Red Hat Enterprise Linux 8

8.10 リリースノート

Red Hat Enterprise Linux 8.10 リリースノート

Last Updated: 2025-08-08

Red Hat Enterprise Linux 8 8.10 リリースノート

Red Hat Enterprise Linux 8.10 リリースノート

Legal Notice

Abstract

このリリースノートでは、Red Hat Enterprise Linux 8.10 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、テクニカルプレビュー、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。

Table of Contents

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 8.10 における主な変更点

インストーラーおよびイメージの作成
RHEL Image Builder の主なハイライト:

- **auto-lvm**、**lvm**、**raw** などのさまざまなパーティション設定モードを作成できます。
- ルールを追加および削除するための selected オプションと unselected オプションを使用して、プロファイルの調整オプションをカスタマイズし、ブループリントのカスタマイズにオプションを追加できます。

詳細は、新機能 - インストーラーとイメージの作成 を参照してください。

セキュリティー
SCAP セキュリティーガイド 0.1.72 には、更新された CIS プロファイル、PCI DSS ポリシーバージョン 4.0 に準拠したプロファイル、および最新の DISA STIG ポリシーのプロファイルが含まれています。

Linux カーネル暗号化 API (**libkcapi**) 1.4.0 では、新しいツールとオプションが導入されます。特に、新しい **-T** オプションを使用すると、ハッシュ値の計算でターゲットファイル名を指定できます。

stunnel TLS/SSL トンネリングサービス 5.71 では、FIPS モードにある OpenSSL 1.1 以降のバージョンの動作が変更されます。この変更に加えて、バージョン 5.71 では、最新の PostgreSQL クライアントのサポートなど、多くの新機能が提供されます。

OpenSSL TLS ツールキットには、RSA PKCS #1 v1.5 復号化プロセスに対する Bleichenbacher のような攻撃に対する API レベルの保護が含まれるようになりました。

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
次の Application Streams の新しいバージョンが利用可能になる

- **Python 3.12**
- **Ruby 3.3**
- **PHP 8.2**
- **nginx 1.24**
- **MariaDB 10.11**
- **PostgreSQL 16**

以下のコンポーネントがアップグレードされました。

- **Git** がバージョン 2.43.0 へ
- **Git LFS** がバージョン 3.4.1 へ

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。

Identity Management

RHEL 8.10 の Identity Management (IdM) では、OAuth 2 デバイス認証付与フローをサポートする外部アイデンティティープロバイダー (IdP) へのユーザー認証の委任が導入されています。これは現在完全にサポートされている機能です。

外部 IdP で認証と認可を実行した後、IdM ユーザーはシングルサインオン機能を備えた Kerberos チケットを受け取ります。

詳細は、新機能 - Identity Management を参照してください。

コンテナー
主な変更点は、以下のとおりです。

- マルチアーキテクチャーコンテナーイメージを作成するための **podman farm build** コマンドが、テクノロジープレビューとして利用できます。
- Podman は、事前に定義された設定セットをロードするための **containers.conf** モジュールをサポートするようになりました。
- Container Tools パッケージが更新されました。
- Podman v4.9 RESTful API では、イメージをレジストリーにプルまたはプッシュするときに、進捗データが表示されるようになりました。
- SQLite は、Podman のデフォルトのデータベースバックエンドとして完全にサポートされるようになりました。
- **Containerfile** は、複数行の HereDoc 命令をサポートするようになりました。
- ネットワーク名としての **pasta** は非推奨になりました。
- BoltDB データベースバックエンドは非推奨になりました。
- **container-tools:4.0** モジュールは非推奨になりました。
- Container Network Interface (CNI) ネットワークスタックは非推奨となり、今後のリリースでは削除される予定です。

詳細は、新機能 - コンテナー を参照してください。

1.2. インプレースアップグレードおよび OS 移行

RHEL 7 から RHEL 8 へのインプレースアップグレード
現在、考えられるインプレースアップグレードパスは以下のとおりです。

- 64 ビット Intel、IBM POWER 8 (little endian)、IBM Z アーキテクチャーでの RHEL 7.9 から RHEL 8.8 および RHEL 8.10 へのアップグレード。
- 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 8.8 および RHEL 8.10 へのアップグレード。

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は、How to in-place upgrade SAP environments from RHEL 7 to RHEL 8 を参照してください。

Red Hat がインプレースアップグレードプロセスをサポートする方法は、インプレースアップグレードサポートポリシー を参照してください。

主な機能拡張は、次のとおりです。

- アップグレード後の **systemd** サービスの予想される状態を決定するための新しいロジックが実装されました。
- ローカルに保存された DNF リポジトリーをインプレースアップグレードに使用できるようになりました。
- プロキシーを使用してアップグレードできるように DNF を設定できるようになりました。
- HTTPS を使用してアクセスしたカスタム DNF リポジトリーでインプレースアップグレードを実行する際の問題が修正されました。
- **/etc/pki/tls/openssl.cnf** 設定ファイルが変更されている場合は、アップグレード後に問題が発生しないように、アップグレード中にそのファイルがターゲットのデフォルトの OpenSSL 設定ファイルに置き換えられるようになりました。詳細は、アップグレード前のレポートを参照してください。

RHEL 6 から RHEL 8 へのインプレースアップグレード
RHEL 6 から RHEL 8 へのインプレースアップグレードを直接実行することはできません。ただし、RHEL 6 から RHEL 7 へのインプレースアップグレードを実行してから、RHEL 8 への 2 回目のインプレースアップグレードを実行することはできます。詳細は、RHEL 6 から RHEL 7 へのアップグレード を参照してください。

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
Alma Linux 8、CentOS Linux 8、Oracle Linux 8、または Rocky 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 7 または Oracle Linux 7 を使用している場合は、オペレーティングシステムを RHEL に変換してから、RHEL 8 へのインプレースアップグレードを実行できます。

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

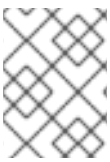
Red Hat Customer Portal Labs は、カスタマーポータルのセクションにあるツールセットで、https://access.redhat.com/labs/ から入手できます。Red Hat Customer Portal Labs のアプリケーションは、パフォーマンスの向上、問題の迅速なトラブルシューティング、セキュリティー問題の特定、複雑なアプリケーションの迅速なデプロイメントおよび設定に役立ちます。最も一般的なアプリケーションには、以下のものがあります。

- 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
- Red Hat Out of Memory Analyzer

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。



注記

リリースノートには、元の追跡チケットにアクセスするためのリンクが含まれています。プライベートチケットにはリンクがなく、代わりにこの脚注が表示されます。[1]

[1] リリースノートには、元の追跡チケットにアクセスするためのリンクが含まれています。プライベートチケットにはリンクがなく、代わりにこの脚注が表示されます。

第2章 アーキテクチャー

Red Hat Enterprise Linux 8.10 には、カーネルバージョン 4.18.0-553 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 リポジトリーおよび AppStream リポジトリーが含まれ、リポジトリーを追加しなくてもインストールを完了できる完全インストールイメージです。

注記

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

- Boot ISO - インストールプログラムを起動するのに使用する最小限の ISO ブートイメージです。このオプションでは、ソフトウェアパッケージをインストールするのに、BaseOS リポジトリーおよび AppStream リポジトリーにアクセスする必要があります。リポジトリーは、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は、インストールメディアからの RHEL の対話型インストール ドキュメントを参照してください。自動キックスタートインストールやその他の高度なトピックについては、RHEL の自動インストール ドキュメントを参照してください。

3.2. リポジトリー

Red Hat Enterprise Linux 8 は、2 つのメインリポジトリー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ーも必要で、すべての RHEL サブスクリプションで利用できます。

BaseOS リポジトリーのコンテンツは、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リポジトリーのコンテンツには、さまざまなワークロードとユースケースに対応するために、ユーザー空間アプリケーション、ランタイム言語、およびデータベースが含まれています。Application Streams は、**モジュール** と呼ばれる RPM 形式への拡張、または Software Collections として通常の RPM 形式で利用できます。AppStream で利用可能なパッケージのリストは、パッケージマニフェスト を参照してください。

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

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

3.3. アプリケーションストリーム

Red Hat Enterprise Linux 8 では、アプリケーションストリームの概念が導入されました。ユーザー空間コンポーネントのバージョンが複数配信され、オペレーティングシステムのコアパッケージよりも頻繁に更新されるようになりました。これにより、プラットフォームや特定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 をカスタマイズできる柔軟性が向上しました。

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

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

モジュールストリームは、アプリケーションストリームコンポーネントのバージョンを表します。たとえば、**postgresql:10** のデフォルトのストリーム以外に、**postgresql** モジュールでは、PostgreSQL データベースサーバーの複数のストリーム (バージョン) を利用できます。システムにインストールできるモジュールストリームは 1 つだけです。複数のコンテナー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 8.10 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

ブループリントファイルシステムのカスタマイズでパーティションモードを使用する機能

この更新により、RHEL イメージビルダーを使用しながら、選択したファイルシステムのカスタマイズでブループリントをカスタマイズできるようになります。イメージを作成するときに、次のいずれかのパーティションモードを選択できます。

- デフォルト: **auto-lvm**
- LVM: イメージは、追加のパーティションがなくても論理ボリュームマネージャー (LVM) を使用します。
- Raw: イメージは追加のパーティションがあっても raw パーティションを使用します

Jira:RHELDOCS-16337[1]

イメージビルダーでのファイルシステムカスタマイズポリシーの変更

ブループリントで RHEL イメージビルダーのファイルシステムのカスタマイズを使用する場合は、次のポリシー変更が適用されます。

現在、**mountpoint** と最小パーティション **minsize** を設定できます。次のイメージタイプは、ファイルシステムのカスタマイズ **image-installeredge-installeredge-simplified-installer** をサポートしていません。次のイメージタイプは、パーティションで分割されたオペレーティングシステムイメージを作成しません。ファイルシステム **edge-commitedge-containertarcontainer** をカスタマイズしても意味がありません。ブループリントは、**tpm** とそのサブディレクトリーの **mountpoint** のカスタマイズをサポートするようになりました。

Jira:RHELDOCS-17261[1]

4.2. セキュリティー

SCAP セキュリティーガイドが 0.1.72 にリベース

SCAP セキュリティーガイド (SSG) パッケージが、アップストリームバージョン 0.1.72 にリベースされました。このバージョンでは、バグ修正とさまざまな機能拡張が行われています。主なものは次のとおりです。

- CIS プロファイルは最新のベンチマークに合わせて更新されます。
- PCI DSS プロファイルは、PCI DSS ポリシーバージョン 4.0 に準拠します。
- STIG プロファイルは、最新の DISA STIG ポリシーに準拠します。

詳細は、SCAP Security Guide release notes を参照してください。

Jira:RHEL-25250[1]

OpenSSL に、Bleichenbacher のような攻撃に対する保護が含まれるようになりました。

このリリースの OpenSSL TLS ツールキットでは、RSA PKCS #1 v1.5 復号化プロセスにおける

Bleichenbacher のような攻撃に対する API レベルの保護が導入されています。PKCS #1 v1.5 復号化中にパディングをチェックする際にエラーを検出すると、RSA の復号化がエラーではなく、無作為に生成された確定的なメッセージを返すようになりました。この変更により、CVE-2020-25659 および CVE-2020-25657 などの脆弱性に対する一般的な保護が提供されます。

この保護を無効にするには、次を呼び出します: **EVP_PKEY_CTX_ctrl_str(ctx, "rsa_pkcs1_implicit_rejection"."0")** 関数。これは、RSA 復号化コンテキストで呼び出しますが、これによりシステムがより脆弱になります。

Jira:RHEL-17689[1]

librdkafka が 1.6.1 にリベースされました

Apache Kafka プロトコルの **librdkafka** 実装がアップストリームバージョン 1.6.1 にリベースされました。これは RHEL 8 の最初の主要な機能リリースです。リベースにより、多くの重要な機能拡張とバグ修正が提供されます。関連するすべての変更は、**librdkafka** パッケージに付属する **CHANGELOG.md** ドキュメントを参照してください。

注記

この更新により、設定のデフォルトが変更され、一部の設定プロパティーが非推奨になります。詳細は、**CHANGELOG.md** のアップグレードに関する考慮事項のセクションをお読みください。このバージョンの API (C および C++) および ABI © は、**librdkafka** の古いバージョンと互換性があります。ただし、設定プロパティーの一部を変更すると、既存のアプリケーションの変更が必要になる場合があります。

Jira:RHEL-12892[1]

libkcapi が 1.4.0 にリベース

Linux カーネル暗号化 API へのアクセスを提供する **libkcapi** ライブラリーは、アップストリームバージョン 1.4.0 にリベースされました。この更新には、さまざまな機能拡張とバグ修正が含まれています。主なものは次のとおりです。

- **sm3sum** および **sm3hmac** ツールを追加しました。
- **kcapi_md_sm3** および **kcapi_md_hmac_sm3** API を追加しました。
- SM4 の便利な機能を追加しました。
- リンク時最適化 (LTO) のサポートを修正しました。
- LTO リグレッションテストを修正しました。
- **kcapi-enc** による任意サイズの AEAD 暗号化のサポートを修正しました。

Jira:RHEL-5366[1]

stunnel が 5.71 にリベースされました

stunnel TLS/SSL トンネリングサービスが、アップストリームバージョン 5.71 にリベースされました。この更新により、FIPS モードでの OpenSSL 1.1 以降のバージョンの動作が変更されます。OpenSSL が FIPS モードであり、**stunnel** のデフォルトの FIPS 設定が **no** に設定されている場合、**stunnel** は OpenSSL に適応し、FIPS モードが有効になります。

追加の新機能は次のとおりです。

- 最新の PostgreSQL クライアントのサポートが追加されました。
- **protocolHeader** サービスレベルオプションを使用して、カスタム **connect** プロトコルネゴシエーションヘッダーを挿入できます。
- **protocolHost** オプションを使用して、クライアントの SMTP プロトコルネゴシエーションの HELO/EHLO 値を制御できます。
- クライアントサイドの **protocol = ldap** に関するクライアントサイドサポートが追加されました。
- サービスレベルの **sessionResume** オプションを使用して、セッション再開を設定できるようになりました。
- **CApath** を使用したサーバーモードでのクライアント証明書の要求に対するサポートが追加されました (以前は **CAfile** のみがサポートされていました)。
- ファイルの読み取りとロギングのパフォーマンスが向上しました。
- **retry** オプションの設定可能な遅延のサポートが追加されました。
- クライアントモードでは、**verifyChain** が設定されている場合に OCSP ステープリングの要求および検証が行われます。
- サーバーモードでは、OCSP ステープリングは常に利用可能です。
- 不確定の OCSP 検証により TLS ネゴシエーションが中断されます。これを無効にするには、**OCSPrequire = no** と設定します。

Jira:RHEL-2340[1]

OpenSSH は認証における人為的な遅延を制限します

ログイン失敗後の OpenSSH の応答は、ユーザー列挙攻撃を防ぐために人為的に遅延されます。この更新では、特権アクセス管理 (PAM) 処理などでリモート認証に時間がかかりすぎる場合に、このような人為的な遅延が過度に長くならないように上限が導入されています。

Jira:RHEL-1684

libkcapi が、ハッシュ値の計算でターゲットファイル名を指定するオプションを提供するようになりました

この **libkcapi** (Linux カーネル暗号化 API) パッケージの更新により、ハッシュ値の計算でターゲットファイル名を指定するための新しいオプション **-T** が導入されます。このオプションの値は、処理済みの HMAC ファイルで指定されたファイル名をオーバーライドします。このオプションは、**-c** オプションとの併用でのみ使用できます。以下に例を示します。

```
$ sha256hmac -c <hmac_file> -T <target_file>
```

Jira:RHEL-15300[1]

audit が 3.1.2 にリベース

Linux の Audit システムがバージョン 3.1.2 に更新されました。これにより、以前にリリースされたバージョン 3.0.7 に対するバグ修正、機能拡張、およびパフォーマンス向上が実現しました。主な機能拡張は、次のとおりです。

- **auparse** ライブラリーは、名前のないソケットと匿名ソケットを解釈するようになりました。
- **ausearch** および **aureport** ツールの **start** オプションと **end** オプションで、新しいキーワード **this-hour** を使用できます。
- シグナル用のユーザーフレンドリーなキーワードが、**auditctl** プログラムに追加されました。
- **auparse** での破損したログの処理が改善されました。
- **auditd** サービスで、**ProtectControlGroups** オプションがデフォルトで無効になりました。
- 除外フィルターのルールチェックが修正されました。
- **OPENAT2** フィールドの解釈が改善されました。
- **audispd af_unix** プラグインがスタンドアロンプログラムに移動しました。
- Python バインディングが変更され、Python API から Audit ルールが設定できなくなりました。この変更は、Simplified Wrapper and Interface Generator (SWIG) のバグのために行われました。

Jira:RHEL-15001[1]

4.3. シェルおよびコマンドラインツール

openCryptoki がバージョン 3.22.0 にリベース

opencryptoki パッケージがバージョン 3.22.0 に更新されました。主な変更点は、以下のとおりです。

- **CPACF** で保護されたキーを使用することで、**AES-XTS** キータイプのサポートを追加しました。
- 証明書オブジェクト管理のサポートが追加されました。
- **no-login** オプションによるパブリックセッションのサポートが追加されました。
- セキュリティーオフィサー (SO) としてログインするためのサポートが追加されました。
- **Edwards** キーと **Montgomery** キーのインポートおよびエクスポートのサポートが追加されました。
- **RSA-PSS** キーと証明書のインポートのサポートが追加されました。
- セキュリティー上の理由から、AES-XTS キーの 2 つのキー部分を同一にすることはできません。この更新では、同一になるのを確実に避けるために、キー生成およびインポートプロセスに対するチェックが追加されます。
- さまざまなバグ修正が実装されました。

Jira:RHEL-11413[1]

4.4. インフラストラクチャーサービス

chrony がバージョン 4.5 にリベース

chrony スイートがバージョン 4.5 に更新されました。主な変更点は、以下のとおりです。

- ホスト名で指定された Network Time Protocol (NTP) ソースの IP アドレスの定期的な更新を追加しました。デフォルトの間隔は 2 週間です。**chrony.conf** ファイルに **refresh 0** を追加することで無効にできます。
- 到達不能な NTP ソースの自動置換が改善されました。
- **chronyc** ユーティリティーによって行われる重要な変更のロギングが改善されました。
- ソース選択の失敗と falseticker のロギングが改善されました。
- 遅延ハードウェア送信タイムスタンプのタイムアウトを設定するための **hwtstimeout** ディレクティブを追加しました。
- ハードウェアタイムスタンプで PTP の精度に到達するために、Precision Time Protocol (PTP) 透過クロックが提供する修正に対する実験的なサポートが追加されました。
- **interleaved** モードの **presend** オプションを修正しました。
- IP アドレスで指定された変更済みソースの **sourcedir** ディレクトリーからのリロードを修正しました。

Jira:RHEL-21069

linuxptp がバージョン 4.2 にリベース

linuxptp プロトコルがバージョン 4.2 に更新されました。主な変更点は、以下のとおりです。

- **phc2sys** ユーティリティーに複数のドメインのサポートが追加されました。
- Precision Time Protocol (PTP) の親データセット (クロッククラスなど) におけるクロックの更新および変更に関する通知のサポートが追加されました。
- PTP Power Profile (IEEE C37.238-2011 および IEEE C37.238-2017) のサポートが追加されました。

Jira:RHEL-21326[1]

4.5. ネットワーク

firewalld が不要なファイアウォールルールのフラッシュを実行しなくなりました

firewalld サービスは、以下の条件を両方満たす場合、**iptables** 設定から既存のルールすべてを削除しません。

- **firewalld** が **nftables** バックエンドを使用している。
- **--direct** オプションを使用して作成されたファイアウォールルールが存在しない。

この変更は、不要な操作 (ファイアウォールルールのフラッシュ) を削減し、他のソフトウェアとの統合を改善することを目的としています。

Jira:RHEL-47595

ss ユーティリティーが TCP バインドされた非アクティブソケットの可視性を改善する

iproute2 スイートは、TCP/IP ネットワークトラフィックを制御するためのユーティリティーのコレクションを提供します。TCP バインドされた非アクティブソケットは、IP アドレスとポート番号に接続されていますが、両方とも TCP ポートに接続もリッスンもされていません。ソケットサービス (**ss**) ユーティリティーは、TCP バインドされた非アクティブソケットをダンプするためのカーネルのサポートを追加します。次のコマンドオプションを使用して、これらのソケットを表示できます。

- **ss --all**: TCP バインドされた非アクティブソケットを含むすべてのソケットをダンプします。
- **ss --bound-inactive**: バインドされた非アクティブソケットのみをダンプします。

Jira:RHEL-6113[1]

nispor がバージョン 1.2.10 にリベースされました。

nispor パッケージはアップストリームバージョン 1.2.10 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拡張がいくつか追加されました。

- 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した。
- Single Root Input and Output Virtualization (SR-IOV) インターフェイスで、SR-IOV Virtual Function (SR-IOV VF) 情報を (VF) ごとにクエリーできます。
- ボンディングオプション **lacp_active**、**arp_missed_max**、および **ns_ip6_target** が新しくサポートされました。

Bugzilla:2153166

4.6. カーネル

RHEL 8.10 のカーネルバージョン

Red Hat Enterprise Linux 8.10 は、カーネルバージョン 4.18.0-553 で配布されます。

rtla がアップストリーム kernel ソースコードのバージョン 6.6 にリベース

rtla ユーティリティーが最新のアップストリームバージョンにアップグレードされ、複数のバグ修正および機能拡張が追加されました。主な変更点は、以下のとおりです。

- メインの **rtla** スレッドとは別に、実行する **rtla** スレッドの追加コントロールグループを指定する **-C** オプションが追加されました。
- **rtla** スレッドをハウスキーピング CPU に配置し、測定スレッドを異なる CPU に配置する **--house-keeping** オプションが追加されました。
- **timerlat hist** および **timerlat top** スレッドをユーザー空間で実行できるように、**timerlat** トレーサーのサポートが追加されました。

Jira:RHEL-10081[1]

rteval はアップストリームバージョン 3.7 にアップグレードされました

この更新により、**rteval** ユーティリティーがアップストリームバージョン 3.7 にアップグレードされました。この更新で最も重要な機能は、**isolcpus** カーネルパラメーターに関するものです。これには、**rteval** 内の測定モジュールの **isolcpus** メカニズムを検出して使用する機能が含まれます。その結果、**isolcpus** ユーザーは **rteval** を使用して正確なレイテンシー数値を取得し、リアルタイムカーネルで測定された最良のレイテンシー結果を簡単に達成できるようになります。

Jira:RHEL-8967[1]

SGX が完全にサポートされるようになりました

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 バージョン 1 および 2 の機能を提供します。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で SGX テクノロジーを使用できるようになります。バージョン 2 では、**Enclave Dynamic Memory Management**(EDMM) が追加されています。主な変更には以下のものがあります。

- 一旦初期化されたエンクレーブに属する通常のエンクレーブページの EPCM 権限を変更します。
- 一旦初期化されたエンクレーブに通常のエンクレーブページを動的に追加します。
- より多くのスレッドを収容できるように一旦初期化されたエンクレーブを拡張します。
- 一旦初期化されたエンクレーブから通常のページと TCS ページを削除します。

このリリースでは、SGX はテクノロジープレビューから完全にサポートされる機能に移行します。

Bugzilla:2041881[1]

Intel データストリーミングアクセラレータードライバーが完全にサポートされるようになりました

Intel データストリーミングアクセラレータードライバー (IDXD) は、Intel CPU 統合アクセラレーターを提供するカーネルドライバーです。これには、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。

このリリースでは、IDXD はテクノロジープレビューから完全にサポートされる機能に移行します。

Jira:RHEL-10097[1]

rteval は、デフォルトの測定 CPU リストに対する任意の CPU の追加および削除をサポートするようになりました

rteval ユーティリティーを使用すると、**--measurement-cpulist** パラメーターの使用時に、新しいリスト全体を指定するのではなく、デフォルトの測定 CPU リストに CPU を追加 (+ 記号を使用) または削除 (- 記号を使用) することができます。さらに、分離されたすべての CPU のセットをデフォルトの測定 CPU リストに追加するために **--measurement-run-on-isolcpus** が導入されました。このオプションは、分離された CPU 上で実行されるリアルタイムアプリケーションの最も一般的なユースケースをカバーします。他のユースケースでは、より一般的な機能が必要になります。たとえば、一部のリアルタイムアプリケーションでは、ハウスキーピングのために分離 CPU を 1 つ使用していました。当該 CPU は、デフォルトの測定 CPU リストから除外する必要がありました。その結果、デフォルトの測定 CPU リストに任意の CPU を追加するだけでなく、柔軟な方法で任意の CPU を削除することもできるようになりました。削除は追加よりも優先されます。このルールは、+/- 記号を使用して指定した CPU と、**--measurement-run-on-isolcpus** で定義した CPU の両方に適用されます。

Jira:RHEL-21926[1]

4.7. ブートローダー

起動前段階での DEP/NX サポート

Data Execution Prevention (DEP)、No Execute (NX)、または Execute Disable (XD) と呼ばれるメモリー保護機能は、実行不可としてマークされたコードの実行をブロックします。DEP/NX は、RHEL のオペレーティングシステムレベルで利用可能になりました。

このリリースでは、GRUB および **shim** ブートローダーに DEP/NX サポートが追加されました。これにより、特定の脆弱性 (DEP/NX 保護がなければ特定の攻撃を実行する可能性のある、悪意のある EFI ドライバーなど) を起動前の段階で防ぐことができます。

Jira:RHEL-15856[1]

GRUB および shim での TD RTMR 測定のサポート

Intel® Trust Domain Extension (Intel® TDX) は、Trust Domains (TD) と呼ばれるハードウェア分離された仮想マシン (VM) を展開する機密コンピューティングテクノロジーです。

TDX は、TD VM ゲストを使用して、仮想マシン拡張機能 (VMX) 命令とマルチキー合計メモリー暗号化 (MKTME) 機能を拡張します。TD ゲスト仮想マシンでは、**grub2** や **shim** などのブートチェーン内のすべてのコンポーネントが、イベントと測定ハッシュをランタイム測定レジスター (RTMR) に記録する必要があります。

RTMR での TD ゲストランタイム測定は、認証アプリケーションの基盤となります。TD ゲスト上のアプリケーションは、証明サービスを通じて中継部分からキーなどの機密情報を取得するための信頼の証拠を提供するために TD 測定に依存します。

このリリースでは、GRUB および **shim** ブートローダーが TD 測定プロトコルをサポートするようになりました。

Intel® TDX の詳細は、Intel® Trust Domain Extensions のドキュメント を参照してください。

Jira:RHEL-15583[1]

4.8. ファイルシステムおよびストレージ

Storage RHEL システムロールが共有 LVM デバイス管理をサポートするようになりました

RHEL システムロールが、共有論理ボリュームとボリュームグループの作成と管理をサポートするようになりました。

Jira:RHEL-14022

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

以前は、**multipathd** コマンドは SCSI デバイス上の Integrity Fabric Performance Impact Notification (PFIN-Li) イベントのみを監視していました。**multipathd** は、ファイバーチャネルファブリックが送信するリンク整合性イベントをリッスンし、それを使用してパスをマージナルとしてマークすることができました。この機能は、SCSI デバイス上のマルチパスデバイスでのみサポートされていました。この機能の使用の制限により、**multipathd** は Non-volatile Memory Express (NVMe) デバイスパスをマージナルとしてマークすることができませんでした。

この更新により、**multipathd** は、SCSI デバイスと NVMe デバイスの両方で FPIN-Li イベントの検出を

サポートするようになりました。その結果、マルチパスは、他のパスが利用可能である間は、ファブリック接続が良好でないパスを使用しなくなりました。これにより、そのような状況での IO 遅延を回避できます。

Jira:RHEL-6677

4.9. 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

Python 3.12 が RHEL 8 で利用できるようになりました

RHEL 8.10 では、新しいパッケージ **python3.12** とそのためにビルドされた一連のパッケージ、および **ubi8/python-312** コンテナーイメージによって提供される Python 3.12 が導入されています。

以前にリリースされた Python 3.11 と比較しての主な機能拡張は次のとおりです。

- Python では、ジェネリッククラスと関数に対して新しい **type** ステートメントと新しい型パラメーター構文が導入されています。
- フォーマットされた文字列リテラル (f-strings) が文法で形式化され、パーサーに直接統合できるようになりました。
- Python が、インタープリターごとに固有のグローバルインタープリターロック (GIL) を提供するようになりました。
- Python コードからバッファープロトコルを使用できるようになりました。
- **CPython** のディクショナリー、リスト、セットの内包表記がインライン化されました。これにより、内包表記の実行速度が大幅に向上します。
- **CPython** が Linux **perf** プロファイラーをサポートするようになりました。
- **CPython** が、サポート対象のプラットフォームでスタックオーバーフローの保護を提供するようになりました。

python3.12 スタックからパッケージをインストールするには、たとえば、以下を使用します。

```
# yum install python3.12
# yum install python3.12-pip
```

インタープリターを実行するには、たとえば、以下を使用します。

```
$ python3.12
$ python3.12 -m pip --help
```

詳細は、Python のインストールおよび使用 を参照してください。

Python 3.12 のサポート期間は、Red Hat Enterprise Linux Application Streams ライフサイクル を参照してください。

Jira:RHEL-14942

メールアドレスの解析を制御するための Python の新しい環境変数

CVE-2023-27043 の問題を軽減するために、メールアドレスをより厳密に解析するための後方互換性のない変更が Python 3 に導入されました。

この更新では、新しい **PYTHON_EMAIL_DISABLE_STRICT_ADDR_PARSING** 環境変数が導入されました。この変数を **true** に設定すると、比較的厳密ではない以前の解析動作が、システム全体のデフォルトになります。

```
export PYTHON_EMAIL_DISABLE_STRICT_ADDR_PARSING=true
```

ただし、該当する関数を個別に呼び出すと、より厳密な動作が有効になる可能性があります。

次の内容で **/etc/python/email.cfg** 設定ファイルを作成しても、同じ結果を得ることができます。

```
[email_addr_parsing]
PYTHON_EMAIL_DISABLE_STRICT_ADDR_PARSING = true
```

詳細は、ナレッジベース記事 Mitigation of CVE-2023-27043 introducing stricter parsing of email addresses in Python を参照してください。

Jira:RHELDOCS-17369[1]

新しいモジュールストリーム: ruby:3.3

RHEL 8.10 では、新しい **ruby:3.3** モジュールストリームに Ruby 3.3.0 が導入されました。このバージョンでは、RHEL 8.7 で配布される **Ruby 3.1** に対するパフォーマンスの向上、バグおよびセキュリティー修正、および新機能がいくつか追加されました。

主な機能拡張は、次のとおりです。

- **Ripper** の代わりに新しい **Prism** パーサーを使用できます。**Prism** は、移植可能で、エラー耐性があり、メンテナンス性に優れた Ruby 言語の再帰下降パーサーです。
- Ruby の just-in-time (JIT) コンパイラー実装である YJIT は、もはや実験的機能ではなく、大幅なパフォーマンスの向上をもたらします。
- **Regexp** 一致アルゴリズムが改善され、潜在的な正規表現サービス拒否 (ReDoS) の脆弱性の影響が軽減されました。
- 新しい実験的な RJIT (純粋な Ruby の JIT) コンパイラーが MJIT を置き換えます。実稼働環境では YJIT を使用してください。
- 新しい M:N スレッドスケジューラーが利用可能になりました。

その他の主な変更点:

- 今後は、**Bison** の代わりに **Lrama** LALR パーサージェネレーターを使用する必要があります。
- いくつかの非推奨のメソッドと定数が削除されました。
- **Racc** gem はデフォルトの gem からバンドルされた gem に昇格しました。

ruby:3.3 モジュールストリームをインストールするには、次のコマンドを使用します。

```
# yum module install ruby:3.3
```

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

Ruby 3.3 のサポート期間は、Red Hat Enterprise Linux Application Streams ライフサイクル を参照してください。

Jira:RHEL-17090[1]

新しいモジュールストリーム: php:8.2

RHEL 8.10 では、PHP 8.2 が追加され、バージョン 8.0 に対するバグ修正と機能拡張がいくつか提供されています。

PHP 8.2 では、以下が可能です。

- 列挙 (Enums) 機能を使用して、可能な値の離散数の 1 つに制限されるカスタム型を定義します。
- 初期化後のプロパティー変更を防ぐために、**readonly** 修飾子を使用してプロパティーを宣言します。
- ファイバー、フルスタック、割り込み可能な関数を使用します。
- 読み取り専用クラスを使用します。
- いくつかの新しいスタンドアロン型を宣言します。
- 新しい **Random** 拡張機能を使用します。
- 特性の制約を定義します。

php:8.2 モジュールストリームをインストールするには、次のコマンドを使用します。

```
# yum module install php:8.2
```

以前の**php** ストリームからアップグレードする場合は、新しいストリームへの切り替え を参照してください。

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。

php モジュールストリームのサポート期間の詳細は、Red Hat Enterprise Linux Application Streams のライフサイクル を参照してください。

Jira:RHEL-14705[1]

perl-DateTime-TimeZone モジュールの name() メソッドは、タイムゾーン名を返すようになりました。

perl-DateTime-TimeZone モジュールがバージョン 2.62 に更新され、**name()** メソッドによって返される値がタイムゾーンエイリアスからメインのタイムゾーン名に変更されました。

詳細と例は、ナレッジベースの記事 Change in the perl-DateTime-TimeZone API related to time zone name and alias を参照してください。

Jira:RHEL-35685

新しいモジュールストリーム: nginx:1.24

nginx 1.24 Web およびプロキシーサーバーが、**nginx:1.24** 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。この更新では、以前にリリースされたバージョン 1.22 に対するバグ修正、セキュリティー修正、新機能、および機能拡張がいくつか提供されます。

Transport Layer Security (TLS) に関連する新機能と変更点:

- **ssl_session_cache** ディレクティブで共有メモリーを使用する場合、TLS セッションチケットの暗号鍵が自動的にローテーションされるようになりました。
- Secure Sockets Layer (SSL) プロキシーを使用した設定でメモリー使用量が最適化されました。
- **resolver** ディレクティブの **ipv4=off** パラメーターを使用して、解決時に IPv4 アドレスの検索を無効にできるようになりました。
- nginx は、PROXY v2 TLV プロトコルに表示される Type-Length-Value (TLV) フィールドの値を格納する **$proxy_protocol_tlv_*** 変数をサポートするようになりました。
- **ngx_http_gzip_static_module** モジュールはバイト範囲をサポートするようになりました。

その他の変更点:

- ヘッダー行は、内部 API でリンクされたリストとして表されるようになりました。
- nginx は、**ngx_http_perl_module** の **$r->header_in()** メソッドと、**$http_...**、**$sent_http_...**、**$sent_trailer_...**、**$upstream_http_...**、および **$upstream_trailer_...** 変数の検索中に、FastCGI、SCGI、および uwsgi バックエンドに渡される同一の名前のヘッダー文字列を連結するようになりました。
- nginx は、リスニングソケットのプロトコルパラメーターが再定義されると警告を表示するようになりました。
- nginx は、クライアントによってパイプラインが使用された場合、接続を遅延して閉じるようになりました。
- さまざまな SSL エラーのロギングレベルが、たとえば **Critical** から **Informational** に引き下げられました。

nginx:1.24 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 nginx:1.24
```

以前の **nginx** ストリームからアップグレードするには、新しいストリームに切り替えます。

詳細は、NGINX のセットアップと設定 を参照してください。

nginx モジュールストリームのサポート期間は、Red Hat Enterprise Linux アプリケーションストリームのライフサイクル を参照してください。

Jira:RHEL-14714[1]

新しいモジュールストリーム: mariadb:10.11

MariaDB 10.11 が、新しいモジュールストリーム **mariadb:10.11** として利用できるようになりました。以前に利用可能であったバージョン 10.5 に対する主な機能拡張は、以下のとおりです。

- 新しい **sys_schema** 機能。
- アトミックデータ定義言語 (DDL) ステートメント。
- 新しい **GRANT ... TO PUBLIC** 特権。
- **SUPER** 特権と **READ ONLY ADMIN** 特権の分離。
- 新しい **UUID** データベースデータ型。
- Secure Socket Layer (SSL) プロトコルのバージョン 3 のサポート。MariaDB サーバーの起動には、正しく設定された SSL が必要になりました。
- **natural_sort_key()** 関数による自然なソート順序のサポート。
- 任意のテキストフォーマットのための新しい **SFORMAT** 関数。
- UTF-8 文字セットと UCA-14 照合の変更。
- **/usr/share/** ディレクトリーで利用可能な **systemd** ソケットのアクティベーションファイル。アップストリームとは異なり、これらのファイルは RHEL のデフォルト設定の一部ではないことに注意してください。
- **MySQL** の代わりに **MariaDB** 文字列を含むエラーメッセージ。
- 中国語で利用可能なエラーメッセージ。
- デフォルトの logrotate ファイルへの変更。
- MariaDB および MySQL クライアントの場合、コマンドラインで指定した接続プロパティー (例: **--port=3306**) によって、クライアントとサーバー間の通信のプロトコルタイプ (**tcp**、**socket**、**pipe**、**memory** など) が強制されるようになりました。

MariaDB 10.11 の変更点の詳細は、MariaDB 10.5 と MariaDB 10.11 の主な違い を参照してください。

MariaDB の詳細は、MariaDB の使用 を参照してください。

mariadb:10.11 ストリームをインストールするには、以下を使用します。

```
# yum module install mariadb:10.11
```

mariadb:10.5 モジュールストリームからアップグレードする場合は、MariaDB 10.5 から MariaDB 10.11 へのアップグレード を参照してください。

mariadb モジュールストリームのサポート期間の詳細は、Red Hat Enterprise Linux Application Streams のライフサイクル を参照してください。

Jira:RHEL-3637

新しいモジュールストリーム: postgresql:16

RHEL 8.10 では、PostgreSQL 16 が導入され、バージョン 15 に対する新機能と機能拡張がいくつか提供されています。

主な機能拡張は、次のとおりです。

- 強化された一括ロードによりパフォーマンスが向上します。
- **libpq** ライブラリーが、接続レベルの負荷分散をサポートするようになりました。より効率的な負荷分散のために、新しい **load_balance_hosts** オプションを使用できます。
- カスタム設定ファイルを作成し、それを **pg_hba.conf** ファイルと **pg_ident.conf** ファイルに追加できるようになりました。
- PostgreSQL は、**pg_hba.conf** ファイル内のデータベースおよびロールエントリーに対する正規表現の一致をサポートするようになりました。

その他の変更点は次のとおりです。

- PostgreSQL は **postmaster** バイナリーとともに配布されなくなりました。提供されている **systemd** ユニットファイル (**systemctl start postgres** コマンド) を使用して **postgresql** サーバーを起動するユーザーは、この変更の影響を受けません。以前に **postmaster** バイナリーを介して **postgresql** サーバーを直接起動していた場合は、今後は代わりに **postgres** バイナリーを使用する必要があります。
- PostgreSQL は、パッケージで PDF 形式のドキュメントを提供しなくなりました。代わりに オンラインドキュメント を使用してください。

PostgreSQL の使用 も参照してください。

postgresql:16 ストリームをインストールするには、次のコマンドを使用します。

```
# yum module install postgresql:16
```

RHEL 8 内で以前の **postgresql** 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 の説明に従い、PostgreSQL の RHEL 8 バージョンへの移行 で説明されているように PostgreSQL データを移行します。

postgresql モジュールストリームのサポート期間は、Red Hat Enterprise Linux アプリケーションストリームのライフサイクル を参照してください。

Jira:RHEL-3636

Git がバージョン 2.43.0 にリベース

Git バージョン管理システムがバージョン 2.43.0 に更新され、以前にリリースされたバージョン 2.39 に対するバグ修正、機能拡張、およびパフォーマンス向上が実現しました。

主な機能拡張は、次のとおりです。

- **git check-attr** コマンドで新しい **--source** オプションを使用して、現在の作業ディレクトリーではなく、提供されたツリー状のオブジェクトから **.gitattributes** ファイルを読み取ることができるようになりました。
- Git は、**WWW-Authenticate** レスポンスタイプヘッダーからの情報を認証情報ヘルパーに渡すことができるようになりました。
- コミットが空の場合、**git format-patch** コマンドは空のファイルを作成する代わりに、コミットのヘッダーを含む出力ファイルを書き込むようになりました。

- **git blame --contents=<file> <revision> -- <path>** コマンドを使用して、**<file>** の内容を起点として **<revision>** に至る履歴を通じて、行の起源を検索できるようになりました。
- **git log --format** コマンドは、**--decorate** オプションが提供する機能を拡張するためのさらなるカスタマイズ用に、**%(decorate)** プレースホルダーを受け入れるようになりました。

Jira:RHEL-17103[1]

Git LFS がバージョン 3.4.1 にリベース

Git Large File Storage (LFS) エクステンションがバージョン 3.4.1 に更新されました。これにより、以前にリリースされたバージョン 3.2.0 に対するバグ修正、機能拡張、およびパフォーマンス向上が実現しました。

主な変更点は、以下のとおりです。

- **git lfs push** コマンドが、標準入力から参照とオブジェクト ID を読み取ることができるようになりました。
- Git LFS は、Git に依存せずに代替リモートを処理するようになりました。
- Git LFS は、認証情報ヘルパーとして **WWW-Authenticate** レスポンスタイプヘッダーをサポートするようになりました。

Jira:RHEL-17102[1]

Python インタープリターのパフォーマンスが向上

RHEL 8 でサポートされているすべてのバージョンの Python が、アップストリームのデフォルトである **-O3** 最適化フラグを使用してコンパイルされるようになりました。その結果、Python アプリケーションとインタープリター自体のパフォーマンスが向上しました。

この変更は、次のアドバイザリーのリリースで利用可能です。

- **python3.12** - RHSA-2024:6961
- **python3.11** - RHSA-2024:6962
- **python3** - RHSA-2024:6975
- **python39** モジュール - RHSA-2024:5962

Jira:RHEL-49614[1], Jira:RHEL-49636, Jira:RHEL-49644, Jira:RHEL-49638

新しい nodejs:22 モジュールストリームが利用可能になりました

RHEA-2025:0734 アドバイザリーのリリースに伴い、新しいモジュールストリーム **nodejs:22** が利用可能になりました。

RHEL 8.10 に含まれる **Node.js 22** には、RHEL 8.9 以降で利用可能な **Node.js 20** に比べて、多数の新機能、バグ修正、セキュリティー修正、およびパフォーマンスの改善を提供します。

主な変更点は、以下のとおりです。

- **V8** JavaScript エンジンがバージョン 12.4 にアップグレードされました。

- **V8 Maglev** コンパイラーは、これを利用可能なアーキテクチャー (AMD および Intel 64 ビットアーキテクチャーと 64 ビット ARM アーキテクチャー) でデフォルトで有効になりました。
- **Maglev** は、短命の CLI プログラムのパフォーマンスを向上させます。
- **npm** パッケージマネージャーが、バージョン 10.9.0 にアップグレードされました。
- **node --watch** モードは現在安定していると見なされます。**watch** モードでは、監視対象ファイルの変更により **Node.js** プロセスが再起動されます。
- **WebSocket** のブラウザー互換実装は現在、安定していると見なされ、デフォルトで有効になっています。その結果、外部依存関係なしで Node.js への WebSocket クライアントが利用できるようになります。
- **Node.js** には、**package.json** からのスクリプトを実行するための実験的な機能が含まれるようになりました。この機能を使用するには、**node --run <script-in-package.json>** コマンドを実行します。

nodejs:22 モジュールストリームをインストールするには、次のように入力します。

```
# dnf module install nodejs:22
```

nodejs20 ストリームからアップグレードする場合は、新しいストリームへの切り替え を参照してください。

nodejs Application Streams のサポート期間の詳細は、 Red Hat Enterprise Linux Application Streams のライフサイクル を参照してください。

Jira:RHEL-35991

4.10. コンパイラーおよび開発ツール

新しい GCC Toolset 14

GCC Toolset 14 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。これは、AppStream リポジトリー内の Software Collection の形式で Application Stream として利用できます。

GCC Toolset 14 では、RHEA-2024:8851 アドバイザリーのリリースに伴い、次のツールとバージョンが提供されます。

- GCC 14.2
- GDB 14.2
- **binutils** 2.41
- **annobin** 12.70
- **dwz** 0.14

GCC Toolset 14 をインストールするには、root として次のコマンドを実行します。

```
# yum install gcc-toolset-14
```

GCC Toolset 14 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-14 <tool>
```

GCC Toolset 14 のツールバージョンによってこの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、以下のコマンドを実行します。

```
$ scl enable gcc-toolset-14 bash
```

GCC Toolset 14 コンポーネントは、**gcc-toolset-14-toolchain** コンテナーイメージで利用できます。

詳細は、GCC Toolset 14 および GCC Toolset の使用 を参照してください。

Jira:RHEL-34596[1], Jira:RHEL-30411

GCC Toolset 14: GCC がバージョン 14.2 にリベース

GCC Toolset 14 では、RHEA-2024:8864 アドバイザリーのリリースに伴い、GNU コンパイラーコレクション (GCC) がバージョン 14.2 に更新されました。

主な変更点は、以下のとおりです。

- 最適化と診断の改善
- 一連のハードニングフラグを有効化する新しい包括的な **-fhardened** オプション
- 関数の途中に制御を移す攻撃を検出する新しい **-fharden-control-flow-redundancy** オプション
- 関数と変数のスタックスクラビングプロパティーを制御する新しい **strub** 型属性
- 特定の **mem*** 関数のインライン展開を強制する新しい **-finline-stringops** オプション
- 新しい OpenMP 5.1、5.2、6.0 機能のサポート
- いくつかの C23 の新機能
- 複数の新しい C++23 および C++26 機能
- いくつかの C++ 不具合報告の解決
- C++ ライブラリーにおける C++20、C++23、C++26 の実験的サポートの新規追加と改良
- 64 ビット ARM アーキテクチャーの新しい CPU のサポート
- 64 ビット Intel アーキテクチャーの複数の新しい命令セットアーキテクチャー (ISA) 拡張 (例: AVX10.1、AVX-VNNI-INT16、SHA512、SM4)
- GCC の静的アナライザーの新しい警告
- 一部の警告をエラーに変更 (詳細は、GCC 14 への移植 を参照)
- さまざまなバグ修正

GCC 14 の変更点の詳細は、アップストリームの GCC リリースノート を参照してください。

Jira:RHEL-30412[1]

GCC Toolset 14: GDB がバージョン 14.2 にリベース

GCC Toolset 14 では、RHBA-2024:8862 アドバイザリーのリリースに伴い、GDB がバージョン 14.2 に更新されました。以下の段落では、GDB 12.1 以降の主な変更点を示します。

全般:

- **info breakpoints** コマンドは、無効なブレークポイントの有効なブレークポイントの位置を **y-** 状態で表示するようになりました。
- ELF の Zstandard (**ELFCOMPRESS_ZSTD**) で圧縮されたデバッグセクションのサポートが追加されました。
- テキストユーザーインターフェイス (TUI) では、現在の位置を示すインジケーターで強調表示されるソースコードとアセンブリコードのスタイルがデフォルトで設定されなくなりました。スタイルを再度有効にするには、新しいコマンド **set style tui-current-position** を使用します。
- 新しい簡易変数 **$_inferior_thread_count** には、現在の inferior 内のライブスレッドの数が含まれます。
- コード位置が複数あるブレークポイントの場合、GDB は **<breakpoint_number>.<location_number>** 構文を使用してコードの場所を出力するようになりました。
- ブレークポイントにヒットすると、GDB は **$_hit_bpnum** および **$_hit_locno** 簡易変数をヒットしたブレークポイント番号とコード位置番号に設定するようになりました。これで、**disable $_hit_bpnum** コマンドを使用して最後にヒットしたブレークポイントを無効にしたり、**disable $_hit_bpnum.$_hit_locno** コマンドを使用して特定のブレークポイントコードの位置のみを無効にしたりできるようになりました。
- **NO_COLOR** 環境変数のサポートが追加されました。
- 64 ビットを超える整数型のサポートが追加されました。
- マルチターゲット機能設定用の新しいコマンドを使用して、リモートターゲット機能セットを設定できます (コマンドの **set remote <name>-packet** と **show remote <name>-packet** を参照)。
- デバッガーアダプタープロトコルのサポートが追加されました。
- 新しい **inferior** キーワードを使用して、ブレークポイントを inferior 固有のブレークポイントに設定できるようになりました (コマンドの **break** または **watch** を参照)。
- 新しい **$_shell()** 簡易関数を使用して、式の評価中にシェルコマンドを実行できるようになりました。

既存コマンドの変更点:

- **break**、**watch**
  - **break** および **watch** コマンドで **thread** または **task** キーワードを複数回使用すると、キーワードの最後のインスタンスのスレッドまたはタスク ID が使用されるのではなく、エラーが発生するようになりました。
  - 同じ **break** または **watch** コマンドで **thread**、**task**、**inferior** キーワードを複数使用できなくなりました。

- **printf、dprintf**
  - **printf** および **dprintf** コマンドは、**print** コマンドと同じ方法で式をフォーマットする **%V** 出力形式を受け入れるようになりました。たとえば **printf "%V[-array-indexes on]", <array>** のように、コマンドの後に括弧 **[...]** で囲んだ追加の print オプションを使用して、出力形式を変更することもできます。
- **list**
  - **.** 引数を使用して、現行フレームの実行ポイント付近の位置、または inferior が開始されていない場合は **main()** 関数の開始付近の位置を出力できるようになりました。
  - ファイル内で利用可能な行数より多くのソース行をリストしようとすると、警告が出され、ユーザーに **.** 引数を参照するよう指示されます。
- **document user-defined**
  - ユーザー定義のエイリアスを文書化できるようになりました。

新しいコマンド:

- **set print nibbles [on|off]** (デフォルト: **off**)、**show print nibbles** - **print/t** コマンドを使用した場合に 4 ビット (ニブル) のグループでバイナリー値を表示するかどうかを制御します。
- **set debug infcall [on|off]** (デフォルト: **off**)、**show debug infcall** - inferior 関数呼び出しに関する追加のデバッグメッセージを出力します。
- **set debug solib [on|off]** (デフォルト: **off**)、**show debug solib** - 共有ライブラリーの処理に関する追加のデバッグメッセージを出力します。
- **set print characters <LIMIT>**、**show print characters**、**print -characters <LIMIT>** - 文字列うち何文字を出力するか制御します。
- **set debug breakpoint [on|off]** (デフォルト: **off**)、**show debug breakpoint** - ブレークポイントの挿入と削除に関する追加のデバッグメッセージを出力します。
- **maintenance print record-instruction [N]** - 指定された命令の記録された情報を出力します。
- **maintenance info frame-unwinders** - 現在有効なフレームアンワインダーを優先度の高いものから順にリストします。
- **maintenance wait-for-index-cache** - インデックスキャッシュへの保留中の書き込みがすべて完了するまで待機します。
- **info main** - プログラムのエントリーポイントを識別するためにメインシンボルに関する情報を出力します。
- **set tui mouse-events [on|off]** (デフォルト: **on**)、**show tui mouse-events** - マウスクリックイベントを、TUI および Python エクステンションに送信するか (**on** の場合)、ターミナルに送信するか (**off** の場合) を制御します。

Machine Interface (MI) の変更:

- MI バージョン 1 は削除されました。

- MI は、逆実行履歴をすべて使用すると、**no-history** が報告されるようになりました。
- **-break-insert** コマンドの出力で、**thread** および **task** ブレークポイントフィールドが 2 回報告されなくなりました。
- 存在しないスレッド ID でスレッド固有のブレークポイントを作成できなくなりました。
- **-stack-list-arguments**、**-stack-list-locals**、**-stack-list-variables**、および **-var-list-children** コマンドの **--simple-values** 引数は、ターゲットが simple の場合に参照型を simple として扱うようになりました。
- **-break-insert** コマンドは、inferior 固有のブレークポイントを作成するための新しい **-g thread-group-id** オプションを受け入れるようになりました。
- ブレークポイント作成通知と **-break-insert** コマンドの出力に、メインブレークポイントと各ブレークポイントの位置のオプションフィールド **inferior** を追加できるようになりました。
- **breakpoint-hit** の停止理由を示す非同期レコードに、ブレークポイントの位置が複数の場合にコードの位置番号を示すオプションフィールド **locno** が含まれるようになりました。

GDB Python API の変更点:

- Events
  - 新しい **gdb.ThreadExitedEvent** イベント。
  - **progspace** および **reload** 属性を持つ **ExecutableChangedEvent** オブジェクトを出力する新しい **gdb.executable_changed** イベントレジストリー。
  - 新しい **gdb.events.new_progspace** および **gdb.events.free_progspace** イベントレジストリー。**NewProgpspaceEvent** および **FreeProgspaceEvent** イベント型を出力します。両方のイベント型に、GDB に追加または GDB から削除される **gdb.Progspace** プログラムスペースを指定するための単一の属性 **progspace** があります。
- **gdb.unwinder.Unwinder** クラス
  - **name** 属性は読み取り専用になりました。
  - **__init__** 関数の name 引数は **str** 型である必要があります。そうでない場合は **TypeError** が発生します。
  - **enabled** 属性は **bool** 型のみを受け入れるようになりました。
- **gdb.PendingFrame** クラス
  - 新しいメソッド: **name**、**is_valid**、**pc**、**language**、**find_sal**、**block**、**function**。これらは **gdb.Frame** クラスの同様のメソッドを反映しています。
  - **create_unwind_info** 関数の **frame-id** 引数は、**pc**、**sp**、および **special** 属性に対して整数または **gdb.Value** オブジェクトのいずれかにできるようになりました。
- **gdb.PendingFrame.create_unwind_info** 関数に渡すことができる新しい **gdb.unwinder.FrameId** クラス。
- **gdb.disassembler.DisassemblerResult** クラスはサブクラス化できなくなりました。

- **gdb.disassembler** モジュールにスタイルサポートが含まれるようになりました。
- 新しい **gdb.execute_mi(COMMAND, [ARG]...)** 関数。GDB/MI コマンドを呼び出して結果を Python ディクショナリーとして返します。
- 新しい **gdb.block_signals()** 関数。GDB が処理する必要のあるすべてのシグナルをブロックするコンテキストマネージャーを返します。
- **threading.Thread** クラスの新しい **gdb.Thread** サブクラス。**start** メソッドで **gdb.block_signals** 関数を呼び出します。
- **gdb.parse_and_eval** 関数に、グローバルシンボルの解析を制限するための新しい **global_context** パラメーターが追加されました。
- **gdb.Inferior** クラス
  - 新しい **arguments** 属性。既知の場合に inferior へのコマンドライン引数を保持します。
  - 新しい **main_name** 属性。既知の場合に inferior の **main** 関数の名前を保持します。
  - 新しい **clear_env**、**set_env**、および **unset_env** メソッド。inferior が開始される前にその環境を変更できます。
- **gdb.Value** クラス
  - オブジェクトの値を割り当てる新しい **assign** メソッド。
  - 配列のような値を配列に変換する新しい **to_array** メソッド。
- **gdb.Progspace** クラス
  - 新しい **objfile_for_address** メソッド。指定されたアドレス (存在する場合) をカバーする **gdb.Objfile** オブジェクトを返します。
  - **Progspace.filename** 変数に対応する **gdb.Objfile** オブジェクトを保持する新しい **symbol_file** 属性 (ファイル名が **None** の場合は **None**)。
  - 新しい **executable_filename** 属性。**exec-file** または **file** コマンドによって設定されたファイル名の文字列を保持します (実行可能ファイルが設定されていない場合は **None**)。
- **gdb.Breakpoint** クラス
  - 新しい **inferior** 属性。inferior 固有のブレークポイントの inferior ID (整数) が含まれます (そのようなブレークポイントが設定されていない場合は **None**)。
- **gdb.Type** クラス
  - 新しい **is_array_like** および **is_string_like** メソッド。型の実際の型コードにかかわらず、配列型か文字列型かを反映します。
- 新しい **gdb.ValuePrinter** クラス。pretty-printer を適用した結果の基本クラスとして使用できます。
- 新しく実装された **gdb.LazyString.__str__** メソッド。
- **gdb.Frame** クラス

    - 新しい **static_link** メソッド。ネストされた関数フレームの外側のフレームを返します。
    - 新しい **gdb.Frame.language** メソッド。フレームの言語の名前を返します。
- **gdb.Command** クラス
    - GDB は、文字列をヘルプ出力として使用する前に、**gdb.Command** クラスと **gdb.Parameter** サブクラスのドキュメント文字列を再フォーマットして、各行の先頭の不要な空白を削除するようになりました。
- **gdb.Objfile** クラス
    - 新しい **is_file** 属性。
- 新しい **gdb.format_address(ADDRESS, PROGSPACE, ARCHITECTURE)** 関数。逆アセンブラーからアドレス、シンボル、オフセット情報を出力する際に同じ形式を使用します。
- 新しい **gdb.current_language** 関数。現在の言語の名前を返します。
- GDB の逆アセンブラーをラップするための新しい Python API。**gdb.disassembler.register_disassembler(DISASSEMBLER, ARCH)**、**gdb.disassembler.Disassembler**、**gdb.disassembler.DisassembleInfo**、**gdb.disassembler.builtin_disassemble(INFO, MEMORY_SOURCE)**、**gdb.disassembler.DisassemblerResult** が含まれます。
- 新しい **gdb.print_options** 関数。**gdb.Value.format_string** 関数で受け入れられる形式で、一般的な出力オプションのディクショナリーを返します。
- **gdb.Value.format_string** 関数
    - **gdb.Value.format_string** は、**print** またはその他の同様の操作中に呼び出された場合に、**print** コマンドで提供される形式を使用するようになりました。
    - **gdb.Value.format_string** は、**summary** キーワードを受け入れるようになりました。
- 新しい **gdb.BreakpointLocation** Python 型。
- **gdb.register_window_type** メソッドは、受け入れられるウィンドウ名のセットを制限するようになりました。

アーキテクチャー固有の変更:

- 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
    - **libopcodes** ライブラリーを使用した逆アセンブラースタイルのサポートが追加されました。現在、これがデフォルトとして使用されています。**set style disassembler *** コマンドを使用して、逆アセンブラーの出力スタイルを変更できます。代わりに Python Pygments スタイルを使用するには、新しい **maintenance set libopcodes-styling off** コマンドを使用します。
- 64 ビット ARM アーキテクチャー
    - Memory Tagging Extension (MTE) のメモリータグデータをダンプするためのサポートが追加されました。

    - Scalable Matrix Extension 1 および 2 (SME/SME2) のサポートが追加されました。ZA 状態での手動関数呼び出しや、DWARF に基づく Scalable Vector Graphics (SVG) の変更の追跡など、一部の機能はまだ試験版またはアルファ版と見なされています。
    - Thread Local Storage (TLS) 変数のサポートが追加されました。
    - ハードウェアウォッチポイントのサポートが追加されました。
- 64 ビット IBM Z アーキテクチャー
    - IBM Z ターゲット上の新しい **arch14** 命令の記録および再生のサポート (specialized-function-assist 命令 **NNPA** を除く)。
- IBM Power Systems (リトルエンディアン)
    - POWER11 のベース有効化のサポートを追加しました。

Jira:RHELDOCS-18598[1], Jira:RHEL-36225, Jira:RHEL-36518

GCC Toolset 14: annobin がバージョン 12.70 にリベース

GCC Toolset 14 では、RHBA-2024:8863 アドバイザリーのリリースに伴い、**annobin** がバージョン 12.70 に更新されました。更新されたバイナリーテスト用の **annobin** ツールセットにより、さまざまなバグ修正が提供され、新しいテストが導入され、新しいバージョンの GCC、Clang、LLVM、および Go コンパイラーをビルドして使用するようにツールが更新されます。強化されたツールにより、非標準的な方法でビルドされたプログラムの新しい問題を検出できます。

Jira:RHEL-30409[1]

GCC Toolset 13: GCC が AMD Zen 5 をサポートするようになる

RHBA-2024:8829 アドバイザリーのリリースに伴い、GCC の GCC Toolset 13 バージョンに AMD Zen 5 プロセッサー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。サポートを有効にするには、**-march=znver5** コマンドラインオプションを使用します。

Jira:RHEL-36524[1]

LLVM ツールセットが 18.1.8 に更新される

RHBA-2024:8828 アドバイザリーのリリースに伴い、LLVM Toolset がバージョン 18.1.8 に更新されました。

LLVM の主な更新点:

- 以下の命令の定数式バリアントが削除されました。**and**、**or**、**lshr**、**ashr**、**zext**、**sext**、**fptrunc**、**fpext**、**fptoui**、**fptosi**、**uitofp**、**sitofp**。
- **llvm.exp10** 組み込み関数が追加されました。
- グローバル変数の **code_model** 属性が追加されました。
- AArch64、AMDGPU、PowerPC、RISC-V、SystemZ、x86 アーキテクチャーのバックエンドが改善されました。
- LLVM ツールが改善されました。

Clang の主な機能拡張:

- C++20 機能のサポート:
  - Clang は、グローバルモジュールフラグメント内の宣言に対して One Definition Rule (ODR) チェックを実行しなくなりました。より厳密な動作を有効にするには、**-Xclang -fno-skip-odr-check-in-gmf** オプションを使用してください。
- C++23 機能のサポート:
  - ラムダでの属性の使用に関する警告するための新しい診断フラグ **-Wc++23-lambda-attributes** が追加されました。
- C++2c 機能のサポート:
  - 同じスコープ内で、_ 文字をプレースホルダー変数名として複数回使用できるようになりました。
  - 属性が、文字列リテラルである属性パラメーター内で未評価の文字列を要求するようになりました。
  - C++26 の列挙型に対する非推奨の算術変換が削除されました。
  - テンプレートパラメーター初期化の仕様が改善されました。
- 変更点の完全なリストは、Clang の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。

Clang の ABI の変更点:

- x86_64 の SystemV ABI に従い、**__int128** 引数がレジスターとスタックスロット間で分割されなくなりました。
- 詳細は、Clang の ABI 変更点のリスト を参照してください。

後方互換性のない主な変更点:

- テンプレート化された演算子の引数の逆順に関するバグ修正により、以前は C++17 で受け入れられていたコードが C++20 では機能しなくなります。
- **GCC_INSTALL_PREFIX** CMake 変数 (デフォルトの **--gcc-toolchain=** を設定する変数) が非推奨になり、削除される予定です。代わりに、設定ファイルで **--gcc-install-dir=** または **--gcc-triple=** オプションを指定してください。
- プリコンパイル済みヘッダー (PCH) 生成 (**-c -xc-header** および **-c -xc++-header**) のデフォルトの拡張子名が、**.gch** ではなく **.pch** になりました。
- **-include a.h** が **a.h.gch** ファイルをプローブするときに、そのファイルが Clang PCH ファイルまたは Clang PCH ファイルを含むディレクトリーでない場合、include が **a.h.gch** を無視するようになりました。
- **__has_cpp_attribute** および **__has_c_attribute** が特定の C++-11 スタイルの属性に対して誤った値を返すバグが修正されました。
- 逆の **operator==** の追加時に一致する **operator!=** を検出する際のバグが修正されました。
- 関数テンプレートの名前マングリングルールが変更され、テンプレートパラメーターリストまたは requires 句で関数をオーバーロードできるようになりました。

- **-Wenum-constexpr-conversion** 警告が、システムヘッダーとマクロでデフォルトで有効になりました。これは、次の Clang リリースでハード (ダウングレード不可能な) エラーに変更されます。
- C++20 名前付きモジュールのインポートされたモジュールへのパスが、ハードコードできなくなりました。依存するすべてのモジュールをコマンドラインから指定する必要があります。
- **import <module>** を使用してモジュールをインポートできなくなりました。Clang は明示的にビルドされたモジュールを使用します。
- 詳細は、互換性を破壊する可能性のある変更点のリスト を参照してください。

詳細は、LLVM リリースノート および Clang リリースノート を参照してください。

LLVM Toolset は Rolling Application Stream であり、最新バージョンのみがサポートされます。詳細は、Red Hat Enterprise Linux Application Streams ライフサイクル ドキュメントを参照してください。

Jira:RHEL-30907[1]

Rust Toolset がバージョン 1.79.0 にリベース

RHBA-2024:8827 アドバイザリーのリリースに伴い、Rust Toolset がバージョン 1.79.0 に更新されました。以前提供されていたバージョン 1.75.0 以降の主な機能拡張は次のとおりです。

- 新しい **offset_of!** マクロ
- C 文字列リテラルのサポート
- インライン **const** 式のサポート
- 関連型の位置における境界のサポート
- 一時的な自動有効期間延長の改善
- **unsafe** 前提条件のデバッグアサーション

Rust ツールセットは Rolling Application Stream であり、最新バージョンのみがサポートされます。詳細は、Red Hat Enterprise Linux Application Streams ライフサイクル ドキュメントを参照してください。

Jira:RHEL-30073[1]

Go Toolset がバージョン 1.23 にリベース

RHBA-2025:3823 アドバイザリーのリリースに伴い、Go Toolset がバージョン 1.23 に更新されました。

主な機能拡張は、次のとおりです。

- **for-range** ループは、次のタイプのイテレーター関数を受け入れます。
  - **func(func() bool)**
  - **func(func(K) bool)**
  - **func(func(K, V) bool)**

for-range ループのイテレーション値は、イテレーター引数関数の呼び出しによって作成されます。参照リンクは、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。

- Go Toolchain により、使用状況や破損統計情報を収集できます。これは、Go チームが Go Toolchain がどのように使用され、どのように機能するかを理解するのに役立ちます。デフォルトでは、Go Telemetry はテレメトリーデータをアップロードせず、ローカルにのみ保存します。詳細は、アップストリームの Go Telemetry ドキュメント を参照してください。

- **go vet** サブコマンドには、参照ファイルで使用する Go のバージョンに対して新しすぎるシンボルへの参照にフラグを立てる **stdversion** アナライザーが含まれています。

- **cmd** および **cgo** 機能は、C リンカーにフラグを渡すための **-ldflags** オプションをサポートしています。**go** コマンドは、非常に大きな **CGO_LDFLAGS** 環境変数を使用する場合に、**argument list too long** エラーを回避するために、このフラグを自動的に使用します。

- **trace** ユーティリティーは、部分的に壊れたトレースを許容し、トレースデータを回復しようとします。これはクラッシュが発生した場合にクラッシュに至るまでのトレースを取得できるため、特に便利です。

- 未処理のパニックまたはその他の致命的なエラーの後にランタイムによって出力されるトレースバックには、**goroutine** のスタックトレースを最初の **goroutine** と区別するためのインデントが含まれます。

- プロファイルガイドによる最適化を使用したコンパイラービルド時間のオーバーヘッドが 1 桁のパーセンテージに削減されました。

- 新しい **-bindnow** リンカーフラグにより、動的にリンクされた ELF バイナリーをビルドするときに即時の関数バインディングが有効になります。

- **//go:linkname** リンカーディレクティブは、定義で **//go:linkname** でマークされていない標準ライブラリーおよびランタイムの内部シンボルを参照しなくなりました。

- プログラムが **Timer** または **Ticker** を参照しなくなった場合、**Stop** メソッドが呼び出されていなくても、これらはガベージコレクションによってすぐにクリーンアップされます。**Timer** または **Ticker** に関連付けられたタイマーチャネルは、現在バッファーなし (容量 0) になっています。これにより、**Reset** メソッドまたは **Stop** メソッドが呼び出されるたびに、呼び出し後に古い値が送受信されなくなります。

- 新しい **unique** パッケージは、**interning** または **hash-consing** などの値を正規化する機能を提供します。

- 新しい **iter** パッケージは、ユーザー定義のイテレーターを使用するための基本的な定義を提供します。

- **slices** および **maps** パッケージには、イテレーターで使用するいくつかの新しい関数が導入されています。

- 新しい **structs** パッケージは、メモリーレイアウトなど、含まれる struct 型のプロパティーを変更する struct フィールドの型を提供します。

- 次のパッケージにマイナーな変更が加えられました。

  - **archive/tar**

  - **crypto/tls**

- **crypto/x509**
- **database/sql**
- **debug/elf**
- **encoding/binary**
- **go/ast**
- **go/types**
- **math/rand/v2**
- **net**
- **net/http**
- **net/http/httptest**
- **net/netips**
- **path/filepath**
- **reflect**
- **runtime/debug**
- **runtime/pprof**
- **runtime/trace**
- **slices**
- **sync**
- **sync/atomic**
- **syscall**
- **testing/fstest**
- **text/template**
- **time**
- **unicode/utf16**

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。

Go Toolset は Rolling Application Stream であり、Red Hat は最新バージョンのみをサポートします。詳細は、Red Hat Enterprise Linux Application Streams ライフサイクル ドキュメントを参照してください。

Jira:RHEL-83447[1]

Go Toolset がバージョン 1.22 にリベース

RHSA-2024:8876 アドバイザリーのリリースに伴い、Go Toolset がバージョン 1.22 に更新されました。

主な機能拡張は、次のとおりです。

- for ループ内の変数がイテレーションごとに作成されるようになりました。これにより、不慮の共有バグを防ぐことができます。さらに、for ループは整数の範囲をカバーできるようになりました。
- ワークスペース内のコマンドで、ワークスペースの依存関係にベンダーディレクトリーを使用できるようになりました。
- **go get** コマンドが、従来の **GOPATH** モードをサポートしなくなりました。この変更は、**go build** コマンドと **go test** コマンドには影響しません。
- **vet** ツールが、for ループの新しい動作に合わせて更新されました。
- 型ベースのガベージコレクションメタデータを各ヒープオブジェクトの近くに保持することで、CPU パフォーマンスが向上するようになりました。
- インライン化の最適化が改善され、プロファイルに基づく最適化のサポートの強化によりパフォーマンスが向上しました。
- 新しい **math/rand/v2** パッケージが利用可能です。
- メソッドとワイルドカードのサポートにより、HTTP ルーティングパターンが強化されました。

詳細は、アップストリームの Go リリースノートを参照してください。

Go ツールセットは Rolling Application Stream であり、最新バージョンのみがサポートされます。詳細は、Red Hat Enterprise Linux Application Streams ライフサイクル ドキュメントを参照してください。

