OpenShift Container Platform 4.12

リリースノート

新機能のハイライトおよび OpenShift Container Platform リリースの変更内容

Last Updated: 2023-03-31

OpenShift Container Platform 4.12 リリースノート

新機能のハイライトおよび OpenShift Container Platform リリースの変更内容

法律上の通知

Copyright © 2023 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at
http://creativecommons.org/licenses/by-sa/3.0/
. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

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

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

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

Node.js ® is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

以下の OpenShift Container Platform リリースノートでは、新機能および拡張機能のすべて、以前のバージョンからの主な技術上の変更点、主な修正、および一般公開バージョンの既知の問題についてまとめています。

目次

第1章 OPENSHIFT CONTAINER PLATFORM 4.12 リリースノート

Red Hat OpenShift Container Platform では、設定や管理のオーバーヘッドを最小限に抑えながら、セキュアでスケーラブルなリソースに新規および既存のアプリケーションをデプロイするハイブリッドクラウドアプリケーションプラットフォームを開発者や IT 組織に提供します。OpenShift Container Platform は、Java、Javascript、Python、Ruby および PHP など、幅広いプログラミング言語およびフレームワークをサポートしています。

Red Hat Enterprise Linux (RHEL) および Kubernetes にビルドされる OpenShift Container Platform は、最新のエンタープライズレベルのアプリケーションに対してよりセキュアでスケーラブルなマルチテナント対応のオペレーティングシステムを提供するだけでなく、統合アプリケーションランタイムやライブラリーを提供します。OpenShift Container Platform を使用することで、組織はセキュリティー、プライバシー、コンプライアンス、ガバナンスの各種の要件を満たすことができます。

1.1. 本リリースについて

OpenShift Container Platform (RHSA-2022:7399) をご利用いただけるようになりました。このリリースでは、CRI-O ランタイムで Kubernetes 1.25 を使用します。以下では、OpenShift Container Platform 4.12 に関連する新機能、変更点および既知の問題について説明します。

OpenShift Container Platform 4.12 クラスターは https://console.redhat.com/openshift で入手できます。OpenShift Container Platform 向けの Red Hat OpenShift Cluster Manager アプリケーションを使って、OpenShift クラスターをオンプレミスまたはクラウド環境のいずれかにデプロイすることができます。

OpenShift Container Platform 4.12 は、Red Hat Enterprise Linux (RHEL) 8.6 および Red Hat Enterprise Linux CoreOS (RHCOS) 4.12 でサポートされています。

コントロールプレーンには RHCOS マシンを使用する必要があり、コンピュートマシンに RHCOS または RHEL のいずれかを使用できます。

OpenShift Container Platform 4.12 以降、偶数番号のリリースで6 カ月の延長更新サポート (EUS) フェーズが追加され、18 カ月から 2 年になります。詳細は、Red Hat OpenShift Container Platform ライフサイクルポリシー を参照してください。

OpenShift Container Platform 4.8 は Extended Update Support (EUS) リリースです。Red Hat OpenShift EUS の詳細は、OpenShift ライフサイクル、および OpenShift EUS の概要 を参照してください。

バージョン 4.8 のメンテナンスサポートは 2023 年 1 月に終了し、延長ライフフェーズに移行します。詳細は、Red Hat OpenShift Container Platform ライフサイクルポリシー を参照してください。

1.2. OPENSHIFT CONTAINERPLATFORM のレイヤー化された依存関係にあるコンポーネントのサポートと互換性

OpenShift Container Platform のレイヤー化された依存関係にあるコンポーネントのサポート範囲は、OpenShift Container Platform のバージョンに関係なく変更されます。アドオンの現在のサポートステータスと互換性を確認するには、リリースノートを参照してください。詳細は、Red Hat OpenShift Container Platform ライフサイクルポリシー を参照してください。

1.3. 新機能および機能拡張

今回のリリースでは、以下のコンポーネントおよび概念に関連する拡張機能が追加されました。

1.3.1. Red Hat Enterprise Linux CoreOS (RHCOS)

1.3.1.1. 新しいクラスターのデフォルトのコンソールは、インストールプラットフォームによって決定されるようになりました

OpenShift Container Platform 4.12 ブートイメージからインストールされた Red Hat Enterprise Linux CoreOS (RHCOS) ノードは、プラットフォーム固有のデフォルトコンソールを使用するようになりました。クラウドプラットフォームの既定のコンソールは、そのクラウドプロバイダーが期待する特定のシステムコンソールに対応しています。VMware および OpenStack イメージは、プライマリーグラフィカルコンソールとセカンダリーシリアルコンソールを使用するようになりました。他のベアメタルインストールでは、デフォルトでグラフィカルコンソールのみが使用され、シリアルコンソールは有効になりません。**coreos-installer** を使用して実行されるインストールは、既存のデフォルトをオーバーライドして、シリアルコンソールを有効にすることができます。

既存のノードは影響を受けません。既存のクラスター上の新しいノードは、通常、クラスターのインストールに最初に使用されたブートイメージからインストールされるため、影響を受ける可能性はほとんどありません。

シリアルコンソールを有効にする方法については、次のドキュメントを参照してください。

- デフォルトのコンソール設定。
- ライブインストール ISO イメージを変更して、シリアルコンソールを有効化
- ライブインストール PXE 環境を変更して、シリアルコンソールを有効化。

1.3.1.2. IBM zSystems および LinuxONE での IBM Secure Execution (テクノロジープレビュー)

OpenShift Container Platform は、テクノロジープレビュー機能として IBM zSystems および LinuxONE (s390x アーキテクチャー)での IBM Secure Execution 用の Red Hat Enterprise Linux CoreOS (RHCOS) ノードの設定をサポートするようになりました。IBM Secure Execution は、KVM ゲストのメモリー境界を保護するハードウェア拡張機能です。IBM Secure Execution は、クラスターのワークロードに最高レベルの分離とセキュリティーを提供します。これは、IBM Secure Execution 対応の QCOW2 ブートイメージを使用して有効にすることができます。

IBM Secure Execution を使用するには、ホストマシンのホストキーが必要であり、Ignition 設定ファイルで指定する必要があります。IBM Secure Execution は、LUKS 暗号化を使用してブートボリュームを自動的に暗号化します。

詳細は、IBM Secure Execution を使用した RHCOS のインストール を参照してください。

1.3.1.3. RHCOS が RHEL 8.6 を使用するようになる

RHCOS は、OpenShift Container Platform 4.12 で Red Hat Enterprise Linux (RHEL) 8.6 パッケージを使用するようになりました。これにより、最新の修正、機能、拡張機能、および最新のハードウェアサポートおよびドライバー更新を利用できます。OpenShift Container Platform 4.10 は延長更新サポート (EUS) リリースです。このリリースは、ライフサイクル全体に対して RHEL 8.4 EUS パッケージを引き続き使用します。

1.3.2. インストールおよびアップグレード

1.3.2.1. Assisted Installer SaaS は、Nutanix のプラットフォーム統合サポートを提供します。

console.redhat.com の Assisted Installer SaaS は、Assisted Installer ユーザーインターフェイスまたは

REST API を使用したマシン API 統合により、Nutanix プラットフォームへの OpenShift Container Platform のインストールをサポートします。統合により、Nutanix Prism ユーザーは 1 つのインターフェイスからインフラストラクチャーを管理でき、自動スケーリングが実現します。Nutanix と Assisted Installer SaaS の統合を有効にするには、追加のインストール手順がいくつかあります。詳細については、自動インストーラーのドキュメントを参照してください。

1.3.2.2. インストール時における AWS でのロードバランサーのタイプ指定

OpenShift Container Platform 4.12 以降、インストール時に AWS の永続的なロードバランサータイプとして Network Load Balancer (NLB) または Classic のいずれかを指定できます。その後、イングレスコントローラーが削除された場合、ロードバランサーのタイプは、インストール中に設定された lbType で保持されます。

詳細について は、ネットワークのカスタマイズを使用した AWS へのクラスターのインストール を参照してください。

1.3.2.3. Local Zone サブネットを使用して既存の Virtual Private Cloud (VPC) にインストールする場合にはワーカーノードを AWS のエッジに拡張する

今回の更新により、インストーラーによってプロビジョニングされたインフラストラクチャーを使用して OpenShift Container Platform を既存の VPC にインストールし、ワーカーノードを Local Zones サブネットに拡張できます。インストールプログラムは、NoSchedule taint を使用して、ユーザーアプリケーション用に指定された AWS ネットワークのエッジにワーカーノードをプロビジョニングします。エンドユーザーにおいて、Local Zones のロケーションにデプロイされたアプリケーションは低レイテンシーになります。

詳細は、AWS Local Zones を使用したクラスターのインストール を参照してください。

1.3.2.4. Google Cloud Platform Marketplace オファリング

OpenShift Container Platform が GCP Marketplace で利用できるようになりました。GCP Marketplace イメージを使用して OpenShift Container Platform をインストールすると、GCP を介して従量課金制 (時間単位、コア単位) で請求される自己管理型のクラスターデプロイを作成できますが、Red Hat によって直接サポートされます。

インストーラーでプロビジョニングされるインフラストラクチャーを使用したインストールについての詳細は、GCP Marketplace イメージの使用 を参照してください。ユーザーがプロビジョニングしたインフラストラクチャーを使用してインストールする方法は GCP での追加のワーカーマシンの作成 をご覧ください。

1.3.2.5. GCP および Azure へのインストール時におけるブートストラップ障害のトラブルシューティング

インストーラーは、GCP および Azure のブートストラップおよびコントロールプレーンホストからシリアルコンソールログを収集するようになりました。このログデータは標準のブートストラップログバンドルに追加されます。

詳細は、Troubleshooting installation issues を参照してください。

1.3.2.6. IBM Cloud VPC の一般公開 (GA)

IBM Cloud VPC は、OpenShift Container Platform 4.12 で一般提供されるようになりました。

クラスターのインストールについて詳しくは、IBM Cloud VPC へのインストールの準備 を参照してください。

1.3.2.7. OpenShift Container Platform 4.11 から 4.12 へのアップグレード時に、管理者の承認が必要

OpenShift Container Platform 4.12 は Kubernetes 1.25 を使用します。これにより、いくつかの非推奨 API が削除されました。

クラスター管理者は、クラスターを OpenShift Container Platform 4.11 から 4.12 にアップグレードする前に、手動で確認を行う必要があります。削除された API が、クラスター上で実行されている、またはクラスターと対話しているワークロード、ツール、またはその他のコンポーネントによって引き続き使用される OpenShift Container Platform 4.12 にアップグレードした後の問題を防ぐ上で役立ちます。管理者は、削除する予定の使用中の API のクラスターを評価し、影響を受けるコンポーネントを移行して適切な新規 API バージョンを使用する必要があります。これが完了すると、管理者による確認が可能です。

すべての OpenShift Container Platform 4.11 クラスターでは、OpenShift Container Platform 4.12 にアップグレードする前に、この管理者の承認が必要になります。

詳細は、Preparing to update to OpenShift Container Platform 4.12 を参照してください。

1.3.2.8. クラスターのインストール時の機能セットの有効化

OpenShift Container Platform 4.12 以降、インストールプロセスの一部として機能セットを有効にできます。機能セットは、デフォルトで有効にされない OpenShift Container Platform 機能のコレクションです。

インストール時に機能セットを有効にする方法についての詳細は、フィーチャーゲートを使用した OpenShift Container Platform の機能の有効化 を参照してください。

1.3.2.9. ARM 上の OpenShift Container Platform

OpenShift Container Platform 4.12 が、ARM アーキテクチャーベースの Azure インストーラーによってプロビジョニングされたインフラストラクチャーでサポートされるようになりました。AWS Graviton 3 プロセッサーがクラスターのデプロイで利用できるようになり、OpenShift Container Platform 4.11 でもサポートされます。インスタンスの可用性やインストールに関するドキュメントの詳細は、Supported installation methods for different platforms を参照してください。

1.3.2.10. oc-mirror CLI プラグイン (テクノロジープレビュー) を使用した OCI 形式のファイルベースのカタログ Operator イメージのミラーリング

oc-mirror CLI プラグインを使用してファイルベースのカタログ Operator イメージを Docker v2 形式ではなく OCI 形式でミラーリングする機能が、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。

詳細は、OCI フォーマットでのファイルベースのカタログ Operator イメージのミラーリング を参照してください。

1.3.2.11. GCP 上の OpenShift Container Platform クラスターを共有 VPC にインストール (テクノロジープレビュー)

OpenShift Container Platform 4.12 では、テクノロジープレビュー として GCP 上のクラスターを共有 VPC にインストールできます。このインストール方法では、クラスターは別の GCP プロジェクトの VPC を使用するように設定されています。共有 VPC により、組織は複数のプロジェクトから共通の VPC ネットワークにリソースを接続できるようになります。対象のネットワークの内部 IP アドレスを使用して、組織内の通信を安全かつ効率的に実行できます。

詳細は、GCP 上のクラスターを共有 VPC にインストール をご覧ください。

1.3.2.12. プロビジョニングネットワークのないベアメタルインストールでの Ironic API の一貫した IP アドレス

今回の更新により、プロビジョニングネットワークのないベアメタルインストールで、プロキシーサーバー経由で Ironic API サービスにアクセスできるようになりました。このプロキシーサーバーは、Ironic API サービスに一貫した IP アドレスを提供します。**metal3-ironic** を含む Metal3 Pod が別の Pod に再配置された場合、一貫したプロキシーアドレスにより、Ironic API サービスとの継続的な通信が保証されます。

1.3.2.13. サービスアカウント認証を使用して GCP に OpenShift Container Platform をインストールする

OpenShift Container Platform 4.12 では、サービスアカウントがアタッチされた仮想マシンを使用して GCP にクラスターをインストールできます。これにより、サービスアカウントの JSON ファイルを使用せずにインストールを実行できます。

詳細は 、GCP サービスアカウントの作成 をご覧ください。

1.3.2.14. OpenShift Container Platform クラスターによってプロビジョニングされた AWS リソースの PropagateUserTags パラメーター

OpenShift Container Platform 4.12 では、**propagateUserTags** パラメーターは、クラスター内の Operator が作成する AWS リソースのタグに指定されたユーザータグを含めるように指示するフラグです。

詳しくは、オプションの設定パラメーター を参照してください。

1.3.2.15. Ironic コンテナーイメージは RHEL 9 ベースイメージを使用します

以前のバージョンの OpenShift Container Platform では、Ironic コンテナーイメージは Red Hat Enterprise Linux (RHEL) 8 をベースイメージとして使用していました。OpenShift Container Platform 4.12 以降、Ironic コンテナーイメージは RHEL 9 を基本イメージとして使用します。RHEL 9 ベースイメージは、Ironic コンポーネントでの CentOS Stream 9、Python 3.8、および Python 3.9 のサポートを追加します。

Ironic プロビジョニングサービスの詳細については、インストーラーによってプロビジョニングされたクラスターをベアメタルにデプロイする を参照してください。

1.3.2.16. RHOSP 上で実行されるクラスターのクラウドプロバイダー設定の更新

OpenShift Container Platform 4.12 では、Red Hat OpenStack Platform (RHOSP) で実行されるクラスターが、従来の OpenStack クラウドプロバイダーから外部の Cloud Controller Manager (CCM) に切り替えられました。これは、Kubernetes がツリー内の従来のクラウドプロバイダーから、Cloud Controller Manager を使用して実装される外部クラウドプロバイダーに移行したことに伴う変更です。

詳細は、OpenStack Cloud Controller Manager を参照してください。

1.3.2.17. RHOSP 分散コンピュートノードでのワークロードのサポート

OpenShift Container Platform 4.12 では、分散コンピュートノード (DCN) アーキテクチャーを持つ Red Hat OpenStack Platform (RHOSP) クラウドへのクラスターデプロイメントが検証されました。これらのデプロイメントのリファレンスアーキテクチャーは近日中に公開されます。

このタイプのデプロイメントについては、Deploying Your Cluster at the Edge With OpenStack のブログ投稿で概説されています。

1.3.2.18. AWS Outposts 上の OpenShift Container Platform (テクノロジープレビュー)

OpenShift Container Platform 4.12 がテクノロジープレビューとして AWS Outposts プラットフォームでサポートされるようになりました。AWS Outposts を使用すると、コントロールプレーンノードに AWS リージョンを使用しながら、エッジベースのワーカーノードをデプロイできます。詳細は、AWS Outposts のリモートワーカーを使用して AWS にクラスターをインストールする を参照してください。

1.3.2.19. エージェントベースのインストールで 2 つの入力モードをサポート

エージェントベースのインストールでは、次の 2 つの入力モードがサポートされています。

- **install-config.yaml** ファイル
- **agent-config.yaml** ファイル

オプション

- ゼロタッチプロビジョニング (ZTP) マニフェスト

優先モードでは、**install-config.yaml** ファイルを設定し、**agent-config.yaml** ファイルでエージェントベースの特定の設定を指定できます。詳細は、エージェントベースの OpenShift Container Platform インストーラーについて を参照してください。

1.3.2.20. エージェントベースのインストールで FIPS 準拠モードでの OpenShift Container Platform クラスターのインストールをサポート

エージェントベースの OpenShift Container Platform インストーラーは、Federal Information Processing Standards (FIPS) 準拠モードで OpenShift Container Platform クラスターをサポートします。**install-config.yaml** ファイルで **fips** フィールドの値を **True** に設定する必要があります。詳細は、FIPS 準拠について を参照してください。

1.3.2.21. 非接続環境におけるエージェントベースの OpenShift Container Platform クラスターのデプロイメント

非接続環境でエージェントベースのインストールを実行できます。非接続環境で使用されるイメージを作成するには、**install-config.yaml** ファイルの **imageContentSources** セクションにミラー情報または **registries.conf** ファイルが含まれている必要があります (ZTP マニフェストを使用している場合)。これらのファイルで使用する実際の設定オプションは、**oc adm release mirror** または **oc mirror** コマンドによって提供されます。詳細は、非接続インストールのミラーリングについて を参照してください。

1.3.2.22. エージェントベースのインストールでシングルおよびデュアルスタックネットワークをサポート

次の IP アドレス設定でエージェント ISO イメージを作成できます。

- IPv4
- IPv6
- IPv4 と IPv6 の並列 (デュアルスタック)

注記

IPv6 は、ベアメタルプラットフォームでのみサポートされます。

詳細は、デュアルおよびシングル IP スタッククラスター を参照してください。

1.3.2.23. エージェントがデプロイされた OpenShift Container Platform クラスターをハブクラスターとして使用可能

Kubernetes Operator 用のマルチクラスターエンジンをインストールし、エージェントベースの OpenShift Container Platform インストーラーを使用してハブクラスターをデプロイできます。詳細は、Kubernetes Operator 用マルチクラスターエンジンのエージェントベースでインストールされたクラスターの準備 を参照してください。

1.3.2.24. エージェントベースのインストールにおけるインストール検証

エージェントベースの OpenShift Container Platform インストーラーは、以下を検証します。

- インストールイメージの生成: ユーザー提供のマニフェストの有効性と互換性をチェックします。
- インストール: インストールサービスは、インストールに使用できるハードウェアをチェックし、**openshift-install agent wait-for** サブコマンドで取得できる検証イベントを発行します。

詳細は、インストールの検証 を参照してください。

1.3.2.25. エージェントベースのインストールにおける静的ネットワークの設定

エージェントベースの OpenShift Container Platform インストーラーを使用すると、エージェント ISO イメージを作成する前に、すべてのホストの IPv4、IPv6、またはデュアルスタック (IPv4 と IPv6 の両方) の静的 IP アドレスを設定できます。ZTP マニフェストを使用している場合、**agent-config.yaml** ファイルまたは **NMStateConfig.yaml** ファイルの **hosts** セクションに静的アドレスを追加できます。アドレスの設定は、NMState state examples で説明されているように、NMState の構文規則に従う必要があることに注意してください。

注記

IPv6 は、ベアメタルプラットフォームでのみサポートされます。

詳細は、ネットワークについて を参照してください。

1.3.2.26. エージェントベースのインストールにおける CLI ベースの自動デプロイメント

エージェントベースの OpenShift Container Platform インストーラーを使用すると、インストール設定を定義し、すべてのノードの ISO を生成してから、生成された ISO を使用してターゲットシステムを起動することで無人インストールを実行できます。詳細は、エージェントベースの OpenShift Container Platform インストーラーを使用した OpenShift Container Platform クラスターのインストール を参照してください。

1.3.2.27. エージェントベースのインストールでインストール時におけるホスト固有の設定をサポート

エージェントベースのインストールでは、ホスト名、NMState 形式のネットワーク設定、ルートデバイスヒント、ロールを設定できます。

詳細は、ルートデバイスヒントについて を参照してください。

1.3.2.28. エージェントベースのインストールで DHCP をサポート

エージェントベースの OpenShift Container Platform インストーラーを使用すると、DHCP に依存して全ノードのネットワークを設定する環境にデプロイできます。この IP は、すべてのノードがミーティングポイントとして使用するために必要です。詳細は、DHCP を参照してください。

1.3.3. インストール後の設定

1.3.3.1. vSphere クラスターへの CSI ドライバーのインストール

vSphere で実行されているクラスターに CSI ドライバーをインストールするには、次の要件を満たす必要があります。

- ハードウェアバージョン 15 以降の仮想マシン
- VMware vSphere バージョン 7.0 Update 2 以降、バージョン 7 まで。vSphere 8 はサポートされていません。
- vCenter 7.0 Update 2 以降、バージョン 8 まで。vCenter 7 はサポートされていません。
- サードパーティーの CSI ドライバーがクラスターにインストールされていない
  サードパーティーの CSI ドライバーがクラスターに存在する場合、OpenShift Container Platform はそれを上書きしません。

上記よりも前のバージョンのコンポーネントは引き続きサポートされますが、非推奨です。これらのバージョンは引き続き完全にサポートされていますが、OpenShift Container Platform のバージョン 4.12 には、vSphere 仮想ハードウェアバージョン 15 以降が必要です。詳細は、非推奨および削除された機能 を参照してください。

上記の要件を満たさない場合、OpenShift Container Platform は OpenShift Container Platform 4.13 以降にアップグレードできません。

1.3.3.2. クラスター機能

次の新しいクラスター機能が追加されました。

- Console
- Insights
- ストレージ
- CSISnapshot

新しい定義済みクラスター機能セット (**v4.12**) が追加されました。これには、**v4.11** のすべての機能と、今回のリリースで追加された新しい機能が含まれます。

詳細は、クラスター機能の有効化 を参照してください。

1.3.3.3. マルチアーキテクチャーのコンピュートマシンを含む OpenShift Container Platform (テクノロジープレビュー)

マルチアーキテクチャーのコンピュートマシンが含まれる OpenShift Container Platform 4.12 では、イ

メージストリームでマニフェストにリストされたイメージがサポートされるようになりました。マニフェストリストイメージの詳細は、OpenShift Container Platform クラスター上でのマルチアーキテクチャーコンピュートマシンの設定 を参照してください。

マルチアーキテクチャーのコンピュートマシンを含むクラスターでは、Operator の **Subscription** オブジェクトでノードアフィニティーをオーバーライドして、Operator がサポートするアーキテクチャーのノードで Pod をスケジュールできるようになりました。詳細は、ノードアフィニティーを使用した Operator のインストール場所の制御 を参照してください。

1.3.4. Web コンソール

1.3.4.1. 管理者パースペクティブ

今回のリリースにより、Web コンソールの **Administrator** パースペクティブに複数の更新が追加されました。

- クラスターがアップグレード中の場合、OpenShift Container Platform Web コンソールは **ConsoleNotification** を表示します。アップグレードが完了すると、通知は削除されます。
- Action および Kebab メニューでは、**Deployment** リソースの **再起動ロールアウト** オプションと **DeploymentConfig** リソースの **再試行** ロールアウトオプションを使用できます。
- **All Clusters** ドロップダウンリストで、サポート対象クラスターのリストを表示できます。サポート対象のクラスターには、OpenShift Container Platform、OpenShift Container Platform Service on AWS (ROSA)、Azure Red Hat OpenShift (ARO)、ROKS、Red Hat OpenShift Dedicated が含まれます。

1.3.4.1.1. OpenShift Container Platform クラスターでのマルチアーキテクチャーコンピュートマシンの設定

console-operator がすべてのノードをスキャンし、クラスターノードが実行されるすべてのアーキテクチャータイプのセットを構築して、それを **console-config.yaml** に渡すようになりました。**console-operator** は、値が **amd64**、**arm64**、**ppc64le**、または **s390x** のアーキテクチャーを持つノードにインストールできます。

マルチアーキテクチャーコンピュートマシンの詳細について は、OpenShift クラスターでのマルチアーキテクチャー計算マシンの設定 を参照してください。

1.3.4.1.2. 動的プラグインの一般提供

この機能は、OpenShift Container Platform 4.10 でテクノロジープレビューとして導入され、OpenShift Container Platform 4.12 で一般提供が開始されました。動的プラグインを使用すると、高品質で独自のユーザーエクスペリエンスを Web コンソールでネイティブにビルドできます。これにより、以下が可能になります。

- カスタムページの追加。
- 管理者と開発者を超えたパースペクティブを追加します。
- ナビゲーション項目の追加。
- リソースページへのタブおよびアクションの追加。
- 既存ページの拡張。

詳細は、動的プラグインの概要 を参照してください。

1.3.4.2. 開発者パースペクティブ

今回のリリースにより、Web コンソールの **Developer** パースペクティブに複数の更新が含まれるようになりました。次のアクションを実行できます。

- **+Add** ページの **Export application** オプションを使用して、ZIP ファイル形式でアプリケーションを別のプロジェクトまたはクラスターにエクスポートします。
- Kafka イベントシンクを作成して、特定のソースからイベントを受信し、Kafka トピックに送信できるようになりました。
- **User Preferences → Applications** ページでデフォルトのリソース設定を指定します。さらに、別のリソースタイプをデフォルトとして選択できます。
    - 必要に応じて、**Import from Git → Advanced options → Resource type** をクリックし、ドロップダウンリストからリソースを選択して、**Add** ページから別のリソースタイプを設定します。
- Pod の **status.HostIP** ノード IP アドレスが **Pods** ページの **Details** タブに表示されるようにします。
- リソースがクォータに達するたびに、**トポロジー** ページと **追加** ページのリソースクォータアラートラベルを参照してください。アラートラベルのリンクをクリックすると、**ResourceQuotas** リストページに移動します。アラートラベルのリンクが単一のリソースクォータに対するものである場合は、**ResourceQuota の詳細** ページに移動します。
    - デプロイでは、エラーがリソースクォータに関連付けられている場合、トポロジーノードのサイドパネルにアラートが表示されます。また、リソースクォータを超えると、デプロイノードの周囲に黄色の境界線が表示されます。
- フォームまたは YAML ビューを使用して、次の UI アイテムをカスタマイズします。
    - ユーザーに表示されるパースペクティブ
    - ユーザーに表示されるクイックスタート
    - プロジェクトにアクセス可能なクラスターロール
    - **+Add** ページに表示されるアクション
    - **Developer Catalog** のアイテムタイプ
- 次のアクションを実行して、**Pipeline details** と **PipelineRun details** ページの視覚化に対する一般的な更新を確認します。
    - ズーム倍率を変更するには、マウスホイールを使用します。
    - タスクにカーソルを合わせると、タスクの詳細が表示されます。
    - ズームイン、ズームアウト、画面に合わせる、ビューのリセットには、標準アイコンを使用します。
    - **PipelineRun details** ページのみ: 特定のズーム要素では、タスクの背景色が変更され、エラーまたは警告のステータスが示されます。タスクバッジにカーソルを合わせると、タスクと完了したタスクの総数が表示されます。

