



# Red Hat Enterprise Linux 8

## 8.6 リリースノート

Red Hat Enterprise Linux 8.6 リリースノート



# Red Hat Enterprise Linux 8 8.6 リリースノート

---

Red Hat Enterprise Linux 8.6 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 8.6 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 8.6 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	8
1.3. RED HAT CUSTOMER PORTAL LABS	9
1.4. 関連情報	10
<b>第2章 アーキテクチャー</b> .....	<b>11</b>
<b>第3章 RHEL 8 のコンテンツの配布</b> .....	<b>12</b>
3.1. インストール	12
3.2. リポジトリ	12
3.3. アプリケーションストリーム	13
3.4. YUM/DNF を使用したパッケージ管理	13
<b>第4章 新機能</b> .....	<b>14</b>
4.1. インストーラーおよびイメージの作成	14
4.2. RHEL FOR EDGE	14
4.3. サブスクリプションの管理	15
4.4. ソフトウェア管理	15
4.5. シェルおよびコマンドラインツール	16
4.6. インフラストラクチャーサービス	18
4.7. セキュリティー	19
4.8. ネットワーク	25
4.9. カーネル	27
4.10. ファイルシステムおよびストレージ	29
4.11. 高可用性およびクラスター	31
4.12. 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
4.13. コンパイラーおよび開発ツール	33
4.14. IDENTITY MANAGEMENT	38
4.15. デスクトップ	41
4.16. グラフィックインフラストラクチャー	41
4.17. WEB コンソール	42
4.18. RED HAT ENTERPRISE LINUX システムロール	43
4.19. 仮想化	47
4.20. クラウド環境の RHEL	48
4.21. サポート性	49
4.22. コンテナ	50
<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>53</b>
新しいカーネルパラメーター	53
更新されたカーネルパラメーター	54
<b>第6章 デバイスドライバー</b> .....	<b>58</b>
6.1. 新しいドライバー	58
6.2. 更新されたドライバー	58
<b>第7章 バグ修正</b> .....	<b>60</b>
7.1. インストーラーおよびイメージの作成	60
7.2. ソフトウェア管理	60

7.3. シェルおよびコマンドラインツール	61
7.4. セキュリティー	62
7.5. ネットワーク	65
7.6. カーネル	65
7.7. ファイルシステムおよびストレージ	66
7.8. コンパイラーおよび開発ツール	67
7.9. IDENTITY MANAGEMENT	68
7.10. グラフィックインフラストラクチャー	70
7.11. RED HAT ENTERPRISE LINUX システムロール	70
7.12. 仮想化	74
7.13. コンテナ	74
<b>第8章 テクノロジープレビュー</b>	<b>76</b>
8.1. RHEL FOR EDGE	76
8.2. シェルおよびコマンドラインツール	76
8.3. ネットワーキング	76
8.4. カーネル	78
8.5. ファイルシステムおよびストレージ	79
8.6. 高可用性およびクラスター	82
8.7. IDENTITY MANAGEMENT	82
8.8. デスクトップ	84
8.9. グラフィックインフラストラクチャー	85
8.10. WEB コンソール	85
8.11. 仮想化	85
8.12. コンテナ	87
<b>第9章 非推奨になった機能</b>	<b>88</b>
9.1. インストーラーおよびイメージの作成	88
9.2. ソフトウェア管理	89
9.3. シェルおよびコマンドラインツール	89
9.4. セキュリティー	90
9.5. ネットワーキング	92
9.6. カーネル	93
9.7. ブートローダー	94
9.8. ファイルシステムおよびストレージ	94
9.9. 高可用性およびクラスター	96
9.10. 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	97
9.11. コンパイラーおよび開発ツール	97
9.12. IDENTITY MANAGEMENT	97
9.13. デスクトップ	101
9.14. グラフィックインフラストラクチャー	101
9.15. WEB コンソール	101
9.16. RED HAT ENTERPRISE LINUX システムロール	102
9.17. 仮想化	103
9.18. コンテナ	104
9.19. 非推奨のパッケージ	105
9.20. 非推奨のデバイスおよび非保守のデバイス	140
<b>第10章 既知の問題</b>	<b>143</b>
10.1. インストーラーおよびイメージの作成	143
10.2. サブスクリプションの管理	145
10.3. ソフトウェア管理	145
10.4. シェルおよびコマンドラインツール	146
10.5. インフラストラクチャーサービス	146

---

10.6. セキュリティー	146
10.7. ネットワーク	152
10.8. カーネル	153
10.9. ファイルシステムおよびストレージ	159
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	161
10.11. IDENTITY MANAGEMENT	162
10.12. デスクトップ	165
10.13. グラフィックインフラストラクチャー	165
10.14. WEB コンソール	167
10.15. RED HAT ENTERPRISE LINUX システムロール	167
10.16. 仮想化	167
10.17. クラウド環境の RHEL	171
10.18. サポート性	174
10.19. コンテナ	175
<b>第11章 国際化</b> .....	<b>176</b>
11.1. RED HAT ENTERPRISE LINUX 8 の多言語	176
11.2. RHEL 8 における国際化の主な変更点	176
<b>付録A コンポーネント別のチケットリスト</b> .....	<b>178</b>
<b>付録B 改訂履歴</b> .....	<b>185</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 8.6 における主な変更点

### セキュリティー

RHEL 8.6 では、**fapolicyd** ポリシーフレームワークである SELinux、および LUKS 暗号化ドライブの自動ロック解除のための Policy-Based Decryption (PBD) が、SAP HANA データベース管理システムをサポートします。詳細については、ナレッジベースの記事である [Red Hat Enterprise Linux Security Hardening Guide for SAP HANA 2.0](#) を参照してください。

**fapolicyd** のパッケージがアップストリームバージョン 1.1 にアップグレードされました。その他の改善点の中でも、新しい **rules.d/** および **trust.d/** ディレクトリー、**fagenrules** スクリプト、および **fapolicyd-cli** コマンドの新しいオプションを使用できるようになりました。

OpenSSH サーバーは、ドロップイン設定ファイルをサポートするようになりました。

**pcsc-lite** パッケージは、多くの機能拡張とバグ修正を提供するアップストリームバージョン 1.9.5 にリベースされました。

これで、**semodule** コマンドに新しく追加された **--checksum** オプションを使用して、インストールされている SELinux ポリシーモジュールのバージョンを確認できるようになりました。

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 0.1.60 にリベースされ、OpenSCAP パッケージはアップストリームバージョン 1.3.6 にリベースされました。

詳細は、[新機能 - セキュリティー](#) 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。

- PHP 8.0
- Perl 5.32

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。

### コンパイラーおよび開発ツール

#### 更新されたコンパイラーツールセット

以下のコンパイラーツールセットが更新されました。

- GCC Toolset 11
- LLVM Toolset 13.0.1
- Rust Toolset 1.58.1
- Go Toolset 1.17.7

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

### RHEL 8 の Java 実装

RHEL 8 AppStream リポジトリーには、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 8.6 では、新しい **log4j:2** モジュールが導入されています。このモジュールには、Java ロギングユーティリティである **Apache Log4j 2** と、さまざまな出力ターゲットにログステートメントを出力できるライブラリーが含まれています。

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

### Identity Management

**ansible-freeipa** のロールとモジュールが Ansible Automation Hub で利用できるようになりました。これにより、**ansible-freeipa** コンテンツの高速更新が提供されます。

## 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

現在サポートされているインプレースアップグレードパスは次のとおりです。

- 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 8.2 にアップグレードした後に SAP HANA のシステムに対応していることを確認するには、RHEL 8.2 Update Services for SAP Solutions (E4S) リポジトリを有効にします。

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。

主な機能拡張は、次のとおりです。

- RHEL 8.6 のリリースにより、RHEL 7 から RHEL 8 へのインプレースアップグレードに複数のアップグレードパスが利用できるようになりました。これにより、デフォルトで最新の RHEL 8 マイナーバージョンにアップグレードする代わりに、どの RHEL 8 マイナーバージョンにシステムをアップグレードするか決定できます。使用可能なアップグレードパスは、RHEL システムと SAP HANA 搭載 RHEL システムで異なることに注意してください。
- **Leapp** ユーティリティは、アップグレード前およびインプレースアップグレードの初期段階で大幅に高速に実行されるようになりました。

- インプレースアップグレードは、次のクラウドイメージタイプの SAP ホスティングシステムでもサポートされています。
  - RHEL サブスクリプションに Red Hat Subscription Manager (RHSM) を使用するパブリッククラウドプラットフォーム上の Bring-your-own-subscription (BYOS) システム。
  - Red Hat Update Infrastructure (RHUI) を使用する Amazon Web Services (AWS) および Microsoft Azure の Pay-as-you-go (PAYG) インスタンス。

### 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 8.6 には、カーネルバージョン 4.18.0-372 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[の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

Binary DVD ISO イメージが 4.7 GB を超え、1 層 DVD に収まらない場合があります。Binary DVD 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、2 層 DVD または USB キーが推奨されます。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。

### 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 と同じく、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 8.6 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

**Image Builder は、LVM でカスタマイズされたファイルシステムパーティションをサポートします**

この機能拡張により、複数のパーティションがある場合、LVM でカスタマイズされたファイルシステムパーティションを使用してイメージを作成し、実行時にそれらのパーティションのサイズを変更できます。そのために、ブループリントでカスタマイズされたファイルシステム設定を指定してから、目的のディスクレイアウトでイメージを作成できます。デフォルトのファイルシステムレイアウトは変更されません。ファイルシステムをカスタマイズせずにプレーンイメージを使用する場合、ルートパーティションは **cloud-init** によってサイズ変更されます。

(JIRA:RHELPLAN-102505)

### 4.2. RHEL FOR EDGE

**RHEL for Edge は、デフォルトで Greenboot ビルトインヘルスチェックをサポートするようになりました**

この更新により、RHEL for Edge **Greenboot** には、再起動中にハードウェアがハングアップまたはフリーズしないようにする **watchdog** 機能を備えたビルトインヘルスチェックが含まれるようになりました。これにより、次の機能を利用できます。

- **watchdogs** ハードウェアユーザーがビルトインヘルスチェックを簡単に導入できるようになります
- ビルトイン OS コンポーネントに価値を提供するデフォルトのヘルスチェックのセット
- **watchdog** がデフォルトのプリセットとして表示されるようになるため、この機能を簡単に有効または無効にできます。
- すでに利用可能なヘルスチェックに基づいてカスタムヘルスチェックを作成する機能。

(BZ#2083036)

**RHEL 8 を rpm-ostreev2022.2 にリベースしました**

RHEL 8 は、**rpm-ostree** バージョン v2022.2 とともに配布され、複数のバグ修正と機能強化が提供されます。主な変更点は、以下のとおりです。

- 新しい **--append-if-missing** フラグおよび **--delete-if-present** kargs フラグを使用することで、カーネル引数を idempotent 形式で更新できるようになりました。
- YUM の **Count Me** 機能が、すべてのリポジトリクエリーでデフォルトで完全に無効になり、対応する **rpm-ostree-countme.timer** ユニットおよび **rpm-ostree-countme.service** ユニットによってのみトリガーされるようになりました。 [countme](#) を参照してください。
- 後処理ロジックで、**user.ima** IMA 拡張アトリビュートを処理できるようになりました。**xattr** 拡張属性が見つかったら、最終的な **OSTree** パッケージコンテンツの **security.ima** に自動的に変換されます。

- **treefile** には、新しい **repo-packages** 項目があります。これを使用して、一連のパッケージを特定のレポジトリに固定できます。
- Compose とクライアント側でモジュール性を使用する機能。
- コンテナイメージは、作成ターゲットおよびアップグレードソースとして使用されるようになりました。

(BZ#2032594)

### 4.3. サブスクリプションの管理

#### subscription-manager syspurpose で統合されたシステム目的のコマンド

以前は、システムの目的に関連する属性を設定するための複数の subscription-manager モジュール (**addons**、**role**、**service-level**、および **usage**) がありました。これらのモジュールは、新しい **subscription-managersyspurpose** モジュールに移動しました。

元の subscription-manager モジュール (**addons**、**role**、**service-level**、および **usage**) は非推奨になりました。さらに、**syspurpose** コマンドラインツールを提供するパッケージ (**python3-syspurpose**) は、RHEL 8.6 で非推奨になりました。このパッケージのすべての機能は、新しい **subscription-manager syspurpose** モジュールでカバーされています。

この更新により、subscription-manager の1つのコマンドを使用して、すべてのシステムの目的属性を表示、設定、および更新する方法が統一されました。これにより、既存のシステムの目的のコマンドがすべて、新しいサブコマンドとして利用可能な同等のバージョンに置き換わります。たとえば、**subscription-manager role --set SystemRole** は **subscription-manager syspurpose role --set SystemRole** になります。

新しいコマンド、オプション、およびその他の属性の詳細は、**subscription-manager** の man ページの **SYSPURPOSE OPTIONS** セクションを参照してください。

(BZ#2000883)

### 4.4. ソフトウェア管理

#### modulesync コマンドを使用して、RHEL8 の特定のワークフローを置き換えることができますようになりました

Red Hat Enterprise Linux 8 では、モジュラーメタデータなしでモジュラーパッケージをインストールすることはできません。以前は、**yum** コマンドを使用してパッケージをダウンロードしてから、**createrepo\_c** コマンドを使用してそれらのパッケージを再配布できました。

この機能拡張により、**modulesync** コマンドが導入され、モジュラーメタデータの存在が保証され、パッケージのインストールが保証されます。このコマンドは、モジュールから **rpm** パッケージをダウンロードし、作業ディレクトリーにモジュラーメタデータを含むレポジトリを作成します。

(BZ#1868047)

#### 新しい --path CLI オプションが RPM に追加されました

この更新により、新しい **--path** CLI オプションを使用して、現在インストールされていないファイルでパッケージをクエリーできます。このオプションは既存の **--file** オプションに似ていますが、指定されたパスのみに基づいてパッケージと一致します。そのパスにあるファイルはディスク上に存在する必要はないことに注意してください。

**--path** CLI オプションは、ユーザーが **yum** で **--nodocs** オプションを使用して、インストール時にすべてのドキュメントファイルを除外する場合に役立ちます。この場合は、**--path** オプションを使用すると、そのような除外されたファイルの所有パッケージを表示できますが、**--file** オプションは、要求されたファイルが存在しないため、パッケージを表示しません。

(BZ#1940895)

## 4.5. シェルおよびコマンドラインツール

### lsupd パッケージがバージョン 1.7.13 にリベース

**lsupd** パッケージはバージョン 1.7.13 にリベースされる主なバグ修正と機能拡張は、以下のとおりです。

- SCSI ロケーションコードのサポートが追加されました。
- **sysfstreecollector** の絶対パス **getDevTreePath** の長さを修正しました。

(BZ#1993557)

### net-snmp-cert gencert ツールは、SHA1ではなく SHA512 暗号化アルゴリズムを使用するようになる

セキュリティを強化するために、**net-snmp-cert gencert** ツールが更新され、デフォルトで SHA512 暗号化アルゴリズムを使用して証明書が生成されるようになりました。

(BZ#1908331)

### dnn および text モジュールは opencv パッケージで利用可能

イメージ分類推論用の Deep Neural Network を含む **dnn** モジュールと、シーンテキストの検出と認識用の **text** モジュールが **opencv** パッケージで利用できるようになりました。

(BZ#2007780)

### powerpc-utils パッケージがバージョン 1.3.9 にリベース

**powerpc-utils** パッケージがバージョン 1.3.9 にアップグレード主なバグ修正と機能強化は、以下のとおりです。

- **drmgr** でログサイズを 1MB に増やしました。
- 起動時の **HCNID** 配列サイズのチェックを修正しました。
- **hcngr** で HNV 接続に **autoconnect-slaves** を実装しました。
- **hcngr** で、HNV ボンディングリスト接続が改善されました。
- **hcnmgr** の **vim** からの **xxd** の代わりに **util-linux** の **hexdump** を使用します。
- **hcn-init.service** は NetworkManager と一緒に起動します。
- **ofpathname** でマルチパスの論理 FC ルックアップを修正しました。
- **ofpathname** でパーティションを持つ論理ルックアップに対する OF を修正しました。
- 5 つを超えるパスを持つマルチパスデバイスのブートリストを修正しました。

- 実行中の LPAR の NUMA アフィニティースコアを検出するための **lparnumascore** コマンドが導入されました。
- セキュリティーを強化するために、**lpartstat** に **-x** オプションを追加しました。
- **hcnggr** で **udev** 名が変更された **ofpathname** 競合を修正しました。
- HNV の **qrydev** を修正し、**lsdevinfo** を削除しました。

(BZ#2028690)

**powerpc-utils** パッケージは、バックアップデバイスとして vNIC をサポートするようになる

**powerpc-utils** パッケージは、Hybrid Network Virtualization (HNV) のバックアップ **vdevice** として仮想ネットワークインターフェイスカード (vNIC) をサポートするようになりました。

(BZ#2022225)

**opencryptoki** パッケージがバージョン 3.17.0 にリベース

**opencryptoki** パッケージはバージョン 3.17.0 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。

- **p11sak** ツールは、キーをリスト表示する新しい機能を提供します。
- **OpenSSL 3.0** のサポートが追加される
- イベント通知のサポートが追加されました。
- ICA トークンに SW フォールバックを追加しました。
- WebSphere Application Server は、ハードウェア暗号アダプターを使用可能にして起動に失敗しなくなりました。
- **opencryptoki.module** が削除され、**p11-kit list-modules** コマンドでエラーメッセージが表示されなくなりました。

(BZ#1984993)

レスキューイメージを作成するときに、特定のネットワークインターフェイスと IP アドレスを除外できます

**EXCLUDE\_IP\_ADDRESSES** 変数を使用して特定の IP アドレスを無視し、**EXCLUDE\_NETWORK\_INTERFACES** 変数を使用してレスキューイメージを作成するときに特定のネットワークインターフェイスを無視することができます。

フローティングアドレスを持つサーバーでは、元のサーバーが復旧するまで、ReaR レスキュー環境がフェイルオーバーサーバーに移動されるフローティングアドレスを設定しないようにする必要があります。そうしないと、フェイルオーバーサーバーとの競合が発生し、その結果、フェイルオーバーサーバーで実行しているサービスが中断されます。競合を防ぐために、ReaR 設定ファイル **/etc/rear/local.conf** で次のアクションを実行できます。

- **EXCLUDE\_IP\_ADDRESSES** 変数をアドレスのバッシュ配列として提供することにより、ReaR の IP アドレスを除外します。たとえば、**EXCLUDE\_IP\_ADDRESSES=( 192.0.2.27 192.0.2.10 )** です。

- **EXCLUDE\_NETWORK\_INTERFACES** 変数をインターフェイスのバッシュ配列として提供することにより、ReaR のネットワークインターフェイスを除外します。例:  
**EXCLUDE\_NETWORK\_INTERFACES =(eno1d1)**。

(BZ#2035939)

## 4.6. インフラストラクチャーサービス

### 新しい **bind9.16** パッケージバージョン 9.16.23 が導入されました

**bind** コンポーネントバージョン 9.11.36 の代わりに、新しい **bind9.16** パッケージバージョン 9.16.23 が導入されました。主な機能拡張は、次のとおりです。

- DNSSEC に新しいキーおよび署名ポリシー機能が導入されました。
- プライバシーを改善するために QNAME 最小化を導入しました。
- パーマネントに **validate-except** 機能を導入しました。
- DNSSEC 検証を一時的に無効にするネガティブトラストアンカー。
- 応答ポリシーゾーン (RPZ) をリファクタリングしました。
- ゾーンタイプの新しい命名規則が導入されました。プライマリーゾーンタイプとセカンダリーゾーンタイプは、マスターとスレーブの同義語として使用されます。
- **dig**、**mdig**、および **delv** コマンドの補足 YAML 出力モードが導入されました。
- **filter-aaaa** 機能は、別々の **filter-a** プラグインと **filter-aaaa** プラグインに移動しました。
- 新しいゾーンタイプのミラーサポートが導入されました (RFC 8806)。

削除された機能:

- **dnssec-enabled** オプションは削除され、DNSSEC はデフォルトで有効になり、**dnssec-enabled** キーワードは受け入れられなくなりました。
- **lwresd** 軽量リゾルバーデーモンと **liblwres** 軽量リゾルバーライブラリーが削除されました。

(BZ#1873486)

### CUPS はコンテナイメージとして利用可能です

Common Unix Printing System (CUPS) がコンテナイメージとして利用可能になり、Red Hat Container Catalog からデプロイできるようになりました。

(BZ#1913715)

### **bind** コンポーネントがバージョン 9.11.36 にリベースされました

**bind** コンポーネントがバージョン 9.11.36 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。

- より安全になるように **lame-ttl** オプションが改善されました。
- RBTDB インスタンスに影響を与える複数スレッドのバグにより、**free\_rbtodb()** でアサーションが失敗することはなくなりました。



- RFC 8976 に一致するように ZONEMD RR タイプの実装を更新しました。
- NSEC3 反復の最大サポート回数は 150 回に減りました。反復回数が多いレコードは、安全でないものとして扱われます。
- LOC レコードの無効な方向フィールドが失敗することはなくなりました。

(BZ#2013993)

## CUPS ドライバーレス印刷は CUPS Web UI で利用可能

IPP Everywhere モデルに基づく CUPS ドライバーレス印刷は、CUPS Web UI で使用できます。CLI で使用される **lpadmin** コマンドに加えて、CUPS Web UI に IPP Everywhere キューを作成して、特別なソフトウェアなしでネットワークプリンターに印刷できます。

(BZ#2032965)

## 4.7. セキュリティー

### pcsc-lite パッケージが 1.9.5 にリベースされました

**pcsc-lite** パッケージはアップストリームバージョン 1.9.5 にリベースされました。この更新では、特に次のような新しい拡張機能とバグ修正が提供されます。

- **pcscd** デーモンは、手動で起動したときに非アクティブになった後、自動的に終了しなくなりました。
- **pcsc-spy** ユーティリティーは、Python 3 と新しい **--thread** オプションをサポートするようになりました。
- **SCardEndTransaction()** 関数のパフォーマンスが改善されました。
- **poll()** 関数は **select()** 関数に取って代わりました。これにより、**FD\_SETSIZE** よりも大きいファイル記述子番号が許可されます。
- 多くのメモリーリークと同時実行の問題が修正されました。

(BZ#2014641)

### 暗号化ポリシーは **diffie-hellman-group14-sha256** をサポートする

これで、RHEL システム全体の暗号化ポリシーの **libssh** ライブラリーに **diffie-hellman-group14-sha256** キー交換 (KEX) アルゴリズムを使用できます。この更新は、この KEX アルゴリズムもサポートする OpenSSH との同等性も提供します。この更新により、**libssh** ではデフォルトで **diffie-hellman-group14-sha256** が有効になりますが、カスタム暗号化ポリシーを使用して無効にすることができます。

(BZ#2023744)

### OpenSSH サーバーがドロップイン設定ファイルをサポートするようになりました

**sshd\_config** ファイルは **include** ディレクティブをサポートします。これは、設定ファイルを別のディレクトリーに含めることができることを意味します。これにより、Ansible Engine などの自動化ツールを使用して、OpenSSH サーバーにシステム固有の設定を簡単に適用できます。また、**ssh\_config** ファイルの機能との整合性も高くなっています。さらに、ドロップイン設定ファイルを使用すると、着信接続のフィルター処理など、さまざまな用途に合わせてさまざまな設定ファイルを簡単に整理できます。

(BZ#1926103)

## sshd\_config:ClientAliveCountMax=0 は接続終了を無効にします

SSHD 設定オプション **ClientAliveCountMax** を **0** に設定すると、接続の終了が無効になります。これにより、このオプションの動作がアップストリームと一致します。その結果、OpenSSH は、**ClientAliveInterval** オプション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。

(BZ#2015828)

## libssh が 0.9.6 にリベース

**libssh** パッケージは、アップストリームバージョン 0.9.6 にリベースされました。このバージョンでは、主なバグ修正および機能強化が数多く追加されました。

- 複数の ID ファイルのサポート。ファイルは、`~/.ssh/config` ファイルにリストされているように、下から上に処理されます。
- SFTP での 1 秒未満の時間の解析が修正されました。
- **SSH\_AGAIN** を予期せず返す `ssh_channel_poll_timeout()` 関数のリグレッションが修正されました。
- キーの再交換が修正された後に発生する可能性のあるヒープバッファオーバーフロー。
- AEAD 暗号が一致しているが、HMAC の重複がない場合のハンドシェイクのバグが修正されました。
- エラーパスでのいくつかのメモリーリークが修正されました。

(BZ#1896651)

## Libreswan を 4.5 にリベース

Libreswan はアップストリームバージョン 4.5 にリベースされるこのバージョンでは、主なバグ修正および機能強化が数多く追加されました。

- ラベル付き IPsec 用のインターネットキーエクスチェンジバージョン 2 (IKEv2) のサポート。
- インターネットキーエクスチェンジ (IKE) セキュリティアソシエーション (SA) の子なしの開始のサポート。

(BZ#2017352)

## SELinux モジュールのチェックサムを検証するための新しいオプション

**semodule** コマンドに新しく追加された **--checksum** オプションを使用すると、インストールされている SELinux ポリシーモジュールのバージョンを確認できます。

Common Intermediate Language (CIL) はモジュール名とモジュールバージョンをモジュール自体に格納しないため、以前は、インストールされたモジュールがインストールされるはずのモジュールと同じバージョンであることを確認する簡単な方法はありませんでした。