Jira:RHEL-46972[1]

elfutils がバージョン 0.190 にリベース

elfutils パッケージが、バージョン 0.190 に更新されました。以下は、主な改善点です。

- **libelf** ライブラリーが、相対再配置 (RELR) をサポートするようになりました。
- **libdw** ライブラリーが、**.debug_[ct]u_index** セクションを認識するようになりました。
- **eu-readelf** ユーティリティーが、ELF セクションを使用せずに動的セグメントを使用してシンボルを表示する、新しい **-Ds**, **--use-dynamic --symbol** オプションをサポートするようになりました。
- **eu-readelf** ユーティリティーが、**.gdb_index** バージョン 9 を表示できるようになりました。
- 新しい **eu-scrlines** ユーティリティーは、指定された DWARF または ELF ファイルに関連付けられたソースファイルのリストをコンパイルします。
- **debuginfod** サーバースキーマが変更され、ファイル名の表現が 60% 圧縮されました (インデックスの再作成が必要です)。

Jira:RHEL-15924

valgrind が 3.22 に更新される

valgrind パッケージがバージョン 3.22 に更新されました。以下は、主な改善点です。

- **valgrind memcheck** は、C 関数 **memalign**、**posix_memalign**、および **aligned_alloc** に指定された値と、C++17 で調整された **new** Operator が有効なアラインメント値であるかどうかをチェックするようになりました。
- **valgrind memcheck** は、C++14 サイズおよび C++17 に揃えられた **new** および **delete** Operator の不一致検出をサポートするようになりました。
- DWARF デバッグ情報の遅延読み取りのサポートが追加され、**debuginfo** パッケージがインストールされている場合に起動が高速化されました。

Jira:RHEL-15926

Clang リソースディレクトリーが移動しました

Clang が内部ヘッダーとライブラリーを保存する Clang リソースディレクトリーが、**/usr/lib64/clang/17** から **/usr/lib/clang/17** に移動しました。

Jira:RHEL-9299

新しい grafana-selinux パッケージ

以前は、**grafana-server** のデフォルトのインストールは **unconfined_service_t** SELinux タイプとして実行されていました。この更新では、**grafana-server** 用の SELinux ポリシーが含まれ、**grafana-server** とともにデフォルトでインストールされる、新しい **grafana-selinux** パッケージが追加されます。その結果、**grafana-server** は **grafana_t** SELinux タイプとして実行されるようになりました。

Jira:RHEL-7503

GCC Toolset 13 の更新

GCC Toolset 13 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。これは、AppStream リポジトリー内の Software Collection の形式で Application Stream として利用できます。

RHEL 8.10 で導入された注目すべき変更点は次のとおりです。

- GCC コンパイラーがバージョン 13.2.1 に更新され、アップストリーム GCC で利用可能な多くのバグ修正と機能拡張が提供されます。
- **binutils** は、**-march=znver5** コンパイラースイッチを通じて、**znver5** コアに基づく AMD CPU をサポートするようになりました。
- **annobin** がバージョン 12.32 に更新されました。
- GCC の **annobin** プラグインは、オブジェクトファイルに保存するメモに対して、より圧縮された形式をデフォルトで使用するようになりました。その結果、特に大規模で複雑なプログラムでは、オブジェクトファイルが小さくなり、リンク時間が短縮されます。

次のツールとバージョンが GCC Toolset 13 によって提供されます。

ツール	バージョン
GCC	13.2.1
GDB	12.1
binutils	2.40
dwz	0.14
annobin	12.32

GCC Toolset 13 をインストールするには、root として次のコマンドを実行します。

```
# yum install gcc-toolset-13
```

GCC Toolset 13 からツールを実行するには、以下を使用します。

```
$ scl enable gcc-toolset-13 tool
```

GCC Toolset 13 のツールバージョンがこれらの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、以下を使用します。

```
$ scl enable gcc-toolset-13 bash
```

詳細は、GCC Toolset 13 および GCC Toolset の使用 を参照してください。

Jira:RHEL-25405[1]

LLVM Toolset がバージョン 17.0.6 にリベース

LLVM Toolset がバージョン 17.0.6 に更新されました。

主な機能拡張は、次のとおりです。

- Opaque ポインターの移行が完了しました。
- ミドルエンド最適化におけるレガシーパスマネージャーのサポートが削除されました。

Clang の変更:

- C++20 コルーチンは実験的なものとはみなされなくなりました。
- 最適化されていないビルドでの **std::move** 関数などのコード生成が改善されました。

詳細は、LLVM および Clang のアップストリームリリースノートを参照してください。

Jira:RHEL-9028

Rust Toolset がバージョン 1.75.0 にリベース

Rust Toolset がバージョン 1.75.0 に更新されました。

主な機能拡張は、次のとおりです。

- 定数評価時間が無制限になる
- よりクリーンなパニックメッセージ
- Cargo レジストリー認証
- トレイト内の **async fn** と不透明な戻り値のタイプ

Jira:RHEL-12964

Go Toolset がバージョン 1.21.0 にリベース

Go Toolset がバージョン 1.21.0 に更新されました。

主な機能拡張は、次のとおりです。

- **min**、**max**、**clear**ビルトインが追加されました。
- プロファイルに基づく最適化の正式サポートが追加されました。
- パッケージの初期化順序がより正確に定義されるようになりました。
- 型推論が改善されました。
- 下位互換性のサポートが改善されました。

詳細は、アップストリームの Go リリースノートを参照してください。

Jira:RHEL-11872[1]

papi は、新しいプロセッサーマイクロアーキテクチャーをサポートする

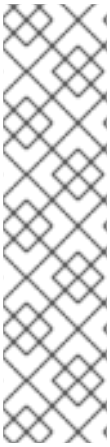
この機能拡張により、次のプロセッサーマイクロアーキテクチャー上の **papi** イベントプリセットを使用して、パフォーマンス監視ハードウェアにアクセスできるようになります。

- AMD Zen 4
- 第 4 世代 Intel® Xeon® スケーラブルプロセッサー

Jira:RHEL-9336[1]、Jira:RHEL-9320、Jira:RHEL-9337

Ant がバージョン 1.10.9 にリベース

ant:1.10 モジュールストリームがバージョン 1.10.9 に更新されました。このバージョンでは、プロバイダークラスとプロバイダー引数を使用したコード署名のサポートが提供されます。



注記

更新された **ant:1.10** モジュールストリームは、**ant** および **ant-lib** パッケージのみを提供します。Ant に関連する残りのパッケージは、サポートされていない CodeReady Linux Builder (CRB) リポジトリーの **javapackages-tools** モジュールで配布されており、更新されていません。

更新された **ant:1.10** モジュールストリームからのパッケージは、**javapackages-tools** モジュールからのパッケージと並行して使用することはできません。Ant 関連パッケージの完全なセットを使用する場合は、**ant:1.10** モジュールをアンインストールして無効にし、CRB リポジトリーを有効 にして、**javapackages-tools** モジュールをインストールする必要があります。

Jira:RHEL-5365

新しいパッケージ: maven-openjdk21

maven:3.8 モジュールストリームに、**maven-openjdk21** サブパッケージが含まれるようになりました。このサブパッケージは、OpenJDK 21 用の Maven JDK バインディングを提供し、システム OpenJDK 21 を使用するように Maven を設定します。

Jira:RHEL-17126[1]

cmake がバージョン 3.26 にリベース

cmake パッケージがバージョン 3.26 に更新されました。以下は、主な改善点です。

- C17 および C18 の言語規格のサポートが追加されました。
- **cmake** は、**/etc/os-release** ファイルでオペレーティングシステムの識別情報をクエリーできるようになりました。
- CUDA 20 および **nvtx3** ライブラリーのサポートが追加されました。
- Python 安定アプリケーションバイナリーインターフェイスのサポートが追加されました。
- Simplified Wrapper and Interface Generator (SWIG) ツールに Perl 5 のサポートが追加されました。

Jira:RHEL-7396

4.11. IDENTITY MANAGEMENT

Identity Management ユーザーは、外部アイデンティティープロバイダーを使用して IdM に認証できるようになる

この機能拡張により、Identity Management (IdM) ユーザーを、OAuth 2 デバイス認証フローをサポートする外部アイデンティティープロバイダー (IdP) に関連付けることができるようになりました。このような IdP の例としては、Red Hat build of Keycloak、Microsoft Entra ID (旧 Azure Active Directory)、GitHub、Google などがあります。

IdM に IdP 参照と関連付けられた IdP ユーザー ID が存在する場合は、それらを使用して IdM ユーザーが外部 IdP で認証できるようにすることができます。外部 IdP で認証と認可を実行した後、IdM ユーザーはシングルサインオン機能を備えた Kerberos チケットを受け取ります。ユーザーは、RHEL 8.7 以降で使用可能な SSSD バージョンで認証する必要があります。

Jira:RHELPLAN-123140[1]

ipa がバージョン 4.9.13 にリベース

ipa パッケージがバージョン 4.9.12 から 4.9.13 に更新されました。主な変更点は、以下のとおりです。

- IdM レプリカのインストールが、Kerberos 認証だけでなく、すべての IPA API および CA 要求に関しても、選択したサーバーに対して実行されるようになりました。
- 証明書の数が多い場合の **cert-find** コマンドのパフォーマンスが大幅に向上しました。
- **ansible-freeipa** パッケージが、バージョン 1.11 から 1.12.1 にリベースされました。

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。

Jira:RHEL-16936

期限切れの KCM Kerberos チケットの削除

以前は、Kerberos Credential Manager (KCM) に新しい認証情報を追加しようとした際にすでにストレージ領域の上限に達していた場合、新しい認証情報は拒否されていました。ユーザーのストレージ領域は、**max_uid_ccaches** 設定オプション (デフォルト値は 64) によって制限されます。この更新により、ストレージ領域の上限にすでに達している場合は、最も古い期限切れの認証情報が削除され、新しい認証情報が KCM に追加されます。期限切れの認証情報がない場合、操作は失敗し、エラーが返されます。この問題を防ぐには、**kdestroy** コマンドを使用して認証情報を削除し、領域を解放します。

Jira:SSSD-6216

ローカルユーザー向けの bcrypt パスワードハッシュアルゴリズムのサポート

この更新により、ローカルユーザーに対して **bcrypt** パスワードハッシュアルゴリズムを有効にできるようになります。**bcrypt** ハッシュアルゴリズムに切り替えるには、以下を行います。

1. **pam_unix.so sha512** 設定を **pam_unix.so blowfish** に変更して、**/etc/authselect/system-auth** および **/etc/authselect/password-auth** ファイルを編集します。
2. 変更を適用します。

```
# authselect apply-changes
```

3. **passwd** コマンドを使用してユーザーのパスワードを変更します。
4. **/etc/shadow** ファイルで、ハッシュアルゴリズムが **$2b$** に設定されていることを確認します。これは、**bcrypt** パスワードハッシュアルゴリズムが使用されていることを示します。

Jira:SSSD-6790

idp Ansible モジュールを使用すると、IdM ユーザーを外部 IdP に関連付けることができます

この更新により、**idp ansible-freeipa** モジュールを使用して、Identity Management (IdM) ユーザーを、OAuth 2 デバイス認可フローをサポートする外部アイデンティティープロバイダー (IdP) に関連付けることができます。IdM に IdP 参照および関連付けられた IdP ユーザー ID が存在する場合は、それらを使用して IdM ユーザーの IdP 認証を有効にすることができます。

外部 IdP で認証と認可を実行した後、IdM ユーザーはシングルサインオン機能を備えた Kerberos チケットを受け取ります。ユーザーは、RHEL 8.7 以降で使用可能な SSSD バージョンで認証する必要があります。

Jira:RHEL-16938

IdM は Ansible モジュール idoverrideuser、idoverridegroup、idview をサポートするようになりました

この更新により、**ansible-freeipa** パッケージに次のモジュールが含まれるようになりました。

idoverrideuser

Identity Management (IdM) LDAP サーバーに保存されているユーザーのユーザー属性 (ユーザーのログイン名、ホームディレクトリー、証明書、SSH キーなど) をオーバーライドできます。

idoverridegroup

IdM LDAP サーバーに保存されているグループの属性 (グループ名、GID、説明など) をオーバーライドできます。

idview

ユーザーおよびグループ ID のオーバーライドを整理し、特定の IdM ホストに適用できます。

将来的には、これらのモジュールを使用して、AD ユーザーがスマートカードを使用して IdM にログインできるようになります。

Jira:RHEL-16933

ansible-freeipa で DNS ゾーン管理の移譲が有効になる

dnszone ansible-freeipa モジュールを使用して、DNS ゾーン管理を委譲できるようになりました。**dnszone** モジュールの **permission** または **managedby** 変数を使用して、ゾーンごとのアクセス委譲権限を設定します。

Jira:RHEL-19133

ansible-freeipa ipauser および ipagroup モジュールが、新しい renamed 状態をサポートするようになる

この更新により、**ansible-freeipa ipauser** モジュールの **renamed** 状態を使用して、既存の IdM ユーザーのユーザー名を変更できるようになりました。また、**ansible-freeipa ipagroup** モジュールでこの状態を使用して、既存の IdM グループのグループ名を変更することもできます。

Jira:RHEL-4963

runasuser_group パラメーターが ansible-freeipa ipasudorule で利用できるようになる

この更新では、**ansible-freeipa ipasudorule** モジュールを使用して、**sudo** ルールの RunAs ユーザーのグループを設定できるようになりました。このオプションは、Identity Management (IdM) コマンドラインインターフェイスと IdM Web UI ですでに利用可能です。

Jira:RHEL-19129

389-ds-base がバージョン 1.4.3.39 にリベース

389-ds-base パッケージがバージョン 1.4.3.39 に更新されました。

Jira:RHEL-19028

HAProxy プロトコルが 389-ds-base パッケージでサポートされるようになりました

以前は、Directory Server はプロキシークライアントと非プロキシークライアント間の着信接続を区別していませんでした。この更新により、新しい多値設定属性 **nsslapd-haproxy-trusted-ip** を使用し

て、信頼できるプロキシーサーバーのリストを設定できるようになりました。**cn=config** エントリーの下に **nsslapd-haproxy-trusted-ip** が設定されている場合、Directory Server は HAProxy プロトコルを使用して追加の TCP ヘッダー経由でクライアント IP アドレスを受信します。これにより、アクセス制御命令 (ACI) を正しく評価し、クライアントトラフィックをログに記録できるようになります。

信頼されていないプロキシーサーバーがバインド要求を開始した場合、Directory Server は要求を拒否し、エラーログファイルに次のメッセージを記録します。

```
[time_stamp] conn=5 op=-1 fd=64 Disconnect - Protocol error - Unknown Proxy - P4
```

Jira:RHEL-19240

samba がバージョン 4.19.4 にリベース

samba パッケージはアップストリームバージョン 4.19.4 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と機能拡張が提供されています。主な変更点は以下のとおりです。

- 一貫したユーザーエクスペリエンスを実現するために、**smbget** ユーティリティーのコマンドラインオプションは名前が変更され、削除されました。ただし、これにより、ユーティリティーを使用する既存のスクリプトまたはジョブが破損する可能性があります。新しいオプションの詳細は、**smbget --help** コマンドと **smbget(1)** の man ページを参照してください。
- **winbind debug traceid** オプションが有効になっている場合、**winbind** サービスは次のフィールドもログに記録するようになりました。
  - **traceid**: 同じリクエストに属するレコードを追跡します。
  - **depth**: リクエストのネストレベルを追跡します。
- Samba は独自の暗号化実装を使用しなくなり、代わりに GnuTLS ライブラリーが提供する暗号化機能を全面的に使用するようになりました。
- **directory name cache size** オプションが削除されました。

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

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

Samba を更新した後、**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。

Jira:RHEL-16483[1]

新しい SSSD オプション: exop_force

exop_force オプションを使用すると、猶予ログインが残っていない場合でもパスワードの変更を強制できます。以前は、LDAP サーバーが猶予ログインが残っていないことを示した場合、SSSD はパスワードの変更を試行しませんでした。現在は、**sssd.conf** ファイルの **[domain/...]** セクションで **ldap_pwmodify_mode = exop_force** を設定すると、SSSD は猶予ログインが残っていなくてもパスワードの変更を試みます。

Jira:RHELDOCS-19863

4.12. WEB コンソール

RHEL Web コンソールで、Ansible およびシェルスクリプトを生成できるようになりました

Web コンソールで、**kdump** 設定ページの自動化スクリプトに簡単にアクセスしてコピーできるようになりました。生成されたスクリプトを使用して、特定の **kdump** 設定を複数のシステムに実装できます。

Jira:RHELDOCS-17060[1]

Storage でのストレージ管理およびパーティションサイズ変更の簡素化

Web コンソールの Storage セクションが再設計されました。新しいデザインにより、すべてのビューの視認性が向上しました。概要ページでは、すべてのストレージオブジェクトが包括的なテーブルに表示されるようになり、直接操作を実行しやすくなります。任意の行をクリックすると、詳細情報と追加のアクションが表示されます。さらに、Storage セクションからパーティションのサイズを変更できるようになりました。

Jira:RHELDOCS-17056[1]

4.13. RED HAT ENTERPRISE LINUX システムロール

ad_integration RHEL システムロールは、動的 DNS 更新オプションの設定をサポートするようになりました

この更新により、**ad_integration** RHEL システムロールは、Active Directory (AD) との統合時に、SSSD を使用した動的 DNS 更新の設定オプションをサポートするようになりました。デフォルトでは、SSSD は以下の場合に DNS レコードの自動更新を試みます。

- アイデンティティープロバイダーがオンラインになるとき (常に)
- 指定した間隔 (任意の設定)。デフォルトでは、AD プロバイダーは 24 時間ごとに DNS レコードを更新します。

ad_integration の新しい変数を使用して、これらの設定やその他の設定を変更できます。たとえば、**ad_dyndns_refresh_interval** を **172800** に設定すれば、DNS レコードの更新間隔を 48 時間に変更できます。ロール変数の詳細は、**/usr/share/doc/rhel-system-roles/ad_integration/** ディレクトリー内のリソースを参照してください。

Jira:RHELDOCS-17372[1]

RHEL System ロールの metrics は、PMIE Webhook の設定をサポートするようになりました。

この更新により、**metrics** RHEL システムロールの **metrics_webhook_endpoint** 変数を使用して、**global webhook_endpoint** PMIE 変数を自動的に設定できるようになりました。これにより、重要なパフォーマンスイベントに関するメッセージを受信する環境用のカスタム URL を提供できるようになります。これは通常、Event-Driven Ansible などの外部ツールで使用されます。

Jira:RHEL-18170

bootloader RHEL システムのロール

この更新では、**bootloader** RHEL システムロールが導入されます。この機能を使用すると、RHEL システム上のブートローダーとカーネルの設定を、安定かつ一貫したものにすることができます。要件、ロール変数、およびサンプル Playbook の詳細は、**/usr/share/doc/rhel-system-roles/bootloader/** ディ

レクトリーの README リソースを参照してください。

Jira:RHEL-3241

logging ロールは、出力モジュールの一般的なキューと一般的なアクションパラメーターをサポートします。

以前は、**logging** ロールで一般的なキューパラメーターと一般的なアクションパラメーターを設定することはできませんでした。この更新により、**logging** RHEL システムロールは、出力モジュールの一般的なキューパラメーターと一般的なアクションパラメーターの設定をサポートするようになりました。

Jira:RHEL-15440

新しい ha_cluster システムロール機能のサポート

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

- **dlm** や **gfs2** などの Resilient Storage パッケージを含むリポジトリーの有効化。リポジトリーにアクセスするには、Resilient Storage サブスクリプションが必要です。
- フェンシングレベルの設定。これにより、クラスターは複数のデバイスを使用してノードをフェンスできます。
- ノード属性の設定。

これらの機能を実装するために設定するパラメーターの詳細は、ha_cluster RHEL システムロールを使用した高可用性クラスターの設定 を参照してください。

Jira:RHEL-4624[1]、Jira:RHEL-22108、Jira:RHEL-14090

fapolicyd を設定するための新しい RHEL システムロール

新しい **fapolicyd** RHEL システムロールを使用すると、Ansible Playbook を使用して **fapolicyd** フレームワークを管理および設定できます。**fapolicyd** ソフトウェアフレームワークは、ユーザー定義のポリシーに基づいてアプリケーションの実行を制御します。

Jira:RHEL-16542

network RHEL システムロールが新しいルートタイプをサポートするようになりました

この機能拡張により、**network** RHEL システムロールで次のルートタイプを使用できるようになりました。

- **blackhole**
- **prohibit**
- **unreachable**

Jira:RHEL-21491[1]

rhc ロールに、表示名を設定するための新しい rhc_insights.display_name オプションが追加されました

新しい **rhc_insights.display_name** パラメーターを使用して、Red Hat Insights に登録されているシステムの表示名を設定または更新できるようになりました。このパラメーターを使用すると、希望に応じてシステムに名前を付けて、Insights Inventory 内のシステムを簡単に管理することができます。システ

ムがすでに Red Hat Insights に接続されている場合は、パラメーターを使用して既存の表示名を更新します。登録時に表示名が明示的に設定されていない場合、表示名はデフォルトでホスト名に設定されます。表示名をホスト名に自動的に戻すことはできませんが、手動でホスト名に設定することは可能です。

Jira:RHEL-16965

RHEL システムロールが LVM スナップショット管理をサポートするようになる

この機能拡張により、新しい **snapshot** RHEL システムロールを使用して、LVM スナップショットを作成、設定、管理できます。

Jira:RHEL-16553

postgresql RHEL システムロールが PostgreSQL 16 をサポートするようになりました

PostgreSQL サーバーをインストール、設定、管理、起動する **postgresql** RHEL システムロールが、PostgreSQL 16 をサポートするようになりました。

このシステムロールの詳細は、postgresql RHEL システムロールを使用した PostgreSQL のインストールと設定 を参照してください。

Jira:RHEL-18963

Ansible ホスト名を設定するための rhc ロールの新しい rhc_insights.ansible_host オプション

新しい **rhc_insights.ansible_host** パラメーターを使用して、Red Hat Insights に登録されているシステムの Ansible ホスト名を設定または更新できるようになりました。このパラメーターを設定すると、**/etc/insights-client/insights-client.conf** ファイル内の **ansible_host** 設定が、選択した Ansible ホスト名に変更されます。システムがすでに Red Hat Insights に接続されている場合、このパラメーターによって既存の Ansible ホスト名が更新されます。

Jira:RHEL-16975

ForwardToSyslog フラグが journald システムロールでサポートされるようになりました

journald RHEL システムロールでは、**journald_forward_to_syslog** 変数は、受信したメッセージを従来の **syslog** デーモンに転送するかどうかを制御します。この変数のデフォルト値は **false** です。この機能拡張により、インベントリーで **journald_forward_to_syslog** を **true** に設定することで、**ForwardToSyslog** フラグを設定できるようになりました。その結果、Splunk などのリモートロギングシステムを使用する場合、ログは **/var/log** ファイルで利用できるようになります。

Jira:RHEL-21123

ratelimit_burst 変数は、logging システムロールで ratelimit_interval が設定されている場合にのみ使用されます。

以前は、**logging** RHEL システムロールで、**ratelimit_interval** 変数が設定されていない場合、ロールは **ratelimit_burst** 変数を使用して rsyslog **ratelimit.burst** を設定していました。しかし、**ratelimit_interval** も設定する必要があるため、効果はありませんでした。

この機能拡張では、**ratelimit_interval** が設定されていない場合、ロールは **ratelimit.burst** を設定しません。**ratelimit.burst** を設定する場合は、**ratelimit_interval** 変数および **ratelimit_burst** 変数の両方を設定する必要があります。

Jira:RHEL-19047

logging システムロールでは、rsyslog_max_message_size の代わりに、logging_max_message_size パラメーターを使用します。

以前は、**rsyslog_max_message_size** パラメーターがサポートされていなかったにもかかわらず、**logging** RHEL システムロールは、**logging_max_message_size** パラメーターではなく **rsyslog_max_message_size** を使用していました。この機能拡張により、ログメッセージの最大サイズを設定するために **rsyslog_max_message_size** ではなく、**logging_max_message_size** が使用されるようになります。

Jira:RHEL-15038

ad_integration RHEL システムロールがカスタム SSSD 設定をサポートするようになる

以前は、**ad_integration** RHEL システムロールを使用する場合、ロールを使用して **sssd.conf** ファイルの **[sssd]** セクションにカスタム設定を追加することはできませんでした。この機能拡張により、**ad_integration** ロールは **sssd.conf** ファイルを変更できるようになり、結果としてカスタム SSSD 設定を使用できるようになります。

Jira:RHEL-21134

ad_integration RHEL システムロールは、カスタム SSSD ドメイン設定をサポートするようになりました。

以前は、**ad_integration** RHEL システムロールを使用する場合、ロールを使用して **sssd.conf** ファイルのドメイン設定セクションにカスタム設定を追加することはできませんでした。この機能拡張により、**ad_integration** ロールは **sssd.conf** ファイルを変更できるようになり、結果としてカスタム SSSD 設定を使用できるようになります。

Jira:RHEL-17667

logging RHEL システムロール用の新しい logging_preserve_fqdn 変数

以前は、**logging** システムロールを使用して完全修飾ドメイン名 (FQDN) を設定することはできませんでした。この更新により、オプションの **logging_preserve_fqdn** 変数が追加されています。これを使用すると、**rsyslog** の **preserveFQDN** 設定オプションを設定して、syslog エントリーで短い名前ではなく完全な FQDN を使用できます。

Jira:RHEL-15933

ファイルシステムを作成せずにボリュームを作成する機能のサポート

この機能拡張により、**fs_type=unformatted** オプションを指定することで、ファイルシステムを作成せずに新しいボリュームを作成できるようになりました。

同様に、セーフモードが無効になっていることを確認したうえで、同じ方法を使用することで既存のファイルシステムを削除できます。

Jira:RHEL-16213

rhc システムロールが RHEL 7 システムをサポートするようになる

rhc システムロールを使用して RHEL 7 システムを管理できるようになりました。RHEL 7 システムを Red Hat Subscription Management (RHSM) および Insights に登録し、**rhc** システムロールを使用してシステムの管理を開始します。

現在 RHEL 7 では Insights Remediation 機能を利用できないため、**rhc_insights.remediation** パラメーターを使用しても RHEL 7 システムには影響しません。

Jira:RHEL-16977

新しい mssql_ha_prep_for_pacemaker 変数

以前は、**microsoft.sql.server** RHEL システムロールには、Pacemaker 用に SQL Server を設定するかどうかを制御する変数がありませんでした。この更新では、**mssql_ha_prep_for_pacemaker** が追加されます。システムを Pacemaker 用に設定せず、別の HA ソリューションを使用する場合は、変数を **false** に設定します。

Jira:RHEL-19204

sshd ロールは証明書ベースの SSH 認証を設定するようになりました

sshd RHEL システムロールを使用すると、SSH 証明書を使用して認証する複数の SSH サーバーを設定および管理できるようになりました。これにより、証明書は信頼できる CA によって署名され、きめ細かいアクセス制御、有効期限、集中型管理が提供されるため、SSH 認証がより安全になります。

Jira:RHEL-5985

selinux ロールは、無効モードでの SELinux の設定をサポートするようになりました。

この更新により、**selinux** RHEL システムロールは、SELinux が無効に設定されているノード上の SELinux ポート、ファイルコンテキスト、およびブールマッピングの設定をサポートするようになりました。これは、システム上で SELinux を許可モードまたは強制モードに有効にする前の設定シナリオに役立ちます。

Jira:RHEL-15871

selinux ロールは、存在しないモジュールを指定するとメッセージを出力するようになりました。

このリリースでは、**selinux_modules.path** 変数に存在しないモジュールを指定すると、**selinux** RHEL システムロールによってエラーメッセージが出力されます。

Jira:RHEL-19044

4.14. 仮想化

RHEL は仮想マシンのマルチ FD 移行をサポートするようになりました

この更新により、仮想マシンの複数のファイル記述子 (マルチ FD) の移行がサポートされるようになりました。マルチ FD 移行では、複数の並列接続を使用して仮想マシンを移行するため、利用可能なネットワーク帯域幅をすべて活用してプロセスを高速化できます。

この機能は高速ネットワーク (20 Gbps 以上) で使用することを推奨します。

Jira:RHELDOCS-16970[1]

Secure Execution VMs on IBM Z が暗号化コプロセッサーをサポートするようになりました。

この更新により、IBM Secure Execution on IBM Z を使用して、暗号化コプロセッサーを仲介デバイスとして仮想マシン (VM) に割り当てることができるようになりました。

暗号化コプロセッサーを仲介デバイスとして Secure Execution 仮想マシンに割り当てることで、仮想マシンのセキュリティーを損なうことなくハードウェア暗号化を使用できるようになります。

Jira:RHEL-11597[1]

Web コンソールで SPICE を VNC に置き換えることができるようになる

この更新により、Web コンソールを使用して、既存の仮想マシン (VM) で SPICE リモート表示プロトコルを VNC プロトコルに置き換えることができるようになりました。

SPICE プロトコルのサポートは RHEL 8 では非推奨となり、RHEL 9 では削除されるため、SPICE プロトコルを使用する仮想マシンは RHEL 9 に移行できません。ただし、RHEL 8 仮想マシンはデフォルトで SPICE を使用するため、移行を成功させるには SPICE から VNC に切り替える必要があります。

Jira:RHELDOCS-18289[1]

RHEL Web コンソールの新しい仮想化機能

今回の更新で、RHEL Web コンソールに Virtual Machines ページに新機能が追加されました。以下を実行することができます。

- 仮想マシン (VM) の作成中に SSH 公開鍵を追加します。この公開鍵は、新しく作成された仮想マシン上の指定された root 以外のユーザーの **~/.ssh/authorized_keys** ファイルに保存され、指定されたユーザーアカウントへの即時 SSH アクセスが提供されます。
- 新しいストレージプールを作成するときに、**pre-formatted block device** タイプを選択します。これは、raw ディスクデバイスの意図しない再フォーマットを防ぐため、**physical disk device** タイプよりも堅牢な代替手段です。

この更新により、仮想マシンページのいくつかのデフォルトの動作も変更されます。

- **Add disk** ダイアログでは、**Always attach** オプションがデフォルトで設定されるようになりました。

Jira:RHELDOCS-18323[1]

4.15. クラウド環境の RHEL

生成された設定ファイルを削除するための新しい cloud-init clean オプション

cloud-init ユーティリティーに **cloud-init clean --configs** オプションが追加されました。このオプションを使用すると、インスタンス上の **cloud-init** によって生成された不要な設定ファイルを削除できます。たとえば、ネットワークのセットアップを定義する **cloud-init** 設定ファイルを削除するには、次のコマンドを使用します。

```
cloud-init clean --configs network
```

Jira:RHEL-7312[1]

EC2 上の RHEL インスタンスが IPv6 IMDS 接続をサポートするようになる

この更新により、Amazon Elastic Cloud コンピュート (EC2) 上の RHEL 8 および 9 インスタンスは IPv6 プロトコルを使用してインスタンスメタデータサービス (IMDS) に接続できるようになります。その結果、EC2 上の **cloud-init** を使用して、デュアルスタック IPv4 および IPv6 接続を備えた RHEL インスタンスを設定できるようになります。さらに、IPv6 のみのサブネットで **cloud-init** を使用して RHEL の EC2 インスタンスを起動することもできます。

Jira:RHEL-7278

4.16. コンテナー

Container Tools パッケージが更新される

Podman、Buildah、Skopeo、crun、runc ツールを含む、更新された Container Tools パッケージが利用可能になりました。以前のバージョンに対する主なバグ修正および機能拡張は、以下のとおりです。

Podman v4.9 の主な変更点:

- **podman --module <your_module_name>** コマンドを使用して Podman でモジュールをオンデマンドでロードし、システム設定ファイルおよびユーザー設定ファイルをオーバーライドできるようになりました。
- **create**、**set**、**remove**、**update** サブコマンドのセットを備えた新しい **podman farm** コマンドが追加されました。これらのコマンドを使用すると、さまざまなアーキテクチャーの podman を実行しているマシンにビルドを委託できます。
- Docker compose などの外部 compose プロバイダーを使用して Compose ワークロードを実行する、新しい **podman-compose** コマンドが追加されました。
- **podman build** コマンドは、**--layer-label** および **--cw** オプションをサポートするようになりました。
- **podman generate systemd** コマンドは非推奨になりました。Quadlet を使用して、**systemd** でコンテナーと Pod を実行してください。
- **podman build** コマンドは、HereDoc 構文を使用した **Containerfiles** をサポートするようになりました。
- **podman machine init** および **podman machine set** コマンドは、新しい **--usb** オプションをサポートするようになりました。QEMU プロバイダーの USB パススルーを許可するには、このオプションを使用します。
- **podman kube play** コマンドは、新しい **--publish-all** オプションをサポートするようになりました。このオプションを使用して、ホスト上のすべての containerPorts を公開します。

注目すべき変更の詳細は、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。

Jira:RHELPLAN-167794[1]

Podman が containers.conf モジュールをサポートするようになりました

Podman モジュールを使用して、事前に定義された設定セットをロードできます。Podman モジュールは、Tom's Obvious Minimal Language (TOML) 形式の **containers.conf** ファイルです。

このモジュールは、次のディレクトリーまたはそのサブディレクトリーにあります。

- rootless ユーザーの場合: **$HOME/.config/containers/containers.conf.modules**
- root ユーザーの場合: **/etc/containers/containers.conf.modules**、または **/usr/share/containers/containers.conf.modules**

podman --module <your_module_name> コマンドを使用してオンデマンドでモジュールをロードし、システム設定ファイルおよびユーザー設定ファイルをオーバーライドできます。モジュールを操作する際には、次の点に留意してください。

- **--module** オプションを使用して、モジュールを複数回指定できます。
- **<your_module_name>** が絶対パスの場合、設定ファイルは直接ロードされます。

- 相対パスは、前述の 3 つのモジュールディレクトリーを基準にして解決されます。
- **$HOME** 内のモジュールは、**/etc/** および **/usr/share/** ディレクトリー内のモジュールをオーバーライドします。

詳細は、アップストリームのドキュメント を参照してください。

Jira:RHELPLAN-167830[1]

Podman v4.9 RESTful API が進捗のデータを表示するようになりました

この機能拡張により、Podman v4.9 RESTful API では、イメージをレジストリーにプルまたはプッシュするときに、進捗データが表示されるようになりました。

Jira:RHELPLAN-167822[1]

SQLite は、Podman のデフォルトのデータベースバックエンドとして完全にサポートされるようになりました

Podman v4.9 では、以前テクノロジープレビューとして利用可能だった Podman の SQLite データベースバックエンドが完全にサポートされるようになりました。SQLite データベースは、コンテナーメタデータを操作する際に、より優れた安定性、パフォーマンス、一貫性を提供します。SQLite データベースバックエンドは、RHEL 8.10 の新規インストールのデフォルトバックエンドです。以前の RHEL バージョンからアップグレードする場合、デフォルトバックエンドは BoltDB になります。

containers.conf ファイルの **database_backend** オプションを使用してデータベースバックエンドを明示的に設定した場合、Podman は指定されたバックエンドを引き続き使用します。

Jira:RHELPLAN-168179[1]

管理者は nftables を使用してファイアウォールルールの分離を設定できます。

iptables がインストールされていないシステムでも、Podman コンテナーネットワークスタックである Netavark を使用できます。以前は、Netavark の前身であるコンテナーネットワーキングインターフェイス (CNI) ネットワーキングを使用すると、**iptables** がインストールされていないシステムでコンテナーネットワーキングを設定する方法はありませんでした。この機能拡張により、**nftables** のみがインストールされているシステムでも Netavark ネットワークスタックが動作するようになり、自動的に生成されたファイアウォールルールの分離が改善されます。

Jira:RHELDOCS-16955[1]

Containerfile が複数行の命令をサポートするようになりました

Containerfile ファイルで複数行の HereDoc 命令 (Here Document 表記) を使用すると、このファイルを簡素化し、複数の **RUN** ディレクティブを実行することで発生するイメージレイヤーの数を減らすことができます。

たとえば、元の **Containerfile** が次の **RUN** ディレクティブを含むとします。

```
RUN dnf update
RUN dnf -y install golang
RUN dnf -y install java
```

複数の RUN ディレクティブの代わりに、HereDoc 表記を使用できます。

```
RUN <<EOF
```

```
dnf update
dnf -y install golang
dnf -y install java
EOF
```

Jira:RHELPLAN-168184[1]

Toolbx が利用可能になる

Toolbx を使用すると、ベースオペレーティングシステムに影響を与えることなく、開発およびデバッグのツール、エディター、ソフトウェア開発キット (SDK) を、Toolbx の完全にミュータブルなコンテナーにインストールできます。Toolbx コンテナーは、**registry.access.redhat.com/ubi8.10/toolbox:latest** イメージをベースとしています。

Jira:RHELDOCS-16241[1]

第5章 利用可能な BPF 機能

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

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

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

表5.1 システム設定とその他のオプション

オプション	値
unprivileged_bpf_disabled	1 (特権ユーザーに限定された bpf() syscall、リカバリーなし)
JIT コンパイラー	1 (有効)
JIT コンパイラーの強化	1 (権限のないユーザーに対して有効)
JIT コンパイラー kallsyms エクスポート	1 (ルートで有効)
非特権ユーザーの JIT のメモリー制限	528482304
CONFIG_BPF	y
CONFIG_BPF_SYSCALL	y
CONFIG_HAVE_EBPF_JIT	y
CONFIG_BPF_JIT	y
CONFIG_BPF_JIT_ALWAYS_ON	y
CONFIG_DEBUG_INFO_BTF	y
CONFIG_DEBUG_INFO_BTF_MODULES	n
CONFIG_CGROUPS	y
CONFIG_CGROUP_BPF	y
CONFIG_CGROUP_NET_CLASSID	y
CONFIG_SOCK_CGROUP_DATA	y

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

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

表5.2 利用可能なプログラムの種類とサポートされているヘルパー

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

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

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

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

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

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

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

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

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

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

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

表5.3 利用可能なマップの種類

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

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

第6章 バグ修正

ここでは、ユーザーに重大な影響を与えるバグで、Red Hat Enterprise Linux 8.10 で修正されたものを説明します。

6.1. インストーラーおよびイメージの作成

インストーラーはキックスタートファイルで追加のタイムゾーン定義を受け入れるようになりました

Anaconda は、タイムゾーンの選択を検証する方法を、より制限的な別の方法に切り替えました。このため、日本などの一部のタイムゾーン定義は、以前のバージョンでは受け入れられていたにもかかわらず、有効ではなくなりました。これらの定義を含むレガシーキックスタートファイルを更新する必要がありました。更新しない場合、デフォルトで **Americas/New_York time** ゾーンになっていました。

有効なタイムゾーンのリストは、以前は **pytz** Python ライブラリーの **pytz.common_timezones** から取得されていました。この更新により、**timezone** キックスタートコマンドの検証設定が **pytz.all_timezones** を使用するように変更されます。これは **common_timezones** リストのスーパーセットであり、より多くのタイムゾーンを指定できるようになります。この変更により、Red Hat Enterprise Linux 6 用に作成された古いキックスタートファイルでも引き続き有効なタイムゾーンが指定されるようになります。

注記: この変更は **timezone** キックスタートコマンドにのみ適用されます。グラフィカルおよびテキストベースの対話型インターフェイスでのタイムゾーンの選択は変更されません。有効なタイムゾーンが選択されている Red Hat Enterprise Linux 8 の既存のキックスタートファイルは、更新する必要がありません。

Jira:RHEL-13151[1]

6.2. ネットワーク

NetworkManager で、VPN 接続プロファイルにおける CVE-2024-3661 (TunnelVision) の影響を軽減できるようになる

VPN 接続は、ルートを利用してトンネルを介してトラフィックをリダイレクトします。ただし、DHCP サーバーがクラスレススタティックルートオプション (121) を使用してクライアントのルーティングテーブルにルートを追加し、DHCP サーバーによって伝播されたルートが VPN と重複する場合、トラフィックは VPN ではなく物理インターフェイスを介して送信される場合があります。この脆弱性は CVE-2024-3661 で説明されており、TunnelVision とも呼ばれています。結果として、VPN によって保護されているはずのトラフィックに攻撃者がアクセスできるようになります。

RHEL では、この問題は LibreSwan IPSec および WireGuard VPN 接続に影響します。影響を受けないのは、**ipsec-interface** プロパティーと **vt-interface** プロパティーの両方が未定義または **no** に設定されているプロファイルを持つ LibreSwan IPSec 接続だけです。

CVE-2024-3661 ドキュメントでは、VPN ルートを優先度の高い専用ルーティングテーブルに配置するように VPN 接続プロファイルを設定することで、TunnelVision の影響を軽減する手順について説明しています。この手順は、LibreSwan IPSec 接続と WireGuard 接続の両方で機能します。ただし、LibreSwan IPSec 接続プロファイルに緩和手順を適用するには、NetworkManager 1.40.16-18 以降を使用する必要があります。RHEL 8.10 では、このバージョンは RHSA-2025:0288 アドバイザリーによって提供されます。

Jira:RHEL-73052

6.3. セキュリティー

SELinux ポリシーに、ip vrf による仮想ルーティングを管理するためのルールが追加されました

ip vrf コマンドを使用して、他のネットワークサービスの仮想ルーティングを管理できます。以前は、**selinux-policy** にはこの用途をサポートするルールは含まれていませんでした。この更新により、SELinux ポリシールールによって、**ip** ドメインから **httpd**、**sshd**、および **named** ドメインへの明示的な遷移が可能になります。これらの遷移は、**ip** コマンドが **setexeccon** ライブラリー呼び出しを使用する場合に適用されます。

Jira:RHEL-9981[1]

SELinux ポリシーにより、staff_r 制限ユーザーが sudo crontab を実行できるようになります。

以前は、SELinux ポリシーには、制限されたユーザーが **sudo crontab** コマンドを実行できるようにするルールは含まれていませんでした。その結果、**staff_r** ロールの制限されたユーザーは、**sudo crontab** を使用して他のユーザーの **crontab** スケジュールを編集することができませんでした。この更新により、ポリシーにルールが追加され、**staff_r** ユーザーは **sudo crontab** を使用して他のユーザーの **crontab** スケジュールを編集できるようになります。

Jira:RHEL-1388

SELinux ポリシーには追加のサービスとアプリケーションに関するルールが含まれています

selinux-policy パッケージのこのバージョンには追加のルールが含まれています。特に注目すべきは、**sysadm_r** ロールのユーザーは次のコマンドを入力できることです。

- **sudo traceroute**
- **sudo tcpdump**
- **sudo dnf**

Jira:RHEL-15398、Jira:RHEL-1679、Jira:RHEL-9947

unconfined_login が off に設定されている場合、SELinux ポリシーは制限のないユーザーの SSH ログインを拒否します。

以前は、**unconfined_login** ブール値が **off** に設定されている場合に、制限のないユーザーが SSH 経由でログインすることを拒否するルールが SELinux ポリシーにありませんでした。その結果、**unconfined_login** が **off** に設定されている場合でも、ユーザーは SSHD を使用して制限のないドメインとしてログインできます。この更新により、SELinux ポリシーにルールが追加され、その結果、**unconfined_login** が **off** の場合、ユーザーは **sshd** 経由で制限なしでログインできなくなります。

Jira:RHEL-1628

rsyslogd による制限されたコマンドの入力が SELinux ポリシーで許可されるようになりました。

以前は、SELinux ポリシーに、**rsyslogd** デーモンが SELinux で制限されたコマンド (**systemctl** など) の入力を許可するルールがありませんでした。その結果、**omprog** ディレクティブの引数として実行されたコマンドは失敗しました。この更新により、SELinux ポリシーにルールが追加され、**omprog** の引

数として実行される **/usr/libexec/rsyslog** ディレクトリー内の実行可能ファイルが **syslogd_unconfined_script_t** の制限のないドメインに含まれるようになります。その結果、**omprog** の引数として実行されたコマンドは正常に終了します。

Jira:RHEL-10087

大きな SSHD 設定ファイルによってログインが妨げられなくなる

以前は、SSHD 設定ファイルが 256 KB より大きいと、システムにログインするときにエラーが発生していました。その結果、リモートシステムにアクセスできなくなりました。この更新によりファイルサイズの制限が削除され、SSHD 設定ファイルが 256 KB より大きい場合でもユーザーはシステムにログインできるようになります。

Jira:RHEL-5279

6.4. ソフトウェア管理

yum needs-restarting --reboothint コマンドは、CPU マイクロコードを更新するために再起動を推奨するようになりました

CPU マイクロコードを完全に更新するには、システムを再起動する必要があります。以前は、更新された CPU マイクロコードを含む **microcode_ctl** パッケージをインストールしたときに、**yum needs-restarting --reboothint** コマンドは再起動を推奨しませんでした。この更新により、この問題は修正され、**yum needs-restarting --reboothint** は CPU マイクロコードを更新するために再起動を推奨するようになりました。

Jira:RHEL-17356

systemd は librepo が作成した /run/user/0 ディレクトリーを正しく管理するようになりました

以前は、root でログインする前に Insights クライアントから **librepo** 関数が呼び出されると、**/run/user/0** ディレクトリーが間違った SELinux コンテキストタイプで作成される場合がありました。これにより、root からのログアウト後に **systemd** がディレクトリーをクリーンアップすることができませんでした。

この更新により、**librepo** パッケージは、SELinux ポリシーで定義されたデフォルトのファイルシステムのラベル付けルールに従って、デフォルトの作成タイプを設定するようになりました。その結果、**systemd** は **librepo** が作成した **/run/user/0** ディレクトリーを正しく管理するようになりました。

Jira:RHEL-10720

systemd は libdnf が作成した /run/user/0 ディレクトリーを正しく管理するようになりました

以前は、root でログインする前に Insights クライアントから **libdnf** 関数が呼び出されると、**/run/user/0** ディレクトリーが間違った SELinux コンテキストタイプで作成される場合がありました。これにより、root からのログアウト後に **systemd** がディレクトリーをクリーンアップすることができませんでした。

この更新により、**libdnf** パッケージは、SELinux ポリシーで定義されたデフォルトのファイルシステムのラベル付けルールに従って、デフォルトの作成タイプを設定するようになりました。その結果、**systemd** は **libdnf** が作成した **/run/user/0** ディレクトリーを正しく管理するようになりました。

Jira:RHEL-6421

6.5. シェルおよびコマンドラインツール

BIOS と UEFI の両方のブートローダーがインストールされている場合に ReaR が BIOS ブートローダーの存在を確認するようになりました

以前は、ハイブリッドブートローダーセットアップ (UEFI と BIOS) で、UEFI を使用して起動すると、Relax-and-Recover (ReaR) によって UEFI ブートローダーのみが復元され、BIOS ブートローダーは復元されませんでした。これにより、システムに **GUID Partition Table** (GPT) と BIOS ブートパーティションは存在するものの、BIOS ブートローダーは存在しない状態が発生していました。この状況では、ReaR はレスキューイメージの作成に失敗し、**rear mkbackup** または **rear mkrescue** コマンドを使用してバックアップまたはレスキューイメージを作成しようとすると、次のエラーメッセージが表示されて失敗していました。

```
ERROR: Cannot autodetect what is used as boot loader, see default.conf about 'BOOTLOADER'.
```

この更新により、ReaR が UEFI と BIOS の両ブートローダーの存在を確認してそれらを復元するようになり、**GPT** の BIOS ブートパーティションを持つシステムで BIOS ブートローダーが検出されない場合でも失敗しなくなりました。その結果、UEFI と BIOS のハイブリッドブートローダーセットアップを備えたシステムで、複数回のバックアップおよび復元が可能になりました。

Jira:RHEL-24729[1]

ReaR はリカバリー中に logbsize、sunit、swidth マウントオプションを使用しなくなりました

以前は、**MKFS_XFS_OPTIONS** 設定を使用して、元のパラメーターとは異なるパラメーターで **XFS** ファイルシステムを復元する場合、Relax-and-Recover (ReaR) は、元のファイルシステムには適用できるものの復元されたファイルシステムには適用できないマウントオプションを使用して、このファイルシステムをマウントしていました。その結果、ReaR が **mount** コマンドを実行したときに、ディスクレイアウトの再作成が失敗し、次のエラーメッセージが表示されていました。

```
wrong fs type, bad option, bad superblock on and missing codepage or helper program, or other error.
```

カーネルログには、次のいずれかのメッセージが表示されていました。

```
logbuf size must be greater than or equal to log stripe size
```

```
alignment check failed: sunit/swidth vs. agsize
```

この更新により、ReaR は再作成された **XFS** ファイルシステムをマウントするときに、**logbsize**、**sunit**、および **swidth** マウントオプションを使用しなくなります。その結果、**MKFS_XFS_OPTIONS** 設定を使用すると、ディスクレイアウトの再作成が成功します。

Jira:RHEL-17354[1]

シンプールのメタデータサイズが小さいシステムで ReaR リカバリーが失敗しなくなりました

以前は、ReaR は、シンプールを含む LVM ボリュームグループのレイアウトを保存するときに、プールメタデータボリュームのサイズを保存していませんでした。システムがデフォルト以外のプールメタデータサイズを使用していた場合でも、ReaR はリカバリー時にデフォルトサイズでプールを再作成していました。

その結果、元のプールメタデータサイズがデフォルトサイズよりも小さく、ボリュームグループに空き領域がない場合、システムのリカバリー時にレイアウトの再作成が失敗し、次の例のようなメッセージがログに記録されていました。

> Insufficient free space: 230210 extents needed, but only 230026 available

または、以下を実行します。

> Volume group "vg" has insufficient free space (16219 extents): 16226 required.

この更新により、リカバリー後のシステムが、元のシステムと同サイズのメタデータボリュームを含むようになります。その結果、シンプールのメタデータサイズが小さく、ボリュームグループに追加の空き領域がないシステムのリカバリーが、正常に完了します。

Jira:RHEL-17353[1]

pkla-compact バイナリーは、polkit が logind-session-monitor イベントで呼び出されたときに実行されます。

以前は、polkit アクションの承認の再検証は、すべてのユーザーの **logind-session-monitor** イベントによってトリガーされていました。各 **CheckAuthorization** 要求は、システムにそのようなファイルが存在しない場合でも、**polkit-pkla-compat** バイナリーを実行してレガシー **.pkla** 設定ファイルをチェックします。これにより、polkit デーモンによる CPU 使用率が増加します。

現在、polkit アクションに関連する **logind-session** の変更のみが監視されます。セッションの状態が変化すると、セッションに関連付けられた polkit オブジェクトが再検証 (**CheckAuthorization**) をトリガーします。更新を成功させるには、gnome-shell (**log out to login screen and re-login** または **reboot**) を再起動する必要があります。

polkit-pkla-compat バイナリーはソフト依存関係になりました。その結果、**/etc/polkit-1/localauthority**、**/etc/polkit-1/localauthority.conf.d**、**/var/lib/polkit-1/localauthority** および各サブディレクトリーに **.pkla** ファイルが存在しない場合にのみ、**polkit-pkla-compat** バイナリーをアンインストールすることで CPU の負荷を軽減できます。

Jira:RHEL-34022[1]

6.6. カーネル

crash がバージョン 8.0.4 にリベース

crash ユーティリティーはバージョン 8.0.4 にアップグレードされ、複数のバグが修正されました。以下は、主な修正です。

- カーネルパニック中にパニックが発生していない CPU が停止に失敗した場合のセグメンテーション違反を修正しました。
- **panic_on_oops** カーネルパラメーターが無効になっている場合にカーネルがパニックにならないように誤って防止する重大なエラーを修正しました。
- **CONFIG_SLAB_FREELIST_HARDENED=y** 設定オプションでコンパイルされたカーネルのハッシュされたフリーリストポインターを解決する **crash** ユーティリティーを修正しました。
- カーネルモジュールメモリーレイアウトの用語が変更され、**crash** ユーティリティーのメモリー関連の側面をより適切に示すために、**module_layout** が **module_memory** に置き換えられました。この変更の前は、**crash** ユーティリティーはセッションを開始できず、次のようなエラーメッセージを返していました。

```
crash: invalid structure member offset: module_core_size
        FILE: kernel.c LINE: 3787 FUNCTION: module_init()
```

Jira:RHEL-9010

tuna は、必要に応じて GUI を起動します

以前は、サブコマンドなしで **tuna** ユーティリティーを実行すると、GUI が起動していました。ディスプレイがある場合、この動作は望ましいものでした。逆の場合、ディスプレイのないマシンの **tuna** は適切に終了しませんでした。この更新により、**tuna** はディスプレイがあるかどうかを検出し、それに応じて GUI を起動するどうかを決定するようになりました。

Jira:RHEL-19179[1]

6.7. ファイルシステムおよびストレージ

Multipathd はデバイスが誤って I/O をキューイングしていないかチェックするようになりました

以前は、次の条件下では、マルチパスデバイスが (失敗するように設定されていたにもかかわらず) I/O のキューイングを再開していました。

- マルチパスデバイスの **queue_if_no_paths** パラメーターが複数回の再試行に設定されていた。
- 動作中のパスがなく、I/O をキューイングしなくなったパスデバイスが、マルチパスデバイスから削除された。

今回の更新により、問題が修正されました。その結果、キューイングが無効化され、使用可能なパスがない状態でパスが削除された場合、マルチパスデバイスは I/O のキューイングを再開しなくなりました。

Jira:RHEL-16563[1]

dm-crypt および dm-verity デバイスの no_read_workqueue、no_write_workqueue、try_verify_in_taskle オプションは一時的に無効になっています

以前は、**no_read_workqueue** または **no_write_workqueue** オプションを使用して作成された **dm-crypt** デバイスと、**try_verify_in_tasklet** オプションを使用して作成された **dm-verity** デバイスによってメモリー破損が発生していました。その結果、ランダムにカーネルメモリーが破損し、さまざまなシステムの問題が発生しました。この更新により、これらのオプションは一時的に無効になります。この修正により、一部のワークロードで **dm-verity** と **dm-crypt** のパフォーマンスが低下する可能性があることに注意してください。

Jira:RHEL-22232[1]

6.8. 高可用性およびクラスター

クローンリソースおよびバンドルリソースの移動および禁止に関する問題が修正されました

このバグ修正により、バンドルされたリソースとクローンリソースの移動に関する 2 つの制限が解消されます。

- ユーザーがバンドルされたリソースをバンドルから移動しようとしたり、バンドル内での実行

を禁止しようとしたりすると、**pcs** が制約を作成していました。しかし、その制約は効果がありませんでした。このため、エラーメッセージが表示されて移動が失敗していました。この修正により、**pcs** は、バンドルされたリソースのバンドルからの移動や、バンドルされたリソースの禁止を許可しなくなりました。また、バンドルされたリソースをバンドルから移動できないことを示すエラーメッセージを出力するようになりました。

- ユーザーがバンドルリソースまたはクローンリソースを移動しようとすると、バンドルリソースまたはクローンリソースを移動できないことを示すエラーメッセージが表示され、**pcs** が終了していました。この修正により、move コマンドの検証が緩和されます。クローンリソースおよびバンドルリソースを移動できるようになりました。クローンリソースを移動する場合、クローンのインスタンスが複数実行されている場合は、宛先ノードを指定する必要があります。移動できるのはレプリカが 1 つのバンドルのみです。

Jira:RHEL-7584

pcs status コマンドの出力に、期限切れの制約に関する警告が表示されなくなりました

以前は、クラスターリソースを移動すると一時的な場所の制約が作成され、制約の有効期限が切れた後でも **pcs status** コマンドによって警告が表示されていました。この修正により、**pcs status** コマンドは期限切れの制約を除外し、コマンド出力に警告メッセージが生成されなくなりました。

Jira:RHEL-7668

SBD フェンシングで必要な場合、auto_tie_breaker クォーラムオプションを無効にできなくなりました

以前は、SBD フェンシングが正しく機能するためにクラスター設定で **auto_tie_breaker** クォーラムオプションが必要な場合でも、**pcs** でこのオプションを無効にすることができました。この修正により、SBD フェンシングで **auto_tie_breaker** オプションを有効にする必要があるシステムで **auto_tie_breaker** を無効にしようとすると、**pcs** はエラーメッセージを生成するようになりました。

Jira:RHEL-7731

--force を指定しない tls および keep_active_partition_tie_breaker クォーラムデバイスオプションの設定

以前は、クォーラムデバイスを設定するときに、**--force** オプションを指定せずにクォーラムデバイスモデル **net** の **tls** および **keep_active_partition_tie_breaker** オプションを設定することができませんでした。この更新により、これらのオプションを設定する際に **--force** を指定する必要がなくなりました。

Jira:RHEL-7745

6.9. コンパイラーおよび開発ツール

システムアップグレードの中断後に ldconfig がクラッシュしなくなりました

以前は、システムアップグレードの中断後、**/usr/lib64** ディレクトリーに残された不完全な共有オブジェクトを処理する際に、**ldconfig** ユーティリティーがセグメンテーション違反で予期せず終了していました。この更新により、**ldconfig** はシステムのアップグレード中に書き込まれた一時ファイルを無視するようになりました。その結果、システムアップグレードの中断後に **ldconfig** がクラッシュしなくなりました。

Jira:RHEL-13720

依存関係のサイクルに関係する共有オブジェクトに dlclose を使用するアプリケーションとの

glibc 互換性が向上しました

以前は、**glibc** の **dlclose** 関数を使用して依存関係サイクル内の共有オブジェクトをアンロードすると、そのオブジェクトの ELF デストラクターが、他のすべてのオブジェクトがアンロードされるまで呼び出されない場合がありました。ELF デストラクターの実行が遅くなることで、アプリケーションでクラッシュやその他のエラーが発生しました。これは、最初にアンロードした共有オブジェクトの依存関係がすでに初期化解除されていることが原因でした。

この更新により、**glibc** が修正され、他の ELF デストラクターを起動する前に、アンロードする直近のオブジェクトの ELF デストラクターを呼び出すようになりました。その結果、依存関係サイクルに関係する共有オブジェクトに **dlclose** を使用するアプリケーションとの互換性が向上し、クラッシュが発生しなくなりました。

Jira:RHEL-10481[1]

glibc ワイド文字書き込みパフォーマンスの向上

以前は、**glibc** のワイド **stdio** ストリーム実装では、デフォルトのバッファーサイズがワイド文字の書き込み操作に十分な大きさであるとは扱われず、代わりに 16 バイトのフォールバックバッファーが使用されていたため、パフォーマンスに悪影響がありました。この更新により、バッファー管理が修正され、書き込みバッファー全体が使用されるようになりました。その結果、**glibc** ワイド文字書き込みパフォーマンスが向上します。

Jira:RHEL-19824[1]

glibc ダイナミックリンカーは、カスタム malloc 実装からの TLS アクセスを使用するアプリケーションによる再入可能 malloc 呼び出しを防止します。

一部のアプリケーションでは、初期実行 TLS の代わりにグローバル動的スレッドローカルストレージ (TLS) を使用するカスタム **malloc** 動的メモリー割り当て実装が提供されます。以前は、グローバル動的 TLS を使用するバンドルされた **malloc** 呼び出しを持つアプリケーションでは、アプリケーションの **malloc** サブシステムへの再入呼び出しが発生する可能性がありました。その結果、スタックの枯渇または内部データ構造の予期しない状態により、アプリケーションの **malloc** 呼び出しがクラッシュしました。

RHBA-2024:5834 アドバイザリーのリリースにより、**glibc** ダイナミックリンカーはカスタム **malloc** 実装からの TLS アクセスを検出するようになりました。**malloc** 呼び出し中に TLS アクセスが検出されると、TLS 処理中のそれ以降の呼び出しはスキップされ、再入可能な **malloc** 呼び出しは防止されます。

Jira:RHEL-39994

6.10. IDENTITY MANAGEMENT

certmonger が、非表示のレプリカ上の KDC 証明書を正しく更新するようになりました。

以前は、証明書の有効期限が近づいたときに、**certmonger** が非表示のレプリカ上の KDC 証明書の更新に失敗していました。これは、更新プロセスで非表示でないレプリカのみがアクティブな KDC として考慮されたために発生していました。この更新により、非表示のレプリカがアクティブな KDC として扱われ、**certmonger** がこれらのサーバー上で KDC 証明書を正常に更新するようになりました。

Jira:RHEL-45908

Automembership プラグインがデフォルトでグループをクリーンアップしなくなる

以前は、自動メンバー再構築タスクは最初にすべてのメンバーシップ値を削除し、次にメンバーシップを最初から再構築していました。その結果、特に他の **be_txn** プラグインが有効になっている場合は、再構築タスクのコストが高くなりました。

この更新により、Automembership プラグインには次の改善が加えられました。

- 一度に許可される再ビルドタスクは 1 つだけです。
- Automembership プラグインが、デフォルトでは以前のメンバーをクリーンアップしなくなりました。新しい CLI オプション **--cleanup** を使用して、最初から再ビルドする前にメンバーシップを意図的にクリーンアップします。

```
# dsconf slapd-instance_name plugins automember fixup -f objectclass=posixaccount -s sub --cleanup "ou=people,dc=example,dc=com"
```

- 修正の進行状況を表示するためのログ記録が改善されました。

Jira:RHEL-5390[1]

割り当てられたメモリーが操作完了後に解放されるようになりました

以前は、各操作に対して KCM によって割り当てられたメモリーは、接続が閉じるまで解放されませんでした。その結果、接続を開いて同じ接続で多くの操作を実行するクライアントアプリケーションでは、接続が閉じるまで割り当てられたメモリーが解放されないため、メモリーが著しく増加しました。この更新により、操作に割り当てられたメモリーは、操作が完了するとすぐに解放されるようになりました。

Jira:SSSD-7015

IdM クライアントは、信頼できる AD ユーザーの名前に大文字と小文字が混在している場合でも、当該 AD ユーザーの情報を適切に取得する

以前は、ユーザーの検索または認証を試行した際に、その信頼できる Active Directory (AD) ユーザーの名前に大文字と小文字が混在しており、かつ IdM でオーバーライドが設定されていた場合、エラーが返され、ユーザーは IdM リソースにアクセスできませんでした。

この更新により、大文字と小文字を区別する比較は、大文字と小文字を区別しない比較に置き換えられました。その結果、IdM クライアントは、ユーザー名に大文字と小文字が混在しており、IdM でオーバーライドが設定されている場合でも、AD の信頼済みドメインのユーザーを検索できるようになりました。

Jira:SSSD-6096

SSSD は、パスワード変更時に猶予ログインが残っていない場合にエラーを正しく返します

以前は、ユーザーの LDAP パスワードの有効期限が切れた場合、猶予ログインが残っていないためにユーザーの最初のバインドが失敗した後でも、SSSD はパスワードの変更を試みていました。しかし、ユーザーに返されたエラーには失敗の理由が示されていませんでした。この更新により、バインドが失敗した場合、パスワード変更要求がキャンセルされるようになりました。SSSD は、猶予ログインが残っていないために別の方法でパスワードを変更する必要があることを示すエラーメッセージを返します。

Jira:SSSD-6184

realm leave コマンドを使用したドメインからのシステム削除

以前は、**sssd.conf** ファイルの **ad_server** オプションに複数の名前が設定されている場合、**realm**

leave コマンドを実行すると解析エラーが発生し、システムがドメインから削除されませんでした。この更新により、**ad_server** オプションが適切に評価されるようになり、正しいドメインコントローラー名が使用され、システムがドメインから適切に削除されるようになりました。

Jira:SSSD-6081

KCM は正しい sssd.kcm.log ファイルにログを記録します

以前、**logrotate** は Kerberos Credential Manager (KCM) ログファイルを正しくローテーションしていましたが、KCM はログを誤って古いログファイル **sssd_kcm.log.1** に書き込んでいました。KCM を再起動した場合、正しいログファイルが使用されました。この更新により、**logrotate** が呼び出されるとログファイルがローテーションされ、KCM は **sssd_kcm.log** ファイルに正しくログを記録するようになりました。

Jira:SSSD-6652

realm leave --remove コマンドが認証情報を要求しなくなる

以前は、**realm** ユーティリティーは、**realm leave** 操作を実行するときに、有効な Kerberos チケットが利用可能かどうかを正しく確認していませんでした。その結果、有効な Kerberos チケットが利用可能であったにもかかわらず、ユーザーはパスワードの入力を求められました。この更新により、**realm** は有効な Kerberos チケットがあるかどうかを正しく検証するようになり、**realm leave --remove** コマンドを実行するときにユーザーにパスワードの入力を要求しなくなりました。

Jira:SSSD-6425

FIPS モードで IdM Vault 暗号化および復号化に失敗しなくなりました

以前は、IdM Vault は、デフォルトのパディングラッピングアルゴリズムとして OpenSSL RSA-PKCS1v15 を使用していました。しかし、RHEL の FIPS 認定モジュールはいずれも FIPS で承認されたアルゴリズムとして PKCS#1 v1.5 をサポートしていなかったため、IdM Vault は FIPS モードで失敗していました。この更新により、IdM Vault は、RSA-OAEP パディングラッピングアルゴリズムをフォールバックとしてサポートするようになりました。その結果、IdM Vault の暗号化と復号化が FIPS モードで正しく動作するようになりました。

Jira:RHEL-12153[1]

非 CA IdM レプリカのインストールは、サーバーアフィニティーが設定されている場合に失敗しなくなる

一部のシナリオでは、証明機関 (CA) なしで IdM レプリカをインストールすると、**CA_REJECTED** エラーが発生して失敗しました。この障害は、**certmonger** サービスが証明書を取得しようとしたために発生し、複雑なトポロジーに新しいレプリカを追加したときにレプリケーションの詳細が不完全になりました。

この更新により、IdM レプリカのインストールプロセスは、Kerberos 認証や IdM API および CA 要求などの必要なサービスを提供する特定の IdM サーバーに対して実行されるようになりました。これにより、新しいレプリカを追加するときに完全なレプリケーションの詳細が保証されます。

Jira:RHEL-4964

Kerberos キー配布センターバージョン 1.20 以降では、バージョン 1.18.2 以前を実行している KDC から生成されたチケットが処理されるようになりました。

以前は、Kerberos バージョン 1.20 以降を実行しているキー配布センター (KDC) とバージョン 1.18.2 以前を実行している KDC の間で互換性の問題が発生していました。その結果、Kerberos 1.20 以降を実行している KDC によって発行された証拠チケットが Kerberos 1.18.2 以前を実行している KDC に送信され

ると、古い KDC は **AD-SIGNTICKET** 属性をサポートしていなかったため、チケット交付サービス要求を拒否しました。

この更新により、特権属性証明書 (PAC) が存在する場合に **AD-SIGNTICKET** が不要になったため、KDC の以前のバージョンは、Kerberos 1.20 以降を実行している KDC によって生成された証拠チケットを受け入れるようになりました。

Jira:RHEL-10495

dirsrv ファイルの SELinux ラベル付けが DEBUG ログレベルに移動する

以前は、**dirsrv** ファイルの SELinux ラベル付けには **INFO** ログレベルがありました。この更新では、以前のバージョンと同じように、**dirsrv** ファイルに **DEBUG** ログレベルが使用されます。

Jira:RHEL-5143

バックエンドが関連する接尾辞なしで設定されている場合に、Directory Server がセグメンテーション違反を引き起こさなくなる

以前は、バックエンドが関連する接尾辞なしで設定されている場合、Directory Server の起動時にセグメンテーションエラーが発生していました。この更新により、Directory Server は接尾辞にアクセスする前に、接尾辞がバックエンドに関連付けられているかどうかを確認します。その結果、セグメンテーション違反が発生しなくなります。

Jira:RHEL-5107

ページ結果検索を中止した後に Directory Server が失敗しなくなりました

以前は、競合状態が、ページ化された結果の検索を中止する際にヒープ破損と Directory Server 障害の原因となっていました。この更新により、競合状態が修正され、Directory Server 障害は発生しなくなりました。

Jira:RHEL-16338

接続テーブルのサイズにカスタム値を設定した場合は、アップグレード後に Directory Server が正しく起動するようになる

以前は、接続テーブルサイズにカスタム値を設定し、**nsslapd-conntablesize** 属性が **dse.ldif** ファイルに存在する場合は、アップグレード後に Directory Server が起動しませんでした。このリリースでは、**dse.ldif** ファイルに **nsslapd-conntablesize** が存在する状態でアップグレード後、Directory Server が正しく起動します。

Jira:RHEL-14025

コンテンツ同期プラグインが動的に有効化されても Directory Server が失敗しなくなりました。

以前は、コンテンツ同期プラグインが動的に有効になっている場合は、操作前コールバックが登録されていないため、操作後プラグインコールバックでセグメンテーションエラーが発生していました。この更新により、操作後のプラグインコールバックによってメモリーが初期化されているかどうかが確認され、Directory Server に障害が発生しなくなりました。

Jira:RHEL-5135

mod_auth_gssapi がユーザーマッピングの GssapiLocalName 設定を反映するようになる

mod_auth_gssapi-1.6.1-9.el8 による更新の前は、リグレッションにより、**mod_auth_gssapi** が **krb5** 設定の **auth_to_local_names** 設定を使用してプリンシパルをローカルユーザー名に変換することが許

されていました。これは、この動作は明示的に有効化していなくても可能でした。その結果、**GssapiLocalName** オプションが **On** に設定されていない場合でも、**mod_auth_gssapi** を使用する **httpd** セットアップで、**auth_to_local_names** 設定に基づいてクライアントアイデンティティーが意図せずマッピングされる可能性がありました。この更新により、当初の動作が復元されます。**auth_to_local_names** によってユーザーマッピングを有効にするには、**httpd** 設定で **GssapiLocalName** を明示的に **On** に設定する必要があります。

Jira:RHELDOCS-19805

6.11. RED HAT ENTERPRISE LINUX システムロール

SBD の delay-start 値が高い場合に、クラスターの起動がタイムアウトしなくなりました

以前は、ユーザーが **ha_cluster** システムロールを使用してクラスターで SBD フェンシングを設定し、**delay-start** オプションを 90 秒に近い値または 90 秒を超える値に設定すると、クラスターの起動がタイムアウトしていました。これは、デフォルトの **systemd** 起動タイムアウトが 90 秒であり、システムが SBD の起動遅延値より前にこの起動タイムアウトに達していたためです。この修正により、**ha_cluster** システムロールが、**systemd** の **sbd.service** 起動タイムアウトを、**delay-start** よりも高い値でオーバーライドするようになりました。これにより、**delay-start** オプションの値が高い場合でも、システムを正常に起動できるようになります。

Jira:RHEL-4684[1]

network ロールが、0.0.0.0/0 または ::/0 のルーティングルールを検証する

以前は、ルーティングルールで **from:** または **to:** 設定が **0.0.0.0/0** または **::/0** アドレスに設定されていた場合、**network** RHEL システムロールはルーティングルールを設定できず、設定を無効として拒否していました。この更新により、**network** ロールでは、ルーティングルールの検証で **from:** と **to:** に **0.0.0.0/0** と **::/0** が許可されるようになりました。その結果、ロールは検証エラーを発生させることなくルーティングルールを正常に設定します。

Jira:RHEL-16501

ha_cluster システムロールが qnetd ホスト上のファイアウォールを正しく設定するようになりました

以前は、ユーザーが **qnetd** ホストを設定し、**ha_cluster** システムロールを使用して **ha_cluster_manage_firewall** 変数を **true** に設定しても、そのロールによってファイアウォールの高可用性サービスが有効になりませんでした。この修正により、**ha_cluster** システムロールは **qnetd** ホスト上のファイアウォールを正しく設定するようになりました。

Jira:RHEL-17874

keylime_server ロールが registrar サービスのステータスを正しく報告する

以前は、**keylime_server** ロール Playbook が誤った情報を提供すると、ロールは誤って開始が成功したと報告していました。この更新により、ロールは誤った情報が提供された場合に失敗を正しく報告するようになり、開いているポートを待機する際のタイムアウトが約 300 秒から約 30 秒に短縮されました。

Jira:RHEL-21946

postgresql RHEL システムロールが、正しいバージョンの PostgreSQL をインストールするようになりました

以前は、RHEL マネージドノードで **postgresql_version: "15"** 変数を定義して **postgresql** RHEL システムロールを実行しようとすると、PostgreSQL バージョン 15 ではなくバージョン 13 がインストールされていました。このバグは修正され、**postgresql** ロールは変数に設定されたバージョンをインストールするようになりました。

Jira:RHEL-21400

podman RHEL システムロールが、ルートレスコンテナーに対して linger を適切に設定およびキャンセルできるようになりました

以前は、**podman** RHEL システムロールは、ルートレスコンテナーに対して linger を適切に設定およびキャンセルしていませんでした。その結果、ルートレスユーザーのシークレットまたはコンテナーをデプロイすると、場合によってはエラーが発生し、場合によってはリソースを削除するときに linger をキャンセルできませんでした。この更新により、**podman** RHEL システムロールは、シークレットまたはコンテナーリソースの管理を行う前にルートレスユーザーに対して linger が有効になっていることを確認するようになりました。また、管理するシークレットまたはコンテナーリソースがなくなったときにルートレスユーザーに対して linger をキャンセルするようになりました。その結果、このロールはルートレスユーザーの残留を正しく管理します。

Jira:RHEL-22228

podman RHEL システムロールが、ルートレスコンテナーに対して linger を適切に設定およびキャンセルできるようになりました

以前は、**podman** RHEL システムロールは、ルートレスコンテナーに対して linger を適切に設定およびキャンセルしていませんでした。その結果、ルートレスユーザーのシークレットまたはコンテナーをデプロイすると、場合によってはエラーが発生し、場合によってはリソースを削除するときに linger をキャンセルできませんでした。この更新により、**podman** RHEL システムロールは、シークレットまたはコンテナーリソースの管理を行う前にルートレスユーザーに対して linger が有効になっていることを確認するようになりました。また、管理するシークレットまたはコンテナーリソースがなくなったときにルートレスユーザーに対して linger をキャンセルするようになりました。その結果、このロールはルートレスユーザーの残留を正しく管理します。

Jira:RHEL-22229

読み取りスケールクラスターの実行と mssql-server-ha のインストールに特定の変数が不要になる

以前は、**mssql** RHEL システムロールを使用して、特定の変数 (**mssql_ha_virtual_ip**、**mssql_ha_login**、**mssql_ha_login_password**、および **mssql_ha_cluster_run_role**) なしで読み取りスケールクラスターを設定すると、ロールは失敗し、**Variable not defined** というエラーメッセージが表示されました。ただし、これらの変数は読み取りスケールクラスターを実行するために必要ではありません。このロールは、読み取りスケールクラスターに必要のない **mssql-server-ha** のインストールも試みました。この修正により、これらの変数の要件は削除されました。その結果、読み取りスケールクラスターは、エラーメッセージなしで正常に実行されます。

Jira:RHEL-19202

kexec_crash_size ファイルがビジー状態のときでも、Kdump システムロールが正常に動作する

/sys/kernel/kexec_crash_size ファイルは、クラッシュカーネルメモリーに割り当てられたメモリーリージョンのサイズを提供します。

以前は、**/sys/kernel/kexec_crash_size** ファイルがビジー状態のときに Kdump システムロールが失敗していました。この更新により、システムロールはファイルが利用可能になるとファイルの読み取りを再試行します。その結果、ファイルがビジー状態のときにシステムロールが失敗しなくなりました。

Jira:RHEL-3354

selinux ロールが item ループ変数を使用しなくなりました

以前は、**selinux** RHEL システムロールは **item** ループ変数を使用していました。これにより、別のロールから **selinux** ロールを呼び出したときに、次の警告メッセージが表示される場合があります。

> [WARNING]: TASK: fedora.linux_system_roles.selinux : Restore SELinux labels on filesystem tree: The loop variable 'item' is already in use.
> You should set the `loop_var` value in the `loop_control` option for the task to something else to avoid variable collisions and unexpected behavior.

