# OpenShift Container Platform 4.8

## 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: 2021-11-25

# OpenShift Container Platform 4.8 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 4.8 リリースノート ..... 7**
1.1. 本リリースについて 7
1.2. 多様性を受け入れるオープンソースの強化 7
1.3. OPENSHIFT CONTAINERPLATFORM のレイヤー化された依存関係にあるコンポーネントのサポートと互換性 8
1.4. 新機能および改良された機能 8
1.4.1. Red Hat Enterprise Linux CoreOS (RHCOS) 8
1.4.1.1. RHCOS が RHEL 8.4 を使用するように 8
1.4.1.2. ブートイメージの自動化を改善するためのストリームメタデータの使用 8
1.4.1.3. Butane config transpiler がマシン設定の作成を単純化する 8
1.4.1.4. クラウドプラットフォームにおけるカスタム chrony.conf のデフォルトへの変更 8
1.4.1.5. ベアメタルインストール時のマルチパスの有効化 9
1.4.2. インストールおよびアップグレード 9
1.4.2.1. クラスターの Azure の既存の空のリソースグループへのインストール 9
1.4.2.2. AWS でのクラスターの既存 IAM ロールの使用 9
1.4.2.3. AWS での既存の Route53 ホストプライベートゾーンの使用 9
1.4.2.4. マシン CIDR 内の GCP サブネットサイズの拡大 9
1.4.2.5. アップグレード期間の短縮 9
1.4.2.6. MCO は、すべてのマシン設定プールが更新を待ってから、更新の完了を報告します。 10
1.4.2.7. ベアメタルノードへのインストールでの Fujitsu iRMC の使用 10
1.4.2.8. RHOSP 上で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するクラスターに対する SR-IOV ネットワークサポート 11
1.4.2.9. VLAN インターフェースに対する Ironic Python Agent のサポート 11
1.4.2.10. OpenShift Update Service を使用した OTA (over-the-air) 更新 11
1.4.3. Web コンソール 11
1.4.3.1. カスタムコンソールルートは新規の CustomDomains クラスター API を使用する 11
1.4.3.2. クイックスタートからのコードスニペットへのアクセス 11
1.4.3.3. クイックスタートの前提条件の表示についての改善 11
1.4.4. IBM Z および LinuxONE 12
主な機能拡張 12
サポートされる機能 12
制限 13
1.4.5. IBM Power Systems 14
主な機能拡張 14
サポートされる機能 14
制限 15
1.4.6. セキュリティーおよびコンプライアンス 15
1.4.6.1. OAuth アクセストークンのログアウト要求についての監査ロギング 15
1.4.6.2. ヘッドレスサービスのサービス提供証明書のワイルドカードサブジェクト 15
1.4.6.3. oc-compliance プラグインが利用可能になりました。 16
1.4.6.4. Kubernetes コントロールプレーンの TLS セキュリティープロファイル 16
1.4.6.5. サーバーとしての kubelet の TLS セキュリティープロファイル 16
1.4.6.6. bcrypt パスワードハッシュのサポート 16
1.4.6.7. インストーラーでプロビジョニングされるクラスターでの管理対象 Secure Boot の有効化 16
1.4.7. ネットワーキング 17
1.4.7.1. OVN-Kubernetes クラスターネットワークプロバイダーを使用した、インストーラーでプロビジョニングされるベアメタルインフラストラクチャーでのデュアルスタックサポート 17
1.4.7.2. 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での OpenShift SDN から OVN-Kubernetes への移行 17
1.4.7.3. OpenShift SDN クラスターネットワークプロバイダーの egress IP 機能によるノード全体での分散 17

1.4.7.4. ネットワークポリシーによるホストネットワーク Ingress コントローラーの選択のサポート 17
1.4.7.5. ネットワークポリシーによるホストネットワークトラフィックの選択のサポート 17
1.4.7.6. ネットワークポリシー監査ログ 18
1.4.7.7. macvlan 追加ネットワークのネットワークポリシーサポート 18
1.4.7.8. SR-IOV でサポートされるハードウェア 18
1.4.7.9. SR-IOV Network Operator の機能拡張 18
1.4.7.10. ネットワークフローの追跡 18
1.4.7.11. CoreDNS-mDNS が IP アドレスにノード名を解決する際に使用されなくなる 19
1.4.7.12. OpenShift Container Platform 4.8 へのアップグレードをサポートする HTTP ヘッダー名の変換 19
1.4.7.13. GCP での Ingress コントローラーのグローバルアクセスの設定 19
1.4.7.14. Ingress コントローラースレッド数の設定 19
1.4.7.15. Ingress コントローラーの PROXY プロトコルの設定 19
1.4.7.16. コントロールプレーンノード上の NTP サーバー 19
1.4.7.17. Kuryr のデフォルト API ロードバランサー管理への変更 20
1.4.7.18. インストール後のプロビジョニングネットワークの有効化 20
1.4.7.19. コントロールプレーンで実行されるネットワークコンポーネントの設定 20
1.4.7.20. apiVIP および ingressVIP トラフィックの外部ロードバランサーの設定 20
1.4.7.21. デュアルスタックネットワークに対する OVN-Kubernetes IPsec のサポート 20
1.4.7.22. OVN-Kubernetes の egress ルーター CNI 20
1.4.7.23. OpenShift Container Platform での IP フェイルオーバーのサポート 21
1.4.7.24. DNS Pod 配置の制御 21
1.4.7.25. RHOSP で実行されるクラスターによるプロバイダーネットワークのサポート 21
1.4.7.26. HAProxy の設定可能な tune.maxrewrite および tune.bufsize 21
1.4.8. ストレージ 21
1.4.8.1. GCP PD CSI Driver Operator を使用した永続ストレージは一般に利用可能 21
1.4.8.2. Azure Disk CSI Driver Operator を使用した永続ストレージ (テクノロジープレビュー) 21
1.4.8.3. vSphere CSI Driver Operator を使用した永続ストレージ (テクノロジープレビュー) 22
1.4.8.4. CSI の自動移行 (テクノロジープレビュー) 22
1.4.8.5. AWS EFS (テクノロジープレビュー) 機能の外部プロビジョナーが削除される 22
1.4.8.6. RHOSP で実行されるクラスターの Cinder ボリュームアベイラビリティーゾーンの制御の強化 22
1.4.9. レジストリー 22
1.4.10. Operator ライフサイクル 22
1.4.10.1. 管理者用のエラーレポートの強化 22
1.4.10.2. インストール計画の再試行 23
1.4.10.3. 無効な Operator グループの指定 23
1.4.10.4. Operator 候補が見つからない場合の特定レポート 23
1.4.11. Operator の開発 23
1.4.11.1. Operator プロジェクトのパッケージマニフェスト形式からバンドル形式への移行 23
1.4.11.2. バンドルされた Operator を含むカタログの公開 24
1.4.11.3. Operator のアップグレードテストの強化 24
1.4.11.4. OpenShift Container Platform バージョンとの Operator 互換性の制御 24
ビルド 24
1.4.11.5. ストラテジーごとのビルド数の新しい Telemetry メトリクス 24
1.4.11.6. カスタム PKI 認証局のマウント 24
1.4.12. イメージ 25
1.4.13. マシン API 25
1.4.13.1. クラスター Autoscaler を使用した、vSphere で実行されるマシンの 0 への/からのスケーリング 25
1.4.13.2. kubelet-ca.crt の自動ローテーションでは、ノードのドレイン (解放) または再起動は必要ありません。 25
1.4.13.3. マシンセットポリシーの強化 25
1.4.13.4. マシンセットの hugepage の拡張機能 25
1.4.13.5. