OpenShift Container Platform 4.4

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: 2023-04-01

OpenShift Container Platform 4.4 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

Enter your first name here. Enter your surname here.
Enter your organisation's name here. Enter your organisational division here.
Enter your email address here.

法律上の通知

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 4.4 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 および Kubernetes にビルドされる OpenShift Container Platform は、エンタープライズレベルの最新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

Red Hat OpenShift Container Platform (RHBA-2020:0581) をご利用いただけるようになりました。本リリースでは、CRI-O ランタイムで Kubernetes 1.17 を使用します。以下では、OpenShift Container Platform 4.4 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。

Red Hat は OpenShift Container Platform 4.4.0 を GA バージョンとしてリリースせず、OpenShift Container Platform 4.4.3 を GA バージョンとしてリリースしています。

OpenShift Container Platform 4.4 クラスターは https://cloud.redhat.com/openshift でご利用いただけます。OpenShift Container Platform 向けの Red Hat OpenShift Cluster Manager アプリケーションを使って、OpenShift Container Platform 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 4.4 は、Red Hat Enterprise Linux 7.6 以降、および Red Hat Enterprise Linux CoreOS (RHCOS) 4.4 でサポートされます。

コントロールプレーン (マスターマシンとしても知られている) には RHCOS を使用する必要があり、コンピュートマシン (ワーカーマシンとしても知られている) には RHCOS または Red Hat Enterprise Linux 7.6 以降のいずれかを使用できます。

重要

コンピュートマシン用にサポートされているのは Red Hat Enterprise Linux バージョン 7.6 以降であるため、Red Hat Enterprise Linux コンピュートマシンをバージョン 8 にアップグレードすることはできません。

OpenShift Container Platform 4.4 のリリースでは、バージョン 4.1 のライフサイクルは終了します。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. 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。

1.2.1. インストールおよびアップグレード

1.2.1.1. ユーザーによって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 Microsoft Azure へのクラスターのインストール

OpenShift Container Platform 4.4 では、ユーザーによって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て Azure にクラスターをインストールするためのサポートが導入されました。Azure でユーザーによってプロビジョニングされるインフラストラクチャーを実行すると、規制、セキュリティー、運用上の制御などの環境に必要となる可能性のあるカスタマイズを使用できます。

デプロイメントプロセスを支援する Red Hat 提供の Azure Resource Manager (ARM) テンプレートのサンプルを組み込むか、または独自のテンプレートを作成することができます。他の方法を使用して必要なリソースを作成することもできます。ARM テンプレートはサンプルとしてのみ提供されます。

詳細は、ARM テンプレートを使用したクラスターの Azure へのインストール について参照してください。

1.2.1.2. 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 Red Hat Virtualization へのクラスターのインストール

OpenShift Container Platform 4.4 では、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て Red Hat Virtualization (RHV) 環境にクラスターをインストールするためのサポートが導入されました。

詳細は、RHV へのクラスターのクィックインストール について参照してください。

1.2.1.3. ユーザーによって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 OpenStack へのクラスターのインストール

OpenShift Container Platform 4.4 では、提供するインフラストラクチャーで実行される Red Hat OpenStack Platform (RHOSP) へのクラスターのインストールについてのサポートが導入されました。独自のインフラストラクチャーを使用することで、クラスターを既存のインフラストラクチャーおよび変更と統合できます。たとえば、Nova サーバー、Neutron ポート、セキュリティーグループなどの RHOSP リソースをすべて作成する必要があります。Red Hat では、デプロイメントプロセスを支援する Ansible Playbook を提供しています。

独自のインフラストラクチャーを使用して、クラスターを Kuryr を使って RHOSP にインストールすることもできます。

詳細は、独自のインフラストラクチャーでのクラスターの OpenStack へのインストール、または 独自のインフラストラクチャーでのクラスターの Kuryr を使用した OpenStack へのインストール について参照してください。

1.2.1.4. OpenStack へのクラスターのインストールに Swift オブジェクトストレージサービスが不要になる

バージョン 4.4 以降、OpenShift Container Platform では Swift オブジェクトストレージサービスがインストールされている RHOSP クラウドで不要になりました。Swift が OpenShift Container Platform インストールで利用不可である場合、インストーラーは Cinder ブロックストレージと Glance イメージレジストリーサービスをその代わりに使用します。

詳細は、独自のインフラストラクチャーでのクラスターの OpenStack へのインストール について参照してください。

1.2.1.5. OpenStack にインストールされたクラスターが自己署名証明書をサポート

OpenShift Container Platform 4.4 は、認可に自己署名証明書を使用する RHOSP クラウドにインストールできるようになりました。

詳細は、独自のインフラストラクチャーでのクラスターの OpenStack へのインストール について参照してください。

1.2.1.6. OpenStack が sha256 チェックサムをチェックして RHCOS イメージを検証する

RHOSP では、インストーラーが Red Hat Enterprise Linux CoreOS (RHCOS) イメージのチェックサムの自動検証を実行するようになりました。

1.2.1.7. Kuryr を使用する OpenStack での OVN 負荷分散による east-west トラフィックのサポート

RHOSP 16 で Kuryr を使用する OpenShift Container Platform インストールでは、Amphora ドライバーの代わりに OVN 負荷分散プロバイダードライバーを使用できるようになりました。OVN および OVN Octavia ドライバーが環境にある場合、OVN は自動的に使用されます。その結果、ロードバランサーのパフォーマンスおよびリソースの使用率が改善されます。各サービスのロードバランサー仮想マシンの必要もなくなります。

1.2.1.8. 4.4 リリースでのアップグレードチャネルの使用

最新の OpenShift Container Platform 4.3.z リリースから 4.4 へのアップグレードは、**fast-4.4** チャネルに切り替えられたクラスターを対象に GA に利用可能になります。**fast-4.4** チャネルでの早期に導入したユーザーからの Telemetry データはモニターされ、アップグレードが **stable-4.4** チャネルにプロモートされる際に通知されます。このモニタリングについては、広範囲に及ぶエンタープライズレベルのテストの範囲を超えた対象外の機能であるため、これには数週間の時間がかかる場合があります。

1.2.2. セキュリティー

1.2.2.1. バインドされたサービスアカウントトークンのサポート

OpenShift Container Platform 4.4 では、AWS IAM などのクラウドプロバイダーのアイデンティティーおよびアクセス管理 (IAM) サービスとの統合機能を強化するバインドされたサービスアカウントトークンのサポートを提供します。

詳細は、バインドされたサービスアカウントトークンの使用 を参照してください。

1.2.2.2. oauth-proxy イメージストリームが利用可能になる

OpenShift Container Platform 4.4 では、サードパーティーの認証統合向けに **oauth-proxy** イメージストリームが導入されました。Red Hat レジストリーの **oauth-proxy** イメージは使用されなくなります。その代わりに、OpenShift Container Platform 4.4 クラスター以降をターゲットにする場合は、**openshift/oauth-proxy:v4.4** イメージストリームを使用する必要があります。これにより、後方互換性が保証され、重要な修正を取得するためのイメージストリームのトリガーを追加することができます。**v4.4** タグは、互換性に影響のある変更なしに、少なくとも今後の OpenShift Container Platform の 3 つのマイナーリリースで利用できます。各マイナーリリースでは、独自のタグも導入されます。

1.2.2.3. kube-apiserver はトークンの前にクライアント証明書をチェックする

以前のバージョンの OpenShift Container Platform では、**kube-apiserver** は認証用にトークンをチェックしてから、クライアント証明書をチェックしていました。**kube-apiserver** は、クライアント証明書をチェックしてからトークンをチェックするようになりました。

たとえば、**system:admin** kube 設定があり、以前のバージョンの OpenShift Container Platform で **oc --token=foo get pod** コマンドを実行している場合、トークン **foo** を持つユーザーとして認証が実行されました。今回のリリースにより、**system:admin** としての認証が実行されるようになりました。以前

のリリースでは、このような場合には、クライアント証明書の使用時にトークンを上書きするのではなく、パラメーター **--as** を使用してユーザーの権限を借用することが推奨されましたが、これは不要になりました。

1.2.3. ノード

1.2.3.1. Descheduler を使用した Pod のエビクト (テクノロジープレビュー)

Descheduler は実行中の Pod をエビクトし、Pod がより適したノードに再スケジュールできるようにします。

Pod のスケジュール解除は、以下のような状況で有効にすることができます。

- ノードの使用率が低くなっているか、使用率が高くなっている。
- テイントまたはラベルなどの、Pod およびノードアフィニティーの各種要件が変更され、当初のスケジュールの意思決定が特定のノードに適さなくなっている。
- ノードの障害により、Pod を移動する必要がある。
- 新規ノードがクラスターに追加されている。

詳細は、Descheduler を使用した Pod のエビクト を参照してください。

1.2.3.2. 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 Container Platform 管理者は、オーバーコミットのレベルを制御し、ノード上のコンテナーの密度を管理できるようになりました。クラスターレベルのオーバーコミットを Cluster Resource Override Operator を使用して設定し、開発者用のコンテナーに設定された要求と制限の比率について上書きすることができます。

詳細は、オーバーコミットされたノード上に Pod を配置するためのクラスターの設定 を参照してください。

1.2.4. クラスターモニタリング

1.2.4.1. Web コンソールでのモニタリングダッシュボード

Dashboards ビューが Web コンソールの **Monitoring** セクションから利用できるようになりました。これにより、OpenShift Container Platform クラスターおよびその依存するコンポーネントに対する透過性をもたらすメトリクスを表示できます。

1.2.4.2. hwmon コレクターが node-exporter で無効にされる

hwmon コレクターは、クラスターメトリクスを収集するために使用されなくなったため、node-exporter モニタリングコンポーネントで無効にされています。

1.2.4.3. cluster-reader でのノードメトリクスの読み取りが可能になる

cluster-reader ロールはデフォルトでノードメトリクスを読み取ることができるようになりした。

1.2.4.4. 複数のコンテナーが強制終了された場合のクラスターのアラート

メモリーの停止により複数のコンテナーが 15 分以内に強制終了される
と、**MultipleContainersOOMKilled** アラートで通知されます。

1.2.4.5. 新規 API サーバーのアラート

OpenShift Container Platform 4.4 では、2 つの新たな API サーバーアラートを利用できます。

- **ErrorBudgetBurn**: API サーバーが **5xx** 要求応答を発行する際に実行されます。
- **AggregatedAPIErrors**: 集約された API サーバーのエラー数が増えると実行されます。

1.2.4.6. Prometheus Operator のパーミッションの更新

Prometheus Operator によって管理されるカスタムリソース定義 (CRD) には、より制限的なパーミッションが含まれるようになりました。

Prometheus Operator が管理するカスタムリソース (CR) には以下が含まれます。

- **Prometheus**
- **ServiceMonitor**
- **PodMonitor**
- **Alertmanager**
- **PrometheusRule**

1.2.4.7. クラスターモニタリングコンポーネントのバージョン更新

以下のモニタリングコンポーネントがアップグレードされました。

- Prometheus: 2.14.0 から 2.15.2 へのバージョンアップグレード
- Alertmanager: 0.19.0 から 0.20.0 へのバージョンアップグレード
- Prometheus Operator: 0.34.0 から 0.35.1 へのバージョンアップグレード
- kube-state-metrics: 1.8.0 から 1.9.5 へのバージョンアップグレード
- Grafana: 6.4.3 から 6.5.3 へのバージョンアップグレード

1.2.5. Web コンソール

1.2.5.1. OperatorHub での IBM Marketplace の統合

IBM Marketplace が OpenShift Container Platform Web コンソールにある OperatorHub と統合されました。この統合により、IBM Marketplace でホストされる Operator を OperatorHub インターフェイス内からインストールし、管理できます。

1.2.5.2. Topology ビューでのアプリケーションの編集

Topology ビューを使用して **Developer** パースペクティブからアプリケーションを編集できるようになりました。

1.2.5.3. Helm リリースの作成

Developer Catalog で提供される Helm チャートから Helm リリースを作成できるようになりました。

1.2.6. ネットワーク

1.2.6.1. OpenShift Container Platform でのストリーム制御転送プロトコル (SCTP)

SCTP は、IP ネットワークの上部で実行される信頼できるメッセージベースのプロトコルです。これを有効にすると、SCTP を Pod とサービスの両方でプロトコルとして使用できます。詳細は、SCTP の使用 について参照してください。

1.2.6.2. DNS 転送の使用

DNS 転送を使用すると、指定のゾーンにどのネームサーバーを使用するかを指定することで、ゾーンごとにデフォルトの転送設定をオーバーライドできます。

詳細は、Using DNS forwarding を参照してください。

1.2.6.3. HAProxy がバージョン 2.0 にアップグレード

Ingress に使用される HAProxy がバージョン 1.8.17 から 2.0.13 にアップグレードされました。このアップグレードでは、OpenShift Container Platform に新たな API やサポートされるユーザー向け機能は追加されていません。しかしこのアップグレードにより、パフォーマンスが大幅に改善し、バグ修正が数多く追加されました。HAProxy 2.0 は、ネイティブ Prometheus メトリクスも追加し、他の OpenShift Container Platform コンポーネントがサポートするように設定されている場合に IPv6 の完全サポートを提供します。

1.2.6.4. Ingress の機能拡張

OpenShift Container Platform 4.4 には、以下の 2 つの重要な **Ingress** オブジェクトが導入されました。

- **Ingress** オブジェクトがルート受付ポリシー API を取得できる: 同じドメイン名を使用して複数の namespace で複数のアプリケーションを実行できます。
- **Ingress** オブジェクトを NodePort サービスで公開できる: 負荷分散ソリューションを詳細に制御できるように既存のロードバランサーの統合を容易にします。

1.2.7. ストレージ

1.2.7.1. CSI スナップショットを使用した永続ストレージ (テクノロジープレビュー)

Container Storage Interface (CSI) を使用して、ボリュームスナップショットを作成し、復元し、削除できるようになりました。この機能は、テクノロジープレビュー機能としてデフォルトで有効にされます。

詳細は、CSI ボリュームスナップショットの使用 を参照してください。

1.2.7.2. CSI クローン作成を使用した永続ストレージ (テクノロジープレビュー)

Container Storage Interface (CSI) を使用して、作成後のストレージボリュームのクローンを作成することができます。この機能は、テクノロジープレビュー機能としてデフォルトで有効にされます。

詳細は、CSI ボリュームのクローン作成の使用 を参照してください。

1.2.8. スケーリング

1.2.8.1. クラスターの最大数

OpenShift Container Platform 4.4 の クラスターの最大値 に関するガイダンスが更新されました。

4.4 でテスト済みの、ノードあたりの Pod 数の最大値は 500 です。

ご使用の環境のクラスター制限を見積もるには、OpenShift Container Platform Limit Calculator を使用できます。

1.2.9. 開発者のエクスペリエンス

1.2.9.1. 自動イメージプルーニング

自動イメージプルーニングを有効にできるようになりました。これはデフォルトでは有効にされていません。OpenShift Container Platform 4.4 のインストールまたは同バージョンへのアップグレード後にこのオプションについて通知されます。この自動化は、cron ジョブを作成し、定期的なイメージプルーニングを実行するイメージレジストリー Operator によって管理されます。

1.2.9.2. ステータスでのビルドオブジェクトの状態の報告

ビルドの状態が、既存の OpenShift Container Platform ビルドフェーズごとに追加されています。これらの状態には、ビルドライフサイクルのビルドについての情報が含まれます。**oc wait** などのコマンドを使用して、特定のビルドフェーズに到達するまで待機することもできます。

1.2.9.3. イメージレジストリーの再作成ロールアウト

イメージレジストリーのデプロイ時に、ロールアウトストラテジーの **Recreate** を使用できるようになりました。これにより、AWS Elastic Block Store などの **ReadWriteOnce** 永続ボリュームを使用できます。これらのストレージタイプを使用する場合、OpenShift Container Platform クラスターを正常にアップグレードするには **Recreate** ロールアウトストラテジーを使用する必要があります。

1.2.9.4. odo 機能拡張

odo には、ユーザーエクスペリエンスに重点を置いたいくつかの拡張機能および改良点が加えられました。

- **odo debug info** コマンドが利用可能になりました。
- **odo url** コマンドに、HTTPS URL を指定するための **--secure** フラグが含まれるようになりました。
- **odo create**、**odo url**、および **odo config** コマンドには、クラスターに対して変更を即時に実行できるように **--now** フラグを使用できるようになりました。
- **odo debug port-forward** コマンドは、デフォルトポートが使用されている場合にポートを自動的に選択できるようになりました。
- **odo storage** および **odo push** コマンドの出力は、読みやすさを向上させるために再構築されています。