1.3.4.2.1. Helm ページの改善

OpenShift Container Platform 4.12 では、**Helm** ページから以下を実行できます。

- **Create** ボタンを使用して、Helm リリースとリポジトリーを作成します。
- クラススコープまたは namespace スコープの Helm チャートリポジトリーを作成、更新、または削除します。
- 既存の Helm チャートリポジトリーのリストとそのスコープを **Repositories** ページで表示できるようになりました。
- **Helm Releases** ページで、新しく作成された Helm リリースを表示します。

1.3.4.2.2. Alertmanager の Negative matcher

今回の更新により、Alertmanager で **Negative matcher** オプションがサポートされるようになりました。**Negative matcher** を使用すると、**Label 値** を Not Equals matcher に更新できます。Negative matcher チェックボックスにより、**=** (等しい値) が **!=** (等しくない値) に変更され、**=~** (正規表現に一致する値) が **!~** (正規表現に一致しない値) に変更されます。また、**Use RegEx** チェックボックスのラベル名が **RegEx** に変更されました。

1.3.5. OpenShift CLI (oc)

1.3.5.1. Krew を使用した OpenShift CLI のプラグインの管理 (テクノロジープレビュー)

Krew を使用して OpenShift CLI (**oc**) のプラグインをインストールおよび管理する機能が、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。

詳細は、Krew を使用した CLI プラグインの管理 を参照してください。

1.3.6. IBM Z および LinuxONE

本リリースでは、IBM Z および LinuxONE は OpenShift Container Platform 4.12 と互換性があります。インストールは z/VM または RHEL KVM で実行できます。インストール手順については、以下のドキュメントを参照してください。

- z/VM のあるクラスターの IBM Z および LinuxONE へのインストール
- ネットワークが制限された環境での z/VM のあるクラスターの IBM Z および LinuxONE へのインストール
- RHEL KVM を使用したクラスターの IBM Z および LinuxONE へのインストール
- ネットワークが制限された環境での RHEL KVM のあるクラスターの IBM Z および LinuxONE へのインストール

主な機能拡張
以下の新機能は、OpenShift Container Platform 4.12 の IBM Z および LinuxONE でサポートされます。

- Cron ジョブ
- Descheduler
- IPv6
- PodDisruptionBudget

- スケジューラーのプロファイル
- SCTP (Stream Control Transmission Protocol)

IBM Secure Execution (テクノロジープレビュー)
OpenShift Container Platform は、テクノロジープレビュー機能として IBM zSystems および LinuxONE (s390x アーキテクチャー)での IBM Secure Execution 用の Red Hat Enterprise Linux CoreOS (RHCOS) ノードの設定をサポートするようになりました。

インストール手順については、以下のドキュメントを参照してください。

- IBM Secure Execution を使用した RHCOS のインストール

サポートされる機能
以下の機能が IBM Z および LinuxONE でもサポートされるようになりました。

- 現時点で、以下の Operator がサポートされています。
  - Cluster Logging Operator
  - コンプライアンス Operator
  - File Integrity Operator
  - Local Storage Operator
  - NFD Operator
  - NMState Operator
  - OpenShift Elasticsearch Operator
  - サービスバインディング Operator
  - Vertical Pod Autoscaler Operator
- 以下の Multus CNI プラグインがサポートされます。
  - ブリッジ
  - host-device
  - IPAM
  - IPVLAN
- 代替の認証プロバイダー
- ローカルストレージ Operator を使用した自動デバイス検出
- CSI ボリューム
  - クローン
  - 拡張
  - スナップショット

- etcd に保存されるデータの暗号化
- Helm
- Horizontal Pod Autoscaling
- ユーザー定義プロジェクトのモニターリング
- マルチパス化
- Operator API
- OC CLI プラグイン
- iSCSI を使用した永続ストレージ
- ローカルボリュームを使用した永続ストレージ (Local Storage Operator)
- hostPath を使用した永続ストレージ
- ファイバーチャネルを使用した永続ストレージ
- Raw Block を使用した永続ストレージ
- IPsec 暗号化を含む OVN-Kubernetes
- 複数ネットワークインターフェイスのサポート
- 3 ノードクラスターのサポート
- SCSI ディスク上の z/VM Emulated FBA デバイス
- 4k FCP ブロックデバイス

これらの機能は、4.12 の IBM Z および LinuxONE の OpenShift Container Platform についてのみ利用できます。

- IBM Z および LinuxONE で有効にされている HyperPAV (FICON 接続の ECKD ストレージの仮想マシン用)。

制約
以下の制限は、IBM Z および LinuxONE の OpenShift Container Platform に影響します。

- マシンヘルスチェックによる障害のあるマシンの自動修復
- Red Hat OpenShift Local
- オーバーコミットの制御およびノード上のコンテナーの密度の管理
- NVMe
- OpenShift Metering
- OpenShift Virtualization
- Precision Time Protocol (PTP) ハードウェア
- OpenShift Container Platform のデプロイメント時の Tang モードのディスク暗号化

- コンピュートノードは Red Hat Enterprise Linux CoreOS (RHCOS) を実行する必要があります。
- 永続共有ストレージは、Red Hat OpenShift Data Foundation またはその他のサポートされるストレージプロトコルを使用してプロビジョニングする必要があります。
- 共有されていない永続ストレージは、iSCSI、FC、DASD、FCP または EDEV/FBA と共に LSO を使用するなど、ローカルストレージを使用してプロビジョニングする必要があります。

1.3.7. IBM Power

本リリースでは、IBM Power は OpenShift Container Platform 4.12 と互換性があります。インストール手順については、以下のドキュメントを参照してください。

- クラスターの IBM Power へのインストール
- ネットワークが制限された環境での IBM Power へのクラスターのインストール

主な機能拡張
以下の新機能は、OpenShift Container Platform 4.12 の IBM Power でサポートされます。

- IBM Cloud 向けクラウドコントローラーマネージャー
- Cron ジョブ
- Descheduler
- PodDisruptionBudget
- スケジューラーのプロファイル
- SCTP (Stream Control Transmission Protocol)
- Topology Manager

サポートされる機能
以下の機能は、IBM Power でもサポートされています。

- 現時点で、以下の Operator がサポートされています。
  - Cluster Logging Operator
  - コンプライアンス Operator
  - File Integrity Operator
  - Local Storage Operator
  - NFD Operator
  - NMState Operator
  - OpenShift Elasticsearch Operator
  - SR-IOV ネットワーク Operator
  - サービスバインディング Operator

  - Vertical Pod Autoscaler Operator
- 以下の Multus CNI プラグインがサポートされます。
  - ブリッジ
  - host-device
  - IPAM
  - IPVLAN
- 代替の認証プロバイダー
- CSI ボリューム
  - クローン
  - 拡張
  - スナップショット
- etcd に保存されるデータの暗号化
- Helm
- Horizontal Pod Autoscaling
- IPv6
- ユーザー定義プロジェクトのモニターリング
- マルチパス化
- Multus SR-IOV
- Operator API
- OC CLI プラグイン
- IPsec 暗号化を含む OVN-Kubernetes
- iSCSI を使用した永続ストレージ
- ローカルボリュームを使用した永続ストレージ (Local Storage Operator)
- hostPath を使用した永続ストレージ
- ファイバーチャネルを使用した永続ストレージ
- Raw Block を使用した永続ストレージ
- 複数ネットワークインターフェイスのサポート
- Power10 のサポート
- 3 ノードクラスターのサポート

- 4K ディスクのサポート

制約
以下の制限は、OpenShift Container Platform が IBM Power に影響を与えます。

- マシンヘルスチェックによる障害のあるマシンの自動修復
- Red Hat OpenShift Local
- オーバーコミットの制御およびノード上のコンテナーの密度の管理
- OpenShift Metering
- OpenShift Virtualization
- Precision Time Protocol (PTP) ハードウェア
- OpenShift Container Platform のデプロイメント時の Tang モードのディスク暗号化
- コンピュートノードは Red Hat Enterprise Linux CoreOS (RHCOS) を実行する必要があります。
- 永続ストレージは、ローカルボリューム、Red Hat OpenShift Data Foundation、Network File System (NFS)、または Container Storage Interface (CSI) を使用する Filesystem タイプである必要があります。

1.3.8. イメージ

新しいインポート値 **importMode** がイメージストリームの **importPolicy** パラメーターに追加されました。この値は、以下のフィールドを使用できます。

- **legacy**:**Legacy** は **importMode** のデフォルト値です。アクティブな場合、マニフェストリストは破棄され、単一のサブマニフェストがインポートされます。プラットフォームは、以下の優先順位で選択されます。
  1. タグのアノテーション
  2. コントロールプレーンアーキテクチャー
  3. Linux/AMD64
  4. 一覧の最初のマニフェスト
- **PreserveOriginal**: アクティブな場合は、元のマニフェストは保持されます。マニフェスト一覧の場合は、マニフェストの一覧とそのすべてのサブマニフェストがインポートされます。

1.3.9. セキュリティーおよびコンプライアンス

1.3.9.1. Security Profiles Operator

OpenShift Container Platform 4.12 以降で、Security Profiles Operator (SPO) が使用可能になりました。

SPO を使用することで、セキュアコンピューティング (seccomp) プロファイルと SELinux プロファイルをカスタムリソースとして定義し、特定の namespace 内のすべてのノードにプロファイルを同期できます。

詳細は、Security Profiles Operator の概要 を参照してください。

1.3.10. ネットワーク

1.3.10.1. API VIP および Ingress VIP のデュアルスタックアドレス指定のサポート

Assisted Installer は、API VIP および Ingress VIP のデュアルスタックネットワークを備えた OpenShift Container Platform 4.12 以降のバージョンのベアメタル上でのインストールのみサポートしています。このサポートにより、IP アドレスのリストを取得できる **api_vips** と **ingress_vips** という 2 つの新しい設定が導入されます。従来の **api_vip** および **ingress_vip** 設定も OpenShift Container Platform 4.12 で行う必要があります。ただし、これらは 1 つの IP アドレスしか使用しないため、従来の **api_vip** および **ingress_vip** 設定を使用して、API VIP および Ingress VIP のデュアルスタックネットワークを設定する場合は、IPv4 アドレスを設定する必要があります。

デュアルスタックネットワークを使用する場合は、API VIP アドレスおよび Ingress VIP アドレスはプライマリー IP アドレスファミリーである必要があります。現在、Red Hat は、IPv6 をプライマリー IP アドレスファミリーとして使用するデュアルスタック VIP またはデュアルスタックネットワークをサポートしていません。ただし、Red Hat は、IPv4 をプライマリー IP アドレスファミリーとして使用するデュアルスタックネットワークをサポートしています。したがって、IPv6 エントリーの前に IPv4 エントリーを配置する必要があります。詳細については、自動インストーラーのドキュメントを参照してください。

1.3.10.2. Red Hat OpenShift Networking

Red Hat OpenShift Networking は、クラスターが 1 つまたは複数のハイブリッドクラスターのネットワークトラフィックを管理するために必要な高度なネットワーキング関連機能を使用して、Kubernetes CNI プラグインを超えて Kubernetes ネットワーキングを拡張する機能、プラグイン、高度なネットワーキング機能のエコシステムです。このネットワーキング機能のエコシステムは、イングレス、エグレス、負荷分散、高性能スループット、セキュリティー、クラスター間およびクラスター内のトラフィック管理を統合し、ロールベースの可観測性ツールを提供して本来の複雑さを軽減します。

詳細は、ネットワークについて を参照してください。

1.3.10.3. OVN-Kubernetes がデフォルトのネットワークプラグインになりました

新しいクラスターをインストールすると、OVN-Kubernetes ネットワークプラグインがデフォルトのネットワークプラグインになります。これまでの OpenShift Container Platform バージョンは、すべて OpenShift SDN がデフォルトのネットワークプラグインでした。

OVN-Kubernetes ネットワークプラグインには、OpenShift SDN よりも幅広い機能が含まれています。

- 既存のすべての OpenShift SDN 機能のサポート
- IPv6 ネットワーク のサポート
- IPsec 暗号化の設定 のサポート
- **NetworkPolicy** API の完全なサポート
- ネットワークポリシーイベントの監査ログの サポート
- NetFlow、sFlow、および IPFIX 形式での ネットワークフロートラッキング のサポート
- Windows コンテナーの ハイブリッドネットワーク のサポート

- 互換性のある NIC への ハードウェアオフロード のサポート

以前のバージョンと比較して、OpenShift Container Platform 4.12 ではスケール、パフォーマンス、および安定性が大幅に向上しています。

OpenShift SDN ネットワークプラグインを使用している場合は、次の点に注意してください。

- OpenShift SDN を使用した既存および今後のデプロイメントは、引き続きサポートされます。
- 4.12 より前の OpenShift Container Platform バージョンでは、引き続き OpenShift SDN がデフォルトとなります。
- OpenShift Container Platform 4.12 以降、OpenShift SDN はインストール時にサポートされるオプションとなります。
- OpenShift SDN は引き続き凍結機能となります。

OpenShift SDN との機能比較表など、OVN-Kubernetes の詳細 については、OVN-Kubernetes ネットワークプラグインについて を参照してください。

OpenShift SDN から OVN-Kubernetes への移行は、OpenShift SDN ネットワークプラグインからの移行 を参照してください。

1.3.10.4. Ingress Node Firewall Operator

今回の更新では、新しいステートレス Ingress Node Firewall Operator が導入されています。ノードレベルでファイアウォールルールを設定できるようになりました。詳細は、Ingress Node Firewall Operator を参照してください。

1.3.10.5. ネットワークメトリクスの強化

OVN-Kubernetes ネットワークプラグインで次のメトリックを使用できるようになりました。

- **ovn_controller_southbound_database_connected**
- **ovnkube_master_libovsdb_monitors**
- **ovnkube_master_network_programming_duration_seconds**
- **ovnkube_master_network_programming_ovn_duration_seconds**
- **ovnkube_master_egress_routing_via_host**
- **ovs_vswitchd_interface_resets_total**
- **ovs_vswitchd_interface_rx_dropped_total**
- **ovs_vswitchd_interface_tx_dropped_total**
- **ovs_vswitchd_interface_rx_errors_total**
- **ovs_vswitchd_interface_tx_errors_total**
- **ovs_vswitchd_interface_collisions_total**

次のメトリックが削除されました。

- **ovnkube_master_skipped_nbctl_daemon_total**

1.3.10.6. マルチゾーンインストーラーによりプロビジョニングされたインフラストラクチャー VMware vSphere のインストール (テクノロジープレビュー)

OpenShift Container Platform 4.12 からテクノロジープレビュー機能として、インストーラーによりプロビジョニングされたインフラストラクチャーを使用した単一 vCenter のインストールで複数の vCenter データセンターと複数の vCenter クラスターを設定できるようになりました。vCenter タグを使用すると、この機能を使用して、vCenter データセンターとコンピュートクラスターを openshift-region と openshift-zone に関連付けることができます。これらの関連付けによって障害ドメインが定義され、アプリケーションのワークロードを特定のロケーションおよび障害ドメインに関連付けることができます。

1.3.10.7. VMware vSphere の Kubernetes NMState がサポートされるようになりました

OpenShift Container Platform 4.12 以降、VMware vSphere インスタンスで Kubernetes NMState Operator を使用して、DNS サーバーまたは検索ドメイン、VLAN、ブリッジ、およびインターフェイス結合などのネットワーク設定を設定できます。

詳細は、About the Kubernetes NMState Operator を参照してください。

1.3.10.8. OpenStack の Kubernetes NMState がサポートされるようになりました

OpenShift Container Platform 4.12 以降、OpenStack インスタンスで Kubernetes NMState Operator を使用して、DNS サーバーまたは検索ドメイン、VLAN、ブリッジ、およびインターフェイス結合などのネットワーク設定を設定できます。

詳細は、About the Kubernetes NMState Operator を参照してください。

1.3.10.9. 外部 DNS Operator

OpenShift Container Platform 4.12 では、External DNS Operator が AzureDNS の ExternalDNS ワイルドカード TXT レコードの形式を変更します。External DNS Operator は、アスタリスクを ExternalDNS ワイルドカード TXT レコードの **any** のものに置き換えます。競合が発生する可能性があるため、ExternalDNS ワイルドカード A および CNAME レコードが **any** の左端のサブドメインを持つことは避ける必要があります。

OpenShift Container Platform 4.12 の **ExternalDNS** のアップストリームバージョンは v0.13.1 です。

1.3.10.10. ルートとシャードの使用に関連するメトリクスと Telemetry のキャプチャ

OpenShift Container Platform 4.12 では、Cluster Ingress Operator は **route_metrics_controller_routes_per_shard** という名前の新しいメトリクスをエクスポートします。メトリックの **shard_name** ラベルは、シャードの名前を指定します。このメトリクスは、各シャードによって許可されたルートの総数を示します。

次のメトリックは、Telemetry を通じて送信されます。

表1.1 Telemetry を通じて送信される指標

名前	ルール式の記録	説明

名前	ルール式の記録	説明
cluster:route_metrics_controller_routes_per_shard:min	**min(route_metrics_controller_routes_per_shard)**	シャードのいずれかによって許可されたルートの最小数を追跡します
cluster:route_metrics_controller_routes_per_shard:max	**max(route_metrics_controller_routes_per_shard)**	シャードのいずれかによって許可されたルートの最大数を追跡します
cluster:route_metrics_controller_routes_per_shard:avg	**avg(route_metrics_controller_routes_per_shard)**	**route_metrics_controller_routes_per_shard** メトリックの平均値を追跡します
cluster:route_metrics_controller_routes_per_shard:median	**quantile(0.5, route_metrics_controller_routes_per_shard)**	**route_metrics_controller_routes_per_shard** メトリクスの中央値を追跡します
cluster:openshift_route_info:tls_termination:sum	**sum (openshift_route_info) by (tls_termination)**	各 **tls_termination** 値のルート数を追跡します。**tls_termination** の可能な値は、**edge**、**passthrough**、および **reencrypt** です。

1.3.10.11. AWS Load Balancer Operator

OpenShift Container Platform 4.12 では、AWS Load Balancer コントローラーは、複数の一致に対して Kubernetes Ingress 仕様を実装するようになりました。Ingress 内の複数のパスが要求に一致する場合、一致する最長のパスが優先されます。2 つのパスが引き続き一致する場合は、正確なパスタイプのパスが接頭辞パスタイプよりも優先されます。

AWS Load Balancer Operator は、**EnableIPTargetType** 機能ゲートを **false** に設定します。AWS Load Balancer コントローラーは、**target-type ip** のサービスとイングレスリソースのサポートを無効にします。

OpenShift Container Platform 4.12 の **aws-load-balancer-controller** のアップストリームバージョンは v2.4.4 です。

1.3.10.12. イングレスコントローラーの自動スケーリング (テクノロジープレビュー)

OpenShift Container Platform Custom Metrics Autoscaler Operator を使用して、デプロイされたクラスター内のメトリック (使用可能なワーカーノードの数など) に基づいて、デフォルトの Ingress コントローラーを動的にスケーリングできるようになりました。カスタムメトリクスオートスケーラーは、テクノロジープレビュー機能として利用できます。

詳細は、イングレスコントローラーの自動スケーリング を参照してください。

1.3.10.13. HAProxy maxConnections のデフォルトは 50,000 になりました

OpenShift Container Platform 4.12 では、**maxConnections** 設定のデフォルト値が 50000 になりました。OpenShift Container Platform 4.11 以降では、**maxConnections** 設定のデフォルト値は 20000 でした。

詳細は、Ingress Controller configuration parameters を参照してください。

1.3.10.14. 手動 DNS 管理のための Ingress コントローラーの設定

自動 DNS 管理を停止し、手動 DNS 管理を開始するように Ingress コントローラーを設定できるようになりました。**dnsManagementPolicy** パラメーターを設定して、自動または手動の DNS 管理を指定します。

詳細については、DNS を手動で管理するための Ingress コントローラーの設る を参照してください。

1.3.10.15. SR-IOV (Single Root I/O Virtualization) でサポートされるハードウェア

OpenShift Container Platform 4.12 は以下の SR-IOV デバイスのサポートを追加します。

- MT2892 Family [ConnectX-6 Dx]
- MT2894 Family [ConnectX-6 Lx]
- ConnectX-6 NIC モードの MT42822 BlueField-2
- Silicom STS ファミリー

詳細は、サポート対象のデバイス を参照してください。

1.3.10.16. OvS (Open vSwitch) Hardware Offload のサポート対象ハードウェア

OpenShift Container Platform 4.12 は、以下のデバイスで OvS Hardware Offload のサポートを追加しています。

- MT2892 Family [ConnectX-6 Dx]
- MT2894 Family [ConnectX-6 Lx]
- ConnectX-6 NIC モードの MT42822 BlueField-2

詳細は、サポート対象のデバイス を参照してください。

1.3.10.17. SR-IOV (テクノロジープレビュー) でサポートされるマルチネットワークポリシー

OpenShift Container Platform 4.12 は、SR-IOV デバイスのマルチネットワークポリシーを設定するためのサポートを追加します。

SR-IOV 追加ネットワークのマルチネットワークを設定できるようになりました。SR-IOV 追加ネットワークの設定はテクノロジープレビュー機能であり、カーネルネットワークインターフェイスカード (NIC) でのみサポートされます。

詳細は、マルチネットワークポリシーの設定 を参照してください。

1.3.10.18. Ingress Controller を削除せずに AWS ロードバランサーのタイプを切り替える

Ingress Controller を削除せずに、AWS Classic Load Balancer (CLB) と AWS Network Load Balancer (NLB) を切り替えるように Ingress Controller を更新できます。

詳細は、Configuring ingress cluster traffic on AWS を参照してください。

1.3.10.19. SR-IOV CNI プラグインで、IPv6 未承認ネイバーアドバタイズメントと IPv4 Gratuitous アドレス解決プロトコルがデフォルトになりました

IP アドレス管理 CNI プラグインが IP を割り当てたシングルルート I/O 仮想化 (SR-IOV) CNI プラグインで作成された Pod は、IPv6 未承認ネイバーアドバタイズメントおよび/または IPv4 Gratuitous アドレス解決プロトコルをデフォルトでネットワークへ送信します。この機能強化により、特定の IP の新しい Pod の MAC アドレスがホストに通知され、正しい情報で ARP/NDP キャッシュが更新されます。

詳細は、サポート対象のデバイス を参照してください。

1.3.10.20. CoreDNS キャッシュチューニングのサポート

CoreDNS によってキャッシュされた成功および失敗した DNS クエリーの存続時間 (TTL) 期間を設定できるようになりました。

詳細については、CoreDNS キャッシュの調整 を参照してください。

1.3.10.21. OVN-Kubernetes は内部サブネットの設定をサポートします

以前は、OVN-Kubernetes が内部で使用するサブネットは、IPv4 の場合は **100.64.0.0/16**、**IPv6** の場合は **fd98::/48** であり、変更できませんでした。これらのサブネットがインフラストラクチャー内の既存のサブネットと重複するインスタンスをサポートするために、これらの内部サブネットを変更して重複を回避できるようになりました。

詳細は、Cluster Network Operator 設定オブジェクト を参照してください。

1.3.10.22. Red Hat OpenStack Platform (RHOSP) での Egress IP サポート

RHOSP は OpenShift Container Platform と組み合わせることで、Egress IP アドレスの自動アタッチおよびデタッチをサポートするようになりました。任意の数の namespace に含まれる 1 つ以上の Pod からのトラフィックでは、クラスター外のサービスのソース IP アドレスに一貫性があります。このサポートは、デフォルトのネットワークプロバイダーとして OpenShift SDN および OVN-Kubernetes に適用されます。

1.3.10.23. OpenShift SDN から OVN-Kubernetes への機能移行のサポート

OpenShift SDN ネットワークプラグインから OVN-Kubernetes ネットワークプラグインへの移行を計画している場合、次の機能の設定は、OVN-Kubernetes で動作するように自動的に変換されます。

- Egress IP アドレス
- Egress ファイアウォール
- Multicast

OVN-Kubernetes への移行がどのように機能するかの詳細は、OpenShift SDN クラスターネットワークプロバイダーからの移行 を参照してください。

1.3.10.24. エグレスファイアウォールの監査ログ

OVN-Kubernetes ネットワークプラグインの場合、エグレスファイアウォールは、ネットワークポリシー監査ログが使用するのと同じメカニズムを使用して、監査ログをサポートします。詳細は、エグレスファイアウォールとネットワークポリシールールのロギングを 参照してください。

1.3.10.25. ノードのサブセットから指定されたアドレスプールから MetalLB をアドバタイズする

今回の更新により、BGP モードで、ノードセレクターを使用して、IP アドレスの特定のプールを使用して、ノードのサブセットから MetalLB サービスをアドバタイズできるようになりました。この機能は、OpenShift Container Platform 4.11 でテクノロジープレビュー機能として導入され、BGP モードのみで OpenShift Container Platform 4.12 で一般的に利用できるようになりました。L2 モードは、テクノロジープレビュー機能のままです。

詳細は、ノードのサブセットからの IP アドレスプールのアドバタイズ を参照してください。

1.3.10.26. MetalLB の追加のデプロイメント仕様

この更新では、MetalLB の追加のデプロイメント仕様が提供されます。カスタムリソースを使用して MetalLB をデプロイする場合、これらの追加のデプロイ仕様を使用して、MetalLB **speaker** および **controller** Pod がクラスターでデプロイおよび実行される方法を管理できます。たとえば、MetalLB デプロイメント仕様を使用して、MetalLB Pod がデプロイされる場所を管理し、MetalLB Pod の CPU 制限を定義し、MetalLB Pod にランタイムクラスを割り当てることができます。

MetalLB のデプロイメント仕様について詳しくは、MetalLB の デプロイメント仕様 を参照してください。

1.3.10.27. ノード IP 選択の改善

以前は、クラスターホストの **nodeip-configuration** サービスが、デフォルトルートが使用するインターフェイスから IP アドレスを選択していました。複数のルートが存在する場合、サービスはメトリック値が最も低いルートを選択します。その結果、ネットワークトラフィックが不適切なインターフェイスから配信される可能性がありました。

OpenShift Container Platform 4.12 では、ユーザーがヒントファイルを作成できる新しいインターフェイスが **nodeip-configuration** サービスに追加されました。ヒントファイルには変数 **NODEIP_HINT** が含まれています。この変数は、デフォルトの IP 選択ロジックをオーバーライドし、サブネット **NODEIP_HINT** 変数から特定のノード IP アドレスを選択します。**NODEIP_HINT** 変数を使用すると、ユーザーは使用する IP アドレスを指定できるため、ネットワークトラフィックが正しいインターフェイスから分散されるようになります。

詳しくは、オプション: デフォルトのノード IP 選択ロジックのオーバーライド を参照してください。

1.3.10.28. CoreDNS がバージョン 1.10.0 に更新される

OpenShift Container Platform 4.12 では、CoreDNS はバージョン 1.10.0 を使用します。これには以下の変更が含まれます。

- 以前に小さい値に設定されていた場合、CoreDNS はクエリー UDP バッファーサイズを拡張しません。
- CoreDNS は、関連するログレベルで Kubernetes クライアントログの各ログ行に常に接頭辞を付けるようになりました。
- CoreDNS は、約 20 ミリ秒の速度でより高速にリロードされるようになりました。

1.3.10.29. HAProxy での設定可能なリロード間隔のサポート

今回の更新により、クラスター管理者はリロード間隔を設定して、ルートとエンドポイントの更新に応じて HAProxy が設定をリロードする頻度を減らすことができます。デフォルトの最小 HAProxy リロード間隔は 5 秒です。

