait-examples
- parfait-javadoc
- pcp-parfait-agent
- pcp-pmda-rpm
- pcp-pmda-vmware
- pcsc-lite-doc
- peripety
- perl-B-Debug
- perl-B-Lint
- perl-Class-Factory-Util
- perl-Class-ISA
- perl-DateTime-Format-HTTP
- perl-DateTime-Format-Mail
- perl-File-CheckTree
- perl-homedir
- perl-libxml-perl

- perl-Locale-Codes
- perl-Mozilla-LDAP
- perl-NKF
- perl-Object-HashBase-tools
- perl-Package-DeprecationManager
- perl-Pod-LaTeX
- perl-Pod-Plainer
- perl-prefork
- perl-String-CRC32
- perl-SUPER
- perl-Sys-Virt
- perl-tests
- perl-YAML-Syck
- phodav
- php-recode
- php-xmlrpc
- pidgin
- pidgin-devel
- pidgin-sipe
- pinentry-emacs
- pinentry-gtk
- pipewire0.2-devel
- pipewire0.2-libs
- platform-python-coverage
- plexus-ant-factory
- plexus-bsh-factory
- plexus-cli
- plexus-component-api
- plexus-component-factories-pom

- plexus-components-pom
- plexus-i18n
- plexus-interactivity
- plexus-pom
- plexus-velocity
- plymouth-plugin-throbgress
- pmreorder
- postgresql-test-rpm-macros
- powermock
- prometheus-jmx-exporter
- prometheus-jmx-exporter-openjdk11
- ptscotch-mpich
- ptscotch-mpich-devel
- ptscotch-mpich-devel-parmetis
- ptscotch-openmpi
- ptscotch-openmpi-devel
- purple-sipe
- pygobject2-doc
- pygtk2
- pygtk2-codegen
- pygtk2-devel
- pygtk2-doc
- python-nose-docs
- python-nss-doc
- python-podman-api
- python-psycopg2-doc
- python-pymongo-doc
- python-redis
- python-schedutils

- python-slip
- python-sqlalchemy-doc
- python-varlink
- python-virtualenv-doc
- python2-backports
- python2-backports-ssl\_match\_hostname
- python2-bson
- python2-coverage
- python2-docs
- python2-docs-info
- python2-funcsigs
- python2-ipaddress
- python2-mock
- python2-nose
- python2-numpy-doc
- python2-psycopg2-debug
- python2-psycopg2-tests
- python2-pymongo
- python2-pymongo-gridfs
- python2-pytest-mock
- python2-sqlalchemy
- python2-tools
- python2-virtualenv
- python3-bson
- python3-click
- python3-coverage
- python3-cpio
- python3-custodia
- python3-docs

- python3-flask
- python3-gevent
- python3-gobject-base
- python3-hivex
- python3-html5lib
- python3-hypothesis
- python3-ipatests
- python3-itsdangerous
- python3-jwt
- python3-libguestfs
- python3-mock
- python3-networkx-core
- python3-nose
- python3-nss
- python3-openipmi
- python3-pillow
- python3-ptyprocess
- python3-pydbus
- python3-pymongo
- python3-pymongo-gridfs
- python3-pyOpenSSL
- python3-pytoml
- python3-reportlab
- python3-schedutils
- python3-scons
- python3-semantic\_version
- python3-slip
- python3-slip-dbus
- python3-sqlalchemy

- python3-syspurpose
- python3-virtualenv
- python3-webencodings
- python3-werkzeug
- python38-asn1crypto
- python38-numpy-doc
- python38-psycopg2-doc
- python38-psycopg2-tests
- python39-numpy-doc
- python39-psycopg2-doc
- python39-psycopg2-tests
- qemu-kvm-block-gluster
- qemu-kvm-block-iscsi
- qemu-kvm-block-ssh
- qemu-kvm-hw-usbredir
- qemu-kvm-tests
- qpdf
- qpdf-doc
- qpidd-proton
- qrencode
- qrencode-devel
- qrencode-libs
- qt5-qtcanvas3d
- qt5-qtcanvas3d-examples
- rarian
- rarian-compat
- re2c
- recode
- redhat-lsb

- redhat-lsb-core
- redhat-lsb-cxx
- redhat-lsb-desktop
- redhat-lsb-languages
- redhat-lsb-printing
- redhat-lsb-submod-multimedia
- redhat-lsb-submod-security
- redhat-lsb-supplemental
- redhat-lsb-trialuse
- redhat-menus
- redhat-support-lib-python
- redhat-support-tool
- reflections
- regexp
- relaxngDatatype
- rhsm-gtk
- rpm-plugin-priorreset
- rpmemd
- rsyslog-udp spoof
- ruby-hivex
- ruby-libguestfs
- rubygem-abrt
- rubygem-abrt-doc
- rubygem-bson
- rubygem-bson-doc
- rubygem-bundler-doc
- rubygem-mongo
- rubygem-mongo-doc
- rubygem-net-telnet

- rubygem-xmlrpc
- s390utils-cmsfs
- samba-pidl
- samba-test
- samba-test-libs
- samyak-devanagari-fonts
- samyak-fonts-common
- samyak-gujarati-fonts
- samyak-malayalam-fonts
- samyak-odia-fonts
- samyak-tamil-fonts
- sane-frontends
- sanlk-reset
- sat4j
- scala
- scotch
- scotch-devel
- SDL\_sound
- selinux-policy-minimum
- sendmail
- sgabios
- sgabios-bin
- shrinkwrap
- sisu-inject
- sisu-mojos
- sisu-plexus
- skkdic
- SLOF
- smc-anjalioldlipi-fonts

- smc-dyuthi-fonts
- smc-fonts-common
- smc-kalyani-fonts
- smc-raghmalayalam-fonts
- smc-suruma-fonts
- softsm-devel
- sonatype-oss-parent
- sonatype-plugins-parent
- sos-collector
- sparsehash-devel
- spax
- spec-version-maven-plugin
- spice
- spice-client-win-x64
- spice-client-win-x86
- spice-glib
- spice-glib-devel
- spice-gtk
- spice-gtk-tools
- spice-gtk3
- spice-gtk3-devel
- spice-gtk3-vala
- spice-parent
- spice-protocol
- spice-qxl-wddm-dod
- spice-server
- spice-server-devel
- spice-qxl-xddm
- spice-server

- spice-streaming-agent
- spice-vdagent-win-x64
- spice-vdagent-win-x86
- sssd-libwbclient
- star
- stax-ex
- stax2-api
- stringtemplate
- stringtemplate4
- subscription-manager-initial-setup-addon
- subscription-manager-migration
- subscription-manager-migration-data
- subversion-javahl
- SuperLU
- SuperLU-devel
- supermin-devel
- swig
- swig-doc
- swig-gdb
- swtpm-devel
- swtpm-tools-pkcs11
- system-storage-manager
- tcl-brlapi
- testng
- tibetan-machine-uni-fonts
- timedatex
- tpm-quote-tools
- tpm-tools
- tpm-tools-pkcs11

- treelayout
- trousers
- trousers-lib
- tuned-profiles-compat
- tuned-profiles-nfv-host-bin
- tuned-utils-systemtap
- tycho
- uglify-js
- unbound-devel
- univocity-output-tester
- univocity-parsers
- usbguard-notifier
- usbredir-devel
- utf8cpp
- uthash
- velocity
- vinagre
- vino
- virt-dib
- virt-p2v-maker
- vm-dump-metrics-devel
- weld-parent
- wodim
- woodstox-core
- wqy-microhei-fonts
- wqy-unibit-fonts
- xdelta
- xmlgraphics-commons
- xmlstreambuffer

- xinetd
- xorg-x11-apps
- xorg-x11-drv-qxl
- xorg-x11-server-Xspice
- xpp3
- xsane-gimp
- xsom
- xz-java
- xz-java-javadoc
- yajl-devel
- yp-tools
- ypbind
- ypserv

## 10.20. 非推奨のデバイスおよび非保守のデバイス

このセクションは、

- RHEL 8 のライフサイクルが終了するまで継続してサポートされるデバイス (ドライバー、アダプター) を説明しますが、本製品の今後のメジャーリリースではサポートされない可能性が高いため、新たに実装することは推奨されません。記載以外のデバイスのサポートは変更しません。これは **非推奨** デバイスです。
- RHEL 8 では入手可能ですが、ルーチンベースでのテストや更新は行われていません。Red Hat は、独自の判断でセキュリティーバグなどの深刻なバグを修正する場合があります。このようなデバイスは実稼働環境では使用しなくなり、次のメジャーリリースでは無効になる可能性が高くなります。これは **未管理** デバイスです。

PCI デバイス ID は、**vendor:device:subvendor:subdevice** の形式です。デバイス ID が記載されていない場合は、対応するドライバーに関連するすべてのデバイスが非推奨になっています。ご使用のシステムでハードウェアの PCI ID を確認するには、**lspci -nn** コマンドを実行します。

表10.1 非推奨のデバイス

デバイス ID	ドライバー	デバイス名
	bnx2	QLogic BCM5706/5708/5709/5716 Driver
	hpsa	Hewlett-Packard Company: Smart アレイコントローラー
0x10df:0x0724	lpfc	Emulex Corporation: OneConnect FCoE Initiator (Skyhawk)

デバイス ID	ドライ バー	デバイス名
0x10df:0xe200	lpfc	Emulex Corporation: LPe15000/LPe16000 Series 8Gb/16Gb Fibre Channel Adapter
0x10df:0xf011	lpfc	Emulex Corporation: Saturn: LightPulse Fibre Channel Host Adapter
0x10df:0xf015	lpfc	Emulex Corporation: Saturn: LightPulse Fibre Channel Host Adapter
0x10df:0xf100	lpfc	Emulex Corporation: LPe12000 Series 8Gb Fibre Channel Adapter
0x10df:0xfc40	lpfc	Emulex Corporation: Saturn-X: LightPulse Fibre Channel Host Adapter
0x10df:0xe220	be2net	Emulex Corporation: OneConnect NIC (Lancer)
0x1000:0x005b	megaraid_sas	Broadcom / LSI: MegaRAID SAS 2208 [Thunderbolt]
0x1000:0x006E	mpt3sas	Broadcom / LSI: SAS2308 PCI-Express Fusion-MPT SAS-2
0x1000:0x0080	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0081	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0082	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0083	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0084	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0085	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0086	mpt3sas	Broadcom / LSI: SAS2308 PCI-Express Fusion-MPT SAS-2
0x1000:0x0087	mpt3sas	Broadcom / LSI: SAS2308 PCI-Express Fusion-MPT SAS-2
	myri10ge	Myricom 10G driver (10GbE)
	netxen_nic	QLogic/NetXen (1/10) GbE Intelligent Ethernet Driver

デバイス ID	ドライ バー	デバイス名
0x1077:0x2031	qla2xxx	QLogic Corp.: ISP8324-based 16Gb Fibre Channel to PCI Express Adapter
0x1077:0x2532	qla2xxx	QLogic Corp.: ISP2532-based 8Gb Fibre Channel to PCI Express HBA
0x1077:0x8031	qla2xxx	QLogic Corp.: 8300 Series 10GbE Converged Network Adapter (FCoE)
	qla3xxx	QLogic ISP3XXX ネットワークドライバー v2.03.00-k5
0x1924:0x0803	sfc	Solarflare Communications: SFC9020 10G Ethernet Controller
0x1924:0x0813	sfc	Solarflare Communications: SFL9021 10GBASE-T Ethernet Controller
	Soft- RoCE (rdma_r xe)	
	HNS- RoCE	HNS GE/10GE/25GE/50GE/100GE RDMA Network Controller
	liquidio	Cavium LiquidIO Intelligent Server Adapter Driver
	liquidio_ vf	Cavium LiquidIO Intelligent Server Adapter Virtual Function Driver

表10.2 未管理デバイス

デバイス ID	ドライ バー	デバイス名
	e1000	Intel® PRO/1000 ネットワークドライバー
	mptbase	Fusion MPT SAS ホストドライバー
	mptsas	Fusion MPT SAS ホストドライバー
	mptscsi h	Fusion MPT SCSI ホストドライバー
	mptspi	Fusion MPT SAS ホストドライバー

デバイス ID	ドライ バー	デバイス名
0x1000:0x0071 <sup>[a]</sup>	megarai d_sas	Broadcom / LSI: MR SAS HBA 2004
0x1000:0x0073 <sup>[a]</sup>	megarai d_sas	Broadcom / LSI: MegaRAID SAS 2008 [Falcon]
0x1000:0x0079 <sup>[a]</sup>	megarai d_sas	Broadcom / LSI: MegaRAID SAS 2108 [Liberator]
	nvmet_t cp	NVMe/TCP ターゲットドライバー

[a] RHEL 8.0 で無効になり、顧客の要求により RHEL 8.4 で再度有効になりました。

## 第11章 既知の問題

このパートでは、Red Hat EnterpriseLinux 8.7 の既知の問題について説明します。

### 11.1. インストーラーおよびイメージの作成

#### LPAR およびセキュアブートが有効になっている IBM Power 10 システムでのインストールが失敗します

RHEL インストーラーは、IBM Power 10 システムの静的キーセキュアブートと統合されていません。したがって、セキュアブートオプションを使用して論理パーティション (LPAR) を有効にすると、インストールに失敗し、**Unable to proceed with RHEL-x.x Installation** というエラーが表示されます。

この問題を回避するには、セキュアブートを有効にせずに RHEL をインストールします。システムを起動したら、以下を行います。

1. **dd** コマンドを使用して、署名されたカーネルを PReP パーティションにコピーします。
2. システムを再起動し、セキュアブートを有効にします。

ファームウェアがブートローダーとカーネルを検証すると、システムは正常に起動します。

詳細については、<https://www.ibm.com/support/pages/node/6528884> を参照してください。

(BZ#2025814)

#### Anaconda がアプリケーションとして実行されているシステムでの予期しない SELinux ポリシー

Anaconda がすでにインストールされているシステムでアプリケーションとして実行されている場合 (たとえば、**-image anaconda** オプションを使用してイメージファイルに別のインストールを実行する場合)、システムはインストール中に SELinux のタイプと属性を変更することを禁止されていません。そのため、SELinux ポリシーの特定の要素は、Anaconda が実行されているシステムで変更される可能性があります。この問題を回避するには、実稼働システムで Anaconda を実行せず、一時的な仮想マシンで実行します。そうすることで、実稼働システムの SELinux ポリシーは変更されません。**boot.iso** や **dvd.iso** からのインストールなど、システムインストールプロセスの一部として anaconda を実行しても、この問題の影響は受けません。

(BZ#2050140)

#### キックスタートコマンドの **auth** および **authconfig** で AppStream リポジトリが必要になる

インストール中に、キックスタートコマンドの **auth** および **authconfig** で **authselect-compat** パッケージが必要になります。**auth** または **authconfig** を使用したときに、このパッケージがないとインストールに失敗します。ただし、設計上、**authselect-compat** パッケージは AppStream リポジトリでしか利用できません。

この問題を回避するには、BaseOS リポジトリおよび AppStream リポジトリがインストーラーで利用できることを確認するか、インストール中にキックスタートコマンドの **authselect** コマンドを使用します。

(BZ#1640697)

#### **reboot --kexec** コマンドおよび **inst.kexec** コマンドが、予測可能なシステム状態を提供しない

キックスタートコマンド **reboot --kexec** またはカーネル起動パラメーター **inst.kexec** で RHEL インストールを実行しても、システムの状態が完全な再起動と同じになるわけではありません。これにより、

システムを再起動せずにインストール済みのシステムに切り替えると、予期しない結果が発生することがあります。

**kexec** 機能は非推奨になり、Red Hat Enterprise Linux の今後のリリースで削除されることに注意してください。

(BZ#1697896)

### USB CD-ROM ドライブが Anaconda のインストールソースとして利用できない

USB CD-ROM ドライブがソースで、キックスタート **ignoredisk --only-use=** コマンドを指定すると、インストールに失敗します。この場合、Anaconda はこのソースディスクを見つけ、使用できません。

この問題を回避するには、**harddrive --partition=sdX --dir=/** コマンドを使用して USB CD-ROM ドライブからインストールします。その結果、インストールは失敗しなくなりました。

(BZ#1914955)

### インストールプログラムでは、ネットワークアクセスがデフォルトで有効になっていない

一部のインストール機能、たとえば、コンテンツ配信ネットワーク (CDN) を使用したシステムの登録、NTP サーバーサポート、およびネットワークインストールソースなどには、ネットワークアクセスが必要です。ただし、ネットワークアクセスはデフォルトでは有効になっていません。そのためこの機能は、ネットワークアクセスが有効になるまで使用できません。