新しいコマンド **semodule -l --checksum** を使用すると、指定したモジュールの SHA256 ハッシュを受け取り、それを元のファイルのチェックサムと比較できます。これは、モジュールを再インストールするよりも高速です。

使用例:



```
# semodule -l --checksum | grep localmodule
localmodule sha256:db002f64ddfa3983257b42b54da7b182c9b2e476f47880ae3494f9099e1a42bd

# /usr/libexec/selinux/hll/pp localmodule.pp | sha256sum
db002f64ddfa3983257b42b54da7b182c9b2e476f47880ae3494f9099e1a42bd -
```

(BZ#1731501)

### OpenSCAP はローカルファイルを読み取ることができます

OpenSCAP は、リモート SCAP ソースデータストリームコンポーネントの代わりにローカルファイルを使用できるようになりました。以前は、インターネットにアクセスできないシステムで、リモートコンポーネントを含む SCAP ソースデータストリームの完全な評価を実行できませんでした。これらのシステムでは、リモートコンポーネントをインターネットからダウンロードする必要があるため、OpenSCAP はこれらのデータストリームの一部のルールを評価できませんでした。この更新により、OpenSCAP スキャンを実行する前にリモート SCAP ソースデータストリームコンポーネントをダウンロードしてターゲットシステムにコピーし、**oscap** コマンドで **--local-files** オプションを使用してそれらを OpenSCAP に提供できます。

(BZ#1970529)

### SSG は、ホームディレクトリーとインタラクティブユーザーのルールをスキャンして修正するようになりました

対話型ユーザーが使用するホームディレクトリーに関連するすべての既存のルールをチェックおよび修正するための OVAL コンテンツが、SCAP セキュリティーガイド (SSG) スイートに追加されました。多くのベンチマークでは、インタラクティブユーザーのホームディレクトリー内に通常見られるプロパティーとコンテンツの検証が必要です。システム内のインタラクティブユーザーの存在と数はさまざまである可能性があるため、これまで、OVAL 言語を使用してこのギャップを埋める堅牢なソリューションはありませんでした。この更新により、システム内のローカルの対話型ユーザーとそれぞれのホームディレクトリーを検出する OVAL チェックと修復が追加されます。その結果、SSG は、関連するすべてのベンチマーク要件を安全にチェックおよび修正できます。

(BZ#1884687)

### SCAP ルールに、大規模システムの監査ログバッファを設定するための警告メッセージが表示されるようになりました

SCAP ルール **xccdf\_org.ssgproject.content\_rule\_audit\_basic\_configuration** は、このルールによって設定された監査ログバッファが小さすぎてカスタム値を上書きできる大規模システムのユーザーを示唆するパフォーマンス警告を表示するようになりました。この警告は、より大きな監査ログバッファを設定するプロセスについても説明しています。この機能拡張により、大規模システムのユーザーはコンプライアンスを維持し、監査ログバッファを正しく設定できます。

(BZ#1993826)

### SSG が `/etc/security/faillock.conf` ファイルをサポートするようになりました

この機能拡張により、SCAP セキュリティーガイド (SSG) の `/etc/security/faillock.conf` ファイルのサポートが追加されます。この更新により、SSG は **pam\_faillock** 設定の定義について `/etc/security/faillock.conf` ファイルを評価および修正できます。**authselect** ツールは、**pam** ファイルの整合性を確保しながら **pam\_faillock** モジュールを有効にするためにも使用されます。その結果、**pam\_faillock** モジュールの評価と修正は、最新バージョンとベストプラクティスに沿ったものになります。

(BZ#1956972)

## SCAP セキュリティーガイドが 0.1.60 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 0.1.60 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。

- 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。
- DISA STIG 自動 SCAP コンテンツと SCAP 自動コンテンツ (デルタ調整) の違いを表すプロファイルを定義する調整ファイルがサポートされるようになりました。
- ルール **xccdf\_org.ssgproject.content\_enable\_fips\_mode** は、FIPS モードが正しく有効になっているかどうかのみをチェックするようになりました。システムコンポーネントが FIPS 認定を受けていることを保証するものではありません。

(BZ#2014485)

## DISA STIG プロファイルは Red Hat Virtualization 4.4 をサポートします

**DISA STIG for Red Hat Enterprise Linux 8** プロファイルバージョン V1R5 は、Red Hat Virtualization 4.4 をサポートするように拡張されました。このプロファイルは、国防情報システム局 (DISA) が提供する RHEL 8 セキュリティー技術実装ガイド (STIG) の手動ベンチマークに準拠しています。ただし、一部の設定は、Red Hat Virtualization (RHV) がインストールされているホストには適用されません。これは、Red Hat Virtualization のインストールと正常な動作が妨げられるためです。

STIG プロファイルが Red Hat Virtualization ホスト (RHVH)、セルフホストインストール (RHELH)、または RHV Manager がインストールされているホストに適用される場合、次のルールは適用外になります。

- **package\_gss\_proxy\_removed**
- **package\_krb5-workstation\_removed**
- **package\_tuned\_removed**
- **sshd\_disable\_root\_login**
- **sudo\_remove\_nopasswd**
- **sysctl\_net\_ipv4\_ip\_forward**
- **xwindows\_remove\_packages**



### 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#2021802)

## OpenSCAP が 1.3.6 にリベースされました。

OpenSCAP パッケージがアップストリームバージョン 1.3.6 にリベースされました。このバージョンは、さまざまなバグ修正と機能拡張を提供します。特に、次のとおりです。

- **--local-files** オプションを使用して、リモート SCAP ソースデータストリームコンポーネントのローカルコピーを提供できます。
- OpenSCAP は、複数の **--rule** 引数を受け入れて、コマンドラインで複数のルールを選択します。
- OpenSCAP では、**--skip-rule** オプションを使用して一部のルールの評価をスキップできます。
- **OSCAP\_PROBE\_MEMORY\_USAGE\_RATIO** 環境変数を使用して、OpenSCAP プロブによって消費されるメモリーを制限できます。
- OpenSCAP は、修復タイプとして OSBuild ブループリントをサポートするようになりました。

(BZ#2041781)

### clevis-systemd は nc に依存しなくなりました

この機能拡張により、**clevis-systemd** パッケージは **nc** パッケージに依存しなくなりました。Extra Packages for Enterprise Linux (EPEL) で使用すると、依存関係が正しく機能しませんでした。

(BZ#1949289)

### audit が 3.0.7 にリベースされました。

**audit** パッケージはバージョン 3.0.7 にアップグレードされ、多くの機能拡張とバグ修正が行われました。以下に例を示します。

- Audit の基本ルールに **sudoers** を追加しました。
- **--eoe-timeout** オプションを **ausearch** コマンドに追加し、それに類似した **eoe\_timeout** オプションを **auditd.conf** ファイルに追加しました。これは、イベント終了タイムアウトの値を指定し、**ausearch** が同じ場所にあるイベントを解析する方法に影響を与えます。
- リモートロケーションが利用できないときに CPU 容量の 100% を使用する **audisp-remote** プラグインの修正を導入しました。

(BZ#1939406)

### Audit は、イベントタイムアウトの終了を指定するためのオプションを提供するようになりました

このリリースでは、**ausearch** ツールは **--eoe-timeout** オプションをサポートし、**auditd.conf** ファイルには **end\_of\_event\_timeout** オプションが含まれています。これらのオプションを使用して、イベントタイムアウトの終了を指定し、同じ場所に配置されたイベントの解析に関する問題を回避できます。イベントタイムアウトの終了のデフォルト値は 2 秒に設定されています。

(BZ#1921658)

### Audit の基本ルールに sudoer を追加します。

この機能拡張により、**/etc/sudoers** および **etc/sudoers.d/** ディレクトリーが、Payment Card Industry Data Security Standard (PCI DSS) や Operating Systems Protection Profile (OSPP) などの Audit 基本ルールに追加されます。そのため、**sudoers** などの特権領域の設定変更を監視することでセキュリティが強化されます。

(BZ#1927884)

## Rsyslog には、より高性能な操作と CEF のための mmfields モジュールが含まれます

Rsyslog には、**mmfields** モジュールを提供する **rsyslog-mmfields** サブパッケージが含まれるようになりました。これは、プロパティー置き換えフィールド抽出を使用する代わりにの方法ですが、プロパティー置き換えとは対照的に、すべてのフィールドが一度に抽出され、構造化データ部分の内部に格納されます。その結果、特に Common Event Format (CEF) などのログ形式を処理する場合や、多数のフィールドが必要であったり特定のフィールドを再使用したりする場合などに、**mmfields** を使用できます。このような場合の **mmfields** のパフォーマンスは、既存の Rsyslog 機能よりも優れています。

(BZ#1947907)

## libcap がバージョン 2.48 にリベース

**libcap** パッケージがアップストリームバージョン 2.48 にアップグレードされ、以前のバージョンのバグ修正および機能拡張が数多く追加されました。主な改善点は以下の通りです。

- POSIX セマンティックシステムコール用のヘルパーライブラリー (**libpsx**)
- システムコール関数のオーバーライドのサポート
- 機能セットの IAB 抽象化
- 追加の **capsh** テスト機能

(BZ#2032813)

## fapolicyd が 1.1 にリベースされました。

**fapolicyd** パッケージは、多くの改善とバグ修正を含むアップストリームバージョン 1.1 にアップグレードされました。最も注目すべき変更点は次のとおりです。

- 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。詳細については、新しい **fagenrules (8)** の man ページを参照してください。
- RPM データベース外のファイルを信頼できるものとしてマークするための **/etc/fapolicyd/fapolicyd.trust** ファイルに加えて、信頼できるファイルのリストをより多くのファイルに分割することをサポートする新しい **/etc/fapolicyd/trust.d** ディレクトリーを使用できるようになりました。これらのファイルに **--trust-file** ディレクティブを指定して **fapolicyd-cli-f** サブコマンドを使用して、ファイルのエントリーを追加することもできます。詳細については、**fapolicyd-cli(1)** および **fapolicyd.trust(13)** の man ページを参照してください。
- **fapolicyd trust** データベースは、ファイル名の空白をサポートするようになりました。
- **fapolicyd** は、ファイルを信頼データベースに追加するときに、実行可能ファイルへの正しいパスを格納するようになりました。

(BZ#1939379)

## libseccomp が 2.5.2 にリベース

**libseccomp** パッケージは、アップストリームバージョン 2.5.2 にリベースされました。このバージョンでは、主なバグ修正および機能強化が数多く追加されました。

- Linux の syscall テーブルがバージョン **v5.14-rc7** に更新されました。
- 通知ファイル記述子を取得するために、**get\_notify\_fd()** 関数が Python バインディングに追加されました。
- すべてのアーキテクチャーの多重化されたシステムコール処理が1つの場所に統合されました。
- 多重化されたシステムコールのサポートが、PowerPC (PPC) および MIPS アーキテクチャーに追加されました。
- カーネル内で **SECCOMP\_IOCTL\_NOTIF\_ID\_VALID** 操作の意味が変更されました。
- **libseccomp** ファイル記述子通知ロジックは、カーネルの以前および新しい **SECCOMP\_IOCTL\_NOTIF\_ID\_VALID** の使用をサポートするように変更されました。

(BZ#2019893)

## 4.8. ネットワーク

**CleanUpModulesOnExit** **firewalld** グローバル設定オプションが使用可能になりました。

以前は、**firewalld** を再起動またはシャットダウンすると、**firewalld** はカーネルモジュールを再帰的にアンロードしていました。その結果、他のパッケージがこれらのモジュールまたは依存モジュールを使用すると失敗しました。このアップグレードでは、ユーザーは **CleanUpModulesOnExit** オプションを **no** に設定して、**firewalld** によるこれらのカーネルモジュールのアンロードを停止できます。

(BZ#1980206)

大きな **nftables** セットの復元に必要なメモリーがすくなくなりました

この機能拡張により、**nftables** フレームワークが大きなセットを復元するために必要なメモリーが大幅に減少しました。**netlink** メッセージを準備するアルゴリズムが改善され、その結果、セットを復元する際に使用するメモリーが最大 40% 減少しました。

(BZ#2047821)

**nmstate** API が **OVS-DPDK** に対応するようになりました。

この改善により、Open vSwitch (OVS) Data Plane Development Kit (DPDK) のスキーマが **nmstate** API に追加されました。その結果、**nmstate** を使用して、DPDK ポートで OVS デバイスを設定できます。

(BZ#2003976)

**nmstate** API が、**SR-IOV** 仮想機能の **VLAN** および **QoS ID** に対応するようになりました。

今回の更新で、SR-IOV (Single root I/O virtualization) 仮想機能で、ローカルエリアネットワーク (VLAN) および QoS (Quality of Service) に対応するようになり、**nmstate** API が強化されました。その結果、**nmstate** を使用して、このような機能を設定できます。

(BZ#2004006)

**NetworkManager** がバージョン 1.36.0 にリベースされました。

**NetworkManager** パッケージがアップストリームバージョン 1.36.0 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。



- レイヤ 3 設定の処理は、安定性、パフォーマンス、およびメモリ使用量を改善するために作り直されました。
- NetworkManager は、IBM Z プラットフォームで **rd.znet\_ifnames** カーネルコマンドラインオプションをサポートするようになりました。
- **blackhole**、**unreachable**、および **prohibit** ルートタイプが追加されました。
- NetworkManager は、ルーティングサービスによって管理されるルートが無視するようになりました。
- Wi-Fi Protected Access バージョン 3 (WPA3) ネットワークセキュリティは、SAE (simultaneous authentication of equals) パスワード要素の同時認証を生成するときに H2E (hash-to-element) 方式を有効にすることで改善されました。
- このサービスは、重複するアドレスまたはマスクオプションを送信する DHCP サーバーからの応答を正しく処理するようになりました。
- ブリッジの MAC エージングをオフにできるようになりました。
- NetworkManager は、**qdiscs** や **filters** などのトラフィック制御オブジェクトの **netlink** イベントをリッスンしなくなりました。
- ネットワークボンドは、ボンドポートのキュー ID の設定をサポート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。

- [NetworkManager 1.36.0](#)
- [NetworkManager 1.34.0](#)

(BZ#1996617)

### hostapd パッケージが RHEL 8.6 に追加されました。

今回のリリースで、RHEL は **hostapd** パッケージを提供します。ただし、Red Hat が **hostapd** に対応するのは、イーサネットネットワークで RHEL ホストを 802.1X 認証子として設定することのみです。Wi-Fi アクセスポイントや Wi-Fi ネットワークのオーセンティケーターなど、その他のシナリオには対応していません。

FreeRADIUS バックエンドを備えた 802.1X オーセンティケーターとして RHEL を設定する方法の詳細については、[Setting up an 802.1x network authentication service for LAN clients using hostapd with FreeRADIUS backend](#) を参照してください。

(BZ#2016946)

### NetworkManager は、OVS-DPDK インターフェイスでの受信キュー数 (rx\_queue) の設定をサポートするように

この機能拡張により、NetworkManager を使用して、Open vSwitch (OVS) データプレーン開発キット (DPDK) インターフェイスの **n\_rxq** 設定を設定できます。NetworkManager の **ovs-dpdk.n-rxq** 属性を使用して、OVS-DPDK インターフェイスの受信キューの数を設定します。

たとえば、**ovs-iface0** という名前の OVS インターフェイスで 2 つの受信キューを設定するには、次のように入力します。

```
# nmcli connection modify ovs-iface0 ovs-dpdk.nrxq 2
```

(BZ#2001563)

**nftables** フレームワークは、カウンターが接続された **nft** セット要素をサポートするようになる

以前は、**netfilter** フレームワークでは、**nftables** セットカウンターはサポートされていませんでした。**nftables** フレームワークは、**nft** ツールで設定できます。カーネルを使用すると、このツールは、ステートメント **add @myset {ip saddr counter}** を使用して、特定の送信元アドレスからのネットワークパケットをカウントできます。この更新では、動的セットを使用して特定の基準に一致するパケットと、カウンターが接続されている要素をカウントできます。

(BZ#1983635)

**nispor** パッケージが完全にサポートされるようになる

以前はテクノロジープレビューとして利用可能だった **nispor** パッケージが、完全にサポートされるようになりました。この機能強化により、ネットワークルートおよびインターフェイスでカーネルフィルターを使用するための **NetStateFilter** のサポートが追加されます。

このリリースでは、**nispor** パッケージの単一のルート入力および出力仮想化 (SR-IOV) インターフェイスは、(VF) ごとに SR-IOV Virtual Function (SR-IOV VF) 情報を照会できるようになり、新しいボンディングオプション (**lACP\_active**、**arp\_missed\_max**、および **ns\_ip6\_target**) をサポートします。

(BZ#1848817)

## 4.9. カーネル

### RHEL 8.6 のカーネルバージョン

Red Hat Enterprise Linux 8.6 には、カーネルバージョン 4.18.0-372 が同梱されています。

[外部カーネルパラメーターの重要な変更](#) および [デバイスドライバー](#) も参照してください。

(BZ#1839151)

### Extended Berkeley Packet Filter for RHEL 8.6

**extended Berkeley Packet Filter (eBPF)** 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。この仮想マシンは、特別なアセンブリのようなコードを実行します。

**eBPF** バイトコードが最初にカーネルにロードされ、その後に検証が行われます。次に実行時のコンパイルでコードがネイティブマシンコードに変換され、その後、仮想マシンがコードを実行します。

Red Hat は、**eBPF** 仮想マシンを使用するコンポーネントを数多く提供しています。各コンポーネントの開発フェーズはさまざまです。そのため、現在すべてのコンポーネントが完全にサポートされている訳ではありません。RHEL 8.6 では、以下の **eBPF** コンポーネントがサポートされています。

- **eBPF** を使用して Linux オペレーティングシステムの I/O 分析、ネットワーク、およびモニタリングを行う **BPF コンパイラーコレクション (BCC)** ツールパッケージ。
- **BCC** ライブラリー。これを使用すると、**BCC** ツールパッケージで提供されるツールと同様のツールを開発できます。
- **eBPF for Traffic Control (tc)** 機能。これにより、カーネルネットワークデータパスでのプログラミング可能なパケット処理が可能になります。
- **bpftrace** トレース言語

- カーネルネットワーキングスタックを処理する前に受信パケットへのアクセスを提供する **eXpress Data Path (XDP)** 機能は、特定の条件でサポートされます。詳細は、[XDP に条件付きサポートを追加](#) および [RHEL におけるネットワーク eBPF 機能の概要](#) を参照してください。
- **libbpf** パッケージ。 **bpftrace** および **bpf/xdp** の開発のようなアプリケーションに関連する bpf に極めて重要です。
- **xdp-tools** パッケージが、AMD および Intel 64 ビットアーキテクチャーでサポートされるようになりました。このパッケージには、XDP 機能用のユーザー空間サポートユーティリティーが含まれます。これには、 **libxdp** ライブラリー、XDP プログラムを読み込む **xdp-loader** ユーティリティー、パケットフィルタリングの **xdp-filter** サンプルプログラム、XDP が有効になっているネットワークインターフェイスからパケットを取得する **xdpdump** ユーティリティーなどが含まれます。

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、その他のすべての eBPF 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 コンポーネントは、テクノロジープレビューとして利用できます。

- eXpress Data Path (XDP)パスをユーザー空間に接続する **AF\_XDP** ソケット

テクノロジープレビューのコンポーネントに関する詳細情報は、[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](#)。を参照してください。

(BZ#1780124)

**Red Hat は、デフォルトで、特権ユーザーに対してのみ、すべての RHEL バージョンで eBPF を有効にします。**

Extended Berkeley Packet Filter (**eBPF**) は、ユーザーが Linux カーネル内でカスタムコードを実行できるようにする複雑なテクノロジーです。その性質上、**eBPF** コードはベリファイアやその他のセキュリティメカニズムを通過する必要があります。Common Vulnerabilities and Exposures (CVE) インスタンスがあり、このコードのバグが不正な操作に悪用される可能性があります。このリスクを軽減するため、Red Hat は、特権ユーザーの場合にのみ、すべての RHEL バージョンで **eBPF** を有効にしています。kernel.command-line パラメーター **unprivileged\_bpf\_disabled=0** を使用して、非特権ユーザーに対して **eBPF** を有効にすることができます。

ただし、次の点に注意してください。

- **unprivileged\_bpf\_disabled=0** を適用すると、カーネルが Red Hat サポートの資格を失い、システムがセキュリティリスクにさらされます。
- Red Hat は、**CAP\_BPF** 機能を持つプロセスを、その機能が **CAP\_SYS\_ADMIN** と同等であるかのように扱うことを推奨します。
- **unprivileged\_bpf\_disabled=0** を設定しても、非特権ユーザーが多くの BPF プログラムを実行するには不十分です。これは、ほとんどの BPF プログラムタイプのロードには追加の機能 (通常は **CAP\_SYS\_ADMIN** または **CAP\_PERFMON**) が必要になるためです。

カーネルコマンドラインパラメーターの設定方法は、[カーネルコマンドラインパラメーターの設定](#) を参照してください。

(BZ#2089409)

**osnoise および timerlat** トレーサーが RHEL8 に追加される

**osnoise** トレーサーは、オペレーティングシステムのノイズを測定します。つまり、OS によるアプリケーションの中断とハードウェアの中断です。また、OS ノイズの原因を見つけるのに役立つ一連のト



レースポイントも提供します。**timerlat** トレーサーは、ウェイクアップレイテンシーを測定し、リアルタイム (RT) スレッドのそのようなレイテンシーの原因を特定するのに役立ちます。RT コンピューティングでは、レイテンシーは絶対に重要であり、最小限の遅延でさえ有害になる可能性があります。**osnoise** および **timerlat** トレーサーを使用すると、OS がアプリケーションに干渉したり、RT スレッドのウェイクアップ遅延が発生したりする原因を調査して見つけることができます。

(BZ#1979382)

**strace** ユーティリティーは、実際の SELinux コンテキストと SELinux コンテキストデータベースから抽出された定義との間の不一致を表示できるようになりました

**strace** の既存の **--secontext** オプションは、**mismatch** パラメーターで拡張されました。このパラメーターを使用すると、不一致の場合にのみ、実際のコンテキストとともに期待されるコンテキストを出力できます。出力は、2つの感嘆符 (!!) で区切られます。最初は実際のコンテキスト、次に期待されるコンテキストです。以下の例では、コンテキストのユーザー部分が不一致であるため、**full,mismatch** パラメーターは、実際のコンテキストとともに期待される完全なコンテキストを出力します。ただし、単独の **mismatch** を使用する場合は、コンテキストのタイプ部分のみをチェックします。コンテキストのタイプ部分が一致するため、予期されるコンテキストは出力されません。

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

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

SELinux コンテキストの不一致は、SELinux に関連するアクセス制御の問題を引き起こすことがよくあります。システムコールトレースに出力された不一致により、SELinux コンテキストの正確性のチェックが大幅に迅速化されます。システムコールトレースは、アクセス制御チェックに関する特定のカーネルの動作を説明することもできます。

(BZ#2038992、BZ#2038810)

**--cyclictest-threshold** オプションが **rteval** ユーティリティーに追加されました

この機能拡張により、**--cyclictest-threshold=USEC** オプションが **rteval** テストスイートに追加されました。このオプションを使用すると、しきい値を指定できます。待ち時間の測定値がこのしきい値を超えると、**rteval** テストの実行はすぐに終了します。待ち時間の期待が満たされない場合、実行は失敗ステータスで中止されます。

(BZ#2012285)

## 4.10. ファイルシステムおよびストレージ

**RHEL 8.6 は RHEL 9 XFS イメージと互換性があります**

この更新により、RHEL 8.6 は RHEL 9 イメージを使用できるようになりました。RHEL 9 XFS ゲストイメージは、RHEL 8.6 でゲストイメージをマウントするために、**bigtime** および inode btree カウンター (**inobtcount**) のオンディスク機能が許可されている必要があります。**bigtime** および **inobtcount** 機能を使用して作成されたファイルシステムは、RHEL 8.6 より前のバージョンと互換性がないことに注意してください。

(BZ#2022903, BZ#2024201)

一貫したユーザーエクスペリエンスのために、Samba ユーティリティのオプションの名前が変更され、削除されました。

Samba ユーティリティが改善され、一貫したコマンドラインインターフェイスが提供されるようになりました。この改善には、オプションの名前変更や削除が含まれます。そのため、更新後の問題を回避するには、Samba ユーティリティを使用するスクリプトを確認し、必要に応じてスクリプトを更新します。

Samba 4.15 では、Samba ユーティリティに以下の変更が加えられました。

- Samba コマンドラインユーティリティは、不明なオプションを暗黙的に無視していました。予期しない動作を防ぐために、ユーティリティが、不明なオプションを常に拒否するようになりました。
- いくつかのコマンドラインオプションには、デフォルト値を制御するのに対応する **smb.conf** が追加されました。コマンドラインオプションに **smb.conf** 変数名があるかどうかを確認するには、ユーティリティの man ページを参照してください。
- デフォルトで、Samba ユーティリティが標準エラー (**stderr**) にログを記録するようになりました。この挙動を変更するには、**--debug-stdout** を使用します。
- 一般的なパーサーに **--client-protection=off|sign|encrypt** が追加されました。
- 以下のオプションは、すべてのユーティリティで名前が変更されています。
  - **--kerberos** から **--use-kerberos=required|desired|off** へ
  - **--krb5-ccache** から **--use-krb5-ccache=CCACHE** へ
  - **--scope** から **--netbios-scope=SCOPE** へ
  - **--use-ccache** から **--use-winbind-ccache** へ
- 以下のオプションがすべてのユーティリティから削除されました。
  - **-e** および **--encrypt**
  - **--use-winbind-ccache** から削除された **-C**
  - **--netbios-scope** から削除された **-i**
  - **-S** および **--signing**
- オプションの重複を防ぐため、次のユーティリティから特定のオプションが削除されたり、名前が変更されたりしています。
  - **ndrdump: -l** は、**--load-dso** では使用できなくなりました。
  - **net: -l** は、**--long** では使用できなくなりました。
  - **sharesec: -V** は、**--viewsddl** では使用できなくなりました。
  - **smbcquotas: --user** の名前が **--quota-user** に変更になりました。
  - **nmbd: --log-stdout** の名前が **--debug-stdout** に変更になりました。
  - **smbd: --log-stdout** の名前が **--debug-stdout** に変更になりました。
  - **winbindd: --log-stdout** の名前が **--debug-stdout** に変更になりました。

(BZ#2062117)

## 機能ポインターとの名前の競合を回避するために、コンパイラバリアが静的インライン機能 `compiler_barrier` に変更されました

この機能拡張は、潜在的なデータ破損バグに対する追加機能とパッチを提供します。これで、コンパイラバリアが静的インライン機能 `compiler_barrier` に設定されます。機能ポインターを使用しているときに、非一時的な `memcpy` バリエーションにハードウェアフェンシングを実装する場合、ハードウェアストアバリアとの名前の競合は発生しません。その結果、RHEL 8.6 には `pmdk` バージョン 1.11.1 が含まれるようになりました。

(BZ#2009889)

## 4.11. 高可用性およびクラスター

### `pcmk_delay_base` パラメーターは、ノードごとに異なる値を取る可能性があります

フェンスデバイスを設定するときに、`pcmk_delay_base parameter` を使用してノードごとに異なる値を指定できるようになりました。これにより、ノードごとに異なる遅延を使用して、単一のフェンスデバイスを2ノードクラスターで使用できます。これは、各ノードが同時に他のノードをフェンスしようとする状況を防ぐのに役立ちます。ノードごとに異なる値を指定するには、`pcmk_host_map` と同様の構文を使用して、ホスト名をそのノードの遅延値にマップします。たとえば、`node1:0;node2:10s` は、`node1` をフェンシングするときに遅延を使用せず、`node2` をフェンシングするときに 10 秒の遅延を使用します。

(BZ#1082146)

### リソースの移動に伴う場所の制約の自動削除の指定

`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。以前はテクノロジープレビューとして使用可能だった `pcs resource move` コマンドの新しい `--autodelete` オプションが完全にサポートされるようになりました。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#1990784)

### 内部エラーの詳細なペースメーカーステータス表示

エージェントがインストールされていない、内部タイムアウトが発生したなど、何らかの理由で Pacemaker がリソースまたはフェンスエージェントを実行できない場合は、Pacemaker ステータス表示に内部エラーの詳細な終了理由が表示されるようになりました。

(BZ#1470834)

### `pcmk_host_map` 値内の特殊文字のサポート

`pcmk_host_map` プロパティは、値の前に円記号 (\) を使用して、`pcmk_host_map` 値内の特殊文字をサポートできるようになりました。たとえば、`pcmk_host_map="node3:plug\ 1"` を指定して、ホストエイリアスにスペースを含めることができます。

(BZ#1376538)

### OCF Resource Agent API 1.1 標準の `pcs` サポート

`pcs` コマンドラインインターフェイスは、OCF 1.1 リソースと STONITH エージェントをサポートするようになりました。OCF 1.1 エージェントのメタデータは、OCF 1.1 スキーマに準拠している必要があります。OCF 1.1 エージェントのメタデータが OCF 1.1 スキーマに準拠していない場合、`pcs` はエージェン

トを無効と見なし、**--force** オプションが指定されていない限り、エージェントのリソースを作成または更新しません。エージェントをリスト表示するための **pcsd** Web UI および **pcs** コマンドは、無効なメタデータを持つ OCF1.1 エージェントをリスト表示から除外します。

1.1 以外の OCF バージョンを実装することを宣言する、またはバージョンをまったく宣言しない OCF エージェントは、OCF1.0 スキーマに対して検証されます。検証の問題は警告として報告されますが、これらのエージェントの場合、エージェントのリソースを作成または更新するときに **--force** オプションを指定する必要はありません。

(BZ#1936833)

## OpenShift 用の新しいフェンシングエージェント

現在、**fence\_kubevirt** フェンシングエージェントは、Red Hat OpenShift Virtualization の RHEL High Availability で使用できます。**fence\_kubevirt** エージェントの詳細については、**fence\_kubevirt(8) man** ページを参照してください。

(BZ#1977588)

## 4.12. 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

### 新しいモジュールストリーム: php:8.0

RHEL 8.6 は **PHP 8.0** を追加します。これにより、バージョン 7.4 に比べて多くのバグ修正と機能拡張が提供されます。

主な機能拡張は、次のとおりです。

- 新しい名前付き引数は順序に依存せず、自己ドキュメント化されており、必要なパラメーターのみを指定できるようになりました。
- 新しい属性により、PHP のネイティブ構文で構造化メタデータを使用できます。
- 新しいユニオンタイプにより、複数のタイプの組み合わせに対して、PHPDoc アノテーションの代わりに実行時に検証されるネイティブのユニオンタイプ宣言を使用できます。
- 内部関数は、パラメーターの検証に失敗した場合に警告ではなく、Error 例外を常に発生させるようになりました。
- Just-In-Time コンパイルにより、パフォーマンスが向上しました。
- PHP の **Xdebug** デバッグおよび生産性拡張機能がバージョン 3 に更新されました。このバージョンでは、**Xdebug 2** と比較した機能および設定に大きな変更が加えられました。

**php:8.0** モジュールストリームをインストールするには、以下を実行します。

```
# yum module install php:8.0
```

**php:7.4** 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。

RHEL 8 での PHP の使用方法の詳細は、[PHP スクリプト言語の使用](#) を参照してください。

(BZ#1978356、BZ#2027285)

## 新しいモジュールストリーム: perl:5.32

RHEL 8.6 では **Perl 5.32** が導入されており、RHEL 8.3 で配布されている **Perl 5.30** に比べて多くのバグ修正と機能拡張が提供されています。

主な機能強化は、次のとおりです。

- **Perl** が unicode バージョン 13.0 に対応しました。
- **qr** quote-like 演算子が強化されました。
- **POSIX::mblen()**、**mbtowc**、**wctomb** 関数がシフト状態のロケールで動作するようになり、ロケールのスレッドセーフ機能を持つプラットフォームで実行した場合に、C99 以上のコンパイラではスレッドセーフになりました。長さのパラメーターはオプションになりました。
- 新しい実験的な **isainfix** 演算子は、与えられたオブジェクトが、与えられたクラスのインスタンスであるか、そこから派生したクラスであるかをテストします。
- アルファアサーションはもはや実験的なものではありません。
- スクリプトの実行は実験的なものではなくなりました。
- 機能チェックが速くなりました。
- **Perl** は最適化の前にコンパイルされたパターンをダンプできるようになりました。

以前の **perl** モジュールストリームからアップグレードするには、[後続のストリームへの切り替え](#) を参照してください。

([BZ#2021471](#))

## 新しいパッケージ: nginx-mod-devel

新しい **nginx-mod-devel** パッケージが **nginx:1.20** モジュールストリームに追加されました。このパッケージは、**nginx** の外部動的モジュールを構築するために、RPM マクロや **nginx** ソースコードを含むすべての必要なファイルを提供します。

([BZ#1991787](#))

## MariaDB Galera には、garbd systemd サービスのアップストリームバージョンとラッパースクリプトが含まれるようになりました

RHEL 8 の MariaDB 10.3 および MariaDB 10.5 には、Red Hat バージョンの **garbd** systemd サービスと、**/usr/lib/systemd/system/garbd.service** ファイルおよび **/usr/sbin/garbd-wrapper** ファイルにそれぞれある **galera** パッケージのラッパースクリプトが含まれています。

これらのファイルの Red Hat バージョンに加えて、RHEL 8 はアップストリームバージョンも提供するようになりました。アップストリームファイルは、**/usr/share/doc/galera/garbd-systemd** および **/usr/share/doc/galera/garbd.service** にあります。

RHEL 9 は、**/usr/lib/systemd/system/garbd.service** および **/usr/sbin/garbd-systemd** にあるこれらのファイルのアップストリームバージョンのみを提供します。

([BZ#2042306](#)、[BZ#2042298](#)、[BZ#2050543](#)、[BZ#2050546](#))

## 4.13. コンパイラーおよび開発ツール

## glibc 最適化データをキャプチャーするための新しいコマンド

新しい `ld.so --list-diagnostics` コマンドは、IFUNC の選択や `glibc-hwcaps` の設定など、`glibc` の最適化の決定に影響を与えるデータを単一の機械可読ファイルにキャプチャーします。

([BZ#2023420](#))

## glibc 文字列関数が Fujitsu A64FX 用に最適化される

今回の更新により、`glibc` 文字列関数は A64FX CPU でスループットの向上とレイテンシーの削減を実現します。

([BZ#1929928](#))

## 12 時間制の新しい UTF-8 ロケール en\_US@ampm

この更新により、12 時間制の新しい UTF-8 ロケール `en_US@ampm` を使用できるようになりました。この新しいロケールは、`LC_TIME` 環境変数を使用して他のロケールと組み合わせることができます。

([BZ#2000374](#))

## libffi の自己変更コードの新しい場所

この更新では、`libffi` の自己変更コードは、RHEL 8 カーネルの機能を利用して、ファイルシステムに依存しない適切なファイルを作成します。その結果、`libffi` の自己修正コードは、ファイルシステムの一部を不安定にすることに依存しなくなりました。

([BZ#1875340](#))

## GCC Toolset 11 の更新

GCC Toolset 11 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。

RHEL 8.6 で導入された主な変更点は、以下のとおりです。

- GCC コンパイラーがバージョン 11.2.1 に更新されました。
- `annobin` がバージョン 10.23 に更新されました。

以下のツールおよびバージョンは、GCC Toolset 10 で利用できます。

ツール	バージョン
GCC	11.2.1
GDB	10.2
Valgrind	3.17.0
SystemTap	4.5
Dyninst	11.0.0

ツール	バージョン
binutils	2.36.1
elfutils	0.185
dwz	0.14
make	4.3
strace	5.13
ltrace	0.7.91
annobin	10.23

GCC Toolset 11 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-11
```

GCC Toolset 11 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-11 tool
```

GCC Toolset バージョン 11 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-11 bash
```

使用についての詳細は、[GCC Toolset の使用](#) を参照してください。

GCC Toolset 11 コンポーネントが、以下のコンテナイメージ 2 つで利用可能になりました。

- GCC コンパイラー、GDB デバッガー、**make** 自動化ツールを含む **rhel8/gcc-toolset-11-toolchain**
- SystemTap や Valgrind などのパフォーマンス監視ツールを含む **rhel8/gcc-toolset-11-perftools**

コンテナイメージをプルするには、root で以下のコマンドを実行します。

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

GCC Toolset 11 コンテナイメージのみがサポートされるようになりました。以前のバージョンの GCC Toolset コンテナイメージが非推奨になりました。

コンテナイメージの詳細は、[GCC Toolset コンテナイメージの使用](#) を参照してください。

([BZ#1996862](#))



## GDB 逆アセンブラーが新しい arch14 命令をサポートするようになりました

この更新により、GDB は新しい arch14 命令を逆アセンブルできるようになりました。

(BZ#2012818)

## LLVM Toolset がバージョン 13.0.1 にリベース

LLVM Toolset がバージョン 13.0.1 にアップグレードされました。主な変更点は、以下のとおりです。

- Clang は、C++ のステートメント属性 `[[clang::musttail]]` および C の `__attribute__((musttail))` をサポートするようになりました。
- Clang は、`-Wreserved-identifier` 警告をサポートするようになりました。これは、コードで予約済み識別子を使用するときに開発者に警告します。
- Clang の `-Wshadow` フラグは、シャドウされた構造化バインディングもチェックするようになりました。
- Clang の `-Wextra` は、`Wnull-pointer-subtraction` も意味するようになりました。

(BZ#2001133)

## RustToolset が 1.58.1 にリベース

RustToolset はバージョン 1.58.1 にリベース以下は、主な変更点です。

- Rust コンパイラーは、クロージャーの disjoint キャプチャー、配列の `Intolterator`、新しい Cargo 機能リゾルバーなどを備えた 2021 年版の言語をサポートするようになりました。
- 新しいカスタムプロファイルの Cargo サポートが追加されました。
- Cargo はコンパイラーエラーを重複排除します。
- 新しいオープンレンジパターンが追加されました。
- フォーマット文字列にキャプチャーされた識別子を追加しました。

詳細は、以下を参照してください。

- [Rust 1.55](#)
- [Rust 1.56](#)
- [Rust 1.57](#)
- [Rust 1.58](#)

(BZ#2002883)

## Go Toolset がバージョン 1.17.7 にリベース

Go Toolset がバージョン 1.17.7 にアップグレードされました。主な変更点は、以下のとおりです。

- スライスを配列ポインターに変換するオプションを追加しました。
- `//go:build` 行のサポートを追加しました。
- amd64 での関数呼び出しのパフォーマンスを改善しました。



- 関数引数を、スタックトレースでより明確にフォーマットしました。
- クロージャーを含む関数をインライン化できるようにしました。
- x509 証明書の解析におけるリソース消費を削減しました。

(BZ#2014088)

### pcp が 5.3.5-8 にリベースされました。

**pcp** パッケージはバージョン 5.3.5 にリベースされました。主な変更点は、以下のとおりです。

- CPU とディスクの飽和に関する新しい **pmieconf(1)** ルールが追加されました。
- **pmproxy(1)** サービスの安定性とスケーラビリティが向上しました。
- **pmlogger(1)** サービスのサービスレイテンシーと堅牢性が向上しました。
- 電力に関連する新しいパフォーマンスメトリックを追加しました。
- **pcp-htop(1)** ユーティリティーに新機能を追加しました。
- **pcp-atop(1)** ユーティリティーに新機能を追加しました。
- Nvidia GPU メトリックを更新しました。
- 新しい Linux カーネル KVM とネットワークメトリックを追加しました。
- 新しい MongoDB メトリックエージェントが追加されました。
- 新しいソケットメトリックエージェントと **pcp-ss(1)** ユーティリティーが追加されました。
- **pmcd(1)** および **pmproxy(1)** Avahi サービスのアドバタイズをデフォルトで無効にしました。

(BZ#1991763)

### grafana パッケージがバージョン 7.5.11 にリベース

**grafana** パッケージがバージョン 7.5.11 にリベースされました。主な変更点は、以下のとおりです。

- 新しいデータフレーム形式をサポートしないパネルの下位互換性のために、新しい **prepare time series** 変換が追加されました。

(BZ#1993214)

### grafana-pcp が 3.2.0 にリベース

**grafana-pcp** パッケージがバージョン 3.2.0 にリベースされました。主な変更点は、以下のとおりです。

- PCP Redis 用の新しい MS SQL サーバーダッシュボードに追加
- PCP Vector eBPF/BCC 概要ダッシュボードに空のヒストグラムバケットの可視性を追加しました。
- PCP Redis の **metric()** 関数がすべてのメトリック名を返さないバグを修正しました。

(BZ#1993149)

## js-d3-flame-graph が 4.0.7 にリベース

**js-d3-flame-graph** パッケージはバージョン 4.0.7 にリベースされました。主な変更点は、以下のとおりです。

- 新しい青と緑の配色を追加しました。
- フレームグラフのコンテキストを表示する機能が追加されました。

(BZ#1993194)

## PCP で電力消費メトリックが利用可能に

新しい **pmda-denki** Performance Metrics Domain Agent (PMDA) は、消費電力に関連するメトリックを報告します。具体的には以下の項目が報告されます。

- 最近のインテル製 CPU で利用可能な RAPL(Running Average Power Limit) 測定値に基づく消費メトリック
- バッテリーを搭載したシステムで利用可能な、バッテリー放電に基づく消費メトリック

(BZ#1629455)

## 新しいモジュール: log4j:2

新しい **log4j:2** モジュールが AppStream リポジトリで利用できるようになりました。このモジュールには、Java ロギングユーティリティである **Apache Log4j 2** と、さまざまな出力ターゲットにログステートメントを出力できるライブラリーが含まれています。

**Log4j 2** は、**Log4j 1** よりも大幅に改善されています。特に、**Log4j 2** では、**Logback** フレームワークに拡張機能が導入され、**Logback** アーキテクチャーでいくつかの固有の問題を修正しています。

**log4j:2** モジュールストリームをインストールするには、以下を実行します。

```
# yum module install log4j:2
```

(BZ#1937468)

## 4.14. IDENTITY MANAGEMENT

すべての依存関係を持つ AppStream リポジトリで **ansible-freeipa** が利用できるようになりました。

以前の RHEL 8 では、**ansible-freeipa** パッケージをインストールする前に、まず Ansible リポジトリを有効にして **ansible** パッケージをインストールする必要がありました。RHEL 8.6 および RHEL 9 では、準備手順なしで **ansible-freeipa** をインストールできます。**ansible-freeipa** をインストールすると、依存関係として、**ansible** のより基本的なバージョンである **ansible-core** パッケージが自動的にインストールされます。**ansible-freeipa** と **ansible-core** の両方が、**rhel-9-for-x86\_64-appstream-rpms** リポジトリで利用できます。

RHEL 8.6 および RHEL 9 の **ansible-freeipa** には、RHEL 8 で含まれていたモジュールがすべて含まれています。

(JIRA:RHELPLAN-100359)

IdM は、**automountlocation**、**automountmap**、および **automountkey** Ansible モジュールをサポートするようになりました。

この更新では、**ansible-freeipa** パッケージに、**ipaautomountlocation**、**ipaautomountmap**、および **ipaautomountkey** モジュールが含まれています。これらのモジュールを使用して、IdM の場所にある IdM クライアントにログインした IdM ユーザーが自動的にマウントされるようにディレクトリーを設定できます。現在サポートされているのはダイレクトマップのみであることに注意してください。

(JIRA:RHELPLAN-79161)

### サブ ID 範囲の管理は、shadow-utils で可能です。

以前では、**shadow-utils** が設定したサブ ID は、**/etc/subuid** ファイルおよび **/etc/subgid** ファイルから自動的に範囲を設定していました。今回の更新で、**subid** フィールドに値を設定することで、**/etc/nsswitch.conf** ファイルで subID 範囲の設定を利用できるようになりました。詳細は **man subuid** および **man subgid** を参照してください。また、今回の更新で、IPA サーバーからのサブ ID 範囲を提供する、**shadow-utils** プラグインの SSSD 実装が利用可能になりました。この機能を使用するには、**subid: sss** を **/etc/nsswitch.conf** に追加します。このソリューションは、コンテナ化した環境でルートレスコンテナを容易にするために役立ちます。

**/etc/nsswitch.conf** ファイルが **authselect** ツールで設定されている場合は、**authselect** のドキュメントに記載されている手順に従う必要があります。そうでない場合は、**/etc/nsswitch.conf** を手動で修正できます。

(JIRA:RHELPLAN-103579)

### 従来の RHEL の ansible-freeipa リポジトリに代わる Ansible 自動化ハブ

この更新では、標準の RHEL リポジトリからダウンロードする代わりに、Ansible Automation Hub (AAH) から **ansible-freeipa** モジュールをダウンロードできます。AAH を使用することで、このリポジトリで利用可能な **ansible-freeipa** モジュールのより高速な更新の恩恵を受けることができます。

AAH では、**ansible-freeipa** のロールとモジュールがコレクション形式で配布されます。AAH ポータルのコンテンツにアクセスするには、Ansible Automation Platform (AAP) サブスクリプションが必要であることを注意してください。また、**ansible** バージョン 2.9 以降も必要です。

**redhat.rhel\_idm** コレクションには、従来の **ansible-freeipa** パッケージと同じコンテンツが含まれています。ただし、コレクション形式では、名前空間とコレクション名で設定される完全修飾コレクション名 (FQCN) が使用されます。たとえば、**redhat.rhel\_idm.ipadnsconfig** モジュールは、RHEL リポジトリによって提供される **ansible-freeipa** の **ipadnsconfig** モジュールに対応します。名前空間とコレクション名の組み合わせにより、オブジェクトが一意になり、競合することなく共有できるようになります。

(JIRA:RHELPLAN-103147)

### ansible-freeipa モジュールを IdM クライアントでリモートで実行できるようになりました

以前は、**ansible-freeipa** モジュールは IdM サーバーでのみ実行できました。これには、Ansible 管理者が IdM サーバーへの **SSH** アクセスを持っている必要があり、潜在的なセキュリティの脅威を引き起こしていました。この更新により、IdM クライアントであるシステム上で **ansible-freeipa** モジュールをリモートで実行できるようになります。その結果、IdM の設定とエンティティーをより安全な方法で管理できます。

IdM クライアントで **ansible-freeipa** モジュールを実行するには、次のいずれかのオプションを選択します。

- Playbook の **hosts** 変数を IdM クライアントホストに設定します。
- **ansible-freeipa** モジュールを使用する Playbook タスクに **ipa\_context: client** 行を追加します。

`ipa_context` 変数を IdM サーバー上の `client` に設定することもできます。ただし、通常、サーバーコンテキストの方がパフォーマンスが向上します。`ipa_context` が設定されていない場合、`ansible-freeipa` はサーバーまたはクライアントで実行されているかどうかを確認し、それに応じてコンテキストを設定します。IdM クライアントホスト上の `server` に `context` が設定された `ansible-freeipa` モジュールを実行すると、`missing libraries` エラーが発生することに注意してください。

(JIRA:RHELPLAN-103146)

`ipadnsconfig` モジュールには、グローバルフォワーダーを除外するための `action: member` が必要になりました。

今回の更新で、`ansible-freeipa ipadnsconfig` モジュールを使用して Identity Management (IdM) のグローバルフォワーダーを除外するには、`state: absent` オプションの他に `action: member` オプションを使用する必要があります。Playbook で `action: member` を使用せずに `state: absent` だけを使用すると、その Playbook は失敗します。そのため、すべてのグローバルフォワーダーを削除するには、Playbook でこれらをすべて個別に指定する必要があります。一方、`state: present` オプションに `action: member` は必要ありません。

(BZ#2046325)

Identity Management はデフォルトで SHA384withRSA 署名をサポートするようになりました

この更新により、IdM の認証局 (CA) は、SHA-384 With RSA 暗号化署名アルゴリズムをサポートします。SHA384withRSA は、連邦情報処理標準 (FIPS) に準拠しています。

(BZ#1731484)

SSSD のデフォルトの SSH ハッシュ値が OpenSSH 設定と一致するようになりました

`ssh_hash_known_hosts` のデフォルト値が `false` に変更になりました。これは、デフォルトでホスト名をハッシュしない OpenSSH 設定と一致するようになりました。

ただし、引き続きホスト名をハッシュする必要がある場合は、`/etc/sss/sss.conf` 設定ファイルの `[ssh]` セクションに `ssh_hash_known_hosts = True` を追加します。

(BZ#2015070)

samba がバージョン 4.15.5 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 4.15.5 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。

- 一貫したユーザーエクスペリエンスのために、Samba ユーティリティーのオプションの名前が変更され、削除されました。
- サーバーのマルチチャンネルサポートがデフォルトで有効になりました。
- `SMB2_22`、`SMB2_24`、および `SMB3_10` のダイアレクトは、Windows のテクニカルレビューでのみ使用されていましたが、削除されました。

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。

重要な変更の詳細は、更新の前に [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#2013596](#))

## Directory Server がバージョン 1.4.3.28 にリベース

**389-ds-base** パッケージがアップストリームバージョン 1.4.3 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。

- レプリカで潜在的なデッドロックが修正されました。
- **dnalinterval** が **0** に設定されていると、サーバーが予想外に終了しなくなりました。
- 接続処理のパフォーマンスが改善されました。
- アクセス制御命令 (ACI) での **targetfilter** のパフォーマンスが改善されました。

([BZ#2016014](#))

## Directory Server が、tmpfs ファイルシステムのデータベースのメモリーマッピングされたファイルを保存するようになりました。

Directory Server の **nsslapd-db-home-directory** パラメーターは、データベースのメモリーマッピングファイルの場所を定義します。この改善により、パラメーターのデフォルト値が `/var/lib/dirsrv/slapd-instance_name/db/` から `/dev/shm/` に変更になりました。その結果、**tmpfs** ファイルシステムに保存されている内部データベースがあると、Directory Server のパフォーマンスが向上します。

([BZ#1780842](#))

## 4.15. デスクトップ

### ログイン時およびデスクトップセッション時のセキュリティー分類バナー

分類バナーを設定して、システムの全体的なセキュリティー分類レベルを示すことができるようになりました。これは、ログインしているシステムのセキュリティー分類レベルをユーザーが認識している必要があるデプロイメントに役立ちます。

分類バナーは、設定に応じて、次のコンテキストで表示されます。

- 実行中のセッション内
- ロック画面
- ログイン画面

分類バナーは、無視できる通知または永続的なバナーのいずれかの形式をとることができます。

詳細については、[Displaying the system security classification](#) を参照してください。

([BZ#1751336](#))

## 4.16. グラフィックインフラストラクチャー

### Intel Alder Lake-P GPU に対応しました

今回のリリースで、統合グラフィックスを備えた Intel Alder Lake-P CPU マイクロアーキテクチャーのサポートが追加されました。これには、以下の CPU モデルを含む Intel UHD グラフィックと Intel Xe 統合 GPU が含まれます。

- Intel Core i7-1280P
- Intel Core i7-1270P
- Intel Core i7-1260P
- Intel Core i5-1250P
- Intel Core i5-1240P
- Intel Core i3-1220P

Alder Lake-P グラフィックのサポートはデフォルトで無効になっています。これを有効にするには、カーネルコマンドラインに次のオプションを追加します。