このリリースでは、**selinux** ロールは **__selinux_item** をループ変数として使用します。その結果、別のロールから **selinux** ロールを呼び出しても、**item** 変数がすでに使用されているという警告が表示されなくなりました。

Jira:RHEL-19042

秘密データは、詳細なロギングで記録されなくなりました。

以前は、シークレットデータを処理する一部のタスクでは、その内容がログに記録されていました。その結果、詳細なロギングが使用されていた場合、ログにはシークレットデータが表示されました。この更新では、シークレットデータをログに記録できるタスクに **no_log: true** ディレクティブが追加されます。その結果、シークレットデータは詳細なロギングでは記録されません。

Jira:RHEL-19242

ボリューム quadlet サービス名が失敗しなくなる

以前は、ボリュームサービス名を開始すると、次のようなエラーが発生しました。

> Could not find the requested service NAME.volume: host

この更新により、ボリュームクワドレットのサービス名が **basename-volume.service** に変更されます。その結果、ボリュームサービスはエラーなしで開始されます。

詳細は、ボリュームユニット の man ページを参照してください。

Jira:RHEL-21402

nbde_server ロールがソケットオーバーライドで動作するようになる

以前は、**nbde_server** RHEL システムロールは、**tangd** ソケットオーバーライドディレクトリー内のファイルがカスタムポート用の **override.conf** ファイルだけであると想定していました。その結果、ポートのカスタマイズがないと、ロールは他のファイルをチェックせずにディレクトリーを削除し、システムはその後の実行でディレクトリーを再作成しました。

このリリースでは、ポートオーバーライドファイルの属性の変更や他のファイルがある場合のディレクトリーの削除を防止するためにロールが修正されました。その結果、**tangd** ソケットオーバーライドファイルがロールの外部でも管理されていると、ロールは正しく機能します。

Jira:RHEL-25509

6.12. 仮想化

ダンプの失敗によって IBM Z VMs with Secure Execution がブロックされなくなりました。

以前は、IBM Z virtual machine (VM) with Secure Execution のダンプが失敗すると、仮想マシンは一時停止状態のままになり、実行がブロックされていました。たとえば、ディスクに十分なスペースがない場合、**virsh dump** コマンドを使用して仮想マシンをダンプすることは失敗します。

基礎となるコードが修正され、ダンプ失敗後も Secure Execution 仮想マシンは正常に操作を再開します。

Jira:RHEL-16696[1]

第7章 テクノロジープレビュー

ここでは、Red Hat Enterprise Linux 8.10 で利用可能なテクノロジープレビューのリストを提示します。

テクノロジープレビュー機能に対する Red Hat のサポート範囲の詳細は、テクノロジープレビュー機能のサポート範囲 を参照してください。

7.1. インフラストラクチャーサービス

TuneD 用のソケット API がテクノロジープレビューとして利用可能になる

UNIX ドメインソケットを通じて TuneD を制御するためのソケット 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。ソケット API は D-Bus API と 1 対 1 でマッピングされ、D-Bus が利用できない場合に代替通信方法を提供します。ソケット API を使用すると、TuneD デーモンを制御してパフォーマンスを最適化したり、さまざまなチューニングパラメーターの値を変更したりできます。ソケット API はデフォルトでは無効になっていますが、**tuned-main.conf** ファイルで有効にできます。

Bugzilla:2113900

7.2. ネットワーク

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。

Bugzilla:1633143[1]

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。

Bugzilla: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定します。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す)。
- 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 mirred egress redirect dev enp0s2
```

Bugzilla:1814836[1]、Bugzilla:1856415

act_mpls モジュー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。

act_mpls モジュールが、テクノロジープレビューとして **kernel-modules-extra** rpm で利用可能になりました。モジュールを使用すると、トラフィック制御 (TC) フィルターを使用した Multiprotocol Label Switching (MPLS) アクション (TC フィルターを使用した MPLS ラベルスタックエントリーの push や pop など) の適用が可能になります。また、このモジュールでは、Label、Traffic Class、Bottom of Stack、および Time to Live フィールドを独立して設定できます。

Bugzilla:1839311[1]

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ダー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。

Bugzilla:1906489

7.3. カーネル

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。

Bugzilla:1605216[1]

eBPF がテクノロジープレビューとして利用可能になりました。

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

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

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

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

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

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

Bugzilla:1559616[1]

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** 高速リブートでは、最初に基本入出力システム (BIOS) やファームウェアを経由せずに 2 番目のカーネルを直接ブートできるため、ブート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. 変更を有効にするために再起動します。

kexec 高速リブート機能は、RHEL 9 以降のリリースではサポート範囲が限定されていることに注意してください。

Bugzilla: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。

Bugzilla:1843266[1]

7.4. ファイルシステムおよびストレージ

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

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

Bugzilla:1627455[1]

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ーや 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ーエンジン (**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、コンテナーエンジングラフ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ー 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ー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ー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。

Bugzilla:1690207[1]

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、追加機能を備えたストレージプール上に管理されたファイルシステムを提供する、新しいローカルストレージマネージャーです。これはテクノロジープレビューとして提供されます。

Stratis を使用すると、次のストレージタスクを実行できます。

- 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、バックグランドサービス **stratisd** と通信する **stratis** ユーティリティーを使用します。詳細は、Stratis ファイルシステムのセットアップ ドキュメントを参照してください。

RHEL 8.5 は Stratis をバージョン 2.42 に更新した。詳細は、Stratis 2.4.2 リリースノート を参照してください。

Jira:RHELPLAN-1212[1]

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。

Bugzilla:1696451[1]

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** ユーティリティーは、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

ティリティーは、**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[1]

7.5. 高可用性およびクラスター

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

Pacemaker コンテナーバンドルは、テクノロジープレビューとして利用できるコンテナー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。

Bugzilla:1619620[1]

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、任意で定期的に実行さ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、クオ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。

Bugzilla: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、フェンシグを行うエージェントで **off** 操作を行う前に、ヒューリスティックエージェントで、この操作を行います。このヒューリスティックエージェントが **off** アクションに対して失敗する場合、このフェンシングレベルが成功しないのはすでに明らかです。そのため、Pacemaker フェンシングは、フェンシングを行うエージェントで **off** 操作を行うステップをスキップします。ヒューリスティックエージェントはこの動作を利用して、特定の条件下で、実際のフェンシングを行うエージェントがフェンシングできないようにできます。

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

Bugzilla:1775847[1]

7.6. IDENTITY MANAGEMENT

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

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

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

- 管理者は、管理しているクライアント以外のサーバーで、IdM の以前のバージョンもしくは最近のバージョンを使用できます。
- サーバーで IdM のバージョンを変更しても、開発者は特定バージョンの IdM コールを使用できます。

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

API の使用方法は 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。

Bugzilla:1664719

DNSSEC が IdM でテクノロジープレビューとして利用可能になる

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。

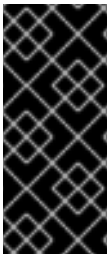
Bugzilla:1664718

ACME がテクノロジープレビューとして利用可能になる

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
```

Bugzilla:1628987[1]

sssd-idp サブパッケージがテクノロジープレビューとして利用可能

SSSD の **sssd-idp** サブパッケージには、Identity Management (IdM) サーバーに対して OAuth2 認証を実行するクライアント側のコンポーネントである **oidc_child** プラグインおよび krb5 **idp** プラグインが含まれます。この機能は、RHEL 8.7 以降の IdM サーバーのみで使用できます。

Bugzilla:2065692

SSSD の内部 krb5 idp プラグインがテクノロジープレビューとして利用可能

SSSD krb5 **idp** プラグインを使用すると、OAuth2 プロトコルを使用して外部アイデンティティープロバイダー (IdP) に対して認証できます。この機能は、RHEL 8.7 以降の IdM サーバーのみで使用できます。

Bugzilla:2056483

7.7. デスクトップ

64 ビット ARM アーキテクチャーの GNOME がテクノロジープレビューとして利用可能になる

GNOME デスクトップ環境は、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できます。

VNC を使用して 64 ビット ARM サーバーのデスクトップセッションに接続できるようになりました。その結果、グラフィカルアプリケーションを使用してサーバーを管理できます。

64 ビット ARM では、限定されたグラフィカルアプリケーションのセットを使用できます。以下に例を示します。

- Firefox Web ブラウザー
- Red Hat Subscription マネージャー (**subscription-manager-cockpit**)
- ファイアウォール設定 (**firewall-config**)
- ディスク使用状況アナライザー (**baobab**)

Firefox を使用して、サーバー上の Cockpit サービスに接続できます。

LibreOffice などの特定のアプリケーションは、コマンドラインインターフェイスのみを提供し、グラフィカルインターフェイスは無効になっています。

Jira:RHELPLAN-27394[1], Bugzilla:1667516、Bugzilla:1724302、Bugzilla:1667225

テクノロジープレビューとして利用可能な IBM Z アーキテクチャー用の GNOME

GNOME デスクトップ環境は、テクノロジープレビューとして IBM Z アーキテクチャーで利用できます。

VNC を使用して IBM Z サーバーのデスクトップセッションに接続できるようになりました。その結果、グラフィカルアプリケーションを使用してサーバーを管理できます。

IBM Z では、限定されたグラフィカルアプリケーションのセットを使用できます。たとえば、次のようになります。

- Firefox Web ブラウザー
- Red Hat Subscription マネージャー (**subscription-manager-cockpit**)
- ファイアウォール設定 (**firewall-config**)
- ディスク使用状況アナライザー (**baobab**)

Firefox を使用して、サーバー上の Cockpit サービスに接続できます。

LibreOffice などの特定のアプリケーションは、コマンドラインインターフェイスのみを提供し、グラフィカルインターフェイスは無効になっています。

Jira:RHELPLAN-27737[1]

7.8. グラフィックインフラストラクチャー

64 ビット ARM アーキテクチャーで VNC リモートコンソールがテクノロジープレビューとして利用可能に

64 ビットの ARM アーキテクチャーでは、Virtual Network Computing (VNC) リモートコンソールがテクノロジープレビューとして利用できます。グラフィックススタックの残りの部分は、現在、64 ビット ARM アーキテクチャーでは検証されていません。

Bugzilla:1698565[1]

7.9. 仮想化

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

Bugzilla:1519039[1]

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ーが向上します。

さらに、強化された SEV (Encrypted State) バージョンの SEV (SEV-ES) もテクノロジープレビューとして提供されます。SEV-ES は、仮想マシンの実行が停止すると、すべての CPU レジスターの内容を暗号化します。これにより、ホストが仮想マシンの CPU レジスターを変更したり、そこから情報を読み取ったりできなくなります。

SEV および SEV-ES は、第 2 世代の AMD EPYC CPU (コードネーム Rome) 以降でのみ機能することに注意してください。また、RHEL 8 には SEV および SEV-ES の暗号化が含まれますが、SEV および SEV-ES のセキュリティー証明は含まれません。

Bugzilla:1501618[1], Bugzilla:1501607、Jira:RHELPLAN-7677

Intel vGPU がテクノロジープレビューとして利用可能になる

テクノロジープレビューとして、物理 Intel GPU デバイスを、**mediated devices** 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。

この機能は非推奨であり、今後の RHEL メジャーリリースでは完全に削除される予定であることに注意してください。

Bugzilla:1528684[1]

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[1]、Jira:RHELPLAN-24437

テクノロジープレビュー: 一部の Intel ネットワークアダプターが、Hyper-V の RHEL ゲストに SR-IOV を提供するようになりました

テクノロジープレビューとして、Hyper-V ハイパーバイザーで実行している Red Hat Enterprise Linux のゲストオペレーティングシステムは、**ixgbevf** および **ixgbevf** ドライバーがサポートする Intel ネットワークアダプターに、シングルルート I/O 仮想化 (SR-IOV) 機能を使用することができるようになりました。この機能は、以下の条件が満たされると有効になります。

- ネットワークインターフェイスコントローラー (NIC) に対して SR-IOV サポートが有効になっている
- 仮想 NIC の SR-IOV サポートが有効になっている
- 仮想スイッチの SR-IOV サポートが有効になっている
- NIC からの VF (Virtual Function) が仮想マシンに割り当てられている

この機能は現在、Microsoft Windows Server 2016 以降で提供されています。

Bugzilla:1348508[1]

RHEL ゲストのインテル TDX

テクノロジープレビューとして、Intel Trust Domain Extension (TDX) 機能が RHEL 8.8 以降のゲストオペレーティングシステムで使用できるようになりました。ホストシステムが TDX をサポートしている場合は、トラストドメイン (TD) と呼ばれる、ハードウェアから分離された RHEL 9 仮想マシン (VM) をデプロイできます。ただし、TDX は現在 **kdump** では機能せず、TDX を有効にすると VM 上で **kdump** が失敗することに注意してください。

Bugzilla:1836977[1]

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。

Bugzilla:1741615[1]

7.10. クラウド環境の RHEL

RHEL Confidential VM がテクノロジープレビューとして Azure で利用可能になりました

更新された RHEL カーネルを使用すると、Microsoft Azure で機密仮想マシン (VM) をテクノロジープレビューとして作成して実行できるようになりました。ただし、Azure での起動中に RHEL 機密 VM イメージを暗号化することはまだできません。

Jira:RHELPLAN-122316[1]

7.11. コンテナー

podman-machine コマンドはサポート対象外です。

仮想マシンを管理するための **podman-machine** コマンドは、テクノロジープレビューとしてのみ利用可能です。代わりに、コマンドラインから直接 Podman を実行してください。

Jira:RHELDOCS-16861[1]

マルチアーキテクチャーイメージのビルド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

マルチアーキテクチャーコンテナーイメージの作成に使用できる **podman farm build** コマンドが、テクノロジープレビューとして利用できます。

ファームとは、UNIX podman ソケットが実行されているマシンのグループです。ファーム内のノードには、さまざまなアーキテクチャーのさまざまなマシンが存在する場合があります。**podman farm build** コマンドは、**podman build --arch --platform** コマンドよりも高速です。

podman farm build を使用して、次のアクションを実行できます。

- ファーム内のすべてのノードにイメージをビルドします。
- ノードをマニフェストリストにバンドルします。
- すべてのファームノードで **podman build** コマンドを実行します。
- **--tag** オプションを使用して指定されたレジストリーにイメージをプッシュします。
- マニフェストリストをローカルに作成します。
- マニフェストリストをレジストリーにプッシュします。
  マニフェストリストには、ファーム内に存在するネイティブアーキテクチャータイプごとに 1 つのイメージが含まれます。

Jira:RHELPLAN-154435[1]

第8章 非推奨の機能

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。

非推奨のデバイスは完全にサポートされています。つまり、非推奨のデバイスはテストおよび保守されています。デバイスが非推奨になっても、Red Hat Enterprise Linux 8 内でのサポート状況は変わりません。ただし、非推奨のデバイスは、次のメジャーバージョンのリリースではサポートされない可能性が高く、最新または今後のメジャーバージョンの新規 RHEL デプロイメントには推奨されません。

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。サポート期間の詳細は、Red Hat Enterprise Linux のライフサイクル および Red Hat Enterprise Linux アプリケーションストリームのライフサイクル を参照してください。

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。特定の状況下では、製品からパッケージが削除されることがあります。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。

RHEL 8 で使用され、RHEL 9 で削除された機能の詳細は RHEL 9 の導入における検討事項 を参照してください。

8.1. 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。

Bugzilla:1642765[1]

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。

Bugzilla:1637872[1]

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。

Bugzilla:1904251[1]

8.2. 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。

Bugzilla: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ナレッジベースの記事 Strong crypto defaults in RHEL 8 and deprecation of weak crypto algorithms および man ページの **update-crypto-policies(8)** を参照してください。

Bugzilla: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。

Bugzilla:1646541[1]

fapolicyd.rules が非推奨になる

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のみに使用されます。

Bugzilla:2054741

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。この機能のサポートは、Red Hat Enterprise Linux 8 の今後のリリースで完全に削除される可能性があります。

Bugzilla:1645153[1]

/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。

Bugzilla: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。

Bugzilla:1461914[1]

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ー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。

Bugzilla:1657927[1]

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。

Bugzilla:2011208

RHEL 8 および 9 OpenSSL 証明書および署名コンテナーが非推奨になる

Red Hat Ecosystem Catalog の **ubi8/openssl** および **ubi9/openssl** リポジトリーで利用可能な OpenSSL ポータブル証明書および署名コンテナーは、需要が低いため非推奨になりました。

Jira:RHELDOCS-17974[1]

8.3. サブスクリプションの管理

subscription-manager register の非推奨の --token オプションは、2024 年 11 月末に機能しなくなります

subscription-manager register コマンドの非推奨の **--token=<TOKEN>** オプションは、2024 年 11 月末以降はサポート対象の認証方法ではなくなります。デフォルトのエンタイトルメントサーバー **subscription.rhsm.redhat.com** では、トークンベースの認証が許可されなくなります。したがって、**subscription-manager register --token=<TOKEN>** を使用すると、次のエラーメッセージが表示されて登録が失敗します。

> Token authentication not supported by the entitlement server

システムを登録するには、サポート対象の他の認可方法を使用します。たとえば、**subscription-manager register** コマンドにペアのオプション **--username** / **--password** または **--org** / **--activationkey** を含めます。

Bugzilla:2170082

8.4. 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。

Bugzilla:1688849

8.5. シェルおよびコマンドラインツール

ReaR 設定ファイルでの TMPDIR 変数の設定が非推奨になる

export TMPDIR=... などのステートメントを使用して、**/etc/rear/local.conf** または **/etc/rear/site.conf** ReaR 設定ファイルで **TMPDIR** 環境変数を設定することは非推奨となりました。

ReaR 一時ファイルのカスタムディレクトリーを指定するには、ReaR を実行する前にシェル環境で変数をエクスポートします。たとえば、**export TMPDIR=...** ステートメントを入力し、同じシェルセッションまたはスクリプトで **rear** コマンドを入力します。

Jira:RHELDOCS-18049[1]

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。

Bugzilla:1886310

dump からの dump ユーティリティーが非推奨になる

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で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。

Bugzilla:1997366[1]

hidepid=n マウントオプションが、RHEL 8 systemd で未サポート

マウントオプションの **hidepid=n** は、**/proc/[pid]** ディレクトリーの情報にアクセスできるユーザーを制御しますが、RHEL 8 で提供されている **systemd** インフラストラクチャーと互換性がありません。

また、このオプションを使用すると、**systemd** が起動する特定のサービスで SELinux の AVC 拒否メッセージが生成され、その他の操作が完了しないようにする場合があります。

詳細は、関連するナレッジベースのソリューション記事 Is mounting /proc with "hidepid=2" recommended with RHEL7 and RHEL8? を参照してください。

Bugzilla: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

Bugzilla:1875485

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。

Bugzilla:2055826[1]

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。

Bugzilla:2083301

Bacula の SQLite データベースバックエンドが非推奨になりました

Bacula バックアップシステムは、複数のデータベースバックエンド (PostgreSQL、MySQL、および SQLite) をサポートしていました。SQLite バックエンドに非推奨となり、RHEL の今後のリリースではサポートされなくなります。代わりに、他のバックエンド (PostgreSQL または MySQL) のいずれかに移行し、新しい展開では SQLite バックエンドを使用しないでください。

Jira:RHEL-6859

raw コマンドが非推奨になりました

raw (**/usr/bin/raw**) コマンドが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。

Jira:RHELPLAN-133171[1]

8.6. インフラストラクチャーサービス

geoipupdate パッケージが非推奨に

geoipupdate パッケージにはサードパーティーのサブスクリプションが必要で、プロプライエタリーコンテンツもダウンロードします。したがって、**geoipupdate** パッケージは非推奨となり、次の RHEL メジャーバージョンで削除されます。

Bugzilla:1874892[1]

8.7. ネットワーク

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。

Bugzilla:1647725[1]

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の 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。

Bugzilla:1929173

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。

Bugzilla:2009113[1]

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ング交換デーモン (**cgdcbxd**) 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、**cgdcbxd** 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。

Bugzilla:2006665

WEP Wi-Fi 接続方法が非推奨になりました。

安全でない WEP (wired equivalent privacy) の Wi-Fi 接続方法は、RHEL 8 では非推奨となり、RHEL 9.0 では削除されます。安全な Wi-Fi 接続には、Wi-Fi Protected Access 3 (WPA3) または WPA2 の接続方法を使用します。

Bugzilla: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、**iptable** のユーザーはパケットヘッダーまたはペイロード内の任意の 32 ビットにマッチできます。RHEL 8.6 以降、**xt_u32** モジュールが非推奨になり、RHEL 9 では削除されます。

xt_u32 を使用する場合は、**nftable** パケットフィルタリングフレームワークに移行します。たとえば、最初にファイアウォールを、個々のルールを段階的に置き換えるために、ネイティブ一致で **iptable** 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。

Bugzilla:2061288

8.8. カーネル

rdma_rxe Soft-RoCE ドライバーが非推奨に

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 機能がテクノロジープレビューとして利用できます。さらに、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。

Bugzilla:1878207[1]

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。

Bugzilla:1871863[1]

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 8.3 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した。

Bugzilla:1748980

カーネルライブパッチが、すべての RHEL マイナーリリースに対応するようになりました。

RHEL 8.1 以降、カーネルライブパッチは、影響度が重大および重要な Common Vulnerabilities and Exposures (CVE) を修正するために、Extended Update Support (EUS) ポリシーの対象となる RHEL の一部のマイナーリリースストリームに提供されています。同時にカバーされるカーネルとユースケースの最大数に対応するため、各ライブパッチのサポート期間は、カーネルのマイナー、メジャー、および zStream の各バージョンで 12 カ月から 6 カ月に短縮されました。これは、カーネルライブパッチがリリースされると、過去 6 カ月間に配信されたすべてのマイナーリリースとスケジュール済みのエラータカーネルが含まれます。

この機能の詳細は、Applying patches with kernel live patching を参照してください。

利用可能なカーネルライブパッチの詳細は、Kernel Live Patch life cycles を参照してください。

Bugzilla:1958250

crash-ptdump-command パッケージは非推奨です

クラッシュユーティリティーの **ptdump** 拡張モジュールである **crash-ptdump-command** パッケージは非推奨であり、将来の RHEL リリースでは利用できなくなる可能性があります。**ptdump** コマンドは、Single Range Output モードで作業している場合、ログバッファーの取得に失敗し、Table of Physical Addresses (ToPA) モードでのみ機能します。**crash-ptdump-command** は現在、アップストリームに維持されていません

Bugzilla:1838927[1]

8.9. ブートローダー

kernelopts 環境変数は非推奨になる

RHEL 8 では、GRUB 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。

Bugzilla:2060759

8.10. ファイルシステムおよびストレージ

Resilient Storage Add-On が非推奨になりました。

Red Hat Enterprise Linux (RHEL) Resilient Storage Add-On が非推奨になりました。Resilient Storage Add-On は、Red Hat Enterprise Linux 10 および RHEL 10 以降のリリースではサポートされなくなります。RHEL Resilient Storage Add-On は、以前のバージョンの RHEL (7、8、9) で、および各バージョンのメンテナンスサポートライフサイクル中は引き続きサポートされます。

Jira:RHELDOCS-19027

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。

詳細は、ディスクスケジューラーの設定 を参照してください。

Bugzilla:1665295[1]

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。

Bugzilla:1592011[1]

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。

Bugzilla:1871953

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[1]

VDO マネージャーが非推奨に

python ベースの VDO 管理ソフトウェアは非推奨となり、RHEL 9 から削除される予定です。RHEL 9 では、LVM-VDO 統合に置き換えられます。そのため、**lvcreate** コマンドを使用して VDO ボリュームを作成することが推奨されます。

VDO 管理ソフトウェアを使用して作成した既存のボリュームは、**lvm2** パッケージが提供する **/usr/sbin/lvm_import_vdo** スクリプトを使用して変換できます。LVM-VDO 実装の詳細は、RHEL での論理ボリュームの重複排除および圧縮 を参照してください。

Bugzilla:1949163

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。

Bugzilla:1794513[1]

8.11. 高可用性およびクラスター

cluster ツールに対応する pcs コマンドが非推奨になりました

clufter ツールに対応する pcs コマンドが非推奨になりました。

クラスター設定フォーマットを分析する **clufter** ツールに対応する **pcs** コマンドが非推奨になりました。これらのコマンドにより、コマンドが非推奨になり、コマンドに関連するセクションが **pcs** ヘルプ表示と、**pcs(8)** man ページから削除されていることを示す警告が出力されるようになりました。

以下のコマンドが非推奨になりました。

- **pcs config import-cman**: CMAN / RHEL6 HA クラスター設定のインポート
- **pcs config export**: クラスター設定を、同じクラスターを再作成する **pcs** コマンドのリストにエクスポート

Bugzilla:1851335[1]

8.12. 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

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。

Bugzilla:2225332

8.13. コンパイラーおよび開発ツール

gdb.i686 パッケージが非推奨に

RHEL 8.1 では、別のパッケージの依存関係の問題が原因で、32 ビットバージョンの GNU Debugger(GDB) **gdb.i686** が同梱されていました。RHEL 8 は 32 ビットハードウェアに対応していないため、RHEL 8.4 以降、**gdb.i686** パッケージは非推奨になりました。64 ビットバージョンの GDB (**gdb.x86_64**) は、32 ビットアプリケーションをデバッグできます。

gdb.i686 を使用する場合は、以下の重要な問題に注意してください。

- **gdb.i686** パッケージは更新されなくなりました。代わりに **gdb.x86_64** をインストールする必要があります。
- **gdb.i686** をインストールしている場合は、**gdb.x86_64** をインストールすると、**yum** が **package gdb-8.2-14.el8.x86_64 obsoletes gdb < 8.2-14.el8 provided by gdb-8.2-12.el8.i686** を報告します。これは想定される状況です。**gdb.i686** をアンインストールするか、**--allowerasing** オプションを **dnf** に渡して **gdb.i686** を削除し、**gdb.x8_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。

Bugzilla:1853140[1]

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。

Bugzilla:1920624

8.14. 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。

Bugzilla:1871025

DES および 3DES 暗号化タイプが削除されました。

RHEL 7 以降、セキュリティー上の理由から、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。

Bugzilla:1877991

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。

Bugzilla:1947671

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** クラスター Samba サービスを使用することが推奨されます。

- **ctdb** サービスは、resource-agent **ctdb** を使用して **pacemaker** リソースとして管理されます。
- **ctdb** サービスは、Red Hat Gluster Storage 製品または GFS2 ファイルシステムが提供する GlusterFS ファイルシステムのいずれかが含まれるストレージボリュームを使用します。

ctdb サービスのスタンドアロンユースケースは非推奨となり、Red Hat Enterprise Linux の次期メジャーリリースには含まれません。Samba のサポートポリシーの詳細は、ナレッジベースの記事 Support Policies for RHEL Resilient Storage - ctdb General Policies を参照してください。

Bugzilla:1916296[1]

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。

非推奨になっ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ー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ている場合。認証は、**krb5** および LDAP 認証パッケージを使用して、またはメインの FreeRADIUS パッケージの PAM 認証として行われます。
- FreeRADIUS を使用して、Python 3 認証パッケージで IdM の認証用に信頼できる情報源を提供している場合。

これらの非推奨化とは対照的に、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:RHELDOCS-17573[1]

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ー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[1]

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 をプライマリードメインコントローラー (PDC) やバックアップドメインコントローラー (BDC) などの NT4 として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。

Bugzilla:1926114

Samba で SMB1 プロトコルが非推奨になる

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。

セキュリティー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ーで SMB1 が無効になっています。

Jira:RHELDOCS-16612[1]

8.15. 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。

Bugzilla:1607766[1]

LibreOffice が非推奨になる

LibreOffice RPM パッケージは非推奨となり、今後の RHEL メジャーリリースで削除される予定です。LibreOffice は、RHEL 7、8、および 9 のライフサイクル全体を通じて引き続き完全にサポートされます。

Red Hat は、RPM パッケージの代わりに、The Document Foundation が提供する次のいずれかのソースから LibreOffice をインストールすることを推奨します。

- Flathub リポジトリーの公式 Flatpak パッケージ: https://flathub.org/apps/org.libreoffice.LibreOffice
- 公式 RPM パッケージ: https://www.libreoffice.org/download/download-libreoffice/

Jira:RHELDOCS-16300[1]

いくつかのビットマップフォントが非推奨になりました

次のビットマップフォントパッケージは非推奨になりました。

- **bitmap-console-fonts**
- **bitmap-fixed-fonts**
- **bitmap-fonts-compat**

- **bitmap-lucida-typewriter-fonts**

ビットマップフォントにはピクセルサイズが制限されています。使用できないフォントサイズを設定しようとすると、テキストが異なるサイズや異なるフォント (場合によってはスケーラブルなフォント) で表示されることがあります。これにより、ビットマップフォントのレンダリング品質も低下し、ユーザーエクスペリエンスが損なわれます。

さらに、**fontconfig** システムは、主要なビットマップフォント形式の 1 つである Portable Compiled Format (PCF) を無視します。これは、当該フォント形式が、言語の範囲を推定するためのメタデータを含まないためです。

なお、**bitmap-fangsongti-fonts** ビットマップフォントパッケージは、Lorax ツールの依存関係として引き続きサポートされます。

Jira:RHELDOCS-17623[1]

8.16. 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。

Bugzilla:1569610[1]

Motif が非推奨になる

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。

開発バリアントおよびデバッグバリアントを含む、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[1]

8.17. 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。

Bugzilla: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[1]

8.18. 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、network システムロールが非推奨の警告を表示します。

ネットワークチーミング機能は、RHEL 9 では非推奨になりました。その結果、RHEL 8 制御ノードで **network** RHEL システムロールを使用して RHEL 9 ノードでネットワークチームを設定すると、非推奨の警告が表示されます。

Bugzilla: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 リポジトリー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は、RHEL 9 および RHEL 8.6 以降の AppStream リポジトリーに含まれる Ansible Core パッケージのサポート対象範囲 を参照してください。

ユーザーは、システムを Ansible Engine から Ansible Core に手動で移行する必要があります。そのためには、以下の手順に従います。

手順

1. システムが RHEL 8.7 以降のリリース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** リポジトリーを無効にします。

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

4. RHEL 8 AppStream リポジトリーから Ansible Core パッケージをインストールします。

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

詳細は、RHEL8.6 以降での Ansible の使用 を参照してください。

Bugzilla:2006081

mssql_ha_cluster_run_role が非推奨になりました

mssql_ha_cluster_run_role 変数は非推奨になりました。代わりに、**mssql_manage_ha_cluster** 変数を使用します。

Jira:RHEL-19203

8.19. 仮想化

virsh iface-* コマンドが非推奨になりました。

virsh iface-start、**virsh iface-destroy** などの **virsh iface-*** コマンドは非推奨になり、将来のメジャーバージョンの RHEL では削除される予定です。また、このようなコマンドは設定の依存関係により頻繁に失敗します。

したがって、ホストネットワーク接続の設定および管理には **virsh iface-*** コマンドを使用しないことが推奨されます。代わりに、NetworkManager プログラムと、関連する管理アプリケーション (**nmcli** など) を使用します。

Bugzilla:1664592[1]

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[1]

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。

Bugzilla:1686057

Cirrus VGA 仮想 GPU タイプが非推奨に

Red Hat Enterprise Linux の今後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。

Bugzilla:1651994[1]

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。RHEL 8 では SPICE が引き続きサポートされていますが、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。

Bugzilla:1849563[1]

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[1]

SHA1 ベースの署名を使用した SecureBoot イメージ検証が非推奨になる

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、SHA-2 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。

Bugzilla:1935497[1]

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。

Bugzilla:2059626

RDMA ベースのライブマイグレーションが非推奨になりました

この更新により、リモートダイレクトメモリーアクセス (RDMA) を使用した実行中の仮想マシンの移行は非推奨になりました。その結果、**rdma://** 移行 URI を使用して RDMA 経由の移行を要求することは可能ですが、この機能は RHEL の将来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-153267[1]

8.20. コンテナー

GIMP flatpak および GIMP モジュールが非推奨になりました

GIMP flatpak および **GIMP** モジュールは、どちらもラスターグラフィックス用の GNU Image Manipulation Program を表します。これらのモジュールは、Python 2 のライフサイクル終了 (EOL) に伴い、非推奨とマークされました。次の RHEL メジャーリリースで削除される予定です。代わりに、Python 3 ベースのアップストリームの flatpak バージョンを使用できます。

Jira:RHEL-18958

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[1]

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、セキュリティー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[1]

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、セキュリティー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[1]

GIMP 以外の Flatpak イメージが非推奨になりました

rhel8/firefox-flatpak、**rhel8/thunderbird-flatpak**、**rhel8/inkscape-flatpak**、および **rhel8/libreoffice-flatpak** RHEL 8 Flatpak アプリケーションが非推奨になり、RHEL 9 バージョンに置き換えられました。RHEL 9 にはまだ代替品がないため、**rhel8/gimp-flatpak** Flatpak アプリケーションは非推奨ではありません。

Bugzilla:2142499

CNI ネットワークスタックが非推奨になる

Container Network Interface (CNI) ネットワークスタックは非推奨となり、RHEL の今後のマイナーリリースでは Podman から削除される予定です。以前は、コンテナーは DNS 経由のみで単一の Container Network Interface (CNI) プラグインに接続していました。Podman v.4.0 では、新しい Netavark ネットワークスタックが導入されました。Netavark ネットワークスタックは、Podman およびその他の Open Container Initiative (OCI) コンテナー管理アプリケーションとともに使用できます。Podman 用の Netavark ネットワークスタックは、高度な Docker 機能とも互換性があります。複数のネットワーク内のコンテナーは、それらのネットワークのいずれかにあるコンテナーにアクセスできます。

詳細は、CNI から Netavark へのネットワークスタックの切り替え を参照してください。

Jira:RHELDOCS-16755[1]

container-tools:3.0 が非推奨になりました。

container-tools:3.0 モジュールは非推奨となり、セキュリティー更新を受信しなくなります。RHEL 上で Linux コンテナーの構築と実行を続けるには、**container-tools:4.0** など、より新しく安定したサポートされているモジュールストリームを使用してください。

後続のストリームに切り替える手順は、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-146398[1]

rhel8/openssl が非推奨になりました

rhel8/openssl コンテナーイメージは非推奨になりました。

Jira:RHELDOCS-18107[1]

Inkscape および LibreOffice Flatpak イメージが非推奨になる

テクノロジープレビューとして利用可能な **rhel9/inkscape-flatpak** および **rhel9/libreoffice-flatpak** Flatpak イメージは非推奨になりました。

Red Hat は、これらのイメージに代わる次の代替手段を推奨します。

- **rhel9/inkscape-flatpak** を置き換えるには、**inkscape** RPM パッケージを使用します。
- **rhel9/libreoffice-flatpak** を置き換えるには、LibreOffice 非推奨化に関するリリースノート を参照してください。

Jira:RHELDOCS-17102[1]

ネットワーク名としての pasta が非推奨になる

ネットワーク名の値としての **pasta** に関するサポートは非推奨になり、Podman の次のメジャーリリースであるバージョン 5.0 では使用できなくなります。**pasta** というネットワーク名の値は、**podman run --network** コマンドと **podman create --network** コマンドを使用して、Podman 内に一意のネットワークモードを作成するために使用できます。

Jira:RHELDOCS-17038[1]

BoltDB データベースバックエンドが非推奨になる

BoltDB データベースバックエンドは、RHEL 8.10 以降では非推奨です。RHEL の今後のバージョンでは、BoltDB データベースバックエンドが削除され、Podman では利用できなくなります。Podman の場合は、RHEL 8.10 以降ではデフォルトとなっている SQLite データベースバックエンドを使用してください。

Jira:RHELDOCS-17461[1]

CNI ネットワークスタックが非推奨になる

Container Network Interface (CNI) ネットワークスタックは非推奨となり、今後のリリースでは削除される予定です。代わりに Netavark ネットワークスタックを使用してください。詳細は、CNI から Netavark へのネットワークスタックの切り替え を参照してください。

Jira:RHELDOCS-17518[1]

container-tools:4.0 が非推奨になりました。

container-tools:4.0 モジュールは非推奨となり、セキュリティー更新を受信しなくなります。RHEL 上で Linux コンテナーの構築と実行を続けるには、より新しく安定したサポートされているモジュールストリーム **container-tools:rhel8** を使用してください。

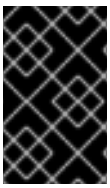
後続のストリームに切り替える手順は、後続のストリームへの切り替え を参照してください。

Jira:RHELPLAN-168223[1]

8.21. 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。



重要

非推奨パッケージのサポート状況は、RHEL 8 内でも変更されません。サポート期間の詳細は、Red Hat Enterprise Linux のライフサイクル および Red Hat Enterprise Linux アプリケーションストリームのライフサイクル を参照してください。

次のパッケージは RHEL 8 で非推奨になりました。

- 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
- arpwatch
- 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

- bash-doc
- batik
- batik-css
- batik-util
- bea-stax
- bea-stax-api
- bind-export-devel
- bind-export-libs
- bind-libs-lite
- bind-pkcs11
- bind-pkcs11-devel
- bind-pkcs11-libs
- bind-pkcs11-utils
- bind-sdb
- bind-sdb
- bind-sdb-chroot
- bitmap-console-fonts
- bitmap-fixed-fonts
- bitmap-fonts-compat
- bitmap-lucida-typewriter-fonts
- 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
- culmus-keteryg-fonts
- culmus-shofar-fonts
- custodia
- cyrus-imapd-vzic
- dbus-c++
- dbus-c++-devel
- dbus-c++-glib

- dbxtool
- dejavu-fonts-common
- dhcp-libs
- directory-maven-plugin
- directory-maven-plugin-javadoc
- dirsplit
- dleyna-connector-dbus
- dleyna-core
- dleyna-renderer
- dleyna-server
- dnssec-trigger
- dnssec-trigger-panel
- dotnet
- 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
- dptfxtract
- drpm
- drpm-devel
- dump
- dvd+rw-tools
- dyninst-static
- eclipse-ecf
- eclipse-ecf-core
- eclipse-ecf-runtime
- eclipse-emf
- eclipse-emf-core
- eclipse-emf-runtime
- eclipse-emf-xsd
- eclipse-equinox-osgi
- eclipse-jdt
- eclipse-license
- eclipse-p2-discovery
- eclipse-pde
- eclipse-platform
- eclipse-swt
- ed25519-java

- ee4j-parent
- elfutils-devel-static
- elfutils-libelf-devel-static
- emacs-terminal
- emoji-picker
- enca
- enca-devel
- environment-modules-compat
- evince-browser-plugin
- exec-maven-plugin
- farstream02
- felix-gogo-command
- felix-gogo-runtime
- felix-gogo-shell
- felix-scr
- felix-osgi-compendium
- felix-osgi-core
- felix-osgi-foundation
- felix-parent
- file-roller
- fipscheck
- fipscheck-devel
- fipscheck-lib
- firewire
- fonts-tweak-tool
- forge-parent
- freeradius-mysql
- freeradius-perl
- freeradius-postgresql

- freeradius-rest
- freeradius-sqlite
- freeradius-unixODBC
- fuse-sshfs
- fusesource-pom
- future
- gamin
- gamin-devel
- gavl
- gcc-toolset-9
- gcc-toolset-9-annobin
- gcc-toolset-9-build
- gcc-toolset-9-perftools
- gcc-toolset-9-runtime
- gcc-toolset-9-toolchain
- gcc-toolset-10
- gcc-toolset-10-annobin
- gcc-toolset-10-binutils
- gcc-toolset-10-binutils-devel
- gcc-toolset-10-build
- gcc-toolset-10-dwz
- gcc-toolset-10-dyninst
- gcc-toolset-10-dyninst-devel
- gcc-toolset-10-elfutils
- gcc-toolset-10-elfutils-debuginfod-client
- gcc-toolset-10-elfutils-debuginfod-client-devel
- gcc-toolset-10-elfutils-devel
- gcc-toolset-10-elfutils-libelf
- gcc-toolset-10-elfutils-libelf-devel

- gcc-toolset-10-elfutils-libs
- gcc-toolset-10-gcc
- gcc-toolset-10-gcc-c++
- gcc-toolset-10-gcc-gdb-plugin
- gcc-toolset-10-gcc-gfortran
- gcc-toolset-10-gdb
- gcc-toolset-10-gdb-doc
- gcc-toolset-10-gdb-gdbserver
- gcc-toolset-10-libasan-devel
- gcc-toolset-10-libatomic-devel
- gcc-toolset-10-libitm-devel
- gcc-toolset-10-liblsan-devel
- gcc-toolset-10-libquadmath-devel
- gcc-toolset-10-libstdc++-devel
- gcc-toolset-10-libstdc++-docs
- gcc-toolset-10-libtsan-devel
- gcc-toolset-10-libubsan-devel
- gcc-toolset-10-ltrace
- gcc-toolset-10-make
- gcc-toolset-10-make-devel
- gcc-toolset-10-perftools
- gcc-toolset-10-runtime
- gcc-toolset-10-strace
- gcc-toolset-10-systemtap
- gcc-toolset-10-systemtap-client
- gcc-toolset-10-systemtap-devel
- gcc-toolset-10-systemtap-initscript
- gcc-toolset-10-systemtap-runtime
- gcc-toolset-10-systemtap-sdt-devel

- gcc-toolset-10-systemtap-server
- gcc-toolset-10-toolchain
- gcc-toolset-10-valgrind
- gcc-toolset-10-valgrind-devel
- gcc-toolset-11-make-devel
- gcc-toolset-12-annobin-annocheck
- gcc-toolset-12-annobin-docs
- gcc-toolset-12-annobin-plugin-gcc
- gcc-toolset-12-binutils
- gcc-toolset-12-binutils-devel
- gcc-toolset-12-binutils-gold
- 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-droid-kufi-fonts
- google-gson
- google-noto-kufi-arabic-fonts
- google-noto-naskh-arabic-fonts
- google-noto-naskh-arabic-ui-fonts
- google-noto-nastaliq-urdu-fonts
- google-noto-sans-balinese-fonts
- google-noto-sans-bamum-fonts
- google-noto-sans-batak-fonts
- google-noto-sans-buginese-fonts
- google-noto-sans-buhid-fonts
- google-noto-sans-canadian-aboriginal-fonts
- google-noto-sans-cham-fonts
- google-noto-sans-cuneiform-fonts
- google-noto-sans-cypriot-fonts
- google-noto-sans-gothic-fonts
- google-noto-sans-gurmukhi-ui-fonts
- google-noto-sans-hanunoo-fonts
- google-noto-sans-inscriptional-pahlavi-fonts
- google-noto-sans-inscriptional-parthian-fonts
- google-noto-sans-javanese-fonts
- google-noto-sans-lepcha-fonts
- google-noto-sans-limbu-fonts
- google-noto-sans-linear-b-fonts
- google-noto-sans-lisu-fonts
- google-noto-sans-mandaic-fonts
- google-noto-sans-meetei-mayek-fonts

- google-noto-sans-mongolian-fonts
- google-noto-sans-myanmar-fonts
- google-noto-sans-myanmar-ui-fonts
- google-noto-sans-new-tai-lue-fonts
- google-noto-sans-ogham-fonts
- google-noto-sans-ol-chiki-fonts
- google-noto-sans-old-italic-fonts
- google-noto-sans-old-persian-fonts
- google-noto-sans-oriya-fonts
- google-noto-sans-oriya-ui-fonts
- google-noto-sans-phags-pa-fonts
- google-noto-sans-rejang-fonts
- google-noto-sans-runic-fonts
- google-noto-sans-samaritan-fonts
- google-noto-sans-saurashtra-fonts
- google-noto-sans-sundanese-fonts
- google-noto-sans-syloti-nagri-fonts
- google-noto-sans-syriac-eastern-fonts
- google-noto-sans-syriac-estrangela-fonts
- google-noto-sans-syriac-western-fonts
- google-noto-sans-tagalog-fonts
- google-noto-sans-tagbanwa-fonts
- google-noto-sans-tai-le-fonts
- google-noto-sans-tai-tham-fonts
- google-noto-sans-tai-viet-fonts
- google-noto-sans-tibetan-fonts
- google-noto-sans-tifinagh-fonts
- google-noto-sans-ui-fonts
- google-noto-sans-yi-fonts

- google-noto-serif-bengali-fonts
- google-noto-serif-devanagari-fonts
- google-noto-serif-gujarati-fonts
- google-noto-serif-kannada-fonts
- google-noto-serif-malayalam-fonts
- google-noto-serif-tamil-fonts
- google-noto-serif-telugu-fonts
- gphoto2
- graphviz-ruby
- gsl-devel
- gssntlmssp
- gtest
- gtest-devel
- gtkmm24
- gtkmm24-devel
- gtkmm24-docs
- gtksourceview3
- gtksourceview3-devel
- gtkspell
- gtkspell-devel
- gtkspell3
- guile
- gutenprint-gimp
- gutenprint-libs-ui
- gvfs-afc
- gvfs-afp
- gvfs-archive
- hamcrest-core
- hawtjni

- hawtjni
- hawtjni-runtime
- HdrHistogram
- HdrHistogram-javadoc
- highlight-gui
- hivex-devel
- hostname
- hplip-gui
- hspell
- 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
- inkscape
- inkscape-docs
- inkscape-view
- iptables

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

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

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

- libcgroup
- libcgroup-pam
- 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
- libicu-doc
- libIDL
- libIDL-devel
- libidn
- libiec61883
- libindicator-gtk3
- libindicator-gtk3-devel
- libiscsi-devel

- libjose-devel
- libkkc
- libkkc-common
- libkkc-data
- libldb-devel
- liblogging
- libluksmeta-devel
- libmalaga
- libmcpp
- libmemcached
- libmemcached-libs
- libmetalink
- libmodulemd1
- libmongocrypt
- libmtp-devel
- libmusicbrainz5
- libmusicbrainz5-devel
- libnbd-devel
- libnice
- libnice-gstreamer1
- liboauth
- liboauth-devel
- libpfm-static
- libpng12
- libpsm2-compat
- libpurple
- libpurple-devel
- libraw1394
- libreport-plugin-mailx

- libreport-plugin-rhtsupport
- libreport-plugin-ureport
- libreport-rhel
- libreport-rhel-bugzilla
- librpmem
- librpmem-debug
- librpmem-devel
- libsass
- libsass-devel
- libselinux-python
- libsqlite3x
- libtalloc-devel
- libtar
- libtdb-devel
- libtevent-devel
- libtpms-devel
- libunwind
- libusal
- libvarlink
- libverto-libevent
- libvirt-admin
- libvirt-bash-completion
- libvirt-daemon-driver-storage-gluster
- libvirt-daemon-driver-storage-iscsi-direct
- libvirt-devel
- libvirt-docs
- libvirt-gconfig
- libvirt-gobject
- libvirt-lock-sanlock

- libvirt-wireshark
- libvmem
- libvmem-debug
- libvmem-devel
- libvmmalloc
- libvmmalloc-debug
- libvmmalloc-devel
- libvncserver
- libwinpr-devel
- libwmf
- libwmf-devel
- libwmf-lite
- libXNVCtrl
- libyami
- log4j12
- log4j12-javadoc
- lohit-malayalam-fonts
- lohit-nepali-fonts
- lorax-composer
- lua-guestfs
- lucene
- lucene-analysis
- lucene-analyzers-smartcn
- lucene-queries
- lucene-queryparser
- lucene-sandbox
- lz4-java
- lz4-java-javadoc
- mailman

- mailx
- make-devel
- malaga
- malaga-suomi-voikko
- marisa
- maven-antrun-plugin
- maven-assembly-plugin
- maven-clean-plugin
- maven-dependency-analyzer
- maven-dependency-plugin
- maven-doxia
- maven-doxia-sitetools
- maven-install-plugin
- maven-invoker
- maven-invoker-plugin
- maven-parent
- maven-plugins-pom
- maven-reporting-api
- maven-reporting-impl
- maven-resolver-api
- maven-resolver-connector-basic
- maven-resolver-impl
- maven-resolver-spi
- maven-resolver-transport-wagon
- maven-resolver-util
- maven-scm
- maven-script-interpreter
- maven-shade-plugin
- maven-shared

- maven-verifier
- maven-wagon-file
- maven-wagon-http
- maven-wagon-http-shared
- maven-wagon-provider-api
- maven2
- meanwhile
- mercurial
- mercurial-hgk
- metis
- metis-devel
- mingw32-bzip2
- mingw32-bzip2-static
- mingw32-cairo
- mingw32-expat
- mingw32-fontconfig
- mingw32-freetype
- mingw32-freetype-static
- mingw32-gstreamer1
- mingw32-harfbuzz
- mingw32-harfbuzz-static
- mingw32-icu
- mingw32-libjpeg-turbo
- mingw32-libjpeg-turbo-static
- mingw32-libpng
- mingw32-libpng-static
- mingw32-libtiff
- mingw32-libtiff-static
- mingw32-openssl

- mingw32-readline
- mingw32-sqlite
- mingw32-sqlite-static
- mingw64-adwaita-icon-theme
- mingw64-bzip2
- mingw64-bzip2-static
- mingw64-cairo
- mingw64-expat
- mingw64-fontconfig
- mingw64-freetype
- mingw64-freetype-static
- mingw64-gstreamer1
- mingw64-harfbuzz
- mingw64-harfbuzz-static
- mingw64-icu
- mingw64-libjpeg-turbo
- mingw64-libjpeg-turbo-static
- mingw64-libpng
- mingw64-libpng-static
- mingw64-libtiff
- mingw64-libtiff-static
- mingw64-nettle
- mingw64-openssl
- mingw64-readline
- mingw64-sqlite
- mingw64-sqlite-static
- modello
- mojo-parent
- mongo-c-driver

- mousetweaks
- mozjs52
- mozjs52-devel
- mozjs60
- mozjs60-devel
- mozvoikko
- msv-javadoc
- msv-manual
- munge-maven-plugin
- mythes-lb
- mythes-mi
- mythes-ne
- nafees-web-naskh-fonts
- nbd
- nbdkit-devel
- nbdkit-example-plugins
- nbdkit-gzip-plugin
- nbdkit-plugin-python-common
- nbdkit-plugin-vddk
- ncompress
- ncurses-compat-libs
- net-tools
- netcf
- netcf-devel
- netcf-libs
- network-scripts
- network-scripts-ppp
- nkf
- nodejs-devel

- nodejs-packaging
- nss_nis
- nss-pam-ldapd
- objectweb-asm
- objectweb-asm-javadoc
- objectweb-pom
- ocaml-bisect-ppx
- ocaml-camlp4
- ocaml-camlp4-devel
- ocaml-lwt
- ocaml-mmap
- ocaml-ocplib-endian
- ocaml-ounit
- ocaml-result
- ocaml-seq
- opencryptoki-tpmtok
- opencv-contrib
- opencv-core
- opencv-devel
- openhpi
- openhpi-libs
- OpenIPMI-perl
- openssh-cavs
- openssh-ldap
- openssl-ibmpkcs11
- opentest4j
- os-maven-plugin
- overpass-mono-fonts
- pakchois

- pandoc
- paps-libs
- paranamer
- paratype-pt-sans-caption-fonts
- parfait
- parfait-examples
- parfait-javadoc
- pcp-parfait-agent
- pcp-pmda-rpm
- pcp-pmda-vmware
- pcsc-lite-doc
- peripety
- perl-B-Debug
- perl-B-Lint
- perl-Class-Factory-Util
- perl-Class-ISA
- perl-DateTime-Format-HTTP
- perl-DateTime-Format-Mail
- perl-File-CheckTree
- perl-homedir
- perl-libxml-perl
- perl-Locale-Codes
- perl-Mozilla-LDAP
- perl-NKF
- perl-Object-HashBase-tools
- perl-Package-DeprecationManager
- perl-Pod-LaTeX
- perl-Pod-Plainer
- perl-prefork

- perl-String-CRC32
- perl-SUPER
- perl-Sys-Virt
- perl-tests
- perl-YAML-Syck
- phodav
- php-recode
- php-xmlrpc
- pidgin
- pidgin-devel
- pidgin-sipe
- pinentry-emacs
- pinentry-gtk
- pipewire0.2-devel
- pipewire0.2-libs
- platform-python-coverage
- plexus-ant-factory
- plexus-bsh-factory
- plexus-cli
- plexus-component-api
- plexus-component-factories-pom
- plexus-components-pom
- plexus-i18n
- plexus-interactivity
- plexus-pom
- plexus-velocity
- plymouth-plugin-throbgress
- pmreorder
- postgresql-test-rpm-macros

- powermock
- prometheus-jmx-exporter
- prometheus-jmx-exporter-openjdk11
- ptscotch-mpich
- ptscotch-mpich-devel
- ptscotch-mpich-devel-parmetis
- ptscotch-openmpi
- ptscotch-openmpi-devel
- purple-sipe
- pygobject2-doc
- pygtk2
- pygtk2-codegen
- pygtk2-devel
- pygtk2-doc
- python-nose-docs
- python-nss-doc
- python-podman-api
- python-psycopg2-doc
- python-pymongo-doc
- python-redis
- python-schedutils
- python-slip
- python-sqlalchemy-doc
- python-varlink
- python-virtualenv-doc
- python2-backports
- python2-backports-ssl_match_hostname
- python2-bson
- python2-coverage

- python2-docs
- python2-docs-info
- python2-funcsigs
- python2-ipaddress
- python2-mock
- python2-nose
- python2-numpy-doc
- python2-psycopg2-debug
- python2-psycopg2-tests
- python2-pymongo
- python2-pymongo-gridfs
- python2-pytest-mock
- python2-sqlalchemy
- python2-tools
- python2-virtualenv
- python3-bson
- python3-click
- python3-coverage
- python3-cpio
- python3-custodia
- python3-docs
- python3-flask
- python3-gevent
- python3-gobject-base
- python3-hivex
- python3-html5lib
- python3-hypothesis
- python3-ipatests
- python3-itsdangerous

- python3-jwt
- python3-libguestfs
- python3-mock
- python3-networkx-core
- python3-nose
- python3-nss
- python3-openipmi
- python3-pillow
- python3-ptyprocess
- python3-pydbus
- python3-pymongo
- python3-pymongo-gridfs
- python3-pyOpenSSL
- python3-pytoml
- python3-reportlab
- python3-schedutils
- python3-scons
- python3-semantic_version
- python3-slip
- python3-slip-dbus
- python3-sqlalchemy
- python3-syspurpose
- python3-virtualenv
- python3-webencodings
- python3-werkzeug
- python38-asn1crypto
- python38-numpy-doc
- python38-psycopg2-doc
- python38-psycopg2-tests

- python39-numpy-doc
- python39-psycopg2-doc
- python39-psycopg2-tests
- qemu-kvm-block-gluster
- qemu-kvm-block-iscsi
- qemu-kvm-block-ssh
- qemu-kvm-hw-usbredir
- qemu-kvm-device-display-virtio-gpu-gl
- qemu-kvm-device-display-virtio-gpu-pci-gl
- qemu-kvm-device-display-virtio-vga-gl
- qemu-kvm-tests
- qpdf
- qpdf-doc
- qperf
- qpid-proton
- qrencode
- qrencode-devel
- qrencode-libs
- qt5-qtcanvas3d
- qt5-qtcanvas3d-examples
- rarian
- rarian-compat
- re2c
- recode
- redhat-lsb
- redhat-lsb-core
- redhat-lsb-cxx
- redhat-lsb-desktop
- redhat-lsb-languages

- redhat-lsb-printing
- redhat-lsb-submod-multimedia
- redhat-lsb-submod-security
- redhat-lsb-supplemental
- redhat-lsb-trialuse
- redhat-menus
- redhat-support-lib-python
- redhat-support-tool
- reflections
- regexp
- relaxngDatatype
- resteasy-javadoc
- rhsm-gtk
- rpm-plugin-prioreset
- rpmemd
- rsyslog-udpspoof
- ruby-hivex
- ruby-libguestfs
- rubygem-abrt
- rubygem-abrt-doc
- rubygem-bson
- rubygem-bson-doc
- rubygem-bundler-doc
- rubygem-mongo
- rubygem-mongo-doc
- rubygem-net-telnet
- rubygem-xmlrpc
- s390utils-cmsfs
- samba-pidl

- samba-test
- samba-test-libs
- samyak-devanagari-fonts
- samyak-fonts-common
- samyak-gujarati-fonts
- samyak-malayalam-fonts
- samyak-odia-fonts
- samyak-tamil-fonts
- sane-frontends
- sanlk-reset
- sat4j
- scala
- scotch
- scotch-devel
- SDL_sound
- selinux-policy-minimum
- sendmail
- sgabios
- sgabios-bin
- shim-ia32
- shrinkwrap
- sil-padauk-book-fonts
- sisu-inject
- sisu-mojos
- sisu-plexus
- skkdic
- SLOF
- smc-anjalioldlipi-fonts
- smc-dyuthi-fonts

- smc-fonts-common
- smc-kalyani-fonts
- smc-raghumalayalam-fonts
- smc-suruma-fonts
- softhsm-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
- systemd-tests
- tcl-brlapi
- testng
- thai-scalable-laksaman-fonts
- tibetan-machine-uni-fonts
- timedatex
- torque-libs
- 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
- voikko-tools
- vorbis-tools
- 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
- zsh-html

8.22. 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 は独自の裁量で、セキュリティーバグを含む重大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8.1 非推奨のデバイス

デバイス ID	ドライバー	デバイス名
	hns_roce	
	ebtables	
	arp_tables	
	ip_tables	
	ip6_tables	
	ip6_set	
	ip_set	
	nft_compat	
	usnic_verbs	
	vmw_pvrdma	
	hfi1	
	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

デバイス ID	ドライバー	デバイス名
0x10df:0xf100	lpfc	Emulex Corporation: LPe12000 Series 8Gb Fibre Channel Adapter
0x10df:0xfc40	lpfc	Emulex Corporation: Saturn-X: LightPulse Fibre Channel Host Adapter
0x10df:0xe220	be2net	Emulex Corporation: OneConnect NIC (Lancer)
0x1000:0x005b	megaraid_sas	Broadcom / LSI: MegaRAID SAS 2208 [Thunderbolt]
0x1000:0x006E	mpt3sas	Broadcom / LSI: SAS2308 PCI-Express Fusion-MPT SAS-2
0x1000:0x0080	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0081	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0082	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0083	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0084	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0085	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0086	mpt3sas	Broadcom / LSI: SAS2308 PCI-Express Fusion-MPT SAS-2
0x1000:0x0087	mpt3sas	Broadcom / LSI: SAS2308 PCI-Express Fusion-MPT SAS-2
	myri10ge	Myricom 10G driver (10GbE)
	netxen_nic	QLogic/NetXen (1/10) GbE Intelligent Ethernet Driver
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)

デバイス ID	ドライバー	デバイス名
	qla3xxx	QLogic ISP3XXX ネットワークドライバー v2.03.00-k5
0x1924:0x0803	sfc	Solarflare Communications: SFC9020 10G Ethernet Controller
0x1924:0x0813	sfc	Solarflare Communications: SFL9021 10GBASE-T Ethernet Controller
	Soft-RoCE (rdma_rxe)	
	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8.2 未管理デバイス

デバイス ID	ドライバー	デバイス名
	dl2k	
	dlci	
	dnet	
	hdlc_fr	
	rdma_rxe	
	nicvf	
	nicpf	
	siw	

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
	mptscsih	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 [a]	megaraid_sas	Broadcom / LSI: MR SAS HBA 2004
0x1000:0x0073 [a]	megaraid_sas	Broadcom / LSI: MegaRAID SAS 2008 [Falcon]
0x1000:0x0079 [a]	megaraid_sas	Broadcom / LSI: MegaRAID SAS 2108 [Liberator]
	nvmet_tcp	NVMe/TCP ターゲットドライバー
	nvmet-fc	NVMe/Fabrics FC target driver

[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9章 既知の問題

ここでは、Red Hat Enterprise Linux 8.10 の既知の問題を説明します。

9.1. インストーラーおよびイメージの作成

IBM Z への RHEL インストール中に、udev が FID によって列挙された RoCE カードに予測可能なインターフェイス名を割り当てない

net.naming-scheme=rhel-8.7 カーネルコマンドラインオプションを使用して RHEL 8.7 以降のインストールを開始すると、RHEL インストールメディア上の **udev** デバイスマネージャーが、機能識別子 (FID) によって列挙された RoCE カードのこの設定を無視します。その結果、**udev** はこのデバイスに予測できないインターフェイス名を割り当てます。インストール中の回避策はありませんが、インストール後に機能を設定できます。詳細は、IBM Z プラットフォームでの予測可能な RoCE デバイス名の決定 を参照してください。

(JIRA:RHEL-11397)

LPAR およびセキュアブートが有効になっている IBM Power 10 システムでのインストールが失敗します

RHEL インストーラーは、IBM Power 10 システムの静的キーセキュアブートと統合されていません。したがって、セキュアブートオプションを使用して論理パーティション (LPAR) を有効にすると、インストールに失敗し、**Unable to proceed with RHEL-x.x Installation** というエラーが表示されます。

この問題を回避するには、セキュアブートを有効にせずに RHEL をインストールします。システムを起動したら、以下を行います。

1. **dd** コマンドを使用して、署名されたカーネルを PReP パーティションにコピーします。
2. システムを再起動し、セキュアブートを有効にします。

ファームウェアがブートローダーとカーネルを検証すると、システムが正常に起動します。

詳細は、https://www.ibm.com/support/pages/node/6528884 を参照してください。

Bugzilla:2025814[1]

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 を実行しないでください。代わりに、一時的な仮想マシンで Anaconda を実行して、実稼働システムの SELinux ポリシーを変更しないようにします。**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。

Bugzilla:2050140

キックスタートコマンドの auth および authconfig で AppStream リポジトリーが必要になる

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 リポジトリーでのみ使用可能です。

この問題を回避するには、BaseOS リポジトリーおよび AppStream リポジトリーがインストールプログラム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。

Bugzilla:1640697[1]

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。

Bugzilla:1697896[1]

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。

Jira:RHEL-4707

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ーを渡しても、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。

Bugzilla:1757877[1]

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設定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。

Jira:RHEL-4711

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。

Bugzilla:2028361[1]

rpm-ostree ペイロードをインストールすると、RHEL for Edge インストーラーイメージがマウントポイントの作成に失敗する

RHEL for Edge インストーラーイメージなどで使用される **rpm-ostree** ペイロードをデプロイする場合、インストーラーはカスタムパーティションの一部のマウントポイントを適切に作成しません。その結果、インストールは以下のエラーで中止されます。

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

この問題を回避するには、以下を実行します。

- 自動パーティション設定スキームを使用し、手動でマウントポイントを追加しないでください。
- マウントポイントは、**/var** ディレクトリー内のみに手動で割り当てます。たとえば、**/var/my-mount-point** や、/、**/boot**、**/var** などの標準ディレクトリーです。

その結果、インストールプロセスは正常に終了します。

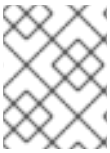
Jira:RHEL-4744

stig プロファイル修復でビルドされたイメージが FIPS エラーで起動に失敗する

FIPS モードは、RHEL Image Builder ではサポートされていません。**xccdf_org.ssgproject.content_profile_stig** プロファイル修復でカスタマイズされた RHEL Image Builder を使用すると、システムは次のエラーで起動に失敗します。

```
Warning: /boot//.vmlinuz-<kernel version>.x86_64.hmac does not exist
FATAL: FIPS integrity test failed
Refusing to continue
```

/boot ディレクトリーが別のパーティションにあるため、システムイメージのインストール後に **fips-mode-setup --enable** コマンドを使用して FIPS ポリシーを手動で有効にしても機能しません。FIPS が無効になっている場合、システムは正常に起動します。現在、使用可能な回避策はありません。



注記

イメージのインストール後に、**fips-mode-setup --enable** コマンドを使用して、FIPS を手動で有効にすることができます。

Jira:RHEL-4649

9.2. セキュリティー

OpenSC が CardOS V5.3 カードオブジェクトを正しく検出しない可能性がある

OpenSC ツールキットは、一部の CardOS V5.3 カードで使用されているさまざまな PKCS #15 ファイルオフセットからキャッシュを正しく読み取りません。その結果、OpenSC はカードオブジェクトをリストできず、別のアプリケーションからカードオブジェクトを使用できなくなる可能性があります。

この問題を回避するには、**/etc/opensc.conf** ファイルで **use_file_caching = false** オプションを設定してファイルキャッシュをオフにします。

Jira:RHEL-4077

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。

Bugzilla:2044354[1]

インストール中にシステムをハードニングすると、RHV ハイパーバイザーが正しく動作しない可能性がある

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 アドオンによってシステムが RVH-H ではなく RHEL としてハードニングされ、RHV-H の必須パッケージが削除される可能性があります。その結果、RHV ハイパーバイザーが機能しない可能性があります。この問題を回避するには、プロファイルのハードニングを適用せずに 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。

Jira:RHEL-1826

CVE OVAL フィードが圧縮形式のみになり、データストリームが SCAP 1.3 標準に準拠していない

Red Hat は、CVE OVAL フィードを bzip2 圧縮形式で提供しています。これらは XML ファイル形式では利用できなくなりました。圧縮されたコンテンツの参照は Security Content Automation Protocol (SCAP) 1.3 仕様で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ルールのスキャンで問題が発生する可能性があります。