詳細は、HAProxy の再ロード間隔の設定 を参照してください。

1.3.10.30. ネットワークトラフィックフローを監視する新しい Network Observability Operator

管理者は、Network Observability Operator をインストールして、コンソールで OpenShift Container Platform クラスターのネットワークトラフィックを監視できるようになりました。さまざまなグラフィック表現でネットワークトラフィックデータを表示および監視できます。Network Observability Operator は、eBPF テクノロジーを使用してネットワークフローを作成します。その後、ネットワークフローは OpenShift Container Platform 情報で強化され、Loki に保存されます。ネットワークトラフィック情報を使用して、詳細なトラブルシューティングと分析を行うことができます。

詳細は、Network Observability を参照してください。

1.3.10.31. RHOSP セカンダリーネットワークインターフェイスの IPv6

RHOSP 上で実行されるクラスターで、セカンダリーネットワークインターフェイスの IPv6 がサポートされるようになりました。

詳細は、RHOSP 上の Pod への IPv6 接続の有効化 を参照してください。

1.3.10.32. RHOSP 上のロードバランサーの UDP サポート

外部の OpenStack クラウドプロバイダーに切り替えた結果、そのプラットフォームで実行されるクラスターの **LoadBalancer** サービスで UDP がサポートされるようになりました。

1.3.10.33. ホステッドコントロールプレーンへの SR-IOV Operator のデプロイメント (テクノロジープレビュー)

ホスティングサービスクラスターを設定してデプロイした場合、ホステッドクラスターに SR-IOV Operator をデプロイできるようになりました。詳細は、ホステッドコントロールプレーン用の SR-IOV Operator のデプロイメント を参照してください。

1.3.10.34. ベアメタルでの Ingress VIP および API VIP サービスの IPv6 仮想 IP (VIP)アドレスのサポート

今回の更新により、インストーラーによってプロビジョニングされたインフラストラクチャークラスターで、**install-config.yaml** ファイルの **ingressVIP** および **apiVIP** の設定が非推奨になりました。代わりに、**ingressVIPs** および **apiVIPs** の設定を使用してください。これらの設定は、Ingress VIP および API VIP サービスを使用してクラスターへの IPv4 および IPv6 アクセスを必要とするベアメタル上のアプリケーションのデュアルスタックネットワークをサポートします。**ingressVIPs** および **apiVIPs** の設定では、リスト形式を使用して、IPv4 アドレス、IPv6 アドレス、または両方の IP アドレス形式を指定します。リストの順序は、各サービスのプライマリーおよびセカンダリー VIP アドレスを示しています。デュアルスタックネットワークを使用する場合には、プライマリー IP アドレスは IPv4 ネットワークからのものである必要があります。

1.3.10.35. Bluefield-2 ネットワークデバイスをデータ処理ユニット (DPU) モードからネットワークインターフェイスコントローラー (NIC) モードへの切り替えをサポート (テクノロジープレビュー)

今回の更新により、BluefFeld-2 ネットワークデバイスをデータ処理ユニット (DPU) モードからネットワークインターフェイスコントローラー (NIC) モードに切り替えることができます。

詳細は 、DPU から NIC への Bluefield-2 の切り替え を参照してください。

1.3.11. ストレージ

1.3.11.1. GCP Filestore Driver Operator を使用した永続ストレージ (テクノロジープレビュー)

OpenShift Container Platform は、Google Compute Platform (GCP) Filestore の Container Storage Interface (CSI) ドライバーを使用して永続ボリューム (PV) をプロビジョニングできます。このドライバーを管理する GCP Filestore CSI Driver Operator は、テクノロジープレビュー機能です。

詳細は、GCP Filestore CSI Driver Operator を参照してください。

1.3.11.2. AWS Elastic Block Storage 自動移行における自動 CSI 移行の一般公開

OpenShift Container Platform 4.8 以降、インツリーボリュームプラグインの同等の Container Storage Interface (CSI) ドライバーへの自動移行がテクノロジープレビュー機能として利用可能になりました。Amazon Web Services (AWS) Elastic Block Storage (EBS) のサポートは、OpenShift Container Platform 4.8 のこの機能で提供され、OpenShift Container Platform 4.12 では AWS EBS の自動移行に対するサポートが一般提供されます。AWS EBS の CSI 移行はデフォルトで有効化され、管理者によるアクションは不要になりました。

この機能は in-tree オブジェクトを自動的に対応する CSI 表現に変換するため、ユーザーに対して完全に透過的である必要があります。変換されたオブジェクトはディスクに保存され、ユーザーデータは移行されません。

in-tree ストレージプラグインを参照するストレージクラスは引き続き機能しますが、デフォルトのストレージクラスを CSI ストレージクラスに切り替えることが推奨されます。

詳細は、CSI 自動移行 を参照してください。

1.3.11.3. GCP PD 自動移行における自動 CSI 移行の一般提供を開始

OpenShift Container Platform 4.8 以降、インツリーボリュームプラグインの同等の Container Storage Interface (CSI) ドライバーへの自動移行がテクノロジープレビュー機能として利用可能になりました。Google Compute Engine Persistent Disk (GCP PD) のサポートは、OpenShift Container Platform 4.9 のこの機能で提供され、OpenShift Container Platform 4.12 では GCP PD の自動移行に対するサポートの一般提供が開始されます。GCP PD の CSI 移行はデフォルトで有効化され、管理者によるアクションは不要になりました。

この機能は in-tree オブジェクトを自動的に対応する CSI 表現に変換するため、ユーザーに対して完全に透過的である必要があります。変換されたオブジェクトはディスクに保存され、ユーザーデータは移行されません。

in-tree ストレージプラグインを参照するストレージクラスは引き続き機能しますが、デフォルトのストレージクラスを CSI ストレージクラスに切り替えることが推奨されます。

詳細は、CSI 自動移行 を参照してください。

1.3.11.4. Pod スケジューリング用ストレージ容量の追跡が一般提供を開始

この新機能は、**CSIStorageCapacity** オブジェクトを使用して現在使用可能なストレージ容量を公開し、遅延バインディングで Container Storage Interface (CSI) ボリュームを使用する Pod のスケジューリングを強化します。現在、この機能をサポートする唯一の OpenShift Container Platform ストレージタイプは OpenShift Data Foundation です。

1.3.11.5. VMware vSphere CSI トポロジーの一般提供を開始

OpenShift Container Platform は、異なるゾーンおよびリージョンに OpenShift Container Platform for vSphere をデプロイする機能を提供します。この機能を使用することで、複数のコンピュートクラスターにデプロイできるため、単一障害点を回避するのに役立ちます。

詳細は、vSphere CSI トポロジー を参照してください。

1.3.11.6. ローカル一時ストレージリソース管理の一般提供を開始

ローカル一時ストレージリソース管理機能の一般提供が開始されました。この機能を使用すると、リクエストと制限を指定して、ローカル一時ストレージを管理できます。

詳細は、一時ストレージの管理 を参照してください。

1.3.11.7. ボリュームポピュレーター (テクノロジープレビュー)

ボリュームポピュレーターは **datasource** を使用して、事前に入力されたボリュームの作成を有効にします。

ボリュームの作成は現在有効になっており、テクノロジープレビュー機能としてサポートされています。ただし、OpenShift Container Platform にはボリュームポピュレーターは同梱されていません。

詳細は、ボリュームポピュレーター を参照してください。

1.3.11.8. VMware vSphere CSI Driver Operator の要件

OpenShift Container Platform 4.12 の場合、VMWare vSphere Container Storage Interface (CSI) Driver Operator には、以下の最小限のコンポーネントがインストールされている必要があります。

- VMware vSphere バージョン 7.0 Update 2 以降、バージョン 7 まで。vSphere 8 はサポートされていません。
- vCenter 7.0 Update 2 以降、バージョン 8 まで。vCenter 7 はサポートされていません。
- ハードウェアバージョン 15 以降の仮想マシン
- サードパーティーの CSI ドライバーがクラスターにインストールされていない

サードパーティーの CSI ドライバーがクラスターに存在する場合、OpenShift Container Platform はそれを上書きしません。サードパーティーの CSI ドライバーが存在すると、OpenShift Container Platform を OpenShift Container Platform 4.13 以降にアップグレードできなくなります。

詳細は、VMware vSphere CSI Driver Operator の要件 を参照してください。

1.3.12. Operator ライフサイクル

1.3.12.1. Platform Operator (テクノロジープレビュー)

OpenShift Container Platform 4.12 から、Operator Lifecycle Manager (OLM) は **プラットフォームの Operator** タイプをテクノロジープレビュー機能として導入します。プラットフォーム Operator メカニズムは、同じく OpenShift Container Platform 4.12 で導入された RukPak コンポーネントのリソースに依存して、コンテンツの調達と管理を行います。

プラットフォーム Operator は OLM ベースの Operator であり、OpenShift Container Platform クラスターの Day 0 操作中または操作後にインストールでき、クラスターのライフサイクルに参加します。クラスター管理者は、プラットフォーム Operator を使用して OpenShift Container Platform インストー

ルをさらにカスタマイズし、要件とユースケースを満たすことができます。

プラットフォーム Operator について、詳しくは プラットフォーム Operator の管理 を参照してください。RukPak とそのリソースについて、詳しくは Operator Framework のパッケージ化形式 を参照してください。

1.3.12.2. Operator のインストール場所の制御

デフォルトでは、Operator をインストールすると、OpenShift Container Platform は Operator Pod をワーカーノードの 1 つにランダムにインストールします。

OpenShift Container Platform 4.12 では、Operator の **Subscription** オブジェクトにアフィニティー制約を追加することで、Operator Pod がインストールされる場所を制御できます。

詳細は、Operator のインストール場所の制御 を参照してください。

1.3.12.3. ユーザーが作成した openshift-* namespace の Pod セキュリティーアドミッションの同期

OpenShift Container Platform 4.12 では、ユーザーが作成した **openshift-** プレフィックスを持つ namespace に Operator がインストールされている場合、Pod セキュリティーアドミッションの同期がデフォルトで有効になります。クラスターサービスバージョン (CSV) が namespace に作成されると、同期が有効になります。同期されたラベルは、namespace のサービスアカウントのパーミッションを継承します。

詳細は、Pod セキュリティー標準とのセキュリティーコンテキスト制約の同期 を参照してください。

1.3.13. Operator の開発

1.3.13.1. カタログ Pod のセキュリティーコンテキストの設定

run bundle および **bundle-upgrade** サブコマンドで **--security-context-config** フラグを使用して、カタログ Pod のセキュリティーコンテキストを設定できます。このフラグにより、seccomp プロファイルが Pod のセキュリティーアドミッションに準拠できるようになります。このフラグは、**restricted** および **legacy** の値を許可します。値を指定しない場合、seccomp プロファイルはデフォルトで **restricted** になります。制限付きパーミッションではカタログ Pod を実行できない場合、次の例に示すとおり、フラグを **legacy** に設定します。

```
$ operator-sdk run bundle \
  --security-context-config=legacy
```

1.3.13.2. Kubernetes 1.25 から削除された API のバンドルマニフェストの検証

bundle validate サブコマンドで Operator Framework スイートのテストを使用することにより、Kubernetes 1.25 から削除された非推奨 API のバンドルマニフェストを確認できるようになりました。

以下に例を示します。

```
$ operator-sdk bundle validate .<bundle_dir_or_image> \
  --select-optional suite=operatorframework \
  --optional-values=k8s-version=1.25
```

Operator が Kubernetes 1.25 から削除された API のいずれかを使用するパーミッションを要求した場合に、コマンドは警告メッセージを表示します。

Kubernetes 1.25 から削除された API バージョンのいずれかが Operator のクラスターサービスバージョン (CSV) に含まれている場合に、コマンドはエラーメッセージを表示します。

詳細は Kubernetes 1.25 から削除されたベータ API および Operator SDK CLI リファレンス を参照してください。

1.3.14. マシン API

1.3.14.1. コントロールプレーンマシンセット

OpenShift Container Platform 4.12 では、コントロールプレーンのマシンセットが導入されています。コントロールプレーンマシンセット は、計算マシンセットが計算マシンに提供するものと同様の管理機能をコントロールプレーンマシンに提供します。詳細については、コントロールプレーンマシンの管理を参照してください。

1.3.14.2. クラスターオートスケーラーのログレベルの詳細度の指定

OpenShift Container Platform は、**ClusterAutoscaler** カスタムリソースで **logVerbosity** パラメーターを設定することにより、クラスターオートスケーラーのログレベルの詳細度の設定をサポートするようになりました。詳細は、**ClusterAutoscaler** リソース定義 を参照してください。

1.3.14.3. Azure ブート診断の有効化

OpenShift Container Platform は、マシンセットが作成する Azure マシンでのブート診断の有効化をサポートするようになりました。詳細については、コンピューティングマシン または コントロールプレーンマシン の Azure ブート診断の有効化を参照してください。

1.3.15. Machine Config Operator

1.3.15.1. RHCOS イメージのレイヤー化

Red Hat Enterprise Linux CoreOS (RHCOS) イメージの階層化により、ベース RHCOS イメージの上に新しいイメージを追加できます。このレイヤー化は、RHCOS の基本イメージを変更しません。代わりに、すべての RHCOS 機能を含む **カスタムレイヤーイメージ** を作成し、クラスター内の特定のノードに追加機能を追加します。

現在、RHCOS イメージの階層化により、Customer Experience and Engagement (CEE) と連携して、Red Hat Hotfix ポリシー に基づいて、RHCOS イメージの上に Hotfix パッケージを取得して適用することができます。将来のリリースでは、RHCOS イメージのレイヤー化を使用して、libreswan や numactl などのサードパーティーソフトウェアパッケージを組み込むことができるようになる予定です。

詳細は、RHCOS イメージのレイヤー化 を参照してください。

1.3.16. ノード

1.3.16.1. インターフェイス固有のセーフリストの更新 (テクノロジープレビュー)

OpenShift Container Platform は、デフォルトのインターフェイス固有の安全な **sysctl** の更新をサポートするようになりました。

事前定義されたリストから **sysctl** を追加または削除できます。**sysctls** を追加すると、すべてのノードにわたって設定できます。インターフェイス固有の安全な **sysctl** リストの更新は、テクノロジープレビュー機能のみです。

詳しく は、インターフェイス固有の安全な sysctls リストの更新 を参照してください。

1.3.16.2. Cron ジョブのタイムゾーン (テクノロジープレビュー)

cron ジョブスケジュールのタイムゾーンの設定が、テクノロジープレビュー として提供されるようになりました。タイムゾーンが指定されていない場合、Kubernetes コントローラーマネージャーは、ローカルタイムゾーンを基準にしてスケジュールを解釈します。

詳細は、cron ジョブの作成 を参照してください。

1.3.16.3. Linux Control Group バージョン 2 がテクノロジープレビューに昇格

Linux Control Group バージョン 2 (cgroup v2) の OpenShift Container Platform サポートは、テクノロジープレビューに昇格しました。cgroup v2 は、カーネル コントロールグループ の次のバージョンです。cgroups v2 は、統一された階層、より安全なサブツリー委任、Pressure Stall Information などの新機能、拡張されたリソース管理と分離など、複数の改善を提供します。詳細は、Enabling Linux Control Group version 2 (cgroup v2) を参照してください。

1.3.16.4. crun コンテナーランタイム (テクノロジープレビュー)

OpenShift Container Platform は、テクノロジープレビューで crun コンテナーランタイムをサポートするようになりました。必要に応じて、**ContainerRuntimeConfig** カスタムリソース (CR) を使用して、crun コンテナーランタイムと既定のコンテナーランタイムを切り替えることができます。詳細 については、コンテナーエンジンとコンテナーランタイムについて を参照してください。

1.3.16.5. Self Node Remediation Operator の機能拡張

OpenShift Container Platform は、Self Node Remediation Operator によるコントロールプレーンフェンシングをサポートするようになりました。ノードに障害が発生した場合は、ワーカーノードとコントロールプレーンノードの両方で修復ストラテジーに従うことができます。詳細は、コントロールプレーンフェンシング を参照してください。

1.3.16.6. Node Health Check Operator の拡張機能

OpenShift Container Platform は、Node Health Check Operator でのコントロールプレーンフェンシングをサポートするようになりました。ノードに障害が発生した場合は、ワーカーノードとコントロールプレーンノードの両方で修復ストラテジーに従うことができます。詳細は、コントロールプレーンフェンシング を参照してください。

Node Health Check Operator には、Node Health Checks を管理するための Web コンソールプラグインも含まれるようになりました。詳細は、ノードヘルスチェックの作成 を参照してください。

Node Health Check Operator の最新バージョンのインストールまたは更新には、**stable** サブスクリプションチャネルを使用します。詳細は、CLI を使用した Node Health Check Operator のインストール を参照してください。

1.3.17. モニタリング

本リリースのモニターリングスタックには、以下の新機能および変更された機能が含まれています。

1.3.17.1. モニターリングスタックコンポーネントおよび依存関係の更新

このリリースには、スタックコンポーネントと依存関係を監視するための次のバージョン更新が含まれています。

- kube-state-metrics to 2.6.0
- node-exporter to 1.4.0
- prom-label-proxy to 0.5.0
- Prometheus to 2.39.1
- prometheus-adapter to 0.10.0
- prometheus-operator to 0.60.1
- Thanos to 0.28.1

1.3.17.2. アラートルールの変更

注記

Red Hat は、記録ルールまたはアラートルールの後方互換性を保証しません。

- **New**
  - **TelemeterClientFailures** アラートが追加されました。これは、クラスターが一定期間にわたって特定のレートで Telemetry データを送信しようとして失敗した場合にトリガーされます。アラートは、15 分の時間枠内で失敗したリクエストの割合が合計の 20% に達すると発生します。
- **変更済み**
  - **KubeAggregatedAPIDown** アラートは、300 秒ではなく 900 秒待機してから通知を送信するようになりました。
  - **NodeClockNotSynchronising** および **NodeClockSkewDetected** アラートは、**node-exporter** ジョブからのメトリクスのみ評価するようになりました。
  - **NodeRAIDDegraded** および **NodeRAIDDiskFailure** アラートには、**mmcblk.p.|nvme.|sd.|vd.|xvd.|dm-.|dasd.+** によって返される値のみに一致するデバイスラベルフィルターが含まれるようになりました。
  - **PrometheusHighQueryLoad** および **ThanosQueryOverload** アラートは、クエリーレイヤーに高いクエリーロードが存在する場合にもトリガーされるようになりました。

1.3.17.3. コンポーネントを監視するための Pod トポロジー分散制約を指定する新しいオプション

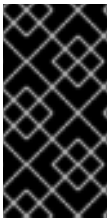
Pod トポロジー分散制約を使用して、OpenShift Container Platform Pod が複数のアベイラビリティーゾーンにデプロイされている場合に、Prometheus、Thanos Ruler、および Alertmanager Pod がネットワークトポロジー全体にどのように分散されるかを制御できるようになりました。

1.3.17.4. Prometheus Adapter のデータ整合性を向上させる新しいオプション

Prometheus Adapter (PA) でオプションの kubelet サービスモニターを設定できるようになりました。これにより、複数の自動スケーリングリクエスト間でデータの一貫性が向上します。このサービスモニターを有効にすると、PA によって実行される基礎となる PromQL クエリーが異なる Prometheus サーバー上にあることで、PA に同時に送信される 2 つのクエリーが異なる結果をもたらす可能性がなくなります。

1.3.17.5. 秘密鍵を追加するための Alertmanager 設定の更新

このリリースでは、追加のキーを保持するために Alertmanager のシークレットを設定する場合、Alertmanager 設定がこれらのキーをファイル（テンプレート、TLS 証明書、またはトークンなど）として参照する場合は、設定は相対パスではなく絶対パスを使用してこれらのキーをポイントする必要があります。これらのキーは、**/etc/alertmanager/config** ディレクトリーの下にあります。OpenShift Container Platform の以前のリリースでは、Alertmanager 設定ファイルがキーと同じディレクトリーに配置されていたため、設定で相対パスを使用してこれらのキーを指すことができました。



重要

OpenShift Container Platform 4.12 にアップグレードし、ファイルとして参照される追加の Alertmanager 秘密鍵の相対パスを指定している場合、Alertmanager 設定でこれらの相対パスを絶対パスに変更する必要があります。そうしないと、ファイルを使用するアラート受信者が通知を配信できなくなります。

1.3.18. スケーラビリティーおよびパフォーマンス

1.3.18.1. ワークロードヒントを使用してリアルタイムを無効にすると、クラスターから受信パケットステアリングが削除されます

クラスターレベルでは、デフォルトで systemd サービスが仮想ネットワークインターフェイスの受信パケットステアリング (RPS) マスクを設定します。RPS マスクは、パフォーマンスプロファイルで定義された予約済み CPU のリストに従って、仮想ネットワークインターフェイスからの割り込み要求をルーティングします。コンテナーレベルでは、**CRI-O** フックスクリプトもすべての仮想ネットワークデバイスの RPS マスクを設定します。

今回の更新により、パフォーマンスプロファイルで **spec.workloadHints.realTime** を **False** に設定すると、システムは systemd サービスと、RPS マスクを設定する **CRI-O** フックスクリプトの両方も無効にします。RPS は通常、低レイテンシーのリアルタイムワークロードのみを必要とするユースケースに関連するため、システムはこれらの RPS 機能を無効にします。

spec.workloadHints.realTime を **False** に設定した場合でも RPS 機能を維持するには、Red Hat ナレッジベースソリューション Performance addons operator advanced configuration の **RPS 設定** セクションを参照してください。

ワークロードヒントの設定の詳細については、ワークロードヒントについて についてを参照してください。

1.3.18.2. Tuned プロファイル

tuned プロファイルは、デフォルトで **fs.aio-max-nr sysctl** 値を定義するようになり、デフォルトノードプロファイルの非同期 I/O パフォーマンスが向上しました。

1.3.18.3. 新しいカーネル機能とオプションのサポート

低遅延チューニングが更新され、最新のカーネル機能とオプションを使用できるようになりました。2117780 の修正により、CPU ごとの新しい **kthread** である **ktimers** が導入されました。このス

レッドは、適切な CPU コアに固定する必要があります。今回の更新では機能上の変更はなく、ワークロードの分離は同じままです。詳細は、2102450 を参照してください。

1.3.18.4. 省電力設定

OpenShift Container Platform 4.12 では、C ステートと OS 制御の P ステートを有効にすることで、重要なワークロードとそうでないワークロードに対して異なる省電力設定を使用できます。新しい **perPodPowerManagement** ワークロードヒント、**cpu-c-states.crio.io** および **cpu-freq-governor.crio.io** CRI-O アノテーションを使用して設定を適用できます。この機能の詳細は、省電力設定 を参照してください。

1.3.18.5. GitOps ZTP (テクノロジープレビュー) を使用したワーカーノードによる単一ノード OpenShift クラスターの拡張

OpenShift Container Platform 4.11 では、ワーカーノードを単一ノードの OpenShift クラスターに手動で追加できる機能が導入されました。この機能は、GitOps ZTP でも利用できるようになりました。

詳しくは、GitOps ZTP を使用した単一ノード OpenShift クラスターへのワーカーノードの追加 を参照してください。

1.3.18.6. OpenShift Container Platform および Operator のデプロイメント時間を短縮する factory-precaching-cli ツール (テクノロジープレビュー)

OpenShift Container Platform 4.12 では、factory-precaching-cli ツールを使用して OpenShift Container Platform および Operator イメージを事前に工場でサーバーにキャッシュし、その事前キャッシュされたサーバーをデプロイメント用のサイトに含めることができます。factory-precaching-cli ツールの詳細については、単一ノード OpenShift デプロイメントのイメージの事前キャッシュ を参照してください。

1.3.18.7. factory-precaching-cli ツールのゼロタッチプロビジョニング (ZTP) インテグレーション (テクノロジープレビュー)

OpenShift Container Platform 4.12 では、GitOps ZTP ワークフローで factory-precaching-cli ツールを使用できます。詳細は、単一ノード OpenShift デプロイメントのイメージの事前キャッシュ を参照してください。

1.3.18.8. ホステッドクラスターでのノードチューニング (テクノロジープレビュー)

Node Tuning Operator を使用して、ホステッドクラスター内のノードの OS レベルチューニングを設定できるようになりました。ノードチューニングを設定するには、**Tuned** オブジェクトを含む管理クラスターで設定マップを作成し、ノードプールで作成した設定マップを参照します。**Tuned** オブジェクトで定義されたチューニング設定は、ノードプール内のノードに適用されます。詳細は、ホステッドクラスターでのノードチューニングの設定 を参照してください。

1.3.18.9. カーネルモジュール管理 Operator

カーネルモジュール管理 (KMM) Operator は、Special Resource Operator (SRO) を置き換えます。KMM には、接続環境専用の次の機能が含まれています。

- エッジデプロイメント用のハブおよびスポークのサポート
- アップグレードサポートのプリフライトチェック
- セキュアブートカーネルモジュールの署名

- トラブルシューティングを支援するためのログの収集が必要
- バイナリーファームウェアのデプロイメント

1.3.18.10. ハブおよびスポーククラスターのサポート (テクノロジープレビュー)

インターネットにアクセスできる環境でハブおよびスポークをデプロイする場合、ハブクラスターにデプロイされたカーネルモジュール管理 (KMM) Operator を使用して、必要なカーネルモジュールの 1 つ以上のマネージドクラスターへのデプロイを管理できます。

1.3.18.11. Topology Aware Lifecycle Manager (TALM)

Topology Aware Lifecycle Manager (TALM) では、より詳細なステータス情報とメッセージ、および再設計された条件が提供されるようになりました。**ClusterLabelSelector** フィールドを使用すると、更新するクラスターをより柔軟に選択できます。タイムアウト設定を使用して、クラスターの更新が失敗した場合にどうするかを決定できます。たとえば、失敗したクラスターをスキップして他のクラスターのアップグレードを続行するか、すべてのクラスターのポリシー修正を停止できます。詳細は、クラスター更新のための Topology Aware Lifecycle Manager を参照してください。

1.3.18.12. マウント namespace のカプセル化 (テクノロジープレビュー)

カプセル化は、すべての Kubernetes 固有のマウントポイントを別の namespace に移動して、デフォルトの namespace にある多数のマウントポイントの可視性とパフォーマンスへの影響を軽減するプロセスです。以前は、特に GitOps ZTP を使用してインストールされた分散ユニット (DU) 用に、マウント namespace のカプセル化が透過的に OpenShift Container Platform にデプロイされていました。OpenShift Container Platform v4.12 では、この機能が設定可能なオプションとして利用できるようになりました。

標準のホストオペレーティングシステムは、systemd を使用してすべてのマウント namespace (標準の Linux マウントと、Kubernetes が操作に使用する多数のマウントの両方) を常にスキャンします。kubelet と CRI-O の現在の実装はどちらも、すべてのコンテナーと Kubelet マウントポイントに最上位の namespace を使用します。これらのコンテナー固有のマウントポイントをプライベート namespace にカプセル化すると、systemd のオーバーヘッドが削減され、CPU パフォーマンスが向上します。カプセル化のセキュリティーは、Kubernetes 固有のマウントポイントを特権のないユーザーによる検査から安全な場所に保存することでも向上します。

詳細は、マウント namespace のカプセル化による CPU 使用率の最適化 を参照してください。

1.3.18.13. GitOps ZTP を使用してデプロイされた単一ノード OpenShift クラスター内におけるワークロードのパーティション設定 CPU セットの変更

GitOps ZTP を使用してデプロイする単一ノードの OpenShift クラスターで、ワークロードのパーティション設定 CPU セットを設定できます。これを設定するには、**SiteConfig** カスタムリソース (CR) の **cpuset** フィールドとグループ **PolicyGenTemplate** CR の **reserved** フィールドを使用してクラスター管理 CPU リソースを指定します。**cpuset** に設定する値は、ワークロードのパーティション設定のためにクラスターの **PerformanceProfile** CR **.spec.cpu.reserved** フィールドに設定された値と一致する必要があります。