```
i915.force_probe=PCI_ID
```

`PCI_ID` を、Intel GPU の PCI デバイス ID に置き換えるか、\* 文字に置き換えて、**i915** ドライバーを使用するすべてのアルファベット品質のハードウェアに対応できるようにします。

(BZ#1964761)

## 4.17. WEB コンソール

### sudo および Web コンソールからの SSH 用のスマートカード認証

これまでは、スマートカード認証を使用して sudo 権限を取得したり、Web コンソールで SSH を使用したりすることはできませんでした。この更新により、Identity Management のユーザーがスマートカードを使用して sudo 権限を取得したり、SSH で別のホストに接続したりできるようになります。



#### 注記

1 枚のスマートカードを使用して、認証を行い sudo 権限を得ることしかできません。sudo に別のスマートカードを使用することはサポートされていません。

(JIRA:RHELPLAN-95126)

### RHEL Web コンソールがデフォルトで Insights 登録を提供

この更新では、Red Hat Enterprise Linux Web コンソールを使用して RHEL システムを登録するときに、このシステムを Red Hat Insights に接続します。チェックボックスはデフォルトでチェックされています。Insights サービスに接続しない場合は、チェックボックスの選択を解除します。

(BZ#2049441)

### コックピットで、既存の TLS 証明書の使用をサポートするようになりました

この機能拡張により、証明書には厳密なファイルパーミッション要件 (`root:cockpit-ws 0640` など) がなくなり、他のサービスと共有できるようになりました。

(JIRA:RHELPLAN-103855)

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

### ファイアウォール RHEL システムのロールが RHEL 8 に追加されました

**rhel-system-roles.firewall** RHEL System Role が **rhel-system-roles** パッケージに追加されました。その結果、管理者はマネージドノードのファイアウォール設定を自動化できます。

(BZ#1854988)

### HA クラスタ RHEL システムのロールの完全サポート

以前はテクノロジープレビューとして利用可能だった高可用性クラスタ (HA クラスタ) のロールが完全にサポートされるようになりました。次の注目すべき設定が利用可能です。

- フェンスデバイス、リソース、リソースグループ、およびリソースクローン (メタ属性およびリソース操作を含む) の設定
- リソースの場所の制約、リソースのコロケーションの制約、リソースの順序の制約、およびリソースチケットの制約の設定
- クラスタプロパティの設定
- クラスタノード、カスタムクラスタ名およびノード名の設定
- マルチリンククラスタの設定
- システムの起動時にクラスタが自動的に起動するかどうかの設定

ロールを実行すると、ロールでサポートされていない設定、またはロールの実行時に指定されていない設定が削除されます。

現在、HA Cluster システムロールは SBD をサポートしていません。

(BZ#1893743)

### Networking システムロールが、OWE に対応するようになりました。

Opportunistic Wireless Encryption (OWE) は、Wi-Fi ネットワーク向けの日和見セキュリティのモードで、ワイヤレスメディアの暗号化を提供しますが、公共のホットスポットなどの認証は提供しません。OWE は、Wi-Fi クライアントとアクセスポイント間で暗号化を使用し、クライアントを盗聴攻撃から保護します。この機能拡張により、Networking RHEL システムロールは OWE に対応します。これにより、管理者は Networking システムロールを使用して、OWE を使用する Wi-Fi ネットワークへの接続を設定できるようになりました。

(BZ#1993379)

### Networking システムロールが、SAE に対応するようになりました。

Wi-Fi Protected Access バージョン 3 (WPA3) ネットワークでは、SAE (authentication of equals) 方法により、暗号鍵が送信されないようになります。この機能強化により、Networking RHEL システムロールは SAE に対応します。これにより、管理者は Networking RHEL System Role を使用して、WPA-SAE を使用する Wi-Fi ネットワークへの接続を設定できるようになりました。

(BZ#1993311)

### Cockpit RHEL System Role がサポートされるようになりました

この機能拡張により、システムに Web コンソールをインストールして設定できます。そのため、Web コンソールを自動化された方法で管理できます。



(BZ#2021661)

## LVM ボリュームの `raid_level` のサポートを追加しました

Storage RHEL System Role で、LVM ボリュームの `raid_level` パラメーターを指定できるようになりました。その結果、`lvraid` 機能を使用して LVM ボリュームを RAID にグループ化できます。

(BZ#2016514)

## NBDE クライアントシステムロールは、静的 IP アドレスを持つシステムをサポートします

以前は、静的 IP アドレスを使用してシステムを再起動し、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、静的 IP アドレスを持つシステムは NBDE クライアントシステムロールによってサポートされ、それらの IP アドレスは再起動後に変更しません。

(BZ#1985022)

## Storage システムロールでキャッシュされたボリュームのサポートを使用できます

Storage RHEL システムロールは、キャッシュされた LVM 論理ボリュームを作成および管理できるようになりました。LVM キャッシュを使用すると、LV のデータのサブセットを SSD などのより小さくて高速なデバイスに一時的に保存することで、より低速な論理ボリュームのパフォーマンスを向上させることができます。

(BZ#2016511)

## `rsyslog` からの認証用に `Elasticsearch` のユーザー名とパスワードを追加するためのサポート

この更新では、`Elasticsearch` ユーザー名とパスワードのパラメーターが `logging` System Role に追加され、`rsyslog` がユーザー名とパスワードを使用して `Elasticsearch` に対して認証できるようになります。

(BZ#2010327)

## RHEL System Roles の Ansible Core サポート

RHEL 8.6 GA リリースの時点で、サポートの範囲が限定された Ansible Core が提供され、RHEL でサポートされる自動化のユースケースが可能になります。Ansible Core は、以前は別のリポジトリで提供されていた Ansible Engine を置き換えます。Ansible Core は、RHEL の AppStream リポジトリで利用できます。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 移行の AppStream リポジトリに含まれている Ansible Core パッケージのサポート範囲](#) 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。

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#2012316)

## network RHEL System Role は、静的ルートで `named` ルーティングテーブルと `numeric` ルーティングテーブルの両方をサポートするようになりました。

この更新により、静的ルートの `named` ルーティングテーブルと `numeric` ルーティングテーブルの両方のサポートが追加されます。これは、ポリシールーティング (ソースルーティングなど) をサポートするための前提条件です。ユーザーは、後でポリシールーティングルールを定義して、正しいルートを決定的に使用するテーブルをシステムに指示できます。その結果、ユーザーが `route` で `table` 属性を指定した後、システムはルーティングテーブルにルートを追加できます。

(BZ#2031521)



Certificate ロールは、フックスクリプトで一貫して "Ansible\_managed" コメントを使用します

この機能拡張により、Certificate ロールは、プロバイダーをサポートするためのプレスクリプトとポストスクリプトを生成します。ロールはこれに、Ansible 標準の "ansible\_managed" 変数を使用して "Ansible managed" のコメントを挿入します。

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

コメントは、Certificate ロールがファイルを上書きする可能性があるため、スクリプト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。その結果、設定ファイルには、設定ファイルが Ansible によって管理されていることを示す宣言が含まれています。

(BZ#2054364)

Terminal Session Recoring システムロールは、マネージド設定ファイルで "Ansible managed" コメントを使用します

Terminal Session Recording ロールは、以下の 2 つの設定ファイルを生成します。

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

この更新により、Terminal Session Recoring ロールは、標準の Ansible 変数 **ansible\_managed** を使用して、**Ansible managed** コメントを設定ファイルに挿入します。コメントは、Terminal Session Recoring ロールがファイルを上書きする可能性があるため、設定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。その結果、設定ファイルには、設定ファイルが Ansible によって管理されていることを示す宣言が含まれています。

(BZ#2054363)

Microsoft SQL システムのロールは、切断されたサブスクリプションまたは Satellite サブスクリプション用にカスタマイズされたリポジトリをサポートするようになりました

以前は、カスタムサーバーからパッケージをプルする必要がある切断された環境のユーザー、または Satellite または Capsule を指す必要がある Satellite ユーザーは、Microsoft SQL Role のサポートを受けていませんでした。この更新により、ユーザーが **RPM** キー、**client**、および **server** の `mssql` リポジトリに使用するカスタマイズされた URL を提供できるようになり、問題が修正されます。URL が指定されていない場合、**mssql** ロールは公式の Microsoft サーバーを使用して RPM をダウンロードします。

(BZ#2038256)

Microsoft SQL ロールは、マネージド設定ファイルで一貫して "Ansible\_managed" コメントを使用します

`mssql` ロールは、次の設定ファイルを生成します。

- `/var/opt/mssql/mssql.conf`

この更新により、Microsoft SQL ロールは、Ansible 標準の **ansible\_managed** 変数を使用して、設定ファイルに "Ansible managed" コメントを挿入します。コメントは、**mssql** ロールがファイルを上書きする可能性があるため、設定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。その結果、設定ファイルには、設定ファイルが Ansible によって管理されていることを示す宣言が含まれています。

(BZ#2057651)

## Networking システムロールに追加されたすべてのボンディングオプションのサポート

この更新は、Networking RHEL システムロールへのすべてのボンディングオプションをサポートします。その結果、ボンディングされたインターフェイスを介したネットワーク伝送を柔軟に制御できます。その結果、そのインターフェイスにいくつかのオプションを指定することにより、ボンディングされたインターフェイスを介したネットワーク伝送を制御できます。

(BZ#2008931)

## NetworkManager は、PCI アドレスを使用したネットワークカードの指定をサポートしています

以前は、接続プロファイルの設定中に、NetworkManager は名前または MAC アドレスのいずれかを使用してネットワークカードを指定することしか許可されていませんでした。この場合、デバイス名は安定しておらず、MAC アドレスには、使用された MAC アドレスの記録を維持するためのインベントリが必要です。これで、接続プロファイルの PCI アドレスに基づいてネットワークカードを指定できます。

(BZ#1695634)

## 新しいオプション `auto_gateway` は、デフォルトルートの動作を制御します

以前は、`DEFROUTE` パラメーターは設定ファイルで設定できませんでしたが、すべてのルートに名前を付けることによって手動で設定することしかできませんでした。この更新により、接続の `ip` 設定セクションに新しい `auto_gateway` オプションが追加されます。これを使用して、デフォルトルートの動作を制御できます。`auto_gateway` は、次の方法で設定できます。

- `true` に設定すると、デフォルトゲートウェイ設定がデフォルトルートに適用されます。
- `false` に設定すると、デフォルトルートが削除されます。
- 指定しない場合、`network` ロールは選択した `network_provider` のデフォルトの動作を使用します。

(BZ#1897565)

## VPN ロールは、マネージド設定ファイルで一貫して `Ansible_managed` コメントを使用します

VPN ロールは、次の設定ファイルを生成します。

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

この更新により、VPN ロールは、Ansible 標準の `ansible_managed` 変数を使用して、設定ファイルに `Ansible managed` コメントを挿入します。コメントは、VPN ロールがファイルを上書きする可能性があるため、設定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。その結果、設定ファイルには、設定ファイルが Ansible によって管理されていることを示す宣言が含まれています。

(BZ#2054365)

## Firewall システムロールの新しい `source` パラメーター

これで、Firewall システムロールの **source** パラメーターを使用して、ファイアウォール設定のソースを追加または削除できるようになりました。

(BZ#1932678)

**Networking システムロールは、マネージド設定ファイルで '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#2057656)

**Firewall システムロールが、ファイアウォールデフォルトゾーンの設定に対応**

Firewall System ロールでデフォルトのファイアウォールゾーンを設定できるようになりました。ゾーンは、着信トラフィックをより透過的に管理する概念を表しています。ゾーンはネットワークインターフェイスに接続されているか、ソースアドレスの範囲に割り当てられます。各ゾーンのファイアウォールルールは個別に管理されるため、管理者は複雑なファイアウォール設定を定義してトラフィックに適用できます。この機能を使用すると、**firewall-cmd --set-default-zone zone-name** と同じように、インターフェイスを割り当てるデフォルトゾーンとして使用されるデフォルトゾーンを設定できます。

(BZ#2022458)

**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#2057645)

**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#2057661)

## 4.19. 仮想化

仲介されたデバイスが、IBM Z の仮想化 CLI に対応するようになりました。

**virt-install** または **virt-xml** を使用して、**vfio-ap**、**vfio-ccw** などの仲介デバイスを仮想マシンに接続できるようになりました。たとえば、これにより、IBM Z ホストで DASD ストレージデバイスおよび暗号化コプロセッサをより柔軟に管理できます。また、**virt-install** を使用して、既存の DASD 仲介デバ

イスをプライマリーディスクとして使用する仮想マシンを作成できます。手順は、RHEL 8 での仮想化の設定および管理ガイドを参照してください。

(BZ#1995125)

### Intel Atom P59 シリーズプロセッサの仮想化サポート

今回の更新で、RHEL 8 の仮想化で、以前の Snow Ridge として知られる Intel Atom P59 シリーズプロセッサのサポートが追加されました。その結果、RHEL 8 でホストされる仮想マシンは、**Snowridge** CPU モデルを使用し、プロセッサが提供する新機能を活用できるようになりました。

(BZ#1662007)

### 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)

### Windows 11 および Windows Server 2022 ゲストがサポートされるようになりました

RHEL 8 は、KVM 仮想マシンのゲストオペレーティングシステムとして Windows 11 および Windows Server 2022 の使用をサポートするようになりました。

(BZ#2036863、BZ#2004162)

## 4.20. クラウド環境の RHEL

### RHEL 8 仮想マシンが、Azure 上の特定の ARM64 ホストでサポートされるようになりました

ゲストオペレーティングシステムとして RHEL 8.6 を使用する仮想マシンは、Ampere Altra ARM ベースのプロセッサで実行されている Microsoft Azure ハイパーバイザーでサポートされるようになりました。

(BZ#1949614)

### 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)

### cloud-init は、Microsoft Azure 上のユーザーデータをサポートする

`--user-data` オプションが `cloud-init` ユーティリティーに導入されました。このオプションを使用すると、Azure で RHEL 8 仮想マシンをセットアップするときに、Azure Instance Metadata Service (IMDS) からスクリプトとメタデータを渡すことができます。

(BZ#2023940)

### cloud-init が VMware GuestInfo データソースに対応

今回の更新で、`cloud-init` ユーティリティーが VMware `guestinfo` データのデータソースを読み取ることができるようになりました。その結果、`cloud-init` を使用した VMware vSphere に RHEL 8 仮想マシンのセットアップが、より効率的で信頼性が高くなりました。

(BZ#2026587)

## 4.21. サポート性

### 新しいパッケージ: `rig`

RHEL 8 には、`rig` システムの監視およびイベント処理ユーティリティーを提供する `rig` パッケージが導入されています。

`rig` ユーティリティーは、システム管理者とサポートエンジニアが、発生がランダムに見える問題や、人間の介入が不適切なときに発生する問題の診断データ収集を支援するように設計されています。

(BZ#1888705)

### `sos report` は推定モードの実行を提供するようになりました

この `sos report` の更新により、RHEL サーバーから `sos` レポートを収集するために必要なディスク容量を概算できる `--estimate-only` オプションが追加されます。`sos report --estimate-only` コマンドの実行:

- `sos report` のドライランを実行します
- すべてのプラグインを連続して模倣し、それらのディスクサイズを推定します。

最終的なディスクスペースの見積もりは非常に概算であることに注意してください。したがって、推定値を 2 倍にすることを推奨します。

(BZ#1873185)

### Red Hat Support Tool が Hydra API を使用するようになる

**Red Hat Support Tool** は、非推奨の Strata API から新しい Hydra API に移行しました。これは機能に影響を与えません。ただし、Strata API `/rs/` パスのみを明示的に許可するようにファイアウォールを設定した場合は、ファイアウォールが正しく機能するように `/support/` に更新してください。

さらに、この変更により、**Red Hat Support Tool** を使用するとき 5GB を超えるファイルをダウンロードできるようになりました。

(BZ#2018194)

### Red Hat Support Tool が Red Hat Secure FTP をサポートするようになる

**Red Hat Support Tool** を使用する場合は、**Red Hat Secure FTP** によってファイルをケースにアップロードできるようになりました。**Red Hat Secure FTP** は、**Red Hat Support Tool** が以前のバージョンでサポートしていた非推奨の **Dropbox** ユーティリティをより安全に置き換えたものです。

(BZ#2018195)

### Red Hat Support Tool が S3 API をサポートする

**Red Hat Support Tool** は、S3 API を使用してファイルを Red Hat テクニカルサポートケースにアップロードできるようになりました。その結果、ユーザーは 1GB を超えるファイルをケースに直接アップロードできます。

(BZ#1767195)

## 4.22. コンテナ

### container-tools:4.0 の安定したストリームが利用可能に

Podman、Buildah、Skopeo、および runc ツールを含む **container-tools:4.0** モジュールストリームが利用できるようになりました。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が追加されました。

以前の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。

(JIRA:RHELPLAN-100175)

### NFS ストレージが利用可能になりました

ファイルシステムで xattr がサポートされている場合は、NFS ファイルシステムをコンテナとイメージのバックエンドストレージとして使用できるようになりました。

(JIRA:RHELPLAN-75169)

### container-tools:rhel8 モジュールが更新されました。

Podman、Buildah、Skopeo、crun、および runc ツールを含む **container-tools:rhel8** モジュールが利用できるようになりました。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。

- ネットワークスタックの変更により、Podman v3 以前で作成されたコンテナは v4.0 では使用できなくなります。
- ネイティブオーバーレイファイルシステムは、ルートレスユーザーとして使用できます。

- コンテナ内の NFS ストレージがサポートされます。
- すべてのコンテナの破棄を再作成を行わない限り、Podman の以前のバージョンへのダウングレードはサポートされません

Podman ツールはバージョン 4.0 にアップグレードされました。注目すべき変更の詳細については、[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。

(JIRA:RHELPLAN-100174)

### ユニバーサルベースイメージが Docker Hub で利用可能に

これまでユニバーサルベースイメージは、Red Hat コンテナカタログからしか入手できませんでした。この機能拡張により、ユニバーサルベースイメージも Docker Hub から [確認済みパブリッシュイメージ](#) として利用できます。

(JIRA:RHELPLAN-101137)

### podman コンテナイメージが利用可能に

これまでテクノロジープレビューとして提供されていた [registry.redhat.io/rhel8/podman](https://registry.redhat.io/rhel8/podman) コンテナイメージが、完全にサポートされるようになりました。[registry.redhat.io/rhel8/podman](https://registry.redhat.io/rhel8/podman) コンテナイメージは、**podman** パッケージをコンテナ化した実装です。**podman** ツールは、コンテナおよびイメージ、それらのコンテナにマウントされたボリューム、およびコンテナのグループから作成された Pod を管理します。

(JIRA:RHELPLAN-57941)

### Podman は、YAML ファイルを使用した Pod の自動ビルドと自動実行をサポートするようになりました

**podman play kube** コマンドは、YAML ファイルを使用して、Pod 内に複数のコンテナを持つ複数の Pod を自動的にビルドして実行します。

(JIRA:RHELPLAN-108830)

### Podman は、IdM から subUID および subGID の範囲を取得できるようになりました。

subUID と subGID の範囲を IdM で管理できるようになりました。同じ `/etc/subuid` ファイルおよび `/etc/subgid` ファイルをすべてのホストにデプロイする代わりに、単一の中央ストレージで範囲を定義できるようになりました。`/etc/nsswitch.conf` ファイルを変更し、**services: files sss** のようにサービスマップ行に **sss** を追加する必要があります。

詳細については、IdM ドキュメントの [subID 範囲を手動で管理](#) を参照してください。

(JIRA:RHELPLAN-101133)

### openssl コンテナイメージが利用可能になりました

**openssl** イメージは、OpenSSL 暗号化ライブラリーのさまざまな機能を使用するための **openssl** コマンドラインツールを提供します。OpenSSL ライブラリーを使用すると、秘密鍵の生成、証明書署名要求 (CSR) の作成、および証明書情報の表示を行うことができます。

**openssl** コンテナイメージは、次のリポジトリで利用できます。

- [registry.redhat.io/rhel8/openssl](https://registry.redhat.io/rhel8/openssl)
- [registry.access.redhat.com/ubi8/openssl](https://registry.access.redhat.com/ubi8/openssl)



(JIRA:RHELPLAN-101138)

## Netavark ネットワークスタックが利用可能になりました。

Podman 4.1.1-7 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 つのツールで構成されます。以前はテクノロジープレビューとして利用できた Netavark スタックは、[RHBA-2022:7127](#) アドバイザリーのリリースで完全にサポートされています。

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。

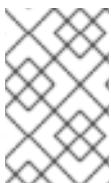
(JIRA:RHELPLAN-137623)

## 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

[RHBA-2022:7127](#) アドバイザリーのリリースに伴い、`podman run` コマンドおよび `podman create` コマンドは、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する `--health-on-failure` オプションをサポートす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#2130912)



## 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 8.6 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

#### **fw\_devlink.strict = [KNL]**

形式: <bool>

このパラメーターを使用すると、推測されるすべての依存関係を必須の依存関係として扱うことができます。この設定は、**fw\_devlink=on|rpm** の場合にのみ適用されます。

#### **no\_hash\_pointers**

このパラメーターを使用すると、コンソールまたはバッファーに出力されるポインターを強制的にハッシュ解除することができます。デフォルトでは、ポインターが **%p** フォーマット文字列を使用して出力される場合、そのポインターの値はハッシュによって隠されます。これは、権限のないユーザーから実際のカーネルアドレスを隠すセキュリティ機能です。ただし、同等でないポインターを比較できないため、カーネルのデバッグがより困難になります。このコマンドラインパラメーターを指定すると、すべての通常のポインターの真の値が出力されます。**%pK** フォーマット文字列を使用して出力されたポインターは、引き続きハッシュできます。**no\_hash\_pointers** は、カーネルをデバッグする場合にのみ指定し、本番環境では使用しないでください。

#### **no\_entry\_flush = [PPC]**

このパラメーターを使用すると、カーネルに入るときの L1-D キャッシュのフラッシュを回避できます。

#### **no\_uaccess\_flush = [PPC]**

このパラメーターを使用すると、ユーザーデータにアクセスした後に L1-D キャッシュがフラッシュされるのを回避できます。

#### **rcutorture.nocbs\_nthreads = [KNL]**

このパラメーターを使用すると、Read-copy-update (RCU) コールバックオフロードトグルの数を設定できます。

デフォルト値は 0 (ゼロ) で、トグルを無効にします。

#### **rcutorture.nocbs\_toggle = [KNL]**

このパラメーターを使用すると、連続するコールバックとオフロードの切り替え試行間の遅延をミリ秒単位で設定できます。

#### **refscale.verbose\_batched = [KNL]**

このパラメーターを使用すると、追加の **printk()** ステートメントをバッチ処理できます。

ゼロ (デフォルト) または負の値を指定すると、すべてを印刷できます。それ以外の場合は、N 番目の冗長ステートメントごとに出力します。ここで、N は指定された値です。

#### **strict\_sas\_size = [X86]**

形式: <bool>

このパラメーターを使用すると、サポートされている浮動小数点ユニット (FPU) 機能に応じて、必要なシグナルフレームサイズに対する厳密な **sigaltstack** サイズチェックを有効または無効にすることができます。このパラメーターを使用して、**AT\_MINSIGSTKSZ** 補助ベクトルをまだ認識していないバイナリーを除外できます。

#### **torture.verbose\_sleep\_frequency = [KNL]**

このパラメーターは、各スリープの間にいくつの詳細な **printk()** ステートメントを発行するかを指定します。

デフォルト値の 0 (ゼロ) は、**verbose-printk()** のスリープを無効にします。

#### **torture.verbose\_sleep\_duration = [KNL]**

このパラメーターは、各 **verbose-printk()** スリープの期間を **jiffy** 単位で指定します。

#### **tsc\_early\_khz = [X86]**

形式: <unsigned int>

このパラメーターにより、初期の Time Stamp Counter (TSC) キャリブレーションをスキップし、代わりに指定された値を使用できます。このパラメーターは、初期の TSC 周波数発見手順が信頼できない場合に役立ちます。CPUID.16h をサポートし、CPUID.15h を部分的にサポートするオーバークロックシステムなど。

### 更新されたカーネルパラメーター

#### **amd\_iommu = [HW,X86-64]**

システム内の AMD IOMMU ドライバーにパラメーターを渡すことができます。

可能な値は次のとおりです。

- **fullflush** - IO/TLB エントリーがマップされていないときのフラッシュを有効にします。それ以外の場合は、再利用される前にフラッシュされます。これにより、はるかに高速になります。
- **off** - システムで見つかった AMD IOMMU を初期化しません。
- **force\_isolation** - すべてのデバイスのデバイス分離を強制します。IOMMU ドライバーは、必要に応じて分離要件を引き上げることができなくなりました。このオプションは **iommu=pt** をオーバーライドしません。
- **force\_enable** - IOMMU を有効にするとバグがあることがわかっているプラットフォームで、IOMMU を強制的に有効にします。このオプションは注意して使用してください。

#### **acpi.debug\_level = [HW,ACPI,ACPI\_DEBUG]**

形式: <int>

Advanced Configuration and Power Interface (ACPI) デバッグ出力を生成するには、**CONFIG\_ACPI\_DEBUG** を有効にする必要があります。**debug\_layer** のビットは、ACPI ソースファイルの **\_COMPONENT** に対応します。たとえば、**debug\_level** の **#define \_COMPONENT ACPI\_EVENTS** ビットは、**ACPI\_DEBUG\_PRINT** ステートメントのレベルに対応します。たとえば、**ACPI\_DEBUG\_PRINT((ACPI\_DB\_INFO, ...**

**debug\_level** マスクのデフォルトは **info** です。デバッグレイヤーとレベルの詳細については、[Documentation/acpi/debug.txt](#) を参照してください。

プロセッサードライバー情報メッセージを有効にします。

#### **acpi.debug\_layer=0x20000000**

AML の Debug 出力を有効にします。たとえば、AML の解釈中に Debug オブジェクトに保存します。

**acpi.debug\_layer=0xffffffff, acpi.debug\_level=0x2** ACPI ハードウェアに関連するすべてのメッセージを有効にします (**acpi.debug\_layer=0x2, acpi.debug\_level=0xffffffff**)。)

一部の値は、システムが使用できないほど多くの出力を生成します。`log_buf_len` パラメーターは、より多くの出力を取得する必要がある場合に役立ちます。

#### `acpi_mask_gpe = [HW,ACPI]`

形式: <byte> または <bitmap-list>

`_Lxx/_Exx` が存在するため、サポートされていないハードウェアまたはファームウェア機能によってトリガーされる一部の汎用イベント (GPE) によって、GPE ディスパッチャーによって自動的に無効にできない GPE フラッドィングが発生する可能性があります。この機能を使用して、このような制御されていない GPE フラッドィングを防ぐことができます。

#### `cgroup_disable = [KNL]`

形式: <無効にするコントローラーまたは機能の名前>

このパラメーターを使用すると、特定のコントローラーまたはオプション機能を無効にすることができます。

`cgroup_disable = <controller/feature>` の効果は次のとおりです。

- すべての **cgroup** を単一の階層にマウントする場合、**controller/feature** は自動マウントされません
- **controller/feature** は、個別にマウント可能なサブシステムとして表示されません
- **controller/feature** がオプションの機能である場合、その機能は無効になり、対応する **cgroups** ファイルは作成されません  
現在、メモリーコントローラーのみがこれを処理してオーバーヘッドを削減し、他のコントローラーは使用を無効にするだけです。したがって、**cgroup\_disable=memory** だけが実際に価値があります。

`pressure` を指定すると、cgroup ごとの Pressure Stall Information のアカウントィング機能が無効になります。

#### `clearcpuid = BITNUM[,BITNUM...] [X86]`

このパラメーターを使用すると、カーネルの CPUID 機能 X を無効にすることができます。有効なビット番号は、`arch/x86/include/asm/cpufeatures.h` を参照してください。Linux 固有のビットは、カーネルオプションに対して必ずしも安定しているわけではありませんが、ベンダー固有のビットは安定しているはずで、CPUID を直接呼び出すユーザープログラム、または何もチェックせずに機能を使用するユーザープログラムは、引き続き CPUID を認識します。これは、カーネルによって使用されたり、`/proc/cpuinfo` に表示されたりするのを防ぐだけです。また、いくつかの重要なビットを無効にすると、カーネルが誤動作する可能性があることに注意してください。

#### `iommu.strict = [ARM64, X86]`

形式: <"0" | "1">

このパラメーターを使用すると、変換ルックアサイドバッファ (TLB) の無効化動作を設定できます。

可能な値は次のとおりです。

- 0 - レイジーモード。Direct Memory Access (DMA) 解除操作の使用を延期するように要求します。
- 1 - 厳密モード (デフォルト)。DMA アンマップ操作は、IOMMU ハードウェア TLB を同期的に無効にします。  
AMD64 および Intel 64 では、デフォルトの動作は同等のドライバー固有のパラメーターに依存します。ただし、いずれかの方法で明示的に指定された厳格モードが優先されます。

**rcutree.use\_softirq = [KNL]**

このパラメーターをゼロに設定すると、すべての **RCU\_SOFTIRQ** 処理を per-CPU rcuc kthreads に移動します。デフォルトはゼロ以外の値です。これは、**RCU\_SOFTIRQ** がデフォルトで使用されることを意味します。

**rcutree.use\_softirq = 0** を指定して rcuc kthreads を使用します。ただし、**CONFIG\_PREEMPT\_RT=y** カーネルは、このカーネルブートパラメーターを無効にする (強制的にゼロに設定する) ことに注意してください。

**rcupdate.rcu\_normal\_after\_boot = [KNL]**

このパラメーターにより、起動が完了すると、通常の猶予期間プリミティブのみを使用できるようになります。これは、**rcu\_end\_inkernel\_boot()** 呼び出しが呼び出された後です。**CONFIG\_TINY\_RCU** カーネルには影響しません。

**CONFIG\_PREEMPT\_RT=y** 設定のカーネルは、このカーネルブートパラメーターを有効にし、強制的に値 1 に設定します。つまり、迅速な Read-copy-update (RCU) 猶予期間でのブート後の試行を、代わりに通常の非迅速な猶予期間処理を使用するように変換します。

**spectre\_v2 = [X86]**

このパラメーターを使用すると、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。

デフォルトの操作は、カーネルをユーザー空間攻撃から保護します。

可能な値は次のとおりです。

- on - 無条件に有効にします。 **spectre\_v2\_user=on** を意味します
- off - 無条件に無効にし、 **spectre\_v2\_user=off** を意味します
- auto - CPU モデルが脆弱かどうかをカーネルが検出します  
on を選択すると、CPU に応じて実行時に緩和方法が選択されます。使用可能なマイクロコード、**CONFIG\_RETPOLINE** 設定オプションの設定、およびカーネルのビルドに使用されたコンパイラー。

on を選択すると、ユーザー空間からユーザー空間へのタスク攻撃に対する緩和も有効になります。

オフを選択すると、カーネルとユーザー空間の保護の両方が無効になります。

特定の軽減策を手動で選択することもできます。

- retpoline - 間接分岐を置き換える
- retpoline,generic - Retpolines
- retpoline,lfence - LFENCE;間接分岐
- retpoline,amd - retpoline,lfence の別名
- eibrs - 強化された間接分岐制限投機 (IBRS)
- eibrs,retpoline - 強化された IBRS + Retpolines
- eibrs,lfence - 強化された IBRS + LFENCE
- ibrs - IBRS を使用してカーネルを保護する

- `ibrs_always` - IBRS を使用してカーネルとユーザーランドの両方を保護する
- `retpoline,ibrs_user` - 間接ブランチを `retpolines` に置き換え、IBRS を使用してユーザーランドを保護します

このオプションを指定しないことは、**`spectre_v2=auto`** と同等です。

## 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

- MT7921E 802.11ax ワイヤレスドライバー (mt7921e.ko.xz)
- Realtek 802.11ax ワイヤレスコアモジュール (rtw89\_core.ko.xz)
- Realtek 802.11ax ワイヤレス PCI ドライバー (rtw89\_pci.ko.xz)
- ntb\_netdev (ntb\_netdev.ko.xz)
- RDMA 用 Intel® イーサネットプロトコルドライバー (irdma.ko.xz)
- Intel® PCI-E Non-Transparent Bridge Driver (ntb\_hw\_intel.ko.xz)

#### グラフィックドライバーとその他のドライバー

- 汎用カウンターインターフェイス (counter.ko.xz)
- Intel Quadrature Encoder Peripheral ドライバー (intel-qep.ko.xz)
- AMD® PCIe MP2 通信ドライバー (amd\_sfh.ko.xz)
- Thrustmaster (hid-thrustmaster.ko.xz) から一部のハンドルジョイスティックを初期化するドライバー
- HID over I2C ACPI ドライバー (i2c-hid-acpi.ko.xz)
- インテル PMC コアドライバー (intel\_pmc\_core.ko.xz)
- ThinkLMI Driver (think-lmi.ko.xz)
- Processor Thermal Reporting Device ドライバー (int3401\_thermal.ko.xz)
- Processor Thermal Reporting Device Driver (processor\_thermal\_device\_pci.ko.xz)
- Processor Thermal Reporting Device ドライバー (processor\_thermal\_device\_pci\_legacy.ko.xz)
- TI TPS6598x USB Power Delivery Controller ドライバー (tps6598x.ko.xz)

### 6.2. 更新されたドライバー

#### ネットワークドライバー

- Intel® PRO/1000 ネットワークドライバー (e1000e.ko.xz) が更新されました。
- Intel® Ethernet Switch Host Interface ドライバー (fm10k.ko.xz) が更新されました。
- Intel® Ethernet Connection XL710 Network ドライバー (i40e.ko.xz) が更新されました。
- Intel® Ethernet Adaptive Virtual Function Network ドライバー (iavf.ko.xz) が更新されました。
- Intel® Gigabit Ethernet Network ドライバー (igb.ko.xz) が更新されました。

- Intel® Gigabit Virtual Function Network ドライバー (igbvf.ko.xz) が更新されました。
- Intel® 2.5G イーサネット Linux ドライバー (igc.ko.xz) が更新されました。
- Intel® 10 Gigabit PCI Express Network ドライバー (ixgbe.ko.xz) が更新されました。
- Intel® 10 Gigabit Virtual Function Network ドライバー (ixgbev.ko.xz) が更新されました。
- 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.ko.xz) が更新されました。
- VMware vmxnet3 virtual NIC ドライバー (vmxnet3.ko.xz) がバージョン 1.6.0.0-k 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.ko.xz) がバージョン 0:14.0.0.4 に更新されました。
- Broadcom MegaRAID SAS ドライバー (megaraid\_sas.ko.xz) がバージョン 07.719.03.00-rh1 に更新されました。
- LSI MPT Fusion SAS 3.0 デバイスドライバー (mpt3sas.ko.xz) がバージョン 39.100.00.00 に更新されました。
- QLogic Fibre Channel HBA ドライバー (qla2xxx.ko.xz) がバージョン 10.02.06.200-k に更新されました。
- Microchip Smart Family Controller (smartpqi.ko.xz) のドライバーがバージョン 2.1.12-055 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.ko.xz) のスタンドアロン drm ドライバーがバージョン 2.18.1.0 に更新されました。

## 第7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 8.6 のバグで修正されたものを説明します。

### 7.1. インストーラーおよびイメージの作成

**network --defroute オプションが %include スクリプトで正しく機能するようになりました**

以前は、キックスタートインストール中に **%include** スクリプトで使用されると、**network --defroute** オプションが無視されていました。その結果、デバイスはデフォルトルートとして設定されました。

この更新により、キックスタートインストールは **%include** スクリプトで追加された **network --defroute** オプションを無視せず、ネットワーク接続は期待どおりに設定されます。

(BZ#1990145)

**ユーザーは、RHEL for Edge インストーラーブループリントでユーザーアカウントを指定できるようになりました**

以前は、rpm パッケージの追加など、アップグレードのために RHEL for Edge Commit で定義されたユーザーアカウントなしでブループリントの更新を実行すると、アップグレードが適用された後、ユーザーがシステムからロックアウトされていました。これにより、ユーザーは既存のシステムをアップグレードするときにユーザーアカウントを再定義する必要がありました。この問題は、ユーザーが **ostree** コミットの一部としてユーザーを作成するのではなく、インストール時にシステムにユーザーを作成する RHEL for Edge インストーラーの青写真でユーザーアカウントを指定できるように修正されました。

(BZ#1951936)

**osbuild が 4GB を超える ISO イメージのビルドに失敗しなくなりました**

Image Builder ユーザーは、追加のパッケージを追加して、カスタマイズされたイメージを作成できます。パッケージとその依存関係の合計サイズが 4 GB を超えると、RHEL 8.5 以前のリリースのユーザーには次のエラーが表示されます。

