OpenShift Container Platform 4.3

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: 2020-09-21

OpenShift Container Platform 4.3 リリースノート

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

法律上の通知

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 4.3 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-2020:0062) 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 1.16 を使用します。以下では、OpenShift Container Platform 4.3 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。

注記

これは、OpenShift Container Platform 4.2 で使用した Kubernetes 1.14 より 2 バージョン分新しいバージョンです。

OpenShift Container Platform 4.3 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 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 4.3 は、Red Hat Enterprise Linux 7.6 以降、および Red Hat Enterprise Linux CoreOS 4.3 でサポートされます。

コントロールプレーンまたはマスターマシンには Red Hat Enterprise Linux CoreOS (RHCOS) を使用する必要があり、コンピュートまたはワーカーマシンには 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。

1.2. 新機能および改良された機能

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. OpenShift Container Platform アップグレードの段階的ロールアウト

OpenShift Container Platform 4.1 で、Red Hat はクラスターに対する適切なアップグレードバージョンを推奨するアップグレードチャネルの概念を導入しました。アップグレードチャネルは、アップグレー

ドストラテジーを切り分け、更新の頻度を制御するためにも使用されます。チャネルは OpenShift Container Platform のマイナーバージョンに関連付けられます。たとえば、OpenShift Container Platform 4.3 チャネルには 4.4 リリースへのアップグレードが含まれることはありません。これにより、管理者は OpenShift Container Platform の次のマイナーバージョンへのアップグレードに関して明確な決定を行うことができます。チャネルは更新のみを制御し、インストールするクラスターのバージョンには影響を与えません。 OpenShift Container Platform の特定のパッチレベルの **openshift-install** バイナリーは、そのパッチレベルのインストールを常に実行します。

アップグレードする予定の OpenShift Container Platform バージョンに対応するアップグレードチャネルバージョンを選択する必要があります。OpenShift Container Platform 4.3 には、直前の 4.2 リリースからのアップグレードが含まれます。

更新のタイプおよびアップグレードチャネルについての詳細は、「OpenShift 4.x Upgrades phased roll out」を参照してください。

アップグレードは Red Hat Service Reliability Engineering (SRE) チームからのデータに基づいて段階的に展開される際にチャネルに公開されるため、初期リリースでバージョン 4.2.z から 4.3 への更新が利用できるという通知が Web コンソールですぐに表示されない可能性があります。

1.2.1.2. FIPS 暗号のサポート

FIPS で検証された暗号ライブラリー/IUT (Implementation Under Test) 暗号ライブラリーを使用する OpenShift Container Platform クラスターをインストールすることができます。OpenShift Container Platform は、それが使用するオペレーティングシステムのコンポーネント用に Red Hat Enterprise Linux (RHEL) および Red Hat CoreOS (RHCOS) 内の特定の FIPS で検証されたモジュール/IUT (Implementation Under Test) モジュールを使用します。詳細は、「FIPS 暗号のサポート」を参照してください。

1.2.1.3. プライベートクラスターの AWS、Azure、または GCP へのデプロイ

プライベートクラスターを以下にインストールできます。

- Amazon Web Services (AWS) 上の既存の VPC。
- Google Cloud Platform (GCP) 上の既存の VPC。
- Microsoft Azure 上の既存の Azure Virtual Network (VNet)

プライベートクラスターをこれらのクラウドプラットフォームに作成するには、クラスターをホストするための既存のプライベート VPC/VNet およびサブネットを指定する必要があります。インストールプログラムは、プライベートネットワークからのみアクセスできるように Ingress Operator および API サーバーを設定します。

サポートされる各クラウドプラットフォームへのプライベートクラスターのデプロイについての詳細は、AWS、Azure、および GCP のインストールガイドを参照してください。

1.2.2. セキュリティー

1.2.2.1. サービス提供証明書の CA のローテーションの自動化

本リリースの今後の z-stream 更新で、自動のサービス CA ローテーションが利用可能になります。以前のバージョンでは、サービス CA は自動的に更新されないためにサービスの中断が生じ、手動の介入が必要でした。サービス CA および署名するキーについて、有効期限が切れる前に自動ローテーションが行われるようになりました。これにより、管理者は環境について事前に計画し、サービス中断を防ぐことができます。

1.2.2.2. etcd に保存されるデータの暗号化

etcd に保存されているデータを暗号化できるようになりました。クラスターの etcd 暗号化を有効にすると、データセキュリティーの層が追加されます。

etcd の暗号化を有効にすると、以下の OpenShift API サーバーおよび Kubernetes API サーバーリソースが暗号化されます。

- シークレット
- ConfigMap
- Routes
- OAuth アクセストークン
- OAuth 承認トークン

1.2.3. クラスターモニタリング

1.2.3.1. Web コンソールでの PromQL クエリーブラウザーの強化

OpenShift Container Platform Web コンソールで使用される PromQL クエリーブラウザーについてパフォーマンスが強化されました。

1.2.3.2. KubeletTooManyPods アラートの Pod 容量メトリクスの使用

KubeletTooManyPods アラートは、Pod の容量メトリクスを、固定数ではなくしきい値として使用するようになりました。

1.2.3.3. 独自のサービスの監視 (テクノロジープレビュー)

既存のモニタリングスタックは拡張できるため、独自のサービスに対してモニタリングを設定することができます。

1.2.3.4. Web コンソールでのメトリクスのクエリー (テクノロジープレビュー)

メトリクスのクエリーは、OpenShift Container Platform Web コンソール内の Developer パースペクティブで利用できます。

1.2.4. マシン API

1.2.4.1. マシンヘルスチェックで障害のあるマシンを自動的に修復

アウトバウンド管理 (out of band management) の対象外となり、削除されるマシンインスタンスは新規インスタンスの再作成を試行しなくなります。その代わりに、マシンの状態は **failed** フェーズになります。マシンヘルスチェックを設定し、デプロイして、マシンプールにある障害のあるマシンを自動的に修復できます。

MachineHealthCheck リソースを監視するコントローラーは、定義したステータスをチェックします。マシンがヘルスチェックに失敗した場合、これは自動的に検出され、新規マシンがこれに代わって作成されます。マシンが削除されると、machine deleted イベントが表示されます。マシンの削除による破壊的な影響を制限するために、コントローラーは 1 度に 1 つのノードのみをドレイン (解放) し、これを削除します。

チェックを停止するには、リソースを削除します。

1.2.5. ロギング

1.2.5.1. ログ転送 (テクノロジープレビュー)

ログ転送 API は、必ずしも OpenShift Container Platform クラスターロギングインフラストラクチャーによって管理されている訳ではない宛先に対してコンテナーおよびノードのログを送る方法を提供します。宛先エンドポイントは、OpenShift Container Platform クラスターでオンまたはオフにすることができます。ログ転送により、クラスターを Unmanaged に設定せずにログを転送でき、Fluentd プラグインを使用する場合よりもログ転送が容易になります。詳細は、「Forwarding logs using the Log Forwarding API」を参照してください。

1.2.6. 開発者のエクスペリエンス

1.2.6.1. OpenShift Do の拡張機能

OpenShift Do (odo) には、アプリケーションデプロイメントのユーザーエクスペリエンスに焦点を当てたいくつかの拡張機能が含まれています。

- **PushTimeout** が設定可能な待機パラメーターとして追加されています。
- サービスカタログおよびコンポーネント作成の両方が、拡張された出力および情報プロンプトと共に強化されています。
- アーキテクチャーのサポートが IBM Z および Power プラットフォームに拡張され、インストールで利用可能なバイナリーが提供されています。

1.2.6.2. Helm (テクノロジープレビュー)

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 メニューでダウンロードすることができます。

1.2.7. Web コンソール

1.2.7.1. 新規の Project ダッシュボード

新規の **Project** ダッシュボードは、Administrator および Developer パースペクティブから利用できるようになりました。このダッシュボードは、プロジェクトについての以下の情報を提供します。

- ステータス/正常性
- 外部リンク
- インベントリー
- 使用状況
- リソースクォータ

- アクティビティーおよび上位コンシューマー

1.2.7.2. ConsoleLink カスタムリソース定義 (Custom Resource Definition/CRD) の新たな NamespaceDashboard オプション

ConsoleLink カスタムリソース定義の新たな場所オプション **NamespaceDashboard** により、プロジェクト固有のリンクをプロジェクトダッシュボードに追加できます。

1.2.7.3. クラスター全体でのサードパーティーのユーザーインターフェースの提供

クラスター全体でのサードパーティーのユーザーインターフェースを統合し、Operator でバックアップするサービスを ConsoleLink カスタムリソース定義で開発し、管理し、設定できるようになりました。

1.2.7.4. 新規 ConsoleYAML サンプルカスタムリソース定義

新規 **ConsoleYAMLSample** カスタムリソース定義は、YAML サンプルを Kubernetes リソースにいつでも動的に追加できる機能を提供します。

詳細は、「Customizing the web console」を参照してください。

1.2.7.5. Web コンソールからのサポートケースの作成

Web コンソールのヘルプメニューから Red Hat サポートケースを作成することができるようになりました。

1.2.7.6. セキュリティー脆弱性の表示

Web コンソールのダッシュボードからコンテナーの脆弱性を確認できるようになりました。これには、オンプレミスおよび外部 Quay レジストリーの両方をサポートする Quay Operator を活用します。セキュリティー脆弱性は Quay が管理するイメージについてのみ報告されます。

1.2.7.7. 新規のユーザー管理セクション

すべてのユーザー管理リソースは、**User Resource** ナビゲーションセクションで利用可能になりました。

ユーザーの権限を借用する機能も追加されました。これにより、コンソールをナビゲートする際にユーザーに表示されるのと同じ内容を表示することができます。

1.2.7.8. アラートレシーバーの作成

アラートレシーバーを作成して、クラスターの状態についての通知を受信できるようになりました。PagerDuty および Webhook アラートタイプを作成できます。

1.2.7.9. Developer パースペクティブ

Developer パースペクティブを使用して以下を実行できるようになりました。

- サーバーレスアプリケーションおよびリビジョンを作成し、リビジョン間のトラフィックを分割する。
- アプリケーションとそのすべてのコンポーネントを削除する。

- プロジェクト内のユーザーに RBAC パーミッションを割り当てる。
- バインディングコネクターを使用してサービスでアプリケーションをバインドする。

1.2.7.10. CSI プロビジョナーがストレージクラス作成ページに表示される

Container Storage Interface (CSI) プロビジョナーがストレージクラスの作成ページに表示されるようになりました。ストレージクラスはユーザーインターフェースでハードコーディングされます。CSI ベースのストレージクラスは動的であり、静的な名前を持ちません。ユーザーはストレージクラス作成ページで、CSI ベースのプロビジョナーを一覧表示し、かつプロビジョナーを作成できるようになりました。

1.2.8. ネットワーク

1.2.8.1. ネットワークポリシーの設定

Kubernetes **v1** NetworkPolicy 機能は、Egress ポリシータイプおよび IPBlock 以外は OpenShift Container Platform で利用できます。

IPBlock は、制限付きの NetworkPolicy でサポートされています。これは、**except** 節なしで IPBlock をサポートします。**except** 節を含む **ipBlock** セクションのあるポリシーを作成する場合、SDN Pod は警告をログに記録し、そのポリシーの **ipBlock** セクション全体は無視されます。

1.2.8.2. Red Hat OpenStack Platform (RHOSP) でサポートされる Kuryr CNI サポート

Kuryr SDN を使用する RHOSP 13 および 16 にカスタマイズされたクラスターをインストールできます。Kuryr を使用して OpenStack にクラスターをインストールする方法については、インストールガイドを参照することができます。

1.2.9. スケーリング

1.2.9.1. クラスターの最大数

OpenShift Container Platform 4.3 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。

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 を使用します。

1.2.10. ストレージ

1.2.10.1. OpenShift Container Storage 4.2

Red Hat OpenShift Container Storage 4.2 クラスターのデプロイ、管理、監視、移行を実行できるようになりました。詳細は、『Red Hat OpenShift Container Storage 4.2 リリースノート』を参照してください。

1.2.10.2. iSCSI を使用した永続ストレージ

iSCSI を使用した永続ボリューム (以前はテクノロジープレビュー) は OpenShift Container Platform 4.3 で完全にサポートされるようになりました。

1.2.10.3. raw ブロックボリュームのサポート

テクノロジープレビューとして提供された iSCSI raw ブロックボリュームは OpenShift Container Platform 4.3 で完全にサポートされるようになりました。

Cinder を使用した raw ブロックボリュームはテクノロジープレビューとしてご利用いただけます。

1.2.10.4. CSI ボリューム拡張

Container Storage Interface (CSI) を使用して、作成後にストレージボリュームを拡張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。

1.2.10.5. ローカルストレージ Operator での容認 (toleration) の使用

ローカルストレージ Operator はノードテイントを許容するようになり、テイントされたノードからローカルボリュームをプロビジョニングできるようになりました。

1.2.11. Operator

1.2.11.1. Samples Operator

Samples Operator はインストール時にクラスターアーキテクチャーを自動的に認識し、互換性のない x86_64 コンテンツを Power および Z アーキテクチャーにインストールしません。

また、Samples Operator は Prometheus メトリクスを使用してインポートに失敗したイメージストリームや、Samples Operator に無効な設定があるかどうかについての情報を収集します。アラートは、イメージストリームがインポートに失敗した場合や、Samples Operator に無効な設定がある場合に送信されます。

1.2.11.2. イメージレジストリー Operator

以下の拡張機能がイメージレジストリー Operator で利用可能になりました。

- レジストリーの管理状態は、ベアメタル、vSphere、および Red Hat Virtualization プラットフォームで **Removed** として設定されるため、他のストレージプロバイダーを設定できます。新規インストールでは、ストレージのプロビジョニングに加えてレジストリーの状態を **Managed** に設定する必要があります。
- アラートは、レジストリーストレージが変更される際に送信されます。この変更により、データ損失が発生する可能性があるためです。

1.2.11.3. OperatorHub の単純化されたミラーリング

接続されていないクラスターと **oc adm** コマンドが実行されるワークステーションの両方で利用できるレジストリーが非接続環境で実行されている場合、以下の 3 つの手順を実行して OperatorHub をミラーリングできるようになりました。

1. **oc adm catalog build** を使用し、コンテナーイメージに対して Operator カタログをミラーリングし、接続されていないレジストリーにプッシュします。
2. **oc adm catalog mirror** を使用して参照される Operator およびアプリのイメージを解析し、接続されていないレジストリーにプッシュします。
3. **oc apply -f ./manifests** を使用して、ミラーカタログを接続されていないクラスターで有効にします。

詳細は、「ネットワークが制限された環境での Operator Lifecycle Manager の使用」を参照してください。

1.2.11.4. Operator Telemetry およびアラート

Lifecycle Operator Manager (OLM) がインストールされた Operator 情報を報告するようになりました。たとえば、OLM は失敗状態に移行する Operator についてのアラートを送信します。

1.3. 主な技術上の変更点

OpenShift Container Platform 4.3 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。

Operator SDK v0.12.0
OpenShift Container Platform 4.3 は Operator SDK v0.12.0 以降をサポートします。

Operator テストツーリング (scorecard v2) には、以下の拡張機能が含まれるようになりました。

- Operator テストの必須/オプションとしての分類。
- テストの選択および pass/fail 動作の設定。
- カスタムテストの送付。

Helm ベースの Operator の場合、改善点には以下が含まれます。

- Helm v3 のサポート (Operator SDK 0.14.0 以降)。
- ロールベースアクセス制御 (RBAC) の生成。

Ansible ベースの Operator の拡張機能には、以下が含まれます。

- Prometheus メトリクスのサポート。
- Red Hat Universal Base Image (UBI) の使用。
- Molecule ベースのエンドツーエンドのテスト。

最後に、Golang ベースの Operator の拡張機能には以下が含まれます。

- OpenAPI 仕様の生成。
- Kubernetes 1.14 のサポート。
- **dep** ベースプロジェクトの削除。すべての Go プロジェクトが、Go モジュールを使用できるようにスキャフォールディングされます。**operator-sdk new** コマンドの **--dep-manager** フラグが削除されました。
- 必要な Go バージョンの v1.10 から v1.13 への更新。
- Prometheus メトリクスのサポート。

クラスターロギング **Fluentforward** 設定の変更
新規のログ転送 API によって導入された変更により、OpenShift Container Platform 4.3 リリース以降の Fluentd **forward** プロトコルのサポートが変更されました。4.3 リリースでは、テクノロジープレビューの新規のログ転送機能を使用せずに、Fluentd **forward** プロトコル を依然として使用できます。

Fluentd **forward** プロトコルを使用するには、 **fluentd** ConfigMap の **secure-forward.conf** セクション

を編集する代わりに、ConfigMap オブジェクトを作成して **out_forward** を設定する必要があります。さらに、設定で必要になる証明書を、Fluentd Pod にマウントされるシークレットに追加できます。Fluentd Forward プラグインを使用した外部デバイスへのログ送信について参照してください。

4.3 では、Fluentd **forward** メソッドは非推奨となり、今後のリリースで削除されます。

OpenShift Container Platform 4.3 の更新時に、**fluentd** ConfigMap の **secure-forward.conf** セクションへの既存の変更が削除されます。現在の **secure-forward.conf** セクションをコピーしてから更新し、**secure-forward** ConfigMap オブジェクトの作成時にコピーしたデータを使用することができます。

1.3.1. サポートされない機能

クラスターロギングで、**Fluentd Daemonset** を編集してログを転送できなくなる
新規ログ転送 API で導入された変更により、Fluentd DaemonSet を編集してログを外部 Elasticsearch インスタンスに転送できなくなりました。

以前のバージョンでは、**fluentd** Daemonset を使用して **ES_HOST** および **OPS_HOST** 環境変数を使用したり、**fluent-plugin-remote-syslog** プラグインを使用したりすることができました。

新規のログ転送 API 機能や Fluentd **forward** プロトコルを使用して、ログを外部 Elasticsearch インスタンスおよび他のエンドポイントに転送できます。この変更を反映するために、ドキュメントが更新されています。

1.3.1.1. ローカルストレージプロビジョナー

以前に非推奨とされた **ose-local-storage-provisioner** コンテナーのテクノロジープレビューが削除されました。OLM ベースのローカルストレージ Operator（**ose-local-storage-operator**）を使用することが推奨されています。

1.3.1.2. 永続ボリュームスナップショット

永続ボリュームスナップショットは OpenShift Container Platform 4.2 で非推奨となり、OpenShift Container Platform 4.3 で削除されました。

1.3.2. 非推奨の機能

1.3.2.1. Pipeline ビルドストラテジー

Pipeline ビルドストラテジーが非推奨になりました。OpenShift Pipeline を代わりに使用します。

1.3.2.2. ベータ版のワークロードアラート

apps/v1beta1、**apps/v1beta2**、および **extensions/v1beta1** ワークロードアラートは、Kubernetes 1.16 の導入により非推奨になりました。

非推奨の API のいずれかを使用すると、 **UsingDeprecatedAPIExtensionsV1Beta1** アラートのプロンプトが出されます。これらの非推奨の API は OpenShift Container Platform の次のバージョンで削除されるので、サポートされている API に移行する必要があります。

1.3.2.3. 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

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 およびそれらの関連付

けられた Operator は OpenShift Container Platform 4.2 で非推奨となり、今後の OpenShift Container Platform リリースで削除されます。これらが 4.3 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ユーザーに警告するようになりました。

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ています。

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

以下の関連する API は今後のリリースで削除されます。

- **.servicecatalog.k8s.io/v1beta1**
- **.automationbroker.io/v1alpha1**
- **.osb.openshift.io/v1**

1.3.2.4. OperatorSource および CatalogSourceConfig が非推奨になる

OperatorSources および CatalogSourceConfig は OperatorHub から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**

1.3.2.5. CodeReady コンテナーの VirtualBox サポート

CodeReady Container (CRC) で VirtualBox の使用についてのサポートが非推奨になりました。

1.4. バグ修正

認証

- 認証 Operator は、利用不可の状態の理由を「available」という静的な文字列で報告していましたが、これは理由を明確に説明するものではありませんでした。今回のバグ修正により、利用不可の状態についてのより正確な理由が実装され、Operator が利用不可である理由の確認をより明確にできるようになりました。(BZ#1740357)
- **oauth-proxy** プロセスは各要求の CA 証明書を再度読み込み、それらをメモリーに保存していました。そのため、大量のメモリー消費による **oauth-proxy** コンテナーの強制終了が生じました。今回のバグ修正により、CA 証明書は変更されない場合にキャッシュされるようになりました。その結果として、複数の要求が実行される場合の **oauth-proxy** プロセスのメモリー消費量が大幅に減少しました。(BZ#1759169)
- 以前のバージョンでは、RequestHeader アイデンティティープロバイダー (IdP) 用に設定されたクライアント CA 証明書については、OAuth サーバーを使用する TLS ハンドシェイク時にその他の証明書間で通知されませんでした。**login-proxies** が OAuth サーバーへの接続を試行す

ると、それらのクライアント証明書は使用されず、要求が認証されないため、IdP のユーザーはクラスターにログインできませんでした。今回のバグ修正により、TLS 設定の残りの部分に設定済みのクライアント CA が追加されたため、RequestHeader IdP を使用した認証が予想通りに機能するようになりました。(BZ#1764558)

- OpenShift Container Platform 4.1 で導入されたブートストラップユーザーは CLI ログフローを内部で常に利用可能な状態にしました。OCP 3.x の認証トークンの取得方法を示すメッセージは、Web コンソールフローのみが設定されている場合には、CLI からのログインを試行するユーザーに表示されなくなりました。今回のバグ修正により、ブートストラップユーザーのアイデンティティープロバイダー (IdP) はユーザーが無効にした場合に設定されなくなりました。その結果、OCP ドキュメントの手順に従ってブートストラップ IdP を無効にした後に、Web コンソールを使用する場合のみのシナリオで表示された認証トークンの取得方法についてのメッセージが表示されるようになりました。(BZ#1781083)

- 以前のバージョンでは、oauth-server へのルートは Ingress ドメインの変更に応答しませんでした。これにより、認証 Operator のパフォーマンスが低下し、oauth-server が適切に認証されませんでした。oauth-server ルートは、Ingress ドメインの変更が検出されると更新されるようになり、この場合に認証が機能するようになりました。(BZ#1707905)

ビルド

- イメージストリームの作成後にすぐに起動されるビルドの場合、指定されるローカルのプルスルー (pull-through) イメージストリームタグを使用しない場合がありました。ビルドは外部イメージレジストリーからのイメージのプルを試行し、ビルドがそのレジストリーに必要な承認および証明書で設定されていない場合には、ビルドが失敗しました。ビルドコントローラーは、そのイメージストリームのキャッシュにローカルのプルスルー (pull-through) イメージストリームタグを許可するために必要な情報がないことを検出し、他の手段で情報を取得できるように更新されました。ビルドでローカルのプルスルー (pull-through) イメージストリームタグを正常に使用できるようになりました。(BZ#1753731)

- ビルドコントローラーでは、ビルドエンドポイントから直接インスタンス化される場合もビルド設定エンドポイントからインスタンス化されたと誤って仮定することがありました。そのため、ビルドコントローラーのログには、(ビルド設定 API エンドポイントの外部でビルド要求を実行する場合とは異なり) ユーザーが OpenShift ビルドを直接インスタンス化する場合に、存在しないビルド設定についての混乱を生じさせるログが表示される可能性がありました。ビルドコントローラーが更新され、ビルドがビルド設定エンドポイントからインスタンス化されたかどうかをより適切にチェックし、不要なエラーメッセージをログに記録しないようになりました。ビルドコントローラーのログには、(ビルド設定エンドポイントからではなく) 直接インスタンス化されたビルドについての混乱を生じさせるエラーメッセージが含まれなくなりました。(BZ#1767218)、(BZ#1767219)

クラスターバージョン Operator

- 以前のバージョンでは、自動更新を容易にするように設計された更新プロトコルの Cincinnati がペイロードの参照にタグを使用していました。これにより、複数の異なる時点での同じグラフの同じリリースを適用すると、複数の異なる結果が出される可能性がありました。コンテナーレジストリーが **manifestref** を指定する場合、ペイロード参照でイメージの SHA が使用されるようになりました。これにより、クラスターが使用する正確なリリースバージョンが適用されるようになります。(BZ#1686589)

コンソール kubevirt プラグイン

- 以前のバージョンでは、指定された **volumeMode** は新規に作成されたディスクに渡されず、PVC が適切にバインドされないことがありました。**volumeMode** が新たに作成されたディスクに適切に渡されるようになりました。(BZ#1753688)

- 以前のバージョンでは、仮想マシンの詳細ページは URL で直接アクセスされると適切に読み込まれませんでした。このページは適切に読み込まれるようになりました。(BZ#1731480)

- 以前のバージョンでは、**kubevirt-storage-class-defaults** ConfigMap 設定では VMware VM のインポートについて適切に反映されていませんでした。そのため、**blockMode** PVC が VMware VM インポートに使用できませんでした。ストレージクラスのデフォルトが、VMware のインポートされたディスクを要求する際に適切に使用されるようになりました。(BZ#1762217)

- 以前のバージョンでは、仮想マシンのインポートウィザードのタイトルが適切ではなく、混乱を生じさせる可能性がありました。ウィザードに正しいタイトルの **Import Virtual Machine** が使用されるようになりました。(BZ#1768442)

- 以前のバージョンでは、VM 移行ウィザードのストレージおよびネットワーク設定の確認ボタンが間違った場所にありました。確認ボタンが正しい場所に置かれるようになりました。(BZ#1778783)

- 以前のバージョンでは、**Create Virtual Machine** ウィザードが仮想マシンの作成前に確認を求めるプロンプトを出しませんでした。つまり、ユーザーが予期せずに仮想マシンを作成してしまう可能性がありました。今回の修正により、ユーザーが確認ページの「Create Virtual Machine」をクリックしないと仮想マシンが作成されないようになりました。(BZ#1674407)

- 以前のバージョンでは、**Create Virtual Machine** ウィザードに、仮想マシンのインポート時に常に直感的に機能する訳ではない必須フィールドが含まれました。**Create Virtual Machine** ウィザードは予想通りに機能するよう再設計されました。(BZ#1710939)

- 以前のバージョンでは、仮想マシン名の検証についてのエラーメッセージが適切な情報を伝えませんでした。このエラーメッセージはより多くの説明が含まれるように改善されました。(BZ#1743938)

コンテナー

- 以前のバージョンでは、CRI-O は復元操作時に Podman コンテナーを適切にフィルターしませんでした。Podman コンテナーには CRI-O 固有のメタデータがないため、起動時に CRI-O は、自らが認識した Podman コンテナーを誤って作成された CRI-O コンテナーとして解釈していました。そのため、ストレージライブラリーに対してコンテナーを削除するように指示しました。今回のバグ修正により、CRI-O の復元時に Podman コンテナーを適切にフィルターでき、それらが起動時にストレージから削除されないようになりました。(BZ#1758500)

Etcd

- etcd はオブジェクトが多数になるとオーバーロードし、etcd の失敗時にクラスターが停止することがありました。etcd クライアントバランサーは、クライアント接続のタイムアウト時にピアフェイルオーバーを容易にします。(BZ#1706103)

- etcd はアップグレードプロセス時に失敗し、障害復旧の修復手順が生じることがありました。etcd は、致命的なクラスター障害を防ぐために gRPC パッケージを解決するように更新されました。(BZ#1733594)

イメージレジストリー

- イメージレジストリー Operator の設定でストレージタイプを変更した後に、直前のストレージタイプと新規のストレージタイプの両方が Operator のステータスに表示されました。この動作により、イメージレジストリー Operator はその設定を削除した後も削除されませんでした。今回のリリースより、新規ストレージタイプのみが表示されるようになったため、イメージが使用するストレージタイプを変更した後にイメージレジストリー Operator が削除されるようになりました。BZ#1722878)

- 古いイメージストリームに無効な名前が含まれる可能性があり、イメージストリームのタグの仕様が無効な場合にイメージのプルーニングは失敗しました。関連付けられたイメージストリームに無効な名前が含まれる場合に、イメージプルーナーは常にイメージをプルーニングするようになりました。BZ#1749256)

- イメージレジストリー Operator の管理状態が **Removed** の場合、それ自体が Available であることを報告せず、正しいバージョン番号を報告しませんでした。この問題により、イメージレジストリー Operator が **Removed** に設定されている場合にアップグレードは失敗しました。イメージレジストリー Operator のステータスを **Removed** に設定すると、それ自体が Available として、また正確なバージョンで報告されるようになりました。イメージレジストリーをクラスターから削除した場合でも、アップグレードを完了できます。(BZ#1753778)

- イメージレジストリー Operator を無効な Azure コンテナー名で設定する可能性があり、イメージレジストリーは名前が無効であると Azure にデプロイできませんでした。イメージレジストリー Operator の API スキーマにより、入力する Azure コンテナー名が Azure の API 要件に準拠し、有効であることが確認されるようになったため、Operator がデプロイできるようになりました。(BZ#1750675)

kube-apiserver

- 不要なサービスモニタリングオブジェクトが kube-apiserver、kube-controller-manager、および kube-scheduler のそれぞれのコントローラーに作成されていました。使用されないサービスモニタリングオブジェクトは作成されなくなりました。(BZ#1735509)

- テクノロジープレビュー機能またはカスタム機能のいずれかが有効になっているためにクラスターがアップグレード不可能な状態にあると、アラートは送信されませんでした。クラスターは、アップグレードが不可能な状態のクラスターでアップグレードを試行すると **TechPreviewNoUpgrade** アラートを Prometheus 経由で送信するようになりました。(BZ#1731228)

kube-controller-manager

- StatefulSet リソースオブジェクトを定義する際に、カスタムラベルは **volumeClaimTemplates** パラメーターで指定されたテンプレートから PersistentVolumeClaim リソースオブジェクトを作成する場合に適用されませんでした。カスタムラベルは、StatefulSet リソースで定義される **volumeClaimTemplates** から作成される PersistentVolumeClaim オブジェクトに正常に適用されるようになりました。(BZ#1753467)

- 以前のバージョンでは、Kubernetes Controller Manager (KCM) のリース ConfigMap が削除されると、KCM には ConfigMap を再作成するパーミッションがなく、これを実行できませんでした。KCM は、削除されている場合でもリース ConfigMap を再作成できるようになりました。(BZ#1780843)

ロギング

- クラスターバージョンと ClusterLogging バージョンが一致しないと、ClusterLogging がデプロイに失敗しました。kubeversion は、デプロイされた ClusterLogging バージョンをサポートしているかどうかについて検証されるようになりました。(BZ#1765261)

- facility 値についての journald のデータはサニタイズされず、値が間違っているため、fluentd が正しくないレベルでエラーメッセージを出力していました。今回のリリースより、fluentd がデバッグレベルでログを記録し、これらのエラーが適切に報告されるようになりました。(BZ#1753936、BZ#1766187)

- oauth-proxy は誤って設定され、ユーザーがログアウト後にログインすることができませんでした。今回のリリースにより、oauth-proxy が再設定され、ユーザーがログアウト後も再度ログインできるようになりました。(BZ#1725517)

- Eventrouter は不明なイベントタイプを処理できませんでした。これにより Eventrouter のクラッシュが発生しました。eventrouter が不明なイベントタイプを適切に処理できるようになりました。(BZ#1753568)

管理コンソール

- Management Console Dashboard Details では、不要なインフラストラクチャーリソースの監視が行われていました。これにより、初期の Web ソケット接続の終了に関するエラーが生じる可能性がありました。今回のリリースより、Details カードでインフラストラクチャーリソースを監視しなくなり、リソースデータのフェッチが 1 度だけ行われるようになりました。この修正の実装後は、エラーが報告されなくなりました。(BZ#1765083)

- コンソール Operator は、ルーターがホスト名を指定する前にコンソール URL の初期の空の文字列の値を記録しました。今回のリリースより、Operator はホスト名が入力され、空の文字列の値が除去されるまで待機するようになりました。(BZ#1768684)

メータリング Operator

- 以前のバージョンでは、 **metering-operator** CSV バンドルの **containerImage** フィールドは、ART が置換に使用する **image-references** ファイルに一覧表示されていないイメージタグを参照しました。そのため、ART は **containerImage** フィールドに一覧表示されている元のイメージを関連付けられた **image-registry** リポジトリーおよび **sha256** タグに適切に置き換えることができませんでした。今回のバグ修正により、イメージタグ **latest** が、**image-references** ファイルで定義される **release-4.3** に置き換えられました。これにより、ART は **metering-operator** コンテナーイメージを正常に置き換えることができるようになりました。(BZ#1782237)

- 以前のバージョンでは、Hadoop **Dockerfile.rhel** は **gcs-connector** JAR ファイルをコンテナーの正しくない場所にコピーしました。パスが修正され、正しい場所をポイントするようになりました。(BZ#1767629)

ネットワーク

- 以前のバージョンでは、CNO が変更されると、関連するすべてのオブジェクトが削除されず、古い network-attachment-definitions がそのまま残りました。OpenShift Container Platform 4.3 では、コードのリファクタリングが行われ、関連するオブジェクトが適切にクリーンアップされるように一般的な方法で実行できるようになりました。(BZ#1755586)

- 以前のバージョンでは、一部の更新が完了せずにドロップされ、これによりイベントが失敗していました。 イベントがドロップされなくなりました。(BZ#1747532)

- 以前のバージョンでは、ネットワークのトラフィック量が高く、パケットロスのあるクラスターで、サービスへの正常な接続が **Connection reset by peer** エラーを出して失敗する可能性がありました。その結果、クライアントは再接続し、再送信する必要がありました。TCP 再送信を正しく処理するために iptables ルールが更新されました。確立された接続は、閉じられるまで開かれたままになります。(BZ#1762298)

- 以前のバージョンでは、多くの namespace、namespace の変更、および namespace を選択する NetworkPolicies を含むクラスターでは、新規 namespace に対する NetworkPolicy ルールの適用に時間がかかる場合がありました。新規 namespace は、他の namespace からアクセスできるようになるまでに多くの時間がかかる可能性がありました。Namespace および NetworkPolicy コードの更新により、NetworkPolicies は新規の namespace にすぐに適用されます。(BZ#1752636)

- 以前のバージョンでは、SDN Pod がノードで再起動する際に Egress IP アドレスをクリーンアップせず、IP アドレスの競合が生じました。SDN Pod は起動時に古い Egress IP アドレスをクリーンアップし、競合は発生しなくなりました (BZ#1753216)。

- 以前のバージョンでは、DNS 名は EgressNetworkPolicy に出現するたびにクエリーされました。特定の DNS レコードが以前のクエリーによって更新されたかどうかに関わらずレコードのクエリーが行われ、これによりネットワークのパフォーマンスが低下しました。DNS レコードのクエリーは、EgressNetworkPolicy ごとではなく一意の名前に基づいて行われるようになりました。その結果、DNS クエリーのパフォーマンスが大幅に改善しました。(BZ#1684079)

- 複数のサービスのエンドポイント間のルート作成をコンソールから実行できませんでした。今回のリリースにより、GUI が最大 3 つの代替サービスエンドポイントを追加または削除できるように更新されました。(BZ#1725006)

ノード

- 以前のバージョンでは、コンテナーの再起動数が高い場合（または >1 の場合)、kubelet は重複したコンテナーメトリクスをメトリクスストリームに挿入し、これにより kubelet の **/metrics** エンドポイントが 500 エラーをスローしました。今回のバグ修正により、（実行中または停止状態の）最新のコンテナーのメトリクスのみが含まれるようになりました。これにより、**/metrics** エンドポイントが 500 エラーを出さずにメトリクスを Prometheus に送ることができるようになりました。(BZ#1779285)

- アップストリームの変更が長いパス名のテストに対して行われました。255 文字を超える名前を持つ Pod はログに記録されず、警告は出力されませんでした。今回のリリースにより、長い名前のテストが削除され、名前が 255 文字を超える Pod は予想通りにログを記録するようになりました。(BZ#1711544)

- **LocalStorageCapacityIsolation** 機能は無効にされており、ユーザーは **Statefulset.emptyDir.sizeLimit** パラメーターを使用できませんでした。**LocalStorageCapacityIsolation** 機能は有効にされ、**Statefulset.emptyDir.sizeLimit** パラメーターを設定できるようになりました。(BZ#1758434)

oc

- 以前のバージョンでは、サーバー側の print を使用する場合、 wide 出力オプションは watch で使用される場合に無視されました (**oc get clusteroperators -o wide**)。操作が修正され、サーバー側の print を使用する際に選択可能なすべてのオプションを適切に認識できるようになりました。(BZ#1685189)

- アップストリームドキュメントへの **oc explain** コマンドリンクの情報が古くなっていました。これらのリンクが更新され、有効になりました。(BZ#1727781)

- 正しくないフラグエラーと共に詳細な使用方法のメニュー情報が出力されるため、エラーメッセージが表示されない可能性がありました。**oc command --help** コマンドを実行すると、正しくないフラグのエラーのみが表示されるようになりました。(BZ#1748777)

- **oc status** コマンドは、ステータスコード情報がないために、一貫した形式で DaemonSet を表示しませんでした。**oc status** コマンドの出力に Daemonset、Deployment、および Deployment Configuration が適切に出力されるようになりました。(BZ#1540560)

- コマンドの **oc version** および **oopenshift-install version** は、フラグが正しく設定されていない場合に Dirty として表示されました。これらのフラグは更新され、コマンドは Dirty **GitTreeState** または **GitVersion** を表示しなくなりました。(BZ#1715001)

- **oc status** コマンドは、コントローラーによって所有されていた可能性のある Pod を含む Pod が実行中であることを検証するために **oc set probe pod** の提案を表示しました。コントローラーによって所有される Pod はプローブの提案で無視されるようになりました。(BZ#1712697)

- 以前のバージョンでは、**oc new-build** の help コマンドはフラグを適切にフィルターしません

でした。これにより、**oc new-build --help** の呼び出し時に関係のないフラグが出力されました。これは修正されており、help コマンドは関連する出力のみを印刷するようになりました。(BZ#1737392)

openshift-apiserver

- 4.2 および 4.3 の **ClusterResourceQuota** では、OpenAPI スキーマに誤りがあるため、文字列以外の値は制限値として許可されませんでした。そのため、4.1 で実行された場合でも、整数のクォータの値は **ClusterResourceQuota** オブジェクトで設定できませんでした。**ClusterResourceQuota** の OpenAPI スキーマが整数を使用できるように修正されたため、整数を **ClusterResourceQuota** のクォータ値として再び使用できるようになりました。(BZ#1756417)

- アップグレード時に、**openshift-apiserver** は **degraded** を報告しました。パフォーマンスの低下の理由は **MultipleAvailable** としてのみ表示されましたが、この意味をユーザーが理解するのは不可能でした。今回のバグ修正により、パフォーマンス低下の理由が一覧表示され、関連情報がユーザーに表示されるようになりました。(BZ#1767156)

Web コンソール

- コンソールのワークロードには、knative serverless TP1 Operator がインストールされ、管理者以外のユーザーがログインしている場合にアクセス制限についてのエラーが表示されます。今回のバグ修正により、Overview サイドバーのリソースが通常のデプロイメントおよび knative 固有のデプロイメントの両方で予想通りに機能するようになりました。管理者以外のユーザーがワークロードを表示できるようになりました。(BZ#1758628)

- トポロジービューのデータモデルは、当初はプロジェクトの Workloads ページのサブセットでした。他の機能が追加され、トポロジービューは拡張されましたが、同じコードは共有されませんでした。ユースケースがより複雑になると、一部のエッジケースは新しいコードで対応されませんでした。特定の状況では、トポロジービューの Pod の一覧は正しくありませんでした。今回のバグ修正により、コードロジックがトポロジービューとプロジェクトの Workloads ページ間で共有されるようになりました。その結果、トポロジーからサイドバーの Pod 一覧を表示する場合も、プロジェクトの Workloads 一覧から表示する場合も、同一の Pod の詳細が表示されるようになりました。(BZ#1760827)

- 以前のバージョンでは、Route オブジェクトの作成時に、ターゲットポートのドロップダウンメニューで選択したポートを設定するのではなく、利用可能なポート一覧の最初のポートが設定されていました。このため、ユーザーは必要なターゲットポートを選択できませんでした。ターゲットポートのドロップダウンメニューから選択されたポートが Route オブジェクトの作成時に適用されるようになりました。ポートが選択されていない場合は、一覧の最初のポートが設定されます。(BZ#1760836)

- 以前のバージョンでは、アプリケーション名やビルドステータスなどの一部の特長については、Edge ブラウザーの **Topology** ビューでレンダリングされませんでした。今回のバグ修正により、Edge ブラウザーはアプリケーション名とビルドステータスを予想通りにレンダリングします。(BZ#1760858)

- Web コンソールの Overview では、Knative ワークロードではないデプロイメントが選択されている場合でも、管理者以外のユーザーは Knative Operator がインストールされている場合にワークロードを表示できませんでした。今回のバグ修正により、システムが Overview に Knative 固有のリソースを追加しないように、設定が見つからない場合のチェック機能が追加されました。これにより、管理者以外のユーザーが予想通りにワークロードを表示できるようになりました。(BZ#1760810)

- 以前のバージョンでは、Topology コンテキストメニューが開いている場合に、関連付けられたノードを簡単に特定することができませんでした。ユーザーはコンテキストメニューが参照しているノードを特定できないために、混乱を生じさせる状況が生じました。今回のリリースに

より、ノードを右クリックしてコンテキストメニューを開くと、視覚的なホバーやドロップシャドウがノードに適用され、ノードの特定が容易になりました。(BZ#1776401)

- 以前のバージョンでは、Web コンソールの **Import from Git** フォームは制限のある正規表現を使用するために Git URL を検証できず、一部の有効な URL が許可されませんでした。正規表現が更新され、すべての有効な Git URL を許可できるようになりました。(BZ#1766350)、(BZ#1771851)

- 開発者コンソールからのエラーメッセージに重複がありました。今回のリリースにより、このシステムが更新され、クライアント側の値が反映されるようになりました。その結果、エラーメッセージは明確かつ簡潔になりました。(BZ#1688613)

- 以前のバージョンでは、OLM オペランドリソースの Resources タブにアクセスする際に、Web コンソールにランタイムエラーが発生することがありました。OLM オペランドリソースの Resources タブのソートを試行する際に、Web コンソールがフリーズする可能性もありました。これらの問題は解決されています。(BZ#1756319)

- 以前のバージョンでは、Microsoft Edge の OpenShift Web コンソールの Pod の詳細ページにアクセスすると、ランタイムエラーが発生する可能性があり、ページを表示できませんでした。この問題は解決され、Pod の詳細ページが正しく表示されるようになりました。(BZ#1768654)

- 以前のバージョンでは、ダッシュボードカードが Prometheus の結果を監視する場合、古いアラートと新しいアラートの正しくない比較により、ダッシュボードページのパフォーマンスが低下することがありました。比較に関連する不具合が修正されました。(BZ#1781053)

- 以前のバージョンでは、Network Policy ページのドキュメントリンクに誤りがありました。これは正しいリンクに置き換えられました。(BZ#1692227)