詳細は、ワークロードのパーティション設定 を参照してください。

1.3.18.14. GitOps ZTP で RHACM ハブテンプレート関数が使用可能に

Red Hat Advanced Cluster Management (RHACM) および Topology Aware Lifecycle Manager (TALM) を使用して、GitOps ZTP でハブテンプレート関数を使用できるようになりました。ハブ側のクラスターテンプレートを使用すると、設定は似ているが値が異なる多くのクラスターに対して個別のポリ

シーを作成する必要がなくなります。詳細は、PolicyGenTemplate CR でのハブテンプレートの使用 を参照してください。

1.3.18.15. ArgoCD マネージドクラスターの制限

RHACM は **SiteConfig** CR を使用して、ArgoCD の Day 1 マネージドクラスターインストール CR を生成します。各 ArgoCD アプリケーションは、最大 300 個の **SiteConfig** CR を管理できます。詳細は、ArgoCD を使用したハブクラスターの設定 を参照してください。

1.3.18.16. PolicyGenTemplate CR におけるポリシーコンプライアンス評価タイムアウトの設定に対する GitOps ZTP サポート

GitOps ZTP v4.11+ では、デフォルトのポリシーコンプライアンス評価タイムアウト値を **PolicyGenTemplate** カスタムリソース (CR) で使用できます。この値は、RHACM が適用されたクラスターポリシーを再評価する前に、関連する **ConfigurationPolicy** CR がポリシー準拠または非準拠の状態を維持できる期間を指定します。

オプションで、**PolicyGenTemplate** CR の全ポリシーのデフォルト評価間隔をオーバーライドできるようになりました。

詳細は、PolicyGenTemplate CR のポリシーコンプライアンス評価タイムアウトの設定 を参照してください。

1.3.18.17. マネージドクラスターのプラットフォームタイプの指定

Assisted Installer は現在、以下の OpenShift Container Platform プラットフォームをサポートしています。

- **BareMetal**
- **vSphere**
- **なし**

シングルノード OpenShift は、**VSphere** をサポートしていません。

1.3.18.18. 非認証レジストリーを使用するためのハブクラスターの設定

このリリースでは、ハブクラスターを設定する際に、非認証レジストリーの使用がサポートされています。認証を必要としないレジストリーは、**AgentServiceConfig** リソースの **spec.unauthenticatedRegistries** の下に一覧表示されます。このリストにあるレジストリーのエントリーは、スポーククラスターのインストールに使用されるプルシークレットに含める必要はありません。**assisted-service** は、インストールに使用されるすべてのイメージレジストリーの認証情報がプルシークレットに含まれていることを確認して、プルシークレットを検証します。

詳細は、非認証レジストリーを使用するためのハブクラスターの設定 を参照してください。

1.3.18.19. 切断された GitOps ZTP インストールでの Ironic エージェントミラーリング

GitOps ZTP を使用して、切断されたインストールを行うには、コンバージドフローが有効になっているスポーククラスターに OpenShift Container Platform バージョン 4.11 以前をデプロイする場合、デフォルトの Ironic エージェントイメージをローカルイメージリポジトリーにミラーリングする必要があります。デフォルトの Ironic エージェントイメージは次のとおりです。

- AMD64 Ironic エージェントイメージ: **quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:d3f1d4d3cd5fbcf1b9249dd71d01be4b901d337fdc5f8f66569eb71df4d9d446**
- AArch64 Ironic エージェントイメージ: **quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:cb0edf19fffc17f542a7efae76939b1e9757dc75782d4727fb0aa77ed5809b43**

イメージのミラーリングについて詳しくは、OpenShift Container Platform イメージリポジトリーのミラーリング を参照してください。

1.3.18.20. GitOps ZTP を使用して、Discovery ISO のカーネル引数を設定する

OpenShift Container Platform は、GitOps ZTP デプロイメントで Discovery ISO のカーネル引数の指定をサポートするようになりました。手動と自動の両方の GitOps ZTP デプロイメントで、Discovery ISO は、管理対象のベアメタルホストでの OpenShift Container Platform インストールプロセスの一部です。**InfraEnv** リソースを編集して、Discovery ISO のカーネル引数を指定できるようになりました。これは、特定の環境要件を持つクラスターのインストールに役立ちます。たとえば、**rd.net.timeout.carrier** カーネル引数を定義して、静的ネットワーク用にクラスターを設定するために役立てることができます。

カーネル引数を指定する方法の詳細については、GitOps ZTP を使用して、検出 ISO のカーネル引数を設定する および GitOps ZTP を使用して、手動インストール用に検出 ISO のカーネル引数を設定する を参照してください。

1.3.18.21. ハブクラスターから異種スポーククラスターをデプロイする

今回の更新により、AMD64 と AArch64 の両方の CPU アーキテクチャーを備えたホストを特徴とする OpenShift Container Platform 混合アーキテクチャークラスター (異種クラスターとも呼ばれる) を作成できるようになりました。Red Hat Advanced Cluster Management (RHACM) によって管理されるハブクラスターから異種スポーククラスターをデプロイできます。異種スポーククラスターを作成するには、デプロイされた AMD64 クラスターに AArch64 ワーカーノードを追加します。

デプロイされた AMD64 クラスターに AArch64 ワーカーノードを追加するには、**InfraEnv** カスタムリソース (CR) を使用して、ノードに必要な AArch64 アーキテクチャー、マルチアーキテクチャーリリースイメージ、およびオペレーティングシステムを指定できます。次に、Assisted Installer API と **InfraEnv** CR を使用して、AArch64 ワーカーノードを AMD64 クラスターにプロビジョニングできます。

1.3.19. Insights Operator

1.3.19.1. Insights アラート

OpenShift Container Platform 4.12 では、アクティブな Insights の推奨事項がアラートとしてユーザーに提示されるようになりました。これらのアラートは、Alertmanager で表示および設定できます。

1.3.19.2. Insights Operator のデータ収集機能の拡張

OpenShift Container Platform 4.12 では、Insights Operator が以下のメトリクスを収集するようになりました。

- **console_helm_uninstalls_total**
- **console_helm_upgrades_total**

1.3.20. 認証および認可

1.3.20.1. RHOSP のアプリケーション認証情報

Red Hat OpenStack Platform (RHOSP) で実行されるクラスターの **clouds.yaml** ファイルで アプリケーション認証情報 を指定できるようになりました。アプリケーション認証情報は、設定ファイルにユーザーアカウントの詳細を埋め込む代わりの方法です。例として、ユーザーアカウントの詳細を含む **clouds.yaml** ファイルの次のセクションを参照してください。

```
clouds:
  openstack:
    auth:
      auth_url: https://127.0.0.1:13000
      password: thepassword
      project_domain_name: Default
      project_name: theprojectname
      user_domain_name: Default
      username: theusername
    region_name: regionOne
```

そのセクションを、アプリケーション認証情報を使用するセクションと比較します。

```
clouds:
  openstack:
    auth:
      auth_url: https://127.0.0.1:13000
      application_credential_id: '5dc185489adc4b0f854532e1af81ffe0'
      application_credential_secret:
'PDCTKans2bPBbaEqBLiT_lajG8e5J_nJB4kvQHjaAy6ufhod0Zl0NkNoBzjn_bWSYzk587ieIGSlT11c4pV
ehA'
    auth_type: "v3applicationcredential"
    region_name: regionOne
```

RHOSP 管理者としてクラスターでアプリケーション認証情報を使用するには、認証情報を作成します。次に、クラスターをインストールするときに、**clouds.yaml** ファイルでそれらを使用します。または、**clouds.yaml** ファイルを作成し、それを既存のクラスターにローテーションすることもできます。

1.3.21. ホストされているコントロールプレーン (テクノロジープレビュー)

1.3.21.1. HyperShift API ベータリリースが利用可能に

OpenShift Container Platform 上でホストされるコントロールプレーンの API である **hypershift.openshift.io** API のデフォルトバージョンが v1beta1 になりました。既存クラスターの場合、現時点でアルファ版からベータ版への移行はサポートされていません。

1.3.21.2. ホストされたコントロールプレーンのバージョン管理

OpenShift Container Platform のメジャー、マイナー、またはパッチバージョンのリリースごとに、HyperShift Operator がリリースされます。HyperShift コマンドラインインターフェイス (CLI) は、各 HyperShift Operator リリースの一部としてリリースされます。

HostedCluster および **NodePool** API リソースは API のベータ版で利用でき、 OpenShift Container Platform および Kubernetes と同様のポリシーに従います。

1.3.21.3. ホストされたクラスターでの etcd のバックアップと復元

OpenShift Container Platform でホストされたコントロールプレーンを使用する場合、etcd のスナップショットを取得し、S3 バケットなどの後で取得できる場所にアップロードすることで、etcd をバックアップおよび復元できます。必要に応じて、後でスナップショットを復元できます。詳細は、ホストされたクラスターでの etcd のバックアップと復元 を参照してください。

1.3.21.4. AWS リージョン内のホストされたクラスターの障害復旧

ホストされたクラスターの障害復旧が必要な状況では、ホストされたクラスターを AWS 内の同じリージョンに回復できます。詳細は、AWS リージョン内のホストされたクラスターの障害復旧 を参照してください。

1.3.22. Red Hat Virtualization (RHV)

今回のリリースでは、Red Hat Virtualization (RHV) の更新がいくつかあります。このリリースには以下が含まれます。

- oVirt CSI ドライバーのロギングではエラーメッセージが新しく改訂され、ログの明確さと読みやすさが向上しました。
- クラスター API プロバイダーは、OpenShift Container Platform で認証情報が変更されると、oVirt および Red Hat Virtualization (RHV) の認証情報を自動的に更新します。

1.4. 主な技術上の変更点

OpenShift Container Platform 4.12 では、主に以下のような技術的な変更点が加えられています。

AWS Security Token Service のリージョンエンドポイント
Cloud Credential Operator ユーティリティー (**ccoctl**) は、AWS Security Token Service (AWS STS) のリージョンエンドポイントを使用するシークレットを作成するようになりました。このアプローチは、AWS の推奨のベストプラクティスに準拠しています。

Cloud Credential Operator ユーティリティーを使用して GCP リソースを削除する認証情報要求ディレクトリーパラメーター
今回のリリースでは Cloud Credential Operator ユーティリティーを使用して GCP リソースを削除するときに、コンポーネントの **CredentialsRequest** オブジェクトのファイルを含むディレクトリーを指定する必要があります。

Pod セキュリティーアドミッションの今後の限定的な適用
現在、Pod のセキュリティー違反は警告として表示され、監査ログに記録されますが、Pod が拒否されることはありません。

現在、OpenShift Container Platform の次のマイナーリリースでは、Pod のセキュリティーアドミッションに対するグローバルな制限付きの適用が計画されています。この制限付きの適用が有効になっている場合、Pod セキュリティー違反のある Pod は拒否されます。

この今後の変更に備えて、ワークロードが適用される Pod セキュリティーアドミッションプロファイルと一致していることを確認してください。グローバルまたはネームスペースレベルで定義された強制セキュリティー基準に従って設定されていないワークロードは拒否されます。**restricted-v2** SCC は、制限付き Kubernetes の定義に従ってワークロードを許可します。

Pod のセキュリティー違反が発生している場合は、次のリソースを参照してください。

- Pod のセキュリティー違反の原因となっているワークロードを見つける方法については、Pod のセキュリティー違反の特定 を参照してください。

- Pod セキュリティーアドミッションラベルの同期がいつ実行されるかを理解するには、Pod セキュリティー標準とのセキュリティーコンテキスト制約の同期 を参照してください。Pod セキュリティーアドミッションラベルは、次のような特定の状況では同期されません。
  - ワークロードは、**openshift-** で始まるシステム作成の namespace で実行されています。
  - ワークロードは、Pod コントローラーなしで直接作成された Pod で実行されています。
- 必要に応じて、**pod-security.kubernetes.io/enforce** ラベルを設定して、namespace または Pod にカスタムアドミッションプロファイルを設定できます。

カタログソースと制限付き Pod セキュリティーアドミッションの適用

SQLite ベースのカタログ形式と、OpenShift Container Platform 4.11 より前にリリースされたバージョンの **opm** CLI ツールを使用してビルドされたカタログソースは、制限付き Pod セキュリティーの適用下では実行できません。

OpenShift Container Platform 4.12 では、デフォルトで namespace には制限付き Pod セキュリティーが適用されず、カタログソースのデフォルトセキュリティーモードは **legacy** に設定されています。

制限付き Pod セキュリティー適用下で SQLite ベースのカタログソース Pod を実行したくない場合は、OpenShift Container Platform 4.12 でカタログソースを更新する必要はありません。ただし、今後の OpenShift Container Platform リリースでカタログソースが確実に実行されるようにするには、カタログソースを更新して、制限付き Pod セキュリティーの適用下で実行する必要があります。

カタログの作成者は、次のいずれかのアクションを実行することで、制限付き Pod セキュリティー適用との互換性を有効にできます。

- カタログをファイルベースのカタログ形式に移行します。
- OpenShift Container Platform 4.11 以降でリリースされたバージョンの **opm** CLI ツールでカタログイメージを更新します。

SQLite データベースカタログイメージを更新したり、カタログをファイルベースのカタログ形式に移行したりしたくない場合は、昇格されたパーミッションで実行するようにカタログを設定できます。

詳細は、カタログソースと Pod セキュリティーアドミッション を参照してください。

Operator SDK 1.25.4

OpenShift Container Platform 4.12 は Operator SDK 1.25.4 をサポートします。この最新バージョンのインストール、または最新バージョンへの更新については、Operator SDK CLI のインストール を参照してください。

注記

Operator SDK 1.25.4 は Kubernetes 1.25 をサポートします。

詳細は Kubernetes 1.25 から削除されたベータ API および Kubernetes 1.25 から削除された API のバンドルマニフェストの検証 を参照してください。

以前に Operator SDK 1.22.0 で作成または管理されている Operator プロジェクトがある場合は、Operator SDK 1.25.4 との互換性を維持するためにプロジェクトを更新してください。

- Go ベースの Operator プロジェクトの更新
- Ansible ベースの Operator プロジェクトの更新
- Helm ベースの Operator プロジェクトの更新

- Hybrid Helm ベースの Operator プロジェクトの更新
- Java ベースの Operator プロジェクトの更新

LVM Operator の名称を Logical Volume Manager Storage に変更
これまで Red Hat OpenShift Data Foundation で提供されていた LVM Operator は、OpenShift Data Foundation を介してインストールする必要があります。OpenShift Container Platform v4.12 では、LVM Operator の名称が **Logical Volume Manager Storage** に変更されました。今後は、OpenShift Operator カタログからスタンドアロン Operator としてインストールします。Logical Volume Manager Storage は、単一の限られたリソースのシングルノード OpenShift クラスターで、ブロックストレージの動的なプロビジョニングを提供します。

1.5. 非推奨および削除された機能

以前のリリースで利用可能であった一部の機能が非推奨になるか、または削除されました。

非推奨の機能は依然として OpenShift Container Platform に含まれており、引き続きサポートされますが、本製品の今後のリリースで削除されるため、新規デプロイメントでの使用は推奨されません。OpenShift Container Platform 4.12 で非推奨となり、削除された主な機能の最新の一覧については、以下の表を参照してください。非推奨となり、削除された機能の詳細は、表の後に記載されています。

次の表では、機能は次のステータスでマークされています。

- **一般公開 (GA)**
- **非推奨**
- **廃止**

Operator の非推奨および削除された機能

表1.2 オペレーターは廃止され、トラッカーが削除されました

機能	4.10	4.11	4.12
Operator カタログの SQLite データベース形式	非推奨	非推奨	非推奨

イメージの非推奨および削除された機能

表1.3 イメージは廃止され、トラッカーが削除されました

機能	4.10	4.11	4.12
Cluster Samples Operator の **ImageChangesInProgress** 状態	非推奨	非推奨	非推奨
Cluster Samples Operator の **MigrationInProgress** 状態	非推奨	非推奨	非推奨
インストールペイロードからの Jenkins イメージの削除	一般公開 (GA)	廃止	廃止

非推奨および削除された機能の監視

表1.4 非推奨および削除されたトラッカーのモニタリング

機能	4.10	4.11	4.12
モニターリングスタックの Grafana コンポーネント	非推奨	廃止	廃止
モニターリングスタック内の Prometheus および Grafana UI へのアクセス	非推奨	廃止	廃止

インストールの非推奨および削除された機能

表1.5 インストールが非推奨になり、トラッカーが削除されました

機能	4.10	4.11	4.12
vSphere 6.7 Update 2 以前	非推奨	廃止	廃止
vSphere 7.0 Update 1 以前	一般公開 (GA)	非推奨	非推奨
VMware ESXi 6.7 Update 2 以前	非推奨	廃止	廃止
VMware ESXi 7.0 Update 1 以前	一般公開 (GA)	非推奨	非推奨
cluster.local ドメインの CoreDNS ワイルドカードクエリー	一般公開 (GA)	一般公開 (GA)	非推奨
インストーラーによりプロビジョニングされたインフラストラクチャークラスターにおける **install-config.yaml** ファイル内の **ingressVIP** および **apiVIP** 設定	一般公開 (GA)	一般公開 (GA)	非推奨

非推奨および削除されたクラスターの更新

表1.6 非推奨および削除されたトラッカーの更新

機能	4.10	4.11	4.12
仮想ハードウェアバージョン 13	非推奨	廃止	廃止

ストレージの非推奨および削除された機能

表1.7 Storage の廃止と削除されたトラッカー

機能	4.10	4.11	4.12
Snapshot.storage.k8s.io/v1beta1 API エンドポイント	非推奨	廃止	廃止

機能	4.10	4.11	4.12
FlexVolume を使用した永続ストレージ	非推奨	非推奨	非推奨

認証と承認の非推奨および削除された機能

表1.8 認証と承認は廃止され、トラッカーが削除されました

機能	4.10	4.11	4.12
サービスアカウントトークンシークレットの自動生成	一般公開 (GA)	廃止	廃止

特殊なハードウェアとドライバーの有効化 非推奨および削除された機能

表1.9 特殊なハードウェアとドライバーの有効化が非推奨になり、トラッカーが削除されました

機能	4.10	4.11	4.12
Special Resource Operator(SRO)	テクノロジープレビュー	テクノロジープレビュー	廃止

マルチアーキテクチャーの非推奨および削除された機能

表1.10 マルチアーキテクチャーの非推奨および削除されたトラッカー

機能	4.10	4.11	4.12
IBM POWER8 全モデル (**ppc64le**)	一般公開 (GA)	一般公開 (GA)	非推奨
IBM IBM POWER9 AC922 (**ppc64le**)	一般公開 (GA)	一般公開 (GA)	非推奨
IBM IBM POWER9 IC922 (**ppc64le**)	一般公開 (GA)	一般公開 (GA)	非推奨
IBM IBM POWER9 LC922 (**ppc64le**)	一般公開 (GA)	一般公開 (GA)	非推奨
IBM z13 全モデル (**s390x**)	一般公開 (GA)	一般公開 (GA)	非推奨
IBM LinuxONE Emperor (**s390x**)	一般公開 (GA)	一般公開 (GA)	非推奨

機能	4.10	4.11	4.12
IBM LinuxONE Rockhopper (**s390x**)	一般公開 (GA)	一般公開 (GA)	非推奨
AMD64 (x86_64) v1 CPU	一般公開 (GA)	一般公開 (GA)	非推奨

ネットワーキングの非推奨機能と削除された機能

表1.11 ネットワーキングの非推奨化と削除のトラッカー

機能	4.10	4.11	4.12
RHOSP 上の Kuryr	一般公開 (GA)	一般公開 (GA)	非推奨

1.5.1. 非推奨の機能

1.5.1.1. OpenShift Container Platform のホストプラットフォームとしての Red Hat Virtualization (RHV) が非推奨に

Red Hat Virtualization (RHV) は、OpenShift Container Platform の今後のリリースで非推奨になります。RHV での OpenShift Container Platform のサポートは、今後の OpenShift Container Platform リリースから削除される予定です (現時点では OpenShift Container Platform 4.14 に削除予定)。

1.5.1.2. cluster.local ドメインのワイルドカード DNS クエリーは非推奨です

CoreDNS は、**cluster.local** ドメインの下の名前に対するワイルドカード DNS クエリーのサポートを停止します。これらのクエリーは、以前のバージョンと同様に OpenShift Container Platform 4.12 で解決されますが、サポートは将来の OpenShift Container Platform リリースから削除される予定です。

1.5.1.3. ppc64le、s390x、および x86_64 v1 CPU アーキテクチャー上の特定のハードウェアモデルは非推奨です

OpenShift Container Platform 4.12 では、以下の RHCOS 機能のサポートが非推奨になりました。

- IBM POWER8 全モデル (ppc64le)
- IBM POWER9 AC922 (ppc64le)
- IBM POWER9 IC922 (ppc64le)
- IBM POWER9 LC922 (ppc64le)
- IBM z13 全モデル (s390x)
- LinuxONE Emperor (s390x)
- LinuxONE Rockhopper (s390x)

- AMD64 (x86_64) v1 CPU

これらのハードウェアモデルは OpenShift Container Platform 4.12 でも完全にサポートされていますが、Red Hat はそれ以降のハードウェアモデルを使用することを推奨しています。

1.5.1.4. RHOSP 上で実行されるクラスターの Kuryr サポート

OpenShift Container Platform 4.12 では、RHOSP 上で実行されるクラスターにおける Kuryr のサポートは非推奨となりました。サポートは、OpenShift Container Platform 4.14 以降に削除されます。

1.5.2. 削除された機能

1.5.2.1. Kubernetes 1.25 から削除されたベータ API

Kubernetes 1.25 では、次の非推奨 API が削除されたため、マニフェストと API クライアントを移行して、適切な API バージョンを使用する必要があります。削除された API の移行について、詳しくは Kubernetes documentation を参照してください。

表1.12 Kubernetes 1.25 から削除された API

リソース	削除された API	移行先	主な変更
CronJob	**batch/v1beta1**	**batch/v1**	いいえ
EndpointSlice	**discovery.k8s.io/v1beta1**	**discovery.k8s.io/v1**	はい
イベント	**events.k8s.io/v1beta1**	**events.k8s.io/v1**	はい
HorizontalPodAutoscaler	**autoscaling/v2beta1**	**autoscaling/v2**	いいえ
PodDisruptionBudget	**policy/v1beta1**	**policy/v1**	はい
PodSecurityPolicy	**policy/v1beta1**	Pod セキュリティーアドミッション [1]	はい
RuntimeClass	**node.k8s.io/v1beta1**	**node.k8s.io/v1**	いいえ

1. OpenShift Container Platform での Pod セキュリティーアドミッションの詳細については、Pod セキュリティーアドミッションの理解と管理 を参照してください。

1.5.2.2. oc registry login コマンドでの空のファイルと stdout のサポート

oc registry login コマンドの **--registry-config** および **--to option** オプションは、空のファイルを許可しなくなりました。これらのオプションは、存在しないファイルに対して引き続き機能します。出力を **-** (stdout) に書き込む機能も削除されました。

1.5.2.3. OpenShift CLI (oc) の RHEL 7 サポートが削除される

OpenShift CLI (**oc**) で Red Hat Enterprise Linux (RHEL) 7 を使用するためのサポートが削除されました。RHEL で OpenShift CLI (**oc**) を使用する場合は、RHEL 8 以降を使用する必要があります。

1.5.2.4. OpenShift CLI (oc) コマンドが削除される

以下の OpenShift CLI (**oc**) コマンドは本リリースで削除されました。

- **oc adm migrate etcd-ttl**
- **oc adm migrate image-references**
- **oc adm migrate legacy-hpa**
- **oc adm migrate storage**

1.5.2.5. モニターリング スタックから削除された Grafana コンポーネント

Grafana コンポーネントは、OpenShift Container Platform 4.12 モニターリングスタックの一部ではなくなりました。別の方法として、OpenShift Container Platform Web コンソールで **Observe** → **Dashboards** に移動して、モニターリングダッシュボードを表示します。

1.5.2.6. モニターリングスタックから削除された Prometheus および Grafana ユーザーインターフェイスアクセス

サードパーティーの Prometheus および Grafana ユーザーインターフェイスへのアクセスは、OpenShift Container Platform 4.12 モニターリングスタックから削除されました。別の方法として、OpenShift Container Platform Web コンソールで **Observe** をクリックして、モニターリングコンポーネントのアラート、メトリクス、ダッシュボード、およびメトリクスターゲットを表示します。

1.5.2.7. 仮想ハードウェアバージョン 13 のサポートを削除

OpenShift Container Platform 4.11 では、仮想ハードウェアバージョン 13 のサポートが削除されました。仮想ハードウェアバージョン 13 のサポートは OpenShift Container Platform 4.9 で非推奨になりました。Red Hat は、仮想ハードウェアのバージョン 15 以降の使用を推奨します。

1.5.2.8. スナップショット v1beta1 API エンドポイントのサポートを削除

OpenShift Container Platform 4.11 では、**snapshot.storage.k8s.io/v1beta1** API エンドポイントのサポートが削除されました。**snapshot.storage.k8s.io/v1beta1** API エンドポイントのサポートは OpenShift Container Platform 4.7 で非推奨となりました。Red Hat は、**snapshot.storage.k8s.io/v1** の使用を推奨します。**v1beta1** として作成されたすべてのオブジェクトは、v1 エンドポイントで利用できます。

1.5.2.9. カスタムスケジューラーの手動デプロイのサポートが削除される

本リリースでは、カスタムスケジューラーを手動でデプロイするためのサポートが削除されました。代わりに Red Hat OpenShift の Secondary Scheduler Operator を使用して、OpenShift Container Platform にカスタムセカンダリースケジューラーをデプロイします。

1.5.2.10. OpenShiftSDN を使用した単一ノードの OpenShift のデプロイサポート

本リリースでは、OpenShiftSDN を使用した単一ノードの OpenShift クラスターのデプロイのサポートが削除されました。OVN-Kubernetes は、単一ノードの OpenShift デプロイメントのデフォルトのネットワークソリューションです。

1.5.2.11. インストールペイロードからの Jenkins イメージの削除

- OpenShift Container Platform 4.11 は、"OpenShift Jenkins" および "OpenShift Agent Base" イメージを **registry.redhat.io** の **ocp-tools-4** リポジトリーに移動し、Red Hat が OpenShift Container Platform のライフサイクル外でイメージを生成および更新できるようにします。以前は、これらのイメージは OpenShift Container Platform インストールペイロードと、**registry.redhat.io** の **openshift4** リポジトリーにありました。詳細は、OpenShift Jenkins を参照してください。

- OpenShift Container Platform 4.11 は、ペイロードから "OpenShift Jenkins Maven" および "NodeJS Agent" イメージを削除します。OpenShift Container Platform 4.10 は以前、これらのイメージを非推奨にしました。Red Hat はこれらのイメージを生成しなくなり、**registry.redhat.io** の **ocp-tools-4** リポジトリーから入手できなくなりました。ただし、OpenShift Container Platform 4.11 にアップグレードしても、"OpenShift Jenkins Maven" および "NodeJS Agent" イメージは、4.10 以前のリリースから削除されません。また、Red Hat は、OpenShift Container Platform ライフサイクル ポリシー に従って、4.10 リリースライフサイクルの終わりまで、これらのイメージのバグ修正とサポートを提供します。