```
ubprocess.CalledProcessError: Command '['/usr/bin/xorrisofs', '-verbose', '-V', 'RHEL-8-5-0-BaseOS-x86_64', '-sysid', 'LINUX', '-isohybrid-mbr', '/usr/share/syslinux/isohdpx.bin', '-b', 'isolinux/isolinux.bin', '-c', 'isolinux/boot.cat', '-boot-load-size', '4', '-boot-info-table', '-no-emul-boot', '-rock', '-joliet', '-eltorito-alt-boot', '-e', 'images/efiboot.img', '-no-emul-boot', '-isohybrid-gpt-basdat', '-o', '/run/osbuild/tree/installer.iso', '/run/osbuild/inputs/tree']' returned non-zero exit status 32.
```

この問題は、ISO 9660 Level Of Interchange **-isolevel 3** 引数が **xorrisofs** コマンドに渡されなかったために発生しました。この問題を回避するには、ユーザーは ISO レベルの値を永続的に 3 に変更する必要があります。

RHEL 8.6 リリースでは、この問題が修正され、ユーザーが ISO レベル値を永続的に変更する必要がなくなりました。

(BZ#2056451)

### 7.2. ソフトウェア管理

モジュラーリポジトリで **createrepo\_c --update** を実行すると、モジュラーメタデータが保持されるようになりました



以前は、モジュラーメタデータの元のソースが存在しない状態で既存のモジュラーリポジトリで **createrepo\_c --update** コマンドを実行すると、デフォルトのポリシーでは、モジュラーメタデータを含むすべての追加メタデータがこのリポジトリから削除され、その結果、リポジトリが破損していました。メタデータを保持するには、追加の **--keep-all-metadata** オプションを指定して **createrepo\_c --update** コマンドを実行する必要がありました。

この更新では、追加オプションなしで **createrepo\_c --update** を実行することにより、モジュラーリポジトリにモジュラーメタデータを保持できます。

追加のメタデータを削除するには、新しい **--discard-additional-metadata** オプションを使用できます。

(BZ#1992209)

### 7.3. シェルおよびコマンドラインツール

#### info サブパッケージのインストール中のエラーが発生しなくなる

以前は、**fix-info-dir** スクリプトは **/dev/null** ファイルの存在を予期していました。ソフトウェアドキュメンテーション用の新しいバージョンの **texinfo** パッケージでは、**/dev/null** 特殊ファイルを含まないシステムで **info** サブパッケージのインストールが失敗することはありません。これで、**fix-info-dir** スクリプトは **/dev/null** ファイルの存在を予期せず、無限ループの可能性を回避します。

(BZ#2022201)

#### ReaR は、未使用の LVM 物理ボリュームを持つシステムを正しくバックアップします

以前は、未使用の LVM 物理ボリューム (PV) がシステムに存在する場合、**ReaR** は誤ったディスクレイアウトを生成していました。その結果、**mkrescue**、**mkbackup**、**mkbackuponly**、**savelayout** コマンドなど、ディスクレイアウトを生成する必要がある **ReaR** コマンドは、次のエラーメッセージを出して中止されました。

```
ERROR: LVM 'lvmdev' entry in /var/lib/rear/layout/disklayout.conf where volume_group or device is empty or more than one word
```

この更新により、**ReaR** はディスクレイアウトファイル内の未使用の PV をコメントアウトするようになり、未使用の PV を含むシステムを正しくバックアップできるようになりました。

(BZ#2048454)

#### ReaR はバックアップからマルチパスデバイスを誤って除外しません

以前は、**ReaR** は、バックアップから除外されるべきマルチパスデバイスの名前が名前に含まれている特定のマルチパスデバイスを誤って除外していました。

たとえば、**/dev/mapper/mpatha** という名前のデバイスがバックアップから除外された場合、**/dev/mapper/mpathaa** という名前の 2 番目のデバイスも誤って除外されます。これは、27 台以上のマルチパスデバイスで発生します。

バグは修正され、**ReaR** はマルチパスデバイスを除外する必要がない限り、バックアップから除外しないようになりました。バックアップに含める必要があるマルチパスデバイスがある場合は、**ReaR** 設定ファイルで **AUTOEXCLUDE\_MULTIPATH=n** を指定する必要があることに注意してください。指定しない場合、**ReaR** はすべてのマルチパスデバイスを自動的に除外します。この動作は変更されていません。

(BZ#2049091)

## 7.4. セキュリティー

リモートユーザーがスマートカードにアクセスするように繰り返し求められることはなくなりました

以前は、**pcscd** デーモンの **polkit** ポリシーが誤ってユーザーインタラクションを要求していました。その結果、非ローカルおよび非特権ユーザーはスマートカードにアクセスできず、多数のプロンプトが表示されました。この更新により、**pcsc-lite** パッケージポリシーにインタラクティブプロンプトが含まれなくなりました。その結果、リモートカードユーザーが特権の昇格を繰り返し要求されることがなくなりました。

非特権ユーザーの特権を昇格させるためのポリシーの調整の詳細については、RHEL 製品ドキュメントの [セキュリティ強化の polkit を使用したスマートカードへのアクセスの制御](#) を参照してください。

(BZ#1928154)

FIPS モードでインストールするときに 64 ビット IBM Z システムが起動できなくなることはなくなりました

以前は、**--no-bootcfg** オプションを指定した **fips-mode-setup** コマンドは **zipl** ツールを実行しませんでした。**fips-mode-setup** は初期 RAM ディスク (**initrd**) を再生成し、その結果であるシステムを起動するには **zipl** 内部状態を更新する必要があるため、FIPS モードでインストールした後、64 ビット IBM Z システムは起動できない状態になります。今回の更新で、**--no-bootcfg** で呼び出された場合でも、**fips-mode-setup** が 64 ビットの IBM Z システムで **zipl** を実行するようになり、新たにインストールしたシステムが正常に起動するようになりました。

(BZ#2020295)

**crypto-policies** は OpenSSL の ChaCha20 を無効にできます。

以前は、**crypto-policies** コンポーネントは誤ったキーワードを使用して OpenSSL で ChaCha20 暗号を無効にしていました。そのため、**crypto-policies** では、OpenSSL の TLS 1.2 で ChaCha20 の使用を無効にできませんでした。今回の更新で、**crypto-policies** は、**-CHACHA20-POLY1305** キーワードではなく、**-CHACHA20** キーワードを使用するようになりました。その結果、**crypto-policies** を使用して、TLS 1.2 と TLS 1.3 の両方で OpenSSL の ChaCha20 暗号の使用を無効にできるようになりました。

(BZ#2023734)

**systemd** は **/home/user/bin** からファイルを実行できるようになりました

以前は、SELinux ポリシーにそのようなアクセスを許可するポリシールールが含まれていなかったため、**systemd** サービスは **/home/user/bin/** ディレクトリーからファイルを実行できませんでした。したがって、**systemd** サービスは失敗し、最終的に Access Vector Cache (AVC) を拒否する監査メッセージがログに記録されました。この更新により、欠落していたアクセスを許可する SELinux ルールが追加され、**systemd** サービスが **/home/user/bin/** からコマンドを正しく実行できるようになりました。

(BZ#1860443)

STIG 固有のデフォルトのバナーテキストが他のプロファイルから削除される

以前は、STIG プロファイルのバナーテキストは、CIS などのデフォルトテキストが定義されていない他のプロファイルによってデフォルトとして使用されていました。結果として、これらのプロファイルを使用するシステムは、DISA が要求する特定のテキストで設定されました。この更新により、一般的なデフォルトテキストが作成され、ガイドラインに沿った標準の CIS バナーが定義されました。その結果、テキストバナーを明示的に要求するガイドラインに基づくプロファイルが要件に合わせられ、正しいテキストが設定されるようになりました。

(BZ#1983061)

## ANSSI Enhanced Profile は、"Ensure SELinux State is Enforcing" ルールを正しく選択します

以前は、ANSSI Enhanced プロファイル (**anssi\_bp28\_enhanced**) は "Ensure SELinux State is Enforcing" (**selinux\_state**) ルールを選択していませんでした。この更新によりルールの選択が変更され、ANSSI Enhanced Profile が "Ensure SELinux State is Enforcing" ルールを選択するようになりました。

(BZ#2053587)

## restorecon および seunshareSSG ルールの説明が修正される

以前は、"Record Any Attempts to Run restorecon" (CCE-80699-2) および "Record Any Attempts to Run seunshare" (CCE-80933-5) の説明が正しくありませんでした。今回の更新により、これらのルールの説明は自動化された OVAL チェックに合わせて調整されます。その結果、説明で推奨されている修正を適用すると、これらのルールが正しく修正されるようになりました。

(BZ#2023569)

## CIS プロファイルが IPv6 を自動的に無効にすることはなくなる

以前、RHEL 8 の CIS プロファイルは、推奨事項 3.6 IPv6 の無効化に対して不適切な自動修復を提供していました。これは、IPv6 モジュールがロードされないように `/etc/modprobe.d/ipv6.conf` を設定することで IPv6 を無効にしました。これは、依存する機能やサービスに望ましくない影響を与える可能性があります。RHEL 8 CIS Benchmark v1.0.1 では、推奨事項 3.6 を手動で実装する必要があるため、RHEL8 CIS プロファイルはこの設定アイテムの修正を適用しません。その結果、CIS プロファイルはベンチマークと整合し、IPv6 を自動的に無効にしません。CIS が推奨する GRUB2 または `sysctl` 設定を設定して IPv6 を手動で無効にするには、[Red Hat Enterprise Linux で IPv6 プロトコルを無効または有効にするにはどうすればよいですか？](#) を参照してください。

(BZ#1990736)

## CIS プロファイルが SSH サービスをブロックしなくなりました

以前は、**xccdf\_org.ssgproject.content\_rule\_file\_permissions\_sshd\_private\_key** ルールは、デフォルトで SSH 秘密鍵のアクセス許可を `640` に設定していました。その結果、SSH デーモンが起動しませんでした。この更新により、**file\_permissions\_sshd\_private\_key** ルールが CIS プロファイルから削除され、その結果、SSH サービスが正しく機能します。

(BZ#2002850)

## /usr/share/audit/sample-rules 内のファイルが SCAP ルールで受け入れられるようになりました

以前は、SCAP ルール **xccdf\_org.ssgproject.content\_rule\_audit\_ospp\_general** および **xccdf\_org.ssgproject.content\_rule\_audit\_immutable\_login\_uids** の説明に従って、ユーザーは `/usr/share/audit/sample-rules` ディレクトリーから適切なファイルをコピーすることでシステムを準拠させることができました。ただし、これらのルールの OVAL チェックは失敗したため、スキャン後にシステムは非準拠としてマークされました。この更新により、OVAL チェックは `/usr/share/audit/sample-rules` からのファイルを受け入れるようになり、SCAP ルールは正常に渡されます。

(BZ#2000264)

## ANSSI Kickstart は、十分なディスク容量を予約するようになる

以前は、GUI のインストールには、`/usr` パーティションに予約されている ANSSI Kickstart よりも多くのディスクスペースが必要でした。その結果、RHEL 8.6 GUI のインストールが失敗し、**At least 429 MB more space needed on the /usr filesystem** を示すエラーメッセージが表示されました。この更新

により、`/usr` パーティションのディスク容量が増加し、**scap-security-guide** で提供されている ANSSI Kickstart を使用した RHEL8.6 のインストールが正常に完了するようになりました。

(BZ#2058033)

## GRUB2 引数の修正が永続化される

以前は、カーネル引数を設定する GRUB2 ルールの修正は誤った手順を使用しており、設定の変更はカーネルのアップグレード間で永続的ではありませんでした。結果として、修正はカーネルのアップグレードごとに再適用する必要がありました。この更新では、修復は永続的な方法で GRUB2 を設定する **grubby** ツールを使用します。

(BZ#2030966)

## RHEL 8 ホストからリモートシステムをスキャンするときに **scap-workbench** がハングしなくなりました

以前は、スキャンされたシステムにコンテンツファイルを送信するとハングし、**scap-workbench** ユーティリティーがスキャンを完了できませんでした。これは、実行された Qt サブプロセスをブロックするカーネルのバグが原因でした。結果として、RHEL8 ホストから **scap-workbench** コマンドを使用したリモートシステムのスキャンは機能しません。今回の更新により、根底にあるカーネルのバグが修正されたため、リモートシステムへのファイルのコピー時にリモートスキャンがハングすることがなくなり、正常に終了するようになりました。

(BZ#2051890)

## **usbguard-notifier** が Journal に記録するエラーメッセージの数が適正になりました

以前は、**usbguard-notifier** サービスに **usbguard-daemon** IPC インターフェイスに接続するためのプロセス間通信 (IPC) のパーミッションがありませんでした。したがって、**usbguard-notifier** はインターフェイスへの接続に失敗し、対応するエラーメッセージがジャーナルに書き込まれていました。**usbguard-notifier** は `--wait` オプションで始まるため、デフォルトでは接続障害後に毎秒 **usbguard-notifier** が IPC インターフェイスへの接続を試みるため、ログにはこれらのメッセージが過剰に含まれていました。

今回の更新により、**usbguard-notifier** はデフォルトで `--wait` で開始されなくなりました。サービスは、1秒間隔で3回だけデーモンへの接続を試みます。その結果、ログには最大で3つのエラーメッセージが含まれます。

(BZ#2000000)

## アンビエント機能が `root` 以外のユーザーに正しく適用されるようになりました

安全対策として、UID (ユーザー識別子) をルートから非ルートに変更すると、許可された有効な一連のアンビエント機能セットが無効になっていました。

しかし、アンビエントセットに含まれる機能は許可されたセットと継承可能なセットの両方に含まれている必要があるため、**pam\_cap.so** モジュールはアンビエント機能を設定できません。さらに、たとえば **setuid** ユーティリティーを使用して) UID を変更すると許可されたセットが無効になるため、アンビエント機能を設定できません。

この問題を修正するために、**pam\_cap.so** モジュールは **keepcaps** オプションをサポートするようになりました。これにより、プロセスは、UID をルートから非ルートに変更した後も許可された機能を保持できます。**pam\_cap.so** モジュールは、**defer** オプションもサポートするようになりました。これにより、**pam\_cap.so** は、**pam\_end()** へのコールバック内でアンビエント機能を再適用します。このコールバックは、UID を変更した後、他のアプリケーションで使用できます。



そのため、**su** ユーティリティーおよび **login** ユーティリティーが更新済みで PAM に準拠している場合は、**keepcaps** オプションおよび **defer** オプションを指定して **pam\_cap.so** を使用し、root 以外のユーザーにアンビエント機能を設定できるようになりました。

(BZ#1950187)

**usbguard-selinux** パッケージが **usbguard** に依存しなくなりました。

**usbguard-selinux** パッケージは、以前は **usbguard** パッケージに依存していました。これを、このパッケージの他の依存関係と組み合わせると、**usbguard** のインストール時にファイル競合が発生しました。そのため、特定システムに **usbguard** がインストールされなくなりました。このバージョンでは、**usbguard-selinux** が **usbguard** に依存しなくなったため、**yum** が **usbguard** を正しくインストールできるようになりました。

(BZ#1963271)

**audisp-remote** は、リモートロケーションの可用性を正しく検出できるようになりました

以前は、**audisp-remote** プラグインは、リモートサービスが利用できなくなったことを検出しませんでした。その結果、**audisp-remote** プロセスの CPU 使用率が高くなっていました。今回の更新で、**audisp-remote** は利用できなくなったリモートサービスを適切に検出できるようになりました。したがって、プロセスの CPU 使用率は高くなりません。

(BZ#1906065)

自動ロック解除前に特定の設定で **Clevis** が停止しなくなりました

以前は、LUKS で暗号化されたボリュームの自動ロック解除を実行する **Clevis** ユーティリティーは、特定のシステム設定で停止していました。そのため、暗号化されたボリュームは自動的にロック解除されず、管理者はパスワードを手動で提供する必要があります。場合によっては、管理者が Enter キーを押して暗号化されたボリュームのロックを解除しなければ、**Clevis** は再起動しませんでした。今回の更新で、ユーティリティーはこれらの設定で停止しないように修正され、自動ロック解除のプロセスが正しく機能するようになりました。

(BZ#2018292)

## 7.5. ネットワーク

**NetworkManager** が静的 IPv4 IP アドレスをプライマリーとして使用するようになりました

プライマリーアドレスとセカンダリーアドレスの主な目的は、まだ IP アドレスにバインドされていない接続のソースアドレスを選択できるようにすることです。これらの接続では、カーネルが自動的にアドレスを選択します。**NetworkManager** 接続プロファイルでは、1つの接続に対して静的 IPv4 アドレスと DHCP を同時に設定できます。以前は、DHCP および DHCP サーバーによって提供されるものと同じ範囲の静的 IPv4 アドレスを使用して接続を設定した場合、**NetworkManager** は DHCP サーバーから受け取った IP アドレスをプライマリーとして、静的 IP アドレスをセカンダリーとして誤って割り当てていました。

RHEL 8.6 は、これを意図した動作に変更します。その結果、1つの接続プロファイルで静的 IPv4 アドレスと DHCP の両方を設定すると、静的 IP アドレスが常にプライマリーになり、DHCP サーバーから受け取ったアドレスがセカンダリーになります。さらに、**NetworkManager** は、DHCP サーバーによって割り当てられたルートの **src** 属性も設定するようになりました。この機能により、これらのルートを介して到達可能な宛先は、DHCP サーバーから受け取った IP アドレスを送信元として使用します。

(BZ#2096256)

## 7.6. カーネル

**dmidecode --type 17 コマンドは、DDR5 メモリー情報を正常にデコードするようになりました**

以前は、**dmidecode** コマンドは DDR5 メモリー情報のデコードに失敗していました。そのため、**dmidecode --type 17** は **<OUT OF SPEC>** メッセージを返していました。最新のパッケージ更新 (**dmidecode-3.3-3.el8**) でこの問題が修正されました。その結果、**dmidecode --type 17** は、DDR5 メモリー情報を正常にデコードするようになりました

(BZ#2027665)

**デフォルトのメモリー量を使用する KVM 仮想マシンで kdump が失敗しなくなりました**

以前は、デフォルトのメモリー量を使用する一部のカーネルベースの仮想マシン (KVM) で **kdump** が失敗していました。その結果、クラッシュカーネルは次のエラーでクラッシュダンプファイルをキャプチャできませんでした。

**/bin/sh: error while loading shared libraries: libtinfo.so.6: cannot open shared object file: No such file or directory**

この更新により、問題が修正され、デフォルトのメモリー量を使用する KVM 仮想マシンで **kdump** が正しく機能します。

(BZ#2004000)

**トンネルのオフロードが期待どおりに機能し、利用可能なハードウェアをサポートするようになりました**

以前は、ドライバーは特定の機能フラグを設定していませんでした。したがって、トンネルのオフロードは期待どおりに機能していませんでした。この更新では、ドライバーは必要なフラグを設定してトンネルのオフロードを有効にし、期待どおりに機能します。

(BZ#1910885)

**rx リングバッファを最大に設定する際のカーネル警告を修正しました**

以前は、クリーンな入力を期待する内部関数が、再利用され、すでに初期化された構造で呼び出されていました。これにより、カーネルにより **missing unregister, handled but fix driver** という警告メッセージが表示されました。この更新によりバグが修正され、構造を再初期化してから登録を再度試みるようになりました。

(BZ#2040171)

## 7.7. ファイルシステムおよびストレージ

**バックアップの復元中に xfsrestore コマンドが正しく機能する**

以前は、**xfsdump** コマンドを使用して作成されたバックアップを復元しているときに、**xfsrestore** によって孤立ディレクトリーが作成されました。その結果、いくつかのファイルが作成された孤立ディレクトリーに移動され、次のメッセージが表示されました。