- 実験モードが利用できるようになりました。このモードでは、devfile を使用したアプリケーションの作成などのテクノロジープレビュー機能を使用できます。
- テクノロジープレビュー機能の devfile のサポートが利用できるようになりました。詳細は、odo リリースノート を参照してください。

1.2.9.5. OpenShift Pipeline (テクノロジープレビュー)

OpenShift Pipeline は Tekton カスタムリソース (CR) を使用して、デプロイメントを自動化するための拡張可能な CI/CD ソリューションを作成します。これらの CR は Pipeline をアセンブルするためのビルディングブロックとして機能します。OpenShift Pipeline は、Pipeline を簡単にビルドするために使用できる再利用可能な Task のカタログを提供します。各 Pipeline は、CI サーバーのメンテナーンスなしに分離されたコンテナーで実行され、複数のプラットフォーム間で移植できます。

1.2.9.6. Helm 3 GA サポート

Helm は、Kubernetes および OpenShift Container Platform アプリケーションのパッケージマネージャーです。これは Helm チャートと呼ばれるパッケージ形式を使用し、アプリケーションやサービスの定義、インストールおよびアップグレードを単純化します。

Helm CLI は OpenShift Container Platform でビルドされ、これに同梱されており、Web コンソールの **CLI** メニューからダウンロードして利用できます。

1.2.10. Operator

1.2.10.1. etcd クラスター Operator

OpenShift Container Platform 4.4 では、etcd クラスター Operator が導入されています。これは etcd のスケーリングや、TLS 証明書などの etcd の依存関係のプロビジョニングを処理します。etcd クラスター Operator は、クラスターの以前の状態に復元するための障害復旧手順を単純化し、etcd メンバーの追加を自動化し、より正確な etcd メンバーの健全性についてのレポートを提供し、etcd クラスターのデバッグに役立つイベントをレポートします。

今回の更新により、以下の障害復旧スクリプトの名前が変更になりました。

- **etcd-snapshot-backup.sh** が **cluster-backup.sh** になりました。
- **etcd-snapshot-restore.sh** が **cluster-restore.sh** になりました。

詳細は、障害復旧について を参照してください。

1.2.10.2. Insights Operator が匿名の CSR を収集する

今回の機能拡張により、Insights Operator は匿名の証明書署名要求 (CSR) を定期的に収集し、Kubernetes で検証されない、または承認されていない CSR を特定するために使用されるようになりました。さらに、証明書が有効な場合にも Insights Operator はデータを収集します。これにより、OpenShift Container Platform のカスタマーサポートのエクスペリエンスが向上します。

1.2.10.3. registry.redhat.io に接続できない場合の Samples Operator の削除

Samples Operator がインストール時に **registry.redhat.io** に接続できない場合、サンプルのイメージストリームは作成されません。これにより、サンプルコンテンツのインストールによって OpenShift Container Platform クラスターのインストールが失敗することはなくなります。

クラスターのインストール時に生じる問題を回避できるように 代替またはミラーリングされたレジストリーを設定 できます。

1.2.11. ドキュメントの更新および規則

1.2.11.1. Apache ライセンス 2.0 でライセンスが適用された OpenShift Container Platform ドキュメント

OpenShift Container ドキュメントは Apache ライセンス 2.0 でライセンスが適用されています。これまでは Creative Commons Attribution-ShareAlike 3.0 Unported ライセンスに基づいてライセンスが適用されていました。

1.2.11.2. docs.openshift.com サイトの Copy ボタン

docs.openshift.com のすべてのコードブロックで **Copy** ボタンが提供されるようになりました。これにより、コードブロックのすべてのテキストをマシンのクリップボードにコピーできるようになりました。この機能は、カスタマーポータルバージョンの OpenShift Container Platform ドキュメントでは使用できません。

1.2.11.3. OpenShift Container Engine の名前が OpenShift Kubernetes Engine に変更される

Red Hat では、製品の提供する価値を適切に伝えることを目的とし、Red Hat OpenShift Container Engine の名前を Red Hat OpenShift Kubernetes Engine に変更しました。詳細は、About OpenShift Kubernetes Engine を参照してください。

1.2.11.4. Azure Red Hat OpenShift 4.3 バージョンのドキュメントが利用可能になる

この新たなバージョンは、Red Hat と Microsoft の両社によって共同で管理され、サポートされ、および文書化されています。

- Microsoft ドキュメント
- Red Hat ドキュメント

1.3. 主な技術上の変更点

OpenShift Container Platform 4.4 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。

Fluentd syslog プラグインを使用したクラスターログの送信 (RFC 3164)
OpenShift Container Platform 4.3 のログ転送機能で導入された変更により、Fluentd syslog プラグインを使用してログを外部 syslog サーバーに転送できなくなりました。しかし OpenShift Container Platform 4.4 ではこの機能が復元され、syslog プラグインを使用できます。OpenShift Container Platform バージョン 4.4 でのこのプラグインを設定する手順は、バージョン 4.2 とは異なります。詳細は、Fluentd syslog プラグインを使用したログの送信 (RFC 3164) を参照してください。

Operator SDK v0.15.0
OpenShift Container Platform 4.4 では Operator SDK v0.15.0 をサポートし、主に以下のような技術的な変更点が加えられています。

- **olm-catalog gen-csv** サブコマンドは **generate csv** サブコマンドに移行しました。
- **up local** サブコマンドは **run --local** サブコマンドに移行しました。

OpenShift Container Platform リリースのバイナリー sha256sum.txt.sig ファイルの名前が変更される
OpenShift Container Platform リリースに含まれる **sha256sum.txt.sig** ファイルの名前が **sha256sum.txt.gpg** に変更されました。このバイナリーファイルには、各インストーラーおよびクライアントバイナリーのハッシュが含まれており、これらはバイナリーの整合性を確認するために使用されます。

バイナリーファイルの名前を変更すると、GPG が **sha256sum.txt** を正しく検証できるようになりますが、これは、命名の競合により実行できませんでした。

1.4. 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 4.4 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。非推奨になったか、または削除された機能の詳細情報は、表の後に記載されています。

以下の表では、機能は以下のステータスでマークされています。

- **GA**: 一般公開機能
- **DEP**: 非推奨機能
- **-**: 削除された機能

表1.1 非推奨および削除機能のトラッカー

機能	OCP 4.2	OCP 4.3	OCP 4.4
サービスカタログ	DEP	DEP	DEP
テンプレートサービスブローカー	DEP	DEP	DEP
OpenShift Ansible Service Broker	DEP	DEP	-
OperatorSources	DEP	DEP	DEP
CatalogSourceConfigs	DEP	DEP	DEP
Operator Framework のパッケージマニフェスト形式	GA	GA	DEP
Docker のシステムコンテナー、CRI-O	-	-	-
Hawkular エージェント	-	-	-
Pod の PreSet	-	-	-
監査ポリシー	-	-	-

機能	OCP 4.2	OCP 4.3	OCP 4.4
クラスター化された MongoDB テンプレート	-	-	-
クラスター化された MySQL テンプレート	-	-	-
CephFS Provisioner	-	-	-
Manila Provisioner	-	-	-

1.4.1. 非推奨の機能

1.4.1.1. OpenShift CLI config フラグ

oc で使用される **--config** フラグは非推奨となりました。代わりに **--kubeconfig** フラグを使用する必要があります。

1.4.1.2. OpenShift CLI timeout フラグ

oc rsh で使用される **--timeout** フラグは非推奨となりました。代わりに **--request-timeout** フラグを使用する必要があります。

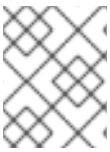
1.4.1.3. OpenShift editor

OS_EDITOR は非推奨となりました。ユーザーは代わりに **KUBE_EDITOR** または **EDITOR** を使用する必要があります。

1.4.1.4. machineCIDR ネットワークパラメーター

install-config.yaml ファイルで使用される **machineCIDR** ネットワークパラメーターは非推奨となりました。**machineNetwork.cidr** を代わりに使用する必要があります。

1.4.1.5. サービスカタログ、テンプレートサービスブローカー、Ansible Service Broker、およびそれらの Operator



注記

サービスカタログは、OpenShift Container Platform 4 ではデフォルトでインストールされません。

OpenShift Container Platform 4.2 以降ではサービスカタログ、テンプレートサービスブローカー、Ansible Service Broker およびそれらに関連付けられた Operator が非推奨になりました。

Ansible Service Broker、Ansible Service Broker Operator、および以下の APB は OpenShift Container Platform 4.4 で削除されています。

- APB ベースイメージ
- APB ツールコンテナー
- PostgreSQL APB

- MySQL APB
- MariaDB APB

以下の関連 API も削除されました。

- **.automationbroker.io/v1alpha1**
- **.osb.openshift.io/v1**

サービスカタログおよびテンプレートサービスブローカーは、以下の関連する API と共に今後の OpenShift Container Platform リリースで削除されます。

- **.servicecatalog.k8s.io/v1beta1**

これらが 4.4 で有効にされている場合、Web コンソールは、これらの機能が依然として有効にされていることをクラスター管理者に警告します。以下のアラートは **Monitoring → Alerts** ページから表示でき、アラートには **Warning** の重大度が設定されます。

- **ServiceCatalogAPIServerEnabled**
- **ServiceCatalogControllerManagerEnabled**
- **TemplateServiceBrokerEnabled**

service-catalog-controller-manager および **service-catalog-apiserver** クラスター Operator も 4.4 で **Upgradeable=false** に設定されています。これは、これらがインストールされている場合に、クラスターの 4.5 などの次のマイナーバージョンへのアップグレードがブロックされることを意味します。ただし、4.4.z などの z-stream リリースへのアップグレードはこの場合も引き続き許可されます。

サービスカタログがインストールされている場合、クラスター管理者は サービスカタログのアンインストール を参照し、OpenShift Container Platform の次のマイナーバージョンがリリースされる前にこれをアンインストールできます。

1.4.1.6. OperatorSources、 CatalogSourceConfigs、およびパッケージ形式が非推奨になる

OperatorSources および **CatalogSourceConfigs** オブジェクトが OperatorHub で非推奨になりました。以下の関連する API は今後のリリースで削除されます。

- **operatorsources.operators.coreos.com/v1**
- **catalogsourceconfigs.operators.coreos.com/v2**
- **catalogsourceconfigs.operators.coreos.com/v1**

Operator Framework の現在のパッケージ形式 **Package Manifest Format** は今回のリリースで非推奨とされており、今後のリリースで新規の **Bundle Format** (バンドル形式) に置き換えられます。そのため、Package Manifest Format でカタログをビルドする **oc adm catalog build** コマンドも非推奨になりました。

今後の Bundle Format および **opm** CLI の詳細は、アップストリームの OKD ドキュメント を参照してください。

1.4.1.6.1. カスタム OperatorSources および CatalogSourceConfigs オブジェクトの変換

OpenShift Container Platform 4.4 のクラスターにカスタム **OperatorSources** または **CatalogSourceConfigs** オブジェクトがある場合、**marketplace** クラスター Operator は

Upgradeable=false 条件を設定し、**Warning** アラートを発行します。これは、これらがインストールされている場合に、クラスターの 4.5 などの次のマイナーバージョンへのアップグレードがブロックされることを意味します。ただし、4.4.z などの z-stream リリースへのアップグレードはこの場合も引き続き許可されます。

クラスター管理者は、カスタム **OperatorSources** または **CatalogSourceConfigs** オブジェクトを変換し、**CatalogSource** オブジェクトを直接使用してこのアラートをクリアできます。

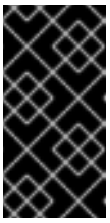
手順

1. カスタム **OperatorSources** または **CatalogSourceConfigs** オブジェクトを削除します。
   a. すべての namespace で **OperatorSources** または **CatalogSourceConfigs** オブジェクトを検索します。

```
$ oc get opsrc --all-namespaces
$ oc get csc --all-namespaces
```

   b. 関連するすべての namespace からすべてのカスタムオブジェクトを削除します。

```
$ oc delete opsrc <custom_opsrc_name> -n <namespace>
$ oc delete csc <custom_csc_name> -n <namespace>
```



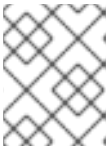
重要

openshift-marketplace namespace でデフォルトの **OperatorSources** オブジェクトを削除しないでください (**redhat-operators**、**community-operators**、**certified-operators**、および **redhat-marketplace**)。ただし、これらは誤って削除された場合にはブートストラップされます。

2. ネットワークが制限された環境での Operator カタログイメージのビルド についてドキュメントで説明されている手順を使用して、新規のカタログイメージを作成およびプッシュし、**oc adm catalog build** コマンド手順で以下の変更を加えます。
   - **--appregistry-org** を Quay.io などの App Registry インスタンスの namespace に変更します。
   - **--to** をビルドされたカタログイメージに適用され、プッシュされるイメージリポジトリータグに変更します。

   以下に例を示します。

```
$ oc adm catalog build \
    --appregistry-org <namespace> \
    --from=registry.redhat.io/openshift4/ose-operator-registry:v4.4 \
    --to=quay.io/<namespace>/<catalog_name>:<tag> \
    [-a ${REG_CREDS}]
```



注記

oc adm catalog build コマンドは非推奨となりましたが、非推奨になった機能は引き続きサポートされます。

3. **CatalogSource** オブジェクトを新規カタログイメージを参照するクラスターに適用します。

```
cat <<EOF | oc apply -f -

apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: my-operator-catalog
  namespace: openshift-marketplace
spec:
  sourceType: grpc
  image: quay.io/<namespace>/<catalog_name>:<tag> 1
  displayName: My Operator Catalog
  updateStrategy:
    registryPoll: 2
      interval: 30m
EOF
```

1. Operator カタログイメージを指定します。
2. **CatalogSource** オブジェクトは新規バージョンの有無を自動的にチェックし、最新の状態を維持します。

1.4.2. 削除された機能

1.4.2.1. OpenShift CLI シークレットのサブコマンド

OpenShift Container Platform 3.9 で非推奨となった以下の **oc secrets** サブコマンドは利用できなくなりました。

- **new**
- **new-basicauth**
- **new-dockercfg**
- **new-sshauth**

その代わりとして **oc create secret** コマンドを使用する必要があります。

1.4.2.2. OpenShift CLI build-logs コマンド

oc build-logs コマンドは OpenShift Container Platform 3.11 で非推奨となり、削除されています。その代わりに **oc logs** を使用する必要があります。

1.4.2.3. 非推奨のアップストリーム Kubernetes メトリクスが削除される

非推奨のアップストリーム Kubernetes メトリクスがすべて削除されました。削除されたメトリクスの詳細な一覧は以下になります。

Kubelet メトリクス

- **kubelet_pod_worker_latency_microseconds**

- **kubelet_pod_start_latency_microseconds**
- **kubelet_cgroup_manager_latency_microseconds**
- **kubelet_pod_worker_start_latency_microseconds**
- **kubelet_pleg_relist_latency_microseconds**
- **kubelet_pleg_relist_interval_microseconds**
- **kubelet_runtime_operations**
- **kubelet_runtime_operations_latency_microseconds**
- **kubelet_runtime_operations_errors**
- **kubelet_eviction_stats_age_microseconds**
- **kubelet_device_plugin_registration_count**
- **kubelet_device_plugin_alloc_latency_microseconds**
- **kubelet_network_plugin_operations_latency_microseconds**

スケジューラーメトリクス

- **scheduler_e2e_scheduling_latency_microseconds**
- **scheduler_scheduling_algorithm_predicate_evaluation**
- **scheduler_scheduling_algorithm_priority_evaluation**
- **scheduler_scheduling_algorithm_preemption_evaluation**
- **scheduler_scheduling_algorithm_latency_microseconds**
- **scheduler_binding_latency_microseconds**
- **scheduler_scheduling_latency_seconds**

API サーバーメトリクス

- **apiserver_request_count**
- **apiserver_request_latencies**
- **apiserver_request_latencies_summary**
- **apiserver_dropped_requests**
- **apiserver_storage_data_key_generation_latencies_microseconds**
- **apiserver_storage_transformation_failures_total**
- **apiserver_storage_transformation_latencies_microseconds**
- **apiserver_proxy_tunnel_sync_latency_secs**