Bugzilla:2028428

特定の Rsyslog 優先度文字列が正しく機能しない

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。そのため、次の優先度文字列は、Rsyslog リモートログアプリケーションでは正しく機能しません。

```
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。

Bugzilla:1679512

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。

Bugzilla:1843932

キックスタートインストール時のサービス関連ルールの修正が失敗する場合がある

キックスタートインストール時に、OpenSCAP ユーティリティーで、サービスの **enable** または **disable** 状態の修正は不要であると誤って表示されることがあります。その結果、OpenSCAP によって、インストール済みシステム上のサービスが非準拠状態に設定され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。

Bugzilla:1834716

RHEL 8 のキックスタートが、com_redhat_oscap の代わりに org_fedora_oscap を使用

キックスタートは、**com_redhat_oscap** ではなく、**org_fedora_oscap** 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 との互換性を維持するために必要です。

Bugzilla:1665082[1]

libvirt が xccdf_org.ssgproject.content_rule_sysctl_net_ipv4_conf_all_forwarding をオーバーライドする

libvirt 仮想化フレームワークは、**route** または **nat** の転送モードを持つ仮想ネットワークが起動するたびに、IPv4 転送を有効にします。これにより、**xccdf_org.ssgproject.content_rule_sysctl_net_ipv4_conf_all_forwarding** ルールによる設定がオーバーライドされ、後続のコンプライアンススキャンでは、このルールを評価するときに **fail** という結果が報告されます。

この問題を回避するには、次のいずれかのシナリオを適用します。

- シナリオで必要がない場合は、**libvirt** パッケージをアンインストールします。
- **libvirt** によって作成された仮想ネットワークの転送モードを変更します。
- プロファイルを調整して、**xccdf_org.ssgproject.content_rule_sysctl_net_ipv4_conf_all_forwarding** ルールを削除します。

Bugzilla:2118758

fapolicyd ユーティリティーは、変更されたファイルの実行を誤って許可する

正しくは、ファイルの IMA ハッシュはファイルに変更が加えられた後に更新され、**fapolicyd** は変更されたファイルの実行を阻止する必要があります。ただし、IMA ポリシーのセットアップと **evctml** ユーティリティーによるファイルハッシュの違いにより、これは起こりません。その結果、変更されたファイルの拡張属性で IMA ハッシュは更新されません。その結果、**fapolicyd** は、変更されたファイルの実行を誤って許可します。

Jira:RHEL-520[1]

semanage fcontext コマンドはローカルの変更を並べ替える

semanage fcontext -l -C コマンドは、**file_contexts.local** ファイルに保存されているローカルファイルコンテキストの変更をリスト表示します。**restorecon** ユーティリティーは、**file_contexts.local** 内のエントリーを最も新しいエントリーから最も古いエントリーへと順番に処理します。ただし、**semanage fcontext -l -C** はエントリーを異なる順序でリストします。処理順序とリスト表示順序の不一致により、SELinux ルールの管理時に問題が発生する可能性があります。

Jira:RHEL-24461[1]

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、OpenSSL を使用する TLS クライアントが **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サーバーへの TLS 接続をキャンセルします。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。

Bugzilla:1810911[1]

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。

Bugzilla: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。

Bugzilla: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、GnuTLS 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。

Bugzilla:1628553[1]

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 リポジトリーで **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、特定のモジュールを使用して **libselinux-python** をインストールできます。

Bugzilla:1666328[1]

udica は、--env container=podman で開始したときにのみ UBI 8 コンテナーを処理します。

Red Hat Universal Base Image 8 (UBI 8) コンテナーは、**podman** の値ではなく、**コンテナー** 環境変数を **oci** 値に設定します。これにより、**udica** ツールがコンテナー 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 コンテナーを起動します。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 コンテナーの SELinux ポリシーを生成することができます。

Bugzilla:1763210

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[1]

/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[1]

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。

Bugzilla:1989050

scap-security-guide がアイドルセッションの終了を設定できない

sshd_set_idle_timeout ルールはデータストリームにまだ存在しますが、**sshd** を設定するアイドル

セッションタイムアウトの以前の方法は使用できなくなりました。したがって、ルールは **applicable** としてマークされるため、何も強化できません。**systemd** (Logind) など、アイドルセッションの終了を設定する他の方法も使用できません。そのため、**scap-security-guide** は、一定時間が経過した後にアイドルセッションを確実に切断するようにシステムを設定できません。

この問題は、次のいずれかの方法で回避できます。これにより、セキュリティー要件を満たせる可能性があります。

- **accounts_tmout** ルールを設定します。ただし、この変数は **exec** コマンドを使用してオーバーライドできます。
- **configure_tmux_lock_after_time** ルールと **configure_bashrc_exec_tmux** ルールを設定します。これには、**tmux** パッケージをインストールする必要があります。
- 適切な SCAP ルールとともに **systemd** 機能がすでに実装されている RHEL 8.7 以降にアップグレードします。

Jira:RHEL-1804

OSCAP Anaconda アドオンは、グラフィカルインストールで調整されたプロファイルをフェッチしない

OSCAP Anaconda アドオンには、RHEL グラフィカルインストールでセキュリティープロファイルの調整を選択または選択解除するオプションがありません。RHEL 8.8 以降、アドオンはアーカイブまたは RPM パッケージからインストールするときにデフォルトで調整を考慮しません。その結果、インストールでは、OSCAP に合わせたプロファイルを取得する代わりに、次のエラーメッセージが表示されます。

> There was an unexpected problem with the supplied content.

この問題を回避するには、キックスタートファイルの **%addon org_fedora_oscap** セクションにパスを指定する必要があります。次に例を示します。

```
xccdf-path = /usr/share/xml/scap/sc_tailoring/ds-combined.xml
tailoring-path = /usr/share/xml/scap/sc_tailoring/tailoring-xccdf.xml
```

その結果、OSCAP 調整プロファイルのグラフィカルインストールは、対応するキックスタート仕様のみで使用できます。

Jira:RHEL-1810

OpenSCAP のメモリー消費の問題

メモリーが限られているシステムでは、OpenSCAP スキャナが途中で停止するか、結果ファイルが生成されない可能性があります。この問題を回避するには、スキャンプロファイルをカスタマイズして、/ ファイルシステム全体の再帰を含むルールの選択を解除します。

- **rpm_verify_hashes**
- **rpm_verify_permissions**
- **rpm_verify_ownership**
- **file_permissions_unauthorized_world_writable**
- **no_files_unowned_by_user**

- **dir_perms_world_writable_system_owned**
- **file_permissions_unauthorized_suid**
- **file_permissions_unauthorized_sgid**
- **file_permissions_ungroupowned**
- **dir_perms_world_writable_sticky_bits**

詳細とその他の回避策は、関連する ナレッジベースの記事 を参照してください。

Bugzilla:2161499

rpm データベースを再構築すると、間違った SELinux ラベルが割り当てられる

rpmdb --rebuilddb コマンドを使用して **rpm** データベースを再構築すると、誤った SELinux ラベルが **rpm** データベースファイルに割り当てられます。その結果、**rpm** データベースを使用する一部のサービスが正しく動作しない可能性があります。データベースの再構築後にこの問題を回避するには、**restorecon -Rv /var/lib/rpm** コマンドを使用してデータベースのラベルを再設定します。

Bugzilla:2166153

Audit 用の ANSSI BP28 HP SCAP ルールが 64 ビット ARM アーキテクチャーで誤って使用される

SCAP セキュリティーガイド (SSG) の ANSSI BP28 High プロファイルには、Linux Audit サブシステムを設定する次の Security Content Automation Protocol (SCAP) ルールが含まれています。しかし、これらのルールは、64 ビット ARM アーキテクチャーでは無効です。

- **audit_rules_unsuccessful_file_modification_creat**
- **audit_rules_unsuccessful_file_modification_open**
- **audit_rules_file_deletion_events_rename**
- **audit_rules_file_deletion_events_rmdir**
- **audit_rules_file_deletion_events_unlink**
- **audit_rules_dac_modification_chmod**
- **audit_rules_dac_modification_chown**
- **audit_rules_dac_modification_lchown**

このプロファイルを使用して 64 ビット ARM マシン上で実行される RHEL システムを設定すると、無効なシステムコールが使用されているため、Audit デーモンが起動しません。

この問題を回避するには、プロファイルの調整を使用して前述のルールをデータストリームから削除するか、**/etc/audit/rules.d** ディレクトリー内のファイルを編集して **-S <syscall>** スニペットを削除します。ファイルに次のシステムコールを含めることはできません。

- creat
- open

- rename
- rmdir
- unlink
- chmod
- chown
- lchown

上記の 2 つの回避策のいずれかを実行すると、64 ビット ARM システムで ANSSI BP28 High プロファイルを使用した後でも、Audit デーモンが起動できるようになります。

Jira:RHEL-1897

9.3. RHEL FOR EDGE

composer-cli は、nodejs または npm が含まれていると、RHEL for Edge イメージのビルドに失敗する

現在、RHEL イメージビルダーを使用している場合は、**nodejs** パッケージを使用して RHEL for Edge イメージをビルドすることはできないため、**nodejs** および **npm** パッケージを使用して RHEL 8 Edge イメージをカスタマイズすることはできません。NPM パッケージマネージャーは、**{prefix}/etc/npmrc** ディレクトリー内に設定があることを想定しており、npm RPM パッケージは、**/etc/npmrc** を指す **/usr/etc/npmrc** ディレクトリーにシンボリックリンクをパッケージ化します。この問題を回避するには、RHEL for Edge システムをビルドした後、**nodejs** および **npm** パッケージをインストールします。

Jira:RHELDOCS-17126[1]

9.4. サブスクリプションの管理

syspurpose addons が subscription-manager attach --auto 出力に影響しない

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。

Bugzilla:1687900

9.5. ソフトウェア管理

YUM 機能またはプラグインは、ログサービスが利用できない場合でもメッセージをログに記録することがあります。

ログサービスが利用できない場合は、特定の YUM 機能またはプラグインによってメッセージが標準出力または標準エラーに記録されることがあります。ログメッセージのレベルは、メッセージが記録される場所を示します。

- 情報メッセージは標準出力に記録されます。

- エラーおよびデバッグメッセージは標準エラーに記録されます。

その結果、**YUM** オプションをスクリプト化する場合は、標準出力または標準エラーに不要なログメッセージが出力され、スクリプトの機能に影響する可能性があります。

この問題を回避するには、**yum -q** コマンドを使用して、標準出力と標準エラーからのログメッセージを抑制します。これにより、ログメッセージは抑制されますが、標準出力で期待されるコマンド結果は抑制されません。

Jira:RHELPLAN-50409[1]

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 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い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。

Bugzilla:1973588[1]

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。

Bugzilla:1986657

9.6. シェルおよびコマンドラインツール

ipmitool は特定のサーバープラットフォームと互換性がありません

ipmitool ユーティリティーは、Intelligent Platform Management Interface (IPMI) をサポートするデバイスの監視、設定、および管理に役立ちます。現在のバージョンの **ipmitool** は、以前の Cipher Suite 3 の代わりに Cipher Suite 17 をデフォルトで使用します。その結果、**ipmitool** は、ネゴシエーション中に Cipher Suite 17 のサポートを発表しましたが、実際にはこの暗号スイートをサポートしていない特定のベアメタルノードとの通信に失敗します。その結果、**ipmitool** が停止し、**no matching cipher suite** というエラーメッセージが表示されます。

詳細は、関連する ナレッジベースの記事 を参照してください。

この問題を解決するには、ベースボード管理コントローラー (BMC) ファームウェアを更新して、Cipher Suite 17 を使用します。

オプションで、BMC ファームウェアの更新が利用できない場合は、**ipmitool** に特定の暗号スイートを強制的に使用させることで、この問題を回避できます。**ipmitool** で管理タスクを呼び出す場合は、使用する暗号スイートの **番号** とともに **ipmitool** コマンドに **-C** オプションを追加します。以下の例を参照してください。

```
# ipmitool -I lanplus -H myserver.example.com -P mypass -C 3 chassis power status
```

Jira:RHEL-6846

復元にクリーンディスクを使用しないと、ReaR がボリュームグループの再作成に失敗する

既存のデータを含むディスクに復元する場合、ReaR は復元の実行に失敗します。

この問題を回避するには、ディスクが以前に使用されていた場合、復元する前にディスクを手動でワイプします。レスキュー環境でディスクをワイプするには、**rear recover** コマンドを実行する前に、次のいずれかのコマンドを使用します。

- ディスクを上書きする **dd** コマンド。
- 使用可能なすべてのメタデータを消去するには、**-a** フラグを指定した **wipefs** コマンド。

/dev/sda ディスクからメタデータをワイプする次の例を参照してください。

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

このコマンドは、最初に **/dev/sda** のパーティションからメタデータをワイプし、次にパーティションテーブル自体をワイプします。

Bugzilla:1925531

セキュアブートが有効になっている UEFI システム上の ReaR レスキューイメージは、デフォルト設定では起動に失敗する

rear mkrescue または **rear mkbackup** コマンドを使用した ReaR イメージの作成が失敗し、次のメッセージが表示されます。

```
grub2-mkstandalone might fail to make a bootable EFI image of GRUB2 (no /usr/*/grub*/x86_64-efi/moddep.lst file)
(...)
grub2-mkstandalone: error: /usr/lib/grub/x86_64-efi/modinfo.sh doesn't exist. Please specify --target or --directory.
```

不足しているファイルは、**grub2-efi-x64-modules** パッケージの一部です。このパッケージをインストールすると、エラーなしでレスキューイメージが正常に作成されます。**UEFI** セキュアブートが有効になっている場合は、レスキューイメージは署名されていないブートローダーを使用するため起動できません。

この問題を回避するには、**/etc/rear/local.conf** または **/etc/rear/site.conf** ReaR 設定ファイルに次の変数を追加します。

```
UEFI_BOOTLOADER=/boot/efi/EFI/redhat/grubx64.efi
SECURE_BOOT_BOOTLOADER=/boot/efi/EFI/redhat/shimx64.efi
```

提案された回避策を使用すると、**grub2-efi-x64-modules** パッケージのないシステムでもイメージを正常に生成でき、セキュアブートが有効になっているシステムで起動できるようになります。さらに、システムのリカバリー中に、リカバリーされたシステムのブートローダーは **EFI** shim ブートローダーに設定されます。

UEFI、セキュアブート、shim ブートローダー の詳細は、ナレッジベースの記事 UEFI: what happens when booting the system を参照してください。

Jira:RHELDOCS-18064[1]

coreutils が誤解を招く EPERM エラーコードを報告する可能性がある

GNU Core Utilities (**coreutils**) が、**statx()** システムコールを使用するようになりました。**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、**coreutils** が誤解を招く EPERM エラーコードを報告する可能性があります。EPERM は、動作中の **statx()** システムコールが返す実際の **Operation not permitted** エラーと区別できないためです。

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。

Bugzilla:2030661

sysstat パッケージの %vmeff メトリックに誤った値が表示される

sysstat パッケージは、ページ再利用効率を測定するための **%vmeff** メトリックを提供します。**sysstat** は、新しいカーネルバージョンで提供されるすべての関連する **/proc/vmstat** 値を解析しないため、**sar -B** コマンドによって返される **%vmeff** 列の値は正しくありません。この問題を回避するには、**/proc/vmstat** ファイルから **%vmeff** 値を手動で計算します。詳細は、Why the **sar(1)** tool reports **%vmeff** values beyond 100 % in RHEL 8 and RHEL 9? を参照してください。

Jira:RHEL-12008

sar および iostat ユーティリティーによって生成された %util 列および svctm 列は無効です

カーネルバージョン **4.18.0-55.el8** 以降のシステムで **sar** または **iostat** ユーティリティーを使用してシステム使用状況統計を収集すると、**sar** または **iostat** によって生成された **%util** 列と **svctm** 列に無効なデータが含まれる場合があります。

Jira:RHEL-23074[1]

9.7. 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 ハッシュ関数が使用できないため、デフォルトの postfix 設定で TLS が正しく機能しない可能性があります。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。

Bugzilla: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。

Bugzilla:2008197

9.8. ネットワーク

negative_advice() 関数を使用する古いサードパーティーモジュールがカーネルをクラッシュさせる可能性がある

コアネットワーク操作 **negative_advice()** は、**dst_negative_advice()** および **__dst_negative_advice()** インライン関数を呼び出します。RHEL 8.10 のカーネルで、これらのインライン関数のセキュリティー問題 (CVE-2024-36971) が修正されました。修正前にサードパーティーのモジュールがコンパイルされていた場合、このモジュールは **negative_advice()** を誤って呼び出す可能性があります。その結果、サードパーティーのモジュールがカーネルをクラッシュさせる可能性があります。この問題を解決するには、**negative_advice()** 関数を正しく呼び出す更新されたモジュールを使用します。

Jira:RHELDOCS-18748

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=!net.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。

Jira:RHEL-11398[1]

IPv6_rpfilter オプションが有効になっているシステムでネットワークスループットが低下

firewalld.conf ファイルで **IPv6_rpfilter** オプションが有効になっているシステムでは、100 Gbps リンクなどの高いトラフィックシナリオの場合、現時点でパフォーマンスは最適ではなくネットワークスループットが低下します。この問題を回避するには、**IPv6_rpfilter** オプションを無効にします。これを行うには、**/etc/firewalld/firewalld.conf** ファイルに次の行を追加します。

```
IPv6_rpfilter=no
```

その結果、システムはパフォーマンスが向上しますが、同時にセキュリティーは低下します。

Bugzilla:1871860[1]

9.9. カーネル

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、ナレッジベースソリューション "Firmware Bug: ECAM area **mem 0x30000000-0x31ffffff** not reserved in ACPI namespace" appears during system boot を参照してください。

Bugzilla:1868526[1]

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) プロセッサーを持つ 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。

Bugzilla:1609288[1]

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

特定に場合において、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。

Bugzilla:1602962[1]

同一の crash 拡張機能を再ロードすると、セグメンテーション違反が発生する可能性がある

すでにロードされている crash 拡張機能ファイルのコピーをロードすると、セグメンテーション違反が発生する可能性があります。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。

Bugzilla:1906482

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。

Bugzilla:1930576[1]

OPEN MPI ライブラリーがデフォルトの PML で実行時エラーを引き起こす可能性がある

OPEN Message Passing Interface (OPEN MPI) 実装 4.0.x シリーズでは、UCX (Unified Communication X) がデフォルトの PML (ポイントツーポイント) です。OPEN MPI 4.0.x シリーズの新しいバージョンでは、**openib** Byte Transfer Layer (BTL) が非推奨になりました。

ただし、OPEN MPI は **同種** クラスター (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 **openib** の一方向操作に BTL を使用します。その結果、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** クラスター (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。その結果、OPEN MPI ジョブの実行時に、不安定なパフォーマンス、応答しない動作、またはクラッシュによる障害が発生す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。

Bugzilla:1866402[1]

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。

Bugzilla:1793389[1]

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 Graviton 1 プロセッサー上で実行される 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 プロセッサーでは、**/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る を参照してください。

Bugzilla:1654962[1]

コア数が多いシステム上のリアルタイムカーネルのハードウェア認定には、skew-tick=1 ブートパラメーターを渡すことが必要になる場合がある

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、マルチプロセッシングシステムでは、レイテンシーの急増やハードウェア認定の遅延が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. ブート中に渡すカーネルパラメーターを表示して、新しい設定を確認します。

```
cat /proc/cmdline
```

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。

Jira:RHEL-9318[1]

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。

Bugzilla:1659609[1]

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。

Bugzilla:1940674[1]

QAT マネージャーが LKCF のスペアデバイスを残さない

Intel® QuickAssist Technology(QAT) マネージャー (**qatmgr**) はユーザー空間プロセスであり、デフォルトではシステム内のすべての QAT デバイスを使用します。これにより、Linux Kernel Cryptographic Framework(LKCF) には QAT デバイスが残っていません。この動作は予想され、大多数のユーザーはユーザースペースからのアクセラレーションを使用するため、この状況を回避する必要はありません。

Bugzilla:1920086[1]

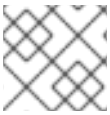
Solarflare が、最大数の VF (Virtual Function) の作成に失敗する

Solarflare NIC は、リソースが十分にないため、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。

Bugzilla:1971506[1]

page_poison=1 を使用すると、カーネルクラッシュが発生する可能性がある

EFI 実装に問題のあるファームウェアでカーネルパラメーターとして **page_poison=1** を使用すると、オペレーティングシステムが原因でカーネルがクラッシュする可能性があります。デフォルトでは、このオプションは無効になっており、特に実稼働システムでは有効にすることは推奨しません。

Bugzilla:2050411[1]

iwl7260-firmware により、Intel Wi-Fi 6 AX200、AX210、および Lenovo ThinkPad P1 Gen 4 で Wi-Fi が切断される

iwl7260-firmware または **iwl7260-wifi** ドライバーを RHEL 8.7 以降で提供されるバージョンに更新すると、ハードウェアが不正な内部状態になり、その状態を誤って報告します。その結果、Intel Wifi 6 カードが機能せず、次のエラーメッセージが表示される場合があります。

```
kernel: iwlwifi 0000:09:00.0: Failed to start RT ucode: -110
kernel: iwlwifi 0000:09:00.0: WRT: Collecting data: ini trigger 13 fired (delay=0ms)
kernel: iwlwifi 0000:09:00.0: Failed to run INIT ucode: -110
```

未確認の回避策は、システムの電源をオフにしてから再度オンにすることです。再起動しないでください。

Bugzilla:2106341[1]

IBM Power Systems のセキュアブートは移行をサポートしていません

現在、IBM Power Systems では、物理ボリューム (PV) の移行が成功した後、論理パーティション (LPAR) が起動しません。その結果、パーティションでセキュアブートが有効になっているタイプの自動移行は失敗します。

Bugzilla:2126777[1]

kmod の weak-modules がモジュールの相互依存関係で機能しない

kmod パッケージによって提供される **weak-modules** スクリプトは、どのモジュールがインストールされたカーネルと kABI 互換であるかを判別します。しかし、**weak-modules** は、モジュールのカーネル互換性をチェックする際に、モジュールシンボルの依存関係を、そのビルド対象のカーネルの新しい

リリースから古いリリースの順に処理します。結果として、異なるカーネルリリースに対して構築された相互依存関係を持つモジュールは互換性がないと解釈される可能性があるため、**weak-modules** はこのシナリオでは機能しません。

この問題を回避するには、新しいカーネルをインストールする前に、最新のストックカーネルに対して追加のモジュールをビルドまたは配置します。

Bugzilla:2103605[1]

Ampere Altra サーバーの kdump が OOM 状態になる

現在、Ampere Altra および Altra Max サーバーのファームウェアが原因で、カーネルが大量のイベント、割り込み、およびコマンドキューを割り当て、メモリーを大量に消費します。その結果、**kdump** カーネルがメモリー不足 (OOM) 状態になります。

この問題を回避するには、**crashkernel=** カーネルオプションの値を 640M に増やして、**kdump** 用に追加のメモリーを予約します。

Bugzilla:2111855[1]

9.10. ファイルシステムおよびストレージ

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。

Bugzilla:1730502[1]

/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。

Bugzilla:1496229[1]

LVM で、複数のブロックサイズを持つボリュームグループが作成できない

vgcreate または **vgextend** などの LVM ユーティリティーでは、物理ボリューム (PV) の論理ブロックサイズが異なるボリュームグループ (VG) を作成できなくなりました。別のブロックサイズの PV で基礎となる論理ボリューム (LV) を拡張するとファイルシステムがマウントに失敗するため、LVM はこの変更を採用しました。

ブロックサイズが混在する VG の作成を再度有効にするには、**lvm.conf** ファイルの **allow_mixed_block_sizes=1** オプションを設定します。

Bugzilla:1768536

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[1]、Bugzilla:1798631、Bugzilla:1808012

IOMMU を有効にするとシステムがパニックになる

intel_iommu パラメーターを **on** に設定してカーネルコマンドラインで入出力メモリー管理ユニット (IOMMU) を有効にすると、**0x6b6b6b6b6b6b6b6b: 0000** 非正規アドレスの一般保護違反が発生し、システムパニックが発生します。

この問題を回避するには、**intel_iommu** が **off** に設定されていることを確認してください。

Jira:RHEL-1765[1]

NVMe/TCP ドライバーを使用する場合、デバイスマッパーマルチパスがサポートされない

NVMe/TCP デバイス上でデバイスマッパーマルチパスを使用すると、パフォーマンスとエラー処理が低下する可能性があります。この問題を回避するには、DM マルチパスツールの代わりにネイティブ NVMe マルチパスを使用します。RHEL 8 の場合、カーネルコマンドラインにオプション **nvme_core.multipath=Y** を追加できます。

Bugzilla:2022359[1]

blk-availability systemd サービスは、複雑なデバイススタックを非アクティブ化する

systemd では、デフォルトのブロック非アクティブ化コードは、仮想ブロックデバイスの複雑なスタックを常に正しく処理するとは限りません。一部の設定では、シャットダウン中に仮想デバイスが削除されない場合があり、エラーメッセージがログに記録されます。この問題を回避するには、次のコマンドを実行して、複雑なブロックデバイススタックを非アクティブ化します。

```
# systemctl enable --now blk-availability.service
```

その結果、複雑な仮想デバイススタックはシャットダウン中に正しく非アクティブ化され、エラーメッセージは生成されません。

Bugzilla:2011699[1]

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。

Bugzilla:2059262[1]

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

Git で所有権が安全でない可能性のあるリポジトリーからのクローン作成や取得が失敗する

リモートコード実行を防ぎ、CVE-2024-32004 の影響を軽減するために、**Git** でのローカルリポジトリーのクローン作成に対して、より厳格な所有権チェックが導入されました。RHSA-2024:4084 アドバイザリーで導入された更新により、所有権が安全でない可能性があるローカルリポジトリーが、**Git** で不審なものとして扱われるようになりました。

その結果、ユーザーが **git-daemon** を通じてローカルにホストされているリポジトリーからクローン作成を試行する際に、そのユーザーがリポジトリーの所有者でない場合、**Git** が不審な所有権に関するセキュリティー警告を返し、リポジトリーからのクローン作成または取得が失敗するようになりました。

この問題を回避するには、次のコマンドを実行して、リポジトリーを明示的に安全としてマークします。

```
git config --global --add safe.directory /path/to/repository
```

Jira:RHELDOCS-18435[1]

virtualenv ユーティリティーを使用すると Python 3.11 仮想環境の作成が失敗する

python3-virtualenv パッケージによって提供される RHEL 8 の **virtualenv** ユーティリティーは、Python 3.11 と互換性がありません。**virtualenv** を使用して仮想環境を作成しようとすると、次のエラーメッセージが表示されて失敗します。

```
$ virtualenv -p python3.11 venv3.11
Running virtualenv with interpreter /usr/bin/python3.11
```

```
ERROR: Virtual environments created by virtualenv < 20 are not compatible with Python 3.11.
ERROR: Use `python3.11 -m venv` instead.
```

Python 3.11 仮想環境を作成するには、代わりに **python3.11 -m venv** コマンドを使用します。このコマンドは、標準ライブラリーの **venv** モジュールを使用します。

Bugzilla:2165702

python3.11-lxml が lxml.isoschematron サブモジュールを提供しない

python3.11-lxml パッケージは、オープンソースライセンスの下にないため、**lxml.isoschematron** サブモジュールなしで配布されます。サブモジュールは ISO Schematron サポートを実装します。代わりに、ISO-Schematron 前の検証を **lxml.etree.Schematron** クラスで利用できます。**python3.11-lxml** パッケージの残りのコンテンツは影響を受けません。

Bugzilla:2157673

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。

Bugzilla: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。その結果、**mod_security** または **mod_auth_openidc** モジュールも **httpd** 設定によってロードされると、PHP **ldap** 拡張機能を使用する Apache **httpd** 子プロセスが予期せず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。

Bugzilla:1819607[1]

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。

Bugzilla:1803161

9.12. IDENTITY MANAGEMENT

Samba をプリントサーバーとして実行し、RHEL 8.4 以前から更新する場合にアクションが必要です

今回の更新で、**samba** パッケージが **/var/spool/samba/** ディレクトリーを作成しなくなりました。プリントサーバーとして Samba を使用し、**[printers]** 共有の **/var/spool/samba/** を使用してプリントジョブをスプールすると、SELinux は Samba ユーザーがこのディレクトリーにファイルを作成しないようにします。したがって、印刷ジョブが失敗し、**auditd** サービスは **/var/log/audit/audit.log** に **denied** メッセージを記録します。8.4 以前からシステムを更新した後にこの問題を回避するには、以下を行います。

1. **/etc/samba/smb.conf** ファイルで **[printers]** 共有を探します。
2. 共有定義に **path = /var/spool/samba/** が含まれる場合は、設定を更新して、**path** パラメーターを **/var/tmp/** に設定します。
3. **smbd** サービスを再起動します。

```
# systemctl restart smbd
```

Samba を RHEL 8.5 以降に新しくインストールした場合、アクションは不要です。その場合、**samba-common** パッケージが提供するデフォルトの **/etc/samba/smb.conf** ファイルは、すでに **/var/tmp/** ディレクトリーを使用してプリントジョブをスプールします。

Bugzilla:2009213[1]

--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、Certificate System 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。

Bugzilla:1729215

FIPS モードは、共有シークレットを使用したフォレスト間の信頼を確立することをサポートしません。

NTLMSSP 認証は FIPS に準拠していないため、FIPS モードでフォレスト間の信頼を確立できません。この問題を回避するには、FIPS モードが有効な IdM ドメインと AD ドメインとの間に信頼を確立する際に、Active Directory (AD) 管理アカウントで認証します。

Jira:RHEL-4847

バージョン 1.2.2 へのリベース後の authselect のダウングレードにより、システム認証の破損

authselect パッケージが、最新のアップストリームバージョン **1.2.2** にリベースされました。**authselect** のダウングレードはサポートされておらず、**root** を含むすべてのユーザーに対してシステム認証が破損しています。

authselect パッケージを **1.2.1** 以前にダウングレードした場合は、この問題を回避するために以下の手順を実行します。

1. GRUB ブート画面で、起動するカーネルのバージョンを含む **Red Hat Enterprise Linux** を選択し、**e** を押してエントリーを編集します。

2. **linux** で始まる行の末尾で、**single** を、別の単語で入力し、**Ctrl+X** を押して起動プロセスを開始します。
3. シングルユーザーモードでの起動時に、root パスワードを入力します。
4. 以下のコマンドを使用して authselect 設定を復元します。

```
# authselect select sssd --force
```

Bugzilla:1892761

IdM から AD へのレルム間の TGS 要求が失敗します

IdM Kerberos チケットの特権属性証明書 (PAC) 情報は、Active Directory (AD) でサポートされていない AES SHA-2 HMAC 暗号化で署名されるようになりました。

その結果、IdM から AD へのレルム間 TGS 要求 (双方向の信頼の設定) は、以下のエラーを出して失敗します。

```
Generic error (see e-text) while getting credentials for <service principal>
```

Jira:RHEL-4910

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/sssd/sssd.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[1]

RHEL 8.6 から RHEL 8.7 以降への pki-core-debuginfo の更新が失敗する

RHEL 8.6 から RHEL 8.7 以降への **pki-core-debuginfo** パッケージの更新が失敗します。この問題を回避するには、以下のコマンドを実行します。

1. **yum remove pki-core-debuginfo**
2. **yum update -y**
3. **yum install pki-core-debuginfo**
4. **yum install idm-pki-symkey-debuginfo idm-pki-tools-debuginfo**

Jira:RHEL-13125[1]

ドメイン SID の不一致により、移行した IdM ユーザーがログインできない可能性がある

ipa migrate-ds スクリプトを使用して IdM デプロイメントから別のデプロイメントにユーザーを移行する場合、そのユーザーの以前のセキュリティー識別子 (SID) には現在の IdM 環境のドメイン SID がないため、ユーザーが IdM サービスを使用する際に問題が発生する可能性があります。たとえば、これらのユーザーは **kinit** ユーティリティーを使用して Kerberos チケットを取得できますが、ログインできません。この問題を回避するには、ナレッジベースの記事 Migrated IdM users unable to log in due to mismatching domain SIDs を参照してください。

Jira:RHELPLAN-109613[1]

FIPS モードの IdM は、双方向のフォレスト間信頼を確立するための NTLMSSP プロトコルの使用をサポートしない

FIPS モードが有効な Active Directory (AD) と Identity Management (IdM) との間で双方向のフォレスト間の信頼を確立すると、New Technology LAN Manager Security Support Provider (NTLMSSP) 認証が FIPS に準拠していないため、失敗します。FIPS モードの IdM は、認証の試行時に AD ドメインコントローラーが使用する RC4 NTLM ハッシュを受け入れません。

Jira:RHEL-4898

Kerberos プリンシパルの有効期限を設定する際の誤った警告

Kerberos プリンシパルのパスワード有効期限を設定すると、32 ビットの符号付き整数変数を使用して、現在のタイムスタンプが有効期限のタイムスタンプと比較されます。有効期限が 68 年以上先の場合、整数変数のオーバーフローが発生し、次の警告メッセージが表示されます。

```
Warning: Your password will expire in less than one hour on [expiration date]
```

このメッセージは無視しても問題ありません。パスワードは設定された日時に正しく期限切れになります。

Bugzilla:2125318

RHEL 8 で NIS マップ内の多数のエントリーを列挙するのが遅い

RHEL 8 に **nis_nss** パッケージをインストールした場合、**/etc/default/NSS** 設定ファイルがありません。このファイルは、**glibc-common** パッケージによって提供されなくなったためです。その結果、RHEL 8 では NIS マップ内の多数のエントリーを列挙するのに、RHEL 7 よりも大幅に時間がかかります。すべての要求が、デフォルトで一括ではなく個別に処理されるためです。

この問題を回避するには、次の内容の **/etc/default/nss** ファイルを作成し、**SETENT_BATCH_READ** 変数を **TRUE** に設定してください。

```
# /etc/default/nss
# This file can theoretically contain a bunch of customization variables
# for Name Service Switch in the GNU C library. For now there are only
# four variables:
#
# NETID_AUTHORITATIVE
# If set to TRUE, the initgroups() function will accept the information
# from the netid.byname NIS map as authoritative. This can speed up the
# function significantly if the group.byname map is large. The content
# of the netid.byname map is used AS IS. The system administrator has
# to make sure it is correctly generated.
#NETID_AUTHORITATIVE=TRUE
#
```