この問題を回避するには、インストールの開始時にネットワークアクセスを有効にする起動オプション **ip=dhcp** を追加します。オプションで、起動オプションを使用して、ネットワーク上にあるキックスタートファイルまたはリポジトリを渡しても、問題が解決されます。結果として、ネットワークベースのインストール機能を使用できます。

(BZ#1757877)

### iso9660 ファイルシステムで、ハードドライブがパーティション分割されたインストールが失敗する

ハードドライブが **iso9660** ファイルシステムでパーティションが設定されているシステムには、RHEL をインストールできません。これは、**iso9660** ファイルシステムパーティションを含むハードディスクを無視するように設定されている、更新されたインストールコードが原因です。これは、RHEL が DVD を使用せずにインストールされている場合でも発生します。

この問題を回避するには、インストールの開始前に、キックスタートファイルに次のスクリプトを追加して、ディスクをフォーマットします。

メモ: 回避策を実行する前に、ディスクで利用可能なデータのバックアップを作成します。**wipefs** は、ディスク内の全データをフォーマットします。

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

その結果、インストールでエラーが発生することなく、想定どおりに機能します。

(BZ#1929105)

### HASH MMU モードの IBM 電源システムが、メモリー割り当ての障害で起動できない

**HASH** メモリー割り当てユニット (MMU) モードの IBM Power Systems は、最大 192 コアの **kdump** に対応します。そのため、**kdump** が 192 コア以上で有効になっていると、メモリー割り当て失敗が原因

でシステムの起動が失敗します。この制限は、**HASH MMU** モードの起動初期段階での RMA メモリーの割り当てによるものです。この問題を回避するには、**kdump** を使用する代わりに、**fadump** を有効にした **Radix MMU** モードを使用します。

(BZ#2028361)

### **rpm-ostree** ペイロードをインストールすると、RHEL for Edge インストーラーイメージがマウントポイントの作成に失敗する

RHEL for Edge インストーラーイメージなどで使用される **rpm-ostree** ペイロードをデプロイする場合、インストーラーはカスタムパーティションの一部のマウントポイントを適切に作成しません。その結果、インストールは以下のエラーで中止されます。

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

この問題を回避するには、以下を実行します。

- 自動パーティション設定スキームを使用し、手動でマウントポイントを追加しないでください。
- マウントポイントは、**/var** ディレクトリー内のみを手動で割り当てます。たとえば、**/var/my-mount-point** や、**/boot**、**/var** などの標準ディレクトリーです。

その結果、インストールプロセスは正常に終了します。

(BZ#2126506)

### **composer-cli compose start** の **--size** パラメーターは、値を MiB ではなくバイトとして扱います

**composer-cli compose start --size size\_value blueprint\_name image\_type** コマンドを使用する場合、**--size** パラメーターはそのパラメーターを MiB 形式で使用する必要があります。ただし、設定のバグにより、**composer-cli** ツールはこのパラメーターをバイト単位として扱います。

この問題を回避するには、サイズの値に 1048576 を掛けます。または、`size_value` をブループリントに追加します。カスタマイズにより、ファイルシステムをより細かく制御でき、MiB や GiB などの単位を受け入れます。[サポートされているイメージのカスタマイズ](#) を参照してください。

(BZ#2033192)

## 11.2. サブスクリプションの管理

### **syspurpose addons** は **subscription-manager attach --auto** 出力に影響しません。

Red Hat Enterprise Linux 8 では、**syspurpose** コマンドラインツールの 4 つの属性 (**role**、**usage**、**service\_level\_agreement**、および **addons**) が追加されました。現在、**role**、**usage**、および **service\_level\_agreement** のみが、**subscription-manager attach --auto** コマンドの実行の出力に影響します。**addons** 引数に値を設定しても、自動登録されたサブスクリプションには影響がありません。

(BZ#1687900)

## 11.3. ソフトウェア管理

### **cr\_compress\_file\_with\_stat()** がメモリーリークを引き起こす可能性がある

`createrepo_c` ライブラリーには API `cr_compress_file_with_stat()` 関数があります。この関数は、`char **dst` を 2 番目のパラメーターとして宣言します。他のパラメーターによって、`cr_compress_file_with_stat()` は、入力パラメーターとして `dst` を使用するか、割り当てられた文字列を返すために使用します。`dst` の内容をいつ解放するかユーザーに通知しないため、この予測できない動作によりメモリーリークが発生する可能性があります。

この問題を回避するために、`dst` パラメーターを入力としてのみ使用する新しい API `cr_compress_file_with_stat_v2` 関数が追加されました。これは `char *dst` として宣言されます。これにより、メモリーリークが回避されます。

`cr_compress_file_with_stat_v2` 関数は一時的で、RHEL 8 のみに存在することに注意してください。後で、`cr_compress_file_with_stat()` が代わりに修正されます。

(BZ#1973588)

### スクリプトレットが失敗したときに成功したと報告された YUM トランザクション

RPM バージョン 4.6 以降、インストール後のスクリプトレットは、トランザクションに致命的な影響を与えることなく失敗することが許可されています。この動作は YUM まで伝播します。これにより、スクリプトレットが作成され、パッケージトランザクション全体が成功したと報告されているときに失敗することがあります。

現在利用できる回避策はありません。

これは、RPM と YUM の間で一貫性を保つことが期待される動作であることに注意してください。スクリプトレットの問題は、パッケージレベルで対処する必要があります。

(BZ#1986657)

### アップグレードによってアーキテクチャーを変更するパッケージのセキュリティー YUM アップグレードが失敗する

[BZ#2088149](#) アドバイザリーでリリースされた [RHBA-2022:7711](#) のパッチでは、次のリグレッションが導入されました。セキュリティーフィルターを使用した YUM アップグレードは、アップグレードによってアーキテクチャが `noarch` に (逆もまた然り) 変更されたパッケージでは失敗します。その結果、システムが脆弱な状態になる可能性があります。

この問題を回避するには、セキュリティーフィルターを使用せずに通常のアップグレードを実行します。

(BZ#2088149)

## 11.4. シェルおよびコマンドラインツール

### ipmitool は特定のサーバプラットフォームと互換性がありません

`ipmitool` ユーティリティーは、Intelligent Platform Management Interface (IPMI) をサポートするデバイスの監視、設定、および管理に役立ちます。現在のバージョンの `ipmitool` は、以前の Cipher Suite 3 の代わりに Cipher Suite 17 をデフォルトで使用します。その結果、`ipmitool` は、ネゴシエーション中に Cipher Suite 17 のサポートを発表しましたが、実際にはこの暗号スイートをサポートしていない特定のベアメタルノードとの通信に失敗します。その結果、`ipmitool` は、`no matching cipher suite` エラーメッセージで異常終了します。

詳細は、関連する [ナレッジベースの記事](#) を参照してください。

この問題を解決するには、ベースボード管理コントローラー (BMC) ファームウェアを更新して、Cipher Suite 17 を使用します。

オプションで、BMC ファームウェアの更新が利用できない場合は、**ipmitool** に特定の暗号スイートを強制的に使用させることで、この問題を回避できます。**ipmitool** で管理タスクを呼び出す場合は、使用する暗号スイートの番号とともに **ipmitool** コマンドに **-C** オプションを追加します。以下の例を参照してください。

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

([BZ#1873614](#))

## 復元にクリーンディスクを使用しないと、ReaR がボリュームグループの再作成に失敗する

既存のデータを含むディスクに復元する場合、ReaR は復元の実行に失敗します。

この問題を回避するには、ディスクが以前に使用されていた場合、復元する前にディスクを手動でワイプします。レスキュー環境でディスクをワイプするには、**rear recover** コマンドを実行する前に、次のいずれかのコマンドを使用します。

- ディスクを上書きする **dd** コマンド。
- 使用可能なすべてのメタデータを消去するには、**-a** フラグを指定した **wipefs** コマンド。

**/dev/sda** ディスクからメタデータをワイプする次の例を参照してください。

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

このコマンドは、最初に **/dev/sda** のパーティションからメタデータをワイプし、次にパーティションテーブル自体をワイプします。

([BZ#1925531](#))

## coreutils は、誤解を招く EPERM エラーコードを報告することがあります。

**statx()** システムコールを使用して、GNU コアユーティリティ (**coreutils**) が起動しました。**seccomp** フィルターが、不明なシステムコールに対して EPERM エラーコードを返す場合、EPERM は動作中の **statx()** の syscall が返す実際の **Operation not permitted** エラーと区別できないため、**coreutils** は、誤解を招く EPERM エラーコードを報告します。

この問題を回避するには、**seccomp** フィルターを更新して、**statx()** の syscall を許可するか、不明の syscall の ENOSYS エラーコードを返すようにします。

([BZ#2030661](#))

## 11.5. インフラストラクチャーサービス

### FIPS モードの Postfix TLS フィンガープリントアルゴリズムを SHA-256 に変更する必要があります。

RHEL 8 のデフォルトでは、**postfix** は後方互換性に TLS を使用する MD5 フィンガープリントを使用します。ただし、FIPS モードでは、MD5 ハッシュ関数が利用できないため、デフォルトの postfix 設定で TLS が誤って機能する可能性があります。この問題を回避するには、postfix 設定ファイルのハッシュ関数を SHA-256 に変更する必要があります。

詳細は、関連するナレッジベースの記事 [Fix postfix TLS in the FIPS mode by switch to SHA-256 instead of the MD5](#) を参照してください。

([BZ#1711885](#))

## --delete と --filter '-x string.\*' オプションを一緒に使用すると rsync が失敗する

ファイルを転送および同期するための **rsync** ユーティリティーは、RHEL 8 の拡張属性を正しく処理できません。したがって、拡張属性の **--filter '-x string.\*'** オプションとともに **--delete** オプションを **rsync** コマンドに渡し、システム上のファイルが正規表現を満たしている場合は、プロトコルの非互換性を示すエラーが発生します。たとえば、**--filter '-x system.\*'** オプションを使用すると、フィルターはシステムに存在する **system.mwmrc** ファイルを検出し、**rsync** は失敗します。**--filter '-x system.\*'** オプションを使用した後に発生する次のエラーメッセージを参照してください。

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

この問題を回避するには、拡張属性に正規表現を慎重に使用してください。

([BZ#2139118](#))

## brltty パッケージは multilib 対応ではない

**brltty** パッケージの 32 ビット版と 64 ビット版の両方をインストールすることはできません。32 ビット版 (**brltty.i686**) または 64 ビット版 (**brltty.x86\_64**) いずれかのパッケージをインストールすることができます。64 ビット版を推奨します。

([BZ#2008197](#))

## 11.6. セキュリティー

### /etc/passwd- のファイル権限が CIS RHEL 8 Benchmark 1.0.0 と合致しない

CIS Benchmark の問題により、**/etc/passwd-** バックアップファイルの権限を保証する SCAP ルールの修正によって、権限が **0644** に設定されます。ただし、**CIS Red Hat Enterprise Linux 8 Benchmark 1.0.0** では、そのファイルに対するファイルパーミッション **0600** が必要です。そのため、修正後、**/etc/passwd-** のファイル権限はベンチマークに合うように設定されません。

([BZ#1858866](#))

### libselinux-python は、そのモジュールからのみ利用可能

**libselinux-python** パッケージには、SELinux アプリケーション開発用の Python 2 バインディングのみが含まれ、後方互換性に使用されます。このため、**yum install libselinux-python** コマンドを使用すると、デフォルトの RHEL 8 リポジトリで **libselinux-python** コマンドを利用できなくなりました。

この問題を回避するには、**libselinux-python** モジュールおよび **python27** モジュールの両方を有効にし、以下のコマンドで **libselinux-python** パッケージとその依存関係をインストールします。

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

または、1つのコマンドでインストールプロファイルを使用して **libselinux-python** をインストールします。

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

これにより、各モジュールを使用して **libselinux-python** をインストールできます。

(BZ#1666328)

**udica は、`--env container=podman` で開始したときにのみ UBI 8 コンテナを処理します。**

Red Hat Universal Base Image 8 (UBI 8) コンテナは、**podman** の値ではなく、**コンテナ環境変数** を **oci** 値に設定します。これにより、**udica** ツールがコンテナ JavaScript Object Notation (JSON) ファイルを分析しなくなります。

この問題を回避するには、**`--env container=podman`** パラメーターを指定して、**podman** コマンドで UBI 8 コンテナを起動します。そのため、**udica** は、上記の回避策を使用している場合に限り、UBI 8 コンテナの SELinux ポリシーを生成することができます。

(BZ#1763210)

**`/etc/selinux/config` の `SELINUX=disabled` が正常に動作しません。**

`/etc/selinux/config` で `SELINUX=disabled` オプションを使用して SELinux を無効にすると、カーネルが SELinux を有効にして起動し、その後のブートプロセスで無効化モードに切り替わります。これにより、メモリーリークが生じる可能性があります。

この問題を回避するには、SELinux を完全に無効にする必要がある場合に [SELinux の使用のシステムの起動時に SELinux モードの変更](#) で説明されているように、**`selinux=0`** パラメーターをカーネルコマンドラインに追加して SELinux を無効にすることが推奨されます。

(JIRA:RHELPLAN-34199)

**sshd -T が、暗号、MAC、および KeX アルゴリズムに関する不正確な情報を提供する**

`sshd -T` コマンドの出力には、システム全体の暗号化ポリシー設定や、`/etc/sysconfig/sshd` 内の環境ファイルから取得でき、`sshd` コマンドの引数として適用されるその他のオプションは含まれていません。これは、アップストリームの OpenSSH プロジェクトが RHEL8 で Red-Hat が提供する暗号化のデフォルトをサポートするための Include ディレクティブをサポートしていなかったために発生します。暗号化ポリシーは、**EnvironmentFile** を使用してサービスを開始するときに、**sshd.service** ユニットの `sshd` 実行可能ファイルにコマンドライン引数として適用されます。この問題を回避するには、`sshd -T $CRYPTO_POLICY` のように、環境ファイルで `source` コマンドを使用し、暗号化ポリシーを引数として `sshd` コマンドに渡します。詳細については、[暗号、MAC、または KeX アルゴリズムが sshd -T とは異なり、現在の暗号ポリシーレベルで提供されるものとは異なる](#) を参照してください。その結果、`sshd -T` からの出力は、現在設定されている暗号化ポリシーと一致します。

(BZ#2044354)

**FIPS モードの OpenSSL が、特定の D-H パラメーターのみを受け入れます。**

FIPS モードでは、OpenSSL を使用する TLS クライアントは `bad dh value` エラーを返し、手動で生成されたパラメーターを使用するようにサーバーへの TLS 接続を中止します。これは、FIPS 140-2 に準拠するよう設定されている場合、OpenSSL が NIST SP 800-56A rev3 付録 D (RFC 3526 で定義されたグループ 14、15、16、17、18、および RFC 7919 で定義されたグループ) に準拠した Diffie-Hellman パラメーターでのみ機能するためです。また、OpenSSL を使用するサーバーは、その他のパラメーターをすべて無視し、代わりに同様のサイズの既知のパラメーターを選択します。この問題を回避するには、準拠するグループのみを使用します。

(BZ#1810911)

**crypto-policies が Camellia 暗号を誤って許可する。**

RHEL 8 システム全体の暗号化ポリシーでは、製品ドキュメントで説明されているように、すべてのポリシーレベルで Camellia 暗号を無効にする必要があります。ただし、Kerberos プロトコルでは、デフォルトでこの Camellia 暗号が有効になります。

この問題を回避するには、**NO-CAMELLIA** サブポリシーを適用します。

```
# update-crypto-policies --set DEFAULT:NO-CAMELLIA
```

これまでに上記のコマンドで、**DEFAULT** から切り替えたことがある場合は、**DEFAULT** を暗号化レベルの名前に置き換えます。

その結果、この回避策を使用して Camellia 暗号を無効にしている場合に限り、システム全体の暗号化ポリシーを使用する全ポリシーで、この暗号化を適切に拒否できます。

(BZ#1919155)

### OpenSC pkcs15-init によるスマートカードのプロビジョニングプロセスが適切に動作しない

**file\_caching** オプションは、デフォルトの OpenSC 設定で有効になっているため、キャッシュ機能は **pkcs15-init** ツールから一部のコマンドを適切に処理しません。したがって、OpenSC を使用したスマートカードのプロビジョニングプロセスは失敗します。

この問題を回避するには、以下のスニペットを `/etc/opensc.conf` ファイルに追加します。

```
app pkcs15-init {
    framework pkcs15 {
        use_file_caching = false;
    }
}
```

**pkcs15-init** を使用したスマートカードのプロビジョニングは、前述の回避策を適用している場合に限り機能します。

(BZ#1947025)

### SHA-1 署名を使用するサーバーへの接続が GnuTLS で動作しない

証明書の SHA-1 署名は、GuTLS セキュアな通信ライブラリーにより、セキュアでないものとして拒否されます。したがって、TLS のバックエンドとして GnuTLS を使用するアプリケーションは、このような証明書を提供するピアへの TLS 接続を確立することができません。この動作は、その他のシステム暗号化ライブラリーと一貫性がありません。

この問題を回避するには、サーバーをアップグレードして、SHA-256 または強力なハッシュを使用して署名した証明書を使用するか、LEGACY ポリシーに切り替えます。

(BZ#1628553)

### IKE over TCP 接続がカスタム TCP ポートで機能しない

**tcp-remoteport** Libreswan 設定オプションが適切に動作しません。したがって、デフォルト以外の TCP ポートを指定する必要があるシナリオでは、IKE over TCP 接続を確立することができません。

(BZ#1989050)

### インストール中にシステムを強化すると、RHV ハイパーバイザーが正しく動作しないことがある

Red Hat Virtualization Hypervisor (RHV-H) をインストールし、Red Hat Enterprise Linux 8 STIG プロファイルを適用すると、OSCAP Anaconda Add-on が RVH-H ではなく RHEL としてシステムを強化し、RHV-H の必須パッケージを削除する場合があります。その結果、RHV ハイパーバイザーが機能し

ない場合があります。この問題を回避するには、プロファイルの強化を適用せずに RHV-H システムをインストールし、インストールが完了したら、OpenSCAP を使用してプロファイルを適用します。その結果、RHV ハイパーバイザーは正しく動作します。

(BZ#2075508)

## Red Hat は、CVE OVAL レポートを圧縮形式で提供する

Red Hat は **bzip2-compressed** 形式で CVE OVAL フィードを提供し、XML ファイル形式では使用できなくなりました。RHEL 8 のフィードの場所は、この変更を反映するために適宜更新されています。圧縮されたコンテンツの参照は標準化されていないため、サードパーティーの SCAP スキャナーでは、フィードを使用するスキャンルールで問題が発生する可能性があります。

(BZ#2028428)

## SSG における相互依存ルールの特定のセットが失敗する可能性がある。

ルールとその依存関係の順序付けを定義しないため、ベンチマークの **SCAP Security Guide (SSG)** ルールの修正が失敗する可能性があります。たとえば、特定の順番で複数のルールを実行する必要がある場合、あるルールがコンポーネントをインストールし、別のルールが同じコンポーネントを設定した場合すると、それらは正しくない順序で実行される可能性があり、修正によってエラーが報告されます。この問題を回避するには、修正を再実行して、番目の実行で依存ルールを修正します。

(BZ#1750755)

## CIS Server プロファイルを使用すると、Server with GUI および Workstation をインストールできない

CIS Server Level 1 および Level 2 のセキュリティープロファイルは、**Server with GUI** および **Workstation** ソフトウェアの選択と互換性がありません。そのため、**Server with GUI** ソフトウェアの選択と CIS プロファイルを使用して RHEL 8 をインストールすることはできません。CIS Server Level 1 または Level 2 プロファイルと、これらのソフトウェアの選択のいずれかを使用したインストール試行では、エラーメッセージが生成されます。

```
package xorg-x11-server-common has been added to the list of excluded packages, but it can't be removed from the current software selection without breaking the installation.
```

CIS ベンチマークに従ってシステムを **Server with GUI** または **Workstation** のソフトウェア選択に合わせる必要がある場合は、代わりに CIS Workstation Level 1 または Level 2 プロファイルを使用してください。

(BZ#1843932)

## RHEL 8 のキックスタートが、com\_redhat\_oscaped の代わりに org\_fedora\_oscaped を使用

キックスタートは、**com\_redhat\_oscaped** ではなく、**org\_fedora\_oscaped** として Open Security Content Automation Protocol (OSCAP) Anaconda アドオンを参照します。これが、混乱を招く可能性があります。これは、Red Hat Enterprise Linux 7 との下位互換性のために必要です。

(BZ#1665082)

## STIG プロファイルの SSH タイムアウトルールが誤ったオプションを設定している

OpenSSH の更新は、次の米国国防情報システム局のセキュリティー技術実装ガイド (DISA STIG) プロファイルのルールに影響を与えました。

- RHEL 8 用 DISA STIG (**xccdf\_org.ssgproject.content\_profile\_stig**)

- RHEL 8 用の GUI を備えた DISA STIG ([xccdf\\_org.ssgproject.content\\_profile\\_stig\\_gui](#))

これらの各プロファイルでは、次の2つのルールが影響を受けます。

Title: Set SSH Client Alive Count Max to zero  
CCE Identifier: CCE-83405-1  
Rule ID: xccdf\_org.ssgproject.content\_rule\_sshd\_set\_keepalive\_0  
STIG ID: RHEL-08-010200

Title: Set SSH Idle Timeout Interval  
CCE Identifier: CCE-80906-1  
Rule ID: xccdf\_org.ssgproject.content\_rule\_sshd\_set\_idle\_timeout  
STIG ID: RHEL-08-010201

SSH サーバーに適用すると、これらの各ルールは、以前のように動作しなくなったオプション (**ClientAliveCountMax** および **ClientAliveInterval**) を設定します。その結果、OpenSSH は、これらのルールで設定されたタイムアウトに達したときに、アイドル状態の SSH ユーザーを切断しなくなりました。回避策として、これらのルールは、ソリューションが開発されるまで、DISA STIG for RHEL8 および DISA STIG with GUI for RHEL8 プロファイルから一時的に削除されました。

([BZ#2038977](#))

### 特定の監査ルールの Bash 修正が正しく機能しない

SCAP セキュリティーガイド (SSG) 次の SCAP ルールの Bash 修復では、Audit キーが追加されません。

- **audit\_rules\_login\_events**
- **audit\_rules\_login\_events\_faillock**
- **audit\_rules\_login\_events\_lastlog**
- **audit\_rules\_login\_events\_tallylog**
- **audit\_rules\_usergroup\_modification**
- **audit\_rules\_usergroup\_modification\_group**
- **audit\_rules\_usergroup\_modification\_gshadow**
- **audit\_rules\_usergroup\_modification\_opasswd**
- **audit\_rules\_usergroup\_modification\_passwd**
- **audit\_rules\_usergroup\_modification\_shadow**
- **audit\_rules\_time\_watch\_localtime**
- **audit\_rules\_mac\_modification**
- **audit\_rules\_networkconfig\_modification**
- **audit\_rules\_sysadmin\_actions**
- **audit\_rules\_session\_events**

- `audit_rules_sudoers`
- `audit_rules_sudoers_d`

その結果、修復スクリプトは修復されたルールのアクセスビットとパスを修正しますが、監査キーのないルールは OVAL チェックに準拠しません。したがって、そのようなルールの修復後のスキャンは **FAIL** を報告します。この問題を回避するには、影響を受けるルールに手動でキーを追加します。

(BZ#2119356)

**特定の rsyslog 優先度の文字列が正常に動作しません。**

`imtcp` に GnuTLS 優先度文字列を設定して、完成していない暗号化をきめ細かく制御できるようになりました。したがって、`rsyslog` では、以下の優先文字列が正常に動作しません。

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

この問題を回避するには、正しく機能する優先度文字列のみを使用します。

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

したがって、現在の設定は、正しく機能する文字列に限定する必要があります。

(BZ#1679512)

**デフォルトのロギング設定がパフォーマンスに与える悪影響**

デフォルトのログ環境設定は、メモリーを 4 GB 以上使用する可能性があり、`rsyslog` で `systemd-journald` を実行している場合は、速度制限値の調整が複雑になります。

詳細は、ナレッジベースの記事 [Negative effects of the RHEL default logging setup on performance and their mitigations](#) を参照してください。

(JIRA:RHELPLAN-10431)

**キックスタートインストール時のサービス関連のルールの修正が失敗する場合があります。**

キックスタートのインストール時に、OpenSCAP ユーティリティーで、サービス **enable** または **disable** 状態の修正が必要でないことが誤って表示されることがあります。これにより、OpenSCAP が、インストール済みシステムのサービスを非準拠状態に設定する可能性があります。回避策として、キックスタートインストール後にシステムをスキャンして修復できます。これにより、サービス関連の問題が修正されます。

(BZ#1834716)

## 11.7. ネットワーク

**NetworkManager は、特定の順番でボンドおよびチームポートをアクティベートしない**

NetworkManager は、インターフェイス名のアルファベット順にインターフェイスをアクティブにします。ただし、起動中にインターフェイスが後で表示される場合、たとえば、カーネルがインターフェイスを検出するのにより多くの時間が必要な場合、NetworkManager は後でこのインターフェイスをアクティブにします。NetworkManager は、ボンドおよびチームポートの優先順位の設定に対応していません。したがって、NetworkManager がこのデバイスのポートをアクティブにする順番は、常に予測できるとは限りません。この問題を回避するには、ディスパッチャスクリプトを作成します。

このようなスクリプトの例は、チケット内の対応する [コメント](#) を参照してください。

([BZ#1920398](#))

### nm-cloud-setup サービスは、手動で設定されたセカンダリー IP アドレスをインターフェイスから削除する

クラウド環境から受け取った情報に基づいて、**nm-cloud-setup** サービスがネットワークインターフェイスを設定します。インターフェイスを手動で設定するには、**nm-cloud-setup** を無効にします。ただし、場合によっては、ホスト上の他のサービスもインターフェイスを設定できます。たとえば、これらのサービスはセカンダリー IP アドレスを追加できます。**nm-cloud-setup** がセカンダリー IP アドレスを削除しないようにするには、

1. **nm-cloud-setup** サービスおよびタイマーを停止して無効にします。

```
# systemctl disable --now nm-cloud-setup.service nm-cloud-setup.timer
```

2. 使用可能な接続プロファイルを表示します。

```
# nmcli connection show
```

3. 影響を受ける接続プロファイルを再アクティブ化します。

```
# nmcli connection up "<profile_name>"
```

その結果、このサービスは、手動で設定されたセカンダリー IP アドレスをインターフェイスから削除しなくなりました。

([BZ#2132754](#))

### IPv6\_rpfilter オプションが有効になっているシステムでネットワークスループットが低下

**firewalld.conf** ファイルで **IPv6\_rpfilter** オプションが有効になっているシステムでは、100-Gbps リンクなどの高いトラフィックシナリオの場合、現時点でパフォーマンスは最適ではなくネットワークスループットが低下します。この問題を回避するには、**IPv6\_rpfilter** オプションを無効にします。これを行うには、**/etc/firewalld/firewalld.conf** ファイルに次の行を追加します。

```
IPv6_rpfilter=no
```

その結果、システムはパフォーマンスが向上しますが、同時にセキュリティーは低下します。

([BZ#1871860](#))

### ネットワークインターフェイス名の予期しない変更により、IBM Z の RoCE インターフェイスの IP 設定が失われる

RHEL 8.6 以前では、IBM Z プラットフォーム上で、**udev** デバイスマネージャーが、一意の識別子 (UID) によって列挙される RoCE インターフェイスに、予測できないデバイス名を割り当てます。一方、RHEL 8.7 以降では、**udev** はこれらのインターフェイスに、**eno** 接頭辞が付いた予測可能なデバイス名を割り当てます。

RHEL 8.6 以前から 8.7 以降に更新すると、これらの UID 列挙インターフェイスには新しい名前が付けられ、NetworkManager 接続プロファイルのデバイス名と一致しなくなります。したがって、更新後、これらのインターフェイスには IP 設定がありません。

更新前に適用できる回避策と、すでにシステムを更新している場合の修正については、[RoCE interfaces on IBM Z lose their IP settings after updating to RHEL 8.7 or later](#) を参照してください。

(BZ#2169382)

## 11.8. カーネル

### IBM Power Systems のセキュアブートは移行をサポートしていません

現在、IBM Power Systems では、物理ボリューム (PV) の移行が成功した後、論理パーティション (LPAR) が起動しません。その結果、パーティションでセキュアブートが有効になっているタイプの自動移行は失敗します。

(BZ#2126777)

### page\_poison=1 を使用すると、カーネルクラッシュが発生する可能性がある

EFI 実装に問題のあるファームウェアでカーネルパラメーターとして **page\_poison=1** を使用すると、オペレーティングシステムが原因でカーネルがクラッシュする可能性があります。デフォルトでは、このオプションは無効になっており、特に実稼働システムでは有効にすることは推奨しません。

(BZ#2050411)

### kmod の weak-modules がモジュールの相互依存関係で機能しない

**kmod** パッケージによって提供される **weak-modules** スクリプトは、どのモジュールがインストールされたカーネルと kABI 互換であるかを判別します。ただし、モジュールのカーネル互換性をチェックしている間、**weak-modules** はモジュールシンボルの依存関係を、それらがビルドされたカーネルの上位リリースから下位リリースへと処理します。結果として、異なるカーネルリリースに対して構築された相互依存関係を持つモジュールは互換性がないと解釈される可能性があるため、**weak-modules** はこのシナリオでは機能しません。

この問題を回避するには、新しいカーネルをインストールする前に、最新のストックカーネルに対して追加のモジュールをビルドまたは配置します。

(BZ#2103605)

### 同一の crash 拡張機能を再読み込みすると、セグメンテーションフォルトが発生する可能性がある

読み込み済みのクラッシュ拡張ファイルのコピーを読み込むと、セグメンテーションフォルトが発生する場合があります。現在、crash ユーティリティは、元のファイルが読み込まれているかどうかを検出します。その結果、crash ユーティリティに同一のファイルが2つ共存するため、名前空間コリジョンが発生し、クラッシュユーティリティが起動してセグメンテーションフォルトが発生します。

この問題を回避するには、クラッシュ拡張ファイルを一度だけ読み込みます。その結果、セグメンテーションフォルトは上記のシナリオでは発生しなくなりました。

(BZ#1906482)

### vmcore キャプチャーが、メモリーのホットプラグまたはアンプラグの操作を実行した後に失敗する

メモリーのホットプラグまたはホットアンプラグ操作の実行後に、メモリーのレイアウト情報を含むデバイスツリーを更新するとイベントが発生します。これにより、**makedumpfile** ユーティリティは存在しない物理アドレスにアクセスしようとします。以下の条件を満たすと問題が発生します。

- IBM Power System (little endian) で RHEL 8 を実行する。

- システムで **kdump** サービスまたは **fadump** サービスが有効になっている。

このような場合に、メモリーホットプラグまたはホットアンプラグの操作後にカーネルクラッシュが発生すると、カーネルのキャプチャーで **vmcore** の保存に失敗します。

この問題を回避するには、ホットプラグまたはホットアンプラグ後に **kdump** サービスを再起動します。

```
# systemctl restart kdump.service
```

これにより、上記のシナリオで **vmcore** が正常に保存されます。

(BZ#1793389)

### RHEL 8 で、デバッグカーネルがクラッシュキャプチャー環境で起動に失敗する

デバッグカーネルはメモリーを大量に消費するので、デバッグカーネルが使用中で、カーネルパニックが発生すると、問題が発生します。その結果、デバッグカーネルはキャプチャーカーネルとして起動できず、代わりにスタクトレースが生成されます。この問題を回避するには、必要に応じてクラッシュカーネルメモリーを増やします。これにより、デバッグカーネルが、クラッシュキャプチャー環境で正常に起動します。

(BZ#1659609)

### 起動時にクラッシュカーネルメモリーの割り当てに失敗する

一部の Ampere Altra システムでは、BIOS 設定で 32 ビットリージョンが無効になっていると、起動時にクラッシュカーネルメモリーを割り当てることに失敗します。したがって、**kdump** サービスが起動できません。これは、クラッシュカーネルメモリーを含むのに十分な大きさのフラグメントがない場合に、4 GB 未満のリージョンのメモリーの断片化によって生じます。

この問題を回避するには、以下のように BIOS で 32 ビットのメモリーリージョンを有効にします。

1. システムで BIOS 設定を開きます。
2. **Chipset** メニューを開きます。
3. **Memory Configuration** で、**Slave 32-bit** オプションを有効にします。

これにより、32 ビットリージョン内のクラッシュカーネルメモリー割り当てに成功し、**kdump** サービスが期待どおりに機能します。

(BZ#1940674)

### QAT マネージャーが LKCF のスペアデバイスを残さない

Intel® QuickAssist Technology(QAT) マネージャー (**qatmgr**) はユーザー空間プロセスであり、デフォルトではシステム内のすべての QAT デバイスを使用します。これにより、Linux Kernel Cryptographic Framework(LKCF) には QAT デバイスが残っていません。この動作は予想され、大多数のユーザーはユーザースペースからのアクセラレーションを使用するため、この状況を回避する必要はありません。

(BZ#1920086)