  詳細は、OpenShift Jenkins を参照してください。

1.5.3. 今後の Kubernetes API の削除

OpenShift Container Platform の次のマイナーリリースでは、Kubernetes 1.26 を使用する予定です。現在、Kubernetes 1.26 では、いくつかの廃止された API が削除される予定です。

予定されている Kubernetes API の削除一覧については、アップストリームの Kubernetes ドキュメントの Deprecated API Migration Guide を参照してください。

削除予定である Kubernetes API のクラスターを確認する方法は、Navigating Kubernetes API deprecations and removals を参照してください。

1.6. バグ修正

API サーバーと認証

- 以前は、**workloadIsBeingUpdatedTooLong** エラーを受け取ると、Cluster Authentication Operator の状態が **progressing = false** に設定されていました。同時に、**degraded = false** は、定義された **inertia** の期間保持されていました。その結果、Progressing の量が減少して Degradedation の時間が増加し、**progressing = false** および **degraded = false** が通常より早く設定される状況が発生していました。これにより、正常な状態が想定されていた OpenShift CI テストの一貫性が失われました。この問題は、**workloadIsBeingUpdatedTooLong** エラーが返されると **progressing = false** 設定を削除することで修正されました。これで **progressing = false** 状態がなくなり、OpenShift CI テストの一貫性が向上します。(BZ#2111842)

ベアメタルハードウェアのプロビジョニング

- サーバーファームウェアの最近のバージョンでは、サーバー操作の間隔が長くなりました。これにより、OpenShift Container Platform インストールプログラムが Baseboard Management Controller (BMC) からの応答を待機しているときに、インストーラーによってプロビジョニングされたインフラストラクチャーのインストール中にタイムアウトが発生します。新しい **python3-sushy** リリースでは、サーバー側で BMC への接続を試行する回数が増えています。この更新プログラムは、待ち時間の延長を考慮し、インストール中のタイムアウトを回避します。(OCPBUGS-4097)

- この更新の前は、Ironic プロビジョニングサービスは、厳密な eTag 検証と組み合わせた弱い eTag を使用するベースボード管理コントローラー (BMC) をサポートしていませんでした。設計上、BMC が弱い eTag を提供する場合、Ironic は 2 つの eTag を返します。元の eTag と、

弱い eTag をサポートしない BMC との互換性のために強い形式に変換された元の eTag です。Ironic は 2 つの eTag を送信できますが、厳密な eTag 検証を使用する BMC は、2 番目の eTag が存在するため、そのような要求を拒否します。その結果、一部の古いサーバーハードウェアでは、ベアメタルプロビジョニングが次のエラーで失敗しました: **HTTP 412 Precondition Failed**。OpenShift Container Platform 4.12 以降では、この動作が変更され、弱い eTag が提供された場合に Ironic が 2 つの eTag を送信しようとしなくなりました。代わりに、eTag に依存する Redfish リクエストが eTag 検証エラーで失敗した場合、Ironic は既知の回避策でリクエストを再試行します。これにより、eTag が厳密に検証されているマシンでベアメタルプロビジョニングが失敗するリスクが最小限に抑えられます。(OCPBUGS-3479)

- 今回の更新の前は、Redfish システムが設定 URI を備えている場合、Ironic プロビジョニングサービスは常にこの URI を使用して、ブート関連の BIOS 設定を変更しようとしまていました。ただし、ベースボード管理コントローラー (BMC) が設定 URI を備えていても、この設定 URI を使用した特定の BIOS 設定の変更をサポートしていない場合、ベアメタルプロビジョニングは失敗します。OpenShift Container Platform 4.12 以降では、システムが設定 URI を備えている場合、Ironic は続行する前に設定 URI を使用して特定の BIOS 設定を変更できることを確認します。それ以外の場合、Ironic はシステム URI を使用して変更を実装します。この追加のロジックにより、Ironic がブート関連の BIOS 設定の変更を適用でき、ベアメタルプロビジョニングが成功することが保証されます。(OCPBUGS-2052)

ビルド

- デフォルトでは、Buildah は環境変数の内容を含むステップをログファイルに出力します。これには、ビルド入力シークレット が含まれる場合があります。**--quiet** ビルド引数を使用してこれらの環境変数の出力を抑制することができますが、source-to-image (S2I) ビルドストラテジーを使用する場合は、この引数は使用できません。現在のリリースではこの問題は修正されています。環境変数の出力を抑制するには、ビルド設定で **BUILDAH_QUIET** 環境変数を設定します。

```
sourceStrategy:
...
  env:
    - name: "BUILDAH_QUIET"
      value: "true"
```

(BZ#2099991)

クラウドコンピュート

- 以前は、自動再起動の GCP インフラストラクチャーのデフォルトオプションを尊重するようにインスタンスが設定されていませんでした。その結果、自動再起動のインフラストラクチャーのデフォルトを使用せずにインスタンスを作成できました。これは、インスタンスが GCP で終了されても、関連付けられたマシンが自動的に再起動されなかったため、まだ **Running** の状態でリストされていることを意味していました。このリリースでは、自動再起動オプションを渡すためのコードが改善され、ユーザーからの既定のオプション選択をより適切に検出して渡すことができるようになりました。インスタンスはインフラストラクチャーのデフォルトを適切に使用するようになり、ユーザーがデフォルトの機能を要求すると自動的に再起動されます。(OCPBUGS-4504)

- **PodDisruptionBudget** オブジェクトの **v1beta1** バージョンは、Kubernetes で非推奨になりました。このリリースでは、**v1beta1** への内部参照が **v1** に置き換えられました。この変更はクラスターオートスケーラーの内部的なものであり、OpenShift Container Platform 4.12 へのアップグレードの準備 に関する Red Hat ナレッジベース記事のアドバイスを超えるユーザーアクションは必要ありません。(OCPBUGS-1484)

- 以前は、GCP マシンコントローラーは 10 時間ごとにマシンの状態を調整していました。他のプロバイダーは、この値を 10 分に設定して、マシン API システムの外部で発生した変更が短期

間で検出されるようにします。GCP の調整期間が長くなると、追加された外部 IP アドレスが長時間検出されないために証明書署名要求 (CSR) の承認が得られないなど、予期しない問題が発生する可能性があります。このリリースでは、GCP マシンコントローラーが更新され、10 分ごとに調整されて他のプラットフォームとの整合性が保たれ、外部の変更がより早く検出されるようになります。(OCPBUGS-4499)

- 以前は、Cluster Machine Approver Operator のデプロイメントの設定ミスが原因で、**TechPreviewNoUpgrade** 機能セットを有効にすると、エラーが発生し、Operator の性能が散発的に低下していました。**TechPreviewNoUpgrade** 機能セットが有効になっているクラスターは Cluster Machine Approver Operator の 2 つのインスタンスを使用し、両方のデプロイメントで同じポートセットを使用したため、単一ノードトポロジーのエラーにつながる競合が発生しました。このリリースでは、Cluster Machine Approver Operator デプロイメントが更新され、デプロイメントごとに異なるポートのセットを使用するようになりました。(OCPBUGS-2621)

- 以前は、Azure のゼロからのスケーリング機能は、インスタンスタイプの名前を CPU の数とインスタンスタイプに割り当てられたメモリー量にマッピングする、静的にコンパイルされたインスタンスタイプのリストに依存していました。このリストは、時間の経過とともに古くなっています。このリリースでは、インスタンスタイプのサイズに関する情報が Azure API から直接動的に収集され、リストが古くなるのを防ぎます。(OCPBUGS-2558)

- 以前はMachine API 終了ハンドラー Pod がスポットインスタンスで開始されませんでした。その結果、汚染されたスポットインスタンスで実行されていた Pod は、インスタンスが終了した場合に終了シグナルを受信しませんでした。これにより、ワークロードアプリケーションのデータが失われる可能性があります。このリリースでは、マシン API 終了ハンドラーのデプロイが変更され、テイントを許容するようになりました。また、テイントを持つスポットインスタンスで実行されている Pod は、終了シグナルを受信するようになりました。(OCPBUGS-1274)

- 以前は、Azure クラスターのエラーメッセージで、内部発行戦略のみを使用する切断されたインストールのパブリック IP アドレスを使用して新しいマシンを作成できないことが説明されていませんでした。このリリースでは、エラーメッセージが更新され、わかりやすくなりました。(OCPBUGS-519)

- 以前は、Cloud Controller Manager Operator は AWS クラスターの **cloud-config** 設定ファイルをチェックしませんでした。その結果、設定ファイルを使用して追加の設定を AWS クラウドコントローラーマネージャーコンポーネントに渡すことができませんでした。このリリースでは、Cloud Controller Manager Operator はインフラストラクチャーリソースをチェックし、**cloud-config** 設定ファイルへの参照を解析して、ユーザーが追加の設定を設定できるようにします。(BZ#2104373)

- 以前は、Azure が新しいインスタンスタイプを追加し、以前はサポートされていなかったインスタンスタイプで高速ネットワークのサポートを有効にしたときに、マシンコントローラー内の Azure インスタンスのリストが古くなりました。その結果、マシンコントローラーは、以前は高速ネットワークをサポートしていなかったインスタンスタイプを使用してマシンを作成できませんでした (たとえ Azure でこの機能がサポートされていたとしても)。このリリースでは、マシンが作成される前に必要なインスタンスタイプ情報が Azure API から取得され、最新の状態に保たれるため、マシンコントローラーは新しいインスタンスタイプおよび更新されたインスタンスタイプを使用してマシンを作成できます。この修正は、今後追加されるすべてのインスタンスタイプにも適用されます。(BZ#2108647)

- 以前は、クラスターオートスケーラーは、Cluster API プロバイダーを使用する場合、CSI ドライバーの AWS、IBM Cloud、および Alibaba Cloud トポロジーラベルを尊重しませんでした。その結果、スケールアウトイベント中にノードのバランスをとろうとしたときに、トポロジーラベルを持つノードがオートスケーラーによって適切に処理されませんでした。このリリースでは、オートスケーラーのカスタムプロセッサーが更新され、このラベルが尊重されるように

なりました。オートスケーラーは、AWS、IBM Cloud、または Alibaba CSI ラベルでラベル付けされた同様のノードグループのバランスを取ることができるようになりました。(BZ#2001027)

- 以前は、Power VS クラウドプロバイダーは DHCP サーバーからマシンの IP アドレスを取得できませんでした。IP アドレスを変更してもノードが更新されなかったため、保留中の証明書署名要求など、いくつかの不整合が発生していました。今回のリリースでは、ノードの IP アドレスがマシンの IP アドレスと一致するように、Power VS クラウドプロバイダーが更新され、DHCP サーバーからマシンの IP アドレスをフェッチするようになりました。(BZ#2111474)

- 以前は、初期バージョンの OpenShift Container Platform で作成された無効な設定のマシンは削除できませんでした。このリリースでは、無効な設定を持つマシンの作成を防止する Webhook は、既存の無効なマシンの削除を防止しなくなりました。ユーザーは、これらのマシンのファイナライザーを手動で削除することにより、クラスターからこれらのマシンを正常に削除できるようになりました。(BZ#2101736)

- 以前は、**NetworkManager** がデーモンとして実行されていない、または連続モードで実行されていないために短い DHCP リース時間が発生したため、初期プロビジョニング中にマシンが停止し、クラスター内のノードにならないことがありました。今回のリリースでは、追加のチェックが追加され、マシンがこの状態で停止した場合にマシンが削除され、自動的に再作成されるようになりました。このネットワーク状態の影響を受けるマシンは、マシン API コントローラーからの再起動後にノードになる可能性があります。(BZ#2115090)

- 以前は、IBM Cloud に存在しないマシンプロファイルを使用して新しい **Machine** リソースを作成すると、マシンが **Provisioning** フェーズで停止していました。このリリースでは、検証が IBM Cloud Machine API プロバイダーに追加され、マシンプロファイルが存在することを確認し、マシンプロファイルが無効なマシンは Machine API によって拒否されます。(BZ#2062579)

- 以前は、AWS の Machine API プロバイダーは、マシンの仕様で定義されたセキュリティーグループが存在することを確認していませんでした。この場合、エラーを返す代わりに、OpenShift Container Platform マシンに使用すべきではないデフォルトのセキュリティーグループを使用し、デフォルトのグループが使用されたことをユーザーに通知せずにマシンを正常に作成しました。このリリースでは、ユーザーがマシン仕様で誤った、または空のセキュリティーグループ名を設定すると、マシン API がエラーを返します。(BZ#2060068)

- 以前は、Machine API プロバイダーの Azure は、ユーザーが指定したインスタンスタイプの値を大文字と小文字を区別して処理しませんでした。これにより、インスタンスタイプは正しいが大文字と小文字が一致しない場合に、誤検知エラーが発生しました。このリリースでは、インスタンスタイプが小文字に変換されるため、ユーザーは大文字と小文字の不一致による誤検知エラーなしで正しい結果を得ることができます。(BZ#2085390)

- 以前は、オブジェクトへのアクセスを試行する前に、マシンオブジェクトの注釈で nil 値のチェックが行われませんでした。このような状況はめったにありませんが、マシンの調整時にマシンコントローラーがパニックに陥る原因となりました。このリリースでは、nil 値がチェックされ、マシンコントローラーは注釈なしでマシンを調整できます。(BZ#2106733)

- 以前は、クラスターの CPU とメモリーの使用量に関するクラスターオートスケーラーメトリクスが、**ClusterAutoscaler** リソースによって設定された制限に到達したり、超えたりすることはありませんでした。その結果、リソースの制限によりクラスターオートスケーラーがスケーリングできなかった場合、アラートは発生しませんでした。今回のリリースでは、クラスターオートスケーラーに **cluster_autoscaler_skiped_scale_events_count** と呼ばれる新しいメトリクスが追加され、リソースの制限に到達または超過したことをより正確に検出できるようになりました。クラスターリソースの制限に達したためにクラスターオートスケーラーがクラスターをスケールアップできない場合に、アラートが発生するようになりました。(BZ#1997396)

- 以前は、マシン API プロバイダーがマシン IP アドレスの取得に失敗した場合、内部 DNS 名が設定されず、マシン証明書署名要求が自動的に承認されませんでした。このリリースでは、Power VS マシンプロバイダーが更新され、IP アドレスの取得に失敗した場合でも、サーバー名を内部 DNS 名として設定するようになりました。(BZ#2111467)

- 以前は、マシン API の vSphere マシンコントローラーは、VM のクローン作成時に **PowerOn** フラグを設定していました。これにより、マシンコントローラーが認識しない **PowerOn** タスクが作成されました。その **PowerOn** タスクが失敗した場合、マシンは **Provisioned** フェーズでスタックし、電源がオンになりませんでした。このリリースでは、この問題を回避するためにクローニングシーケンスが変更されています。さらに、マシンコントローラーは、障害が発生した場合に VM の電源投入を再試行し、障害を適切に報告するようになりました。(BZ#2087981, OCPBUGS-954)

- このリリースでは、AWS セキュリティーグループは、作成後ではなく、すぐにタグ付けされます。これは、AWS に送信されるリクエストが少なくなり、必要なユーザー権限が低下することを意味します。(BZ#2098054, OCPBUGS-3094)

- 以前は、RHOSP レガシークラウドプロバイダーのバグにより、認証が失敗した後に特定の RHOSP 操作が試行された場合にクラッシュが発生していました。たとえば、サーバーをシャットダウンすると、Kubernetes コントローラーマネージャーが RHOSP からサーバー情報を取得し、このバグがトリガーされます。その結果、最初のクラウド認証が失敗したか、正しく設定されていない場合、サーバーをシャットダウンすると、Kubernetes コントローラーマネージャーがクラッシュしました。今回のリリースでは、RHOSP レガシークラウドプロバイダーが更新され、以前に正常に認証されていない場合、RHOSP API 呼び出しを試行しないようになりました。現在、無効なクラウド認証情報を使用してサーバーをシャットダウンしても、Kubernetes コントローラーマネージャーがクラッシュすることはなくなりました。(BZ#2102383)

開発者コンソール

- これまで **openshift-config** namespace は、**ProjectHelmChartRepository** カスタムリソースと同じ namespace であった **HelmChartRepository** カスタムリソースに対してハードコーディングされていました。これにより、ユーザーは目的の namespace にプライベート **ProjectHelmChartRepository** カスタムリソースを追加できませんでした。その結果、ユーザーは **openshift-config** namespace のシークレットと configmap にアクセスできませんでした。今回の更新で、**ProjectHelmChartRepository** カスタムリソース定義が修正され、正しいパーミッションを持つユーザーが選択した namespace からシークレットと configmaps を読み取ることができる **namespace** フィールドが追加されました。さらに、ユーザーはアクセス可能な namespace にシークレットと configmap を追加でき、作成リソースを使用した namespace にプライベート Helm チャートリポジトリーを追加できます。(BZ#2071792)

Image Registry

- これまで、イメージトリガーコントローラーにはオブジェクトを変更するパーミッションがありませんでした。その結果、イメージトリガーアノテーションが一部のリソースで機能しませんでした。今回の更新により、アノテーションに従ってオブジェクトを更新するために必要なパーミッションをコントローラーに提供するクラスターロールバインディングが作成されます。(BZ#2055620)

- これまで、Image Registry Operator には **node-ca** デーモンセットの **progressing** 状態がなく、正しくないオブジェクトからの **generation** が使用されていました。そのため、Operator がまだ実行されているのに **node-ca** デーモンセットが **degraded** としてマークされる可能性がありました。今回の更新では、インストールが完了していないことを示す **progressing** 状態が追加されます。その結果、Image Registry Operator は **node-ca** デーモンセットを正常にインストールし、インストーラーはそれが完全にデプロイされるまで待機するようになりました。(BZ#2093440)

インストーラー

- 以前は、サポートされているユーザー定義タグの数は 8 で、AWS リソース用に予約された OpenShift Container Platform タグは 2 でした。このリリースでは、サポートされるユーザー定義タグの数が 25 になり、AWS リソース用に予約された OpenShift Container Platform タグが 25 になりました。インストール時に最大 25 のユーザータグを追加できるようになりました。(CFE#592)

- 以前は、IAM 管理ユーザーに **s3:GetBucketPolicy** アクセス許可が割り当てられていない場合、Amazon Web Services へのクラスターのインストールが開始され、失敗していました。今回の更新により、必要なすべての権限が割り当てられていることを確認するためにインストールプログラムが使用するチェックリストに、このポリシーが追加されます。その結果、インストールプログラムは、IAM 管理ユーザーに **s3:GetBucketPolicy** 権限がないという警告を表示してインストールを停止するようになりました。(BZ#2109388)

- 以前は、Azure DCasv5 シリーズまたは DCadsv5 シリーズの機密 VM がコントロールプレーンノードとして指定されている場合、Microsoft Azure へのクラスターのインストールに失敗していました。今回の更新により、インストールプログラムは、機密 VM がまだサポートされていないことを示すエラーでインストールを停止するようになりました。(BZ#2055247)

- 以前は、コントロールプレーンマシンが実行されるまで、ブートストラップログを収集することはできませんでした。今回の更新により、ブートストラップログの収集に必要なのは、ブートストラップマシンが使用可能になっていることだけです。(BZ#2105341)

- 以前は、サービスアカウントに十分な権限がないためにクラスターを Google Cloud Platform にインストールできなかった場合、結果として表示されるエラーメッセージには、失敗の原因としてこれが記載されていませんでした。この更新により、エラーメッセージが改善され、サービスアカウントに割り当てられているアクセス許可を確認するようユーザーに指示されるようになりました。(BZ#2103236)

- 以前は、無効な GCP リージョンが指定されたために Google Cloud プロバイダー (GCP) へのインストールが失敗した場合、結果のエラーメッセージで、これが失敗の原因として言及されていませんでした。この更新により、エラーメッセージが改善され、リージョンが無効であることが示されるようになりました。(BZ#2102324)

- 以前は、Hive が古いバージョンの install-config.yaml ファイルを使用していると、Hive を使用したクラスターインストールが失敗することがありました。今回の更新により、インストールプログラムは、Hive が提供する **install-config.yaml** ファイルの古いバージョンを受け入れることができます。(BZ#2098299)

- 以前は、省略形式でアドレスをリストするなど、異なるアドレスを表す場合、インストールプログラムは **apiVIP** および **ingressVIP** パラメーターが同じ IPv6 アドレスを使用することを誤って許可していました。今回の更新では、インストーラーはフォーマットに関係なくこれら 2 つのパラメーターを正しく検証し、パラメーターごとに個別の IP アドレスを必要とします。(BZ#2103144)

- 以前は、インストールプログラムを使用してクラスターをアンインストールすると、クラスター名が 22 文字を超える場合、GCP にインストールされたクラスター内のすべてのリソースを削除できませんでした。今回の更新では、インストールプログラムを使用してクラスターをアンインストールすると、クラスター名が長い場合にすべての GCP クラスターリソースが正しく検索され、削除されます。(BZ#2076646)

- 以前は、**machineNetwork** パラメーターで複数のネットワークが定義されている Red Hat OpenStack Platform (RHOSP) にクラスターをインストールする場合、インストールプログラムは最初のネットワークのセキュリティーグループルールのみを作成していました。今回の更

新により、インストールプログラムは **machineNetwork** で 定義されたすべてのネットワークに対してセキュリティーグループルールを作成するため、ユーザーはインストール後にセキュリティーグループルールを手動で編集する必要がなくなりました。(BZ#2095323)

- 以前は、OpenStack にクラスターをインストールするときに、ユーザーが API および Ingress の仮想 IP アドレスを、DHCP サーバーの割り当てプールと競合する値に手動で設定することができました。これにより、DHCP サーバーが VIP アドレスの 1 つを新しいマシンに割り当て、起動に失敗する可能性があります。今回の更新では、インストールプログラムは、ユーザーが指定した VIP アドレスを検証して、DHCP プールと競合しないことを確認します。(BZ#1944365)

- 以前は、フォルダー内に埋め込まれたデータセンターを使用して vSphere にクラスターをインストールすると、インストールプログラムがデータセンターオブジェクトを見つけることができず、インストールが失敗していました。今回の更新では、インストールプログラムがデータセンターオブジェクトを含むディレクトリーを走査できるようになり、インストールが成功するようになりました。(BZ#2097691)

- 以前は、インストーラーによってプロビジョニングされたインフラストラクチャーで arm64 アーキテクチャーを使用して Azure にクラスターをインストールすると、**hyperVGeneration** V1 のイメージ定義リソースのアーキテクチャー値が誤って **x64** になりました。今回の更新により、**hyperVGeneration** V1 のイメージ定義リソースに **Arm64** の正しいアーキテクチャー値が含まれるようになりました。(OCPBUGS-3639)

- 以前は、VMware vSphere にクラスターをインストールするときに、ユーザーが **install-config.yaml** ファイルの **failureDomain** セクションでユーザー定義フォルダーを指定すると、インストールが失敗することがありました。今回の更新により、インストールプログラムは **install-config.yaml** ファイルの **failureDomain** セクションにあるユーザー定義フォルダーを正しく検証するようになりました。(OCPBUGS-3343)

- 以前は、VMware vSphere でインストールが失敗した後に部分的にデプロイされたクラスターを破棄すると、一部の仮想マシンフォルダーが破棄されませんでした。このエラーは、複数の vSphere データセンターまたは複数の vSphere クラスターで設定されたクラスターで発生する可能性があります。今回の更新により、インストールの失敗後に部分的にデプロイされたクラスターを破棄するときに、インストーラーによってプロビジョニングされたすべてのインフラストラクチャーが正しく削除されるようになりました。(OCPBUGS-1489)

- 以前は、VMware vSphere にクラスターをインストールするときに、ユーザーが **platform.vsphere.vcenters** パラメーターを指定したが、**install-config.yaml** ファイルで **platform.vsphere.failureDomains.topology.networks** パラメーターを指定しなかった場合、インストールは失敗しました。今回の更新により、インストールプログラムは、**platform.vsphere.vcenters** を指定するときに **platform.vsphere.failureDomains.topology.networks** フィールドが必須であることをユーザーに警告します。(OCPBUGS-1698)

- 以前は、VMware vSphere にクラスターをインストールするときに、ユーザーが **platform.vsphere.vcenters** および **platform.vsphere.failureDomains** パラメーターを定義したが、**platform.vsphere.defaultMachinePlatform.zones**、または **compute.platform.vsphere.zones** および **controlPlane.platform.vsphere.zones** を定義していない場合、インストールは失敗しました。今回の更新により、インストールプログラムは、ユーザーがインストール前にマルチリージョンまたはマルチゾーンデプロイメントで **zone** パラメーターを定義したか検証します。(OCPBUGS-1490)

Kubernetes コントローラーマネージャー

- 以前は、モニタリングスタックが存在しない環境で Kubernetes Controller Manager Operator が **degraded** を報告していました。今回の更新により、モニタリングスタックが存在しない場合、Kubernetes Controller Manager Operator は degradation の兆候のモニタリングをチェック

しません。(BZ#2118286)

- 今回の更新により、Kubernetes Controller Manager アラート (**KubeControllerManagerDown**、**PodDisruptionBudgetAtLimit**、**PodDisruptionBudgetLimit**、**GarbageCollectorSyncFailed**) に Github Runbook へのリンクが追加されました。Runbook は、ユーザーがこれらのアラートのデバッグを理解するのに役立ちます。(BZ#2001409)

Kubernetes Scheduler

- 以前は、セカンダリースケジューラーのカスタムリソースが削除されてもセカンダリースケジューラーのデプロイメントは削除されませんでした。その結果、Secondary Schedule Operator と Operand が完全にアンインストールされませんでした。今回の更新により、セカンダリースケジューラーのカスタムリソースに正しい所有者への参照が設定され、セカンダリースケジューラーのデプロイメントを指すようになりました。その結果、セカンダリースケジューラーのカスタムリソースが削除されると、セカンダリースケジューラーのデプロイメントも削除されます。(BZ#2100923)

- OpenShift Container Platform 4.12 リリースでは、デスケジューラーのプロファイルにロールベースアクセス制御 (RBAC) ルールが追加されるため、デスケジューラーは API グループにイベントを発行できるようになりました。(OCPBUGS-2330)

Machine Config Operator

- 以前は、重要な証明書を含む Machine Config Operator (MCO) **ControllerConfig** リソースは、Operator のデーモン同期が成功した場合にのみ同期されていました。設計上、デーモンの同期中にノードの準備ができていないと、デーモンの同期が成功しなくなります。そのため、準備ができていないノードは間接的に **ControllerConfig** リソースの同期を妨げていたため、これらの証明書の同期が妨げられていました。これにより、**ControllerConfig** リソースに含まれる証明書をローテーションできないために、準備ができていないノードが存在する場合、最終的にクラスターの機能が低下していました。今回のリリースでは、**ControllerConfig** リソースの同期はデーモン同期の成功に依存しなくなりました。そのため、デーモン同期が失敗した場合でも **ControllerConfig** リソースは同期を継続するようになりました。これは、準備ができていないノードが **ControllerConfig** リソースの同期を妨げることがなくなるため、準備ができていないノードがあっても証明書が更新され続けることを意味します。(BZ#2034883)

管理コンソール

- 以前は、**オペレータの詳細** ページは複数のエラーメッセージを表示しようとしましたが、エラーメッセージコンポーネントは一度に 1 つのエラーメッセージしか表示できませんでした。その結果、関連するエラーメッセージが表示されませんでした。今回の更新では、**オペレーターの詳細** ページに最初のエラーメッセージのみが表示されるため、関連するエラーがユーザーに表示されます。(OCPBUGS-3927)

- 以前は、カスタマーケースマネジメント (CCM) で Azure Red Hat OpenShift の製品名が正しくありませんでした。その結果、コンソールは、CCM のフィールドに正しく入力するために、同じ誤った製品名を使用する必要がありました。CCM の製品名が更新されたら、コンソールも更新する必要がありました。今回の更新により、コンソールからリンクをたどると、CCM と同じ正しい製品名に正しい Azure 製品名が正しく取り込まれます。(OCPBUGS-869)

- 以前は、プラグインページでエラーが発生した場合、エラーページから離れたときにエラーがリセットされず、エラーの原因ではないページに移動した後もエラーが持続していました。今回の更新により、ユーザーが新しいページに移動したときにエラー状態がデフォルトにリセットされ、新しいページに移動した後にエラーが持続しなくなりました。(BZ#2117738, OCPBUGS-523)

- 以前は、**すべてのネームスペース** が選択されている場合、インストールされた Operator の **Operator 詳細** ペインにある **View it here** リンクが正しく構築されませんでした。その結果、

リンクは **すべてのプロジェクト** のクラスターサービスバージョン (CSV) の **オペレーターの詳細** ページに移動しようとしましたが、これは無効なルートです。今回の更新により、CSV がインストールされている namespace を使用する **ここで表示** リンクが正しくビルドされ、リンクが期待どおりに機能するようになりました。(OCPBUGS-184)

- 以前は、5 桁を超える行番号を使用すると、行番号が行番号と行の内容の間の垂直の仕切りに重なって読みにくくなるという表面的な問題が発生していました。今回の更新で、行番号に使用できるスペースの量が増え、行番号が長くなったため、行番号が垂直の仕切りに重ならなくなりました。(OCPBUGS-183)

- 以前は、Web コンソールの管理者パースペクティブで、**クラスター設定** ページの **既定の更新サーバー** ポップアップウィンドウにある **OpenShift ローカル更新サービスの詳細を確認する** へのリンクで 404 エラーが発生していました。今回の更新により、リンクは期待どおりに機能します。(BZ#2098234)

- 以前は、**MatchExpression** コンポーネントは配列型の値を考慮していませんでした。その結果、このコンポーネントを使用してフォームから入力できる値は 1 つだけでした。今回の更新により、**MatchExpression** コンポーネントはコンマ区切りの値を配列として受け入れるようになりました。(BZ#207690)

- 以前は、モデルの冗長なチェックが行われてタブのリロードが発生し、再レンダリングされたタブのコンテンツがちらつくことがありました。今回の更新で、冗長なモデルチェックが削除され、モデルは 1 回だけチェックされます。その結果、タブのコンテンツがちらつき、再レンダリングされなくなりました。(BZ#2037329)

- 以前は、OpenShift Dedicated ノードページのアクションリストから **edit** ラベルを選択すると、応答がなく、Web フックエラーが返されていました。この問題は修正され、編集が失敗した場合にのみエラーメッセージが返されるようになりました。(BZ#2102098)

- 以前は、問題が保留中の場合、**Insights** リンクをクリックするとページがクラッシュしていました。回避策として、変数の **initialized** を待ってから、**Insights** リンクをクリックします。その結果、Insights ページが期待どおりに開きます。(BZ#2052662)

- 以前は、**MachineConfigPool** リソースが一時停止された場合の一時停止の解除オプションとして **Resume rollouts** が表示されていました。この文言が更新され、**Resume updates** と表示されるようになりました。(BZ#2094240)

- 以前は、マスターノードとワーカーノードのカウントに間違った計算方法が使用されていました。今回の更新により、ノードに **master** と **worker** の両方のロールがある場合に、正しいワーカーノードが計算されるようになりました。(BZ#1951901)

- 以前は、**ImageManifestVuln** の **react-router** ルートが競合していると、**ImageManifestVuln** の詳細ページを **~new** の名前でレンダリングしようとしていました。今回、競合ルートを削除するようにコンテナーセキュリティープラグインが更新され、Operator の詳細ページで動的リストと詳細ページのエクステンションが使用されるようになりました。その結果、コンソールは **ImageManifestVuln** の正しい作成、リスト、および詳細ページをレンダリングします。(BZ#2080260)

- 以前は、不完全な同期されていない YAML がユーザーに表示されることがありました。今回の更新により、常に同期された YAML が表示されるようになりました。(BZ#2084453)

- 以前は、使用するためにカスタムリソース (CR) を作成する必要がある Operator をインストールする場合、**Create resource** ボタンが間違った namespace を指しているために CR のインストールに失敗することがありました。今回の更新により、**Create resource** ボタンが期待どおりに機能するようになりました。(BZ#2094502)

- 以前は、**Cluster update** モーダルでエラーが正しく表示されませんでした。その結

果、**Cluster update** モーダルは、エラーが発生してもエラーを表示または説明しませんでした。今回の更新により、**Cluster update** モーダルでエラーが正しく表示されるようになりました。(BZ#2096350)

モニタリング

- この更新の前は、クラスター管理者は、スケジューリングの問題のために準備ができていない Pod と、kubelet によって開始できなかったために準備ができていない Pod を区別できませんでした。どちらの場合も、**KubePodNotReady** アラートが発生します。今回の更新により、スケジューリングの問題のために Pod の準備ができていない場合に **KubePodNotScheduled** アラートが発生し、kubelet によって開始できなかったために Pod の準備ができていない場合に **KubePodNotReady** アラートが発生するようになりました。(OCPBUGS-4431)

- この更新の前は、**node_exporter** は **tun** インターフェイス、**br** インターフェイス、**ovn-k8s-mp** インターフェイスなどの仮想ネットワークインターフェイスに関するメトリックを報告していました。今回の更新により、これらの仮想インターフェイスのメトリックは収集されなくなり、監視リソースの消費が減少します。(OCPBUGS-1321)

- この更新の前は、DNS 解決が遅いために Alertmanager Pod の起動がタイムアウトすることがあり、Alertmanager Pod が起動しませんでした。今回のリリースでは、タイムアウト値が 7 分に増やされ、Pod の起動がタイムアウトするのを防ぎます。(BZ#2083226)

- この更新の前は、Prometheus Operator が Prometheus Pod の実行またはスケジュールに失敗した場合、システムは失敗の根本的な理由を提供しませんでした。今回の更新により、Prometheus Pod が実行またはスケジュールされていない場合、Cluster Monitoring Operator は **clusterOperator** の監視ステータスを失敗の理由で更新します。これは、根本的な問題のトラブルシューティングに使用できます。(BZ#2043518)

- 今回の更新の前に、OpenShift Container Platform Web コンソールで **開発者** パースペクティブからアラートのサイレンスを作成した場合、アラートと一致しない外部ラベルが含まれていました。したがって、アラートは消音されません。今回の更新により、**Developer** パースペクティブでサイレンスを作成するときに外部ラベルが除外されるようになり、新しく作成されたサイレンスが期待どおりに機能するようになりました。(BZ#2084504)

- 以前は、ユーザー定義プロジェクト専用の Alertmanager のインスタンスを有効にすると、特定の状況で設定ミスが発生する可能性があり、ユーザー定義プロジェクトの Alertmanager 設定マップ設定が Alertmanager のメインインスタンスのどちらにもロードされなかったことが通知されませんでした。またはユーザー定義プロジェクト専用のインスタンス。今回のリリースでは、この設定ミスが発生した場合、Cluster Monitoring Operator は問題を通知し、解決手順を提供するメッセージを表示するようになりました。(BZ#2099939)

- この更新の前は、Cluster Monitoring Operator (CMO) が Prometheus の更新に失敗した場合、CMO は以前のデプロイが実行されているかどうかを確認せず、Prometheus Pod の 1 つがまだ実行されていてもクラスター監視が利用できないと報告していました。今回の更新により、CMO はこの状況で実行中の Prometheus Pod をチェックし、Prometheus Pod が実行されていない場合にのみクラスター監視が利用できないことを報告するようになりました。(BZ#2039411)

- この更新の前に、OpsGenie をアラートレシーバーとして設定した場合、**api_key** と **api_key_file** は相互に排他的であり、**api_key** が優先されるという警告がログに表示されていました。この警告は、**api_key_file** を定義していない場合でも表示されました。今回の更新により、この警告は、**api_key** と **api_key_file** の両方を定義した場合にのみログに表示されます。(BZ#2093892)

- この更新の前は、Telemeter Client (TC) は、手動で再起動したときにのみ新しいプルシークレットをロードしていました。したがって、プルシークレットが変更または更新され、TC が再起動されていない場合、TC はサーバーでの認証に失敗します。今回の更新で問題が解決され、

シークレットがローテーションされるとデプロイが自動的に再開され、更新されたトークンを使用して認証されるようになりました。(BZ#2114721)

ネットワーク

- 以前は、終了状態にあったルーターが **oc cp** コマンドを遅らせ、Pod が終了するまで **oc adm must-gather** コマンドを遅らせていました。今回の更新では、**must-gather** コマンドの実行が遅れないように、発行された **oc cp** コマンドごとにタイムアウトが設定されます。その結果、Pod を終了しても、**must-gather** コマンドが遅延することはなくなりました。(BZ#2103283)

- 以前は、**Private** エンドポイントの公開戦略タイプと PROXY プロトコルの両方を使用してイングレスコントローラーを設定することはできませんでした。今回の更新により、ユーザーは **Private** エンドポイント公開戦略タイプと PROXY プロトコルの両方を使用してイングレスコントローラーを設定できるようになりました。(BZ#2104481)

- 以前は、**routeSelector** パラメーターは、ルーターのデプロイ前にイングレスコントローラーのルートステータスをクリアしていました。このため、ルートステータスが正しく再入力されませんでした。古いデータの使用を避けるために、ルートステータス検出が更新され、Kubernetes オブジェクトキャッシュに依存しなくなりました。さらに、この更新プログラムには、ルートの状態を判断するために、ルートのデプロイメントで生成 ID を確認するための修正が含まれています。その結果、ルートステータスは **routeSelector** の更新で一貫してクリアされます。(BZ#2101878)

- 以前は、OpenShift Container Platform の 4.8 より前のバージョンからアップグレードされたクラスターは、孤立した **Route** オブジェクトを持つ可能性がありました。これは、特定の **Ingress** オブジェクトが示す **IngressClass** に関係なく、OpenShift Container Platform の以前のバージョンが **Ingress** オブジェクトを **Route** オブジェクトに変換することが原因でした。今回の更新により、Ingress から Route への変換後にクラスター内に孤立した Route オブジェクトがまだ存在することについて、アラートがクラスター管理者に送信されます。今回の更新では、**IngressClass** を指定していない Ingress オブジェクトについてクラスター管理者に通知する別のアラートも追加されています。(BZ#1962502)

- 以前は、ルーターのデプロイメントが依存する **configmap** が作成されていない場合、ルーターのデプロイメントは進行しませんでした。今回の更新により、デフォルトのイングレスコントローラーのデプロイが進行中の場合、クラスター Operator は **ingress progressing=true** を報告します。これにより、ユーザーはコマンド **oc get co** を使用してイングレスコントローラーの問題をデバッグすることになります。(BZ#2066560)

- 以前は、誤って作成されたネットワークポリシーが OVN-Kubernetes キャッシュに追加されると、OVN-Kubernetes リーダーが **crashloopbackoff** 状態になることがありました。今回の更新により、OVN-Kubernetes リーダーは nil ポリシーの削除をスキップして **crashloopbackoff** 状態にならなくなりました。(BZ#2091238)

- 以前は、同じ namespace または名前を持つ古い EgressIP Pod を削除してから 60 秒以内に同じ namespace または名前を持つ EgressIP Pod を再作成すると、間違った SNAT が設定されていました。その結果、パケットは EgressIP SNAT ではなく nodeIP で送信される可能性がありました。今回の更新により、トラフィックは nodeIP ではなく EgressIP を使用して Pod から送信されます。(BZ#2097243)

- 以前は、ACL の **arp** から **arp ll nd** への変更により、**arp** を使用する古いアクセス制御リスト (ACL) で、**unexpectedly found multiple equivalent ACLs (arp v/s arp||nd)** エラーが生成されていました。これにより、ネットワークポリシーが適切に作成されませんでした。今回の更新では、**arp** 一致のみを持つ古い ACL が削除され、新しい **arp ll nd** 一致を持つ ACL のみが存在するようになりました。これにより、ネットワークポリシーが正しく作成され、**ovnkube-master** でエラーが観察されなくなります。注: これは、古いバージョンから 4.8.14、4.9.32、4.10.13 以降にアップグレードするお客様に影響します。(BZ#2095852).

- 今回の更新により、CoreDNS は Kubernetes 1.25 に基づくバージョン 1.10.0 に更新されました。これにより、CoreDNS バージョンと、これも Kubernetes 1.25 に基づく OpenShift Container Platform 4.12 の両方が相互に調整されます。(OCPBUGS-1731)

- 今回の更新により、OpenShift Container Platform ルーターは、Kubernetes 1.25 をサポートする **k8s.io/client-go** バージョン 1.25.2 を使用するようになりました。これにより、**openshift-router** と、これも Kubernetes 1.25 に基づく OpenShift Container Platform 4.12 の両方が相互に調整されます。(OCPBUGS-1730)

- 今回の更新により、Ingress Operator は Kubernetes 1.25 をサポートする **k8s.io/client-go** バージョン 1.25.2 を使用するようになりました。これにより、Ingress Operator と、これも Kubernetes 1.25 に基づく OpenShift Container Platform 4.12 の両方が相互に調整されます。(OCPBUGS-1554)

- 以前は、DNS Operator は **openshift-dns** namespace を調整しませんでした。OpenShift Container Platform 4.12 では **openshift-dns** namespace に pod-security ラベルが必要であるため、クラスターの更新時に namespace にこれらのラベルが欠落していました。Pod-security ラベルがないと、Pod は起動できませんでした。今回の更新により、DNS Operator は **openshift-dns** namespace を調整し、pod-security ラベルが存在するようになりました。その結果、Pod は期待どおりに起動します。(OCPBUGS-1549)

- これまで、**ingresscontroller.spec.tuniningOptions.reloadInterval** は有効なパラメーター値として 10 進数をサポートしていませんでした。これは、Ingress オペレーターが指定された値をミリ秒に内部的に変換するためで、これはサポートされている時間単位ではありませんでした。これにより、イングレスコントローラーが削除されませんでした。今回の更新により、**ingresscontroller.spec.tuningOptions.reloadInterval** が 10 進数をサポートするようになり、ユーザーは、以前はサポートされていなかった **reloadInterval** パラメーター値を使用して Ingress コントローラーを削除できるようになりました。(OCPBUGS-236)

- 以前は、クラスター DNS オペレーターは Kubernetes 1.24 に基づく GO Kubernetes ライブラリーを使用していましたが、OpenShift Container Platform 4.12 は Kubernetes 1.25 に基づいていました。今回の更新により、GO Kubernetes API は v1.25.2 になり、クラスター DNS オペレーターが Kubernetes 1.25 API を使用する OpenShift Container Platform 4.12 と連携します。(リンク: OCPBUGS-1558)

- 以前は、**disableNetworkDiagnostics** 設定を **true** に設定しても、**network-operator** Pod が再作成されると保持されませんでした。今回の更新により、ネットワーク Operator を再起動しても、network`operator.openshift.io/cluster` の **disableNetworkDiagnostics** 設定プロパティーはデフォルト値にリセットされなくなりました。(OCPBUGS-392)

- 以前は、**ovn-kubernetes** は **br-ex** ブリッジでボンディングされたインターフェイスの正しい MAC アドレスを設定しませんでした。その結果、プライマリー Kubernetes インターフェイスにボンディングを使用するノードは、クラスターに参加できませんでした。今回の更新により、**ovn-kubernetes** は **br-ex** ブリッジでボンディングされたインターフェイスの正しい MAC アドレスを設定し、プライマリー Kubernetes インターフェイスにボンディングを使用するノードはクラスターに正常に参加できるようになりました。(BZ2096413)

- 以前は、Ingress Operator が mTLS の使用を有効にするように設定されている場合、Operator は、他のイベントによって調整されるまで、CRL の更新が必要かどうかを確認しませんでした。その結果、mTLS に使用される CRL が古くなる可能性がありました。今回の更新により、CRL の有効期限が切れると Ingress Operator が自動的に調整し、CRL は **nextUpdate** フィールドで指定された時間に更新されるようになりました。(BZ#2117524)

ノード

- 以前は、シンボリックリンクのエラーメッセージがエラーとしてフォーマットされるのではなく、生データとして出力されていたため、理解が困難でした。今回の修正により、エラーメッセージが適切にフォーマットされ、理解しやすくなりました。(BZ#1977660)

セーンが適切にフォーマットされ、理解しやすくなりました。(BZ#1977660)

- 以前は、パフォーマンスプロファイルがノードに適用された場合の kubelet のハードエビクションのしきい値は Kubernetes のデフォルトとは異なりました。今回のリリースでは、期待される Kubernetes のデフォルトに一致するようにデフォルトが更新されました。(OCPBUGS-4362).

OpenShift CLI (oc)

- OpenShift Container Platform 4.12 リリースでは、ターゲット namespace に適切なセキュリティーレベルがない場合にターゲットノードでデバッグセッションに入ると問題が修正されます。これにより、**oc** CLI で Pod セキュリティーエラーメッセージが表示されました。既存の namespace に適切なセキュリティーレベルが含まれていない場合、ターゲットノードで **oc** デバッグモードに入ると、OpenShift Container Platform は一時的な namespace を作成するようになりました。(OCPBUGS-852)

- 以前は、macOS arm64 アーキテクチャーでは、**oc** バイナリーを手動で署名する必要がありました。その結果、**oc** バイナリーは期待どおりに機能しませんでした。今回の更新では、**oc** を模倣するための自己署名バイナリーが実装されています。その結果、macOS arm64 アーキテクチャーの **oc** バイナリーは適切に動作します。(BZ#2059125)

- 以前は、**must-gather** はサーバー上に存在しないリソースを収集しようとしていました。そのため、**must-gather** はエラーメッセージを出力していました。今回、**must-gather** はリソースを収集する前にリソースが存在するかどうかをチェックするようになりました。その結果、**must-gather** は、サーバー上に存在しないリソースの収集に失敗してもエラーを出力しなくなりました。(BZ#2095708)

- OpenShift Container Platform 4.12 リリースでは、**oc-mirror** ライブラリーが更新され、ライブラリーがマルチアーキテクチャープラットフォームイメージをサポートするようになりました。これは、プラットフォームリリースペイロードをミラーリングするときに、**arm64** をはじめとする幅広いアーキテクチャーから選択できることを意味します。(OCPBUGS-617)

Operator Lifecycle Manager (OLM)

- OpenShift Container Platform 4.12 リリースより前では、**オンクラスター** 機能の問題により、**package-server-manager** コントローラーは **package-server** クラスターサービスバージョン (CSV) に加えられた変更を元に戻しませんでした。これらの永続的な変更は、Operator がクラスターで開始する方法に影響を与える可能性があります。OpenShift Container Platform 4.12 の場合、**package-server-manager** コントローラーは常に **package-server** CSV を元の状態に再構築するため、クラスターのアップグレード操作後に CSV への変更が保持されることはありません。**on-cluster** 関数は、**package-server** CSV の状態を制御しなくなりました。(OCPBUGS-867)

- 以前は、ラベルが namespace に存在する場合でも、Operator Lifecycle Manager (OLM) は namespace を更新してラベルを適用しようとしました。そのため、更新リクエストにより、API および etcd サービスのワークロードが増加しました。今回の更新により、OLM は、更新を発行する前に namespace で既存ラベルと期待されるラベルを比較するようになりました。その結果、OLM は namespace に対して不要な更新をリクエストしなくなりました。(BZ#2105045)

- これまで Operator Lifecycle Manager (OLM) は、**ClusterVersion** カスタムリソースの **spec.DesiredVersion** フィールドの計算ミスに基づき、ブロックされるべきではないマイナークラスターアップグレードを阻止していました。今回の更新により、OLM は、アップグレードをサポートする必要がある場合にクラスターのアップグレードを阻止しなくなりました。(BZ#2097557)

- これまでリコンサイラーは、リソースのコピーを作成せずにリソースのアノテーションを更新していました。これにより、レコンサイラープロセスを終了させるエラーが発生していまし

た。今回の更新により、リコンサイラーはエラーによって停止しなくなりました。(BZ#2105045)

- **package-server-manifest** (PSM) は、正しい **package-server** Cluster Service Version (CSV) がクラスターにインストールされていることを確認するコントローラーです。以前は、クラスター上のオブジェクトが期待されるオブジェクトに影響を与える可能性がある調整機能の論理エラーが原因で、**package-server** CSV への変更は元に戻りませんでした。ユーザーは **package-server** CSV を変更でき、変更は元に戻りませんでした。さらに、クラスターのアップグレードでは、**package-server** CSV の YAML が更新されませんでした。今回の更新により、期待されるバージョンの CSV が常にゼロからビルドされるようになりました。これにより、クラスター上のオブジェクトが期待される値に影響を与えることができなくなります。その結果、PSM は **package-server** CSV を変更しようとする試みを元に戻すことができ、クラスターをアップグレードすると期待される **package-server** CSV がデプロイされるようになりました。(OCPBUGS-858)

- 以前は、OLM は Operator の CRD ステータスに従って Operator をアップグレードしていました。CRD は、グループ/バージョン/種類 (GVK) 識別子によって定義された順序でコンポーネント参照を一覧表示します。同じコンポーネントを共有する Operator により、GVK が Operator のコンポーネントリストを変更する可能性があり、これにより、OLM が CRD のステータスを継続的に更新するためにより多くのシステムリソースを必要とする可能性があります。今回の更新により、Operator Lifecycle Manager (OLM) は、Operator のコンポーネント参照に従って Operator をアップグレードするようになりました。Operator のカスタムリソース定義 (CRD) ステータスの変更は、OLM Operator のアップグレードプロセスには影響しません。(OCPBUGS-3795)

Operator SDK

- 今回の更新により、Pod 仕様に **securityContext** 設定フィールドを含めることで、レジストリー Pod のセキュリティーコンテキストを設定できるようになりました。これにより、Pod 内のすべてのコンテナーにセキュリティーコンテキストが適用されます。**securityContext** フィールドは、Pod の権限も定義します。(BZ#2091864)

File Integrity Operator

- これまで File Integrity Operator は、Operator のパーミッションで **openshift-file-integrity** namespace を使用してテンプレートをデプロイしていました。Operator が namespace にオブジェクトを作成しようとすると、パーミッションの問題により失敗しました。今回のリリースでは、OLM により使用されるデプロイメントリソースが更新され、パーミッションの問題を修正して正しい namespace を使用するようになりました。これにより、ユーザーはデフォルト以外の namespace に Operator をインストールして使用できるようになりました。(BZ#2104897)

- 以前は、File Integrity Operator の基本的な依存関係によってアラートと通知の処理方法が変更され、結果として Operator はメトリクスを送信しませんでした。このリリースでは、Operator はメトリクスエンドポイントが正しく、起動時に到達可能であることを確認します。(BZ#2115821)

- 以前は、File Integrity Operator によって発行されたアラートは namespace を設定しませんでした。これにより、アラートがどこから発信されたのか、またはどのコンポーネントがアラートを発行したのかを理解することが困難になりました。今回のリリースでは、Operator にインストール先の namespace がアラートに含まれるようになり、注意が必要なコンポーネントを絞り込みやすくなりました。(BZ#2101393)

- 以前は、File Integrity Operator は、アップグレード中にアラートの変更を適切に処理しませんでした。その結果、アラートには、Operator がインストールされた namespace が含まれていませんでした。今回のリリースでは、Operator にインストール先の namespace がアラートに

含まれるようになり、注意が必要なコンポーネントを絞り込みやすくなりました。(BZ#2112394)

- 以前は、File Integrity Operator のサービスアカウント所有権で基礎となる OLM の更新により リグレッションが発生し、0.1.24 から 0.1.29 への更新が壊れていました。今回の更新により、Operator はデフォルトで 0.1.30 にアップグレードされます。(BZ#2109153)

- 以前は、File Integrity Operator デーモンは、最近のアクセス許可の変更に対して **Roles** パラメーターの代わりに **ClusterRoles** パラメーターを使用していました。その結果、OLM は Operator を更新できませんでした。今回のリリースでは、Operator デーモンは **Roles** パラメーターの使用に戻り、古いバージョンからバージョン 0.1.29 へのアップグレードは正常に行われます。(BZ#2108475)

コンプライアンス Operator

- 以前は、Compliance Operator は古いバージョンの Operator SDK を使用していました。これは、Operator を構築するための依存関係です。これにより、Operator SDK で使用される非推奨の Kubernetes 機能に関するアラートが発生しました。今回のリリースでは、Compliance Operator がバージョン 0.1.55 にアップグレードされ、Operator SDK の更新バージョンが含まれています。(BZ#2098581)

- 以前は、**rhcos4-high-master-sysctl-kernel-yama-ptrace-scope** および **rhcos4-sysctl-kernel-core-pattern** ルールに自動修復を適用すると、修復されたにもかかわらず、スキャン結果でこれらのルールが失敗する結果になりました。この問題は、このリリースで修正されています。(BZ#2094382)

- 以前は、Compliance Operator がデフォルトの namespace に通知をハードコーディングしていました。その結果、Operator が別の namespace にインストールされている場合、Operator からの通知は表示されませんでした。この問題は、このリリースで修正されています。(BZ#2060726)

- 以前は、Compliance Operator は、Ignition 仕様のないマシン設定を解析するときに API リソースを取得できませんでした。これにより、**api-check-pods** チェックでクラッシュループが発生しました。今回のリリースでは、Compliance Operator が更新され、Ignition 仕様なしでマシン設定プールを適切に処理できるようになりました。(BZ#2117268)

- これまで Compliance Operator は、マシン設定と kubelet 設定の間の関係を判別できなかったため、マシン設定をスタック状態に保持していました。これは、マシン設定名に関する誤った仮定が原因でした。このリリースでは、コンプライアンスオペレーターは、kubelet 設定がマシン設定のサブセットであるかどうかを判断できます。(BZ#2102511)

OpenShift API サーバー

- 以前は、メンバーを追加すると、以前のメンバーがグループから削除されることがありました。その結果、ユーザーはグループ権限を失いました。今回のリリースでは、依存関係が強化され、ユーザーがグループ権限を失うことはなくなりました。(OCPBUGS-533)

Red Hat Enterprise Linux CoreOS (RHCOS)

- 以前は、Podman 4.0 に更新すると、ユーザーは RHCOS のツールボックスコンテナーでカスタムイメージを使用できなくなりました。この修正により、Podman の新しい動作に対応するようにツールボックスライブラリーコードが更新されるため、ユーザーは期待どおり RHCOS のツールボックスでカスタムイメージを使用できるようになりました。(BZ#2048789)

- 以前は、**podman exec** コマンドはネストされたコンテナーではうまく機能しませんでした。**oc debug** コマンドを使用してノードにアクセスし、**toolbox** コマンドを使用してコンテナーを実行すると、この問題が発生しました。このため、ユーザーは RHCOS でツールボック

スを再利用できませんでした。この修正により、この動作を考慮してツールボックスライブラリーコードが更新され、ユーザーは RHCOS でツールボックスを再利用できるようになりました。(BZ#1915537)

- 今回の更新により、**ツールボックス** コマンドを実行すると、コンテナーを起動する前に既定のイメージの更新がチェックされるようになりました。これにより、セキュリティーが向上し、ユーザーに最新のバグ修正が提供されます。(BZ#2049591)

- 以前は、Podman 4.0 に更新すると、ユーザーは RHCOS で **toolbox** コマンドを実行できなくなりました。この修正により、Podman の新しい動作に対応するようにツールボックスライブラリーコードが更新されるため、ユーザーは期待どおりに RHCOS で **toolbox** を実行できるようになりました。(BZ#2093040)