```
# SERVICES_AUTHORITATIVE
# If set to TRUE, the getservbyname{,_r}() function will assume
# services.byservicename NIS map exists and is authoritative, particularly
# that it contains both keys with /proto and without /proto for both
# primary service names and service aliases. The system administrator
# has to make sure it is correctly generated.
#SERVICES_AUTHORITATIVE=TRUE
#
# SETENT_BATCH_READ
# If set to TRUE, various setXXent() functions will read the entire
# database at once and then hand out the requests one by one from
# memory with every getXXent() call. Otherwise each getXXent() call
# might result into a network communication with the server to get
# the next entry.
SETENT_BATCH_READ=TRUE
#
# ADJUNCT_AS_SHADOW
# If set to TRUE, the passwd routines in the NIS NSS module will not
# use the passwd.adjunct.byname tables to fill in the password data
# in the passwd structure. This is a security problem if the NIS
# server cannot be trusted to send the passwd.adjuct table only to
# privileged clients. Instead the passwd.adjunct.byname table is
# used to synthesize the shadow.byname table if it does not exist.
#ADJUNCT_AS_SHADOW=TRUE
```

Jira:RHEL-34075[1]

新しいオプション local_auth_policy の導入後、スマートカード認証の設定更新が必要になる場合がある

RHEL 8.10 に更新すると、**local_auth_policy** オプションによって導入される変更により、スマートカード認証が失敗する可能性があります。**local_auth_policy** がデフォルト値 **match** に設定されていると、SSSD によって、オフライン認証方法がオンラインで利用可能な方法に制限されます。その結果、たとえば **auth_provider = ldap** を使用する場合など、設定されているバックエンドによってスマートカード認証が提供されない場合、ユーザーがスマートカード認証を利用できなくなります。この問題を回避するには、**sssd.conf** ファイルのドメインセクションに **local_auth_policy = enable:smartcard** を追加してスマートカード認証方法を明示的に有効にし、SSSD を再起動します。

Jira:RHELDOCS-18777

グループのサイズが 1500 人を超えると、SSSD が取得するメンバーリストが不完全なものになる

SSSD と Active Directory の統合時に、グループサイズが 1500 メンバーを超えると、SSSD が取得するメンバーリストが不完全なものになります。この問題は、1 回のクエリーで取得できるメンバーの数を制限する Active Directory の MaxValRange ポリシーが、デフォルトで 1500 に設定されているために発生します。

この問題を回避するには、Active Directory の MaxValRange 設定を変更して、より大きなグループサイズに対応できるようにします。

Jira:RHELDOCS-19603

9.13. デスクトップ

ソフトウェアリポジトリーからの flatpak リポジトリー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** リポジトリーを無効化または削除することはできません。

Bugzilla:1668760

Generation 2 の RHEL 8 仮想マシンが Hyper-V Server 2016 ホストで起動できない場合があります。

Microsoft Hyper-V Server 2016 ホストで実行している仮想マシンで RHEL 8 をゲストオペレーティングシステムとして使用すると、仮想マシンが起動しなくなり、GRUB ブートメニューに戻る場合があります。さらに、以下のエラーが Hyper-V イベントログに記録されます。

> The guest operating system reported that it failed with the following error code: 0x1E

このエラーは、Hyper-V ホストの UEFI ファームウェアバグが原因で発生します。この問題を回避するには、Hyper-V Server2019 以降をホストとして使用します。

Bugzilla:1583445[1]

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。

Bugzilla:1717947

WebKitGTK が IBM Z で Web ページの表示に失敗する

WebKitGTK Web ブラウザーエンジンは、IBM Z アーキテクチャーで Web ページを表示しようとすると失敗します。Web ページは空白のままで、WebKitGTK プロセスが予期せず停止します。

その結果、WebKitGTK を使用して Web ページを表示するアプリケーションの次のような特定の機能を使用できなくなります。

- Evolution メールクライアント
- GNOME Online Account 設定
- GNOME ヘルプアプリケーション

Jira:RHEL-4158

9.14. グラフィックインフラストラクチャー

Radeon ドライバーがハードウェアを正しくリセットできない

現在、**radeon** カーネルドライバーは、**kexec** コンテキストでハードウェアを正しくリセットしません。代わりに **radeon** がフェイルオーバーします。これにより、**kdump** サービスの残りの部分が失敗します。

この問題を回避するには、**/etc/kdump.conf** ファイルに以下の行を追加して、**kdump** で **radeon** を無効にします。

```
dracut_args --omit-drivers "radeon"
force_rebuild 1
```

システムと **kdump** を再起動します。**kdump** の起動後、設定ファイルから **force_rebuild 1** 行が削除される場合があります。

このシナリオでは、ダンププロセス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。

Bugzilla:1694705[1]

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、電源が入らない可能性がある

nouveau ドライバーを搭載した NVIDIA Turing GPU を使用するシステムで、HDR をサポートする複数のモニターが接続された **DisplayPort** ハブ (ラップトップドックなど) を使用すると、電源が入らない可能性があります。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。

Bugzilla:1812577[1]

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状態で GUI を起動すると、GUI が予期せず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 might experience a crash on ESXi 7.0.1 (83194) も参照してください。

Bugzilla:1910358[1]

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。

Bugzilla: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と、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。

Bugzilla: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[1]

9.15. RED HAT ENTERPRISE LINUX システムロール

Ansible 2.9 で RHEL システムロールを使用すると、command モジュールで dnf を使用することに関する警告が表示されることがあります。

RHEL 8.8 以降、RHEL システムロールは **dnf** モジュールで **warn** パラメーターを使用しなくなりました。これは、このパラメーターが Ansible Core 2.14 で削除されたためです。ただし、Ansible 2.9 で最新の **rhel-system-roles** パッケージを使用し、ロールがパッケージをインストールすると、次のいずれかの警告が表示される場合があります。

> [WARNING]: Consider using the dnf module rather than running 'dnf'. If you need to use command because dnf is insufficient you can add 'warn: false' to
> this command task or set 'command_warnings=False' in ansible.cfg to get rid of this message.

> [WARNING]: Consider using the yum, dnf or zypper module rather than running 'rpm'. If you need to use command because yum, dnf or zypper is insufficient
> you can add 'warn: false' to this command task or set 'command_warnings=False' in ansible.cfg to get rid of this message.

これらの警告を非表示にする場合は、**ansible.cfg** ファイルの **[Defaults]** セクションに **command_warnings = False** 設定を追加します。ただし、この設定により Ansible のすべての警告が無効になることに注意してください。

Jira:RHELDOCS-17954

Playbook またはインベントリーでホスト名 localhost を使用して localhost を管理できません

RHEL に **ansible-core 2.13** パッケージが含まれているため、ノードを管理しているのと同じホストで Ansible を実行している場合は、Playbook またはインベントリーで **localhost** ホスト名を使用して実行することはできません。これは、**ansible-core 2.13** が **python38** モジュールを使用し、ライブラリーの多くが欠落しているために発生します。たとえば、**storage** ロールの場合は **blivet**、**network** ロールの場合は **gobject** です。この問題を回避するには、Playbook またはインベントリーでホスト名 **localhost** をすでに使用している場合は、**ansible_connection=local** を使用するか、**ansible_connection=local** オプションを使用して **localhost** をリストするインベントリーファイルを作成することで接続を追加できます。これにより、**localhost** 上のリソースを管理できます。詳細は、記事 RHEL system roles playbooks fail when run on localhost を参照してください。

Bugzilla:2041997

rhc_auth にアクティベーションキーが含まれている場合、rhc システムロールはすでに登録されているシステムで失敗します。

rhc_auth パラメーターにアクティベーションキーが指定されている場合、すでに登録されているシステムで Playbook ファイルを実行すると失敗します。この問題を回避するには、登録済みのシステムで Playbook ファイルを実行するときにアクティベーションキーを指定しないでください。

Bugzilla:2186908

imuxsock 入力基本タイプを設定すると問題が発生する

logging RHEL システムロールおよび **use_imuxsock** オプションを介して "imuxsock" 入力基本タイプを設定すると、その結果となるマネージドノードの設定で問題が発生します。ロールは **name** パラメーターを設定しますが、"imuxsock" 入力タイプは **name** パラメーターをサポートしません。その結果、**rsyslog** ロギングユーティリティーは、**parameter 'name' not known – typo in config file?** エラーを出力します。

Jira:RHELDOCS-18326

RHEL 9 の UEFI 管理対象ノードで、 bootloader RHEL システムロールの bootloader_password 変数が機能しない

以前は、**bootloader_password** 変数によって、パスワード情報が **/boot/efi/EFI/redhat/user.cfg** ファイルに誤って配置されていました。適切な場所は **/boot/grub2/user.cfg** ファイルでした。その結果、管理対象ノードを再起動してブートローダーエントリーを変更したときに、GRUB2 がパスワードの入力を要求しませんでした。この問題を回避するには、**user.cfg** ファイルを誤った **/boot/efi/EFI/redhat/** ディレクトリーから正しい **/boot/grub2/** ディレクトリーに手動で移動し、想定される動作を実現します。

Jira:RHEL-45711

9.16. 仮想化

多数のキューを使用すると、Windows 仮想マシンで障害が発生することがある

仮想 Trusted Platform Module (vTPM) デバイスが有効で、**マルチキュー virtio-net** 機能が 250 を超えるキューを使用するように設定されている場合、Windows 仮想マシン (VM) が失敗することがあります。

この問題は、vTPM デバイスの制限が原因で発生します。vTPM デバイスには、開いているファイル記述子の最大数に関するハードコーディングされた制限があります。新しいキューごとに複数のファイル記述子が開かれるため、内部の vTPM 制限を超えて VM が失敗する可能性があります。

この問題を回避するには、次の 2 つのオプションのいずれかを選択します。

- vTPM デバイスを有効のままにしますが、使用するキューは 250 未満にします。
- 250 を超えるキューを使用するには、vTPM デバイスを無効にします。

Jira:RHEL-13336[1]

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。

Bugzilla:2077770[1]

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。

Bugzilla:1740002

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。

Bugzilla:1777138[1]

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

仮想マシン (VM) に多数の virtio-blk デバイスを追加すると、プラットフォームで使用可能な割り込みベクトルの数が使い果たさ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。

Bugzilla:1719687

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。

Bugzilla:1910848

ibmvfc ドライバーの使用時に IBM POWER ホストが正しく動作するようになりました。

PowerVM 論理パーティション (LPAR) で RHEL 8 を実行すると、**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ことがありました。その結果、次のような特定の状況下で、ホスト上でカーネルパニックが発生していました。

- 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

この更新により、**ibmvfc** の処理が修正され、前述のカーネルパニックは発生しなくなります。

Bugzilla:1961722[1]

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アントで RHEL 8 ホストを使用する場合は、**perf kvm record** コマンドを使用して KVM 仮想マシンのイベントサンプルを収集すると、仮想マシンが応答しなくなることがあります。この状況は、以下の場合に発生します。

- **perf** ユーティリティーは権限のないユーザーによって使用され、**-p** オプションは仮想マシンを識別するために使用されます (**perf kvm record -e trace_cycles -p 12345**)。
- 仮想マシンが **virsh** シェルを使用して起動している。

この問題を回避するには、**perf kvm** ユーティリティーに **-i** オプションを指定して、**virsh** シェルを使用して作成した仮想マシンを監視します。以下に例を示します。

```
# perf kvm record -e trace_imc/trace_cycles/  -p <guest pid> -i
```

-i オプションを使用する場合、子タスクはカウンターを継承しないため、スレッドは監視されないことに注意してください。

Bugzilla:1924016[1]

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。

Bugzilla:1942888[1]

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。

Bugzilla:1741436[1]

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、**virt-customize** コマンドに **--selinux-relabel** オプションを指定して使用します。

Bugzilla: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。

Bugzilla: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なります。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。

Bugzilla:1974622[1]

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、**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。

Bugzilla: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。

Bugzilla:2035158

SUID と SGID が virtiofs で自動的にクリアされない

killpriv_v2 機能を使用して **virtiofsd** サービスを実行すると、一部のファイルシステム操作を実行した後、システムによって SUID および SGID 権限が自動的にクリアされない場合があります。したがって、アクセス許可をクリアしないと、潜在的なセキュリティー上の脅威が発生する可能性があります。この問題を回避するには、次のコマンドを入力して **killpriv_v2** 機能を無効にします。

```
# virtiofsd -o no_killpriv_v2
```

Bugzilla:1966475[1]

ホストで OVS サービスを再起動すると、実行中の VM でネットワーク接続がブロックされることがある

ホストで Open vSwitch (OVS) サービスが再起動またはクラッシュすると、このホストで実行されている仮想マシン (VM) はネットワークデバイスの状態を回復できません。その結果、仮想マシンがパケットを完全に受信できなくなる可能性があります。

この問題は、**virtio** ネットワークスタックで圧縮された virtqueue 形式を使用するシステムのみに影響します。

この問題を回避するには、**virtio** ネットワークデバイス定義で **packed=off** パラメーターを使用して、圧縮された virtqueue を無効にします。圧縮された virtqueue を無効にすると、状況によっては、ネットワークデバイスの状態を RAM から回復できます。

Bugzilla:1792683

nodedev-dumpxml が特定の仲介デバイスの属性を正しく一覧表示しない

現在、**nodedev-dumpxml** は、**nodedev-create** コマンドを使用して作成された仲介デバイスの属性を正しく一覧表示していません。この問題を回避するには、代わりに **nodedev-define** コマンドおよび **nodedev-start** コマンドを使用します。

Bugzilla:2143160

NVIDIA A16 GPU を使用して仮想マシンを起動すると、ホスト GPU が動作を停止する場合がある

現在、NVIDIA A16 GPU パススルーデバイスを使用する仮想マシンを起動すると、ホストシステム上の NVIDIA A16 GPU 物理デバイスが動作を停止する場合があります。

この問題を回避するには、ハイパーバイザーを再起動し、GPU デバイスの **reset_method** を **bus** に設定します。

```
# echo bus > /sys/bus/pci/devices/<DEVICE-PCI-ADDRESS>/reset_method
# cat /sys/bus/pci/devices/<DEVICE-PCI-ADDRESS>/reset_method
bus
```

詳細は、Red Hat ナレッジベースの記事 を参照してください。

Jira:RHEL-2451[1]

CPU あたり 128 を超えるコアを使用すると、起動時に Windows Server 2019 仮想マシンがクラッシュする

現在、Windows Server 2019 ゲストオペレーティングシステムを使用する仮想マシン (VM) は、単一の仮想 CPU (vCPU) に 128 を超えるコアを使用するように設定されている場合、起動に失敗します。仮想マシンは、起動する代わりに青い画面で停止エラーを表示します。この問題を回避するには、vCPU あたり 128 コア未満を使用します。

Jira:RHELDOCS-18863[1]

9.17. クラウド環境の RHEL

VMware ホストの RHEL 仮想マシンで静的 IP を設定できない

現在、VMware ホストで RHEL を仮想マシンのゲストオペレーティングシステムとして使用すると、DatasourceOVF 機能は正しく機能しません。これにより、**cloud-init** ユーティリティーを使用して、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。

この問題を回避するには、VMware のナレッジベース を参照してください。

Jira:RHEL-12122

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
```

Bugzilla:1865745[1]

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することがある

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。したがって、**%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な場合があります。

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 -I 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
```

Bugzilla:1906870[1]

cloud-init によってプロビジョニングされ、NFSv3 マウントエントリーで設定された場合、Azure で RHEL インスタンスが起動しない

現在、仮想マシンが **cloud-init** ツールによってプロビジョニングされ、仮想マシンのゲストオペレーティングシステムで **/etc/fstab** ファイルに NFSv3 マウントエントリーがある場合、Microsoft Azure クラウドプラットフォームで RHEL 仮想マシンの起動に失敗します。

Bugzilla:2081114[1]

9.18. サポート性

getattachment コマンドが複数の添付ファイルを一度にダウンロードできない

redhat-support-tool コマンドは、添付ファイルをダウンロードするための **getattachment** サブコマンドを提供します。ただし、**getattachment** は現在、1 つの添付ファイルしかダウンロードできず、複数の添付ファイルをダウンロードできません。

回避策として、**getattachment** サブコマンドで各添付ファイルのケース番号と UUID を渡すことにより、複数の添付ファイルを 1 つずつダウンロードできます。

Bugzilla: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。

Jira:RHEL-2345

IBM Power Systems (リトルエンディアン) 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 (リトルエンディアン) 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ー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
```

Bugzilla:2011413[1]

9.19. コンテナー

古いコンテナーイメージ内で systemd を実行すると動作しない

古いコンテナーイメージ (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[1]

第10章 国際化

10.1. 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

10.2. 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。
- いくつかの **glibc** ロケールが Unicode Common Locale Data Repository (CLDR) と同期されました。

付録A コンポーネント別のチケットリスト

参考のために、Bugzilla および JIRA チケットのリストをこのドキュメントに記載します。リンクをクリックすると、チケットを説明したこのドキュメントのリリースノートにアクセスできます。

コンポーネント	チケット
389-ds-base	Jira:RHEL-19028、Jira:RHEL-19240、Jira:RHEL-5390、Jira:RHEL-5143、Jira:RHEL-5107、Jira:RHEL-16338、Jira:RHEL-14025、Jira:RHEL-5135
リリースノート	Jira:RHELDOCS-17954, Jira:RHELDOCS-18326, Jira:RHELDOCS-16861, Jira:RHELDOCS-16755, Jira:RHELDOCS-16612, Jira:RHELDOCS-17102, Jira:RHELDOCS-17518
SLOF	Bugzilla:1910848
accel-config	Bugzilla:1843266
anaconda	Jira:RHEL-13151、Bugzilla:2050140、Jira:RHEL-4707、Jira:RHEL-4711、Jira:RHEL-4744
ansible-collection-microsoft-sql	Jira:RHEL-19204、Jira:RHEL-19202、Jira:RHEL-19203
ansible-freeipa	Jira:RHEL-16938、Jira:RHEL-16933、Jira:RHEL-19133、Jira:RHEL-4963、Jira:RHEL-19129
ant	Jira:RHEL-5365
apr	Bugzilla:1819607
audit	Jira:RHEL-15001
authselect	Bugzilla:1892761
bacula	Jira:RHEL-6859
brltty	Bugzilla:2008197
chrony	Jira:RHEL-21069
clang	Jira:RHEL-9299
cloud-init	Jira:RHEL-7312、Jira:RHEL-7278、Jira:RHEL-12122
cmake	Jira:RHEL-7396

コンポーネント	チケット
cockpit	Bugzilla:1666722
coreutils	Bugzilla:2030661
corosync-qdevice	Bugzilla:1784200
crash	Jira:RHEL-9010、Bugzilla:1906482
crash-ptdump-command	Bugzilla:1838927
createrepo_c	Bugzilla:1973588
crypto-policies	Jira:RHEL-2345、Bugzilla:1919155、Bugzilla:1660839
device-mapper-multipath	Jira:RHEL-6677、Jira:RHEL-16563、Bugzilla:2022359、Bugzilla:2011699
distribution	Jira:RHEL-17090、Bugzilla:1657927
dnf	Bugzilla:1986657
dnf-plugins-core	Jira:RHEL-17356、Jira:RHELPLAN-50409
edk2	Bugzilla:1741615、Bugzilla:1935497
elfutils	Jira:RHEL-15924
fapolicyd	Jira:RHEL-520、Bugzilla:2054741
fence-agents	Bugzilla:1775847
firewalld	Bugzilla:1871860
gcc-toolset-13-binutils	Jira:RHEL-25405
gdb	Bugzilla:1853140
git	Jira:RHEL-17103
git-lfs	Jira:RHEL-17102
glibc	Jira:RHEL-13720、Jira:RHEL-10481、Jira:RHEL-19824

コンポーネント	チケット
gnome-shell-extensions	Bugzilla:1717947
gnome-software	Bugzilla:1668760
gnutls	Bugzilla:1628553
golang	Jira:RHEL-11872
grafana	Jira:RHEL-7503
grub2	Jira:RHEL-15856、Jira:RHEL-15583、Bugzilla:1583445
initscripts	Bugzilla:1875485
ipa	Jira:RHEL-16936、Jira:RHEL-12153、Jira:RHEL-4964、Jira:RHEL-10495、Jira:RHEL-4847、Jira:RHEL-4898、Bugzilla:1664719、Bugzilla:1664718
ipmitool	Jira:RHEL-6846
kernel	Bugzilla:2041881、Jira:RHEL-11597、Bugzilla:1868526、Bugzilla:1694705、Bugzilla:1730502、Bugzilla:1609288、Bugzilla:1602962、Bugzilla:1865745、Bugzilla:1906870、Bugzilla:1924016、Bugzilla:1942888、Bugzilla:1812577、Bugzilla:1910358、Bugzilla:1930576、Bugzilla:1793389、Bugzilla:1654962、Bugzilla:1940674、Bugzilla:1920086、Bugzilla:1971506、Bugzilla:2059262、Bugzilla:2050411、Bugzilla:2106341、Bugzilla:1605216、Bugzilla:1519039、Bugzilla:1627455、Bugzilla:1501618、Bugzilla:1633143、Bugzilla:1814836、Bugzilla:1839311、Bugzilla:1696451、Bugzilla:1348508、Bugzilla:1836977、Bugzilla:1878207、Bugzilla:1665295、Bugzilla:1871863、Bugzilla:1569610、Bugzilla:1794513
kernel / DMA Engine	Jira:RHEL-10097
kernel / Networking / NIC Drivers	Jira:RHEL-11398
kernel / Networking / Protocol / tcp	Jira:RHEL-6113
kernel / Storage / Device Mapper / Crypt	Jira:RHEL-22232
kernel / Storage / Storage Drivers	Jira:RHEL-1765

コンポーネント	チケット
kernel / Virtualization / KVM	Jira:RHEL-2451
kernel-rt / Other	Jira:RHEL-9318
kexec-tools	Bugzilla:2111855
kmod	Bugzilla:2103605
krb5	Jira:RHEL-4910、Bugzilla:2125318、Bugzilla:1877991
libdnf	Jira:RHEL-6421
libgnome-keyring	Bugzilla:1607766
libguestfs	Bugzilla:1554735
libkcapi	Jira:RHEL-5366、Jira:RHEL-15300
librdkafka	Jira:RHEL-12892
librepo	Jira:RHEL-10720
libreswan	Bugzilla:1989050
libselinux-python-2.8-module	Bugzilla:1666328
libvirt	Bugzilla:1664592、Bugzilla:1332758、Bugzilla:2143160、Bugzilla:1528684
linuxptp	Jira:RHEL-21326
llvm-toolset	Jira:RHEL-9028
lvm2	Bugzilla:1496229、Bugzilla:1768536
mariadb	Jira:RHEL-3637、Bugzilla:1942330
maven	Jira:RHEL-17126
mesa	Bugzilla:1886147
nfs-utils	Bugzilla:2081114、Bugzilla:1592011

コンポーネント	チケット
nginx	Jira:RHEL-14714
nispor	Bugzilla:2153166
nss	Bugzilla:1817533、Bugzilla:1645153
nss_nis	Bugzilla:1803161
opencryptoki	Jira:RHEL-11413
opencv	Bugzilla:1886310
openmpi	Bugzilla:1866402
opensc	Jira:RHEL-4077、Bugzilla:1947025
openscap	Bugzilla:2161499
openssh	Jira:RHEL-1684、Jira:RHEL-5279、Bugzilla:2044354
openssl	Jira:RHEL-17689、Bugzilla:1810911
osbuild-composer	Jira:RHEL-4649
oscap-anaconda-addon	Jira:RHEL-1826、Bugzilla:1843932、Bugzilla:1834716、Bugzilla:1665082、Jira:RHEL-1810
papi	Jira:RHEL-9336
pcs	Jira:RHEL-7584、Jira:RHEL-7668、Jira:RHEL-7731、Jira:RHEL-7745、Bugzilla:1619620、Bugzilla:1851335
perl-DateTime-TimeZone	Jira:RHEL-35685
php	Jira:RHEL-14705
pki-core	Bugzilla:1729215、Jira:RHEL-13125、Bugzilla:1628987
podman	Jira:RHELPLAN-167794、Jira:RHELPLAN-167830、Jira:RHELPLAN-167822、Jira:RHELPLAN-168179、Jira:RHELPLAN-168184、Jira:RHELPLAN-154435、Jira:RHELPLAN-168223
policycoreutils	Jira:RHEL-24461

コンポーネント	チケット
polkit	Jira:RHEL-34022
postfix	Bugzilla:1711885
postgresql	Jira:RHEL-3636
pykickstart	Bugzilla:1637872
python3.11-lxml	Bugzilla:2157673
python36-3.6-module	Bugzilla:2165702
qemu-kvm	Jira:RHEL-11597、Jira:RHEL-16696、Jira:RHEL-13336、Bugzilla:1740002、Bugzilla:1719687、Bugzilla:1966475、Bugzilla:1792683、Bugzilla:1651994
rear	Jira:RHEL-24729、Jira:RHEL-17354、Jira:RHEL-17353、Bugzilla:1925531、Bugzilla:2083301
redhat-support-tool	Bugzilla:2064575
restore	Bugzilla:1997366
rhel-system-roles	Jira:RHEL-18170、Jira:RHEL-3241、Jira:RHEL-15440、Jira:RHEL-4624、Jira:RHEL-16542、Jira:RHEL-21491、Jira:RHEL-16965、Jira:RHEL-16553、Jira:RHEL-18963、Jira:RHEL-16975、Jira:RHEL-21123、Jira:RHEL-19047、Jira:RHEL-15038、Jira:RHEL-21134、Jira:RHEL-14022、Jira:RHEL-17667、Jira:RHEL-15933、Jira:RHEL-16213、Jira:RHEL-16977、Jira:RHEL-5985、Jira:RHEL-4684、Jira:RHEL-16501、Jira:RHEL-17874、Jira:RHEL-21946、Jira:RHEL-21400、Jira:RHEL-15871、Jira:RHEL-22228、Jira:RHEL-22229、Jira:RHEL-3354、Jira:RHEL-19042、Jira:RHEL-19044、Jira:RHEL-19242、Jira:RHEL-21402、Jira:RHEL-25509、Bugzilla:2186908、Bugzilla:2021685、Bugzilla:2006081
rpm	Bugzilla:1688849
rsyslog	Bugzilla:1679512、Jira:RHELPLAN-10431
rteval	Jira:RHEL-8967、Jira:RHEL-21926
rtla	Jira:RHEL-10081
rust-toolset	Jira:RHEL-12964

コンポーネント	チケット
samba	Jira:RHEL-16483、Bugzilla:2009213、Jira:RHELPLAN-13195
scap-security-guide	Jira:RHEL-25250、Bugzilla:2028428、Bugzilla:2118758、Jira:RHEL-1804、Jira:RHEL-1897
selinux-policy	Jira:RHEL-9981、Jira:RHEL-1388、Jira:RHEL-15398、Jira:RHEL-1628、Jira:RHEL-10087、Bugzilla:2166153、Bugzilla:1461914
sos	Bugzilla:2011413
spice	Bugzilla:1849563
sssd	Jira:SSSD-7015、Bugzilla:2065692、Bugzilla:2056483、Bugzilla:1947671
sssd_kcm	Jira:SSSD-7015
stunnel	Jira:RHEL-2340
subscription-manager	Bugzilla:2170082
sysstat	Jira:RHEL-12008、Jira:RHEL-23074
tuna	Jira:RHEL-19179
tuned	Bugzilla:2113900
udica	Bugzilla:1763210
valgrind	Jira:RHEL-15926
vdo	Bugzilla:1949163
virt-manager	Bugzilla:2026985
wayland	Bugzilla:1673073
webkit2gtk3	Jira:RHEL-4158
xorg-x11-server	Bugzilla:1698565

コンポーネント	チケット
その他	Jira:RHELDOCS-17369、Jira:RHELDOCS-17372、Jira:RHELDOCS-16955、Jira:RHELDOCS-16241、Jira:RHELDOCS-16970、Jira:RHELDOCS-17060、Jira:RHELDOCS-17056、Jira:RHELDOCS-16337、Jira:RHELDOCS-17261、Jira:RHELPLAN-123140、Jira:RHELDOCS-18289、Jira:RHELDOCS-18323、Jira:SSSD-6184、Bugzilla:2025814、Bugzilla:2077770、Bugzilla:1777138、Bugzilla:1640697、Bugzilla:1697896、Bugzilla:1961722、Jira:RHELDOCS-18064、Jira:RHELDOCS-18049、Bugzilla:1659609、Bugzilla:1687900、Bugzilla:1757877、Bugzilla:1741436、Jira:RHELPLAN-27987、Jira:RHELPLAN-34199、Jira:RHELPLAN-57914、Jira:RHELPLAN-96940、Bugzilla:1974622、Bugzilla:2028361、Bugzilla:2041997、Bugzilla:2035158、Jira:RHELPLAN-109613、Bugzilla:2126777、Jira:RHELDOCS-17126、Bugzilla:1690207、Bugzilla:1559616、Bugzilla:1889737、Bugzilla:1906489、Bugzilla:1769727、Jira:RHELPLAN-27394、Jira:RHELPLAN-27737、Jira:RHELDOCS-16861、Bugzilla:1642765、Bugzilla:1646541、Bugzilla:1647725、Bugzilla:1932222、Bugzilla:1686057、Bugzilla:1748980、Jira:RHELPLAN-71200、Jira:RHELPLAN-45858、Bugzilla:1871025、Bugzilla:1871953、Bugzilla:1874892、Bugzilla:1916296、Jira:RHELDOCS-17573、Jira:RHELPLAN-100400、Bugzilla:1926114、Bugzilla:1904251、Bugzilla:2011208、Jira:RHELPLAN-59825、Bugzilla:1920624、Jira:RHELPLAN-70700、Bugzilla:1929173、Jira:RHELPLAN-85066、Jira:RHELPLAN-98983、Bugzilla:2009113、Bugzilla:1958250、Bugzilla:2038929、Bugzilla:2006665、Bugzilla:2029338、Bugzilla:2061288、Bugzilla:2060759、Bugzilla:2055826、Bugzilla:2059626、Jira:RHELPLAN-133171、Bugzilla:2142499、Jira:RHELDOCS-16755、Jira:RHELPLAN-146398、Jira:RHELDOCS-18107、Jira:RHELPLAN-153267、Bugzilla:2225332、Jira:RHELPLAN-147538、Jira:RHELDOCS-16612、Jira:RHELDOCS-17102、Jira:RHELDOCS-16300、Jira:RHELDOCS-17038、Jira:RHELDOCS-17461、Jira:RHELDOCS-17518、Jira:RHELDOCS-17623

付録B 改訂履歴

0.2-5

2025 年 4 月 22 日火曜日、Marc Muehlfeld (mmuuehlfeld@redhat.com)

- 拡張機能 RHEL-83447 (動的プログラミング言語、Web およびデータベースサーバー) を追加しました

0.2-4

2025 年 3 月 25 日火曜日、Gabriela Fialová (gfialova@redhat.com)

- RHELDOCS-19863 (Identity Management) に機能拡張を追加しました。
- RHELDOCS-19805 (Identity Management) のバグ修正を追加しました。

0.2-3

2025 年 2 月 24 日月曜日、Gabriela Fialová (gfialova@redhat.com)

- RHEL-14942 (動的プログラミング言語) の機能強化を更新しました。

0.2-2

2024 年 1 月 30 日木曜日、Gabriela Fialová (gfialova@redhat.com)

- 既知の問題 RHELDOCS-19603 (IdM SSSD) を追加しました。

0.2-1

2025 年 1 月 28 日火曜日、Marc Muehlfeld (mmuuehlfeld@redhat.com)

- 機能拡張 RHEL-35991 (動的プログラミング言語、Web およびデータベースサーバー) を追加しました。

0.2-0

2025 年 1 月 22 日 (水)、Gabriela Fialová (gfialova@redhat.com)

- 既知の問題 RHELDOCS-18863 (仮想化) を追加しました。

0.1-10

2025 年 1 月 13 日月曜日、Marc Muehlfeld (mmuehlfeld@redhat.com)

- バグ修正 RHEL-73052 (ネットワーク) を追加しました。

0.1-9

2024 年 12 月 4 日水曜日、Gabriela Fialová (gfialova@redhat.com)

- カスタマーポータルラボのセクションを更新しました。
- インストールセクションを更新しました。

0.1-8

2024 年 11 月 5 日火曜日、Lenka Špačková (lspackova@redhat.com)

- コンパイラーおよび開発ツール に複数の新機能を追加しました。具体的には、新しい GCC Toolset 14、GCC Toolset 13 GCC の更新、LLVM、Rust、Go Toolset のリベースです。

0.1-7

2024 年 10 月 23 日木曜日、Gabriela Fialová (gfialova@redhat.com)

- 既知の問題 RHELDOCS-18777 (Identity Management) を追加しました。

0.1-6

2024 年 10 月 17 日木曜日、Brian Angelica (bangelic@redhat.com)

- 非推奨の機能 RHELDOCS-19027 (ファイルシステムとストレージ) を追加しました。

0.1-5

2024 年 10 月 9 日水曜日、Brian Angelica (bangelic@redhat.com)

- 非推奨の機能 RHEL-18958 (コンテナー) を追加しました。

0.1-4

2024 年 10 月 9 日水曜日、Brian Angelica (bangelic@redhat.com)

- バグ修正 RHEL-45908 (Identity Management) を追加しました。

0.1-3

2024 年 9 月 24 日火曜日、Lenka Špačková (lspackova@redhat.com)

- 機能拡張 RHEL-49614 (動的プログラミング言語、Web およびデータベースサーバー) を追加しました。

0.1-2

2024 年 8 月 27 日 (火)、Lenka Špačková (lspackova@redhat.com)

- バグ修正 RHEL-39994 (コンパイラーと開発ツール) を追加しました。

0.1-1

2024 年 8 月 14 日水曜日、Brian Angelica (bangelic@redhat.com)

- 既知の問題 RHELDOCS-18748 (ネットワーク) を追加しました。

0.1-0

2024 年 8 月 14 日水曜日、Brian Angelica (bangelic@redhat.com)

- 機能拡張 RHEL-47595 (ネットワーク) を追加しました。

0.0-9

2024 年 8 月 9 日金曜日、Brian Angelica (bangelic@redhat.com)

- 既知の問題 RHEL-11397 (インストーラーとイメージの作成) を追加しました。

0.0-8

2024 年 7 月 18 日木曜日、Gabriela Fialová (gfialova@redhat.com)

- 非推奨の機能 Jira:RHELDOCS-17573 (Identity Management) を更新しました。

0.0-7

2024 年 7 月 11 日木曜日、Lenka Špačková (lspackova@redhat.com)

- 既知の問題 RHEL-45711 (システムロール) を追加

0.0-6

2024 年 7 月 8 日月曜日、Lenka Špačková (lspackova@redhat.com)

- RHEL-25405 (コンパイラーおよび開発ツール) のフォーマットと参照を修正しました。

0.0-5

2024 年 7 月 3 日水曜日、Lenka Špačková (lspackova@redhat.com)

- 既知の問題 RHEL-34075 (Identity Management) を追加しました。

0.0-4

2024 年 6 月 25 日火曜日、Lenka Špačková (lspackova@redhat.com)

- 既知の問題 RHELDOCS-18435 (動的プログラミング言語、Web およびデータベースサーバー) を追加しました。

0.0-3

2024 年 6 月 12 日水曜日、Brian Angelica (bangelic@redhat.com)

- Jira:RHELPLAN-123140 (Identity Management) の機能拡張を更新しました。

0.0-2

2024 年 6 月 7 日 (金)、Brian Angelica (bangelic@redhat.com)

- Jira:RHELDOCS-18326 の既知の問題を更新しました (Red Hat Enterprise Linux システムロール)。

0.0-1

2024 年 5 月 23 日 (木) Brian Angelica (bangelic@redhat.com)

- Red Hat Enterprise Linux 8.10 リリースノートのリリース。

0.0-0

2024 年 3 月 27 日、水曜日、Lucie Vařáková (lvarakova@redhat.com)

- Red Hat Enterprise Linux 8.10 ベータ版リリースノートのリリース。