カーネル ACPI ドライバーは、PCIe ECAM メモリーリージョンにアクセスできないことを報告します。

ファームウェアが提供する Advanced Configuration and Power Interface (ACPI) テーブルは、PCI バスデバイスの現在のリソース設定 (\_CRS) メソッドにおいて PCI バス上のメモリーリージョンを定義しません。したがって、システムの起動時に以下の警告メッセージが表示されます。

```
[ 2.817152] acpi PNP0A08:00: [Firmware Bug]: ECAM area [mem 0x30000000-0x31ffffff] not reserved in ACPI namespace
[ 2.827911] acpi PNP0A08:00: ECAM at [mem 0x30000000-0x31ffffff] for [bus 00-1f]
```

ただし、カーネルは依然として **0x30000000-0x31ffffff** メモリーリージョンにアクセスできます。また、そのメモリーリージョンを PCI Enhanced Configuration Access Mechanism (ECAM) に適切に割り当てることができます。以下の出力で 256 バイトオフセットで PCIe 設定領域にアクセスして、PCI ECAM が正常に機能することを確認できます。

```
03:00.0 Non-Volatile memory controller: Sandisk Corp WD Black 2018/PC SN720 NVMe SSD (prog-if 02 [NVM Express])
...
Capabilities: [900 v1] L1 PM Substates
    L1SubCap: PCI-PM_L1.2- PCI-PM_L1.1- ASPM_L1.2+ ASPM_L1.1- L1_PM_Substates+
              PortCommonModeRestoreTime=255us PortTPowerOnTime=10us
    L1SubCtl1: PCI-PM_L1.2- PCI-PM_L1.1- ASPM_L1.2- ASPM_L1.1-
              T_CommonMode=0us LTR1.2_Threshold=0ns
    L1SubCtl2: T_PwrOn=10us
```

これにより、警告メッセージを無視します。

問題の詳細は、[Firmware Bug: ECAM area mem 0x30000000-0x31ffffff not reserved in ACPI namespace" appears during system boot](#) を参照してください。

(BZ#1868526)

### tuned-adm profile powersave コマンドを使用すると、システムが応答しなくなる

**tuned-adm profile powersave** コマンドを実行すると、古い Thunderx (CN88xx) プロセッサを持つ Penguin Valkyrie 2000 2 ソケットシステムが応答しなくなります。これにより、作業を再開するためシステムを再起動することになります。この問題を回避するには、システムが上記の仕様と一致する場合には **powersave** プロファイルの使用を避けてください。

(BZ#1609288)

### HP NMI ウォッチドッグが常にクラッシュダンプを生成しない

特定の場合において、HP NMI ウォッチドッグの **hpwdt** ドライバーは、マスク不可割り込み (NMI) が **perfmon** ドライバーにより使用されたため、HPE ウォッチドッグタイマーが生成した NMI を要求できません。

欠落している NMI は、以下の 2 つの条件のいずれかによって開始されます。

1. Integrated Lights-Out (iLO) サーバー管理ソフトウェアの **NMI 生成** ボタン。このボタンはユーザーがトリガーします。
2. **hpwdt** ウォッチドッグ。デフォルトでは、有効期限により NMI がサーバーに送信されます。

通常、両方のシーケンスは、システムが応答しない場合に発生します。通常、これらの状況の NMI ハンドラーは **kernel panic()** 関数を呼び出します。また、設定されていれば、**kdump** サービスが **vmcore** ファイルを生成します。

ただし、NMI が見つからないため、**kernel panic()** は呼び出されず、**vmcore** が収集されません。

最初のケース (1.) でシステムが応答しない場合は、その状態のままになります。このシナリオを回避するには、仮想電源 ボタンを使用してサーバーをリセットするか、電源を切って入れ直します。

2 つ目のケース (2.) では、欠落している NMI が Automated System Recovery (ASR) からのリセットの後 9 秒後に続きます。

HPE Gen9 Server ラインでは、1 桁台の割合でこの問題が発生します。Gen10 の周波数がさらに小さくなる。

(BZ#1602962)

### irqpoll を使用すると vmcore の生成に失敗します。

アマゾンウェブサービス Graviton1 プロセッサ上で実行される 64 ビット ARM アーキテクチャー上の **nvme** ドライバーの既存の問題により、最初のカーネルに **irqpoll** カーネルコマンドラインパラメーターを指定すると、**vmcore** の生成が失敗します。したがって、カーネルクラッシュ時に **vmcore** が **/var/crash/** ディレクトリーにダンプされません。この問題を回避するには、以下を実行します。

1. **/etc/sysconfig/kdump** ファイルの **KDUMP\_COMMANDLINE\_REMOVE** 変数に **irqpoll** を追加します。

```
# KDUMP_COMMANDLINE_REMOVE="hugepages hugepagesz slub_debug quiet
log_buf_len swiotlb"
```

2. **/etc/sysconfig/kdump** ファイルの **KDUMP\_COMMANDLINE\_APPEND** 変数から **irqpoll** を削除します。

```
# KDUMP_COMMANDLINE_APPEND="irqpoll nr_cpus=1 reset_devices
cgroup_disable=memory udev.children-max=2 panic=10 swiotlb=noforce novmcoredd"
```

3. **kdump** サービスを再起動します。

```
# systemctl restart kdump
```

その結果、最初のカーネルが正常に起動し、カーネルクラッシュ時に **vmcore** がキャプチャーされることが予想されます。

Amazon Web Services Graviton 2 および Amazon Web Services Graviton 3 プロセッサでは、**/etc/sysconfig/kdump** ファイルの **irqpoll** パラメーターを手動で削除する必要がないことに注意してください。

**kdump** サービスは、大量のクラッシュカーネルメモリーを使用して **vmcore** ファイルをダンプする可能性があります。キャプチャーカーネルには、**kdump** サービス用のメモリーが十分あることを確認します。

この既知の問題の関連情報は、[irqpoll カーネルコマンドラインパラメーターにより、vmcore 生成エラーが発生する場合があります](#) を参照してください。

(BZ#1654962)

### 仮想マシンへの仮想機能の割り当て時に接続に失敗する

**ionic** デバイスドライバーを使用する Pensando ネットワークカードは、VLAN タグ設定要求を許可し、ネットワーク仮想機能 (VF) を VM に割り当てる間にネットワーク接続の設定を試行します。この機能はカードのファームウェアではサポートされていないため、このようなネットワーク接続は失敗します。

(BZ#1930576)

**OPEN MPI ライブラリーは、デフォルトの PML でランタイムが失敗する可能性があります。**

OPEN Message Passing Interface (OPEN MPI) 実装 4.0.x シリーズでは、UCX (Unified Communication X) がデフォルトの PML (ポイントツーポイント) です。OPEN MPI 4.0.x シリーズの新しいバージョンでは、**openib** Byte Transfer Layer (BTL) が非推奨になりました。

ただし、OPEN MPI は **同種** クラスタ (同じハードウェアおよびソフトウェア設定) で実行される場合も、UCX は MPI **openlib** の一方向操作に BTL を使用します。これにより、実行エラーが発生する可能性があります。この問題を回避するには、以下を実行します。

- 以下のパラメーターを使用して **mpirun** コマンドを実行します。

```
-mca btl openib -mca pml ucx -x UCX_NET_DEVICES=mlx5_ib0
```

詳細は以下のようになります。

- **-mca btl openib** パラメーターは **openib** BTL を無効にします。
- **-mca pml ucx** パラメーターは、**ucx** PML を使用するように OPEN MPI を設定します。
- **x UCX\_NET\_DEVICES=** パラメーターは、指定したデバイスを使用するように UCX を制限します。

OPEN MPI は、**異種** クラスタ (ハードウェアおよびソフトウェア設定に異なる) を実行する場合は、デフォルトの PML として UCX を使用します。これにより、OPEN MPI ジョブが不安定なパフォーマンス、応答しない動作で実行されたり、またはクラッシュによる不具合とともに実行される可能性があります。この問題を回避するには、UCX の優先度を以下のように設定します。

- 以下のパラメーターを使用して **mpirun** コマンドを実行します。

```
-mca pml_ucx_priority 5
```

これにより、OPEN MPI ライブラリーは、UCX を介して利用可能な別のトランスポート層を選択することができます。

(BZ#1866402)

**Solarflare が、最大数の VF (Virtual Function) の作成に失敗する**

Solarflare NIC は、リソースが十分でないため、最大数の VF の作成に失敗します。PCIe デバイスが作成できる VF の最大数は、`/sys/bus/pci/devices/PCI_ID/sriov_totalvfs` ファイルで確認できます。この問題を回避するには、起動時に **Solarflare Boot Manager** から、または Solarflare **sfboot** ユーティリティーの使用により、VF の数または VF MSI 割り込みの値を低い値に調整できます。デフォルトの VF MSI 割り込みの値は **8** です。

- **sfboot** を使用して VF MSI 割り込み値を調整するには、以下を実行します。

```
# sfboot vf-msix-limit=2
```

**注記**

VF MSI 割り込みの値を調整すると、VF のパフォーマンスに影響します。

調整されるパラメーターの詳細は、**Solarflare Server Adapter user guide** を参照してください。

(BZ#1971506)

### iwl7260-firmware により、Intel Wi-Fi 6 AX200、AX210、および Lenovo ThinkPad P1 Gen 4 で Wi-Fi が切断される

**iwl7260-firmware** または **iwl7260-wifi** ドライバーを RHEL 8.7 および/または RHEL 9.1 (およびそれ以降) によって提供されるバージョンに更新した後、ハードウェアが正しくない内部状態になります。状態を誤って報告します。その結果、Intel Wifi 6 カードが機能せず、次のエラーメッセージが表示される場合があります。

```
kernel: iwlwifi 0000:09:00.0: Failed to start RT ucode: -110
kernel: iwlwifi 0000:09:00.0: WRT: Collecting data: ini trigger 13 fired (delay=0ms)
kernel: iwlwifi 0000:09:00.0: Failed to run INIT ucode: -110
```

未確認の回避策は、システムの電源をオフにしてから再度オンにすることです。再起動しないでください。

(BZ#2106341)

### 64 ビット ARM アーキテクチャーで **kdump** のメモリー割り当てが失敗する

特定の 64 ビット ARM ベースのシステムでは、ファームウェアは、分散したさまざまな場所にメモリーをランダムに予約する不連続メモリー割り当て方法を使用します。その結果、連続したメモリーブロックが利用できないため、クラッシュカーネルは **kdump** メカニズム用のメモリー領域を予約できません。

この問題を回避するには、RHEL 8.8 以降で提供されるカーネルバージョンをインストールしてください。RHEL の最新バージョンは、説明されているシナリオで適切なメモリー領域を見つけるのに役立つ **fallback** ダンプキャプチャーメカニズムをサポートしています。

(BZ#2214235)

コア数が大きいシステムのリアルタイムカーネルのハードウェア認定では、ロックの競合を回避するために **skew\_tick=1** ブートパラメーターを渡す必要があります。

多数のソケットとコアカウントが大きい大規模なシステムまたは中規模のシステムでは、タイムキーピングシステムで使用される **xtime\_lock** のロック競合により、レイテンシーの急増が発生する可能性があります。その結果、レイテンシーの急増およびハードウェア認証のレイテンシーは、マルチプロセッシングシステムで発生する可能性があります。回避策として、**skew\_tick=1** ブートパラメーターを追加することで、CPU ごとにタイマーティックをオフセットし、別のタイミングで開始できます。

ロックの競合を回避するには、**skew\_tick=1** を有効にします。

1. **grubby** で **skew\_tick=1** パラメーターを有効にします。

```
# grubby --update-kernel=ALL --args="skew_tick=1"
```

2. 変更を有効にするために再起動します。
3. **cat /proc/cmdline** コマンドを実行して、新しい設定を確認します。

**skew\_tick=1** を有効にすると、消費電力が大幅に増加するため、レイテンシーの影響を受けるリアルタイムワークロードを実行している場合にのみ有効にする必要があります。

(BZ#2214508)

## 11.9. ブートローダー

### grubby の動作はドキュメントから逸脱している

**grubby** ツールを使用して新しいカーネルを追加し、引数を指定しない場合、**grubby** はデフォルトの引数を新しいエントリーに渡します。**--copy-default** 引数を渡さなくても、この動作が発生します。**--args** および **--copy-default** オプションを使用すると、これらの引数が、**汚い** ドキュメントに記載されているデフォルトの引数に追加されます。

ただし、**\$tuned\_params** などの追加の引数を追加すると、**--copy-default** オプションが呼び出されない限り、**grubby** ツールはこれらの引数を渡しません。

この状況では、次の 2 つの回避策があります。

- **root=** 引数を設定し、**--args** を空のままにします:

```
# grubby --add-kernel /boot/my_kernel --initrd /boot/my_initrd --args "root=/dev/mapper/rhel-root" --title "entry_with_root_set"
```

- または、**root=** 引数と指定された引数を設定しますが、デフォルトのものは設定しません:

```
# grubby --add-kernel /boot/my_kernel --initrd /boot/my_initrd --args "root=/dev/mapper/rhel-root some_args and_some_more" --title "entry_with_root_set_and_other_args_too"
```

([BZ#1900829](#))

## 11.10. ファイルシステムおよびストレージ

### LVM writecache の制限

**writecache** LVM キャッシュメソッドには以下の制限がありますが、**cache** メソッドには存在しません。

- **pvmove** コマンドを使用すると、**writecache** 論理ボリュームに名前を付けることはできません。
- **writecache** を指定した論理ボリュームは、シンプールまたは VDO と組み合わせて使用できません。

以下の制限は、**cache** メソッドにも適用されます。

- **cache** または **writecache** がアタッチされている間は、論理ボリュームのサイズを変更することはできません。

([JIRA:RHELPLAN-27987](#), [BZ#1798631](#), [BZ#1808012](#))

### XFS クォータ警告が頻繁にトリガーされる

クォータタイマーを使用すると、クォータ警告が頻繁にトリガーされるため、ソフトクォータが必要以上に速く実行されます。この問題を回避するには、警告のトリガーを妨げるソフトクォータを使用しないでください。その結果、警告メッセージの量はソフトクォータ制限を強制せず、設定されたタイムアウトを尊重するようになります。

([BZ#2059262](#))

### LUKS ボリュームを格納する LVM mirror デバイスが応答しなくなることがある

セグメントタイプが **mirror** のミラーリング LVM デバイスで LUKS ボリュームを格納すると、特定の条件下で応答しなくなる可能性があります。デバイスが応答しなくなると、すべての I/O 操作を拒否します。

耐障害性のソフトウェア定義ストレージに、LUKS ボリュームをスタックする必要がある場合に、この問題を回避するには、Red Hat はセグメントタイプが **mirror** ではなく **raid1** の LVM RAID 1 デバイスを使用することを推奨します。

**raid1** のセグメントタイプは、デフォルトの RAID 設定タイプで、**mirror** の代わりに、推奨のソリューションとしてこのタイプが使用されます。

**mirror** デバイスを **raid1** に変換するには、[ミラーリングされた LVM デバイスの RAID1 デバイスへの変換](#) を参照してください。

(BZ#1730502)

### /boot ファイルシステムを LVM に配置することができない

/boot ファイルシステムを LVM 論理ボリュームに配置することはできません。この制限は、以下の理由により存在します。

- EFI システムでは、**EFI システムパーティション** が従来の /boot ファイルシステムとして機能します。UEFI 標準では、特定の GPT パーティションタイプと、このパーティションの特定のファイルシステムタイプが必要です。
- RHEL 8 は、システムブートエントリーに **Boot Loader Specification (BLS)** を使用します。この仕様では、プラットフォームのファームウェアが /boot ファイルシステムを読み込める必要があります。EFI システムでは、プラットフォームファームウェアは UEFI 標準で定義された /boot 設定のみを読み取ることができます。
- GRUB 2 ブートローダーでの LVM 論理ボリュームに対するサポートは完全ではありません。Red Hat は、UEFI や BLS などの標準があるので、この機能のユースケース数が減少しているため、サポートを改善する予定はありません。

Red Hat では、LVM での /boot のサポートを提供する予定はありません。代わりに、Red Hat は、/boot ファイルシステムを LVM 論理ボリュームに配置する必要がないシステムスナップショットおよびロールバックを管理するツールを提供します。

(BZ#1496229)

### LVM で、複数のブロックサイズを持つボリュームグループが作成できない

**vgcreate** または **vgextend** などの LVM ユーティリティーでは、物理ボリューム (PV) の論理ブロックサイズが異なるボリュームグループ (VG) を作成できなくなりました。別のブロックサイズの PV で基礎となる論理ボリューム (LV) を拡張するとファイルシステムがマウントに失敗するため、LVM はこの変更を採用しました。

ブロックサイズが混在する VG の作成を再度有効にするには、**lvm.conf** ファイルの **allow\_mixed\_block\_sizes=1** オプションを設定します。