- 以前は、カスタム SELinux ポリシーモジュールは **rpm-ostree** で適切にサポートされていなかったため、更新時にシステムの残りの部分と共に更新されませんでした。これは、無関係なコンポーネントの障害として表面化します。今後の OpenShift Container Platform リリースでの SELinux ユーザー空間の改善が保留されているため、この更新は、必要に応じてブート中に SELinux ポリシーを再構築および再ロードする RHCOS への回避策を提供します。(OCPBUGS-595)

スケーラビリティーおよびパフォーマンス

- 調整済みプロファイルが\変更され、最近の Red Hat Enterprise Linux (RHEL) カーネルパッチに追加され、新しく導入された CPU ごとの kthreads (**ktimers**) に **ksoftirqd** および **rcuc** と同じ優先度を割り当てるようになりました。詳細は、OCPBUGS-3475、BZ#2117780、BZ#2122220 を参照してください。

- 以前は、**tuned** サービスを再起動すると、**irqbalance** 設定が不適切にリセットされ、分離された CPU で再度 IRQ 操作が行われ、分離保証に違反していました。今回の修正により、**tuned** サービスを (明示的またはバグにより) 再起動しても **irqbalance** サービスの設定が適切に保持され、IRQ サービスに関する CPU 分離保証が保たれるようになりました。(OCPBUGS-585)

- 以前は、調整済みデーモンがクラスター Node Tuning Operator の一部として順不同で再起動されると、割り込みハンドラーの CPU アフィニティーがリセットされ、調整が損なわれていました。今回の修正により、tuned の **irqbalance** プラグインが無効になり、OpenShift Container Platform は **CRI-O** と **irqbalance** の間のロジックとインタラクションに依存するようになりました。(BZ#2105123)

- 以前は、ノードに負荷がかかっているときに新しい **veth** デバイスごとに低遅延フックスクリプトを実行すると、時間がかかりすぎていました。その結果、Pod 開始イベント中に遅延が蓄積され、**kube-apiserver** のロールアウト時間が遅くなり、5 分のロールアウトタイムアウトを超えることもありました。今回の修正により、コンテナーの開始時間が短縮され、5 分のしきい値内に収まるはずです。(BZ#2109965)

- 以前は、**oslat** 制御スレッドがテストスレッドの 1 つと併置されていたため、測定でレイテンシースパイクが発生していました。今回の修正により、**oslat** ランナーは制御スレッド用に 1 つの CPU を確保するようになり、テストがビジー状態のスレッドを実行するために使用する CPU が 1 つ少なくなりました。(BZ#2051443)

- **oslat**、**cyclictest**、**hwlatdetect** とも呼ばれるレイテンシー測定ツールは、完全に分離された CPU 上で実行され、レイテンシースパイクの原因となるヘルパープロセスはバックグラウンドで実行されるようになりました。そのため、より正確なレイテンシー測定が可能になりました。(OCPBUGS-2618)

- 以前は、**group-du-sno-ranGen.yaml** の参照 **PolicyGenTemplate** には 2 つの **StorageClass** エントリーが含まれていましたが、生成されたポリシーには 1 つしか含まれていませんでした。今回の更新により、生成されたポリシーに両方のポリシーが含まれるようになりました。

(BZ#2049306)

ストレージ

- 以前は、汎用的な一時ボリュームのチェックに失敗していました。今回の更新により、拡張可能なボリュームのチェックに汎用的な一時ボリュームが含まれるようになりました。(BZ#2082773)

- 以前は、vSphere に複数のシークレットが存在する場合、vSphere CSI Operator がランダムにシークレットを選択し、Operator が再起動することがありました。今回の更新により、vCenter CSI Operator に複数のシークレットがある場合に警告が表示されるようになりました。(BZ#2108473)

- 以前は、Container Storage Interface (CSI) ドライバーがノードからボリュームをアンマウントできない場合、OpenShift Container Platform はボリュームをデタッチしていました。アンマウントせずにボリュームをデタッチすることは CSI 仕様では許可されておらず、ドライバーが **undocumented** 状態になる可能性があります。今回の更新により、CSI ドライバーは、異常なノードでのみアンマウントする前にデタッチされ、**undocumented** 状態を回避するようになりました。(BZ#2049306)

- 以前は、Manila CSI Driver Operator の VolumeSnapshotClass でアノテーションが欠落していました。そのため、Manila CSI スナップショット作成者はシークレットを見つけることができず、デフォルトの VolumeSnapshotClass でスナップショットを作成できませんでした。今回の更新で問題が修正され、シークレットの名前と namespace がデフォルトの VolumeSnapshotClass に含まれるようになりました。その結果、ユーザーはデフォルトの VolumeSnapshotClass を使用して、Manila CSI Driver Operator でスナップショットを作成できるようになりました。(BZ#2057637)

- ユーザーは、Azure File の実験的な VHD 機能の利用を選択できるようになりました。これを選択するには、ユーザーはストレージクラスで **fstype** パラメーターを指定し、**--enable-vhd=true** で有効にする必要があります。**fstype** が使用され、機能が **true** に設定されていない場合、ボリュームはプロビジョニングに失敗します。
VHD 機能を使用しないようにするには、ストレージクラスから **fstype** パラメーターを削除します。(BZ#2080449)

- 以前は、vSphere に複数のシークレットが存在する場合、vSphere CSI Operator がランダムにシークレットを選択し、Operator が再起動することがありました。今回の更新により、vCenter CSI Operator に複数のシークレットがある場合に警告が表示されるようになりました。(BZ#2108473)

Web コンソール (開発者パースペクティブ)

- 以前は、ユーザーは追加および編集フォームで Git シークレットを選択解除できませんでした。その結果、リソースを再作成する必要がありました。今回の修正では、select secret オプションリストで **No Secret** を選択するオプションを追加することで、問題を解決しました。その結果、ユーザーは添付されたシークレットを簡単に選択、選択解除、または切り離すことができます。(BZ#2089221)

- OpenShift Container Platform 4.9 では、**Developer Perspective** で最小のデータまたはデータがない場合、ほとんどの監視チャートまたはグラフ (CPU 消費、メモリー使用、および帯域幅) は -1 から 1 の範囲を示します。ただし、これらの値はいずれもゼロ未満の値にすることはできません。この問題は、今後のリリースで解決される予定です。(BZ#1904106)

- 今回の更新前は、ユーザー定義の Alertmanager サービスがデプロイされた場合、OpenShift Container Platform Web コンソールの **Developer** パースペクティブでアラートを無音にすることができませんでした。これは、Web コンソールが要求を **openshift-monitoring** namespace

のプラットフォーム Alertmanager サービスに転送するためでした。今回の更新により、Web コンソールで **Developer** パースペクティブを表示してアラートを消音しようとすると、要求が正しい Alertmanager サービスに転送されるようになりました。(OCPBUGS-1789)

- 以前は、プロジェクトの開発者カタログを拡張する **Add Helm Chart Repositories** フォームに既知の問題がありました。**クイックスタート** ガイドでは、必要な namespace に **ProjectHelmChartRepository** CR を追加できると説明されていますが、これを実行するには kubeadmin からのパーミッションが必要である点については言及されていません。この問題は、**クイックスタート** に **ProjectHelmChartRepository** CR を作成するための正しい手順を記載することで解決されました。(BZ#2057306)

1.7. テクノロジープレビューの機能

現在、今回のリリースに含まれる機能にはテクノロジープレビューのものがあります。これらの実験的機能は、実稼働環境での使用を目的としていません。これらの機能に関しては、Red Hat カスタマーポータルの以下のサポート範囲を参照してください。

テクノロジープレビュー機能のサポート範囲

次の表では、機能は次のステータスでマークされています。

- **テクノロジープレビュー**
- **一般公開 (GA)**
- **利用不可**
- **非推奨**

Networking Technology Preview の機能

表1.13 ネットワーキングテクノロジープレビュートラッカー

機能	4.10	4.11	4.12
境界クロックとして設定された PTP シングル NIC ハードウェア	テクノロジープレビュー	一般公開 (GA)	一般公開 (GA)
境界クロックとして設定される PTP デュアル NIC ハードウェア	利用不可	テクノロジープレビュー	テクノロジープレビュー
境界クロックのある PTP イベント	テクノロジープレビュー	一般公開 (GA)	一般公開 (GA)
セカンダリーネットワークの Pod レベルボンディング	一般公開 (GA)	一般公開 (GA)	一般公開 (GA)
外部 DNS Operator	テクノロジープレビュー	一般公開 (GA)	一般公開 (GA)

機能	4.10	4.11	4.12
AWS Load Balancer Operator	利用不可	テクノロジープレビュー	テクノロジープレビュー
Ingress Node Firewall Operator	利用不可	利用不可	テクノロジープレビュー
特定の IP アドレスプールを使用した、ノードのサブセットから MetalLB サービスの BGP モードを使用したアドバタイズ	利用不可	テクノロジープレビュー	一般公開 (GA)
特定の IP アドレスプールを使用した、ノードのサブセットから MetalLB サービスの L2 モードを使用したアドバタイズ	利用不可	テクノロジープレビュー	テクノロジープレビュー
SR-IOV ネットワークのマルチネットワークポリシー	利用不可	利用不可	テクノロジープレビュー
インターフェイス固有の安全な sysctls リストの更新	利用不可	利用不可	テクノロジープレビュー
MT2892 Family [ConnectX-6 Dx] SR-IOV 対応	利用不可	利用不可	テクノロジープレビュー
MT2894 Family [ConnectX-6 Lx] SR-IOV 対応	利用不可	利用不可	テクノロジープレビュー
MT42822 BlueField-2 in ConnectX-6 NIC mode SR-IOV 対応	利用不可	利用不可	テクノロジープレビュー
Silicom STS Family SR-IOV 対応	利用不可	利用不可	テクノロジープレビュー
MT2892 Family [ConnectX-6 Dx] OvS Hardware Offload 対応	利用不可	利用不可	テクノロジープレビュー
MT2894 Family [ConnectX-6 Lx] OvS Hardware Offload 対応	利用不可	利用不可	テクノロジープレビュー

機能	4.10	4.11	4.12
MT42822 BlueField-2 in ConnectX-6 NIC mode OvS Hardware Offload 対応	利用不可	利用不可	テクノロジープレビュー
Bluefield-2 の DPU から NIC への切り替え	利用不可	利用不可	テクノロジープレビュー

ストレージテクノロジープレビュー機能

表1.14 ストレージテクノロジープレビュートラッカー

機能	4.10	4.11	4.12
OpenShift ビルドでの共有リソース CSI ドライバーおよびビルド CSI ボリューム	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
CSI ボリューム拡張	テクノロジープレビュー	一般公開 (GA)	一般公開 (GA)
CSI Azure File Driver Operator	テクノロジープレビュー	一般公開 (GA)	一般公開 (GA)
CSI Google Filestore Driver Operator	利用不可	利用不可	テクノロジープレビュー
CSI 自動移行 (Azure ファイル、VMware vSphere)	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
CSI 自動移行 (Azure Disk、OpenStack Cinder)	テクノロジープレビュー	一般公開 (GA)	一般公開 (GA)
CSI 自動移行 (AWS EBS、GCP ディスク)	テクノロジープレビュー	テクノロジープレビュー	一般公開 (GA)
CSI インラインの一時ボリューム	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー

機能	4.10	4.11	4.12
CSI 汎用一時ボリューム	利用不可	一般公開 (GA)	一般公開 (GA)
Shared Resource CSI Driver	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
CSI Google Filestore Driver Operator	利用不可	利用不可	テクノロジープレビュー
Local Storage Operator を使用した自動デバイス検出およびプロビジョニング	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー

インストールテクノロジープレビュー機能

表1.15 インストールテクノロジープレビュートラッカー

機能	4.10	4.11	4.12
kvc を使用したノードへのカーネルモジュールの追加	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
IBM Cloud VPC クラスター	テクノロジープレビュー	テクノロジープレビュー	一般公開 (GA)
選択可能なクラスターインベントリー	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
マルチアーキテクチャーコンピュートマシン	利用不可	テクノロジープレビュー	テクノロジープレビュー
oc-mirror CLI プラグインを使用した非接続ミラーリング	テクノロジープレビュー	一般公開 (GA)	一般公開 (GA)
RHEL の BuildConfigs で共有資格をマウントする	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
エージェントベースの OpenShift Container Platform インストーラー	利用不可	利用不可	一般公開 (GA)

機能	4.10	4.11	4.12
AWS Outposts プラットフォーム	利用不可	利用不可	テクノロジープレビュー

ノードテクノロジープレビュー機能

表1.16 ノードテクノロジープレビュートラッカー

機能	4.10	4.11	4.12
プリエンプションを実行しない優先順位クラス	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
Node Health Check Operator	テクノロジープレビュー	一般公開 (GA)	一般公開 (GA)
Linux コントロールグループバージョン 2 (cgroup v2)	利用不可	利用不可	テクノロジープレビュー
コンテナーランタイムをクロン	利用不可	利用不可	テクノロジープレビュー

マルチアーキテクチャーテクノロジーのプレビュー機能

表1.17 マルチアーキテクチャーテクノロジープレビュートラッカー

機能	4.10	4.11	4.12
x86_64 アーキテクチャーでの **kdump**	テクノロジープレビュー	一般公開 (GA)	一般公開 (GA)
arm64 アーキテクチャーでの **kdump**	利用不可	テクノロジープレビュー	テクノロジープレビュー
s390x アーキテクチャーでの **kdump**	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー

機能	4.10	4.11	4.12
ppc64le アーキテクチャーでの **kdump**	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
IBM zSystems および LinuxONE での IBM Secure Execution	利用不可	利用不可	テクノロジープレビュー

サーバーレステクノロジープレビューの機能

表1.18 サーバーレステクノロジープレビュートラッカー

機能	4.10	4.11	4.12
Serverless functions	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー

特殊なハードウェアとドライバーの有効化テクノロジープレビュー機能

表1.19 専用のハードウェアとドライバーの有効化テクノロジープレビュートラッカー

機能	4.10	4.11	4.12
ドライバーツールキット	テクノロジープレビュー	テクノロジープレビュー	一般公開 (GA)
Special Resource Operator(SRO)	テクノロジープレビュー	テクノロジープレビュー	利用不可
ハブアンドスポーククラスターのサポート	利用不可	利用不可	テクノロジープレビュー

Web コンソールのテクノロジープレビュー機能

表1.20 Web コンソールテクノロジープレビュートラッカー

機能	4.10	4.11	4.12
マルチクラスターコンソール	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー

機能	4.10	4.11	4.12
動的プラグイン	テクノロジープレビュー	テクノロジープレビュー	一般公開 (GA)

スケーラビリティとパフォーマンステクノロジープレビュー機能

表1.21 スケーラビリティとパフォーマンステクノロジープレビュートラッカー

機能	4.10	4.11	4.12
ハイパースレッディング対応の CPU マネージャーポリシー	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
Node Observability Operator	利用不可	テクノロジープレビュー	テクノロジープレビュー
factory-precaching-cli ツール	利用不可	利用不可	テクノロジープレビュー
GitOps ZTP を使用して単一ノードの OpenShift クラスターにワーカーノードを追加する	利用不可	利用不可	テクノロジープレビュー
Topology Aware Lifecycle Manager (TALM)	テクノロジープレビュー	テクノロジープレビュー	一般公開 (GA)
マウント namespace のカプセル化	利用不可	利用不可	テクノロジープレビュー

Operator テクノロジープレビュー機能

表1.22 オペレーターテクノロジープレビュートラッカー

機能	4.10	4.11	4.12
ハイブリッド Helm Operator	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー

機能	4.10	4.11	4.12
Java ベースの Operator	利用不可	テクノロジープレビュー	テクノロジープレビュー
Node Observability Operator	利用不可	利用不可	テクノロジープレビュー
ネットワーク可観測性オペレーター	利用不可	利用不可	一般公開 (GA)
プラットフォーム Operator	利用不可	利用不可	テクノロジープレビュー
RukPak	利用不可	利用不可	テクノロジープレビュー
Cert-manager Operator	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー

モニタリングテクノロジープレビュー機能

表1.23 モニタリングテクノロジープレビュートラッカー

機能	4.10	4.11	4.12
ユーザー定義プロジェクトのモニターリングのアラートルーティング	テクノロジープレビュー	一般公開 (GA)	一般公開 (GA)
プラットフォームモニターリングメトリクスに基づいたアラートルール	利用不可	テクノロジープレビュー	テクノロジープレビュー

Red Hat OpenStack Platform (RHOSP) テクノロジープレビュー機能

表1.24 RHOSP テクノロジープレビュートラッカー

機能	4.10	4.11	4.12

機能	4.10	4.11	4.12
RHOSP DCN のサポート	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
RHOSP 上のクラスターの外部クラウドプロバイダーのサポート	テクノロジープレビュー	テクノロジープレビュー	一般公開 (GA)
RHOSP 上のクラスターの OVS ハードウェアオフロード	テクノロジープレビュー	一般公開 (GA)	一般公開 (GA)

アーキテクチャーテクノロジープレビューの機能

表1.25 アーキテクチャーテクノロジープレビュートラッカー

機能	4.10	4.11	4.12
ベアメタル上の OpenShift Container Platform のホスト型コントロールプレーン	利用不可	利用不可	テクノロジープレビュー
Amazon Web Services (AWS) 上の OpenShift Container Platform のホスト型コントロールプレーン	利用不可	テクノロジープレビュー	テクノロジープレビュー

マシン管理テクノロジープレビュー機能

表1.26 マシン管理テクノロジープレビュートラッカー

機能	4.10	4.11	4.12
Cluster API によるマシンの管理	利用不可	テクノロジープレビュー	テクノロジープレビュー
Cron ジョブのタイムゾーン	利用不可	利用不可	テクノロジープレビュー
Alibaba Cloud のクラウドコントローラーマネージャー	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
Amazon Web Services のクラウドコントローラーマネージャー	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー

機能	4.10	4.11	4.12
Google Cloud Platform のクラウドコントローラーマネージャー	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
Microsoft Azure のクラウドコントローラーマネージャー	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
Red Hat OpenStack Platform (RHOSP) のクラウドコントローラーマネージャー	テクノロジープレビュー	テクノロジープレビュー	一般公開 (GA)
VMware vSphere のクラウドコントローラーマネージャー	テクノロジープレビュー	テクノロジープレビュー	テクノロジープレビュー
Custom Metrics Autoscaler Operator	利用不可	テクノロジープレビュー	テクノロジープレビュー

1.8. 既知の問題

- OpenShift Container Platform 4.1 では、匿名ユーザーは検出エンドポイントにアクセスできました。後のリリースでは、一部の検出エンドポイントは集約された API サーバーに転送されるため、このアクセスを無効にして、セキュリティーの脆弱性の可能性を減らすことができます。ただし、既存のユースケースに支障がで出ないように、認証されていないアクセスはアップグレードされたクラスターで保持されます。
  OpenShift Container Platform 4.1 から 4.12 にアップグレードされたクラスターのクラスター管理者の場合、認証されていないアクセスを無効にするか、またはこれを引き続き許可することができます。認証なしのアクセスが必要な理由が特に無い限り、無効にしてください。認証されていないアクセスを引き続き許可する場合は、それに伴ってリスクが増大することに注意してください。

警告

認証されていないアクセスに依存するアプリケーションがある場合、認証されていないアクセスを取り消すと HTTP **403** エラーが生じる可能性があります。

以下のスクリプトを使用して、検出エンドポイントへの認証されていないアクセスを無効にします。

```
## Snippet to remove unauthenticated group from all the cluster role bindings
$ for clusterrolebinding in cluster-status-binding discovery system:basic-user
```

```
system:discovery system:openshift:discovery ;
do
### Find the index of unauthenticated group in list of subjects
index=$(oc get clusterrolebinding ${clusterrolebinding} -o json | jq 'select(.subjects!=null) |
.subjects | map(.name=="system:unauthenticated") | index(true)');
### Remove the element at index from subjects array
oc patch clusterrolebinding ${clusterrolebinding} --type=json --patch "[{'op': 'remove','path':
'/subjects/$index'}]";
done
```

このスクリプトは、認証されていないサブジェクトを以下のクラスターロールバインディングから削除します。

- **cluster-status-binding**
- **discovery**
- **system:basic-user**
- **system:discovery**
- **system:openshift:discovery**

(BZ#1821771)

- 断続的に、一部のワーカーマシンが起動しないために、IBM Cloud VPC クラスターのインストールが失敗することがあります。むしろ、これらのワーカーマシンは **Provisioned** フェーズのままです。
この問題には回避策があります。最初のインストールを実行したホストから、失敗したマシンを削除し、インストールプログラムを再度実行します。

  1. マスター API サーバーの内部アプリケーションロードバランサー (ALB) のステータスが **active** であることを確認します。

     a. 次のコマンドを実行して、クラスターのインフラストラクチャー ID を特定します。

     ```
     $ oc get infrastructure/cluster -ojson | jq -r '.status.infrastructureName'
     ```

     b. クラスターの IBM Cloud アカウントにログインし、クラスターの正しいリージョンをターゲットにします。

     c. 次のコマンドを実行して、内部 ALB ステータスが **active** であることを確認します。

     ```
     $ ibmcloud is lb <cluster_ID>-kubernetes-api-private  --output json | jq -r
     '.provisioning_status'
     ```

  2. 次のコマンドを実行して、**Provisioned** フェーズにあるマシンを特定します。

     ```
     $ oc get machine -n openshift-machine-api
     ```

     出力例

     ```
     NAME                                PHASE        TYPE      REGION   ZONE       AGE
     example-public-1-x4gpn-master-0     Running      bx2-4x16  us-east  us-east-1  23h
     example-public-1-x4gpn-master-1     Running      bx2-4x16  us-east  us-east-2  23h
     ```

```
example-public-1-x4gpn-master-2        Running      bx2-4x16  us-east  us-east-3  23h
example-public-1-x4gpn-worker-1-xqzzm  Running      bx2-4x16  us-east  us-east-1
22h
example-public-1-x4gpn-worker-2-vg9w6  Provisioned  bx2-4x16  us-east  us-east-2
22h
example-public-1-x4gpn-worker-3-2f7zd  Provisioned  bx2-4x16  us-east  us-east-3
22h
```

3. 次のコマンドを実行して、失敗した各マシンを削除します。

```
$ oc delete machine <name_of_machine> -n openshift-machine-api
```

4. 削除されたワーカーマシンが置き換えられるまで待ちます。これには最大 10 分かかる場合があります。

5. 次のコマンドを実行して、新しいワーカーマシンが **Running** フェーズにあることを確認します。

```
$ oc get machine -n openshift-machine-api
```

出力例

```
NAME                                PHASE     TYPE       REGION    ZONE       AGE
example-public-1-x4gpn-master-0        Running   bx2-4x16  us-east  us-east-1  23h
example-public-1-x4gpn-master-1        Running   bx2-4x16  us-east  us-east-2  23h
example-public-1-x4gpn-master-2        Running   bx2-4x16  us-east  us-east-3  23h
example-public-1-x4gpn-worker-1-xqzzm  Running   bx2-4x16  us-east  us-east-1  23h
example-public-1-x4gpn-worker-2-mnlsz  Running   bx2-4x16  us-east  us-east-2
8m2s
example-public-1-x4gpn-worker-3-7nz4q  Running   bx2-4x16  us-east  us-east-3
7m24s
```

6. 次のコマンドを実行して、インストールを完了します。インストールプログラムを再度実行すると、クラスターの **kubeconfig** が適切に初期化されます。

```
$ ./openshift-install wait-for install-complete
```

(OCPBUGS#1327)

- **oc annotate** コマンドは、等号 (**=**) が含まれる LDAP グループ名では機能しません。これは、コマンドがアノテーション名と値の間に等号を区切り文字として使用するためです。回避策として、**oc patch** または **oc edit** を使用してアノテーションを追加します。(BZ#1917280)

- 一部のイメージインデックスに古いイメージが含まれているため、**oc adm catalog mirror** および **oc image mirror** を実行すると、**error: unable to retrieve source image** エラーが発生する場合があります。一時的な回避策として、**--skip-missing** オプションを使用してエラーを回避し、イメージインデックスのダウンロードを続行できます。詳細は、Service Mesh Operator mirroring failed を参照してください。

- RHOSP 上の OpenShift Container Platform で egress IP アドレス機能を使用する場合、Floating IP アドレスを予約ポートに割り当てて、egress トラフィックの予測可能な SNAT アドレスを持つことができます。Floating IP アドレスの関連付けは、OpenShift Container Platform クラスターをインストールしたユーザーにより作成される必要があります。そうでない場合、

権限が不十分なために、egress IP アドレスの削除または移動が無期限にハングアップします。この問題が発生した場合、問題を解決するには、十分な権限を持つユーザーが Floating IP アドレスの関連付けを手動で設定解除する必要があります。(OCPBUGS-4902)

- Prism Central 2022.x で 4096 ビットの証明書を使用すると、Nutanix のインストールに失敗するという既知の問題があります。代わりに、2048 ビットの証明書を使用します。(KCS)

- 双方向転送検出 (BFD) プロファイルを削除し、ボーダーゲートウェイプロトコル (BGP) ピアリソースに追加された **bfdProfile** を削除しても、BFD は無効になりません。代わりに、BGP ピアはデフォルトの BFD プロファイルの使用を開始します。BGP ピアリソースから BFD をディセーブルにするには、BGP ピア設定を削除し、BFD プロファイルなしで再作成します。(BZ#2050824)

- 未解決のメタデータ API の問題により、ベアメタルワーカーを使用するクラスターを RHOSP 16.1 にインストールすることはできません。RHOSP 16.2 上のクラスターは、この問題の影響を受けません。(BZ#2033953)

- **loadBalancerSourceRanges** 属性は、RHOSP で実行され、OVN Octavia プロバイダーを使用するクラスター内のロードバランサータイプのサービスではサポートされていないため、無視されます。この問題に対する回避策はありません。(OCPBUGS-2789)

- カタログソースの更新後、OLM がサブスクリプションステータスを更新するのに時間がかかります。これは、Topology Aware Lifecycle Manager (TALM) が修正が必要かどうかを判断したときに、サブスクリプションポリシーのステータスが準拠として表示され続ける可能性があることを意味します。その結果、サブスクリプションポリシーで指定された Operator はアップグレードされません。回避策として、次のように、カタログソースポリシーの **spec** セクションに **status** フィールドを含めます。

```
metadata:
  name: redhat-operators-disconnected
spec:
  displayName: disconnected-redhat-operators
  image: registry.example.com:5000/disconnected-redhat-operators/disconnected-redhat-operator-index:v4.11
status:
  connectionState:
    lastObservedState: READY
```

これにより、OLM が新しいインデックスイメージを取得して Pod を準備するまでの遅延が軽減され、カタログソースポリシーの修正が完了してからサブスクリプションステータスが更新されるまでの時間が短縮されます。問題が解決せず、サブスクリプションポリシーステータスの更新がまだ遅れている場合は、同じサブスクリプションポリシーで別の **ClusterGroupUpdate** CR を適用するか、別の名前で同じ **ClusterGroupUpdate** CR を適用できます。(OCPBUGS-2813)

- **ClusterGroupUpdate** CR の開始時に、選択したすべてのクラスターが準拠している場合、TALM はポリシーの修正をスキップします。同じ **ClusterGroupUpdate** CR 内のカタログソースポリシーとサブスクリプションポリシーが変更された Operator の更新は完了しません。サブスクリプションポリシーは、カタログソースの変更が適用されるまで準拠の状態が続くためスキップされます。回避策として、次の例のように、**common-subscription** ポリシーの 1 つの CR に次の変更を追加します。

```
metadata.annotations.upgrade: "1"
```