```
# xfsdump -L test -M test -f /scratch.dmp /mnt/test
...
xfsdump: NOTE: root ino 128 differs from mount dir ino 1024, bind mount?
...
xfsdump: Dump Status: SUCCESS

# xfsrestore -f /scratch.dmp /mnt/restore/
...
```

```
xfsrestore: restoring non-directory files
xfsrestore: NOTE: ino 128 salvaging file, placing in orphanage/1024.0/dir17/file60
xfsrestore: NOTE: ino 129 salvaging file, placing in orphanage/1024.0/dir17/file61
xfsrestore: NOTE: ino 130 salvaging file, placing in orphanage/1024.0/dir17/file62
xfsrestore: NOTE: ino 131 salvaging file, placing in orphanage/1024.0/dir17/file63
xfsrestore: NOTE: ino 132 salvaging file, placing in orphanage/1024.0/dir17/file64
xfsrestore: NOTE: ino 133 salvaging file, placing in orphanage/1024.0/dir17/file65
xfsrestore: NOTE: ino 134 salvaging file, placing in orphanage/1024.0/dir17/file66
...
```

今回の更新で問題が修正され、**xfsrestore** が正しく機能するようになりました。

(BZ#2020494)

### **multipathd.socket** ユニットファイルは、起動の試行回数が多すぎても **multipathd** を無効にしなくなりました

以前は、**multipath.service** ユニットファイルの **multipathd** の開始条件は、**multipathd.socket** のトリガー条件とは異なりました。その結果、ユニットファイルは **multipathd** を繰り返し開始しようとして失敗しました。これにより、試行の失敗が多すぎると、**multipathd** が無効になりました。この修正により、**multipathd.socket** と **multipathd.service** の開始条件が同じ値に設定されました。その結果、**multipathd.socket** ユニットファイルは、**multipathd.service** の開始条件が満たされていない場合に **multipathd** を開始しようとしなくなりました。

(BZ#2008101)

### 保護イベントにより、マルチパスデバイスのリロードエラーが発生しなくなりました

以前は、**read-only** パスデバイスが再スキャンされると、カーネルは2つの書き込み保護イベントを送信しました。1つはデバイスが **read/write** に設定され、次はデバイスが **read-only** に設定されています。その結果、パスデバイスで **read/write** イベントが検出されると、**multipathd** はマルチパスデバイスをリロードしようとして失敗しました。これにより、リロードエラーメッセージが表示されました。この更新により、**multipathd** は、デバイスの読み取り/書き込みをリロードする前に、すべてのパスが **read/write** に設定されていることを確認するようになりました。その結果、**multipathd** は、**read-only** デバイスが再スキャンされるたびに **read/write** をリロードしようとしなくなりました。

(BZ#2009624)

## 7.8. コンパイラーおよび開発ツール

### Makefile で使用した場合に **-j** フラグが機能するように

以前は、Makefile 内の MAKEFLAGS に **-j** フラグを追加すると、ターゲットは並列ではなく順次に構築されていました。このバグは修正され、Makefile で **-j** フラグを使用すると同時にターゲットがビルドされるようになりました。

(BZ#2004246)

### 静的にリンクされたアプリケーションがクラッシュしなくなる

以前は、静的にリンクされたバイナリーにリンクされている動的ローダーの初期化コードは、リンクマップ変数を正しく初期化しませんでした。その結果、**LD\_LIBRABY\_PATH** に動的トークン文字列が含まれていると、静的にリンクされたアプリケーションがクラッシュしました。この更新により、静的にリンクされたアプリケーションがクラッシュしなくなりました。

(BZ#1934162)

**glibc の pthread\_once() は、C++ 例外を正しくサポートするように修正されました。**

以前は、**pthread\_once()** の実装により、**libstdc++** ライブラリー関数を使用するときにハングが発生する可能性があります。たとえば、**libstdc++** の **std::call\_once()** は、ハングを引き起こす例外を出力する関数を呼び出しました。この更新により、**pthread\_once()** が修正され、例外が出力されたときにハングしなくなりました。

(BZ#2007327)

## 7.9. IDENTITY MANAGEMENT

**Certmonger は、登録に challengePassword が必要な場合に、AD を使用して SCEP 証明書を自動的に更新できるようになりました。**

以前は、**certmonger** から Active Directory (AD) Network Device Enrollment Service (NDES) サーバーに送信された SCEP 証明書の更新要求には、証明書を最初に取得するために使用された **challengePassword** が含まれていました。ただし、AD は **challengePassword** をワンタイムパスワード (OTP) として扱います。その結果、更新リクエストは拒否されました。

この更新により、**challenge\_password\_otp** オプションが **certmonger** に追加されます。このオプションを有効にすると、**certmonger** が SCEP 更新要求とともに OTP を送信できなくなります。管理者は、AD レジストリーの **HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Cryptography\MSCEP** サブキーに値 **1** の **DisableRenewalSubjectNameMatch** エントリーも追加する必要があります。この変更により、AD は署名者証明書と要求された証明書のサブジェクト名を一致させる必要がなくなりました。その結果、SCEP 証明書の更新は成功します。

SCEP 更新が機能するように **certmonger** と AD サーバーを設定するには、次のようにします。

1. AD サーバーで **regedit** を開きます。
2. **HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Cryptography\MSCEP** サブキーに、新しい 32 ビットの **REG\_DWORD** エントリー **DisableRenewalSubjectNameMatch** を追加し、その値を **1** に設定します。
3. **certmonger** が実行されているサーバーで、**/etc/certmonger/certmonger.conf** ファイルを開き、次のセクションを追加します。

```
[scep]
challenge_password_otp = yes
```

4. **certmonger** を再起動します。

```
# systemctl restart certmonger
```

(BZ#1577570)

**2 番目の FreeRADIUS サーバーが使用できない場合でも、FreeRADIUS プロキシサーバーは動作を停止しなくなりました**

FreeRADIUS サーバーがプロキシサーバーとして設定されている場合、要求メッセージを別の FreeRADIUS サーバーに転送します。以前は、これら 2 つのサーバー間の接続が中断された場合、FreeRADIUS プロキシサーバーは機能しなくなりました。この修正により、FreeRADIUS プロキシサーバーは、他のサーバーが使用可能になったときに接続を再確立できるようになりました。

(BZ#2030173)



PBKDF2 ハッシュパスワードを使用した FIPS モードの Directory Server への認証が期待どおりに機能するようになりました。

Directory Server が FIPS (Federal Information Processing Standard) モードで実行している場合は、**PK11\_ExtractKeyValue()** 機能を使用できません。これにより、パスワードベースの鍵派生機能 2 (PBKDF2) のハッシュパスワードを持つユーザーは、FIPS モードが有効になっているときにサーバーを認証できませんでした。今回の更新で、Directory Server が **PK11\_Decrypt()** 機能を使用してパスワードハッシュデータを取得するようになりました。その結果、FIPS モードでの Directory Server への認証が、PBKDF2 ハッシュパスワードを使用するユーザーに対して機能するようになりました。

(BZ#2033398)

SSSD キャッシュが SSSD ユーザーとして **tmpfs** にマウントされている場合、SSSD のソケット アクティベーションは成功します。

以前は、**sssd** ユーザーが **/var/lib/sss/db/config.ldb** SSSD 設定ファイルを所有していなかったため、SSSD キャッシュが **tmpfs** 一時ファイルシステムにマウントされている場合は、SSSD のソケット アクティベーションが失敗していました。今回の修正により、SSSD は **sssd** ユーザーとして **config.ldb** ファイルを作成し、ソケットのアクティベーションが成功するようになりました。**/var/lib/sss/db/** SSSD ディレクトリーを **tmpfs** にマウントした場合は、**sssd** ユーザーとして再マウントして、SSSD がその場所に **config.ldb** ファイルを作成できるようにする必要があります。



### 警告

以下の手順は、[Identity Management](#) でのパフォーマンスのチューニング ガイドの手順に従ってパフォーマンスを高速化するために SSSD キャッシュを **tmpfs** にマウントした場合にのみ実行してください。標準的な状況では、Red Hat は SSSD キャッシュのデフォルトの場所である標準ディスクストレージを代わりに使用することを推奨します。

## 手順

1. **/var/lib/sss/db** がマウントポイントであることを確認します。

```
# mount -t tmpfs | grep /var/lib/sss/db
tmpfs on /var/lib/sss/db type tmpfs
(rw,relatime,rootcontext=system_u:object_r:sss_var_lib_t:s0,seclabel,size=307200k,mode=700)
```

2. **/var/lib/sss/db** が有効なマウントポイントである場合は、**root** ユーザーが所有しているかどうかを確認します。

```
# ls -l /var/lib/sss | grep db
drwx-----. 2 *root root* 40 Jul 26 04:48 db
```

3. **db** ディレクトリーがマウントポイントであり、**root** ユーザーが所有している場合は、**/etc/fstab** ファイルの対応するエントリーに **uid=sssdb,gid=sssdb** を追加して、SSSD ユーザーとしてマウントします。

```
tmpfs /var/lib/sss/db/ tmpfs
size=300M,mode=0700,*uid=sssd,gid=sssd*,rootcontext=system_u:object_r:sssd_var_lib_t:s0
0 0
```

- ディレクトリーを再マウントし、SSSD サービスを再起動します。

```
# systemctl stop sssd
# umount /var/lib/sss/db
# mount /var/lib/sss/db
# systemctl start sssd
```

## 検証

- `/var/lib/sss/db` ディレクトリーが **sssd** ユーザーによって所有されていることを確認します。