(BZ#1768536)

### blk-availability systemd サービスは、複雑なデバイススタックを非アクティブ化する

**systemd** では、デフォルトのブロック非アクティブ化コードは、仮想ブロックデバイスの複雑なスタックを常に正しく処理するとは限りません。一部の設定では、シャットダウン中に仮想デバイスが削除されない場合があり、エラーメッセージがログに記録されます。この問題を回避するには、次のコマンドを実行して、複雑なブロックデバイススタックを非アクティブ化します。

```
# systemctl enable --now blk-availability.service
```

その結果、複雑な仮想デバイススタックはシャットダウン中に正しく非アクティブ化され、エラーメッセージは生成されません。

(BZ#2011699)

## VDO ドライバーのバグにより、ジャーナルブロックを介してデバイスがフリーズする可能性があります

デバイスマッパーの **suspend** 操作を追跡しているときに、VDO ドライバーのバグにより、システムが一部のジャーナルブロックをメタデータの更新待ちとしてマークします。**suspend** 呼び出し以降、更新はすでに適用されています。

ジャーナルが同じ物理ブロックに戻ると、ブロックは使用できなくなります。最終的に、ブロックが再び使用可能になるまで、すべての書き込みが停止します。VDO デバイスでの **growPhysical**、**growLogical**、および **setWritePolicy** 操作には、一時停止/再開サイクルが含まれているため、多数のジャーナル更新後にデバイスがフリーズする可能性があります。

VDO プールまたはその上の論理ボリュームのサイズを増やすか、LVM ツールで管理されている VDO デバイスで **pvmove** および **lvchange** 操作を使用すると、この問題が発生する可能性があります。

回避策として、一時停止/再開サイクルを含む方法で VDO デバイスの設定を変更し、VDO デバイスを完全にシャットダウンしてから、再度起動します。これにより、メモリー内の不適切な状態がクリアされ、ジャーナルブロックがリセットされます。その結果、デバイスはフリーズしなくなり、正常に動作します。

(BZ#2109047)

## VDO ボリュームの起動中にソフトロックアップが原因でシステムがハングする

**pv\_mmu\_ops** 構造のカーネル ABI の破損 (修正中) により、カーネルバージョン **4.18.0-425.10.1.el8\_7** (RHEL-8.7.0.2-BaseOS) を搭載した RHEL 8.7 システムで、Virtual Data Optimizer (VDO) ボリュームの起動時に、ソフトロックアップが原因でハングまたはカーネルパニックが発生します。この問題を回避するには、**kernel-4.18.0-425.10.1.el8\_7** で起動する前に有効な VDO ボリュームを無効にして、システムのハングを防ぐか、以前のバージョンのカーネル (**4.18.0-425.3.1.el8**) にダウングレードして、VDO 機能を保持します。

(BZ#2158783)

## 11.11. 動的プログラミング言語、WEB サーバー、およびデータベースサーバー

### 32 ビットアプリケーションで呼び出されると **getpwnam()** が失敗する可能性がある

NIS のユーザーが **getpwnam()** 関数を呼び出す 32 ビットアプリケーションを使用する場合は、**nss\_nis.i686** パッケージがないと呼び出しに失敗します。この問題を回避するには、**yum install nss\_nis.i686** コマンドを使用して、不足しているパッケージを手動でインストールします。

(BZ#1803161)

### MariaDB では PAM プラグインバージョン 1.0 が機能しない

**MariaDB 10.3** は、PAM (Pluggable Authentication Modules) プラグインバージョン 1.0 を提供します。**MariaDB 10.5** は、プラグインバージョン 1.0 および 2.0 を提供します。バージョン 2.0 がデフォルトです。

RHEL 8 では、**MariaDB** PAM プラグインバージョン 1.0 は機能しません。この問題を回避するには、**mariadb:10.5** モジュールストリームによって提供される PAM プラグインバージョン 2.0 を使用します。

(BZ#1942330)

## OpenLDAP ライブラリー間のシンボルの競合により、**httpd** でクラッシュが発生することがある

OpenLDAP が提供する **libldap** ライブラリーと **libldap\_r** ライブラリーの両方が、単一のプロセス内にロードされ、使用されると、これらのライブラリー間でシンボルの競合が発生する可能性があります。そのため、**httpd** 設定によって **mod\_security** または **mod\_auth\_openidc** モジュールもロードされると、PHP **ldap** 拡張機能を使用する Apache **httpd** 子プロセスが突然終了する可能性があります。

Apache Portable Runtime (APR) ライブラリーに対する RHEL 8.3 の更新では、**APR\_DEEPBIND** 環境変数を設定することでこの問題を回避できます。これにより、**httpd** モジュールのロード時に **RTLD\_DEEPBIND** 動的リンカーオプションを使用できるようになります。**APR\_DEEPBIND** 環境変数を有効にすると、競合するライブラリーをロードする **httpd** 設定でクラッシュが発生しなくなります。

(BZ#1819607)

## 11.12. IDENTITY MANAGEMENT

**--agent-uid pkidbuser** オプションを指定して **cert-fix** ユーティリティーを使用すると、証明書システムが破損します。

**--agent-uid pkidbuser** オプションを指定して **cert-fix** ユーティリティーを使用すると、証明書システムの LDAP 設定が破損します。したがって、証明書システムは不安定になり、システムの復元に手動の操作が必要になる可能性があります。

(BZ#1729215)

IdM ホストの **/var/log/lastlog** 分析ファイルが、パフォーマンスの問題を引き起こす可能性があります。

IdM のインストール時に、利用できる合計 10,000 の範囲からの 200,000 の UID の範囲が無作為に選択され、割り当てられます。このようにランダムな範囲を選択すると、今後別の 2 つの IdM ドメインを統合する場合に、ID の競合が発生する可能性を大幅に削減できます。

ただし、UID が多いと、**/var/log/lastlog** ファイルで問題が発生する可能性があります。たとえば、1280000008 の UID を持つユーザーが IdM クライアントにログインすると、ローカルの **/var/log/lastlog** ファイルサイズは、約 400 GB に増えます。実際のファイルはスパースで、その領域をすべて使用しません。ただし、一部のアプリケーションはデフォルトではスパースファイルを識別するように設計されています。そのため、それら进行处理する特定のオプションが必要になる場合があります。たとえば、設定が複雑でバックアップ、コピーアプリケーションがスパースファイルを正しく処理しない場合、ファイルはサイズが 400 GB であるかのようにコピーされます。この動作により、パフォーマンスの問題が発生する可能性があります。

この問題を回避するには、以下を実行します。

- 標準パッケージの場合は、そのドキュメントを参照して、スパースファイル进行处理するオプションを特定します。
- カスタムアプリケーションの場合、**/var/log/lastlog** などのスパースファイルを正しく管理できることを確認してください。

(JIRA:RHELPLAN-59111)

FIPS モードは、共有シークレットを使用したフォレスト間の信頼を確立することをサポートしません。

NTLMSSP 認証は FIPS に準拠していないため、FIPS モードでフォレスト間の信頼を確立できません。この問題を回避するには、FIPS モードが有効な IdM ドメインと AD ドメインとの間に信頼を確立する際に、Active Directory (AD) 管理アカウントで認証します。

(BZ#1924707)

### FreeRADIUS サーバーが FIPS モードでの実行に失敗する

デフォルトでは、FIPS モードでは、OpenSSL は MD5 ダイジェストアルゴリズムの使用を無効にします。RADIUS プロトコルでは、RADIUS クライアントと RADIUS サーバー間のシークレットを暗号化するために MD5 が必要であるため、FreeRADIUS サーバーが FIPS モードで失敗します。

この問題を回避するには、以下の手順に従います。

#### 手順

1. **radiusd** サービスの環境変数 **RADIUS\_MD5\_FIPS\_OVERRIDE** を作成します。

```
systemctl edit radiusd

[Service]
Environment=RADIUS_MD5_FIPS_OVERRIDE=1
```

2. 変更を適用するには、**systemd** 設定を再読み込みし、**radiusd** サービスを開始します。

```
# systemctl daemon-reload
# systemctl start radiusd
```

3. デバッグモードで FreeRADIUS を実行するには、以下を実行します。

```
# RADIUS_MD5_FIPS_OVERRIDE=1 radiusd -X
```

FreeRADIUS は FIPS モードで実行できますが、FIPS モードでは弱い暗号と関数が使用されるため、これは FIPS に準拠するわけではありません。

FIPS モードでの FreeRADIUS 認証の設定方法は、[FIPS モードで FreeRADIUS 認証を設定する方法](#) を参照してください。

(BZ#1958979)

### IdM から AD へのレルム間の TGS 要求が失敗します

IdM Kerberos チケットの特権属性証明書 (PAC) 情報は、Active Directory (AD) でサポートされていない AES SHA-2 HMAC 暗号化で署名されるようになりました。

その結果、IdM から AD へのレルム間 TGS 要求 (双方向の信頼の設定) は、以下のエラーを出して失敗します。

```
"Generic error (see e-text) while getting credentials for <service principal>"
```

(BZ#2125182)

ドメイン SID の不一致により、移行した IdM ユーザーがログインできない可能性がある

**ipa migrate-ds** スクリプトを使用して IdM デプロイメントから別のデプロイメントにユーザーを移行する場合、そのユーザーの以前のセキュリティー識別子 (SID) には現在の IdM 環境のドメイン SID がいないため、ユーザーが IdM サービスを使用する際に問題が発生する可能性があります。たとえば、これらのユーザーは **kinit** ユーティリティーを使用して Kerberos チケットを取得できますが、ログインできません。この問題を回避するには、ナレッジベースの記事 [Migrated IdM users unable to log in due to mismatching domain SIDs](#) を参照してください。

(JIRA:RHELPLAN-109613)

### FIPS モードの IdM は、双方向のフォレスト間信頼を確立するための NTLMSSP プロトコルの使用をサポートしない

FIPS モードが有効な Active Directory (AD) と Identity Management (IdM) との間で双方向のフォレスト間の信頼を確立すると、New Technology LAN Manager Security Support Provider (NTLMSSP) 認証が FIPS に準拠していないため、失敗します。FIPS モードの IdM は、認証の試行時に AD ドメインコントローラーが使用する RC4 NTLM ハッシュを受け入れません。

(BZ#2120572)

### FIPS モードで IdM Vault 暗号化および復号化に失敗する

FIPS モードが有効な場合は、OpenSSL RSA-PKCS1v15 パディング暗号化がブロックされます。その結果、現在は IdM が PKCS1v15 パディングを使用してセッションキーをトランスポート証明書でラップするため、Identity Management (IdM) Vault が正しく機能しません。

(BZ#2122919)

### Samba をプリントサーバーとして実行し、RHEL 8.4 以前から更新する場合にアクションが必要です

今回の更新で、**samba** パッケージが `/var/spool/samba/` ディレクトリーを作成しなくなりました。プリントサーバーとして Samba を使用し、**[printers]** 共有の `/var/spool/samba/` を使用してプリントジョブをスプールすると、SELinux は Samba ユーザーがこのディレクトリーにファイルを作成しないようにします。したがって、印刷ジョブが失敗し、**auditd** サービスは `/var/log/audit/audit.log` に **denied** メッセージを記録します。8.4 以前からシステムを更新した後にこの問題を回避するには、以下を行います。

1. `/etc/samba/smb.conf` ファイルで **[printers]** 共有を探します。
2. 共有定義に `path = /var/spool/samba/` が含まれる場合は、設定を更新して、`path` パラメーターを `/var/tmp/` に設定します。
3. **smbd** サービスを再起動します。

```
# systemctl restart smbd
```

Samba を RHEL 8.5 以降に新しくインストールした場合、アクションは不要です。その場合、**samba-common** パッケージが提供するデフォルトの `/etc/samba/smb.conf` ファイルは、すでに `/var/tmp/` ディレクトリーを使用してプリントジョブをスプールします。

(BZ#2009213)

### バージョン 1.2.2 へのリベース後の authselect のダウングレードにより、システム認証の破損

**authselect** パッケージが、最新のアップストリームバージョン **1.2.2** にリベースされました。**authselect** のダウングレードはサポートされておらず、**root** を含むすべてのユーザーに対してシステム認証が破損しています。

**authselect** パッケージを **1.2.1** 以前にダウングレードした場合は、この問題を回避するために以下の手順を実行します。

1. GRUB ブート画面で、起動するカーネルのバージョンを含む **Red Hat Enterprise Linux** を選択し、**e** を押してエントリーを編集します。
2. **linux** で始まる行の末尾で、**single** を、別の単語で入力し、**Ctrl+X** を押して起動プロセスを開始します。
3. シングルユーザーモードでの起動時に、root パスワードを入力します。
4. 以下のコマンドを使用して **authselect** 設定を復元します。

```
# authselect select sssd --force
```

([BZ#1892761](#))

**NSS** で有効になっている暗号の **default** キーワードは、他の暗号と組み合わせても機能しません

Directory Server では、**default** キーワードを使用して、ネットワークセキュリティーサービス (NSS) で有効になっているデフォルトの暗号を参照することができます。しかし、コマンドラインまたは Web コンソールを使用してデフォルトの暗号および追加の暗号を有効にする場合、Directory Server は **default** キーワードの解決に失敗します。その結果、サーバーは追加で指定された暗号のみを有効にし、次のようなエラーをログに記録します。

```
Security Initialization - SSL alert: Failed to set SSL cipher preference information: invalid ciphers
<default,+cipher_name>: format is +cipher1,-cipher2... (Netscape Portable Runtime error 0 - no error)
```

回避策としては、追加で有効にしたいものも含めて、NSS でデフォルトで有効になっているすべての暗号を指定してください。

([BZ#1817505](#))

**RHEL 8.6** から **RHEL 8.7** への **pki-core-debuginfo** の更新が失敗する

**RHEL 8.6** から **RHEL 8.7** への **pki-core-debuginfo** パッケージの更新が失敗します。この問題を回避するには、以下のコマンドを実行します。

1. **yum remove pki-core-debuginfo**
2. **yum update -y**
3. **yum install pki-core-debuginfo**
4. **yum install idm-pki-symkey-debuginfo idm-pki-tools-debuginfo**

([BZ#2134093](#))

**ldap\_id\_use\_start\_tls** オプションのデフォルト値を使用する場合の潜在的なリスク

ID ルックアップに TLS を使用せずに **ldap://** を使用すると、攻撃ベクトルのリスクが生じる可能性があります。特に、中間者 (MITM) 攻撃は、攻撃者が、たとえば、LDAP 検索で返されたオブジェクトの UID または GID を変更することによってユーザーになりすますことを可能にする可能性があります。

現在、TLS を強制する SSSD 設定オプション **ldap\_id\_use\_start\_tls** は、デフォルトで **false** に設定されています。セットアップが信頼できる環境で動作していることを確認し、**id\_provider = ldap** に暗号

化されていない通信を使用しても安全かどうかを判断してください。注記: `id_provider = ad` および `id_provider = ipa` は、SASL および GSSAPI によって保護された暗号化接続を使用するため、影響を受けません。

暗号化されていない通信を使用することが安全ではない場合は、`/etc/sss/sss.conf` ファイルで `ldap_id_use_start_tls` オプションを `true` に設定して TLS を強制します。デフォルトの動作は、RHEL の将来のリリースで変更される予定です。

(JIRA:RHELPLAN-155168)

## 11.13. デスクトップ

**ソフトウェアリポジトリからの flatpak リポジトリの無効化ができません。**

現時点で、GNOME Software ユーティリティの Software Repositories ツールで `flatpak` リポジトリを無効化または削除することはできません。

(BZ#1668760)

**Generation 2 の RHEL 8 仮想マシンが Hyper-V Server 2016 ホストで起動できない場合があります。**

Microsoft Hyper-V Server 2016 ホストで実行している仮想マシンで RHEL 8 をゲストオペレーティングシステムとして使用すると、仮想マシンが起動しなくなり、GRUB ブートメニューに戻る場合があります。さらに、以下のエラーが Hyper-V イベントログに記録されます。

```
The guest operating system reported that it failed with the following error code: 0x1E
```

このエラーは、Hyper-V ホストの UEFI ファームウェアバグが原因で発生します。この問題を回避するには、Hyper-V Server2019 以降をホストとして使用します。

(BZ#1583445)

**ドラッグアンドドロップが、デスクトップとアプリケーション間で機能しません。**

`gnome-shell-extensions` パッケージのバグにより、ドラッグアンドドロップ機能は現在、デスクトップとアプリケーションの間では機能しません。この機能のサポートは、今後のリリースで追加される予定です。

(BZ#1717947)

## 11.14. グラフィックインフラストラクチャー

**radeon がハードウェアを適切なハードウェアリセットに失敗します。**

現在、`radeon` カーネルドライバーは、`kexec` コンテキストでハードウェアを正しくリセットしません。代わりに `radeon` がフェイルオーバーします。これにより、`kdump` サービスの残りの部分が失敗します。

この問題を回避するには、`/etc/kdump.conf` ファイルに以下の行を追加して、`kdump` で `radeon` を無効にします。

```
dracut_args --omit-drivers "radeon"
force_rebuild 1
```

マシンと **kdump** を再起動します。 **kdump** の起動後、設定ファイルから **force\_rebuild 1** 行が削除される可能性があります。

このシナリオでは、 **kdump** 中にグラフィックは利用できませんが、 **kdump** は正常に動作します。

(BZ#1694705)

1つの MST トポロジーで複数の HDR ディスプレイを使用すると、電源が入らないことがあります。

**nouveau** ドライバーの NVIDIA Turing GPUs を使用するシステムで、 **DisplayPort** ハブ (ラップトップのドックなど) を使用して HDR プラグインのサポートがあるモニターを複数接続すると、電源が入らないことがあります。これは、全ディスプレイをサポートする帯域幅がハブ上にないと、システムが誤って判断してしまうことが原因で発生します。

(BZ#1812577)

**ビデオメモリーが少なくなったため、ESXi の GUI がクラッシュする可能性がある**

vCenter Server 7.0.1 を使用する VMware ESXi 7.0.1 ハイパーバイザーの RHEL 仮想マシンでグラフィカルユーザーインターフェイス (GUI) には、一定量のビデオメモリーが必要です。複数のコンソールまたは高解像度のモニターを仮想マシンに接続する場合、GUI には少なくとも 16 MB のビデオメモリーが必要です。ビデオメモリーが少ないで GUI を起動すると、GUI が突然終了する可能性があります。

この問題を回避するには、仮想マシンに 16 MB 以上のビデオメモリーを割り当てるようにハイパーバイザーを設定します。その結果、仮想マシンの GUI がクラッシュしなくなりました。

この問題が発生した場合は、VMware に報告することを推奨します。

VMware の記事、 [VMs with high resolution VM console may experience a crash on ESXi 7.0.1 \(83194\)](#) 、も参照してください。

(BZ#1910358)

**VNC Viewer が、IBM Z で 16 ビットのカラーデプスで誤った色を表示**

VNC Viewer アプリケーションは、16 ビットのカラーデプスで IBM Z サーバーの VNC セッションに接続すると、誤った色を表示します。

この問題を回避するには、VNC サーバーで 24 ビットのカラーデプスを設定します。 **Xvnc** サーバーの場合は、 **Xvnc** 設定で **-depth 16** オプションを **-depth 24** に置き換えます。

その結果、VNC クライアントで色が正しく表示されますが、サーバーでは、より多くのネットワーク帯域幅が使用されます。

(BZ#1886147)

**sudo コマンドを使用してグラフィカルアプリケーションを実行できません。**

権限が昇格されたユーザーで、グラフィカルアプリケーションを実行しようとする、エラーメッセージが表示され、アプリケーションを開くことができません。この障害は、 **Xauthority** ファイルで、通常ユーザーの認証情報を使用して認証するように、 **Xwayland** に制限が加えられているため発生します。

この問題を回避するには、 **sudo -E** コマンドを使用して、 **root** ユーザーとしてグラフィカルアプリケーションを実行します。

(BZ#1673073)

## ARM でハードウェアアクセラレーションがサポートされない

組み込みグラフィックドライバーは、64 ビット ARM アーキテクチャー上のハードウェアアクセラレーションまたは Vulkan API に対応していません。

ARM でハードウェアアクセラレーションまたは Vulkan を有効にするには、プロプライエタリーの Nvidia ドライバーをインストールします。

(JIRA:RHELPLAN-57914)

## Matrox G200e が VGA ディスプレイに出力を表示しない

以下のシステム設定を使用すると、ディスプレイにグラフィカル出力が表示されない場合があります。

- Matrox G200e GPU
- VGA コントローラーで接続されたディスプレイ

したがって、この設定で RHEL を使用またはインストールすることはできません。

この問題を回避するには、以下の手順に従います。

1. ブートローダーメニューにシステムを起動します。
2. `module_blacklist=mgag200` オプションをカーネルコマンドラインに追加します。

これにより、RHEL が起動し、予想どおりにグラフィカル出力が表示されますが、最大解像度は 16 ビットの色深度で 1024x768 に制限されます。

(BZ#2130159)

## 11.15. WEB コンソール

### VNC コンソールが特定の解像度で正しく動作しない

特定のディスプレイ解像度で Virtual Network Computing (VNC) コンソールを使用すると、マウスオフセットの問題が発生したり、インターフェイスの一部しか表示されない場合があります。そのため、VNC コンソールを使用できない場合があります。この問題を回避するには、VNC コンソールのサイズを拡大するか、代わりにコンソールタブのデスクトップビューアーを使用してリモートビューアーを起動します。

(BZ#2030836)

## 11.16. RED HAT ENTERPRISE LINUX システムロール

### Playbook またはインベントリーでホスト名 localhost を使用して localhost を管理できません

RHEL に **ansible-core 2.13** パッケージが含まれているため、ノードを管理しているのと同じホストで Ansible を実行している場合は、Playbook またはインベントリーで **localhost** ホスト名を使用して実行することはできません。これは、**ansible-core 2.13** が **python38** モジュールを使用し、ライブラリーの多くが欠落しているために発生します。たとえば、**storage** ロールの場合は **blivet**、**network** ロールの場合は **gobject** です。この問題を回避するには、Playbook またはインベントリーでホスト名 **localhost** をすでに使用している場合は、**ansible\_connection=local** を使用するか、**ansible\_connection=local** オプションを使用して **localhost** をリストするインベントリーファイルを作成することで接続を追加できます。これにより、**localhost** 上のリソースを管理できます。詳細については、記事 [ローカルホストで実行すると RHEL System Roles の Playbook が失敗する](#) を参照してください。

(BZ#2041997)

## 11.17. 仮想化

### 多数のキューを使用すると、Windows 仮想マシンで障害が発生することがある

仮想 Trusted Platform Module (vTPM) デバイスが有効で、マルチキュー `virtio-net` 機能が 250 を超えるキューを使用するように設定されている場合、Windows 仮想マシン (VM) が失敗することがあります。

この問題は、vTPM デバイスの制限が原因で発生します。vTPM デバイスには、開いているファイル記述子の最大数に関するハードコーディングされた制限があります。新しいキューごとに複数のファイル記述子が開かれるため、内部の vTPM 制限を超えて VM が失敗する可能性があります。

この問題を回避するには、次の 2 つのオプションのいずれかを選択します。

- vTPM デバイスを有効のままにしますが、使用するキューは 250 未満にします。
- 250 を超えるキューを使用するには、vTPM デバイスを無効にします。

(BZ#2020133)

### Milan 仮想マシンの CPU タイプは、AMD Milan システムで利用できないことがある

一部の AMD Milan システムでは、Enhanced REP MOVSB (**erms**) および Fast Short REP MOVSB (**fsrm**) 機能フラグがデフォルトで BIOS で無効になっています。したがって、**Milan** CPU タイプは、これらのシステムで利用できない可能性があります。さらに、機能フラグ設定が異なる Milan ホスト間の仮想マシンのライブマイグレーションが失敗する可能性があります。これらの問題を回避するには、ホストの BIOS で **erms** および **fsrm** を手動で有効にします。

(BZ#2077770)

### virtio-blk を使用して仮想マシンに LUN デバイスを割り当てると機能しません。

q35 マシンタイプは、移行用の virtio 1.0 デバイスをサポートしないため、RHEL 8 では virtio 1.0 で非推奨となった機能はサポートされません。特に、RHEL 8 ホストで virtio-blk デバイスから SCSI コマンドを送信することはできません。したがって、virtio-blk コントローラーを使用する場合は、物理ディスクを LUN デバイスとして仮想マシンに割り当てると失敗します。

物理ディスクをゲストオペレーティングシステムを通して渡すことは引き続き可能ですが、**device='lun'** オプションではなく、**device='disk'** オプションで設定する必要があることに留意してください。

(BZ#1777138)

### iommu\_platform=on が IBM POWER で起動に失敗する

RHEL 8 は現在、IBM POWER システムの仮想マシン用の **iommu\_platform=on** パラメーターに対応していません。これにより、IBM POWER ハードウェアでこのパラメーターを使用して仮想マシンを起動すると、仮想マシンがシステムの起動プロセス時に応答しなくなります。

(BZ#1910848)

### ibmvfc ドライバーを使用すると、IBM POWER ホストがクラッシュする可能性がある

PowerVM 論理パーティション (LPAR) で RHEL 8 を実行すると、現在、**ibmvfc** ドライバーの問題により、さまざまなエラーが発生する可能性があります。結果として、ホストのカーネルは、以下のような特定の状況下でパニックになる可能性があります。

- Live Partition Mobility (LPM) 機能の使用
- ホストアダプターのリセット
- SCSI エラー処理機能 (SCSI EH) 機能の使用

(BZ#1961722)

### IBM POWER Systems で `perf kvm` レコードを使用すると、仮想マシンがクラッシュする可能性があります。

IBM POWER ハードウェアのリトルエンディアンバリエントで RHEL 8 ホストを使用する場合は、**`perf kvm record`** コマンドを使用して KVM 仮想マシンのイベントサンプルを収集すると、仮想マシンが応答しなくなることがあります。この状況は、以下の場合に発生します。

- **`perf`** ユーティリティーは権限のないユーザーによって使用され、**`-p`** オプションは仮想マシンを識別するために使用されます (**`perf kvm record -e trace_cycles -p 12345`**)。
- 仮想マシンが **`virsh`** シェルを使用して起動している。

この問題を回避するには、**`perf kvm`** ユーティリティーに **`-i`** オプションを指定して、**`virsh`** シェルを使用して作成した仮想マシンを監視します。以下に例を示します。

```
# perf kvm record -e trace_imc/trace_cycles/ -p <guest pid> -i
```

**`-i`** オプションを使用する場合、子タスクはカウンターを継承しないため、スレッドは監視されないことに注意してください。

(BZ#1924016)

### 特定の CPU モデルの使用時に Hyper-V を有効化した Windows Server 2016 仮想マシンが起動に失敗する

現在、Windows Server 2016 をゲストオペレーティングシステムとして使用し、Hyper-V ロールが有効になっていて、以下の CPU モデルのいずれかを使用する仮想マシンを起動できません。

- EPYC-IBPB
- EPYC

この問題を回避するには、**`EPYC-v3`** CPU モデルを使用するか、仮想マシンの **`xsaves`** CPU フラグを手動で有効にします。

(BZ#1942888)

### RHEL 7-ALT ホストから RHEL 8 への POWER9 ゲストの移行に失敗する

現在のリリースでは、RHEL 7-ALT ホストシステムから RHEL 8 に POWER9 仮想マシンを移行すると、**`Migration status: active`** のステータスで応答がなくなります。

この問題を回避するには、RHEL 7-ALT ホストで Transparent Huge Pages (THP) を無効にすることで、移行が正常に完了します。

(BZ#1741436)

**`virt-customize`** を使用すると、**`guestfs-firstboot`** が失敗することがあります。

**virt-customize** ユーティリティを使用して仮想マシン (VM) ディスクイメージを変更すると、SELinux パーMISSIONが正しくないために **guestfs-firstboot** サービスが失敗します。これにより、ユーザーの作成やシステム登録の失敗など、仮想マシンの起動時にさまざまな問題が発生します。

この問題を回避するには、**virt-customize** でディスクイメージを変更した後に、仮想マシンのカーネルコマンドラインに **--selinux-relabel** を追加します。

(BZ#1554735)

**macvtap** 仮想ネットワークから正引きインターフェイスを削除すると、このネットワークの接続数がすべてリセットされます。

現在、複数のフォワードインターフェイスを持つ **macvtap** 仮想ネットワークからフォワードインターフェイスを削除すると、ネットワークの他のフォワードインターフェイスの接続ステータスもリセットされます。したがって、ライブネットワーク XML の接続情報が正しくありません。ただし、これは仮想ネットワークの機能に影響を与えるわけではないことに注意してください。この問題を回避するには、ホストで **libvirtd** サービスを再起動します。

(BZ#1332758)

**SLOF** が指定された仮想マシンは **netcat** インターフェイスでの起動に失敗する

**netcat(nc)** インターフェイスを使用して、現在 Slimline Open Firmware(SLOF) プロンプトで待機中の仮想マシンのコンソールにアクセスすると、ユーザー入力は無視され、仮想マシンが応答しないままとまります。この問題を回避するには、仮想マシンに接続する場合は **nc -C** オプションを使用するか、代わりに **telnet** インターフェイスを使用します。

(BZ#1974622)

場合によっては、**virt-manager** で仲介デバイスを仮想マシンに接続すると失敗します

**virt-manager** アプリケーションは現在、仲介されたデバイスを検出できますが、デバイスがアクティブかどうかを認識できません。結果として、**virt-manager** を使用して、非アクティブな仲介デバイスを実行中の仮想マシン (VM) に接続しようとするとう失敗します。同様に、非アクティブな仲介デバイスを使用する新しい VM を作成しようとするとう、**device not found** エラーで失敗します。

この問題を回避するには、**virt-manager** で使用する前に、**virsh nodedev-start** または **mdevctl start** コマンドを使用して仲介デバイスをアクティブにします。

(BZ#2026985)

**RHEL 9** 仮想マシンが **POWER8** 互換モードでの起動に失敗する

現在、仮想マシン (VM) が次のような CPU 設定も使用している場合、ゲストオペレーティングシステムとして RHEL 9 を実行する仮想マシンの起動は失敗します。

```
<cpu mode="host-model">
  <model>power8</model>
</cpu>
```

この問題を回避するには、RHEL 9 仮想マシンで POWER8 互換モードを使用しないでください。

さらに、POWER8 ホストでは RHEL 9 VM を実行できないことに注意してください。

(BZ#2035158)

ホストで **OVS** サービスを再起動すると、実行中の VM でネットワーク接続がブロックされることがある

ホストで Open vSwitch (OVS) サービスが再起動またはクラッシュすると、このホストで実行されている仮想マシン (VM) はネットワークデバイスの状態を回復できません。その結果、仮想マシンがパケットを完全に受信できなくなる可能性があります。

この問題は、**virtio** ネットワークスタックで圧縮された virtqueue 形式を使用するシステムのみに影響します。

この問題を回避するには、**virtio** ネットワークデバイス定義で **packed=off** パラメーターを使用して、圧縮された virtqueue を無効にします。圧縮された virtqueue を無効にすると、状況によっては、ネットワークデバイスの状態を RAM から回復できます。

(BZ#1792683)

### 多数の virtio-blk ディスクを使用すると、仮想マシンが起動しないことがある

多数の virtio-blk デバイスを仮想マシンに追加すると、プラットフォームで利用可能な割り込みベクトルの数が使い切られる可能性があります。これが発生すると、仮想マシンのゲスト OS は起動できず、**dracut-initqueue[392]: Warning: Could not boot** エラーが表示されます。

(BZ#1719687)

### SUID と SGID が virtiofs で自動的にクリアされない

**killpriv\_v2** 機能を使用して **virtiofsd** サービスを実行すると、一部のファイルシステム操作を実行した後、システムが SUID および SGID アクセス許可を自動的にクリアしない場合があります。したがって、アクセス許可をクリアしないと、潜在的なセキュリティ上の脅威が発生する可能性があります。この問題を回避するには、次のコマンドを入力して **killpriv\_v2** 機能を無効にします。

```
# virtiofsd -o no_killpriv_v2
```

(BZ#1966475)

### AMD EPYC でホストパススルーモードを使用する際に、SMT CPU トポロジーが仮想マシンで検出されない

AMD EPYC ホストで行われた CPU ホストパススルーモードで仮想マシンを起動すると、**TOPOEXT** 機能フラグは存在しません。したがって、仮想マシンは、コアごとに複数のスレッドを持つ仮想 CPU トポロジーを検出できません。この問題を回避するには、ホストパススルーの代わりに EPYC CPU モデルを使用して仮想マシンを起動します。

(BZ#1740002)

## 11.18. クラウド環境の RHEL

### VMware ホストの RHEL 仮想マシンで静的 IP を設定できない

現在、VMware ホストで RHEL を仮想マシンのゲストオペレーティングシステムとして使用すると、DatasourceOVF 機能は正しく機能しません。これにより、**cloud-init** ユーティリティを使用して、仮想マシンのネットワークを静的 IP に設定し、仮想マシンを再起動すると、仮想マシンのネットワークが DHCP に変更されます。

(BZ#1750862)

### Azure および Hyper-V で kdump が起動しないことがある

Microsoft Azure または Hyper-V ハイパーバイザーでホストされている RHEL 8 ゲストオペレーティングシステムでは、実行後通知が有効な場合に **kdump** カーネルの起動が失敗することがあります。

この問題を回避するには、`crash kexec post notifiers` を無効にします。

```
# echo N > /sys/module/kernel/parameters/crash_kexec_post_notifiers
```

(BZ#1865745)

## 複数のゲストディスクで Hyper-V 仮想マシンを起動する際に、SCSI ホストアドレスが変更されることがある

現在、Hyper-V ハイパーバイザーで RHEL 8 仮想マシンを起動すると、場合によっては、**Host, Bus, Target, Lun** (HBTL) SCSI アドレスのホスト部分が変わることがあります。したがって、仮想マシンで HBTL SCSI 識別またはデバイスノードで設定した自動タスクは一貫して動作しません。これは、仮想マシンに複数のディスクがある場合、またはディスクに異なるサイズがある場合に発生します。

この問題を回避するには、以下のいずれかの方法でキックスタートファイルを変更します。

### 方法 1: SCSI デバイスに永続的な識別子を使用

たとえば、以下の powershell スクリプトを使用すると、特定のデバイス識別子を特定できます。

```
# Output what the /dev/disk/by-id/<value> for the specified hyper-v virtual disk.
# Takes a single parameter which is the virtual disk file.
# Note: kickstart syntax works with and without the /dev/ prefix.
param (
  [Parameter(Mandatory=$true)][string]$virtualdisk
)

$what = Get-VHD -Path $virtualdisk
$part = $what.DiskIdentifier.ToLower().split('-')

$p = $part[0]
$s0 = $p[6] + $p[7] + $p[4] + $p[5] + $p[2] + $p[3] + $p[0] + $p[1]

$p = $part[1]
$s1 = $p[2] + $p[3] + $p[0] + $p[1]

[string]::format("/dev/disk/by-id/wwn-0x60022480{0}{1}{2}", $s0, $s1, $part[4])
```

このスクリプトは、ハイパーホストで使用することができます。以下に例を示します。

```
PS C:\Users\Public\Documents\Hyper-V\Virtual hard disks> .\by-id.ps1 .\Testing_8\disk_3_8.vhdx
/dev/disk/by-id/wwn-0x60022480e00bc367d7fd902e8bf0d3b4
PS C:\Users\Public\Documents\Hyper-V\Virtual hard disks> .\by-id.ps1 .\Testing_8\disk_3_9.vhdx
/dev/disk/by-id/wwn-0x600224807270e09717645b1890f8a9a2
```

その後、以下のようにキックスタートファイルでディスクの値を使用できます。

```
part / --fstype=xfs --grow --asprimary --size=8192 --ondisk=/dev/disk/by-id/wwn-
0x600224807270e09717645b1890f8a9a2
part /home --fstype="xfs" --grow --ondisk=/dev/disk/by-id/wwn-
0x60022480e00bc367d7fd902e8bf0d3b4
```

これらの値は仮想ディスクごとに固有であるため、仮想マシンインスタンスごとに設定を行う必要があります。そのため、**%include** 構文を使用して、ディスク情報を別のファイルに配置すると便利です。

## 方法 2: デバイス選択をサイズで設定

サイズに基づいてディスク選択を設定するキックスタートファイルには、以下のような行を含める必要があります。

```
...

# Disk partitioning information is supplied in a file to kick start
#include /tmp/disks

...

# Partition information is created during install using the %pre section


```


```


```

(BZ#1906870)

### cloud-init によってプロビジョニングされ、NFSv3 マウントエントリーで設定された場合、Azure で RHEL インスタンスが起動しない

現在、仮想マシンが **cloud-init** ツールによってプロビジョニングされ、仮想マシンのゲストオペレーティングシステムで **/etc/fstab** ファイルに NFSv3 マウントエントリーがある場合、Microsoft Azure クラウドプラットフォームで RHEL 仮想マシンの起動に失敗します。

(BZ#2081114)

## 11.19. サポート性

### getattachment コマンドが複数の添付ファイルを一度にダウンロードできない

**redhat-support-tool** コマンドは、添付ファイルをダウンロードするための **getattachment** サブコマンドを提供します。ただし、**getattachment** は現在、1つの添付ファイルしかダウンロードできず、複数の添付ファイルをダウンロードできません。

回避策として、**getattachment** サブコマンドで各添付ファイルのケース番号と UUID を渡すことにより、複数の添付ファイルを1つずつダウンロードできます。

(BZ#2064575)

## redhat-support-tool が FUTURE 暗号化ポリシーを使用すると機能しない

カスタマーポータル API の証明書が使用する暗号化キーは **FUTURE** のシステム全体の暗号化ポリシーが定義する要件を満たさないので、現時点で **redhat-support-tool** ユーティリティは、このポリシーレベルでは機能しません。

この問題を回避するには、カスタマーポータル API への接続中に **DEFAULT** 暗号化ポリシーを使用します。

(BZ#1802026)

## IBM Power Systems (Little Endian) で sos report を実行するとタイムアウトする

数百または数千の CPU を搭載した IBM Power Systems (Little Endian) で **sos report** コマンドを実行すると、`/sys/devices/system/cpu` ディレクトリーの膨大なコンテンツを収集する際のプロセッサプラグインはデフォルトのタイムアウトである 300 秒に達します。回避策として、それに応じてプラグインのタイムアウトを増やします。

- 1 回限りの設定の場合は、次を実行します。

```
# sos report -k processor.timeout=1800
```

- 永続的な変更を行うには、`/etc/sos/sos.conf` ファイルの **[plugin\_options]** セクションを編集します。

```
[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
```

値の例は 1800 に設定されています。特定のタイムアウト値は、特定のシステムに大きく依存します。プラグインのタイムアウトを適切に設定するには、次のコマンドを実行して、タイムアウトなしで 1 つのプラグインを収集するために必要な時間を最初に見積もることができます。

```
# time sos report -o processor -k processor.timeout=0 --batch --build
```

(BZ#2011413)

## 11.20. コンテナ

### 古いコンテナイメージ内で systemd を実行すると動作しない

古いコンテナイメージ (例:**centos:7**) で **systemd** を実行しても動作しません。

```
$ 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.
```

この問題を回避するには、以下のコマンドを使用します。

```
# mkdir /sys/fs/cgroup/systemd
# mount none -t cgroup -o none,name=systemd /sys/fs/cgroup/systemd
# podman run --runtime /usr/bin/crun --annotation=run.oci.systemd.force_cgroup_v1=/sys/fs/cgroup -
```

---

```
-rm -ti centos:7 /usr/lib/systemd/systemd
```

(JIRA:RHELPLAN-96940)

## 第12章 国際化

### 12.1. RED HAT ENTERPRISE LINUX 8 の多言語

Red Hat Enterprise Linux 8 は、複数の言語のインストールと、要件に応じた言語の変更に対応します。

- 東アジア言語 - 日本語、韓国語、簡体字中国語、および繁体字中国語。
- ヨーロッパ言語 - 英語、ドイツ語、スペイン語、フランス語、イタリア語、ポルトガル語、およびロシア語。

次の表は、さまざまな主要言語に提供されるフォントと入力方法を示しています。

言語	デフォルトフォント (フォントパッケージ)	入力メソッド
英語	dejavu-sans-fonts	
フランス語	dejavu-sans-fonts	
ドイツ語	dejavu-sans-fonts	
イタリア語	dejavu-sans-fonts	
ロシア語	dejavu-sans-fonts	
スペイン語	dejavu-sans-fonts	
ポルトガル語	dejavu-sans-fonts	
簡体字中国語	google-noto-sans-cjk-ttc-fonts、 google-noto-serif-cjk-ttc-fonts	ibus-libpinyin、libpinyin
繁体字中国語	google-noto-sans-cjk-ttc-fonts、 google-noto-serif-cjk-ttc-fonts	ibus-libzhuyin、libzhuyin
日本語	google-noto-sans-cjk-ttc-fonts、 google-noto-serif-cjk-ttc-fonts	ibus-kkc、libkkc
韓国語	google-noto-sans-cjk-ttc-fonts、 google-noto-serif-cjk-ttc-fonts	ibus-hangul、libhangul

### 12.2. RHEL 8 における国際化の主な変更点

RHEL 8 では、RHEL 7 の国際化に以下の変更が加えられています。

- Unicode 11 コンピューティングの業界標準のサポートが追加されました。

- 国際化は複数のパッケージで配布され、より小さなフットプリントのインストールを可能にします。詳細は、[Using langpacks](#) を参照してください。
- 多くの **glibc** ロケールが Unicode Common Locale Data Repository (CLDR) と同期されています。

## 付録A コンポーネント別のチケットリスト

本書には Bugzilla と JIRA ID が記載されています。一般にアクセス可能な Bugzilla バグには、チケットへのリンクが含まれます。

コンポーネント	チケット
<b>389-ds-base</b>	<a href="#">BZ#2052528</a> 、 <a href="#">BZ#2057063</a> 、 <a href="#">BZ#2057066</a> 、 <a href="#">BZ#2062679</a> 、 <a href="#">BZ#1817505</a>
<b>NetworkManager</b>	<a href="#">BZ#2097270</a> 、 <a href="#">BZ#2082000</a> 、 <a href="#">BZ#2063109</a> 、 <a href="#">BZ#1943153</a> 、 <a href="#">BZ#1920398</a> 、 <a href="#">BZ#2132754</a>
<b>SLOF</b>	<a href="#">BZ#1910848</a>
<b>accel-config</b>	<a href="#">BZ#1843266</a>
<b>anaconda</b>	<a href="#">BZ#1899494</a> 、 <a href="#">BZ#1970726</a> 、 <a href="#">BZ#2029101</a> 、 <a href="#">BZ#2050140</a> 、 <a href="#">BZ#1914955</a> 、 <a href="#">BZ#1929105</a> 、 <a href="#">BZ#2126506</a>
<b>ansible-collection-microsoft-sql</b>	<a href="#">BZ#2066338</a>
<b>ansible-collection-redhat-rhel_mgmt</b>	<a href="#">BZ#2112435</a>
<b>ansible-freeipa</b>	<a href="#">BZ#2076554</a>
<b>apr</b>	<a href="#">BZ#1819607</a>
<b>authselect</b>	<a href="#">BZ#1892761</a>
<b>bacula</b>	<a href="#">BZ#2089399</a>
<b>brltty</b>	<a href="#">BZ#2008197</a>
<b>chrony</b>	<a href="#">BZ#2062356</a>
<b>clevis</b>	<a href="#">BZ#2107081</a>
<b>cloud-init</b>	<a href="#">BZ#1750862</a>
<b>cockpit-appstream</b>	<a href="#">BZ#2030836</a>
<b>cockpit</b>	<a href="#">BZ#2056786</a> 、 <a href="#">BZ#1666722</a>
<b>coreutils</b>	<a href="#">BZ#2030661</a>

コンポーネント	チケット
corosync-qdevice	<a href="#">BZ#1784200</a>
crash-ptdump-command	<a href="#">BZ#1838927</a>
crash	<a href="#">BZ#1906482</a>
createrepo_c	<a href="#">BZ#1973588</a>
cronie	<a href="#">BZ#1832510</a>
crypto-policies	<a href="#">BZ#1919155</a> , <a href="#">BZ#1660839</a>
cups-filters	<a href="#">BZ#2064606</a>
device-mapper-multipath	<a href="#">BZ#2065477</a> , <a href="#">BZ#2011699</a>
distribution	<a href="#">BZ#2063772</a> , <a href="#">BZ#1657927</a>
dnf	<a href="#">BZ#2060815</a> , <a href="#">BZ#1986657</a>
dotnet7.0	<a href="#">BZ#2112096</a>
dyninst	<a href="#">BZ#2057676</a>
ec2-images	<a href="#">BZ#1862930</a>
edk2	<a href="#">BZ#1741615</a> , <a href="#">BZ#1935497</a>
fapolicyd	<a href="#">BZ#2070639</a> , <a href="#">BZ#2100087</a> , <a href="#">BZ#2054741</a>
fence-agents	<a href="#">BZ#1775847</a>
firewalld	<a href="#">BZ#1871860</a>
freeradius	<a href="#">BZ#1958979</a>
frr	<a href="#">BZ#1714984</a>
gcc-toolset-12-annobin	<a href="#">BZ#2077447</a>
gcc-toolset-12-binutils	<a href="#">BZ#2077448</a>
gcc-toolset-12-gcc	<a href="#">BZ#2077276</a>

コンポーネント	チケット
<b>gcc-toolset-12-gdb</b>	BZ#2077492
<b>gdb</b>	BZ#1853140
<b>glibc</b>	BZ#1888660、BZ#1982608、BZ#2065588、BZ#1961109、BZ#2089247、BZ#2091553、BZ#2104907、BZ#2033684、BZ#2096189、BZ#2077835
<b>gnome-control-center</b>	BZ#2079139
<b>gnome-shell-extensions</b>	<a href="#">BZ#1717947</a>
<b>gnome-software</b>	<a href="#">BZ#1668760</a>
<b>gnutls</b>	BZ#1628553
<b>golang</b>	BZ#2075162
<b>grub2</b>	BZ#2074762, BZ#1583445
<b>grubby</b>	<a href="#">BZ#1978226</a> , <a href="#">BZ#1900829</a>
<b>initscripts</b>	<a href="#">BZ#1875485</a>
<b>ipa</b>	<a href="#">BZ#2059396</a> , <a href="#">BZ#2022028</a> , <a href="#">BZ#782917</a> , <a href="#">BZ#2062379</a> , <a href="#">BZ#1924707</a> , <a href="#">BZ#2120572</a> , <a href="#">BZ#2122919</a> , <a href="#">BZ#1664719</a> , <a href="#">BZ#1664718</a> , <a href="#">BZ#2101770</a>
<b>ipmitool</b>	<a href="#">BZ#1873614</a>
<b>iptables</b>	BZ#2058444
<b>kdump-anaconda-addon</b>	<a href="#">BZ#2086100</a>
<b>kernel</b>	BZ#2084242, BZ#2062870, BZ#2068429, BZ#2101938, JIRA:RHELPLAN-121252, BZ#2096127, BZ#2054656, BZ#1868526, BZ#1694705, BZ#1730502, BZ#1609288, BZ#1602962, BZ#1865745, BZ#1906870, BZ#1924016, BZ#1942888, BZ#1812577, BZ#1910358, BZ#1930576, BZ#2046396, BZ#1793389, BZ#1654962, BZ#1940674, BZ#1920086, BZ#1971506, BZ#2059262, BZ#2050411, BZ#2106341, BZ#2130159, BZ#1605216, BZ#1519039, BZ#1627455, BZ#1501618, BZ#1633143, BZ#1814836, BZ#1696451, BZ#1348508, BZ#1837187, BZ#1904496, BZ#1660337, BZ#1905243, BZ#1878207, BZ#1665295, BZ#1871863, BZ#1569610, BZ#1794513
<b>kexec-tools</b>	BZ#2006000

コンポーネント	チケット
<b>kmod-kvdo</b>	<a href="#">BZ#2109047</a>
<b>kmod</b>	<a href="#">BZ#2103605</a>
<b>krb5</b>	<a href="#">BZ#2026462</a> , <a href="#">BZ#2125182</a> , <a href="#">BZ#1877991</a>
<b>libdnf</b>	<a href="#">BZ#2088149</a>
<b>libgnome-keyring</b>	<a href="#">BZ#1607766</a>
<b>libguestfs</b>	<a href="#">BZ#1554735</a>
<b>libpfm</b>	<a href="#">BZ#2067218</a>
<b>libreswan</b>	<a href="#">BZ#1989050</a>
<b>libselinux-python-2.8-module</b>	<a href="#">BZ#1666328</a>
<b>libva</b>	<a href="#">BZ#2099907</a>
<b>libvirt</b>	<a href="#">BZ#1664592</a> , <a href="#">BZ#1332758</a> , <a href="#">BZ#2067126</a> , <a href="#">BZ#1528684</a>
<b>libvpd</b>	<a href="#">BZ#2051319</a>
<b>llvm-toolset</b>	<a href="#">BZ#2061042</a> , <a href="#">BZ#2088315</a>
<b>lsvpd</b>	<a href="#">BZ#2051316</a>
<b>lvm2</b>	<a href="#">BZ#1496229</a> , <a href="#">BZ#1768536</a>
<b>make43</b>	<a href="#">BZ#2083419</a>
<b>mariadb-java-client</b>	<a href="#">BZ#2043212</a>
<b>mariadb</b>	<a href="#">BZ#1944653</a> , <a href="#">BZ#1942330</a>
<b>maven</b>	<a href="#">BZ#2083114</a>
<b>mercurial</b>	<a href="#">BZ#2089849</a>
<b>mesa</b>	<a href="#">BZ#1886147</a>
<b>motif</b>	<a href="#">BZ#2060571</a>