これにより、**ClusterGroupUpdate** CR の開始前にポリシーが非準拠になります。(OCPBUGS-2812)

- シングルノード OpenShift インスタンスでは、ノードをドレインして実行中のすべての Pod を削除せずに再起動すると、ワークロードコンテナーのリカバリーで問題が発生する可能性があります。再起動後、すべてのデバイスプラグインの準備が整う前にワークロードが再開され、その結果、リソースが利用できなくなったり、ワークロードが間違った NUMA ノードで実行されたりします。回避策としては、再起動リカバリーの手順中にすべてのデバイスプラグインが再登録された時点でワークロード Pod を再起動します。(OCPBUGS-2180)

- 現在のデフォルト **dataset_comparison** は **ieee1588** です。推奨される dataset_comparison は **G.8275.x** です。これは、OpenShift Container Platform の今後のバージョンで修正される予定です。短期的には、推奨される **dataset_comparison** を含むように ptp 設定を手動で更新できます。(OCPBUGS-2336)

- デフォルトの **step_threshold** は 0.0 です。推奨される **step_threshold** は 2.0 です。これは、OpenShift Container Platform の今後のバージョンで修正される予定です。短期的には、推奨される **step_threshold** を含むように ptp 設定を手動で更新できます。(OCPBUGS-3005)

- ACM がデプロイされたマルチクラスター環境で metal3 サービスがハブクラスターでのみ実行されている場合、**BMCEventSubscription** CR はスポーククラスターの Redfish サブスクリプションを作成できません。回避策としては、たとえば次のコマンドを実行し、Redfish API を直接呼び出してサブスクリプションを作成します。

```
curl -X POST -i --insecure -u "<BMC_username>:<BMC_password>"
https://<BMC_IP>/redfish/v1/EventService/Subscriptions \
    -H 'Content-Type: application/json' \
    --data-raw '{
    "Protocol": "Redfish",
    "Context": "any string is valid",
    "Destination": "https://hw-event-proxy-openshift-bare-metal-
events.apps.example.com/webhook",
    "EventTypes": ["Alert"]
    }'
```

  201 Created レスポンスと、Redfish イベントサブスクリプションが正常に作成されたことを示す **Location: /redfish/v1/EventService/Subscriptions/<sub_id>** を含むヘッダーを受け取るはずです。(OCPBUGSM-43707)

- GitOps ZTP パイプラインを使用して、切断された環境に単一ノードの OpenShift クラスターをインストールする場合、クラスターに 2 つの **CatalogSource** CR が適用されている必要があります。ノードを複数回再起動すると、**CatalogSource** CR の 1 つが削除されます。回避策として、カタログソースのデフォルト名 (**certified-operators** や **redhat-operators** など) を変更できます。(OCPBUGSM-46245)

- クラスターのアップグレードを実行するために使用されるサブスクリプションポリシーで無効なサブスクリプションチャネルが指定されている場合、Topology Aware Lifecycle Manager は、ポリシーが適用された直後にアップグレードの成功を示します。これは、**Subscription** の状態が **AtLatestKnown** のままであるためです。(OCPBUGSM-43618)

- クラスター内の複数のノードに適用すると、**SiteConfig** ディスクパーティションの定義が失敗します。**SiteConfig** CR を使用してコンパクトクラスターをプロビジョニングする場合は、複数のノードで有効な **diskPartition** 設定を作成すると、Kustomize プラグインエラーで失敗します。(OCPBUGSM-44403)

- セキュアブートが現在無効になっていて、ZTP を使用して有効にしようとすると、クラスターのインストールが開始しません。ZTP を使用してセキュアブートを有効にすると、仮想 CD が接続される前にブートオプションが設定されます。したがって、既存のハードディスクからの

最初の起動では、セキュアブートが有効になります。システムが CD から起動しないため、クラスターのインストールが停止します。(OCPBUGSM-45085)

- イメージをディスクに書き込んだ後、仮想メディアが iDRAC コンソールで ISO を切断しない場合は、Red Hat Advanced Cluster Management (RHACM) を使用して、Dell PowerEdge R640 サーバーでのスポーククラスターの導入がブロックされます。回避策として、iDRAC コンソールの仮想メディアタブから ISO を手動で切断します。(OCPBUGSM-45884)

- 高解像度タイマーに依存してスレッドをウェイクアップする低レイテンシーアプリケーションでは、想定よりも長いウェイクアップレイテンシーが発生する可能性があります。想定されるウェイクアップレイテンシーは 20us 未満ですが、cylictest ツールを長時間 (24 時間以上) 実行すると、これを超えるレイテンシーが発生することがあります。テストでは、サンプルの 99.999999% 超でウェイクアップレイテンシーが 20us 未満であることが確認されています。(RHELPLAN-138733)

- Intel の Chapman Beach NIC を分岐 PCIe スロットに取り付けて、両方のポートが見えるようにする必要があります。RHEL 8.6 の現在の devlink ツールにも制限があり、分岐 PCIe スロットで 2 つのポートを設定できません。(RHELPLAN-142458)

- ポートがダウンしたときに SR-IOV VF を無効にすると、Intel NIC で 3 - 4 秒の遅延が発生する可能性があります。(RHELPLAN-126931)

- Intel NIC を使用している場合、SR-IOV VF に IPV6 アドレスが割り当てられると、IPV6 トラフィックが停止します。(RHELPLAN-137741)

- VLAN ストリップオフロードを使用する場合、オフロードフラグ (**ol_flag**) が iavf ドライバーで一貫して正しく設定されません。(RHELPLAN-141240)

- ice ドライバーで設定変更中に割り当てが失敗すると、デッドロックが発生する可能性があります。(RHELPLAN-130855)

- Intel NIC を使用している場合、SR-IOV VF は間違った MAC アドレスで GARP パケットを送信します。(RHELPLAN-140971)

- GitOps ZTP メソッドを使用してクラスターを管理し、インストールが完了していないクラスターを削除すると、ハブクラスター上のクラスター namespace のクリーンアップが無期限にハングアップすることがあります。namespace の削除を完了するには、クラスター namespace の 2 つの CR から、**baremetalhost.metal3.io** ファイナライザーを削除します。

  1. BareMetalHost CR **.spec.bmc.credentialsName** が指すシークレットからファイナライザーを削除します。

  2. **BareMetalHost** CR からファイナライザーを削除します。これらのファイナライザーが削除されると、 namespace の終了は数秒以内に完了します。(OCPBUGS-3029)

- OCP 4.12 に追加された UDP GRO を有効にする新機能より、すべての veth デバイスも使用可能な CPU ごとに 1 つの RX キューを持つようになります (以前は、各 veth に 1 つのキューがありました)。これらのキューは OVN によって動的に設定され、レイテンシー調整とこのキューの作成は同期されません。レイテンシー調整ロジックは、veth NIC 作成イベントをモニタリングし、すべてのキューが適切に作成される前に、RPS キュー CPU マスクの設定を開始します。これは、一部の RPS キューマスクが設定されないことを意味します。すべての NIC キューが適切に設定されるわけではないため、タイミングに敏感な CPU を使用して他のコンテナー内のサービスと通信するリアルタイムアプリケーションでは、レイテンシースパイクが発生する可能性があります。カーネルネットワークスタックを使用しないアプリケーションは影響を受けません。(OCPBUGS-4194)

- Platform Operator および RukPak の既知の問題:

- プラットフォーム Operator を削除すると、基盤となるリソースが連鎖的に削除されます。このカスケード削除ロジックは、Operator Lifecycle Manager ベース (OLM) Operator のバンドル形式で定義されているリソースのみを削除できます。プラットフォーム Operator がそのバンドル形式の外で定義されたリソースを作成する場合、プラットフォーム Operator はこのクリーンアップインタラクションを処理する責任があります。この動作は、cert-manager Operator をプラットフォーム Operator としてインストールしてから削除した場合に確認できます。予想される動作は、cert-manager Operator が作成した namespace が取り残されることです。

- プラットフォーム Operators マネージャーには、管理しているクラスタースコープの **BundleDeployment** リソースの現在の状態と望ましい状態を比較するロジックがありません。これにより、十分なロールベースのアクセス制御 (RBAC) を持つユーザーがその基礎となる **BundleDeployment** リソースを手動で変更する可能性が残り、ユーザーが自分のアクセス許可を **cluster-admin** ロールにエスカレートできる状況につながる可能性があります。デフォルトでは、このリソースへのアクセスを明示的にアクセスを必要とする少数のユーザーに制限する必要があります。このテクノロジープレビューリリース中に **BundleDeployment** リソースでサポートされる唯一のクライアントは、プラットフォーム Operators マネージャーコンポーネントです。

- OLM の Marketplace コンポーネントは、無効にできるオプションのクラスター機能です。プラットフォーム Operator は現在、Marketplace コンポーネントによって管理されている **redhat-operators** カタログソースからのみ供給されているため、これはテクノロジープレビューリリース中に影響します。回避策として、クラスター管理者はこのカタログソースを手動で作成できます。

- RukPak プロビジョナーの実装には、管理しているリソースの正常性や状態を検査する機能がありません。これは、生成された **BundleDeployment** リソースの状態を、それを所有する **PlatformOperator** リソースに提示することに影響します。**レジストリー + v1** バンドルにクラスターに正常に適用できるマニフェストが含まれているが、存在しないイメージを参照する **Deployment** オブジェクトなど、実行時に失敗する場合、結果は成功のステータスであり、個々の **PlatformOperator** および **BundleDeployment** リソースに反映されます。

- クラスターの作成前に **PlatformOperator** リソースを設定するクラスター管理者は、既存のクラスターを活用したり、文書化された例に頼ったりしないと、目的のパッケージ名を簡単に判断できません。現在、個別に設定された **PlatformOperator** リソースがクラスターに正常にロールアウトできることを保証する検証ロジックはありません。

- oc-mirror CLI プラグインでテクノロジープレビュー OCI 機能を使用すると、イメージセット設定ファイルで指定されたものだけをフィルタリングするのではなく、ミラーリングされたカタログにすべての Operator バンドルが埋め込まれます。(OCPBUGS-5085)

- 現在、エージェントベースの OpenShift Container Platform インストーラーを実行して、ISO イメージの生成に以前のリリースが使用されているディレクトリーから ISO イメージを生成すると既知の問題が発生することがあります。リリースバージョンが一致しないというエラーメッセージが表示されます。回避策としては、新しいディレクトリーを作成して使用します。(OCPBUGS#5159)

- **install-config.yaml** ファイルで定義されたケイパビリティーは、エージェントベースの OpenShift Container Platform インストールでは適用されません。現在、回避策はありません。(OCPBUGS#5129)

- OVN ドライバーで作成された RHOSP 上の完全実装のロードバランサーには、作成保留ステータスでスタックしているプールが含まれている可能性があります。この問題により、RHOSP 上にデプロイされたクラスターで問題が発生する可能性があります。この問題を解決するには、RHOSP パッケージを更新します。(BZ#2042976)

- RHOSP でロードバランサーメンバーを一括更新すると、**PUT** リクエストへのレスポンスとして 500 コードが返される場合があります。この問題により、RHOSP 上にデプロイされたクラスターで問題が発生する可能性があります。この問題を解決するには、RHOSP パッケージを更新します。(BZ#2100135)

- 外部クラウドプロバイダーを使用するクラスターは、ローテーション後に更新された認証情報の取得に失敗する可能性があります。その場合、次のプラットフォームが影響を受けます。

  - Alibaba Cloud
  - IBM Cloud VPC
  - IBM Power
  - OpenShift Virtualization
  - RHOSP

  回避策として、次のコマンドを実行して **openshift-cloud-controller-manager** Pod を再起動します。

  ```
  $ oc delete pods --all -n openshift-cloud-controller-manager
  ```

  (OCPBUGS-5036)

- **cloud-provider-openstack** が API を使用して OVN ロードバランサー上にヘルスモニターを作成し、完全実装のロードバランサーを作成しようとすると、既知の問題が発生します。これらのヘルスモニターは、**PENDING_CREATE** ステータスでスタックします。削除後、関連するロードバランサーは **PENDING_UPDATE** ステータスでスタックします。回避策はありません。(BZ#2143732)

- 既知の問題があるため、RHOSP 上で実行されるクラスターでステートフル IPv6 ネットワークを使用するには、ワーカーノード のカーネル引数に **ip=dhcp,dhcpv6** を含める必要があります。(OCPBUGS-2104)

- 仮想機能 (VF) がすでに存在する場合、Physical Fundtion (PF) で macvlan を作成することはできません。この問題は、Intel E810 NIC に影響します。(BZ#2120585)

- 現在、IPv4 OpenShift Container Platform クラスターで IPv6 アドレスとルートを手動で設定すると、既知の問題が発生します。デュアルスタッククラスターに変換すると、新しく作成された Pod は **ContainerCreating** ステータスのままになります。現在、回避策はありません。この問題は、今後の OpenShift Container Platform リリースで対処される予定です。(OCPBUGS-4411)

- IBM Public Cloud にインストールされた OVN クラスターに 60 を超えるワーカーノードがある場合、2000 以上のサービスとルートオブジェクトを同時に作成すると、同時に作成された Pod が **ContainerCreating** ステータスのままになる可能性があります。この問題が発生した場合、**oc describe pod <podname>** コマンドを入力すると、**FailedCreatePodSandBox…failed to configure pod interface: timed out waiting for OVS port binding (ovn-installed)** の警告とともにイベントが表示されます。現在、この問題に対する回避策はありません。(OCPBUGS-3470)

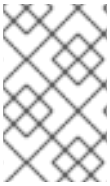
- OVN-Kubernetes ネットワークプロバイダーを使用するクラスターでコントロールプレーンマシンを交換すると、交換したマシンで OVN-Kubernetes に関連する Pod が起動しない場合があります。この状態が発生すると、新しいマシンにネットワーク接続がないため、etcd が古いマシンを置き換えることができなくなります。その結果、クラスターはその状態で停止し、パ

フォーマンスが低下する可能性があります。この動作は、手動またはコントロールプレーンマシンセットによってコントロールプレーンが置き換えられた場合に発生する可能性があります。
現在、この問題が発生した場合の解決策はありません。この問題を回避するため、クラスターが OVN-Kubernetes ネットワークプロバイダーを使用している場合は コントロールプレーンマシンセットを無効化 し、コントロールプレーンマシンを手動で置き換えないいようにしてください。(OCPBUGS-5306)

1.9. エラータの非同期更新

OpenShift Container Platform 4.12 のセキュリティー、バグ修正、拡張機能の更新は、Red Hat Network 経由で非同期エラータとして発表されます。OpenShift Container Platform 4.12 のすべてのエラータは Red Hat カスタマーポータルから入手できます。非同期エラータについては、OpenShift Container Platform ライフサイクル を参照してください。

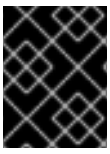
Red Hat カスタマーポータルのユーザーは、Red Hat Subscription Management (RHSM) のアカウント設定でエラータの通知を有効にすることができます。エラータ通知を有効にすると、登録されたシステムに関連するエラータが新たに発表されるたびに、メールで通知が送信されます。



注記

OpenShift Container Platform のエラータ通知メールを生成させるには、Red Hat カスタマーポータルのユーザーアカウントでシステムが登録されており、OpenShift Container Platform エンタイトルメントを使用している必要があります。

以下のセクションは、これからも継続して更新され、今後の OpenShift Container Platform 4.12 バージョンの非同期リリースで発表されるエラータの拡張機能およびバグ修正に関する情報を追加していきます。たとえば、OpenShift Container Platform 4.12.z などのバージョン付けされた非同期リリースについてはサブセクションで説明します。さらに、エラータの本文がアドバイザリーで指定されたスペースに収まらないリリースについては、詳細についてその後のサブセクションで説明します。



重要

OpenShift Container Platform のいずれのバージョンについても、クラスターの更新 に関する指示には必ず目を通してください。

1.9.1. RHSA-2022:7399 - OpenShift Container Platform 4.12.0 イメージリリース、バグ修正およびセキュリティー更新アドバイザリー

発行日: 2023-01-17

セキュリティー更新を含む OpenShift Container Platform リリース 4.12.0 が利用可能になりました。更新に含まれるバグ修正のリストは、RHSA-2022:7399 アドバイザリーに記載されています。この更新に含まれる RPM パッケージは、RHSA-2022:7398 アドバイザリーで提供されています。

このアドバイザリーでは、このリリースのすべてのコンテナーイメージに関する説明は除外されています。このリリースのコンテナーイメージに関する情報については、以下の記事を参照してください。

以下のコマンドを実行して、本リリースでコンテナーイメージを表示できます。

```
$ oc adm release info 4.12.0 --pullspecs
```

1.9.1.1. 機能

1.9.1.1.1. セカンダリーネットワークの Pod レベルのボンディングの一般提供

今回の更新により、**Pod レベルのボンディングの使用** が一般提供されるようになりました。

1.9.2. RHSA-2023:0449 - OpenShift Container Platform 4.12.1 のバグ修正とセキュリティー更新

発行日: 2022-01-30

セキュリティー更新を含む OpenShift Container Platform リリース 4.12.1 が利用可能になりました。更新に含まれるバグ修正のリストは、RHSA-2023:0449 アドバイザリーに記載されています。更新に含まれる RPM パッケージは、RHBA-2023:0448 アドバイザリーによって提供されます。

以下のコマンドを実行して、本リリースでコンテナーイメージを表示できます。

```
$ oc adm release info 4.12.1 --pullspecs
```

1.9.2.1. バグ修正

- 以前のリリースでは、OpenStack クラウドプロバイダーのチェックが間違っていたため、すべての Octavia ロードバランサーが作成されたときに、ロードバランサーに外部 IP アドレスが設定されていました。これにより、ロードバランサーの処理時間が長くなりました。今回の更新では、ロードバランサーは引き続き順次作成され、外部 IP アドレスは 1 つずつ設定されます。(OCPBUGS-5403)

- 以前のリリースでは、**cluster-image-registry-operator** は、Swift に到達できなかった場合、デフォルトで永続ボリューム要求 (PVC) を使用していました。今回の更新により、Red Hat OpenStack Platform (RHOSP) API への接続の失敗またはその他の偶発的な失敗により、**cluster-image-registry-operator** がプローブを再試行するようになりました。再試行中に、RHOSP カタログが問題なく見つかり、オブジェクトストレージが含まれていない場合、または RHOSP カタログがあり、現在のユーザーにコンテナーをリストするパーミッションがない場合に、デフォルトで PVC が使用されます。(OCPBUGS-5154)

1.9.2.2. 更新

既存の OpenShift Container Platform 4.12 クラスターをこの最新リリースに更新する方法については CLI を使用したクラスターの更新 を参照してください。

1.9.3. RHSA-2023:0569 - OpenShift Container Platform 4.12.2 のバグ修正とセキュリティー更新

発行日: 2023-02-07

セキュリティー更新を含む OpenShift Container Platform リリース 4.12.2 が利用可能になりました。更新に含まれるバグ修正のリストは、RHSA-2023:0569 アドバイザリーに記載されています。更新に含まれる RPM パッケージは、RHBA-2023:0568 アドバイザリーによって提供されます。

以下のコマンドを実行して、本リリースでコンテナーイメージを表示できます。

```
$ oc adm release info 4.12.2 --pullspecs
```

1.9.3.1. 更新

既存の OpenShift Container Platform 4.12 クラスターをこの最新リリースに更新する方法については CLI を使用したクラスターの更新 を参照してください。

1.9.4. RHSA-2023:0728 - OpenShift Container Platform 4.12.3 のバグ修正とセキュリティー更新

発行日: 2023-02-16

セキュリティー更新を含む OpenShift Container Platform リリース 4.12.3 が利用可能になりました。更新に含まれるバグ修正のリストは、RHSA-2023:0728 アドバイザリーに記載されています。この更新に含まれる RPM パッケージは、RHSA-2023:0727 アドバイザリーで提供されています。

以下のコマンドを実行して、本リリースでコンテナーイメージを表示できます。

```
$ oc adm release info 4.12.3 --pullspecs
```

1.9.4.1. バグ修正

- 以前は、OVN-Kubernetes ネットワークプロバイダーを使用するクラスターでコントロールプレーンマシンを交換すると、交換したマシンで OVN-Kubernetes に関連する Pod が起動せず、etcd によって古いマシンの交換が許可されないことがありました。今回の更新により、OVN-Kubernetes に関連する Pod が、期待どおりに交換したマシンで起動するようになりました (**OCPBUGS-6494**)。

1.9.4.2. 更新

既存の OpenShift Container Platform 4.12 クラスターをこの最新リリースに更新する方法については CLI を使用したクラスターの更新 を参照してください。

1.9.5. RHSA-2023:0769 - OpenShift Container Platform 4.12.4 のバグ修正とセキュリティー更新

発行日: 2023-02-20

セキュリティー更新を含む OpenShift Container Platform リリース 4.12.4 が利用可能になりました。更新に含まれるバグ修正のリストは、RHSA-2023:0769 アドバイザリーに記載されています。更新に含まれる RPM パッケージは、RHBA-2023:0768 アドバイザリーによって提供されます。

以下のコマンドを実行して、本リリースでコンテナーイメージを表示できます。

```
$ oc adm release info 4.12.4 --pullspecs
```

1.9.5.1. 更新

既存の OpenShift Container Platform 4.12 クラスターをこの最新リリースに更新する方法については CLI を使用したクラスターの更新 を参照してください。

1.9.6. RHSA-2023:0890 - OpenShift Container Platform 4.12.5 のバグ修正とセキュリティー更新

発行日: 2023-02-28

セキュリティー更新を含む OpenShift Container Platform リリース 4.12.5 が利用可能になりました。更新に含まれるバグ修正のリストは、RHSA-2023:0890 アドバイザリーに記載されています。更新に含まれる RPM パッケージは RHBA-2023:0889 アドバイザリーによって提供されます。

以下のコマンドを実行して、本リリースでコンテナーイメージを表示できます。

```
$ oc adm release info 4.12.5 --pullspecs
```

1.9.6.1. バグ修正

- 以前は、リポジトリーリストで、ステータスが **Succeeded** または **Failed** の場合にのみ **PipelineRuns** が表示され、ステータスが **Running** の場合は表示されませんでした。今回の修正により、**PipelineRun** がトリガーされたときに、リポジトリーリストに **Running** ステータスで表示されるようになりました。(OCPBUGS-6816)

- 以前は、**Secret** を作成するときに、**Start Pipeline** モデルが無効な JSON 値を作成しました。その結果、Secret が使用できなくなり、**PipelineRun が** 失敗する可能性がありました。今回の修正により、**Start Pipeline** モデルが **Secret** の有効な JSON 値を作成するようになりました。パイプラインの起動時に有効な Secret を作成できるようになりました。(OCPBUGS-6671)

- 以前は、**BindableKinds** リソースにステータスがない場合、Web コンソールがクラッシュし、ループ内で同じデータを取得して表示していました。今回の修正により、**BindableKinds** リソースのステータス配列を **[]** に設定できるようになり、ステータスフィールドがなくても存在すると見なされるようになりました。その結果、Web ブラウザーやアプリケーションがクラッシュすることがなくなりました。(OCPBUGS-4072)

- 以前は、OpenShift Container Platform から Knative (**kn**) サービスを削除するときに、関連する Webhook **<kn-service-name>-github-webhook-secret** が削除されませんでした。今回の修正により、関連するすべての Webhook シークレットが削除されるようになりました。これで、削除されたサービスと同じ名前の Knative (**kn**) サービスを作成できるようになりました。(OCPBUGS-7437)

1.9.6.2. 更新

既存の OpenShift Container Platform 4.12 クラスターをこの最新リリースに更新する方法については CLI を使用したクラスターの更新 を参照してください。

1.9.7. RHSA-2023:1034 - OpenShift Container Platform 4.12.6 のバグ修正とセキュリティー更新

発行日: 2023-03-07

セキュリティー更新を含む OpenShift Container Platform リリース 4.12.6 が利用可能になりました。更新に含まれるバグ修正のリストは、RHBA-2023:1034 アドバイザリーに記載されています。更新に含まれる RPM パッケージは、RHSA-2023:1033 アドバイザリーによって提供されます。

以下のコマンドを実行して、本リリースでコンテナーイメージを表示できます。

```
$ oc adm release info 4.12.6 --pullspecs
```

1.9.7.1. 更新

既存の OpenShift Container Platform 4.12 クラスターをこの最新リリースに更新する方法については CLI を使用したクラスターの更新 を参照してください。

1.9.8. RHBA-2023:1163 - OpenShift Container Platform 4.12.7 バグ修正の更新

発行日: 2023-03-13

OpenShift Container Platform リリース 4.12.7 が利用可能になりました。更新に含まれるバグ修正のリストは、RHBA-2023:1163 アドバイザリーに記載されています。更新に含まれる RPM パッケージは、RHBA-2023:1162 アドバイザリーによって提供されます。

以下のコマンドを実行して、本リリースでコンテナーイメージを表示できます。

```
$ oc adm release info 4.12.7 --pullspecs
```

1.9.8.1. 更新

既存の OpenShift Container Platform 4.12 クラスターをこの最新リリースに更新する方法については CLI を使用したクラスターの更新 を参照してください。

1.9.9. RHBA-2023:1269 - OpenShift Container Platform 4.12.8 のバグ修正とセキュリティー更新

発行日: 2023-03-21

セキュリティー更新を含む OpenShift Container Platform リリース 4.12.8 が利用可能になりました。更新に含まれるバグ修正のリストは、RHBA-2023:1269 アドバイザリーに記載されています。更新に含まれる RPM パッケージは、RHSA-2023:1268 アドバイザリーによって提供されます。

以下のコマンドを実行して、本リリースでコンテナーイメージを表示できます。

```
$ oc adm release info 4.12.8 --pullspecs
```

1.9.9.1. 更新

既存の OpenShift Container Platform 4.12 クラスターをこの最新リリースに更新する方法については CLI を使用したクラスターの更新 を参照してください。

1.9.10. RHSA-2023:1409 - OpenShift Container Platform 4.12.9 バグ修正およびセキュリティー更新

発行日: 2023-03-27

セキュリティー更新を含む OpenShift Container Platform リリース 4.12.9 が利用可能になりました。この更新に含まれるバグ修正の一覧は、RHSA-2023:1409 アドバイザリーにまとめられています。この更新に含まれる RPM パッケージは、RHSA-2023:1408 アドバイザリーで提供されています。

以下のコマンドを実行して、本リリースでコンテナーイメージを表示できます。

```
$ oc adm release info 4.12.9 --pullspecs
```

1.9.10.1. バグ修正

- 以前のバージョンでは、テクノロジープレビュー機能ゲートを有効にしなかった場合、検証により、ユーザーが GCP クラスターを共有 VPC にインストールできませんでした。したがって、テクノロジープレビュー機能ゲートを有効にせずに、クラスターを共有 VPC にインストー

ルできます。このリリースでは、機能ゲート検証が 4.12 に追加されたため、**featureSet: TechPreviewNoUpgrade** を有効にして、GCP クラスターを共有 VPC にインストールする必要があります。(OCPBUGS-7469)

- 以前は、移行の完了前に MTU 移行設定がクリーンアップされ、移行が失敗することがありました。このリリースでは、移行中に MTU の移行が保持され、移行が正常に完了するようになりました。(OCPBUGS-7445)

1.9.10.2. 更新

既存の OpenShift Container Platform 4.12 クラスターをこの最新リリースに更新する方法については CLI を使用したクラスターの更新 を参照してください。