```
# ls -l /var/lib/sss | grep db
drwx-----. 2 sssd sssd 160 Jul 26 05:00 db
```

(BZ#2108316)

## 7.10. グラフィックインフラストラクチャー

### VGA ディスプレイを備えた Matrox GPU が期待どおりに動作するようになりました

このリリースより前は、次のシステム設定を使用した場合、ディスプレイにグラフィカル出力が表示されませんでした。

- Matrox MGA G200 ファミリーの GPU
- VGA コントローラーで接続されたディスプレイ
- UEFI のレガシーモードへの切り替え

したがって、この設定で RHEL を使用またはインストールできませんでした。

今回の更新により、**mgag200** ドライバーが大幅に書き直され、その結果、グラフィック出力が期待どおりに機能するようになりました。

(BZ#1953926)

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

### Grafana admin パスワードが変更された場合でも、Metrics ロールを使用する Playbook は複数回の実行で正常に完了します

以前は、**metrics\_graph\_service: yes** ブール値で Metrics ロールを実行した後に Grafana **admin** ユーザーパスワードを変更すると、Metrics ロールの後続の実行が失敗していました。これにより、Metrics ロールを使用した Playbook が失敗し、影響を受けるシステムはパフォーマンス分析用にのみ部分的に設定されました。現在、Metrics ロールは、Grafana **deployment** API が利用可能であり、必要な設定アクションを実行するためにユーザー名またはパスワードの知識を必要としない場合に、その API を使用します。その結果、管理者が Grafana **admin** パスワードを変更した場合でも、Metrics ロールを使用する Playbook は複数回の実行で正常に完了します。

(BZ#1967321)

## SSHD システムロールは正しいテンプレートファイルを使用する

RHEL 8.5 では、SSHD System Role が間違ったテンプレートファイルを使用していました。結果として、生成された `sshd_config` ファイルには `# Ansible managed` コメントが含まれていませんでした。欠落しているコメントは、システムの機能に影響を与えませんでした。この更新により、システムロールは正しいテンプレートファイルを使用し、`sshd_config` には正しい `# Ansible managed` コメントが含まれます。

(BZ#2040038)

## IPv6 が無効になると、Networking システムロールが DNS 検索ドメインの設定に失敗しなくなりました。

以前では、IPv6 プロトコルが無効になっている場合、`libnm` ライブラリーの `nm_connection_verify()` 機能は DNS 検索ドメインを無視していました。そのため、Networking RHEL システムロールを使用し、`dns_search` を `ipv6_disabled: true` と一緒に設定すると、システムロールに障害が発生し、以下のエラーが表示されます。

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

今回の更新で、IPv6 が無効になっている場合、`nm_connection_verify()` 機能は DNS 検索ドメインを無視します。これにより、IPv6 が無効になっていても、期待どおりに `dns_search` を使用できます。

(BZ#2041627)

## Networking システムロールの nm プロバイダーがブリッジを正しく管理するようになりました

以前は、`initscripts` プロバイダーを使用した場合、Networking システムロールは、ブリッジインターフェイスをマネージド外としてマークするように NetworkManager を設定する `ifcfg` ファイルを作成しました。また、NetworkManager はフォローアップ `initscript` アクションを検出できませんでした。たとえば、`initscript` プロバイダーの `down` および `absent` アクションは、`down` および `absent` アクションの後に接続をリロードしない場合は、このインターフェイスのアンマネージ状態に関する NetworkManager の理解を変更しません。この修正により、Networking システムロールは `NM.Client.reload_connections_async()` 関数を使用して、NetworkManager 1.18 を使用してマネージドホストに NetworkManager をリロードします。その結果、NetworkManager は、プロバイダーを `initscript` から `nm` に切り替えるときにブリッジインターフェイスを管理します。

(BZ#2034908)

## SSH サーバーロールは、FIPS モードを検出し、FIPS モードでタスクを正しく処理するようになりました

以前は、RHEL 8 以前のシステムを FIPS モードで管理する場合、デフォルトのホストキーの1つを作成することは許可されていませんでした。そのため、SSH サーバーロールの操作では、呼び出し時に `not allowed key` タイプを生成できませんでした。この修正により、SSH サーバーロールは FIPS モードを検出し、それに応じてデフォルトのホストキーリストを調整するようになりました。その結果、SSH サーバーロールは、デフォルトのホストキー設定を使用して FIPS モードでシステムを管理できるようになりました。

(BZ#1979714)

## Logging システムロールがタスクを複数回呼び出さなくなりました

以前は、Logging ロールは1回だけ呼び出すべきタスクを複数回呼び出していました。そのため、余分なタスク呼び出しによりロールの実行速度が低下していました。この修正では、Logging ロールが変更され、タスクを1回だけ呼び出すようになり、Logging ロールのパフォーマンスが向上しました。

(BZ#2005727)

## RHEL システムロールは、生成されたファイル内の複数行の `ansible_managed` コメントを処理するようになりました

以前は、一部の RHEL システムロールは `#{ansible_managed}` を使用して一部のファイルを生成していました。そのため、顧客が複数行のカスタム `ansible_managed` 設定を持っている場合、ファイルは正しく生成されませんでした。今回の修正により、すべてのシステムロールでファイルの生成時に `{{ ansible_managed | comment }}` と同等のものが使用されるようになり、複数行の `ansible_managed` 値を含め、`ansible_managed` 文字列は常に正しくコメントされるようになりました。その結果、生成されたファイルには正しい複数行の `ansible_managed` 値が含まれます。

(BZ#2006231)

## Logging ロールは、`immark` モジュールの間隔値の引用を見逃さなくなりました

以前は、`immark` モジュールが適切に設定されていなかったため、`immark` モジュールの `interval` フィールド値が適切に引用されていませんでした。今回の修正により、`interval` の値が適切に引用されるようになりました。これで、`immark` モジュールは期待どおりに機能します。

(BZ#2021678)

## `group` オプションによりグループが証明書にアクセスできなくなる状況が解消されました。

以前は、証明書のグループを設定する際に、`mode` はグループの読み取り権限を許可するように設定されませんでした。そのため、グループメンバーは、Certificate ロールが発行した証明書を読み取ることができませんでした。今回の修正で、グループ設定により、ファイルモードにグループ読み取り権限が確実に含まれるようになりました。その結果、グループの Certificate ロールにより発行された証明書に、グループメンバーがアクセスできます。

(BZ#2021683)

## `/etc/tuned/kernel_settings/tuned.conf` ファイルには適切な `ansible_managed` ヘッダーがあります

以前は、カーネル設定の RHEL システムロールには、`/etc/tuned/kernel_settings/tuned.conf` ファイルの `ansible_managed` ヘッダーにハードコードされた値がありました。その結果、ユーザーはカスタムの `ansible_managed` ヘッダーを提供できませんでした。この更新で問題が修正され、`kernel_settings` が `/etc/tuned/kernel_settings/tuned.conf` のヘッダーをユーザーの `ansible_managed` 設定で更新するようになりました。つまり、`/etc/tuned/kernel_settings/tuned.conf` には適切な `ansible_managed` ヘッダーがあります。

(BZ#2047504)

## `logging_purge_confs` オプションは、不要な設定ファイルの削除に失敗しなくなりました

以前は、`logging_purge_confs` 変数は、不要なロギング設定ファイルを削除するために準備されていましたが、それらをクリーンアップできませんでした。その結果、`logging_purge_confs` 変数が `true` に設定されていても、不要な設定ファイルはクリーンアップされず、設定ディレクトリーに残されました。この問題は修正され、`logging_purge_confs` 変数が次のように機能するように再定義されました。

- `logging_purge_confs` が `true` に設定されている場合は、`rpm` パッケージに属していない `rsyslog.d` 内のファイルを削除します。これには、前回の `logging` ロールの実行によって生成された設定ファイルが含まれます。`logging_purge_confs` のデフォルト値は `false` です。

(BZ#2040812)

## 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#2064388)

## Metrics ロールによる設定がシンボリックリンクを正しくたどるようになりました

**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#2058655)

## カーネル設定システムロールが **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、カーネル設定ロールは **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#2058772)

## Kdump システムロールはホストを無視しなくなりました

以前は、Kdump ロールがクラッシュカーネル用に予約されたメモリを持たないマネージドノードを無視し、その結果、システムを正しく設定していなくても成功ステータスで完了していました。このロールは、マネージドノードにクラッシュカーネル用に予約されたメモリがない場合は失敗し、マネージドノード上で **kdump** を正しく設定するために、ユーザーに **kdump\_reboot\_ok** 変数を **true** に設定するように求めるように再設計されました。その結果、Kdump ロールはホストを無視しなくなり、正しい設定で正常に完了するか、問題を修正するためにユーザーが何を行う必要があるかを説明するエラーメッセージが表示されて失敗します。

(BZ#2029605)

## Firewall システムロールは、**target** が変更されるとすぐにファイアウォールをリロードするようになりました

以前は、**target** パラメーターが変更されても Firewall システムロールはファイアウォールを再ロードしませんでした。この修正により、**target** が変更されると Firewall ロールがファイアウォールを再ロードするようになりました。その結果、**target** の変更は即座に行われ、後続の操作で使用できるようになりました。

(BZ#2057172)

## HA Cluster システムロールのデフォルトの **pcsd** 権限により、グループ **haclient** へのアクセスが許可されるようになりました



以前は、ユーザーが `ha_cluster_pcs_permission_list` 変数で設定されたデフォルトの `pcsd` 権限で HA Cluster システムロールを実行した場合、グループ `hacluster` のメンバーのみがクラスターにアクセスできました。今回の修正で、デフォルトの `pcsd` 権限により、グループ `haclient` がクラスターを管理できるようになり、`haclient` のすべてのメンバーがクラスターにアクセスして管理できるようになりました。

(BZ#2049747)

## 7.12. 仮想化

**strict NUMA バインディングポリシーにより、ランタイムメモリーの移動が許可されなくなりました**

以前は、**strict** NUMA バインディングポリシーが仮想マシン (`<memory mode='strict'/>`) で有効になっていると、ランタイムメモリーをその仮想マシンから別の NUMA ノードに移動しようとすると、部分的または完全に失敗する場合があります。この問題を回避するために、**strict** ポリシーによりランタイムメモリーの移動が完全に禁止されるようになりました。

さらに、**restrictive** ポリシーが追加されました。これは、以前の **strict** ポリシーと同様に機能します。つまり、ランタイムメモリーを他の NUMA ノードに移動することはできますが、メモリーが完全に移動することを保証することはできません。

(BZ#2014369)

**multifd 移行が確実に機能するようになりました**

以前は、QEMU の **multifd** 機能を使用して仮想マシンを移行しようとすると、移行に失敗し、仮想マシンが予期せず終了していました。基になるコードが修正され、**multifd** 移行が期待どおりに機能するようになりました。

(BZ#1982993)

**virtio-balloon が原因で仮想マシンの移行とスナップショットが失敗しなくなりました**

以前は、仮想マシン (VM) が **virtio-balloon** デバイスを使用している場合、より新しいゲストオペレーティングシステム (RHEL 9 など) で仮想マシン (VM) を移行しようとすると失敗していました。同様に、そのような仮想マシンのスナップショットの作成は失敗しました。今回の更新により、**virtio-balloon** の **page poison** 機能のバグが修正され、説明されている問題が発生しなくなります。

(BZ#2004416)

**PowerVM 上の IBMVFC デバイスのホットアンプラグが期待どおりに機能するようになる**

以前は、PowerVM ハイパーバイザー上の RHEL 8 ゲストオペレーティングシステムで仮想マシン (VM) を使用する場合は、実行中の仮想マシンから IBM Power Virtual Fibre Channel (IBMVFC) デバイスを削除しようとすると失敗します。代わりに、**outstanding translation** エラーが表示されます。基礎となるコードが修正され、IBMVFC デバイスのライブホットアンプラグが PowerVM で正しく機能するようになりました。

(BZ#1959020)

## 7.13. コンテナ

**RHEL 8.5 以前で fuse-overlays を使用して作成されたルートレスコンテナは、削除されたファイルを認識するようになりました**

RHEL 8.4 以前では、ルートレスイメージやコンテナは `fuse-overlays` ファイルシステムを使用して

作成または保存されていました。RHEL 8.5以降でこのようなイメージやコンテナを使用すると、カーネルが提供する overlayfs の実装を使用している権限のないユーザーが、RHEL 8.4 のコンテナやイメージからファイルやディレクトリーを削除した場合に問題が発生します。この問題は、root アカウントで作成されたコンテナには該当しませんでした。

その結果、コンテナやイメージから削除するファイルやディレクトリーは、fuse-overlayfs ファイルシステムを使用する場合、ホワイトアウトフォーマットを使用して、そのようにマークされていました。しかし、フォーマットの違いにより、カーネルの overlayfs の実装では、fuse-overlayfs を使用して作成されたホワイトアウトフォーマットを認識できませんでした。その結果、削除されたファイルやディレクトリーはそのまま表示されていました。この問題は、root アカウントで作成されたコンテナには該当しませんでした。

今回の更新で問題は解決しました。

(JIRA:RHELPLAN-92741)

## 第8章 テクノロジープレビュー

ここでは、Red Hat Enterprise Linux 8.6 で利用可能なすべてのテクノロジープレビュー機能のリストを提示します。

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。

### 8.1. RHEL FOR EDGE

#### FDO プロセスがテクノロジープレビューとして利用可能に

RHEL for Edge イメージの自動プロビジョニングおよびオンボード用の FDO プロセスは、テクノロジープレビューとして利用できます。これにより、RHEL for Edge Simplified Installer イメージを構築し、それを RHEL for Edge イメージにプロビジョニングし、FDO(FIDO デバイスオンボーディング) プロセスを使用して、Edge デバイスを自動的にプロビジョニングおよびオンボーディングし、ネットワークに接続されている他のデバイスやシステムとデータを交換できます。その結果、FIDO デバイスのオンボーディングプロトコルは、製造段階でデバイスの初期化を実行し、次に実際にデバイスを使用するための遅延バインディングを実行します。

(BZ#1989930)

### 8.2. 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)

### 8.3. 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](#))

## 8.4. 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)

## 8.5. 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)

### Stratis 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。

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)

## 8.6. 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せん。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)

## 8.7. 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)

## 8.8. 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)**、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)

## 8.9. 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)

## 8.10. WEB コンソール

### Stratis が RHEL Web コンソールでテクノロジープレビューとして利用可能

今回の更新で、Red Hat Enterprise Linux Web コンソールは、Stratis ストレージをテクノロジープレビューとして管理できるようになりました。

Stratis の詳細は、[Stratis とは](#) を参照してください。

(JIRA:RHELPLAN-108438)

## 8.11. 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)

## 8.12. 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。

これまでの Toolbox ユーティリティーは、RHEL CoreOS の [github.com/coreos/toolbox](https://github.com/coreos/toolbox) をベースにしています。今回のリリースでは、Toolbox は [github.com/containers/toolbox](https://github.com/containers/toolbox) に置き換えられました。

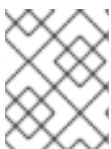
(JIRA:RHELPLAN-77238)

### Netavark ネットワークスタックは、テクノロジープレビューとして利用できます

Podman バージョン 4.1.1-7 より前のバージョンでは、コンテナ用の Netavark ネットワークスタックがテクノロジープレビューとして利用できます。

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-dns** 権限のある DNS サーバーを使用する必要があります。

(JIRA:RHELPLAN-137622)

### 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:RHELDPCS-16861)

## 第9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。

### 9.1. 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)

## 9.2. 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)

## 9.3. 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)

### 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 [RHEL7](#) および [RHEL8](#) では、`/proc` を `hidepid=2` でマウントすることが推奨されますか？を参照してください。

(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)

## 9.4. 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の上記記事 [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コープの導入により、**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、スコープを指定しない **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)

## 9.5. 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)

### **cgdcbxd** 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 8.6 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せん。RHEL 8.6 では、**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、ネイティブ一致で **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ーを使用して **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](#))

### slaves 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、[多様性を受け入れるオープンソースの強化](#) を参照してください。したがって、**slaves** 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:RHELDOS-17641](#))

## 9.6. 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 か月から 6 か月に減少します。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 か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。たとえば、8.4.x には 1 年間のサポートウィンドウがありますが、8.4.x+1 には 6 か月があります。

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)

**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)

## 9.7. 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)

## 9.8. 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 ポリ्यू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ポリ्यूームは、**lvcreate** コマンドを使用して変換できます。LVM-VDO 実装の詳細は、[RHEL 9 の論理ポリ्यू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)

## 9.9. 高可用性およびクラスター

### clutter 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 **clutter** 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)

## 9.10. 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)

## 9.11. 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、`dnf` 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、`--allow-erasing` 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)

## 9.12. 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\_enctypes** 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\_enctypes**、**default\_tgs\_enctypes**、**default\_tkt\_enctypes** 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せん。
- **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、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

## 9.13. 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)

## 9.14. 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)

## 9.15. 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。ブラウザー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)

**remotectl** コマンドは非推奨になりました

**remotectl** コマンドは非推奨となり、RHEL の将来のリリースでは使用できなくなります。代わりに **cockpit-certificate-ensure** コマンドを使用できます。ただし、**cockpit-certificate-ensure** には

**remotectl** と同等の機能がないことに注意してください。バンドルされた証明書とキーチェーンファイルはサポートされていないため、それらを分割する必要があります。

(JIRA:RHELPLAN-147538)

## 9.16. 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**の警告を表示します。

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 8.6 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

**geopupdate** 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、**geopupdate** 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)

## 9.17. 仮想化

### 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)

### 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)

## 9.18. 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)

## 9.19. 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
- 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
- cal10n
- 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
- 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-emf

- eclipse-license
- 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-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-sqlite
- freeradius-unixODBC
- fuse-sshfs
- fusesource-pom

- future
- gamin
- gamin-devel
- gavl
- gcc-toolset-10
- gcc-toolset-10-annobin
- gcc-toolset-10-binutils
- gcc-toolset-10-binutils-devel
- gcc-toolset-10-build
- gcc-toolset-10-dwz
- gcc-toolset-10-dyninst
- gcc-toolset-10-dyninst-devel
- gcc-toolset-10-elfutils
- gcc-toolset-10-elfutils-debuginfod-client
- gcc-toolset-10-elfutils-debuginfod-client-devel
- gcc-toolset-10-elfutils-devel
- gcc-toolset-10-elfutils-libelf
- gcc-toolset-10-elfutils-libelf-devel
- gcc-toolset-10-elfutils-libs
- gcc-toolset-10-gcc
- gcc-toolset-10-gcc-c++
- gcc-toolset-10-gcc-gdb-plugin
- gcc-toolset-10-gcc-gfortran
- gcc-toolset-10-gdb
- gcc-toolset-10-gdb-doc
- gcc-toolset-10-gdb-gdbserver
- gcc-toolset-10-libasan-devel
- gcc-toolset-10-libatomic-devel
- gcc-toolset-10-libitm-devel

- gcc-toolset-10-libsan-devel
- gcc-toolset-10-libquadmath-devel
- gcc-toolset-10-libstdc++-devel
- gcc-toolset-10-libstdc++-docs
- gcc-toolset-10-libtsan-devel
- gcc-toolset-10-libubsan-devel
- gcc-toolset-10-ltrace
- gcc-toolset-10-make
- gcc-toolset-10-make-devel
- gcc-toolset-10-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
- 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
- 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
- jffi

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

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



- `libertas-sd8686-firmware`
- `libertas-usb8388-firmware`
- `libertas-usb8388-olpc-firmware`
- `libgdither`
- `libGLEW`
- `libgovirt`
- `libguestfs-benchmarking`
- `libguestfs-devel`
- `libguestfs-gfs2`
- `libguestfs-gobject`
- `libguestfs-gobject-devel`
- `libguestfs-java`
- `libguestfs-java-devel`
- `libguestfs-javadoc`
- `libguestfs-man-pages-ja`
- `libguestfs-man-pages-uk`
- `libguestfs-tools`
- `libguestfs-tools-c`
- `libhugetlbfs`
- `libhugetlbfs-devel`
- `libhugetlbfs-utils`
- `libIDL`
- `libIDL-devel`
- `libidn`
- `libiec61883`
- `libindicator-gtk3`
- `libindicator-gtk3-devel`
- `libiscsi-devel`
- `libjose-devel`

- libkcc
- libkcc-common
- libkcc-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
- 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
- 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
- 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
- qpid-proton
- qrencode
- qrencode-devel
- qrencode-libs
- qt5-qtcanvas3d
- qt5-qtcanvas3d-examples
- rarian
- rarian-compat

- re2c
- recode
- redhat-menus
- redhat-support-lib-python
- redhat-support-tool
- reflections
- regexp
- relaxngDatatype
- rasm-gtk
- rpm-plugin-priorset
- rpmdm
- rsyslog-udpspoof
- ruby-hivex
- ruby-libguestfs
- rubygem-abrt
- rubygem-abrt-doc
- rubygem-bson
- rubygem-bson-doc
- rubygem-mongo
- rubygem-mongo-doc
- 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
- 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

## 9.20. 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9.1 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)
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]

デバイス ID	ドライ バー	デバイス名
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
	myri10g e	Myricom 10G driver (10GbE)
	netxen_ nic	QLogic/NetXen (1/10) GbE Intelligent Ethernet Driver
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

デバイス ID	ドライ バー	デバイス名
	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9.2 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
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10章 既知の問題

このパートでは Red Hat Enterprise Linux 8.6 の既知の問題を説明します。

### 10.1. 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、インストーラーに失敗し、**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)

## 10.2. 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)

## 10.3. 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)

セキュリティー DNF のアップグレードでは、廃止されたパッケージをスキップできます

[RHBA-2022:5816](#) アドバイザリーでリリースされた [BZ#2095764](#) のパッチでは、次のリグレーションが導入されました。DNF アップグレードで **--security** オプションのようなセキュリティーフィルターを使用すると、廃止されたパッケージのアップグレードをスキップすることができます。この問題は、イ

インストール済みのパッケージが別の利用可能なパッケージによって廃止され、利用可能なパッケージのアドバイザーが存在する場合に特に発生します。

その結果、**dnf** は古いパッケージをシステムに残し、セキュリティーアップグレードが完全には実行されず、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完全なアップグレードを実行するか、最初に、アップグレードプロセスに古いパッケージが含まれていないことを確認します。

([BZ#2095764](#))

## 10.4. シェルおよびコマンドラインツール

**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](#))

## 10.5. 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](#))

**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](#))

## 10.6. 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)

### 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)

### 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、修正を回実行して、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 つのルールに影響を受けます。

```
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 **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)

### Ansible 修復には追加のコレクションが必要

**ansible-core** パッケージによる Ansible Engine の置き換えにより、RHEL サブスクリプションで提供される Ansible モジュールのリストが削減されました。これにより、**scap-security-guide** パッケージに含まれる Ansible コンテンツを使用する修復を実行するには、**rhc-worker-playbook** パッケージからのコレクションが必要です。

Ansible 修復の場合は、以下の手順を実行します。

1. 必要なパッケージをインストールします。

```
# dnf install -y ansible-core scap-security-guide rhc-worker-playbook
```

2. **/usr/share/scap-security-guide/ansible** ディレクトリーに移動します。

```
# cd /usr/share/scap-security-guide/ansible
```

3. 追加の Ansible コレクションへのパスを定義する環境変数を使用して、関連する Ansible Playbook を実行します。

```
# ANSIBLE_COLLECTIONS_PATH=/usr/share/rhc-worker-playbook/ansible/collections/ansible-playbook -c local -i localhost, rhel9-playbook-cis_server_I1.yml
```

**cis\_server\_I1** を、システムを修正するプロファイルの ID に置き換えます。

これにより、Ansible コンテンツは正しく処理されます。



## 注記

**rhc-worker-playbook** で提供されるコレクションのサポートは、**scap-security-guide** から取得する Ansible コンテンツの有効化だけに限定されます。

(BZ#2114981)

## 10.7. ネットワーク

### 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)

### インスタンスのプライマリー IP アドレスは、Alibaba Cloud で nm-cloud-setup サービスを開始した後に変更されます

Alibaba Cloud でインスタンスを起動した後、**nm-cloud-setup** サービスはプライマリー IP アドレスをインスタンスに割り当てます。ただし、インスタンスに複数のセカンダリー IP アドレスをインスタンスに割り当てて **nm-cloud-setup** サービスを開始すると、以前のプライマリー IP アドレスはすでに割り当てられているセカンダリー IP アドレスの1つに置き換えられます。返されたメタデータのリストは、同じことを確認します。この問題を回避するには、プライマリー IP アドレスが変更されないように、セカンダリー IP アドレスを手動で設定します。その結果、インスタンスは両方の IP アドレスを保持し、プライマリー IP アドレスは変更されません。

(BZ#2079849)

### 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



タイプにします。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](#))

### 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、RoCE インターフェイスの IP 設定が失われる

RDMA over Converged Ethernet (RoCE) インターフェイスは、次の両方の条件が満たされた場合、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 IP 設定を失います。

- ユーザーが RHEL 8.6 以前のシステムからアップグレードする。
- RoCE カードが UID によって列挙されている。

この問題を回避するには、以下を実行します。

1. 次の内容を含む `/etc/systemd/network/98-rhel87-s390x.link` ファイルを作成します。

```
[Match]
Architecture=s390x
KernelCommandLine=lnet.naming-scheme=rhel-8.7

[Link]
NamePolicy=kernel database slot path
AlternativeNamesPolicy=database slot path
MACAddressPolicy=persistent
```

2. システムを再起動して、変更を有効にします。
3. RHEL 8.7 以降にアップグレードします。

機能 ID (FID) によって列挙され、一意ではない RoCE インターフェイスは、`net.naming-scheme=rhel-8.7` カーネルパラメーターを設定しない限り、引き続き予測できないインターフェイス名を使用することに注意してください。この場合、RoCE インターフェイスは "ens" 接頭辞が付いた予測可能な名前に切り替わります。

([BZ#2169382](#))

## 10.8. カーネル

v1 モードで `net_prio` または `net_cls` コントローラーを使用すると、`cgroup-v2` 階層の一部のコントローラーが非アクティブ化されます。

`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ります。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` 階層がアクティブにならず、`dmesg` コマンドは次のメッセージを報告します。

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

(BZ#2046396)

### 暗号化されたデバイスのパスフレーズを入力すると、Anaconda が失敗することがある

インストールの準備時に `kdump` が無効になっており、ユーザーが暗号化されたディスクのパーティション分割を選択した場合は、暗号化されたデバイスのパスフレーズを入力すると、Anaconda インストーラーはトレースバックで失敗します。

この問題を回避するには、以下のいずれかを実行します。

- `kdump` を無効にする前に、暗号化されたディスクパーティションを作成します。
- インストール中は `kdump` を有効にしておき、インストールプロセスが完了したら無効にします。

(BZ#2086100)

### 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場合があります

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` ユーティリティーは、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` ユーティリティー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ー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` ユーティリティー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、代わりにスタッ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)

### 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 0x30000000-0x31fffff 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 PPL (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)

### 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る場合があります。



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)

## 10.9. 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)

### NVMe/TCP ドライバーでデバイスマッパーマルチパスを使用すると、システムが不安定になる

DM マルチパスは、NVMe/TCP ドライバーではサポートされていません。これを使用すると、カーネル内のスリープ関数が Atomic コンテキストで呼び出され、システムが不安定になります。

この問題を回避するには、ネイティブ NVMe マルチパスを有効にします。DM マルチパスツールは使用しないでください。RHEL 8 の場合、オプション **nvme\_core.multipath=Y** をカーネルコマンドラインに追加します。



(BZ#2022359)

### 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します

**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ます、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)

## 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

### 32 ビットアプリケーションで呼び出されると `getpwnam()` が失敗する場合がある

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)

### OQGraph プラグインが有効な場合に、MariaDB 10.5 が存在しないテーブルの破棄について警告しない

OQGraph ストレージエンジンプラグインが **MariaDB 10.5** サーバーに読み込まれると、**MariaDB** は存在しないテーブルの削除について警告しません。特に、ユーザーが `DROP TABLE` コマンドまたは **DROP TABLE** 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、**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。

OQGraph プラグインは **mariadb-oqgraph-engine** パッケージにより提供されることに注意してください。デフォルトではインストールされません。

(BZ#1944653)

### 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)

## 10.11. IDENTITY MANAGEMENT

### Windows Server 2008 R2 以前に対応しなくなった

RHEL 8.4 以降では、Identity Management (IdM) は、Windows Server 2008 R2 以前のバージョンを実行している Active Directory ドメインコントローラーとの間で Active Directory への信頼を確立することに対応していません。RHEL IdM との信頼関係を確立する際に、SMB 暗号化が必要になりました。これは、Windows Server 2012 以降でのみ利用可能です。

(BZ#1971061)

**--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。

NLMSSP 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)

### 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。

- 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。
- 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。

- 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。
- 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。
- 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。
- 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、以下のエラーを記録します。

```
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)

### 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)

## 10.12. 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 ユーティリティーの 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 Server 2019 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)

## 10.13. 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時、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)

## 10.14. 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しない

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 VM に渡された後に変更される場合があります。結果として、Web コンソールを使用してそ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に失敗します。この問題を回避するには、VM の XML 設定から USB デバイスの `<hostdev>` 部分を削除します。

(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は機能しない

Web コンソールを使用して仮想マシン (VM) 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、残りは無視されます。この問題を回避するには、一度に1つのデバイスのみを接続します。

(JIRA:RHELPLAN-115603)

## 10.15. 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 2.12** 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 2.12** 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)

## 10.16. 仮想化

仮想マシンのネットワークトラフィックのパフォーマンスが低下する可能性があります

場合によっては、RHEL 8.6 ゲスト仮想マシン (VM) は、高レベルのネットワークトラフィックを処理するときにパフォーマンスをいくらか低下させます。

(BZ#2069047)

### 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)

### フェイルオーバー VF を使用した VM のコピー後のライブ移行が機能しない

現在、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、実行中の仮想マシン (VM) のコピー後移行の試行は失敗します。この問題を回避するには、コピー後の移行ではなく、標準の移行タイプを使用します。

(BZ#2054656)

### RHEL 8 の以前のマイナーバージョンから RHEL 8.6 Intel ホストへの仮想マシンのライブマイグレーションが機能しない

Intel Transactional Synchronization Extensions (TSX) 機能が廃止されたため、Intel ハードウェア上の RHEL 8.6 ホストは `hle` および `rtm` CPU フラグを使用しなくなりました。その結果、ソースホストが RHEL 8.5 またはそれ以前のマイナーバージョンの RHEL 8 を使用している場合、RHEL 8.6 ホストへの仮想マシン (VM) のライブマイグレーションは失敗します。

TSX の非推奨の詳細は、[Red Hat KnowledgeBase](#) を参照してください。

(BZ#2134184)

### 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ーションで 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** ユーティリティー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 パーミッション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、`--selinux-relabel` オプションを **virt-customize** コマンドに追加します。

(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である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](#))

## 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](#))

## AMD Milan で Windows Server 2022 ゲストの起動が非常に遅い場合がある

現在、Windows Server 2022 ゲストオペレーティングシステムと **qemu64** CPU モデルを使用する仮想マシン (VM) は、AMD EPYC 7003 シリーズプロセッサ (**AMD Milan** と呼ばれます) を搭載したホストでの起動に非常に長い時間がかかります。

この問題を回避するには、CPU モデルとして **qemu64** を使用しないでください。これは、RHEL 8 の VM ではサポートされていない設定であるためです。たとえば、代わりに **host-model** の CPU を使用します。

([BZ#2012373](#))

## 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](#))

## 仮想マシンを RHEL 8.6 以降の z-stream バージョンに移行すると失敗することがある

移行先ホストが **qemu-kvm** パッケージバージョン 6.2.0-11.module+el8.6.0+21121 以降の RHEL 8.6 バージョンを使用し、ソースホストが RHEL 8.6 で **qemu-kvm** バージョン 6.2.0-11.module+el8.6.0+21120 以前を使用している場合、仮想マシン (VM) の移行は失敗します。

([JIRA:RHELDPCS-17666](#))

## 10.17. クラウド環境の RHEL

SR-IOV は、Azure 上の ARM 64 RHEL 8 仮想マシンで準最適に動作します

現在、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なっており、レイテンシーは高くなっています。

(BZ#2068429)

### VMware ホストの RHEL 8 仮想マシンで静的 IP を設定できませんでした。

現在、VMware ホストで RHEL 8 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** ユーティリティーを使用して、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


```

%pre --interpreter /bin/bash --log /tmp/ks_pre.log

# Dump whole SCSI/IDE disks out sorted from smallest to largest ouputting
# just the name
disks=(`lsblk -n -o NAME -l -b -x SIZE -d -l 8,3`) || exit 1

# We are assuming we have 3 disks which will be used
# and we will create some variables to represent
d0=${disks[0]}
d1=${disks[1]}
d2=${disks[2]}

echo "part /home --fstype="xfs" --ondisk=$d2 --grow" >> /tmp/disks
echo "part swap --fstype="swap" --ondisk=$d0 --size=4096" >> /tmp/disks
echo "part / --fstype="xfs" --ondisk=$d1 --grow" >> /tmp/disks
echo "part /boot --fstype="xfs" --ondisk=$d1 --size=1024" >> /tmp/disks

%end

```


```

(BZ#1906870)

**cloud-init** を使用して AWS で RHEL 8 仮想マシンを起動すると、予想よりも時間がかかる

現在、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)

## 10.18. サポート性

### **getattachment** コマンドが複数の添付ファイルのダウンロードに失敗する

**getattachment** コマンドは、単一の添付ファイルのみをダウンロードできますが、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** 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** ユーティリティーは、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)



## 10.19. 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)

### 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ない

現在、RHEL Beta のコンテナイメージをプルしようとする、`podman` が **Error: Source image rejected: None of the signatures were accepted** のエラーメッセージと共に終了します。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ます。

回避策としては、Red Hat Beta GPG キーがローカルシステムに保存されていることを確認し、`podman image trust set` コマンドで適切な beta 名前空間の既存の信頼範囲を更新します。