Docker メトリクス

- **kubelet_docker_operations**
- **kubelet_docker_operations_latency_microseconds**
- **kubelet_docker_operations_errors**
- **kubelet_docker_operations_timeout**

Reflector メトリクス

- **reflector_items_per_list**
- **reflector_items_per_watch**
- **reflector_list_duration_seconds**
- **reflector_lists_total**
- **reflector_short_watches_total**
- **reflector_watch_duration_seconds**
- **reflector_watches_total**

etcd メトリクス

- **etcd_helper_cache_hit_count**
- **etcd_helper_cache_miss_count**
- **etcd_helper_cache_entry_count**
- **etcd_request_cache_get_latencies_summary**
- **etcd_request_cache_add_latencies_summary**
- **etcd_request_latencies_summary**

変換メトリクス

- **transformation_latencies_microseconds**
- **transformation_failures_total**

他のメトリクス

- **admission_quota_controller_adds**
- **crd_autoregistration_controller_work_duration**
- **APIServiceOpenAPIAggregationControllerQueue1_adds**
- **AvailableConditionController_retries**
- **crd_openapi_controller_unfinished_work_seconds**
- **APIServiceRegistrationController_retries**

- admission_quota_controller_longest_running_processor_microseconds
- crdEstablishing_longest_running_processor_microseconds
- crdEstablishing_unfinished_work_seconds
- crd_openapi_controller_adds
- crd_autoregistration_controller_retries
- crd_finalizer_queue_latency
- AvailableConditionController_work_duration
- non_structural_schema_condition_controller_depth
- crd_autoregistration_controller_unfinished_work_seconds
- AvailableConditionController_adds
- DiscoveryController_longest_running_processor_microseconds
- autoregister_queue_latency
- crd_autoregistration_controller_adds
- non_structural_schema_condition_controller_work_duration
- APIServiceRegistrationController_adds
- crd_finalizer_work_duration
- crd_naming_condition_controller_unfinished_work_seconds
- crd_openapi_controller_longest_running_processor_microseconds
- DiscoveryController_adds
- crd_autoregistration_controller_longest_running_processor_microseconds
- autoregister_unfinished_work_seconds
- crd_naming_condition_controller_queue_latency
- crd_naming_condition_controller_retries
- non_structural_schema_condition_controller_queue_latency
- crd_naming_condition_controller_depth
- AvailableConditionController_longest_running_processor_microseconds
- crdEstablishing_depth
- crd_finalizer_longest_running_processor_microseconds
- crd_naming_condition_controller_adds

- **APIServiceOpenAPIAggregationControllerQueue1_longest_running_processor_microseconds**
- **DiscoveryController_queue_latency**
- **DiscoveryController_unfinished_work_seconds**
- **crd_openapi_controller_depth**
- **APIServiceOpenAPIAggregationControllerQueue1_queue_latency**
- **APIServiceOpenAPIAggregationControllerQueue1_unfinished_work_seconds**
- **DiscoveryController_work_duration**
- **autoregister_adds**
- **crd_autoregistration_controller_queue_latency**
- **crd_finalizer_retries**
- **AvailableConditionController_unfinished_work_seconds**
- **autoregister_longest_running_processor_microseconds**
- **non_structural_schema_condition_controller_unfinished_work_seconds**
- **APIServiceOpenAPIAggregationControllerQueue1_depth**
- **AvailableConditionController_depth**
- **DiscoveryController_retries**
- **admission_quota_controller_depth**
- **crdEstablishing_adds**
- **APIServiceOpenAPIAggregationControllerQueue1_retries**
- **crdEstablishing_queue_latency**
- **non_structural_schema_condition_controller_longest_running_processor_microseconds**
- **autoregister_work_duration**
- **crd_openapi_controller_retries**
- **APIServiceRegistrationController_work_duration**
- **crdEstablishing_work_duration**
- **crd_finalizer_adds**
- **crd_finalizer_depth**
- **crd_openapi_controller_queue_latency**
- **APIServiceOpenAPIAggregationControllerQueue1_work_duration**

- **APIServiceRegistrationController_queue_latency**
- **crd_autoregistration_controller_depth**
- **AvailableConditionController_queue_latency**
- **admission_quota_controller_queue_latency**
- **crd_naming_condition_controller_work_duration**
- **crd_openapi_controller_work_duration**
- **DiscoveryController_depth**
- **crd_naming_condition_controller_longest_running_processor_microseconds**
- **APIServiceRegistrationController_depth**
- **APIServiceRegistrationController_longest_running_processor_microseconds**
- **crd_finalizer_unfinished_work_seconds**
- **crdEstablishing_retries**
- **admission_quota_controller_unfinished_work_seconds**
- **non_structural_schema_condition_controller_adds**
- **APIServiceRegistrationController_unfinished_work_seconds**
- **admission_quota_controller_work_duration**
- **autoregister_depth**
- **autoregister_retries**
- **kubeproxy_sync_proxy_rules_latency_microseconds**
- **rest_client_request_latency_seconds**
- **non_structural_schema_condition_controller_retries**

1.4.2.4. Prometheus での高粒度の要求期間バケット

粒度の高い要求期間バケットは Prometheus でドロップされ、**apiserver_request_duration_seconds_bucket** メトリクスで追跡されました。これにより、他のモニタリングコンポーネントからの意味のある情報を提供するアラートのために十分なバケットが残されますが、カーディナリティーは大幅に減少します。

1.5. バグ修正

apiserver-auth

- 以前のバージョンでは、ブラウザーベースのログインのみが設定されている場合にユーザーが CLI からログインしようとすると、ユーザー名とパスワードの入力を要求するプロンプトが出されました。ブラウザーベースのログインのみが設定されている場合にユーザーが CLI からロ

グインを試行すると、ログイントークンの取得方法を説明するメッセージが表示されるようになりました。(BZ#1671604)

- 以前のバージョンでは、競合状態により、マウントされた提供証明書が変更または表示されても認識されず、提供証明書は HTTPS エンドポイントのメトリクススクレーパーによって信頼されませんでした。競合状態が取り除かれ、**library-go** をベースとする Operator は提供証明書を正しく再読み込みできるようになりました。(BZ#1779438)

- 以前のバージョンでは、IPv6 アドレスが使用されている場合に、Kubernetes API サーバーサービスネットワークアドレスが適切に処理されませんでした。OAuth プロキシーは、IPv6 アドレスで動作する場合に Kubernetes API サーバーに正しく接続できるようになりました。(BZ#1789462)

Build

- ビルドを開始する前に、OpenShift Container Platform ビルダーは提供された Dockerfile を解析し、ビルドに使用する修正バージョンの再構築を行い、ラベルを追加し、 **FROM** 命令で名前が付けられたイメージの置き換えを処理します。生成された Dockerfile は **ENV** および **LABEL** 命令を常に正しく再構築する訳ではありませんでした。生成された Dockerfile には元の Dockerfile には含まれない **=** 文字が含まれることがあり、これにより、ビルドは構文エラーを出して失敗しました。今回のバグ修正により、変更した Dockerfile を生成する際に、**ENV** および **LABEL** 命令の元のテキストがそのまま使用されるようになりました。その結果、ビルドプロセスでは、**ENV** および **LABEL** の命令に構文エラーが出されなくなりました。(BZ#1821860)

- **JenkinsPipeline** ビルドストラテジーは OpenShift Container Platform 4.3.0 では非推奨になりました。代わりに Jenkins または OpenShift Pipeline で **Jenkinsfile** オブジェクトを直接使用します。(BZ#1804976)

- ビルドラベルの生成と検証は Kubernetes の期待値に完全に対応していませんでした。ビルドは無効なラベルのエラーを出して特定の **BuildConfig** オブジェクト名で失敗する可能性がありました。今回のバグ修正により、ビルドコントローラーおよびビルド API サーバーが更新され、完全な Kubernetes 検証ルーチンを使用して、追加されたビルドラベルが Kubernetes ラベルの基準を満たすようになりました。その結果、有効な **BuildConfig** オブジェクト名を持つビルドは、ビルドラベルの値が無効であるという理由で失敗しなくなりました。(BZ#1804934)

- 以前のバージョンでは、Samples Operator の **samplesRegistry** フィールドが変更され、これがイメージストリームのインポートエラーを生じさせる場合、イメージストリームのステータスを表示する際に Samples Operator が設定変更を取得していないかのように表示されました。Samples Operator の **samplesRegistry** フィールドが変更され、これがイメージストリームのインポートエラーを生じさせる場合、新たな失敗の理由がイメージストリームのステータスに適切に表示されるようになりました。(BZ#1795705)

- 以前のバージョンでは、**RUN** 命令の後に、OpenShift Container Platform ビルダーは作成された各バインドマウントのアンマウントを試行し、プロセスで発生したエラーをログに記録していました。ビルダーは最上位のディレクトリーのみをアンマウントし、カーネルがバインドマウントをアンマウントするようになりました。エラーが発生することはなくなり、そのためエラーの報告がなくなりました。(BZ#1772179)

- 以前のバージョンでは、ビルドストラテジーで **incremental** および **forcePull** フラグの両方を **true** に設定すると、ビルドでイメージのプルにイメージをプッシュするための認証情報が使用される可能性がありました。その結果、プライベートレジストリーからのイメージのプルが失敗しました。ビルドイメージは、**incremental** および **forcePull** の両方が **true** に設定されている場合にレジストリーのプッシュおよびプルの認証情報を適切に管理できるようになりました。(BZ#1774492)

- コマンド **oc new-build** には、**oc new-app** コマンドで利用可能な **--insecure-registries** フラグ

がありませんでした。このフラグにより、セキュアでないイメージの参照 URL をソースとして使用できました。そのため、**oc new-build** 呼び出しは、ビルドのベースイメージとして指定される HTTP ベースのイメージ参照を使用して HTTPS 接続を試行する際にエラーを受信しました。**--insecure-registries** オプションが **oc new-build** コマンドに追加され、ユーザーは、セキュアでないレジストリーをベースイメージとして参照するビルドを作成できるようになりました。(BZ#1780714)

Cloud Credential Operator

- Cloud Credential Operator (CCO) は、CCO が無効にされている場合でも、条件と共に **CredentialsRequest** CR について報告しました。アラートは、Operator が無効にされるように設定されている場合でも表示されました。今回のバグ修正により、CCO が disabled に設定されている場合は条件が報告されなくなりました。(BZ#1794536)

- **CredentialsRequest** CR を調整すると、すでに存在するロール割り当ての作成が試行され、Microsoft Azure ログに **create role assignment** エラーが表示されました。今回のバグ修正により、すでに存在するロールの割り当ての有無がチェックされ、存在する場合には作成されなくなりました。その結果、Azure ログのエラーメッセージの数も少なくなりました。(BZ#1776079)

コンソール **kubevirt** プラグイン

- アノテーションを使用せずに仮想マシンテンプレートを選択すると、仮想マシンウィザードが予期せずに閉じました。仮想マシンウィザードで、アノテーションのないテンプレートを使用できるようになりました。(BZ#1776190)

- 以前のバージョンでは、URL をディスクイメージソースとして使用する仮想マシンテンプレートを作成する場合、テンプレートの使用時に作成される仮想マシンについて永続ボリューム要求 (PVC) は作成されませんでした。このようなテンプレートから仮想マシンを新規に作成する場合、PVC のクローンが作成され、ディスクイメージに使用できるようになりました。(BZ#1779116)

- 以前のバージョンでは、テンプレート内のメモリーおよびストレージの値を解釈する際に異なる単位が使用されていたため、仮想マシンの作成要求が失敗していました。仮想マシンテンプレートのメモリーおよびストレージの値には、Gi 単位が一貫して使用されるようになりました。(BZ#1792101)

- 以前のバージョンでは、仮想マシンウィザードは、仮想マシンテンプレートのディスク設定を無視していました。仮想マシンウィザードでは、テンプレートで指定されている場合にディスク設定を使用できるようになりました。(BZ#1782434)

- 以前のバージョンでは、UI で失敗した仮想マシンの移行が成功したと報告されていました。仮想マシンの移行時に、UI は仮想マシンの移行に失敗するとそれを正確に報告するようになりました。(BZ#1785344)

- 以前のバージョンでは、仮想マシンに複数の CD-ROM ドライブが定義されている場合、各 CD-ROM のダイアログを保存してから再度開くことなしにその ドライブを削除することはできませんでした。複数の CD-ROM ドライブを、ダイアログを保存したり再度開いたりせずに削除できるようになりました。(BZ#1786070)

- 以前のバージョンでは、YAML が無効であったため、仮想マシンウィザードによって使用されるデフォルトの YAML で仮想マシンを作成することができませんでした。ウィザードで仮想マシンを作成する際に、デフォルトの仮想マシンテンプレートを使用できるようになりました。(BZ#1808304)

- 以前のバージョンでは、テンプレート YAML のブート順序が認識されないため、ビジュアルエディターを使用して起動順序を変更することはできませんでした。ビジュアルエディターを使用してブート順序を変更できるようになりました。(BZ#1789269)

イメージ

- **oc tag** コマンドは、**ImageStreamTag** オブジェクトがアクセスできない場合にイメージストリームを更新しませんでした。このコマンドは、新規タグが作成されたと報告しますが、実際は作成されませんでした。今回のバグ修正により、**oc tag** コマンドが更新され、**ImageStreamTag** API のパーミッションがない場合でも、タグが実際に作成されるようになりました。(BZ#1746149)

- Base64 がパディングされているか/いないかを検出する際に、デコーダーは文字列の長さに依存していました。そのため、デコーダーは空白を含むプルシークレットを処理できませんでした。今回のバグ修正により、文字列の末尾にパディング記号があるかどうかのチェックが行われるようになりました。その結果、空白のあるプルシークレットを使用してイメージをプルできるようになりました。(BZ#1776599)

イメージレジストリー

- 以前のバージョンでは、イメージレジストリー Operator は、非管理 (unmanaged) 状態の場合に新規バージョンを報告しませんでした。これによりアップグレードがブロックされました。今回のバグ修正により、イメージレジストリー Operator が非管理状態でも正確なバージョンを報告するようになり、アップグレードが正常に実行されるようになりました。(BZ#1791934)

- 以前のバージョンでは、**nodeca** デーモンセットは **NoSchedule** テイントを容認しませんでした。これにより、ノード上で Pod が欠落する状態が生じました。今回のバグ修正により容認が追加され、テイントが付けられたノードが **nodeca** デーモンセットから更新を受信するようになりました。(BZ#1785115

- イメージレジストリー Operator は、RWO ボリュームと互換性がないローリング更新ストラテジーを使用しており、RWO ボリュームを使用することができませんでした。今回のバグ修正により、イメージレジストリー Operator はローリング更新ストラテジーを選択できるようになり、高可用性のない設定 (単一のレプリカのみを持つ設定など) の RWO ボリュームでデプロイできるようになりました。(BZ#1798759)

- 以前のバージョンでは、イメージレジストリー Operator はストレージの削除時にストレージのステータスをクリーンアップしませんでした。レジストリーが **Managed** 状態に戻ると、ブートストラップする必要があるストレージを検出できませんでした。今回のバグ修正により、イメージレジストリー Operator はストレージのステータスをクリーンアップし、Operator が **Managed** 状態に戻る際にストレージを作成できるようになりました。(BZ#1787488)

インストーラー

- 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で AWS でプロビジョニングしたクラスターの以前のバージョンには、TCP および UDP ポート 30000-32767 でコントロールプレーンホストからワーカーへのトラフィックを許可するセキュリティーグループルールが含まれていません。このため、新規の OVN Networking コンポーネントは意図された通りに機能しません。今回のリリースより、必要なセキュリティーグループルールがこれらのクラスターに追加され、TCP および UDP ポート 30000-32767 でのコントロールプレーンとワーカーマシン間の通信が可能になり、OVN Networking コンポーネントが意図された通りに機能するようになりました。(BZ#1763936)

- 以前のバージョンでは、Red Hat Enterprise Linux (RHEL) ノードでのアップグレードプロセスはブロックされていました。プロキシーの背後からイメージをプルできない場合に、マシン設定の不要な適用ステップが失敗しました。この不要なステップが削除され、プロキシーの背後での RHEL ノードへのアップグレードを正常に実行することができます。(BZ#1786297)

- 以前のバージョンでは、RHEL 7 ノードをバージョン 4.2.12 からアップグレードすると、マシン設定が MCO によって適切に更新されませんでした。パッケージが更新したファイルをローカルディスクにインストールするため、MCO は RHEL ノードで設定の更新を処理しませんでした。今回のリリースにより、マシン設定の適用ステップが復元し、イメージのプルプロセスをプロキシーの背後で実行できるようになりました。マシン設定はパッケージの更新後に正しく適用され、RHEL 7 ノードでのアップグレードは正常に実行できます。(BZ#1792139)

kube-apiserver

- 集約された API サーバーステータスのメトリクスは存在しますが、それらのアラートは提供されませんでした。今回のリリースより、集約された API が短期間に多くのエラーを報告するとエラーが表示されるようになりました (これは、サービスの可用性が頻繁に変更されることを示すためです)。(BZ#1772564)

kube-controller-manager

- 以前のバージョンでは、証明書が適切に伝播されていなかったため、クラウドプロバイダーは man-in-the-middle プロキシーの背後で初期化できませんでした。今回のリリースより、証明書は **kube-controller-manager** に適切に伝播され、クラウドプロバイダーは man-in-the-middle プロキシーと予想通りに機能するようになりました。(BZ#1772756)

ロギング

- 以前のバージョンでは、Fluentd プラグインを使用し、syslog プロトコル (RFC 3164) を使用してログを外部システムに転送することができました。OpenShift Container Platform 4.3 に追加されたログ転送 API では、syslog を使用してログ転送を設定するプロセスを変更しました。そのため、OpenShift Container Platform 4.2 と同じ方法を使用してログを転送できなくなりました。この変更に対応するために新たなプロセスが設定され、syslog プロトコルを使用したログ転送が許可されるようになりました。この変更は OpenShift Container Platform 4.3.7 にバックポートされました。そのため、引き続きログを外部 syslog サーバーに転送できます。(BZ#1799024)

Machine Config Operator

- Kuryr などの一部のアプリケーションは HAProxy タイムアウト値による影響を受ける可能性があるため、API LB には 24 時間のタイムアウト値が使用されていました。HAProxy の再読み込み操作が短期間に複数回トリガーされた場合には、HAProxy プロセスが多数累積される可能性があります。今回のバグ修正により、デフォルトタイムアウトの 120 秒後に **SIGTERM** が終了していない古い HAProxy プロセスに強制的に送信されるようになりました。その結果、長い期間存在する重複した HAProxy プロセスの数が減ります。(BZ#1771566)

メータリング Operator

- メータリングは S3 バケットデータを管理したり、削除したりしません。レポートを削除する際に、メータリングデータを保存するために使用される S3 バケットは手動でクリーンアップする必要があります。S3 バケットに保存されているレポーティングデータを手動で消去せず、同じレポートが再作成される場合には、元のレポーティングデータが引き続き存在するため、行エントリーの重複が生じます。(BZ#1728350)

モニタリング

- OAuth プロキシーコンテナーの readiness プローブの設定が正しくないため、コンテナーログのエラーメッセージが 10 秒ごとに増加していました。readiness プローブが適切な設定で設定されるようになりました。その結果、エラーメッセージはログに表示されなくなりました。(BZ#1658899)

- **cluster-reader** ロールにはノードまたは Pod メトリクスを表示するパーミッションがないた

め、そのロールにバインドされたユーザーは、**oc top node** などのコマンドを使用してメトリクスにアクセスできませんでした。**cluster-reader** ロールが更新され、メトリクスの表示を許可するパーミッションが含まれるようになりました。(BZ#1723662)

- 新たな実験的な Prometheus ユーザーインターフェイスがアップストリームで導入されました。この新規の実験的なインターフェイスは完全にテストされておらず、まだ安定していません。そのため、実験的なインターフェイスからデフォルトインターフェイスに切り替えると、空のページが返されました。この問題を回避するために、実験的な UI へのリンクが非表示にされています。その結果、実験的な Prometheus インターフェイスにアクセスできなくなっています。(BZ#1781415)

- OpenShift Container Platform は一部の記録ルールを正しく評価せず、その記録ルールから生成されたメトリクスが欠落していました。記録ルールは修正されました。すべての記録ルールで評価を正常に実行できるようになりました。(BZ#1807843, BZ#1779324)

ネットワーク

- 以前の Egress IP バグ修正では Egress IP の削除後に完全にクリーンアップを実行できず、無害な余分の iptables ルールがノードに残される可能性がありました。今回のバグ修正により、余分のルールが使用されなくなった場合にそれらが削除されるようになりました。(BZ#1787488)

- 以前のバージョンでは、**httpProxy** または **httpsProxy** ホスト名を大文字で使用すると、CNO が fatal (致命的) の状態になりました (これは RFC 3986 の違反を意味します)。そのため、CNO は動作していませんでした。今回のバグ修正により、これが **golang url.ParseRequestURI** で解析されるようになりました (これにより、RFC 3986 およびいくつかの RFC が適切に実装されます)。そのため、**httpProxy** および **httpsProxy** で大文字を使用できるようになりました。(BZ#1802606)

- 以前のバージョンでは、SDN で使用される kubelet が使用する kubeconfig がそのパスを変更することにより、SDN に空のファイルの解析を試行する null 逆参照 (null dereference) が含まれました。今回のバグ修正により、SDN は古いパスと新しいパスの両方を処理できるようになりました。(BZ#1781707)

ノード

- 以前のバージョンでは、kubelet 証明書の有効期限が切れるとアラートが発行されませんでした。そのため、管理者が認識しない状態で kubelets が機能を停止しました。今回の修正により、期限切れの証明書を報告する **server_expiration_renew_errors** メトリクスが追加されました。(BZ#1767523)

- Conmon は kubelet exec liveness プローブでタイムアウトが生じました。これにより、一部の exec プローブが失敗し、コンテナーが強制終了し、再起動していました。exec liveness プローブが想定通りに機能するようになりました。(BZ#1817568)

- ノードの再起動時に、CRI-O は Pod が復元されない場合に IP アドレスを適切にクリーンアップしませんでした。これにより、ノード IP が使い切られ、Pod が起動しなくなりました。再起動後に Pod を復元できない場合、Pod ネットワークは破棄され、今後の使用のために IP アドレスが解放されるようになりました。(BZ#1781824)

- CRI-O が起動しないため、RHEL 7 を OpenShift Container Platform クラスターに追加できませんでした。この問題は Conmon パッケージの問題によって発生しました。今回のリリースの修正により Conmon が修正され、RHEL 7 を OpenShift Container Platform クラスターに追加できるようになりました。(BZ#1809906)

- Horizontal Pod Autoscaler (HPA) は終了した init コンテナーからメトリクスを受信しませんでした。この問題は、終了した init コンテナーのゼロベースのメトリクスを送信することにより

修正され、HPA が init コンテナーを使って Pod で分析を実行できるようになりました。(BZ#1814283)

- kubelet メトリクスエンドポイントは定期的に **500** ステータスコードを返し、Prometheus が kubelet エンドポイントおよびノードのメトリクスを収集することを防いでいました。**500** コードはメトリクスストリームに混在する実行されないコンテナーによって生じ、これにより重複するメトリクスが挿入されました。このバグは修正され、メトリクスは kubelet から正しく報告されるようになりました。(BZ#1748073)

oc

- **oc logs** コマンドは、OpenShift CLI の内部コードが API の新規バージョンをサポートしないため、一部のリソースについてのエラーを返しました。OpenShift CLI がすべての既知の API タイプおよびバージョンをサポートするようになり、**oc logs** がすべてのリソースで機能するようになりました。(BZ#1774366)

- **--since** 引数を指定して **oc adm node-logs** コマンドを実行すると、エラーが発生しました。これは予想されるタイムスタンプ形式にある入力ミスによって生じました。入力ミスが修正され、**oc adm node-logs** が **--since** 引数と共に機能するようになりました。(BZ#1779563)

openshift-apiserver

- 以前のバージョンでは、Helm 3.0.0+ は OpenShift Container Platform オブジェクトと共に機能しませんでした。これにより、有効な Helm チャートのデプロイの試行時にエラーが生じました。今回の更新により、OpenShift Container Platform オブジェクトと共に Helm チャートをデプロイできるようになりました。(BZ#1773682)

openshift-controller-manager

- 以前のバージョンでは、**openshift-controller-manager** メトリクスは 1.16 Kubernetes Prometheus レジストリーに正しく登録されませんでした。そのため、OpenShift Container Platform コントロールプレーンのメトリクスが欠落していました。今回の更新により、**openshift-controller-manager** メトリクスが適切に登録され、欠落していた OpenShift Container Platform コントロールプレーンメトリクスが復元されています。(BZ#1810304)

- 以前のバージョンでは、プルシークレットは、関連付けられたトークンが削除される際に削除されないことがありました。そのため、プルシークレットは Kubernetes サービスアカウントに関連付けられたままになりました。今回の更新により、プルシークレットと関連付けられたトークンシークレットの所有者への参照が設定されました。これで、プルシークレットが削除されると、関連付けられたトークンも削除されます。(BZ#1806792)

- 以前のバージョンでは、内部レジストリーでプルシークレットを作成するためのレート制限が低くされていました。これにより、短時間で多数の namespace が作成されると待機時間が長くなっていました。今回の更新により、内部レジストリーでプルシークレットを作成するためのレート制限が引き上げられました。プルシークレットは、負荷が大きい場合でも迅速に作成できるようになりました。(BZ#1819849)

RHCOS

- ネットワークチーミングが Red Hat Enterprise Linux CoreOS (RHCOS) でサポートされるようになりました。**teamd** および **NetworkManager-team** RPM が RHCOS に追加され、チーミング対象のネットワークデバイスのセットアップおよび管理が可能になりました。(BZ#1758162)

サンプル

- IPv6 は **registry.redhat.io** ではサポートされていませんでした。つまり、すべての Red Hat サンプルが **registry.redhat.io** でホストされるため、Samples Operator ではイメージストリーム

のインストールの試行が意図されていませんでした。IPv6 は Red Hat および OpenShift Container Platform の主要な機能として取り組まれており、Samples Operator による IPv6 での OpenShift Container Platform のインストールによって破損が生じることはなくなりました。(BZ#1788676)

- 以前のバージョンでは、Samples Operator は s390x または ppc64le で実行される際にそのバージョンを報告しなかったため、それらのアーキテクチャーへのインストールは正常に完了しませんでした。今回の修正により、Samples Operator がそのバージョンを正しく報告し、s390x または ppc64le へのインストールを防ぐことがなくなりました。(BZ#1779933)

- 以前のバージョンでは、最新の Java 11 イメージストリームタグはイメージストリームの詳細ページのバージョンに正しくリンクされませんでした。つまり、**ImageStreamTag** オブジェクトを Web コンソールから検査することができませんでした。今回の修正により、Java 11 の正しい **ImageStreamTag** 仕様を Web コンソールから適切に検査できるようになりました。(BZ#1778613)

- 以前のバージョンでは、コントローラーマネージャーが起動直後の、イメージストリームメタデータが更新される前にイメージストリームにアクセスした場合、イメージストリームのローカル参照設定が無視される可能性がありました。これにより、プライベートレジストリーでサポートされるイメージストリームへの要求が失敗していました。今回の更新により、コントローラーマネージャーはメタデータの初期化が完了した後にイメージストリームのキャッシュを更新するようになりました。これにより、起動直後でも正確なローカル参照イメージストリームポリシーが作成されます。(BZ#1775973)

ストレージ

- **storage-class** アノテーションがローカルストレージ Operator の **storageClassName** の代わりに使用される場合に、Pod はスケジュールされず、PVC は保留中のままになる可能性があります。今回の修正により、Kubernetes スケジューラーは、Pod およびその PVC の評価時に、**volume.beta.kubernetes.io/storage-class** および **PVC.Spec.StorageClassName** の両方をチェックするようになりした。StorageClass を参照するためにベータアノテーションを使用する Pod がスケジュールされ、これを実行できるようになりました。(BZ#1791786)

- 以前のバージョンでは、Kubernetes は、ボリュームマウントが CSI ドライバーによる完了前にタイムアウトしている際に Pod が削除されると CSI ボリュームをアンマウントしませんでした。これにより、Kubernetes が認識しない状態でボリュームがノードにマウントされました。その結果、ボリュームはその他の場所にマウントすることができませんでした。今回の修正により、Kubernetes は CSI ドライバーがタイムアウトや他の同様の一時的なエラーを返した後の最終的な成功またはエラーを待機するようになりました。Kubernetes はボリュームがマウントまたはアンマウントされたかどうかを認識し、Pod の削除時に発生する可能性のあるマウントをクリーンアップできるようになりました。(BZ#1745776)

テンプレート

- **--param-file** オプションを **oc new-app** または **oc process** などのテンプレート処理コマンドと共に使用する場合、ファイルのサイズが 64K を上回る場合、ファイルは完全に読み取られません。これにより、**--param-file** を使用した **oc** ベースのテンプレート処理が失敗します。OpenShift Container Platform では、**--process-file** で指定されるファイルのサイズをチェックし、ファイル全体を読み取るために使用されるパラメーターを拡張できます。64K を超えるファイルを参照する **--param-file** を指定した **oc** ベースのテンプレート処理が機能するようになりました。(BZ#1748061)

- 以前のバージョンでは、プロジェクト作成に使用される **new-app**/**new-build** サンプルのいずれかにより、FIPS 環境でエラーが発生することがありました。これはこれらのサンプルが FIPS に準拠していないために生じました。今回のリリースより、FIPS 準拠の **new-app**/**new-build** サンプルのみが新規プロジェクト作成に表示され、ユーザーは FIPS 環境でそれらのサンプルを使用できるようになりました。(BZ#1774318)

Web コンソール (Administrator パースペクティブ)

- 以前のバージョンでは、**cluster-admin** 権限を持たないユーザーに対して、OpenShift Console ビルドページの **Rebuild** アクションが誤って無効にされていました。今回のリリースよりこの問題は解決され、このアクションがビルドのクローン作成パーミッションを持つ通常ユーザーに対して適切に有効にされるようになりました。(BZ#1774842)

- 以前のバージョンでは、クラスター管理者のみがコンソールの YAML エディターで **ConsoleYAMLSample** リソースを使用して作成された YAML テンプレートのサンプルを表示できました。今回のリリースにより、すべてのユーザーがこれらのテンプレートを表示できるようになりました。(BZ#1783163)

- cron ジョブの一覧ページで、**Starting Deadlines Seconds** または **Concurrency Policy** フィールドによるソートが正しく実行されませんでした。今回のリリースにより、**sortField** が更新され、cron ジョブを適切にソートできるようになりました。(BZ#1787096)

- **Local Volumes** ページには、コンテンツの重複を発生させた条件により重複したコンテンツが表示されました。今回のリリースより、これらの条件はステータス記述子から削除され、コンテンツは重複しなくなりました。(BZ#1776131)

- **Role Bindings** ページには、プロジェクトのないユーザー向けのクリック不可能な **Create Binding** ボタンがありました。今回のリリースにより、**Create Binding** ボタンはプロジェクトを持たないユーザーに表示されなくなりました。(BZ#1785487)

- 以前のバージョンでは、OLM 記述子を持つ一部の必須フィールドには、Web コンソールの **Create Operand** フォームで必要な赤いアスタリスクが欠落していました。今回のリリースでは、必須フィールドすべてに適切にラベルが付けられるようになりました。(BZ#1779858)

- 以前のバージョンでは、Web コンソールでマシンまたはレプリカのカウント値に負の値を設定できました。今回のリリースでは、0 未満の値を設定することはできません。(BZ#1780367)

- 以前のバージョンでは、**PodRing** GUI コンポーネントには、**Deployment Config Page** ページでカウントが更新されてから **Actions** および **Edit Count** をクリックして同じページでこれを再度更新した場合に正確な Pod 数が反映されませんでした。たとえば、Pod 数が **Deployment Config Page** の **Edit Count** を使用して 5 から 10 Pod に増加し、ユーザーが **up**矢印を使用して **PodRing** コンポーネントの数を増やすと、**PodRing** カウンターは 10 から 11 ではなく 5 から 6 Pod に誤って増加しました。今回の更新により、**Deployment Config Page** の **Edit Count** を使用して変更が加えられると、**PodRing** コンポーネントが正確に更新された Pod 数を表示するようになりました。さらに、ユーザーが **PodRing** コンポーネントで **up** または **down** 矢印をクリックすると、Pod 数は正確に更新されます。(BZ#1787210)

- 以前のバージョンでは、ユーザーは Web コンソールの Installed Operators ページでクラスタースコープのオペランドを編集できませんでした。その代わりに、オペランド YAML エディターについての HTTP の **404 Not Found** クライアントエラーの応答コードが Web ブラウザーの発生しました。今回の更新により、Web コンソールでオペランド YAML エディターの新規 Web ブラウザーが正常に開けるようになり、これによりユーザーはクラスタースコープのオペランドを更新できるようになります。(BZ#1781246)

- 以前のバージョンでは、Web コンソールでは、**ConsoleExternalLogLink** URL テンプレートで複数回発生する変数のすべてのインスタンスを置き換えませんでした。その代わりに、変数の最初の式のみが置き換えられました。今回の更新により、コンソールがテンプレートの変数のすべてのインスタンスを正しい値に問題なく置き換えるようになりました。(BZ#1781827)

- 以前のバージョンでは、Web コンソールの **Operator Details** ページ上の **Subscription Overview** からの **InstallPlan** リソースへのリンクが破損していました。これにより、ユーザーが Web コンソールで **InstallPlan** を承認することは容易ではありませんでした。**InstallPlan** を

承認するリンク (**1 requires approval** の出力を表示するリンクなど) が予想通りに機能するようになりました。(BZ#1783651)

- 以前のバージョンでは、Web コンソールの **OperatorHub Details** ページの **Source** タブでユーザーがソース名を使用してフィルターするとエラーが発生していました。このフィルターは修正され、ユーザーがソース名を入力すると予想通りに機能するようになりました。(BZ#1786418)

- 以前のバージョンでは、Web コンソールの **Explore** ページの **Endpoints** リソースについての API ドキュメントがありませんでした。説明やスキーマ情報などを含む API ドキュメンテーションが **Endpoints** リソースで利用できるようになりました。(BZ#1794754)

- 以前のバージョンでは、無効な OLM 記述子が Operator によって設定されている場合、Web コンソールではオペランドを表示できませんでした。その結果、エラーが予想される Web コンソールページで発生していました。今回の更新により、無効な OLM 記述子が許容され、コンソールにオペランドの詳細が正しく表示されるようになりました。(BZ#1797727)

- 以前のバージョンでは、一部のステータス値にアイコンが関連付けられませんでした。そのため、アイコンと共に表示される値もあれば、表示されない値もありました。アイコンが定義され、それらのアイコンがすべての値と共に表示されるようになりました。(BZ#1780629)

- 以前のバージョンでは、コンソールはルーティングラベルの特殊文字を確認しないため、以下のエラーが発生する可能性がありました。

  > AlertmanagerFailedReload Alert:
  > Reloading Alertmanager's configuration has failed for openshift-monitoring/alertmanager-main-x.

  Create Receiver フォームでは、ラベル名が有効な文字のみに制限されるようになりました。(BZ#1784725)

- 以前のバージョンでは、Operator が無効な **K8sResourceLink** OLM 記述子を宣言した場合、Web コンソールは空のページを表示していました。コンソールが正しくない **K8sResourceLink** 記述子を許容することにより、空のページが表示されなくなりました。(BZ#1795407)

- 以前のバージョンでは、必要な Operator リソースを Web UI から変更しても、それらの Operator の YAML ファイルは更新されませんでした。YAML ファイルは予想通りに更新されるようになりました。(BZ#1797769)

- 以前のバージョンでは、アラートがサイレンスにされた後に、それらのアラートは通知ドロワーに保持されました。サイレンスにされたアラートは通知ドロワーに保持されなくなりました。(BZ#1808062)

- Web コンソールビルドのバグにより、特定のページのランタイムエラーが発生する場合がありました。このバグは修正されました。(BZ#1818978)

- Web コンソールのクエリーブラウザーの結果は、ハードコーディングされたソートでレンダリングされ、カスタムソートを使用した場合とは異なる結果がレンダリングされます。ハードコーディングされたソートが削除され、クエリー結果にカスタムソートが反映されるようになりました。(BZ#1808437)

- コンソールの **Compute → Machine Config Pools → Create Machine Config Pool**ボタンを使用して新規の **MachineConfigPool** を作成すると、ノードに一致しない **MachineConfigPool** が生成されます。これは、一致するノードを選択するために **spec.machineSelector** キーを使用するテンプレートによって生じます。ただし、このキーは API によって認識されません。

ノードを選択する正しいキーは **spec.nodeSelector** です。ノードを選択するためのキーが更新され、GUI に適切なノードに一致するマシンセレクターが表示されるようになりました。(BZ#1818944)

- 以前のバージョンでは、Web コンソール Pod ターミナルは Unicode 文字を正しく処理しませんでした。この問題は修正され、Unicode 文字が正しく表示されるようになりました。(BZ#1821285)
- 以前のバージョンでは、Web コンソールのワークロードページのボリューム表が、ページをスクロールすると部分的に非表示になる場合がありました。このバグは修正されています。(BZ#1822195)
- 以前のバージョンでは、レポートおよびレポートクエリーの作成に使用されるデフォルトのテンプレートは、**v1** ではなく **apiVersion v1alpha** を使用していました。アルファバージョンの番号が付けられたテンプレートは依然として機能しましたが、それらのケースサポートは処理されない可能性がありました。このテンプレートは **apiVersion: metering.openshift.io/v1** を使用するように更新されました。(BZ#1772694)
- Web コンソールの **Dashboard** をクリックして **Dashboard** 内でナビゲーションアイテムを選択すると、選択された複数のナビゲーションアイテムが強調表示されたままになります。今回のバグ修正により、複数のナビゲーションアイテムが同時に強調表示されなくなるように新たな CSS ルールが適用され、アクティブなナビゲーションアイテムのみが強調表示されるようになりました。(BZ#1774702)

Web コンソール (Developer パースペクティブ)

- Microsoft Edge ブラウザーは、スクロールに使用される機能を認識しませんでした。ログ画面で読み込みができず、エラーが出されました。スクリーンリーダーのサポートが有効になり、ログがレンダリングされるようになりました。(BZ#1777980)
- **Service** YAML ファイルなどの Serverless リソースは、**v1** ではなく、**v1beta1** として一覧表示されます。ただし、**v1beta1** は非推奨とされています。今回のバグ修正により、**apiVersion** は **v1** に更新されています。(BZ#1796421)
- トポロジーのサービスバインディング要求 (SBR) は、とポロジー ビュー内の **Revisions** に関連付けられます。そのため、新規リビジョンは関連付けられたシークレットを取得しません。SBR は knative サービスを通過することで、 シークレットが挿入され、そこから新規リビジョンが作成されるようになります。(BZ#1798288)
- **KnativeServing** リソース **serving.knative.dev** の API グループが非推奨となり、これは Serverless Operator バージョン 1.4 の **operator.knative.dev** に変更されました。Serverless Operator の次のリリースでは、**serving.knative.dev** は古くなります。(BZ#1800598)
- コンテナーイメージのデプロイメントでは、内部イメージストリームが knative に選択されている場合、knative は新規イメージストリームの作成を試行しますが、これは失敗する可能性があります。内部イメージストリームを knative サービスとしてデプロイすることはできません。内部イメージの選択用のイメージストリームはすでに存在するため、新規に作成する必要はありません。(BZ#1808280)
- Kubernetes デプロイメントでは、外部イメージレジストリーからのイメージに **openshift/hello-world:1.0** などのタグがある場合、タグは適用されていませんでした。ユーザーはタグを使用して外部イメージをインポートできませんでした。今回のバグ修正により、デプロイメントに適したタグが渡されるようになりました。(BZ#1801736
- 以前のバージョンでは、2 つの連続するロールアウトが失敗すると、**Topology** ビューは最後のアクティブなリビジョンを表示するのではなく、失敗した Pod を表示していました。今回のバグ修正により、ロールアウトが失敗すると、最後のアクティブなリビジョンが表示されるよう

になりました。(BZ#1760828)

- 以前のバージョンでは、namespace の既存イメージストリームはアプリケーションの作成時に検出されませんでした。これは、制限されたクラスター全体でのパーミッションを持つユーザーが **Add** ページの internal registry オプションから **Container Image → Image** 名を使用する場合に生じました。イメージストリームのフェッチロジックがクラスターレベルから namespace レベルに移動し、namespace へのパーミッションを持つユーザーがその namespace のイメージストリームを確認できるようになりました。(BZ#1784264)

- Knative サービスおよびリビジョンリソースには **Topology** ビューでバインディングまたはビジュアルなコネクターがないため、Knative ワークロードは他のワークロードに接続できませんでした。これらのリソースに **Topology** ビューのコネクターが使用されるようになり、他のワークロードに接続できるようになりました。(BZ#1779201)

- Eclipse Che Operator がインストールされ、設定されている場合、**Topology** ビューには Che アイコンではなく Git アイコンが表示されました。そのため、そのアイコンをクリックすると Che ワークスペースにアクセスできることがユーザーに示唆されませんでした。Che が設定されている場合に **Topology** ビューが正しく Che アイコンを表示するようになり、ユーザーが Che ワークスペースにアクセスするのが容易になりました。(BZ#1780338)

- **Topology** ビューが開いている間に CLI を使用して Knative サービスを作成すると、GUI エラーが発生しました。GUI エラーを発生させずにこのワークロードを処理するためのチェックが追加されました。(BZ#1781188)

- 内部レジストリー機能には、サーバー側のエラーが発生した際に不明なエラーメッセージが表示されました。エラーメッセージングが改善され、ユーザーが問題の原因を特定するために使用できるようになりました。(BZ#1787492)

1.6. 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

以下の表では、機能は以下のステータスでマークされています。

- **TP**: テクノロジープレビュー
- **GA**: 一般公開機能
- **-**: 利用不可の機能

表1.2 テクノロジープレビュートラッカー

機能	OCP 4.2	OCP 4.3	OCP 4.4
Prometheus クラスターモニタリング	GA	GA	GA
Precision Time Protocol (PTP)	-	TP	TP
ランタイム Pod の CRI-O	GA	GA	GA

機能	OCP 4.2	OCP 4.3	OCP 4.4
oc CLI プラグイン	TP	TP	TP
ネットワークポリシー	GA	GA	GA
Multus	GA	GA	GA
プロジェクト追加に関する新たなフロー	GA	GA	GA
検索カタログ	GA	GA	GA
Cron ジョブ	GA	GA	GA
Kubernetes デプロイメント	GA	GA	GA
ステートフルセット	GA	GA	GA
明示的なクォータ	GA	GA	GA
マウントオプション	GA	GA	GA
experimental-qos-reserved	TP	TP	TP
Pod sysctl	GA	GA	GA
外部プロジェクトトラフィックの静的 IP	GA	GA	GA
テンプレート完了の検出	GA	GA	GA
replicaSet	GA	GA	GA
Kubernetes リソースでのイメージストリームの使用	GA	GA	GA
デバイスマネージャー	GA	GA	GA
永続ボリュームのサイズ変更	GA	GA	GA
Huge Page	GA	GA	GA
CPU ピニング	GA	GA	GA
受付 Webhook	GA	GA	GA
AWS EFS の外部プロビジョナー	TP	TP	TP
Pod Unidler	TP	TP	TP

機能	OCP 4.2	OCP 4.3	OCP 4.4
一時ストレージの制限/要求	TP	TP	TP
Descheduler	-	-	TP
Podman	TP	TP	TP
Kuryr CNI プラグイン	TP	GA	GA
PID Namespace のコントロール共有	TP	TP	TP
クラスター管理者コンソール	GA	GA	GA
クラスターの自動スケーリング	GA	GA	GA
Container Storage Interface (CSI)	GA	GA	GA
Operator Lifecycle Manager	GA	GA	GA
Red Hat OpenShift Service Mesh	GA	GA	GA
完全に自動化された Egress IP	GA	GA	GA
Pod の優先順位とプリエンプション	GA	GA	GA
Dockerfiles のマルチステージビルド	GA	GA	GA
OVN-Kubernetes Pod ネットワークプロバイダー	TP	TP	TP
Prometheus に基づく HPA カスタムメトリクスアダプター	TP	TP	TP
マシンのヘルスチェック	TP	GA	GA
iSCSI を使用した永続ストレージ	TP	GA	GA
iSCSI での raw ブロック	TP	GA	GA
Cinder での raw ブロック	-	TP	TP
OperatorHub	GA	GA	GA
3 ノードのベアメタルデプロイメント	TP	TP	TP
SR-IOV ネットワーク Operator	TP	GA	GA
Helm CLI	-	TP	GA

機能	OCP 4.2	OCP 4.3	OCP 4.4
サービスバインディング	-	TP	TP
ログ転送	-	TP	TP
ユーザーワークロードの監視	-	TP	TP
OpenShift Serverless	TP	TP	GA
コンピュートノードトポロジーマネージャー	-	TP	TP
CSI ボリュームスナップショット	-	-	TP
CSI ボリュームのクローン作成	-	-	TP
CSI ボリューム拡張	-	-	TP
OpenShift Pipeline	-	-	TP
コスト管理	TP	GA	GA

1.7. 既知の問題

- Day 2 のプロキシーサポート対応に関して Machine Config Operator (MCO) で問題が生じます。既存のプロキシーされていないクラスターがプロキシーを使用するように再設定されるタイミングが記述されます。MCO は設定マップの新たに設定されたプロキシー CA 証明書を RHCOS 信頼バンドルに適用する必要がありますが、これが正常に機能しません。回避策として、プロキシー CA 証明書を信頼バンドルに手動で追加してから、信頼バンドルを更新する必要があります。

```
$ cp /opt/registry/certs/<my_root_ca>.crt /etc/pki/ca-trust/source/anchors/
$ update-ca-trust extract
$ oc adm drain <node>
$ systemctl reboot
```

(BZ#1784201)

- 自己署名の Red Hat OpenStack Platform (RHOSP) 16 クラスターを使用する場合には、内部イメージレジストリーからプルまたはこれにプッシュすることはできません。回避策として、**configs.imageregistry/cluster** リソースで **spec.disableRedirect** を **true** に設定する必要があります。これにより、クライアントは Swift からの直接のリンクからではなく、イメージレジストリーからイメージ階層をプルできるようになります。(BZ#1810461)

- クラスタープロキシー設定 **HTTP_PROXY** は OpenShift Container Platform コンポーネントでのみ利用でき、ユーザーアプリケーションでは使用できません。回避策として、以下のコマンドを実行してユーザーアプリケーションのクラスタープロキシー設定を有効にする必要があります。

```
$ oc set env dc/jenkins \
```

```
http_proxy=$(oc get proxy cluster -o jsonpath='{.status.httpProxy}') \
https_proxy=$(oc get proxy cluster -o jsonpath='{.status.httpsProxy}') \
no_proxy=$(oc get proxy cluster -o jsonpath='{.status.noProxy}')
```

(BZ#1780125)

- HTTPS プロキシーを通過する **git clone** 操作は失敗します。HTTP プロキシーは問題なく使用できます。(BZ#1750650)
- ソース URI が **git://** または **ssh://** スキームを使用する場合、すべての **git clone** 操作はプロキシーの背後で実行されているビルドで失敗します。(BZ#1751738)
- ミラーを使用してイメージをビルドする際に、ミラーレジストリーのプルシークレットがビルダーのサービスアカウントにのみリンクする場合はビルドに失敗します。プルシークレットもビルド設定オブジェクトにリンクする必要があります。(BZ#1810904)
- Kuryr を使用する Red Hat OpenStack Platform (RHOSP) 13 では、FIPS が無効にされている場合、サービスカタログを有効にすることはできません。Service Catalog のコントローラーマネージャーおよび API サーバーコンポーネントの Pod は **CrashLoopBackOff** のステータスを表示します。これは、**https://etcd.openshift-etcd.svc.cluster.local:2379** URL が常に解決しないためです。OpenShift Container Platform 4 で etcd クラスター URL を取得する新たな手法を使用できます。(BZ#1821589)
- Kuryr を使用した RHOSP 16 のインストールは、初期設定後の **ovn_controller** のクラッシュにより機能しません。(BZ#1812009, BZ#1818844)
- Red Hat Virtualization (RHV) マシンの **instance-state** アノテーションおよび **providerStatus.instanceState** ステータスは常に一致しているとは限りません。この不一致がある場合、クライアントは失敗するか、または RHV マシンのステータスに誤ったパッチを適用します。(BZ#1815394)
- RHV でマシンセットをスケールアップする場合、新規マシンは **Provisioned** フェーズで終了できません。これにより、マシンは実行されなくなります。(BZ#1815435, BZ#1817853)
- クラスターリソースの計算エラーにより、RHV での OpenShift Container Platform クラスターの自動スケーリングは失敗します。(BZ#1822118)
- Firefox ブラウザーを使用して **Topology** ビューでノードまたはノードのグループを選択すると、関連するすべてのラベルとノードの背景が透過的になります。(BZ#1822337)
- **Topology** ビューで、ユーザーがノードまたはワークロードを選択し、サイドパネルで **Monitoring → View monitoring dashboard** をクリックすると、ユーザーには特定のワークロードのモニタリングダッシュボードが表示されます。このフィルターされたワークロードダッシュボードビューには明確に名前が付けられていないため、すべてのワークロードのメトリクスを表示する汎用ダッシュボードと混乱する場合があります。(BZ#1822331)
- ピリオド (.) などの無効な文字が **Developer** パースペクティブから serverless トラフィック分散タグに入力される場合、トラフィック分散機能は機能を停止します。タグに使用できない文字の入力を防ぐエラーメッセージは表示されません。(BZ#1822344)
- アイデンティティープロバイダー (IDP) によるユーザーの認証に 60 秒を超える時間がかかると、他の IDP を試行する前に認証が失敗する可能性があります。回避策として、問題のある IDP を IDP の一覧から削除し、ユーザーが正常な IDP を使用して認証できるようにできます。(BZ#1826484)

- クラスターロギングをバージョン 4.3 から 4.4 に更新する場合、Elasticsearch Pod は **CrashLoopBackOff** ステータスのままになる可能性があります。この問題の回避策として、Elasticsearch デプロイメントを順番に削除することができます。(BZ#1824006)

- OpenShift Container Platform 4.4 には v.4.4 メータリング Operator が同梱されていません。V.4.3 メータリング Operator を OpenShift Container Platform 4.4 クラスターにインストールするか、または引き続き実行できます。(BZ#1829035)

- OpenShift Container Platform クラスターをバージョン 4.3 から 4.4 に更新する場合、etcd Operator は動作が低下した状態のためにアップグレードに失敗することがあります。これは、**InstallerPod** の失敗によって生じます。回避策として、**InstallerPod** の失敗を解決できるように etcd で新規リビジョンを強制的に実行する必要があります。これにより、etcd Operator をリカバリーできます。

  1. etcd で新規リビジョンを強制的に実行します。

     ```
     $ oc patch etcd cluster -p='{"spec": {"forceRedeploymentReason": "recovery-"$( date --rfc-3339=ns )""}}' --type=merge
     ```

  2. ノードが最新のリビジョンにあることを確認します。

     ```
     $ oc get etcd '-o=jsonpath={range .items[0].status.conditions[?(@.type=="NodeInstallerProgressing")]}{.reason}{"\n"}{.message}{"\n"}'
     ```

  (BZ#1830789)

- Web コンソールダウンロードのデプロイメントは、 **ipv6.disable=1** で設定されるノードで失敗します。(BZ1795325)

- **Topology Manager** の問題により、Guaranteed QoS Pod が同じノードで同時に作成されると、NUMA リソースが同じ NUMA ノードに配置されない可能性がありました。その結果、Pod 仕様で要求されるリソースが NUMA で配置されない可能性があります。
  4.4 でこの問題を回避するには、ノード上の Guaranteed QoS を持つ複数の Pod を同時にスピンアップしないでください。

  この問題が発生する場合には、Pod を削除してから再作成します。(BZ#1834979)

- **runStrategy** 属性が **Manual** に設定されている仮想マシンは仮想マシンが実行中か、または停止しているかどうかを示さず、Web コンソールは仮想マシンが実行していると誤って想定します。この問題を回避するには、Web コンソールで仮想マシンを使用する際に **runStrategy** を **Manual** に設定しないでください。代わりに、**running** 属性を使用するか、または **runStrategy** を **Always**、 **RerunOnFailure**、または **Halted** に設定します。(BZ#1834717)

- 新規 OpenShift Container Platform z-stream リリースにアップグレードする場合、ノードがアップグレードされると API サーバーへの接続が中断され、API 要求が失敗する可能性があります。(BZ#1791162)

- 新規 OpenShift Container Platform z-stream リリースにアップグレードする場合、ルーター Pod が更新されているためにルーターへの接続が中断される可能性があります。アップグレードの期間中、一部のアプリケーションには常に到達できなくなる可能性があります。(BZ#1809665)

- OAuth トークンの期限切れに関連する問題が原因で、Kibana コンソールに **security_exception** エラーが表示され、Kibana インデックスにアクセスできなくなる可能性があります。このエラーが表示された場合、Kibana コンソールからログアウトしてから再度ログインします。これにより OAuth トークンが更新され、インデックスにアクセスできるはずです。(BZ#1791837)

- HTTPS プロキシーを通過する git clone 操作は失敗します。非 TLS (HTTP) プロキシーは問題なく使用できます。(BZ#1750650)

- ソース URI が **git://** または **ssh://** スキームを使用する場合、git clone 操作はプロキシーの背後で実行されているビルドで失敗します。(BZ#1751738)

- 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 4.4 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。特定の必要がなければ、認証されていないアクセスを無効にすることが推奨されます。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)

1.8. エラータの非同期更新

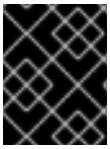
OpenShift Container Platform 4.4 のセキュリティー、バグ修正、拡張機能の更新は、Red Hat Network 経由で非同期エラータとして発表されます。OpenShift Container Platform 4.4 のすべてのエラータは Red Hat カスタマーポータルから入手 できます。非同期エラータについては、OpenShift Container Platform ライフサイクル を参照してください。

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。エラータの通知を有効にすると、登録している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 4.4 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 4.4.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 のいずれのリリースについても、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。

1.8.1. RHBA-2020:0581 - OpenShift Container Platform 4.4 イメージリリースおよびバグ修正アドバイザリー

発行日: 2020-05-04

OpenShift Container Platform リリース 4.4 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2020:0581 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、 RHBA-2020:0582 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。

OpenShift Container Platform 4.4.3 コンテナーイメージの一覧

1.8.2. RHSA-2020:1936 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-05-04

haproxy の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:1936 アドバイザリーに記載されています。

1.8.3. RHSA-2020:1937 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-05-04

cri-o の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:1937 アドバイザリーに記載されています。

1.8.4. RHSA-2020:1938 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-05-04

hadoop-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:1938 アドバイザリーに記載されています。

1.8.5. RHSA-2020:1939 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-05-04

ose-machine-config-operator-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:1939 アドバイザリーに記載されています。

1.8.6. RHSA-2020:1940 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-05-04

ose-cluster-policy-controller-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:1940 アドバイザリーに記載されています。

1.8.7. RHSA-2020:1942 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-05-04

presto-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:1942 アドバイザリーに記載されています。

1.8.8. RHBA-2020:2133 - OpenShift Container Platform 4.4.4 バグ修正の更新

発行日: 2020-05-18

OpenShift Container Platform リリース 4.4.4 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2020:2133 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、 RHBA-2020:2132 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。

OpenShift Container Platform 4.4.4 コンテナーイメージの一覧

1.8.8.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.9. RHSA-2020:2136 - Important (重要): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-05-18

cluster-image-registry-operator の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2136 アドバイザリーに記載されています。

1.8.10. RHBA-2020:2180 - OpenShift Container Platform 4.4.5 バグ修正の更新

発行日: 2020-05-26

OpenShift Container Platform リリース 4.4.5 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2020:2180 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、RHBA-2020:2179 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。

OpenShift Container Platform 4.4.5 コンテナーイメージの一覧

1.8.10.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.11. RHBA-2020:2310 - OpenShift Container Platform 4.4.6 バグ修正の更新

発行日: 2020-06-01

OpenShift Container Platform リリース 4.4.6 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2020:2310 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、 RHBA-2020:2309 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。

OpenShift Container Platform 4.4.6 コンテナーイメージの一覧

1.8.11.1. バグ修正

- 以前のバージョンでは、Samples Operator の非接続サポートの実装により、一部のユーザーに問題が発生していました。この実装では、CVO ベースの Jenkins イメージストリームに対する **samplesRegistry** の上書きの適用が許可されました。これにより、他のシナリオで **samplesRegistry** の上書きを使用することがより困難になりました。今回の更新により、**samplesRegistry** の上書きが必要なくなり、以前の実装に関連する問題を回避できるようになりました。(BZ#1824280)

- 以前のバージョンでは、**Installed Operators** ページのオペランド一覧で、条件に **status: true** が含まれないカスタムリソース **status.conditions** のステータスが表示される可能性がありました。つまり、表示されるステータスが正しくない可能性がありました。Web コンソールには、**status: true** の条件のステータスのみが表示されるようになりました。(BZ#1829591)

- 以前のバージョンでは、以前のリリースからのサンプルテンプレートが後続のリリースで削除されると、見つからないテンプレートが更新を必要とするものとして誤って追跡される場合に、後続のリリースへのアップグレードが失敗する可能性がありました。今回のリリースより、アップグレードプロセスで、以前のリリースに存在していたが、アップグレードするリリースには存在しないテンプレートの更新は試行されなくなりました。(BZ#1832344)

- 以前のバージョンでは、Cluster Version Operator が HTTPS 経由でメトリクスを提供しようとし、TLS キーおよび証明書が見つからない場合、アクションが失敗しました。今回の更新により、モニタリング Operator は TLS キーおよび証明書を作成し、Cluster Version Operator が HTTPS 経由でメトリクスを提供できるようになりました。(BZ#1835483)

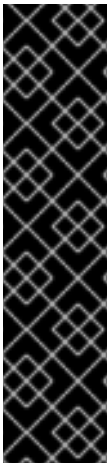
- 以前のバージョンでは、ワーカーまたはマスターの仮想マシン仕様をカスタマイズする唯一の方法は、インストール前に RHV または oVirt テンプレートをカスタマイズし、インストーラーがそのテンプレートを使用できるように環境変数を設定する方法でした。今回のリリースにより、**MachinePool** オブジェクトがこのプラットフォーム用に実装され、**install-config.yaml**

ファイルで公開されるようになりました。ワーカーおよびマスター仮想マシンインスタンスは、**MachinePool** 設定に含まれる仕様で作成されるようになりました。異なるディスクサイズに対応するようになったため、デフォルトのディスクサイズは 120 GB に更新されます。(BZ#1835795)

- 以前のバージョンでは、クォータの誤った動作により、Pod のデプロイやビルドが失敗する可能性がありました。今回のリリースにより、ビルドコントローラーが更新され、クォータがビルドについて適切に処理されるようになりました。(BZ#1835913)
- 以前のバージョンでは、ユーザーが CLI または YAML を使用して **PipelineRun** オブジェクトを作成すると、Web コンソールは応答しなくなりました。今回の更新により、Web コンソールのエラーを回避するためにチェックが追加されました。(BZ#1838792)

1.8.11.2. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。



重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.12. RHBA-2020:2445 - OpenShift Container Platform 4.4.8 バグ修正の更新

発行日: 2020-06-15

OpenShift Container Platform リリース 4.4.8 が公開されました。この更新に含まれるバグ修正の一覧は、 RHBA-2020:2445 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、 RHBA-2020:2444 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。

OpenShift Container Platform 4.4.8 コンテナーイメージの一覧

1.8.12.1. 機能

1.8.12.1.1. コントロールプレーンの証明書の自動リカバリー

OpenShift Container Platform はコントロールプレーン証明書の期限切れの状態から自動的にリカバリーできるようになりました。例外として、kubelet 証明書を回復するために保留状態の **node-bootstrapper** 証明書署名要求 (CSR) を手動で承認する必要があります。

詳細は、Recovering from expired control plane certificates を参照してください。

1.8.12.2. バグ修正

- 以前のバージョンでは、Red Hat Virtualization (RHV) に OpenShift Container Platform クラスターを作成する前に、インストールプログラムでは、ユーザーは仮想マシンテンプレートを手動で作成する必要がありました。これは、インストールプログラムが RHV バージョン 4.3.9 で以下の要件を満たしていないためです。

  - インストールプログラムは ignition を仮想マシンに渡す必要があります。
  - テンプレートは、その OS タイプを Red Hat Enterprise Linux CoreOS (RHCOS) として指定する必要があります。

  インストールプログラムは、RHCOS を OS タイプとして指定するテンプレートを作成し、Ignition を仮想マシンに渡します。ユーザーが仮想マシンテンプレートを作成する必要がなくなりました。(BZ#1821638)

- 以前のバージョンでは、Elasticsearch Operator および Cluster Logging Operator は Fluentd の挿入された CA Bundle の内容を調整しませんでした。そのため、Fluentd および Kibana には挿入された CA バンドルでの設定マップへのボリュームマウントがありませんでした。設定マップの内容は調整時にフェッチされ、ボリュームのマウントが可能になりました。これにより、Fluentd および Kibana は CA バンドル設定マップを適切にマウントし、認定が再び機能するようになりました。(BZ#1833288)

- **oc adm node drain** コマンドは、OpenShift Container Platform コードの正しくない状態により。ノードのドレイン (解放) 時にデーモンセットおよび Pod に割り当てられるローカルデータに適切に対応しませんでした。ロジックが修正され、ノードのドレイン (解放) 時にすべての Pod が考慮されるようになりました。デーモンセットの Pod が実行中のノードをドレイン (解放) しようとする場合や、Pod にローカルボリュームデータが割り当てられている場合、**oc adm node drain** コマンドが失敗し、この 2 つのケースを無視するフラグを使用するように指示します。(BZ#1835739)

- 以前のバージョンでは、Web コンソールでは YAML ファイルの編集を試行する際に Safari Web ブラウザーの JavaScript 例外により、YAML 編集ページが読み込まれなくなりました。このバグは修正され、YAML ファイルの編集が Safari Web ブラウザーで機能するようになりました。(BZ#1838815)

- 以前のバージョンでは、Web コンソールの **Installed Operators** 列では、インストールされた Operator にサブスクリプションがあることが前提とされました。Package Server Operator にはサブスクリプションがないため、そのステータスは存在する場合でも誤って削除済みとして表示されました。Package Server Operator のステータスは、サブスクリプションステータスに依存しないように修正されたため、Installed Operators ページに正しく表示されるようになりました。(BZ#1840647)

- Web コンソールの **Developer** パースペクティブから **Advanced → Project Details → Inventory** セクションに移動する場合、デプロイメント設定は一覧表示されませんでした。デプロイメント設定は追跡され、ダッシュボードの **Inventory** セクションに含まれるようになりました。(BZ#1825975)

- 以前のバージョンでは、Pod ログページには、選択されたコンテナーを示すクエリー文字列パラメーターが含まれませんでした。これにより、複数のコンテナーを持つ Pod がページの更新時やページの URL にアクセスする際に正しくないコンテナーログを報告していました。新規のクエリー文字列パラメーターが追加され、URL は表示されるコンテナーのログを示します。(BZ#1827197)

- 以前のバージョンでは、Web コンソールには、**Search** ページに OLM Subscription を一覧表示する際に名前、namespace、および作成日のみが表示されました。Web コンソールには、追加の OLM Subscription の詳細が表示されるようになりました。(BZ#1827746)

- コントロールプレーン証明書の期限切れの状態からのリカバリー時に、クラスターはポート 7443 のリカバリー API サーバーに接続できません。これは、リカバリー API サーバーのポートが OpenStack、oVirt、ベアメタル、および vSphere に使用される HAProxy ポートと競合するためです。これにより、**Unable to connect to the server: x509: certificate signed by unknown authority** エラーが生じます。HAProxy はポート 9443 でリッスンするようになり、リカバリー API サーバーはポート 7443 を使用して期限切れのコントロールプレーン証明書のリカバリープロセスを容易にします。(BZ#1831008)

- Cloud Credential Operator (CCO) には、クラウドルート認証情報を必要とする **CredentialsRequest** CR についての特別なケース処理がありました。クラウドルート認証情報が見つからない場合、CCO は独自の読み取り専用 **CredentialsRequest** CR を調整できませんでした。これは、読み取り専用の **CredentialsRequest** 認証情報を使用して読み取り専用 **CredentialsRequest** を検証することにより修正されました。クラウドルート認証情報を削除しても、CCO のパフォーマンスが低下しなくなりました。(BZ#1838810)

- 以前のバージョンでは、関連付けられた Task を持たない Pipeline Builder で **Task** バブルをクリックすると、空の画面が表示されました。ノードを一覧ノードに変換してこれを変更できるようになり、これは修正されました。**Task** または **ClusterTask** リソースにポイントしなくなったタスク参照を更新できるようになりました。(BZ#1840954)

- Sample Operator ファイルシステムのエラーは、Cluster Operator の reason フィールドで API サーバーのエラーとして誤って報告されました。また、API サーバーオブジェクトを操作する際の実際の API サーバーエラーの詳細には、実際の障害タイプの詳細が含まれませんでした。このため、パフォーマンス低下について誤ったエラーが報告されました。Sample Operator ファイルシステムのエラーは、degraded reason フィールドでファイルシステムエラーとして報告されるようになり、degrade reason フィールドで報告される API サーバーエラーには特定のエラータイプが含まれるようになりました。(BZ#1842560)

- Cluster Version Operator (CVO) には競合状態があり、この場合にタイムアウトした更新の調整サイクルが成功した更新と見なされていました。これは、Operator でリリースイメージ署名の取得の試行がタイムアウトしたネットワークが制限されたクラスターについてのみ生じました。このバグにより、CVO が shuffled-manifest 調整モードに入りました。このモードでは、コンポーネントが処理できない順序でマニフェストが適用されるとクラスターが破損する可能性がありました。CVO はタイムアウトした更新を失敗として処理するようになろ。更新が正常に実行される前に調整モードに入らなくなりました。(BZ#1843732)

- 以前のバージョンでは、Azure で実行されているクラスターの RHEL 8 仮想マシンがネットワーク接続を失うことがありました。これは、RHEL の Hyper-V netvsc ドライバーの不具合によって生じました。このバグは RHEL で修正され、クラスターで使用される RHEL 仮想マシンで修正を利用できるようになりました。その結果、Azure で実行されるクラスターでは、netvsc ドライバーの不具合によりネットワーク接続の問題が発生しなくなりました。(BZ#1841900)

今回の更新により、以下の拡張機能も導入されています。

- **oc adm release mirror** コマンドを拡張するために OpenShift Container Platform の拡張機能が追加されました。クラスターのアップグレードは、インターネットへのアクティブな接続のないクラスターで実行できます。以前のバージョンでは、イメージ更新の検証に必要な署名データが含まれる設定マップを作成するために、手動の手順が必要でした。このコマンドは、Cluster Version Operator がミラーリングされたリリースを検証するために使用するリリースイメージ署名が含まれる設定マップマニフェストを自動的に作成し、適用するようになりました。(BZ#1837675)

1.8.12.3. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.13. RHSA-2020:2403 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-06-15

containernetworking-plugins の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2403 アドバイザリーに記載されています。

1.8.14. RHSA-2020:2448 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-06-15

openshift の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2448 アドバイザリーに記載されています。

1.8.15. RHSA-2020:2449 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-06-15

openshift-enterprise-hyperkube-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2449 アドバイザリーに記載されています。

1.8.16. RHBA-2020:2580 - OpenShift Container Platform 4.4.9 バグ修正の更新

発行日: 2020-06-22

OpenShift Container Platform リリース 4.4.9 が公開されました。この更新に含まれるバグ修正の一覧は、 RHBA-2020:2580 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、RHBA-2020:2579 および RHEA-2020:2623 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。

OpenShift Container Platform 4.4.9 コンテナーイメージの一覧

1.8.16.1. 機能

1.8.16.1.1. Node.js Jenkins Agent v10 および v12 を追加

jenkins-agent-nodejs-10-rhel7 および **jenkins-agent-nodejs-12-rhel7** イメージが OpenShift Container Platform に追加されました。これらの新規イメージにより、Jenkins Pipeline は Node.js Jenkins エージェントの v10 または v12 のいずれかを使用するためにアップグレードできます。Node.js v8 Jenkins エージェントは非推奨となりましたが、引き続き提供されます。既存のクラスターの場合、Node.js Jenkins エージェントを手動でアップグレードする必要があります。これは namespace ごとに実行できます。手動アップグレードを実行するには、以下の手順に従います。

1. Jenkins Pipeline をアップグレードするプロジェクトを選択します。

```
$ oc project <project_name>
```

2. 新規 Node.js Jenkins Agent イメージをインポートします。

```
$ oc import-image nodejs openshift4/jenkins-agent-nodejs-10-rhel7 --
from=registry.redhat.io/openshift4/jenkins-agent-nodejs-10-rhel7 --confirm
```

   このコマンドは、v10 イメージをインポートします。v12 を選択する場合は、それに応じてイメージの仕様を更新します。

3. 現在の Node.js Jenkins Agent を、インポートした新規の Node.js Jenkins Agent で上書きします。

```
$ oc label is nodejs role=jenkins-slave --overwrite
```

4. Jenkins ログで、新規の Jenkins Agent テンプレートが設定されていることを確認します。

```
$ oc logs -f jenkins-1-<pod>
```

詳細は、Jenkins エージェント について参照してください。

1.8.16.1.2. IBM Power Systems

本リリースでは、IBM Power Systems は OpenShift Container Platform 4.4 と互換性があります。IBM Power へのクラスターのインストール、または 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 について参照してください。

制限
IBM Power の OpenShift Container Platform については、以下の制限に注意してください。

- IBM Power Systems 向けの OpenShift Container Platform には、以下のテクノロジープレビュー機能が含まれていません。
  - Container-native virtualization (CNV)
  - OpenShift Serverless
- 以下の OpenShift Container Platform 機能はサポートされていません。
  - Red Hat OpenShift Service Mesh
  - OpenShift Do (**odo**)

    - CodeReady Containers (CRC)
    - Tekton をベースとする OpenShift Pipeline
    - OpenShift Container Platform Metering
    - SR-IOV CNI プラグイン
- ワーカーノードは 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、Network File System (NFS)、OpenStack Cinder、または Container Storage Interface (CSI) を使用する **Filesystem** モードである必要があります。
- ネットワークは、Red Hat OpenShift SDN で DHCP または静的アドレス指定のいずれかを使用する必要があります。

1.8.16.1.3. IBM Z および LinuxONE

本リリースでは、IBM Z および LinuxONE は OpenShift Container Platform 4.4 と互換性があります。インストール手順については、IBM Z および LinuxONE へのクラスターのインストール について参照してください。

制限
IBM Z および LinuxONE の OpenShift Container Platform については、以下の制限に注意してください。

- IBM Z 向けの OpenShift Container Platform には、以下のテクノロジープレビューが含まれていません。
    - Container-native virtualization (CNV)
    - ログ転送
    - Precision Time Protocol (PTP) ハードウェア
    - CSI ボリュームスナップショット
    - CSI ボリュームのクローン作成
    - OpenShift Pipeline
- 以下の OpenShift Container Platform 機能はサポートされていません。
    - Red Hat OpenShift Service Mesh
    - OpenShift Do (**odo**)
    - CodeReady Containers (CRC)
    - OpenShift Container Platform Metering
    - Multus CNI プラグイン
    - OpenShift Container Platform アップグレードの段階的ロールアウト
    - FIPS 暗号
    - etcd に保存されるデータの暗号化

- 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
- 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
- OpenShift Serverless
- Helm コマンドラインインターフェイス (CLI) ツール
- 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
- etcd クラスター Operator

- ワーカーノードは 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のタイプは Filesystem: NFS である必要があります。
- これらの機能は 4.4 の場合に IBM Z での OpenShift Container Platform に利用できますが、x86 での OpenShift Container Platform 4.4 には利用できません。
  - IBM System Z 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)。

1.8.16.2. バグ修正

- 仮想マシンおよび仮想マシンテンプレートウィザードでは、**virtIO** が CD-ROM を割り当てる際にデフォルトのインターフェイスになります。ただし、**virtIO** CD-ROM は仮想マシンの検証をパスせず、これを作成できません。回避策として、仮想マシンおよび仮想マシンテンプレートを作成する際に、**SATA** を CD-ROM インターフェイスとして選択します。(BZ#1817394)
- 以前のバージョンでは、Kibana ダッシュボードからログアウトする際に、ログイン認証情報を指定せずに新規ブラウザータブから再度ログインすることができました。これは、Kibana のセキュリティーを提供する OAuth プロキシーの正しくないハンドラーをポイントするサインオフリンクによって生じました。サインオフリンクが修正され、Kibana ダッシュボードへの再アクセスを試行する際に、ログイン認証情報の使用が強制されるようになりました。(BZ#1823305)
- **Installed Operators** ページの **View more** リンクが、正しいページにリンクされるようになりました。(BZ#1824255)
- 今回のリリースにより、Web コンソールのオペランドビューの **Status** 列が更新され、**Details** および **YAML** ビューで利用可能な最新のステータスが表示されるようになりました。(BZ#1831808)
- 今回のリリースにより、**eventSources** API グループは、最新のサポートされている API グループ **sources.knative.dev** に更新されます。今回の更新により、新規の API グループによって生成されたソースが Web コンソールの **Topology** ビューで認識されるようになりました。(BZ#1836807)
- 以前のバージョンでは、Knative サービスの環境変数は Web コンソールの **Developer** パースペクティブの **Add** ビューから指定できませんでした。そのため、環境変数を必要とするアプリケーションは予想通りに機能しない可能性があります。今回のリリースにより、ユーザーは **Add** ビューから環境変数を追加できるようになりました。(BZ#1839115)
- 以前のバージョンでは、ユーザー名に **#** などの特殊文字が含まれる場合に、Web コンソールにユーザーの詳細は表示されませんでした。Web コンソールには、ユーザー名の特殊文字に関係なくユーザーの詳細が表示されるようになりました。(BZ#1840812)

- Octavia を OpenStack 13 から 16 にアップグレードすると、UDP リスナーがサポートされ、TCP プロトコルで DNS 解決を実行するストラテジーが削除されます。この変更では、UDP プロトコルを指定する既存の DNS サービスに新しいリスナーを追加する必要があります。既存の DNS ロードバランサーの古い Amphora イメージは新規のリスナーをサポートしないため、リスナーの作成は失敗します。今回のリリースにより、UDP を必要とする DNS サービスが再作成され、ロードバランサーが新規の Amphora バージョンで再作成されるようになりました。サービスとロードバランサーを再作成すると、DNS 解決のダウンタイムが発生します。このプロセスが完了すると、DNS サービスのロードバランサーが必要なリスナーすべてと共に作成されます。(BZ#1841029)

- 以前のバージョンでは、Web コンソールで **operatorframework.io/cluster-monitoring=true** アノテーションが **true** に設定されている場合に、OpenShift Monitoring Prometheus Operator が Operator メトリクスを収集するために必要なロールバインディングが作成されませんでした。この問題は本リリースで解決されています。(BZ#1841149)

- 以前のバージョンでは、Autoscaler で Node および Machine オブジェクトでのプロバイダー ID が完全に一致する必要があり、Machine 設定がリソースグループ名を大文字/小文字が混在する状態で指定している場合、ID は完全一致となりませんでした。この場合、Autoscaler はマシンにノードがあることを認識せず、15 分後にマシンを終了しました。今回のリリースにより、Autoscaler はプロバイダー ID の文字の大文字/小文字を無視し、大文字/小文字に関係なく一致するようになりました。(BZ#1841478)

- Azure プラットフォームでは、Pod のボリュームマウントを作成するために **cifs-utils** パッケージが必要になります。今回のリリースにより、OpenShift Container Platform のインストール時に RHEL 7 ホスト用にインストールされるパッケージに **cifs-utils** が含まれます。(BZ#1845819)

- 以前のバージョンでは、SELinux パーミッションは Azure プラットフォームでマウントされたボリュームへの読み取り/書き込みアクセスをブロックしました。今回のリリースにより、SELinux ブール値は RHCOS 8.x に一致するように更新され、適切なアクセスを許可するようになりました。(BZ#1845830)

1.8.16.3. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.17. RHSA-2020:2583 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-06-22

python-psutil の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2583 アドバイザリーに記載されています。

1.8.18. RHBA-2020:2713 - OpenShift Container Platform 4.4.10 バグ修正の更新

発行日: 2020-06-29

OpenShift Container Platform リリース 4.4.10 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2020:2713 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、 RHBA-2020:2734 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。

OpenShift Container Platform 4.4.10 コンテナーイメージの一覧

1.8.18.1. バグ修正

- 以前のバージョンでは、Ingress コントローラーは、HTTP 要求をアプリケーションに転送する際に標準以外の **proto-version** パラメーターを持つ **Forwarded** HTTP ヘッダーを追加しました。この標準以外のヘッダーでは、アプリケーションがヘッダーの値を解析しようとする際に問題が発生しました。今回のリリースにより、Ingress コントローラーは **Forwarded** ヘッダーに **proto-version** パラメーターを指定しなくなりました。(BZ#1816544)

- 以前のバージョンでは、MachineHealthCheck の **maxUnhealthy** フィールドの値には、複数の値形式 (例: **10**、**"10"**、または **"10%"**) が許可されました。引用符で囲まれた値は、パーセンテージ記号が含まれていなくてもパーセント値として解釈されました。**maxUnhealthy** 値の解釈は、ユーザーの意図に一致しない可能性があり、マシンは意図されない場合に修正されたり、意図される場合に修正されない可能性がありました。今回のリリースにより、パーセンテージ記号が含まれる値のみがパーセンテージとして解釈されるようになりました。たとえば、**10** および **"10"** は 10% ではなく、同じ値として解釈されるようになりました。(BZ#1816606)

- 以前のバージョンでは、メトリクスは IPv6 アドレスでバインドされず、収集できませんでした。今回のリリースにより、メトリクスは IPv6 アドレスで正常に収集できるようになりました。(BZ#1819770)

- 以前のバージョンでは、Edit ClusterServiceVersion は **ClusterServiceVersion** の Actions メニューに表示されました。これにより、ユーザーに **ClusterServiceVersion** を編集する必要があるかのような正しくない印象を与えました。今回のリリースにより、Edit ClusterServiceVersion が **ClusterServiceVersion** オブジェクトの Actions メニューから削除されました。(BZ#1827306)

- 以前のバージョンでは、リソースがない場合に、一部のリソースに空のステータス **None** が含まれていませんでした。この状態がない場合、他のリソースとの一貫性がなくなり、曖昧さが生じました。今回のリリースにより、リソースがない場合にこれらのこれらのリソースに空の状態の **None** が含まれるようになりました。(BZ#1827766)

- 以前のバージョンでは、マストヘッドのドロップダウン項目間のスペースが必要以上に多く設定されていました。この表示の問題は解決されています。(BZ#1829030)

- 以前のバージョンでは、Pipeline Builder は、空の文字列 ('') のデフォルト値をデフォルトなしとして誤って解釈していました。ただし、Operator が提供する一部のタスクは、空の文字列をデフォルト値として指定しないと機能できません。今回のリリースにより、OpenShift Pipeline Operator が空の文字列を含め、デフォルト値として許可する値が有効なデフォルト値として認識されるようになりました。(BZ#1829568)

- 以前のバージョンでは、OpenShift Controller Manager Operator 内のコントローラーは名前付きのワークキューを使用しておらず、**workqueue_depth** などの一部のメトリクスが Prometheus に表示されませんでした。今回のリリースにより、コントローラーは名前付きのワークキューを使用し、これらのメトリクスが Prometheus に表示されるようになりました。(BZ#1832839)

- 以前のバージョンでは、OpenShift Container Platform の Ironic コンテナーは、ユーザー定義の **PROV_IFACE** インターフェイスが IPv6 用に設定され、グローバルにルーティング可能なアドレス指定ではなくリンクローカルアドレス指定を使用する場合に起動できませんでした。今回のリリースにより、コンテナー起動スクリプトはグローバルアドレス指定に加えて、リンクローカルアドレス指定を受け入れるようになりました。(BZ#1838083)

- 以前のリリースでは、OVN コンテナーは必要以上の CPU および RAM を要求していました。これにより、ユーザーワークロード用に確保できる分が少なくなりました。今回のリリースで、OVN 要求は実際の要件を満たすように調整されるようになりました。(BZ#1844245)

- 以前のリリースでは、Terraform ステップの **openstack_networking_floatingip_associate_v2** がその依存するステップをすべて一覧表示せず、依存するステップが省略されたために競合状態が生じました。この競合状態により、負荷の高いシステムなどではとくに Terraform ジョブが失敗することがありました。今回のリリースにより、依存する Terraform ステップが **depends_on** として一覧表示され、Terraform ステップが正しい順序で実行されるようになりました。(BZ#1847957)