- 以前のバージョンでは、Prometheus クエリーには範囲セレクターが含まれ、これにより Prometheus UI のデフォルトページのチャートがレンダリングされませんでした。クエリーに範囲セレクターが含まれなくなったため、クエリーのレンダリングが適切に行われるようになりました。(BZ#1746979)

- **Recycle** は Persistent Volume Reclaim ポリシーのデフォルト値です (このオプションが非推奨になっている場合も同様)。 永続ボリュームには、デフォルトで非推奨の値が含まれていました。デフォルトの Persistent Volume Reclaim ポリシーが **Retain** になり、新規の永続ボリュームに非推奨の値が含まれなくなりました。(BZ#1751647)

- 以前のバージョンでは、クラスターのアップグレード後に Web コンソールはキャッシュされた CSS スタイルシートを使用できました。これにより、コンソールの読み込み時にレンダリングの問題が発生する可能性がありました。この問題は修正され、アップグレード後に Web コンソールが正しいスタイルシートを適切に使用できるようになりました。(BZ#1772687)

- 以前のバージョンでは、Web コンソールを使用する場合に、オプションメニューの一部がページ上の他の要素の後ろに置かれ、非表示になる場合がありました。オプションメニューは他のページ要素の後ろに表示されなくなり、メニュー全体が常に表示されるようにページの表示可能なスペースに展開できるようになりました。(BZ#1723254)

- 以前のバージョンでは、長いノード名により OpenShift コンソールの Pod テーブルの表の列にオーバーフローが生じる可能性がありました。今回のバグ修正により、それらが適切にラップされるようになりました。(BZ#1713193)

- 以前のバージョンでは、サンプル YAML を使用してレポートクエリーを作成するとエラーが生じていました。今回のバグ修正により、すべての必須フィールドを含むレポートクエリーの新たな YAML サンプルが追加され、エラーが発生しなくなりました。(BZ#1753124)

- 以前のバージョンでは、Install Plan Details ページで関連付けられたカタログソースの namespace が正しく設定されませんでした。これにより、namespace がないことからリンクの破損が生じました。今回のバグ修正により、InstallPlan リソースの **status.plan** フィールドを使用して、カタログソースを正しい namespace に関連付けてリンクを作成できるようになりました。その結果、カタログソースのリンクが予想通りに機能するようになりました。(BZ#1767072)

- 以前のバージョンでは、不明なカスタムリソースは、ユーザーへの表示用として自動的に単語単位に分割されました。ただし、一部のリソースは適切に分割されませんでした。今回のバグ修正により、カスタムリソースは、単語ベースで分割されるのではなく、カスタムリソース定義で定義された名前を使用するようになりました (BZ#1722811)。

1.5. 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、**TP** というマークの付いた機能は テクノロジープレビュー を指し、**GA** というマークの付いた機能は 一般公開機能 を指します。- としてマークされている機能は、機能がリリースから削除されているか、非推奨にされていることを示しています。

表1.1 テクノロジープレビュートラッカー

機能	OCP 4.1	OCP 4.2	OCP 4.3
Prometheus クラスターモニタリング	GA	GA	GA
Precision Time Protocol (PTP)	-	-	TP
ランタイム Pod の CRI-O	GA	GA	GA
oc CLI プラグイン	TP	TP	TP
サービスカタログ	GA	-	-
テンプレートサービスブローカー	GA	-	-
OpenShift Ansible Service Broker	GA	-	-
ネットワークポリシー	GA	GA	GA
Multus	GA	GA	GA

機能	OCP 4.1	OCP 4.2	OCP 4.3
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
StatefulSets	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
Docker のシステムコンテナー、CRI-O	-	-	-
Hawkular エージェント	-	-	-
Pod の PreSet	-	-	-
experimental-qos-reserved	TP	TP	TP
Pod sysctl	GA	GA	GA
中央監査	-	-	-
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
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	-	-	-
Kubernetes リソースのあるイメージストリーム	GA	GA	GA

機能	OCP 4.1	OCP 4.2	OCP 4.3
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
一時ストレージの制限/要求	TP	TP	TP
CephFS Provisioner	-	-	-
Podman	TP	TP	TP
Kuryr CNI プラグイン	-	TP	GA
PID Namespace のコントロール共有	TP	TP	TP
Manila Provisioner	-	-	-
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)	TP	GA	GA
Operator Lifecycle Manager	GA	GA	GA
Red Hat OpenShift Service Mesh	GA	GA	GA

機能	OCP 4.1	OCP 4.2	OCP 4.3
「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
Dockerfile 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	TP	GA
iSCSI を使用した永続ストレージ	TP	TP	GA
iSCSI での raw ブロック	-	TP	GA
Cinder での raw ブロック			TP
OperatorHub		GA	GA
3 ノードのベアメタルデプロイメント		TP	TP
SR-IOV ネットワーク Operator		TP	GA
Helm CLI			TP
サービスバインディング			TP
ログ転送			TP
ユーザーワークロードの監視			TP
OpenShift Serverless	TP	TP	TP

機能	OCP 4.1	OCP 4.2	OCP 4.3
コンピュートノードトポロジーマネージャー			TP

1.6. 既知の問題

- Service Mesh をインストールしている場合、OpenShift Container Platform をアップグレードする前に Service Mesh をアップグレードします。回避策については、「Updating OpenShift Service Mesh from version 1.0.1 to 1.0.2」を参照してください。
- ロールアウト失敗時のアクティブな Pod の判別が **Topology** ビューで正しく行われない可能性があります。(BZ#1760828)
- 制限されたクラスタースコープのパーミッションを持つユーザーが **Add** ページで **Container Image** オプションを使用してアプリケーションを作成し、**Image name from internal registry** オプションを選択しても、イメージストリームが存在する場合でもイメージストリームがプロジェクトで検出されません。(BZ#1784264)
- **ImageContentSourcePolicy** はリリース時にレジストリーではサポートされません (BZ#1787112)。非接続環境では、Jenkins を有効にして、デフォルトでプルスルー (pull through) を実行できます。このコマンドを、非接続環境で Jenkins を使用するための回避策として使用できます。

```
$ oc tag <jenkins_source_image> jenkins:2 --reference-policy=source -n openshift
```

- OpenShift Cluster Version Operator (CVO) は、ホストから SSL 証明書を正しくマウントしません。そのため、MITM プロキシーチェックを使用する際にクラスターのバージョン更新が行われません。(BZ#1773419)
- **defaultProxy** および **gitProxy** を **builds.config.openshift.io** に追加すると、Jenkins Pipeline ビルドはプロキシー設定を取得できません。(BZ#1753562)
- OpenStack エンドポイントが自己署名 TLS 証明書で設定された Red Hat OpenStack Platform 13 または 16 にインストールする場合は、インストールは失敗します。(BZ#1786314、BZ#1769879、BZ#1735192)
- 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の OpenStack へのインストールは、OpenStack Neutron に高い負荷がかかると、**Security group rule already exists** エラーを出して失敗します。(BZ#1788062)
- クラスターは、**etcd** のバックアップまたは復元機能が**etcd** 暗号化の移行プロセスで実行された後にエラーおよび異常な状態を表示します。(BZ#1776811)
- RHCOS マスターおよびワーカーノードは、4.2.12 から 4.3.0 にアップグレード中に **NotReady,SchedulingDisabled** 状態になる場合があります。(BZ#1786993)
- FIPS モードを有効にすると、RHEL 用のパブリッククラウドアクセスイメージを直接使用できません。これは、パブリッククラウドイメージがカーネルの整合性チェックを許可しないために生じます。これを可能にするには、独自のイメージをアップロードする必要があります。(BZ#1788051)
- Operator Lifecycle Manager (OLM) は、Kuryr SDN が有効にされている場合は OpenShift Container Platform では機能しません。(BZ#1786217)

- **oc adm catalog build** および **oc adm catalog mirror** コマンドは、制限のあるクラスターでは機能しません。(BZ#1773821)

- OpenShift Container Platform クラスターを 4.1 から 4.2、次いで 4.3 にアップグレードする場合、Node Tuning Operator のチューニングされた Pod が **ContainerCreating** 状態のままになる可能性があります。
問題を確認するには、以下を実行します。

```
$ oc get pods -n openshift-cluster-node-tuning-operator
```

1 つ以上のチューニングされた Pod が **ContainerCreating** 状態のままになっています。

この問題を解決するには、以下の回避策を適用します。以下を実行します。

```
$ oc delete daemonset/tuned -n openshift-cluster-node-tuning-operator
$ oc get daemonset/tuned -n openshift-cluster-node-tuning-operator
$ oc get pods -n openshift-cluster-node-tuning-operator
```

Pod が **Running** 状態にあることを確認します。(BZ#1791916)

- Node Feature Discovery (NFD) Operator バージョン 4.3 は、OpenShift Container Platform Web コンソールの OpertorHub からのデプロイに失敗します。回避策として、オペレーティングシステムの **oc** クライアントをダウンロードし、インストーラーから **kubeconfig** ファイルを ~/**.kube**/**config**に配置します。これらのコマンドを実行して、CLI および GitHub から NFD Operator をデプロイします。

```
$ cd $GOPATH/src/openshift
$ git clone https://github.com/openshift/cluster-nfd-operator.git
$ cd cluster-nfd-operator
$ git checkout release-4.3
$ PULLPOLICY=Always make deploy
$ oc get pods -n openshift-nfd
```

出力例:

```
$ oc get pods -n openshift-nfd
NAME READY STATUS RESTARTS AGE
nfd-master-gj4bh 1/1 Running 0 9m46s
nfd-master-hngrm 1/1 Running 0 9m46s
nfd-master-shwg5 1/1 Running 0 9m46s
nfd-operator-b74cbdc66-jsgqq 1/1 Running 0 10m
nfd-worker-87wpn 1/1 Running 2 9m47s
nfd-worker-d7kj8 1/1 Running 1 9m47s
nfd-worker-n4g7g 1/1 Running 1 9m47s
```

(BZ#1793535)

- クラスター全体の egress プロキシーが設定され、後に設定が解除される場合、OLM 管理の Operator によって以前にデプロイされたアプリケーションの Pod は **CrashLoopBackOff** 状態になります。これは、デプロイされた Operator がプロキシーに依存するように設定されているために生じます。

注記

この問題は、クラスター全体の egress プロキシーで作成される環境変数、Volume、および VolumeMount に適用されます。これは、SubscriptionsConfig オブジェクトを使用して環境変数、Volume、および VolumeMount を設定する際にも同じ問題が発生します。

OpenShift Container Platform の今後のリリースで修正が予定されていますが、CLI または Web コンソールを使用してデプロイメントを削除することで問題を回避することができます。これにより、OLM が Deployment を再生成し、正しいネットワーク設定で Pod を開始します。

クラスター管理者は、以下のコマンドを実行して、影響を受ける OLM が管理する全 Deployment の一覧を取得できます。

```
$ oc get deployments --all-namespaces \
    -l olm.owner,olm.owner!=packageserver 1
```

1 影響を受けない **packageserver** を除きます。

(BZ#1751903)

- 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 ConfigMap 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)

- 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)

- 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)

1.7. エラータの非同期更新

OpenShift Container Platform 4.3 のセキュリティー、バグ修正、拡張機能の更新は、Red Hat Network

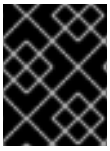
経由で非同期エラータとして発表されます。OpenShift Container Platform 4.3 のすべてのエラータは Red Hat カスタマーポータルから入手できます。非同期エラータについては、OpenShift Container Platform ライフサイクルを参照してください。

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 4.3 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 4.3.z などのバージョン付けされた非同期リリースについてはサブセクションで説明します。さらに、エラータの本文がアドバイザリーで指定されたスペースに収まらないリリースについては、詳細についてその後のサブセクションで説明します。



重要

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

1.7.1. RHBA-2020:0062 - OpenShift Container Platform 4.3 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: 2020-01-23

OpenShift Container Platform release 4.3 が公開されました。この更新に含まれるコンテナーイメージおよびバグ修正の一覧は RHBA-2020:0062 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-2019:0063 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 4.3.0 コンテナーイメージの一覧

1.7.2. RHBA-2020:0390 - OpenShift Container Platform 4.3.1 バグ修正の更新

発行日: 2020-02-12

OpenShift Container Platform リリース 4.3.1 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:0390 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コンテナーイメージおよびバグ修正は、RHBA-2020:0391 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 4.3.1 コンテナーイメージの一覧

1.7.2.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

1.7.3. RHBA-2020:0491 - OpenShift Container Platform 4.3.2 バグ修正の更新

発行日: 2020-02-19

OpenShift Container Platform リリース 4.3.2 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:0491 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コンテナーイメージおよびバグ修正は、 RHBA-2020:0492 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 4.3.2 コンテナーイメージの一覧

1.7.3.1. バグ修正

- Amazon Web Services (AWS) 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および Red Hat OpenStack Platform (RHOSP) の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には、TCP および UDP ポート 30000-32767 でコントロールプレーンとワーカー間の双方向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ありません。このため、新規に導入される OVN Networking コンポーネントはこれらのセキュリティーグループルールがないクラスターでは適切に機能しませんでした。上記の双方向のトラフィックサポートを許可するために、セキュリティーグループルールが利用できるようになりました。(**BZ#1779469**)

- 以前のバージョンでは、Web コンソールの **Installed Operators** ページにアクセスしようとすると **Restricted Access** エラーが出されました。これは、サブスクリプションの詳細を表示するためにコンソールが現在の namespace 外のサブスクリプションリソースにアクセスしようとするために生じました。ユーザーは、**Installed Operators** ページにアクセスできるようになりました。**Subscription** タブは、サブスクリプションリソースにアクセスできないユーザーには非表示になります。(**BZ#1791101**)

1.7.3.2. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

1.7.4. RHBA-2020:0528 - OpenShift Container Platform 4.3.3 バグ修正の更新

発行日: 2020-02-24

OpenShift Container Platform リリース 4.3.3 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:0527 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コンテナーイメージおよびバグ修正は、 RHBA-2020:0528 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 4.3.3 コンテナーイメージの一覧

1.7.4.1. バグ修正

- KnativeServing リソースの **serving.knative.dev** の API グループが非推奨となり、これは Serverless Operator 1.4 の **operator.knative.dev** に変更されました。(BZ#1779469)

1.7.4.2. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

1.7.5. RHSA-2020:0562 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-02-24

jenkins-slave-base-rhel7-container の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:0562 アドバイザリーに記載されています。

1.7.6. RHBA-2020:0675 - OpenShift Container Platform 4.3.5 バグ修正の更新

発行日: 2020-03-10

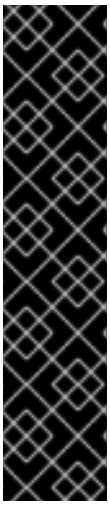
OpenShift Container Platform リリース 4.3.5 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:0675 アドバイザリーにまとめられています。この更新に含まれるコンテナーイメージおよびバグ修正は、RHBA-2020:0676 アドバイザリーで提供されます。

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 4.3.5 コンテナーイメージの一覧

1.7.6.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。



重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.7. RHSA-2020:0679 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-03-10

skopeo の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:0679 アドバイザリーに記載されています。

1.7.8. RHSA-2020:0680 - Low (低): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-03-10

podman の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:0680 アドバイザリーに記載されています。

1.7.9. RHSA-2020:0681 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-03-10

openshift-enterprise-apb-base-container、 **openshift-enterprise-mariadb-apb**、 **openshift-enterprise-mysql-apb**、および **openshift-enterprise-postgresql-apb** の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:0681 アドバイザリーに記載されています。

1.7.10. RHSA-2020:0683 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-03-10

openshift-enterprise-ansible-operator-container の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:0683 アドバイザリーに記載されています。

1.7.11. RHBA-2020:0857 - OpenShift Container Platform 4.3.8 バグ修正の更新

発行日: 2020-03-24

OpenShift Container Platform release 4.3.8 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:0857 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コンテナーイメージおよびバグ修正は、 RHBA-2020:0858 アドバイザリーで提供されます。

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 4.3.8 コンテナーイメージの一覧

1.7.11.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.12. RHSA-2020:0863 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-03-10

openshift-enterprise-builder-container および **openshift-enterprise-cli-container** の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:0863 アドバイザリーに記載されています。

1.7.13. RHSA-2020:0866 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-03-10

openshift-enterprise-template-service-broker-operator-container の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:0866 アドバイザリーに記載されています。

1.7.14. RHSA-2020:0928 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-03-10

openshift-clients の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:0928 アドバイザリーに記載されています。

1.7.15. RHBA-2020:0929: OpenShift Container Platform 4.3.9 バグ修正の更新

発行日: 2020-04-01

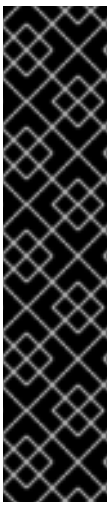
OpenShift Container Platform release 4.3.9 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:0929 アドバイザリーにまとめられています。この更新に含まれるコンテナーイメージおよびバグ修正は、 RHBA-2020:0930 アドバイザリーで提供されます。

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 4.3.9 コンテナーイメージの一覧

1.7.15.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。



重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.16. RHSA-2020:0933 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-04-01

ose-openshift-apiserver-container の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:0933 アドバイザリーに記載されています。

1.7.17. RHSA-2020:0934 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-04-01

ose-openshift-controller-manager-container の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:0934 アドバイザリーに記載されています。

1.7.18. RHBA-2020:1262 - OpenShift Container Platform 4.3.10 バグ修正の更新

発行日: 2020-04-08

OpenShift Container Platform リリース 4.3.10 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:1255 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コンテナーイメージおよびバグ修正は、RHBA-2020:1262 アドバイザリーで提供されます。

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 4.3.10 コンテナーイメージの一覧

1.7.18.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.19. RHSA-2020:1276 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-04-08

openshift の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:1276 アドバイザリーに記載されています。

1.7.20. RHSA-2020:1277 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-04-08

openshift-enterprise-hyperkube-container の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:1277 アドバイザリーに記載されています。

1.7.21. RHBA-2020:1393 - OpenShift Container Platform 4.3.12 バグ修正の更新

発行日: 2020-04-14

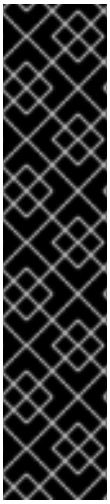
OpenShift Container Platform リリース 4.3.12 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:1392 アドバイザリーにまとめられています。この更新に含まれるコンテナーイメージおよびバグ修正は、RHBA-2020:1393 アドバイザリーで提供されます。

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 4.3.12 コンテナーイメージの一覧

1.7.21.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。



重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.22. RHSA-2020:1396 - Low (低): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-04-14

podman の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:1396 アドバイザリーに記載されています。

1.7.23. RHBA-2020:1482 - OpenShift Container Platform 4.3.13 バグ修正の更新

発行日: 2020-04-20

OpenShift Container Platform リリース 4.3.13 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:1481 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コンテナーイメージおよびバグ修正は、RHBA-2020:1482 アドバイザリーで提供されます。

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 4.3.13 コンテナーイメージの一覧

1.7.23.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.24. RHSA-2020:1485 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-04-20

runc の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:1485 アドバイザリーに記載されています。

1.7.25. RHBA-2020:1529 - OpenShift Container Platform 4.3.18 バグ修正の更新

発行日: 2020-04-29

OpenShift Container Platform リリース 4.3.18 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:1528 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コンテナーイメージおよびバグ修正は、RHBA-2020:1529 アドバイザリーで提供されます。

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 4.3.18 コンテナーイメージの一覧

1.7.25.1. 機能

1.7.25.1.1. IBM Power Systems

本リリースでは、IBM Power Systems は OpenShift Container Platform 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 Container Platform 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 Container (CRC)
  - Tekton をベースとする OpenShift Container Platform Pipeline
  - OpenShift Container Platform Metering
  - Multus CNI プラグイン
- ワーカーノードは Red Hat Enterprise Linux CoreOS 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。

1.7.25.1.2. IBM Z および LinuxONE

本リリースでは、IBM Z および LinuxONE は OpenShift Container Platform 4.3 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)
  - OpenShift Container Platform Serverless
  - ログ転送
  - Helm コマンドラインインターフェース (CLI) ツール
  - Precision Time Protocol (PTP) hardware
- 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 Container (CRC)
  - Tekton をベースとする OpenShift Container Platform Pipeline
  - 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
- ワーカーノードは Red Hat Enterprise Linux CoreOS 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。
- 他のサードパーティーストレージベンダーは、OpenShift Container Platform と連携するソリューションとして認定されているContainer Storage Interface (CSI) 対応ソリューションを提供している場合があります。詳細については、OpenShift Container Platform またはストレー

ジベンダーの OperatorHub について確認してください。

- OpenShift Container Platform インストールの z/VM インスタンスを設定する場合、一時的にワーカーノードの仮想 CPU の容量を増やすか、またはサードパーティーのワーカーノードを追加する必要がある場合があります。(BZ#1822770)

- これらの機能は 4.3 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 4.3 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)。

1.7.25.2. バグ修正

- 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。Web コンソールは無効な記述子を許容し、オペランドの詳細を表示するようになりました。(BZ#1798130)

- 以前のバージョンでは、Web コンソールは、プロジェクトを作成するパーミッションのないユーザーの **Create Project** アクションを表示していました。これにより、適切なパーミッションを持たないユーザーがプロジェクトの作成を試行するために混乱が生じ、プロジェクトを作成できなかったことを示すエラーメッセージが出されました。**Create Project** アクションは、プロジェクトを作成するパーミッションを持たないユーザーには表示されなくなりました。(BZ#1804708)

- **Upgradeable** フィールドは Service Catalog Operator によって適切に設定されませんでした。これにより、OpenShift Container Platform クラスターの新規インストール後に **Unknown** という正しくないアップグレードのステータスが表示されました。**Upgradeable** フィールドは適切に設定されるようになり、クラスターのアップグレードステータスが正確に表示されるようになりました。(BZ#1813488)

- イメージレジストリー Operator は、Operator が **Unmanaged** に設定されている場合に OpenShift Container Platform の新規バージョンを報告しませんでした。このため、クラスターの新しいバージョンへのアップグレードに失敗していました。イメージレジストリー Operator が **Unmanaged** に設定されると、新規のクラスターバージョンが報告され、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#1816656)

- 以前のバージョンでは、ts-loader が正しくない **tsconfig.json** を使用するために Web コンソールの特定のページでランタイムエラーが生じることがありました。ts-loader の問題が解決され、すべての Web コンソールページが適切に読み込まれるようになりました。(BZ#1818980)

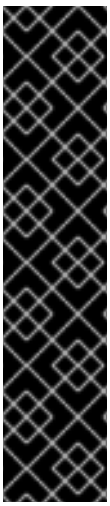
- 以前のバージョンでは、OpenShift Container Platform 内部レジストリーのプルシークレットの作成に使用されるクライアントには低いレート制限が設定されていました。多数の namespace が短時間に作成された場合には、イメージレジストリーのプルシークレットが作成されるまでに時間がかかりました。クライアントのレート制限が引き上げられたため、内部レジストリーのプルシークレットはトラフィック量が多い場合でも迅速に作成されるようになりました。(BZ#1819850)

- 以前のバージョンでは、node-ca デーモンは **NoExecute** テイントを許容しませんでした。これにより、node-ca デーモンは、**NoExecute** テイントが適用されたノード上の証明書を無視していました。今回の修正により、**additionalTrustedCA** がすべてのノードに同期し、すべてのテイントが許容されるようになりました。(BZ#1820242)

- CA 証明書を更新する **oc** コマンドには、操作するリソースタイプがありませんでした。これにより、コマンドがエラーを返していました。不足している ConfigMap が追加され、これによりコマンドエラーが修正されます。(BZ#1824921)

1.7.25.3. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。



重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.25.4. 既知の問題

- **coreos-installer** に関連する問題により、CoreOS は 4K セクターの NVMe ドライブを使用するベアメタルノードにはインストールできません。(BZ#1805249)
- IBM Power システムの **fw_enabled_large_send** 設定に問題があるため、VXLAN パケットの破棄が生じ、デプロイメントが失敗します。(BZ#1816254)
- IBM Power インフラストラクチャー上のクラスターの場合、ホットプラグデバイスに関連するパッケージがないため、ゲスト仮想マシンは動的にプロビジョニングされた永続ボリュームを検出しない可能性があります。そのため、**librtas**、**powerpc-utils**、および **ppc64-diag** などのパッケージおよびサービスをインストールする必要があります。(BZ#1811537)

1.7.26. RHBA-2020:2006 - OpenShift Container Platform 4.3.19 バグ修正の更新

発行日: 2020-05-11

OpenShift Container Platform release 4.3.19 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:2005 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コンテナーイメージおよびバグ修正は、RHBA-2020:2006 アドバイザリーで提供されます。

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 4.3.19 コンテナーイメージの一覧

1.7.26.1. バグ修正

- 以前のバージョンでは、イメージストリームが内部イメージレジストリーに認証情報があるが、コンシューマーに認証情報がないプライベートレジストリーでサポートされている場合に、後続のイメージのプルは失敗しました。クラスターの起動または OpenShift コントローラーマネージャーの再起動の直後にイメージストリームへのアクセスが発生した場合にイメージストリームのローカル参照設定が無視されるため、コントローラーマネージャーはメタデータが不完全な状態でイメージストリームをキャッシュに保存しました。OpenShift コントローラーマネージャーは、メタデータの初期化が不完全であった場合にもイメージストリーム

キャッシュを更新するように更新されました。これにより、クラスターの起動または OpenShift コントローラーマネージャーの再起動の直後のタイミングであっても、ローカル参照イメージストリームポリシーが保存されるようになりました。(BZ#1813420)

- 以前のバージョンでは、OpenShift 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#1821647)

- Multus 関連の DaemonSet は、その YAML 定義に **apps/v1** ではなく、非推奨のバージョンである **extensions/v1beta1** を誤って使用していました。これにより、非推奨の API の使用アラートが有効にされているクラスターのアラートが送信されました。DeamonSet は正しいバージョン名を使用するように更新されました。そのため、非推奨の API 使用のアラートは送信されなくなりました。(BZ#1824866)

- 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ました。このプロセスには、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することが含まれました。生成される 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。生成される Dockerfile には、元の Dockerfile には含まれない **=** 文字が含まれる場合がありました。これにより、ビルドが構文エラーを出して失敗しました。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ことにより、問題が修正されました。(BZ#1821861)

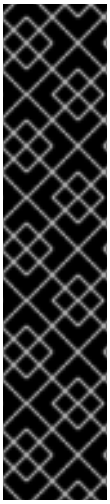
- Node Tuning Operator には、BZ#1702724 および BZ#1774645 に関連する tuned デーモンに対応する修正が同梱されていませんでした。そのため、ユーザーによって無効なプロファイルが指定されると、オペランドの機能のサービス拒否 (Denial of Service、DoS) が発生しました。また、プロファイルを訂正してもオペランドの機能は復元されませんでした。これは、前述のバグ修正を適用することで修正され、tuned デーモンが修正された新規プロファイルを処理し、設定できるようになりました。(BZ#1825007)

- 以前のリリースでは、OpenStack インストーラーは **remote_group_id** を使用してトラフィックの発信元を許可するセキュリティーグループを作成していました。セキュリティールールで **remote_group_id** を使用すると、OVS エージェントによる多くの計算をトリガーしてフローを生成するため非効率なプロセスが生じました。このプロセスでは、フローの生成に割り当てられた期間が超過することがありました。このような場合、すでに負荷がかかっている環境ではとくに、マスターノードはワーカーノードと通信できず、デプロイメントに失敗します。**remote_group_id** ではなく、トラフィックの発信元をホワイトリスト化するための IP プレフィックスが使用されるようになりました。これにより、Neutron リソースの負荷が軽減され、タイムアウトの発生回数が減ります。(BZ#1825973)

- 以前のバージョンでは、チューニングされた Pod はホストから **/etc/sysctl.{conf,d/}** をマウントしませんでした。これにより、ホストが提供する設定が tuned プロファイルでオーバーライドされることが可能になりました。**/etc/sysctl.{conf,d/}** がチューニングされる Pod のホストからマウントされるようになり、これにより、tuned プロファイルが **/etc/sysctl.{conf,d/}** のホスト sysctl 設定をオーバーライドしなくなりました。(BZ#1826167)

1.7.26.2. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。



重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.27. RHSA-2020:2009 - Important (重要): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-05-11

ose-cluster-image-registry-operator-container の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:2009 アドバイザリーに記載されています。

1.7.28. RHBA-2020:2129 - OpenShift Container Platform 4.3.21 バグ修正の更新

発行日: 2020-05-19

OpenShift Container Platform リリース 4.3.21 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:2128 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コンテナーイメージおよびバグ修正は、RHBA-2020:2129 アドバイザリーで提供されます。

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 4.3.21 コンテナーイメージの一覧

1.7.28.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.29. RHBA-2020:2184 - OpenShift Container Platform 4.3.22 バグ修正の更新

発行日: 2020-05-26

OpenShift Container Platform release 4.3.22 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:2183 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コンテナーイメージおよびバグ修正は、RHBA-2020:2184 アドバイザリーで提供されます。

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 4.3.22 コンテナーイメージの一覧

1.7.29.1. バグ修正

- 以前のバージョンでは、Samples Operator が s390x および ppc64le アーキテクチャーで実行されている場合にそのバージョンを報告できないため、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。この問題は修正され、これらのインストールは正常に完了するようになりました。(BZ#1779934)

- 今回の更新により、TLS 証明書および ETCDCTL API バージョンに関連する複数の環境変数は **/root/.profile** に書き込まれるようになりました。その結果、ユーザーが **oc rsh** を実行する際に、etcdctl コマンドはこれらの変数を手動で設定しなくても機能するようになりました。(BZ#1801430)

- レジストリー Pod のいずれかに関する問題がレジストリーの可用性に影響を与えることを防ぐために、レジストリーには、可能な場合に 2 つのレプリカが含まれるようになりました。(BZ#1810563)

- 以前のバージョンでは、Machine Health Check および Machine Config は視覚的に区分けされていませんでした。これら 2 つのアイテムの間に境界線が追加されました。(BZ#1819289)

- 以前のバージョンでは、Fluentd バッファーキューは制限されず、大量の受信ログにより、ノードのファイルシステムが一杯になり、クラッシュする可能性がありました。その結果として、アプリケーションはスケジュール変更されます。この種のクラッシュを防ぐために、Fluentd バッファーキューは出力ごとに固定した量のチャンクに制限されるようになりました (デフォルト: 32)。(BZ#1824427, BZ#1833226)

- 以前のバージョンでは、Dockerfile に ARG ステップがある OpenShift Docker Strategy Build では、buildah を呼び出す前にパニックが発生し、失敗していました。これは、ARG ステップの処理に必要なマップが初期化されないことによって生じました。今回の更新により、マップが初期化されるようになり、Dockerfile に ARG ステップがある OpenShift Docker Strategy Build で buildah を呼び出す前のパニックが発生しなくなりました。(BZ#1832975)

- 以前のバージョンでは、古い ImageStreamImport のエラーメッセージは、ユーザーが現在存在する ImageStreamImport の問題を把握する上で不明確でした。今回の更新により、ImageStreamImport エラーメッセージを更新するロジックが強化され、一連のエラーが別の原因から生じたかどうかを判断し、必要に応じてエラーメッセージを更新できるようになりました。これにより、ユーザーは ImageStreamImport の問題解決に必要なことを繰り返し試行した後により効果的なガイダンスを得られるようになりました。(BZ#1833019)

- 以前のバージョンでは、cluster-network-operator は、Kuryr ブートストラップで非推奨となったセキュリティーグループルールが新規ルールで置き換えられる際に、それらを削除しませんでした。そのため、非推奨のルールは 4.3.z リリース間で OCP のアップグレード時にそのまま残されていました。cluster-network-operator は、非推奨となったセキュリティーグループルールを削除するように更新され、Pod が 4.3.z のアップグレード後も適切なホストの VM アクセス制限を持つようになりました。(BZ#1834858)

1.7.29.2. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.30. RHBA-2020:2256 - OpenShift Container Platform 4.3.23 バグ修正の更新

発行日: 2020-06-02

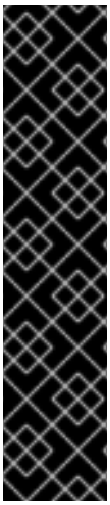
OpenShift Container Platform リリース 4.3.23 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:2255 アドバイザリーにまとめられています。この更新に含まれるコンテナーイメージおよびバグ修正は、RHBA-2020:2256 アドバイザリーで提供されます。