ベータ版の GPG キーがローカルに保存されていない場合は、以下のコマンドを実行することで、そのキーをプルすることができます。

```
sudo wget -O /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-beta
https://www.redhat.com/security/data/f21541eb.txt
```

Beta GPG キーを信頼済みとして名前空間に追加するには、次のいずれかのコマンドを使用します。

```
$ sudo podman image trust set -f /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-beta
registry.access.redhat.com/namespace
```

および

```
$ sudo podman image trust set -f /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-beta
registry.redhat.io/namespace
```

`namespace` を `ubi9-beta` または `rhel9-beta` に置き換えてください。

(BZ#2020301)

## 第11章 国際化

### 11.1. 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

### 11.2. 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#2033398</a> , <a href="#">BZ#2016014</a> , <a href="#">BZ#1817505</a> , <a href="#">BZ#1780842</a>
<b>NetworkManager</b>	<a href="#">BZ#1996617</a> , <a href="#">BZ#2001563</a> , <a href="#">BZ#2079849</a> , <a href="#">BZ#1920398</a>
<b>SLOF</b>	<a href="#">BZ#1910848</a>
<b>accel-config</b>	<a href="#">BZ#1843266</a>
<b>anaconda</b>	<a href="#">BZ#1990145</a> , <a href="#">BZ#2050140</a> , <a href="#">BZ#1914955</a> , <a href="#">BZ#1929105</a>
<b>ansible-collection-microsoft-sql</b>	<a href="#">BZ#2038256</a> , <a href="#">BZ#2057651</a>
<b>apr</b>	<a href="#">BZ#1819607</a>
<b>audit</b>	<a href="#">BZ#1906065</a> , <a href="#">BZ#1939406</a> , <a href="#">BZ#1921658</a> , <a href="#">BZ#1927884</a>
<b>authselect</b>	<a href="#">BZ#1892761</a>
<b>bind9.16</b>	<a href="#">BZ#1873486</a>
<b>bind</b>	<a href="#">BZ#2013993</a>
<b>brltty</b>	<a href="#">BZ#2008197</a>
<b>certmonger</b>	<a href="#">BZ#1577570</a>
<b>clevis</b>	<a href="#">BZ#1949289</a> , <a href="#">BZ#2018292</a>
<b>cloud-init</b>	<a href="#">BZ#2023940</a> , <a href="#">BZ#2026587</a> , <a href="#">BZ#1750862</a>
<b>cockpit</b>	<a href="#">BZ#1666722</a>
<b>coreutils</b>	<a href="#">BZ#2030661</a>
<b>corosync-qdevice</b>	<a href="#">BZ#1784200</a>
<b>crash</b>	<a href="#">BZ#1906482</a>
<b>createrepo_c</b>	<a href="#">BZ#1992209</a> , <a href="#">BZ#1973588</a>

コンポーネント	チケット
<b>crypto-policies</b>	<a href="#">BZ#2020295</a> , <a href="#">BZ#2023734</a> , <a href="#">BZ#2023744</a> , <a href="#">BZ#1919155</a> , <a href="#">BZ#1660839</a>
<b>cups-container</b>	<a href="#">BZ#1913715</a>
<b>cups</b>	<a href="#">BZ#2032965</a>
<b>device-mapper-multipath</b>	<a href="#">BZ#2008101</a> , <a href="#">BZ#2009624</a> , <a href="#">BZ#2011699</a>
<b>distribution</b>	<a href="#">BZ#1657927</a>
<b>dmidecode</b>	<a href="#">BZ#2027665</a>
<b>dnf-plugins-core</b>	<a href="#">BZ#1868047</a>
<b>dnf</b>	<a href="#">BZ#1986657</a>
<b>ec2-images</b>	<a href="#">BZ#1862930</a>
<b>edk2</b>	<a href="#">BZ#1741615</a> , <a href="#">BZ#1935497</a>
<b>fapolicyd</b>	<a href="#">BZ#1939379</a> , <a href="#">BZ#2054741</a>
<b>fence-agents</b>	<a href="#">BZ#1977588</a> , <a href="#">BZ#1775847</a>
<b>fido-device-onboard</b>	<a href="#">BZ#1989930</a>
<b>firewalld</b>	<a href="#">BZ#1980206</a> , <a href="#">BZ#1871860</a>
<b>freeradius</b>	<a href="#">BZ#2030173</a> , <a href="#">BZ#1958979</a>
<b>galera</b>	<a href="#">BZ#2042306</a>
<b>gcc</b>	<a href="#">BZ#1996862</a>
<b>gdb</b>	<a href="#">BZ#2012818</a> , <a href="#">BZ#1853140</a>
<b>glibc</b>	<a href="#">BZ#1934162</a> , <a href="#">BZ#2007327</a> , <a href="#">BZ#2023420</a> , <a href="#">BZ#1929928</a> , <a href="#">BZ#2000374</a>
<b>gnome-shell-extensions</b>	<a href="#">BZ#1751336</a> , <a href="#">BZ#1717947</a>
<b>gnome-software</b>	<a href="#">BZ#1668760</a>
<b>gnutls</b>	<a href="#">BZ#1628553</a>
<b>golang</b>	<a href="#">BZ#2014088</a>

## コンポーネント

## チケット

<b>grafana-pcp</b>	<a href="#">BZ#1993149</a>
<b>grafana</b>	<a href="#">BZ#1993214</a>
<b>grub2</b>	<a href="#">BZ#1583445</a>
<b>hostapd</b>	<a href="#">BZ#2016946</a>
<b>initscripts</b>	<a href="#">BZ#1875485</a>
<b>ipa</b>	<a href="#">BZ#1731484</a> , <a href="#">BZ#1924707</a> , <a href="#">BZ#1664719</a> , <a href="#">BZ#1664718</a>
<b>js-d3-flame-graph</b>	<a href="#">BZ#1993194</a>
<b>kdump-anaconda-addon</b>	<a href="#">BZ#2086100</a>
<b>kernel</b>	<a href="#">BZ#1953926</a> , <a href="#">BZ#2068429</a> , <a href="#">BZ#1910885</a> , <a href="#">BZ#2040171</a> , <a href="#">BZ#2022903</a> , <a href="#">BZ#2036863</a> , <a href="#">BZ#1979382</a> , <a href="#">BZ#1949614</a> , <a href="#">BZ#1983635</a> , <a href="#">BZ#1964761</a> , <a href="#">BZ#2069047</a> , <a href="#">BZ#2054656</a> , <a href="#">BZ#1868526</a> , <a href="#">BZ#1694705</a> , <a href="#">BZ#1730502</a> , <a href="#">BZ#1609288</a> , <a href="#">BZ#1602962</a> , <a href="#">BZ#1865745</a> , <a href="#">BZ#1906870</a> , <a href="#">BZ#1924016</a> , <a href="#">BZ#1942888</a> , <a href="#">BZ#1812577</a> , <a href="#">BZ#1910358</a> , <a href="#">BZ#1930576</a> , <a href="#">BZ#2046396</a> , <a href="#">BZ#1793389</a> , <a href="#">BZ#1654962</a> , <a href="#">BZ#1940674</a> , <a href="#">BZ#1971506</a> , <a href="#">BZ#2022359</a> , <a href="#">BZ#2059262</a> , <a href="#">BZ#1605216</a> , <a href="#">BZ#1519039</a> , <a href="#">BZ#1627455</a> , <a href="#">BZ#1501618</a> , <a href="#">BZ#1633143</a> , <a href="#">BZ#1814836</a> , <a href="#">BZ#1696451</a> , <a href="#">BZ#1348508</a> , <a href="#">BZ#1837187</a> , <a href="#">BZ#1904496</a> , <a href="#">BZ#1660337</a> , <a href="#">BZ#1905243</a> , <a href="#">BZ#1878207</a> , <a href="#">BZ#1665295</a> , <a href="#">BZ#1871863</a> , <a href="#">BZ#1569610</a> , <a href="#">BZ#1794513</a>
<b>kexec-tools</b>	<a href="#">BZ#2004000</a>
<b>krb5</b>	<a href="#">BZ#1877991</a>
<b>libcap</b>	<a href="#">BZ#1950187</a> , <a href="#">BZ#2032813</a>
<b>libffi</b>	<a href="#">BZ#1875340</a>
<b>libgnome-keyring</b>	<a href="#">BZ#1607766</a>
<b>libguestfs</b>	<a href="#">BZ#1554735</a>
<b>libreswan</b>	<a href="#">BZ#2017352</a> , <a href="#">BZ#1989050</a>
<b>libseccomp</b>	<a href="#">BZ#2019893</a>

コンポーネント	チケット
libselinux-python-2.8-module	BZ#1666328
libssh	<a href="#">BZ#1896651</a>
libvirt	BZ#2014369, BZ#1664592, <a href="#">BZ#1332758</a> , BZ#1528684
llvm-toolset	BZ#2001133
log4j-2-module	BZ#1937468
lsvpd	BZ#1993557
lvm2	BZ#1496229, <a href="#">BZ#1768536</a>
make	<a href="#">BZ#2004246</a>
mariadb	<a href="#">BZ#1944653</a> 、 <a href="#">BZ#1942330</a>
mesa	<a href="#">BZ#1886147</a>
net-snmp	<a href="#">BZ#1908331</a>
nfs-utils	BZ#1592011
nftables	<a href="#">BZ#2047821</a>
nginx-1.20-module	<a href="#">BZ#1991787</a>
nispor	BZ#1848817
nmstate	<a href="#">BZ#2003976</a> 、 <a href="#">BZ#2004006</a>
nss_nis	<a href="#">BZ#1803161</a>
nss	<a href="#">BZ#1817533</a> , BZ#1645153
opencryptoki	BZ#1984993
opencv	<a href="#">BZ#2007780</a> , <a href="#">BZ#1886310</a>
openmpi	BZ#1866402
opensc	<a href="#">BZ#1947025</a>

コンポーネント	チケット
<b>openscap</b>	<a href="#">BZ#1970529</a> , <a href="#">BZ#2041781</a>
<b>openssh</b>	<a href="#">BZ#1926103</a> , <a href="#">BZ#2015828</a> , <a href="#">BZ#2044354</a>
<b>openssl</b>	<a href="#">BZ#1810911</a>
<b>osbuild-composer</b>	<a href="#">BZ#1951936</a> , <a href="#">BZ#2056451</a>
<b>oscap-anaconda-addon</b>	<a href="#">BZ#1834716</a> , <a href="#">BZ#2075508</a> , <a href="#">BZ#1843932</a> , <a href="#">BZ#1665082</a>
<b>pacemaker</b>	<a href="#">BZ#1082146</a> , <a href="#">BZ#1470834</a> , <a href="#">BZ#1376538</a>
<b>pcp</b>	<a href="#">BZ#1991763</a> , <a href="#">BZ#1629455</a>
<b>pcs</b>	<a href="#">BZ#1990784</a> , <a href="#">BZ#1936833</a> , <a href="#">BZ#1619620</a> , <a href="#">BZ#1847102</a> , <a href="#">BZ#1851335</a>
<b>pcsc-lite</b>	<a href="#">BZ#1928154</a> , <a href="#">BZ#2014641</a>
<b>perl</b>	<a href="#">BZ#2021471</a>
<b>php</b>	<a href="#">BZ#1978356</a>
<b>pki-core</b>	<a href="#">BZ#1729215</a> , <a href="#">BZ#1628987</a>
<b>pmdk-1_fileformat_v6-module</b>	<a href="#">BZ#2009889</a>
<b>podman</b>	<a href="#">JIRA:RHELPLAN-92741</a> , <a href="#">JIRA:RHELPLAN-108830</a> , <a href="#">JIRA:RHELPLAN-77238</a>
<b>policycoreutils</b>	<a href="#">BZ#1731501</a>
<b>postfix</b>	<a href="#">BZ#1711885</a>
<b>powerpc-utils</b>	<a href="#">BZ#2028690</a> , <a href="#">BZ#2022225</a>
<b>pykickstart</b>	<a href="#">BZ#1637872</a>
<b>qemu-kvm</b>	<a href="#">BZ#1982993</a> , <a href="#">BZ#2004416</a> , <a href="#">BZ#1662007</a> , <a href="#">BZ#2020133</a> , <a href="#">BZ#2012373</a> , <a href="#">BZ#1740002</a> , <a href="#">BZ#1719687</a> , <a href="#">BZ#1651994</a>
<b>rear</b>	<a href="#">BZ#2048454</a> , <a href="#">BZ#2049091</a> , <a href="#">BZ#2035939</a> , <a href="#">BZ#1868421</a> , <a href="#">BZ#2083301</a>
<b>redhat-support-tool</b>	<a href="#">BZ#2018194</a> , <a href="#">BZ#2018195</a> , <a href="#">BZ#1767195</a> , <a href="#">BZ#2064575</a> , <a href="#">BZ#1802026</a>

コンポーネント	チケット
<b>restore</b>	BZ#1997366
<b>rhel-system-roles</b>	BZ#1967321, BZ#2040038, BZ#2041627, BZ#2034908, BZ#1979714, BZ#2005727, BZ#2006231, BZ#2021678, BZ#2021683, BZ#2047504, BZ#2040812, BZ#2064388, BZ#2058655, BZ#2058772, BZ#2029605, BZ#2057172, BZ#2049747, BZ#1854988, BZ#1893743, BZ#1993379, BZ#1993311, BZ#2021661, BZ#2016514, BZ#1985022, BZ#2016511, BZ#2010327, BZ#2012316, BZ#2031521, BZ#2054364, BZ#2054363, BZ#2008931, BZ#1695634, BZ#1897565, BZ#2054365, BZ#1932678, BZ#2057656, BZ#2022458, BZ#2057645, BZ#2057661, BZ#2021685, BZ#2006081
<b>rig</b>	BZ#1888705
<b>rpm-ostree</b>	BZ#2032594
<b>rpm</b>	BZ#1940895, BZ#1688849
<b>rsyslog</b>	BZ#1947907, BZ#1679512, JIRA:RHELPLAN-10431
<b>rteval</b>	BZ#2012285
<b>rust-toolset</b>	BZ#2002883
<b>samba</b>	BZ#2013596, BZ#2009213, JIRA:RHELPLAN-13195, Jira:RHELDOCS-16612
<b>scap-security-guide</b>	BZ#1983061、 BZ#2053587、 BZ#2023569、 BZ#1990736、 BZ#2002850、 BZ#2000264、 BZ#2058033、 BZ#2030966、 BZ#1884687、 BZ#1993826、 BZ#1956972、 BZ#2014485、 BZ#2021802、 BZ#2028428、 BZ#1858866、 BZ#1750755、 BZ#2038977
<b>scap-workbench</b>	BZ#2051890
<b>selinux-policy</b>	BZ#1860443, BZ#1461914
<b>sos</b>	BZ#1873185, BZ#2011413
<b>spice</b>	BZ#1849563
<b>sssd</b>	BZ#2015070, BZ#1947671
<b>strace</b>	BZ#2038992
<b>subscription-manager</b>	BZ#2000883, BZ#2049441

コンポーネント	チケット
<b>texinfo</b>	<a href="#">BZ#2022201</a>
<b>udica</b>	<a href="#">BZ#1763210</a>
<b>usbguard</b>	<a href="#">BZ#2000000</a> , <a href="#">BZ#1963271</a>
<b>vdo</b>	<a href="#">BZ#1949163</a>
<b>virt-manager</b>	<a href="#">BZ#1995125</a> , <a href="#">BZ#2026985</a>
<b>wayland</b>	<a href="#">BZ#1673073</a>
<b>xfsdump</b>	<a href="#">BZ#2020494</a>
<b>xorg-x11-server</b>	<a href="#">BZ#1698565</a>
その他	<a href="#">BZ#1839151</a> , <a href="#">BZ#1780124</a> , <a href="#">BZ#2089409</a> , <a href="#">JIRA:RHELPLAN-100359</a> , <a href="#">JIRA:RHELPLAN-103147</a> , <a href="#">JIRA:RHELPLAN-103146</a> , <a href="#">JIRA:RHELPLAN-79161</a> , <a href="#">BZ#2046325</a> , <a href="#">JIRA:RHELPLAN-108438</a> , <a href="#">JIRA:RHELPLAN-100175</a> , <a href="#">BZ#2083036</a> , <a href="#">JIRA:RHELPLAN-102505</a> , <a href="#">BZ#2062117</a> , <a href="#">JIRA:RHELPLAN-75169</a> , <a href="#">JIRA:RHELPLAN-100174</a> , <a href="#">JIRA:RHELPLAN-101137</a> , <a href="#">JIRA:RHELPLAN-57941</a> , <a href="#">JIRA:RHELPLAN-101133</a> , <a href="#">JIRA:RHELPLAN-101138</a> , <a href="#">JIRA:RHELPLAN-95126</a> , <a href="#">JIRA:RHELPLAN-103855</a> , <a href="#">JIRA:RHELPLAN-103579</a> , <a href="#">BZ#2025814</a> , <a href="#">BZ#2077770</a> , <a href="#">BZ#1777138</a> , <a href="#">BZ#1640697</a> , <a href="#">BZ#1697896</a> , <a href="#">BZ#1971061</a> , <a href="#">BZ#1959020</a> , <a href="#">BZ#1961722</a> , <a href="#">BZ#1659609</a> , <a href="#">BZ#1687900</a> , <a href="#">BZ#1757877</a> , <a href="#">BZ#1741436</a> , <a href="#">JIRA:RHELPLAN-59111</a> , <a href="#">JIRA:RHELPLAN-27987</a> , <a href="#">JIRA:RHELPLAN-34199</a> , <a href="#">JIRA:RHELPLAN-57914</a> , <a href="#">JIRA:RHELPLAN-96940</a> , <a href="#">BZ#1974622</a> , <a href="#">BZ#2020301</a> , <a href="#">BZ#2028361</a> , <a href="#">BZ#2041997</a> , <a href="#">BZ#2035158</a> , <a href="#">JIRA:RHELPLAN-109067</a> , <a href="#">JIRA:RHELPLAN-115603</a> , <a href="#">BZ#1690207</a> , <a href="#">JIRA:RHELPLAN-1212</a> , <a href="#">BZ#1559616</a> , <a href="#">BZ#1889737</a> , <a href="#">JIRA:RHELPLAN-14047</a> , <a href="#">BZ#1769727</a> , <a href="#">JIRA:RHELPLAN-27394</a> , <a href="#">JIRA:RHELPLAN-27737</a> , <a href="#">BZ#1906489</a> , <a href="#">JIRA:RHELPLAN-100039</a> , <a href="#">BZ#1642765</a> , <a href="#">JIRA:RHELPLAN-10304</a> , <a href="#">BZ#1646541</a> , <a href="#">BZ#1647725</a> , <a href="#">BZ#1932222</a> , <a href="#">BZ#1686057</a> , <a href="#">BZ#1748980</a> , <a href="#">JIRA:RHELPLAN-71200</a> , <a href="#">BZ#1827628</a> , <a href="#">JIRA:RHELPLAN-45858</a> , <a href="#">BZ#1871025</a> , <a href="#">BZ#1871953</a> , <a href="#">BZ#1874892</a> , <a href="#">BZ#1916296</a> , <a href="#">JIRA:RHELPLAN-100400</a> , <a href="#">BZ#1926114</a> , <a href="#">BZ#1904251</a> , <a href="#">BZ#2011208</a> , <a href="#">JIRA:RHELPLAN-59825</a> , <a href="#">BZ#1920624</a> , <a href="#">JIRA:RHELPLAN-70700</a> , <a href="#">BZ#1929173</a> , <a href="#">JIRA:RHELPLAN-85066</a> , <a href="#">BZ#2006665</a> , <a href="#">JIRA:RHELPLAN-98983</a> , <a href="#">BZ#2009113</a> , <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> , <a href="#">BZ#2055826</a> , <a href="#">BZ#2059626</a>



## 付録B 改訂履歴

### 0.3-3

2024年5月16日(木) Brian Angelica ([bangelic@redhat.com](mailto:bangelic@redhat.com))

- [JIRA:RHELDOCS-17666](#) の既知の問題を更新(仮想化)。

### 0.3-2

2024年5月9日(木) 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。

### 0.3-1

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。

### 0.2-9

2024年2月29日(木)、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 非推奨の機能 [JIRA:RHELDOCS-17641](#) (Networking) を追加。

### 0.2-8

2024年2月13日火曜日、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。

### 0.2-7

2023年11月10日(金) 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。

### 0.2-6

2023年10月13日金曜日、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。

### 0.2-5

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。

### 0.2-4

2023年9月5日(火)、Jaroslav Klech ([jklech@redhat.com](mailto:jklech@redhat.com))

- 既知の問題 [BZ#2169382](#) (ネットワーク) の順序付きリストを修正しました。

### 0.2-3

2023年8月24日(木) Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 既知の問題 [BZ#2014500](#) (セキュリティ) を追加しました。

- 既知の問題 [BZ#2214508](#) (カーネル) を追加しました。

## 0.2-2

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。

## 0.2-1

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

## 0.2-0

2023 年 8 月 1 日 火曜日 Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 非推奨の機能 [JIRA:RHELPLAN-147538](#) (Web コンソール) を追加しました。

## 0.1-9

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。

## 0.1-8

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。

## 0.1-7

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。

## 0.1-6

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

## 0.1-5

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。

## 0.1-4

2023 年 3 月 2 日 (木)、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 新機能 [BZ#2089409](#) (カーネル) を更新しました。

## 0.1-4

2023 年 1 月 24 日 火曜日、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。

**0.1-3**

2022年12月8日木曜日、Marc Muehlfeld ([mmuehlfeld@redhat.com](mailto:mmuehlfeld@redhat.com))

- 既知の問題 [BZ#2132754](#) (ネットワーキング) を追加。

**0.1-2**

2022年11月8日金曜日、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 新機能 [JIRA:RHELPLAN-137623](#) および [BZ#2130912](#) (コンテナ) を追加しました。
- テクノロジーレビュー [JIRA:RHELPLAN-137622](#) (コンテナ) を追加しました。
- 既知の問題 [BZ#2134184](#) (仮想化) を追加しました。

**0.1-1**

2022年9月7日水曜日、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- バグ修正 [BZ#2096256](#) (ネットワーキング) を追加しました。
- その他の若干の更新

**0.1-0**

Fri Aug 19 2022, Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- バグ修正 [BZ#2108316](#) (ID 管理) を追加しました。

**0.0-9**

2022年8月5日金曜日、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- 既知の問題 [BZ#2114981](#) (セキュリティー) を追加。

**0.0-8**

2022年8月3日(水) Lenka Špačková ([lspackova@redhat.com](mailto:lspackova@redhat.com))

- 既知の問題 [BZ#2095764](#) (ソフトウェア管理) を追加しました。

**0.0-7**

2022年7月22日金曜日、Lucie Vařáková ([lvarakova@redhat.com](mailto:lvarakova@redhat.com))

- バグ修正 [BZ#2020494](#) (ファイルシステムとストレージ) を追加しました。
- 既知の問題 [BZ#2054656](#) (仮想化) を追加。
- その他の若干の更新

**0.0-6**

2022年7月11日(月) Lenka Špačková ([lspackova@redhat.com](mailto:lspackova@redhat.com))

- バグ修正 [BZ#2056451](#) (インストーラーとイメージの作成) を追加しました。
- バグ修正 [BZ#2051890](#) (セキュリティー) を追加しました。

- その他の若干の更新

#### 0.0-5

2022 年 6 月 8 日、Lucie Vařáková ([Imanasko@redhat.com](mailto:Imanasko@redhat.com))

- 新機能 [BZ#2089409](#) (カーネル) を追加しました。

#### 0.0-4

2022 年 5 月 31 日 Lucie Vařáková ([Imanasko@redhat.com](mailto:Imanasko@redhat.com))

- 既知の問題 [BZ#2075508](#) (セキュリティ) および [BZ#2077770](#) (仮想化) を追加しました。
- テクノロジーレビュー [BZ#1989930](#) (RHEL for Edge) および [JIRA:RHELPLAN-108438](#) (Web コンソール) を追加しました。
- RHEL 8 から RHEL 9 へのインプレースアップグレードに関する情報を、[インプレースアップグレードおよび OS 変換](#) のセクションに追加しました。
- その他の若干の更新

#### 0.0-3

2022 年 5 月 18 日 Lucie Maňásková ([Imanasko@redhat.com](mailto:Imanasko@redhat.com))

- 新機能 [BZ#2049441](#) (Web コンソール) を追加しました。
- 既知の問題 [BZ#2086100](#) (カーネル) および [BZ#2020133](#) (仮想化) を追加しました。
- その他の小規模の更新。

#### 0.0-2

2022 年 5 月 16 日 Lucie Maňásková ([Imanasko@redhat.com](mailto:Imanasko@redhat.com))

- バグ修正 [BZ#2014369](#) (仮想化) を追加しました。
- 既知の問題 [BZ#1554735](#) (仮想化) を追加。
- その他の小規模の更新。

#### 0.0-1

May 11 2022, Lucie Maňásková ([Imanasko@redhat.com](mailto:Imanasko@redhat.com))

- Red Hat Enterprise Linux 8.6 リリースノートのリリース。

#### 0.0-0

2022 年 3 月 30 日 Lucie Maňásková ([Imanasko@redhat.com](mailto:Imanasko@redhat.com))

- Red Hat Enterprise Linux 8.6 Beta リリースノートのリリース。