1.8.18.2. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.19. RHSA-2020:2737 - Important (重要): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-06-29

jenkins-2-plugins の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2737 アドバイザリーに記載されています。

1.8.20. RHBA-2020:2786 - OpenShift Container Platform 4.4.11 バグ修正の更新

発行日: 2020-07-06

OpenShift Container Platform リリース 4.4.11 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2020:2786 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、 RHBA-2020:2785 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。

OpenShift Container Platform 4.4.11 コンテナーイメージの一覧

1.8.20.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.21. RHSA-2020:2789 - Low (低) OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-07-06

ose-baremetal-operator-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2789 アドバイザリーに記載されています。

1.8.22. RHSA-2020:2790 - Low (低): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-07-06

ose-azure-machine-controllers-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2790 アドバイザリーに記載されています。

1.8.23. RHSA-2020:2792 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-07-06

grafana-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2792 アドバイザリーに記載されています。

1.8.24. RHSA-2020:2793 - Low (低): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-07-06

atomic-openshift-descheduler-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2793 アドバイザリーに記載されています。

1.8.25. RHBA-2020:2871 - OpenShift Container Platform 4.4.12 バグ修正の更新

発行日: 2020-07-13

OpenShift Container Platform リリース 4.4.12 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2020:2871 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、 RHBA-2020:2875 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。

OpenShift Container Platform 4.4.12 コンテナーイメージの一覧

1.8.25.1. バグ修正

- 以前のリリースでは、**ovn-octavia** ドライバーの同じポートの、複数のプロトコルで複数のリスナーを設定することはサポートされず、ブロックされていました。今回のリリースでは、これはサポートされるようになり、ブロックする必要がなくなりました。異なるプロトコルの複数のリスナーは同じポートで公開できます。つまり、**ovn-octavia** を使用する際に、たとえば DNS サービスが TCP プロトコルと UDP プロトコルの両方でポート 53 を公開することができます。(**BZ#1847558**)

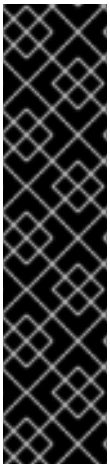
- 以前のバージョンでは、CoreDNS forward プラグインはデフォルトでランダムサーバー選択ポリシーを使用していました。そのため、複数の外部 DNS リゾルバーが指定されている場合に、クラスターは OpenStack API ホスト名の解決に失敗していました。プラグインは、指定された順序で DNS サーバーを使用できるようになりました。(**BZ#1851267**)

- 以前のバージョンでは、Fluentd が CLO を使用してスタンドアロンでデプロイされる場合、設定の詳細がないためにクラッシュしていました。今回のリリースにより、Pod が起動できるように空の Fluentd 設定が提供され、手動での介入が必要であることをユーザーに通知するためにステータスが追加されました。(**BZ#1851381**)

- インストールまたはアップグレード時に、**openshift-controller-manager** は進捗の状態を適切に報告しませんでした。その結果、インストールまたはアップグレードが失敗する可能性があります。Operator はインストールまたはアップグレードが正常に実行されると、その進捗を正しく報告するようになりました。(**BZ#1852249**)

1.8.25.2. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。



重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.26. RHSA-2020:2878 - Low (低): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-07-13

ose-cloud-credential-operator-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2878 アドバイザリーに記載されています。

1.8.27. RHBA-2020:2913 - OpenShift Container Platform 4.4.13 バグ修正の更新

発行日: 2020-07-21

OpenShift Container Platform リリース 4.4.13 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2020:2913 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、 RHBA-2020:2912 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。

OpenShift Container Platform 4.4.13 コンテナーイメージの一覧

1.8.27.1. 機能

1.8.27.1.1. メータリング Operator の更新

メータリング Operator をアップグレードできるようになりましたが、これまでは現在のメータリングのインストールをアンインストールしてからメータリング Operator の新規バージョンを再インストールする必要がありました。詳細は、メータリングのアップグレード を参照してください。

1.8.27.2. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.28. RHSA-2020:2926 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-07-21

openshift-enterprise-hyperkube-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2926 アドバイザリーに記載されています。

1.8.29. RHSA-2020:2927 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-07-21

openshift の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:2927 アドバイザリーに記載されています。

1.8.30. RHBA-2020:3075 - OpenShift Container Platform 4.4.14 バグ修正の更新

発行日: 2020-07-28

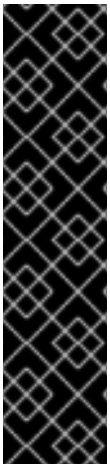
OpenShift Container Platform リリース 4.4.14 が公開されました。この更新に含まれるバグ修正の一覧は RHBA-2020:3075 および RHBA-2020:3288 アドバイザリーにまとめられています。この更新に含まれる RPM パッケージは、 RHBA-2020:3074 アドバイザリーで提供されています。

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。

OpenShift Container Platform 4.4.14 コンテナーイメージの一覧

1.8.30.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。



重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.31. RHSA-2020:3078 - Low (低): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-07-28

ose-cluster-machine-approver-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:3078 アドバイザリーに記載されています。

1.8.32. RHBA-2020:3128 - OpenShift Container Platform 4.4.15 バグ修正の更新

発行日: 2020-08-04

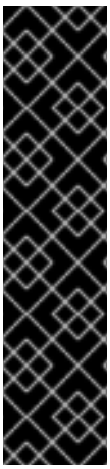
OpenShift Container Platform リリース 4.4.15 が公開されました。この更新に含まれるバグ修正の一覧は RHBA-2020:3128 アドバイザリーにまとめられています。この更新に含まれる RPM パッケージは、RHBA-2020:3127 アドバイザリーで提供されています。

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。

OpenShift Container Platform 4.4.15 コンテナーイメージの一覧

1.8.32.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。



重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.33. RHBA-2020:3237 - OpenShift Container Platform 4.4.16 バグ修正の更新

発行日: 2020-08-06

OpenShift Container Platform リリース 4.4.16 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2020:3237 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、 RHBA-2020:3238 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。

OpenShift Container Platform 4.4.16 コンテナーイメージの一覧

1.8.33.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.34. RHBA-2020:3334 - OpenShift Container Platform 4.4.17 バグ修正の更新

発行日: 2020-08-18

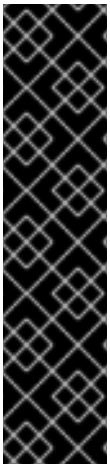
OpenShift Container Platform release 4.4.17 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2020:3334 アドバイザリーにまとめられています。この更新に含まれる RPM パッケージは、RHBA-2020:3335 アドバイザリーで提供されています。

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。

OpenShift Container Platform 4.4.17 コンテナーイメージの一覧

1.8.34.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。



重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.35. RHBA-2020:3440 - OpenShift Container Platform 4.4.18 バグ修正の更新

発行日: 2020-08-25

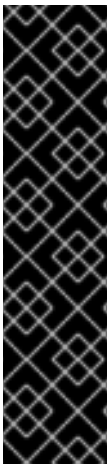
OpenShift Container Platform release 4.4.18 が公開されました。この更新に含まれるバグ修正の一覧は RHBA-2020:3440 アドバイザリーにまとめられています。この更新に含まれる RPM パッケージは、RHBA-2020:3441 アドバイザリーで提供されています。

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。

OpenShift Container Platform 4.4.18 コンテナーイメージの一覧

1.8.35.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。



重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.36. RHBA-2020:3514 - OpenShift Container Platform 4.4.19 バグ修正の更新

発行日: 2020-09-01

OpenShift Container Platform リリース 4.4.19 が公開されました。この更新に含まれるバグ修正の一覧は RHBA-2020:3514 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、RHBA-2020:3515 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。

OpenShift Container Platform 4.4.19 コンテナーイメージの一覧

1.8.36.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.37. RHSA-2020:3579 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-09-01

openshift の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:3579 アドバイザリーに記載されています。

1.8.38. RHSA-2020:3580 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-09-01

openshift-enterprise-hyperkube-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:3580 アドバイザリーに記載されています。

1.8.39. RHBA-2020:3564 - OpenShift Container Platform 4.4.20 バグ修正の更新

発行日: 2020-09-08

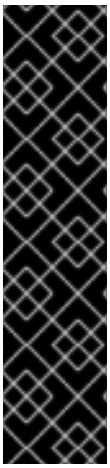
OpenShift Container Platform release 4.4.20 が公開されました。この更新に含まれるバグ修正の一覧は RHBA-2020:3564 アドバイザリーにまとめられています。この更新に含まれる RPM パッケージは、RHBA-2020:3565 アドバイザリーで提供されています。

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。

OpenShift Container Platform 4.4.20 コンテナーイメージの一覧

1.8.39.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。



重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.40. RHSA-2020:3625 - Important (重要): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-09-08

jenkins-2-plugins の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:3625 アドバイザリーに記載されています。

1.8.41. RHBA-2020:3605 - OpenShift Container Platform 4.4.21 バグ修正の更新

発行日: 2020-09-15

OpenShift Container Platform リリース 4.4.21 が公開されました。この更新に含まれるバグ修正の一覧は RHBA-2020:3605 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、 RHBA-2020:3606 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。

OpenShift Container Platform 4.4.21 コンテナーイメージの一覧

1.8.41.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.42. RHBA-2020:3715 - OpenShift Container Platform 4.4.23 バグ修正の更新

発行日: 2020-09-22

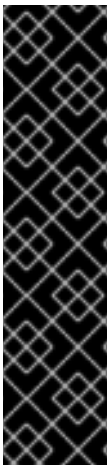
OpenShift Container Platform release 4.4.23 が公開されました。この更新に含まれるバグ修正の一覧は、 RHBA-2020:3715 アドバイザリーにまとめられています。この更新に含まれる RPM パッケージは、 RHBA-2020:3716 アドバイザリーで提供されています。

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。

OpenShift Container Platform 4.4.23 コンテナーイメージの一覧

1.8.42.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。



重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.43. RHSA-2020:3783 - Moderate (中程度): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-09-22

golang.org/x/text の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:3783 アドバイザリーに記載されています。

1.8.44. RHBA-2020:3764 - OpenShift Container Platform 4.4.26 バグ修正の更新

発行日: 2020-10-01

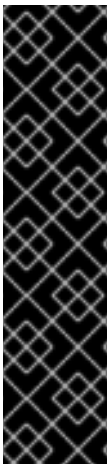
OpenShift Container Platform リリース 4.4.26 が公開されました。この更新に含まれるバグ修正の一覧は、 RHBA-2020:3764 アドバイザリーにまとめられています。この更新に含まれる RPM パッケージは、 RHBA-2020:3765 アドバイザリーで提供されています。

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。

OpenShift Container Platform 4.4.26 コンテナーイメージの一覧

1.8.44.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。



重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.45. RHBA-2020:4063 - OpenShift Container Platform 4.4.27 バグ修正の更新

発行日: 2020-10-13

OpenShift Container Platform release 4.4.27 が公開されました。この更新に含まれるバグ修正の一覧は、 RHBA-2020:4063 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、 RHBA-2020:4064 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。

OpenShift Container Platform 4.4.27 コンテナーイメージの一覧

1.8.45.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.46. RHSA-2020:4220 - Important (重要): OpenShift Container Platform 4.4 セキュリティー更新

発行日: 2020-10-13

openshift-jenkins-2-container の更新が OpenShift Container Platform 4.4 で利用可能になりました。更新の詳細については、RHSA-2020:4220 アドバイザリーに記載されています。

1.8.47. RHBA-2020:4224 - OpenShift Container Platform 4.4.29 バグ修正の更新

発行日: 2020-10-27

OpenShift Container Platform release 4.4.29 が公開されました。この更新に含まれるバグ修正の一覧は、 RHBA-2020:4224 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、RHBA-2020:4225 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。

OpenShift Container Platform 4.4.29 コンテナーイメージの一覧

1.8.47.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.48. RHBA-2020:4321 - OpenShift Container Platform 4.4.30 バグ修正の更新

発行日: 2020-11-11

OpenShift Container Platform リリース 4.4.30 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2020:4321 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、 RHBA-2020:4322 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。

OpenShift Container Platform 4.4.30 コンテナーイメージの一覧

1.8.48.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.49. RHBA-2020:5122 - OpenShift Container Platform 4.4.31 バグ修正の更新

発行日: 2020-12-02

OpenShift Container Platform release 4.4.31 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2020:5122 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、 RHBA-2020:5123 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。

OpenShift Container Platform 4.4.31 コンテナーイメージの一覧

1.8.49.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.50. RHBA-2021:0029 - OpenShift Container Platform 4.4.32 バグ修正の更新

発行日: 2021-01-13

OpenShift Container Platform リリース 4.4.32 が公開されました。この更新に含まれるバグ修正の一覧は、RHBA-2021:0029 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-2021:0030 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。

OpenShift Container Platform 4.4.32 コンテナーイメージの一覧

1.8.50.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。

重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

1.8.51. RHSA-2021:0281 - OpenShift Container Platform 4.4.33 バグ修正およびセキュリティー更新

発行日: 2021-02-02

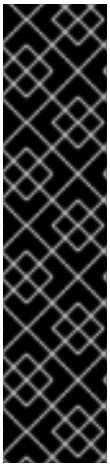
セキュリティー更新を含む OpenShift Container Platform リリース 4.4.33 が利用可能になりました。この更新に含まれるバグ修正の一覧は、RHSA-2021:0281 アドバイザリーにまとめられています。この更新に含まれる RPM パッケージは、RHSA-2021:0282 アドバイザリーで提供されています。

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。

OpenShift Container Platform 4.4.33 コンテナーイメージの一覧

1.8.51.1. アップグレード

既存の OpenShift Container Platform 4.4 クラスターをこの最新リリースにアップグレードする方法については、Web コンソールを使用したクラスターの更新 を参照してください。



重要

OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

これは、OpenShift Container Platform 4.3.5 の時点でサービス CA が自動的にローテーションされるためです。サービス CA はアップグレード時にローテーションされ、その後再起動して、以前のサービス CA の期限が切れる前にすべてのサービスが新しいサービス CA を使用することを確認する必要があります。

この 1 回で実行される手動による再起動の後、後続のアップグレードおよびローテーションにより、サービス CA の期限が切れる前に手動の介入なしの再起動が行われます。

第2章 OPENSHIFT CONTAINER PLATFORM のバージョン管理ポリシー

OpenShift Container Platform では、サポートされているすべての API の厳密な後方互換対応を保証しています。 ただし、アルファ API (通知なしに変更される可能性がある) およびベータ API (後方互換性の対応なしに変更されることがある) は例外となります。

Red Hat では OpenShift Container Platform 4.0 を公的にリリースせず、バージョン 3.11 の後に OpenShift Container Platform 4.1 を直接リリースしました。

OpenShift Container Platform のバージョンは、マスターとノードホストの間で一致している必要があります。ただし、クラスターのアップグレード時にバージョンが一時的に一致しなくなる場合を除きます。たとえば、4.4 クラスターではすべてのマスターは 4.4 で、すべてのノードが 4.4 である必要があります。以前のバージョンの **oc** をインストールしている場合、これを使用して OpenShift Container Platform 4.4 のすべてのコマンドを実行することはできません。新規バージョンの **oc** をダウンロードし、インストールする必要があります。

セキュリティーとは関連性のない理由で API が変更された場合には、古いバージョンの **oc** が更新されるように 2 つ以上のマイナーリリース (例: 4.1、4.2、4.3) 間での更新が行われます。新機能を使用するには新規バージョンの **oc** が必要になる可能性があります。4.3 サーバーにはバージョン 4.2 の **oc** で使用できない機能が追加されている場合や、バージョン 4.3 の **oc** には 4.2 サーバーでサポートされていない追加機能が含まれる場合があります。

表2.1 互換性に関する表

	X.Y (**oc** クライアント)	X.Y+N[a] (**oc** クライアント)
X.Y (サーバー)	1	3
X.Y+N[a] (サーバー)	2	1

[a] ここで、N は 1 よりも大きい数値です。

1 完全に互換性がある。

2 **oc** クライアントはサーバー機能にアクセスできない場合があります。

3 **oc** クライアントでは、アクセスされるサーバーと互換性のないオプションや機能を提供する可能性があります。