コンポーネント	チケット
<b>nfs-utils</b>	BZ#1946283, BZ#2087187, BZ#2081114, BZ#1592011
<b>nispor</b>	BZ#1848817
<b>nodejs</b>	BZ#2083073
<b>nss_nis</b>	<a href="#">BZ#1803161</a>
<b>nss</b>	<a href="#">BZ#1817533</a> , BZ#1645153, <a href="#">BZ#2097837</a>
<b>open-vm-tools</b>	BZ#2061193
<b>opencryptoki</b>	BZ#2043845
<b>opencv</b>	<a href="#">BZ#2104776</a> , <a href="#">BZ#1886310</a>
<b>openmpi</b>	BZ#1866402
<b>opensc</b>	<a href="#">BZ#1947025</a>
<b>openssh</b>	BZ#2044354
<b>openssl</b>	BZ#1810911
<b>osbuild-composer</b>	<a href="#">BZ#2065734</a>
<b>oscap-anaconda-addon</b>	<a href="#">BZ#2075508</a> , <a href="#">BZ#1843932</a> , BZ#1665082
<b>pacemaker</b>	<a href="#">BZ#2036815</a> , <a href="#">BZ#2059638</a> , <a href="#">BZ#1182956</a> , <a href="#">BZ#1724310</a>
<b>papi</b>	BZ#2037426, <a href="#">BZ#2071558</a> , BZ#2037427, BZ#2037417
<b>pcs</b>	<a href="#">BZ#1786964</a> , <a href="#">BZ#1954099</a> , <a href="#">BZ#2023845</a> , <a href="#">BZ#1950551</a> , <a href="#">BZ#1909904</a> , <a href="#">BZ#1874624</a> , BZ#1619620, BZ#1847102, BZ#1851335
<b>pki-core</b>	<a href="#">BZ#1729215</a> , <a href="#">BZ#2134093</a> , BZ#1628987
<b>podman</b>	<a href="#">BZ#2097708</a> , JIRA:RHELPLAN-77238
<b>postfix</b>	<a href="#">BZ#1711885</a>
<b>powerpc-utils</b>	BZ#2078514, BZ#2051330
<b>ppc64-diag</b>	BZ#2051313

コンポーネント	チケット
<b>procps-ng</b>	<a href="#">BZ#2111915</a>
<b>pykickstart</b>	<a href="#">BZ#1637872</a>
<b>qemu-kvm</b>	<a href="#">BZ#2043830</a> , <a href="#">BZ#2020133</a> , <a href="#">BZ#1740002</a> , <a href="#">BZ#1719687</a> , <a href="#">BZ#1966475</a> , <a href="#">BZ#1792683</a> , <a href="#">BZ#1651994</a>
<b>rear</b>	<a href="#">BZ#2072978</a> , <a href="#">BZ#2115918</a> , <a href="#">BZ#2077404</a> , <a href="#">BZ#2021935</a> , <a href="#">BZ#2035872</a> , <a href="#">BZ#1925531</a> , <a href="#">BZ#1868421</a> , <a href="#">BZ#2083301</a>
<b>redhat-support-tool</b>	<a href="#">BZ#2064575</a> , <a href="#">BZ#1802026</a>
<b>redis</b>	<a href="#">BZ#1999873</a>
<b>restore</b>	<a href="#">BZ#1997366</a>
<b>rhel-system-roles</b>	<a href="#">BZ#2064067</a> , <a href="#">BZ#2115884</a> , <a href="#">BZ#2060377</a> , <a href="#">BZ#2072749</a> , <a href="#">BZ#2083378</a> , <a href="#">BZ#2100285</a> , <a href="#">BZ#2060378</a> , <a href="#">BZ#2083426</a> , <a href="#">BZ#2100298</a> , <a href="#">BZ#2101607</a> , <a href="#">BZ#2115161</a> , <a href="#">BZ#2109997</a> , <a href="#">BZ#2065339</a> , <a href="#">BZ#2086869</a> , <a href="#">BZ#1996731</a> , <a href="#">BZ#2065670</a> , <a href="#">BZ#2043009</a> , <a href="#">BZ#2112143</a> , <a href="#">BZ#2066876</a> , <a href="#">BZ#2071011</a> , <a href="#">BZ#2075116</a> , <a href="#">BZ#2079008</a> , <a href="#">BZ#2086934</a> , <a href="#">BZ#2086935</a> , <a href="#">BZ#2093437</a> , <a href="#">BZ#2079114</a> , <a href="#">BZ#2056480</a> , <a href="#">BZ#2065215</a> , <a href="#">BZ#2065216</a> , <a href="#">BZ#2065218</a> , <a href="#">BZ#2100297</a> , <a href="#">BZ#2100939</a> , <a href="#">BZ#2100979</a> , <a href="#">BZ#2115159</a> , <a href="#">BZ#2115160</a> , <a href="#">BZ#2115162</a> , <a href="#">BZ#2021685</a> , <a href="#">BZ#2006081</a>
<b>rpm</b>	<a href="#">BZ#1688849</a>
<b>rsync</b>	<a href="#">BZ#2139118</a>
<b>rsyslog</b>	<a href="#">BZ#1962318</a> , <a href="#">BZ#1679512</a> , <a href="#">JIRA:RHELPLAN-10431</a>
<b>rust-toolset</b>	<a href="#">BZ#2075344</a>
<b>s390utils</b>	<a href="#">BZ#1660911</a>
<b>samba</b>	<a href="#">BZ#2077468</a> , <a href="#">BZ#2009213</a> , <a href="#">JIRA:RHELPLAN-13195</a> , <a href="#">JIRA:RHEDOCS-16612</a>
<b>sblim-wbemcli</b>	<a href="#">BZ#2075807</a>
<b>scap-security-guide</b>	<a href="#">BZ#2064696</a> , <a href="#">BZ#2075384</a> , <a href="#">BZ#2077531</a> , <a href="#">BZ#2078974</a> , <a href="#">BZ#2083109</a> , <a href="#">BZ#2109602</a> , <a href="#">BZ#2070564</a> , <a href="#">BZ#2058203</a> , <a href="#">BZ#1967947</a> , <a href="#">BZ#2032403</a> , <a href="#">BZ#2112937</a> , <a href="#">BZ#2028428</a> , <a href="#">BZ#1858866</a> , <a href="#">BZ#1750755</a> , <a href="#">BZ#2038977</a> , <a href="#">BZ#2119356</a>
<b>selinux-policy</b>	<a href="#">BZ#1461914</a>

コンポーネント	チケット
<b>sos</b>	<a href="#">BZ#2011413</a>
<b>spice</b>	<a href="#">BZ#1849563</a>
<b>sssd</b>	<a href="#">BZ#2065692</a> , <a href="#">BZ#2056483</a> , <a href="#">BZ#1947671</a>
<b>systemtap</b>	<a href="#">BZ#2057565</a>
<b>ubi8-container</b>	<a href="#">BZ#2120378</a>
<b>udica</b>	<a href="#">BZ#1763210</a>
<b>unbound</b>	<a href="#">BZ#2027735</a>
<b>vdo</b>	<a href="#">BZ#1949163</a>
<b>virt-manager</b>	<a href="#">BZ#2026985</a>
<b>vulkan-loader</b>	<a href="#">BZ#2012639</a>
<b>wayland</b>	<a href="#">BZ#1673073</a>
<b>weldr-client</b>	<a href="#">BZ#2033192</a>
<b>xmlstarlet</b>	<a href="#">BZ#1882020</a>
<b>xorg-x11-server</b>	<a href="#">BZ#1698565</a>
<b>xorg-x11-xtrans-devel</b>	<a href="#">BZ#2075132</a>

コンポーネント	チケット
その他	JIRA:RHELPLAN-109067, JIRA:RHELPLAN-115603, <a href="#">BZ#2020301</a> , <a href="#">BZ#2125545</a> , BZ#2128016, JIRA:RHELPLAN-121982, JIRA:RHELPLAN-118463, JIRA:RHELPLAN-100037, BZ#1497089, JIRA:RHELPLAN-121981, JIRA:RHELPLAN-121983, JIRA:RHELPLAN-121980, <a href="#">BZ#2049492</a> , JIRA:RHELPLAN-98420, JIRA:RHELPLAN-100039, JIRA:RHELPLAN-123369, JIRA:RHELPLAN-130379, JIRA:RHELPLAN-130376, JIRA:RHELPLAN-122735, <a href="#">BZ#2070793</a> , <a href="#">BZ#2122716</a> , JIRA:RHELPLAN-135602, JIRA:RHELPLAN-136150, <a href="#">BZ#2139821</a> , BZ#2025814, BZ#2077770, BZ#1777138, BZ#1640697, BZ#1697896, BZ#1961722, BZ#1659609, <a href="#">BZ#1687900</a> , BZ#1757877, BZ#1741436, JIRA:RHELPLAN-59111, JIRA:RHELPLAN-27987, JIRA:RHELPLAN-34199, JIRA:RHELPLAN-57914, JIRA:RHELPLAN-96940, BZ#1974622, BZ#2028361, <a href="#">BZ#2041997</a> , <a href="#">BZ#2035158</a> , JIRA:RHELPLAN-109613, BZ#2126777, BZ#1690207, JIRA:RHELPLAN-1212, BZ#1559616, <a href="#">BZ#1889737</a> , JIRA:RHELPLAN-14047, <a href="#">BZ#1769727</a> , JIRA:RHELPLAN-27394, JIRA:RHELPLAN-27737, <a href="#">BZ#1906489</a> , JIRA:RHELPLAN-75165, JIRA:RHELPLAN-118470, JIRA:RHELPLAN-122316, BZ#1642765, JIRA:RHELPLAN-10304, BZ#1646541, BZ#1647725, <a href="#">BZ#1932222</a> , <a href="#">BZ#1686057</a> , <a href="#">BZ#1748980</a> , JIRA:RHELPLAN-71200, BZ#1827628, JIRA:RHELPLAN-45858, <a href="#">BZ#1871025</a> , <a href="#">BZ#1871953</a> , BZ#1874892, BZ#1916296, JIRA:RHELPLAN-100400, <a href="#">BZ#1926114</a> , BZ#1904251, <a href="#">BZ#2011208</a> , JIRA:RHELPLAN-59825, <a href="#">BZ#1920624</a> , JIRA:RHELPLAN-70700, <a href="#">BZ#1929173</a> , JIRA:RHELPLAN-85066, <a href="#">BZ#2006665</a> , JIRA:RHELPLAN-98983, BZ#2009113, <a href="#">BZ#1958250</a> , <a href="#">BZ#2038929</a> , <a href="#">BZ#2029338</a> , <a href="#">BZ#2061288</a> , <a href="#">BZ#2060759</a> , BZ#2055826, <a href="#">BZ#2059626</a> , JIRA:RHELPLAN-133171, <a href="#">BZ#2142499</a>

## 付録B 改訂履歴

### 0.2-8

2024年5月15日(水) Brian Angelica ([bangelic@redhat.com](mailto:bangelic@redhat.com))

- [BZ#1690207](#) でテクニカルプレビューを更新しました。

### 0.2-7

2024年5月9日(木)、Gabriela Fialova ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 既知の問題 [BZ#1730502](#) (ストレージ) を更新しました。

### 0.2-6

2024年2月1日(木)、Gabriela Fialová ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 非推奨の機能 [JIRA:RHELDOCS-17641](#) (ネットワーク) を追加しました。

### 0.2-5

2024年2月7日(水)、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 非推奨の機能 [JIRA:RHELDOCS-17573](#) (アイデンティティ管理) を追加しました

### 0.2-4

2024年2月7日(水)、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 既知の問題 [BZ#1834716](#) (セキュリティ) を追加しました。

### 0.2-3

2023年11月10日金曜日、Gabriela Fialová ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- RHEL ドキュメントへのフィードバックの提供に関するモジュールを更新しました。

### 0.2-2

2023年10月13日(金) Gabriela Fialová ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- テクノロジープレビュー [JIRA:RHELDOCS-16861](#) (コンテナ) を追加しました。

### 0.2-1

2023年10月9日、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 既知の問題 [BZ#2169382](#) (ネットワーク) を更新しました。

### 0.2-0

2023年9月8日(金)、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 非推奨機能のリリースノート [JIRA:RHELDOCS-16612](#) (Samba) を追加しました。
- [Red Hat ドキュメントへのフィードバック](#) セクションを更新しました。

### 0.1-9

2023年8月24日(木) Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 既知の問題 [BZ#2214500](#) (セキュリティ) を追加しました。

- 既知の向題 [BZ#2214508](#) (カーネル) を追加しました。

#### 0.1-8

2023 年 8 月 4 日 金曜日 Lenka Špačková ([lspackova@redhat.com](mailto:lspackova@redhat.com))

- [BZ#2225332](#) のセクションを修正しました。

#### 0.1-7

2023 年 8 月 1 日 火曜日 Lenka Špačková ([lspackova@redhat.com](mailto:lspackova@redhat.com))

- 非推奨の機能 [BZ#2225332](#) を追加しました。
- 概要の改善

#### 0.1-6

2023 年 7 月 17 日 月曜日 Gabriela Fialová ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- [BZ#2072749](#) (システムロール) の間違いを修正しました。

#### 0.1-5

2023 年 6 月 29 日 木曜日 Marc Muehlfeld ([mmuehlfeld@redhat.com](mailto:mmuehlfeld@redhat.com))

- テクノロジープレビュー [BZ#1570255](#) (カーネル) を追加しました。

#### 0.1-4

2023 年 6 月 16 日 金曜日 Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 既知の問題 [BZ#2214235](#) (カーネル) を追加しました。

#### 0.1-3

2023 年 6 月 15 日 木曜日 Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 機能拡張 [BZ#2070347](#) (ブートローダー) を追加しました。

#### 0.1-3

2023 年 5 月 18 日 木曜日、ガブリエラフィアロヴァ ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 機能拡張 [BZ#2083077](#) (ファイルシステムとストレージ) を追加しました。

#### 0.1-2

2023 年 5 月 10 日 水曜日 Jaroslav Klech ([jklech@redhat.com](mailto:jklech@redhat.com))

- 既知の問題 [BZ#2169382](#) (ネットワーク) を追加しました。

#### 0.1-1

2023 年 4 月 27 日 (木) Gabriela Fialová ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- 既知の問題 [JIRA:RHELPLAN-155168](#) (アイデンティティ管理) を追加

#### 0.1-0

2023 年 4 月 18 日 火曜日 Lenka Špačková ([lspackova@redhat.com](mailto:lspackova@redhat.com))

- 非同期更新 [BZ#2178087](#) による機能強化が追加されました (動的プログラミング言語、Web およびデータベースサーバー)。

#### 0.0-9

2023 年 4 月 13 日 (木) Gabriela Fialová ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- DF と KI の 2 つの壊れたリンクを修正しました。

#### 0.0-8

2023 年 2 月 17 日、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 既知の問題 [BZ#2158783](#) (ファイルシステムおよびストレージ) を追加。

#### 0.0-7

2023 年 2 月 14 日、Marc Muehlfeld ([mmuehlfeld@redhat.com](mailto:mmuehlfeld@redhat.com))

- [BZ#2144898](#) (ネットワーク) の拡張機能を追加。

#### 0.0-6

2023 年 2 月 8 日、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- バグ修正 [BZ#2046396](#) (カーネル) および [BZ#2069379](#) (ID 管理) を追加しました。
- その他の若干の更新

#### 0.0-5

2023 年 1 月 24 日、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 既知の問題 [BZ#2115791](#) (クラウド環境の RHEL) を追加。

#### 0.0-4

2023 年 1 月 18 日、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 既知の問題 [BZ#1920086](#) (カーネル) を追加。

#### 0.0-3

2022 年 12 月 7 日、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- **nodejs:18** モジュールストリームの [BZ#2083073](#) をテクノロジープレビューから完全に対応した機能 (動的プログラミング言語、Web サーバー、およびデータベースサーバー) に移行しました。
- 既知の問題 [BZ#2132754](#) (ネットワーク) を追加。

#### 0.0-2

2022 年 11 月 23 日、Gabriela Fialová ([gfialova@redhat.com](mailto:gfialova@redhat.com))

- Directory Server リリースノートのリリースとその後の RHEL 8.7 リリースノートの再公開。

#### 0.0-1

2022 年 11 月 9 日、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 
- Red Hat Enterprise Linux 8.7 リリースノートも併せて参照してください。

**0.0-0**

2022年9月28日、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- Red Hat Enterprise Linux 8.7 Beta リリースノートのリリース。