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 4.3.23 コンテナーイメージの一覧

1.7.30.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。



重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.31. RHBA-2020:2436 - OpenShift Container Platform 4.3.25 バグ修正の更新

発行日: 2020-06-16

OpenShift Container Platform リリース 4.3.25 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:2435 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コンテナーイメージおよびバグ修正は、RHBA-2020:2436 アドバイザリーで提供されます。

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 4.3.25 コンテナーイメージの一覧

1.7.31.1. バグ修正

- 以前のバージョンでは、ユーザーは間違ったマスターノードで **etc-member-add.sh** を実行すると、etcd がクォーラム (定足数) を失う可能性がありました。今回のリリースにより、追加のチェックが導入され、etcd が指定されたマスターノードですでに実行されている場合、ユーザーがスクリプトを実行できなくなりました。(BZ#1804067)

- 以前のバージョンでは、以前のリリースからのサンプルイメージストリームが後続のリリースで削除されると、見つからないイメージストリーム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イメージストリームの更新は試行されなくなりました。(BZ#1811206)

- 今回のリリースにより、ユーザーは **ConfigMap** オブジェクトから設定についてのデータを収集し、証明書がクラスター CA に使用されるかどうかを判別し、他のクラスター関連の設定を **openshift-config** namespace から収集できるようになりました。(BZ#1825758)

- Red Hat OpenShift Serverless 1 の Serverless Operator バージョン 1.7.1 のリリースにより、Operator は一般的に利用可能になりました。Web コンソールの Developer パースペクティブの Tech Preview バッジが削除されました。(BZ#1829046)

- 今回のリリースにより、問題のトラブルシューティングに役立つ承認されていない証明書サービス要求の匿名データを収集できるようになりました。(BZ#1835094)

- 以前のバージョンでは、Samples Operator は 存在しないサンプルコンテンツを見つけることができないために s390x アーキテクチャーのアップグレードを完了しませんでした。これにより、全体的なアップグレードが失敗しました。今回のリリースにより、Samples Operator は、s390x のアップグレード時にサンプルコンテンツの取得を試行しなくなりました。
  Samples Operator をパフォーマンスが低下した状態から戻すための回避策も実行できます。クラスター管理者は **oc delete config.samples cluster** を実行して、Samples Operator をリセットできます。(BZ#1835996)

- 以前のバージョンでは、API の制限により空のコンテナーが含まれる設定オブジェクトのブートストラップが許可されないため、イメージレジストリー Operator は Azure で IPI を使用する場合に作成されませんでした。本リリースでは、API の制限が削除されました。(BZ#1836753)

- 以前のバージョンでは、証明書ローテーションの正しくない処理により、Prometheus は **/metrics** エンドポイントからデータを取得できませんでした。この問題は本リリースでは解決されています。(BZ#1836939)

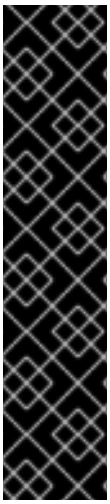
- 以前のバージョンでは、ユーザーが CLI または YAML を使用して PipelineRun を作成すると、Web コンソールは応答しなくなりました。今回の更新により、Web コンソールのエラーを回避するためにチェックが追加されました。(BZ#1839036)

- 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#1841996)

1.7.31.2. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。



重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.32. RHSA-2020:2439 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-06-16

machine-config-operator-container の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:2439 アドバイザリーに記載されています。

1.7.33. RHSA-2020:2440 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-06-16

Kubernetes の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:2440 アドバイザリーに記載されています。

1.7.34. RHSA-2020:2441 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-06-16

Kubernetes の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:2441 アドバイザリーに記載されています。

1.7.35. RHSA-2020:2442 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-06-16

openshift-enterprise-apb-tools-container の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:2442 アドバイザリーに記載されています。

1.7.36. RHSA-2020:2443 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-06-16

containernetworking-plugins の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:2443 アドバイザリーに記載されています。

1.7.37. RHBA-2020:2436 - OpenShift Container Platform 4.3.26 バグ修正の更新

発行日: 2020-06-23

OpenShift Container Platform リリース 4.3.26 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:2435 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コンテナーイメージおよびバグ修正は、RHBA-2020:2436 アドバイザリーで提供されます。

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 4.3.26 コンテナーイメージの一覧

1.7.37.1. 機能

1.7.37.1.1. 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 エージェント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 エージェントをインポートした新規の Node.js Jenkins エージェントで上書きします。

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

4. Jenkins ログで、新規の Jenkins エージェント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。

1.7.37.2. バグ修正

- OpenShift Container Platform 4.2 以降では、サンプルについての非接続クラスターのサポートが提供されるため、Samples Operator は、CVO インストールペイロードミラーを使用できるように samplesRegistry のアプリケーションが CVO ベースの Jenkins イメージストリームを上書きすることを許可する必要がありました。これにより、ミラーリングの対象として選択された場合でも CVO ペイロードの Jenkins イメージ仕様が **quay.io** または **registry.redhat.io** の同様の仕様と一致しないため、SamplesRegistry の上書きがより困難になりました。また、これらのレジストリーの Jenkins イメージは、ベース OpenShift インストールの一部であるため、Red Hat が Jenkins イメージについて提供する特殊なケースのサポート契約に準拠しませんでした。今回のバグ修正により、イメージレジストリーがインストールミラーが有効な場合に Jenkins イメージストリームのインポートを処理できるようになったため、Jenkins イメージストリームの samplesRegistry の上書きの使用が排除されるようになりました。Jenkins イメージストリームのインポートは、samplesOverride を使用して **registry.redhat.io** 外の他の場所から他のサンプルイメージストリームを取得する場合に機能するようになりました。(BZ#1826028)

- 以前のバージョンでは、Web コンソールには、Home → 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 Subscription の詳細が表示されるようになりました。(BZ#1827747)

- Azure インフラストラクチャー名は、生成される Azure コンテナーおよびストレージアカウントに使用されます。そのため、Azure インフラストラクチャー名に大文字が含まれる場合、コンテナーは正常に作成されますが、ストレージアカウントの作成は失敗しました。今回のバグ修正により、コンテナー名の作成ロジックが無効な文字を破棄するように調整され、イメージレジストリーを名前に無効な文字が含まれるインフラストラクチャーにデプロイできるようになりました。(BZ#1832144)

- 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#1844117)

1.7.37.3. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.38. RHSA-2020:2635 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-06-23

python-psutil の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:2635 アドバイザリーに記載されています。

1.7.39. RHBA-2020:2628 - OpenShift Container Platform 4.3.27 バグ修正の更新

発行日: 2020-06-30

OpenShift Container Platform リリース 4.3.27 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:2627 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コンテナーイメージおよびバグ修正は、RHBA-2020:2628 アドバイザリーで提供されます。

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 4.3.27 コンテナーイメージの一覧

1.7.39.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.40. RHBA-2020:2805 - OpenShift Container Platform 4.3.28 バグ修正の更新

発行日: 2020-07-07

OpenShift Container Platform release 4.3.28 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:2804 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コンテナーイメージおよびバグ修正は、RHBA-2020:2805 アドバイザリーで提供されます。

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 4.3.28 コンテナーイメージの一覧

1.7.40.1. バグ修正

- 以前のバージョンでは、トリガーコントローラーが StatefulSet を GA として認識しないため、トリガーは v1.StatefulSet オブジェクトでは機能しませんでした。この問題は本リリースでは解決されています。(**BZ#1831888**)

- 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#1835578**)

- 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が、既存の DNS ロードバランサーの古い Amphora イメージは新しいリスナーをサポートしないため、リスナーの作成が失敗します。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#1846459**)

- 以前のバージョンでは、Azure ディスク用の Kubernetes ボリュームプラグインは、新しい udev ルールがホストオペレーティングシステムにインストールされていることが予想されるため、割り当てられている Azure ボリュームを見つけることができませんでした。そのため、ボリュームを使用する Pod は RHEL 7 で起動できませんでした。今回のリリースにより、RHEL 7 の udev ルールを使用する場合でも、Azure ディスクの Kubernetes ボリュームプラグインが割り当てられている Azure ディスクついてスキャンするようになり、Azure ディスクボリュームを含む Pod が RHEL 7 で起動できるようになりました。(**BZ#1847089**)

- 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#1849171**)

1.7.40.2. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.41. RHBA-2020:2872 - OpenShift Container Platform 4.3.29 バグ修正の更新

発行日: 2020-07-14

OpenShift Container Platform release 4.3.29 が公開されました。この更新に含まれるパッケージの一覧は、RHBA-2020:2879 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コンテナーイメージおよびバグ修正は、RHBA-2020:2872 アドバイザリーで提供されます。

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 4.3.29 コンテナーイメージの一覧

1.7.41.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.42. RHBA-2020:3180 - OpenShift Container Platform 4.3.31 バグ修正の更新

発行日: 2020-08-05

OpenShift Container Platform リリース 4.3.31 が公開されました。この更新に含まれるコンテナーイメージおよびバグ修正の一覧は RHBA-2020:3180 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パッケージの一覧は、RHBA-2020:3179 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 4.3.31 コンテナーイメージの一覧

1.7.42.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。

重要

OpenShift Container Platform 4.2 または 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 を使用してい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。

1.7.43. RHSA-2020:3183 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-08-05

openshift の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:3183 アドバイザリーに記載されています。

1.7.44. RHSA-2020:3184 - Moderate (中程度): OpenShift Container Platform 4.3 セキュリティー更新

発行日: 2020-08-05

openshift-enterprise-hyperkube-container の更新が OpenShift Container Platform 4.3 で利用可能になりました。更新の詳細については、RHSA-2020:3184 アドバイザリーに記載されています。

1.7.45. RHBA-2020:3259 - OpenShift Container Platform 4.3.33 バグ修正の更新

発行日: 2020-08-19

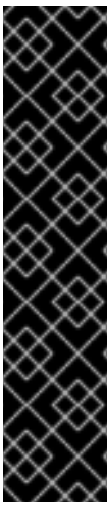
OpenShift Container Platform リリース 4.3.33 が公開されました。この更新に含まれるコンテナーイメージおよびバグ修正の一覧は RHBA-2020:3259 アドバイザリーにまとめられています。この更新に含まれるパッケージの一覧は、RHBA-2020:3258 アドバイザリーで提供されています。

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 4.3.33 コンテナーイメージの一覧

1.7.45.1. アップグレード

既存の OpenShift Container Platform 4.3 クラスターをこの最新リリースにアップグレードするには、CLI の使用によるクラスターの更新について参照してください。



重要

OpenShift Container Platform 4.2 または 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 を使用してい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、4.3 クラスターではすべてのマスターは 4.3 で、すべてのノードが 4.3 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 4.3 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** footnote:versionpolicyn[Where **N** は 2 以上。] (**oc** クライアント)
X.Y (サーバー)	1	3
X.Y+N footnote:versionpolicyn[] (サーバー)	2	1

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。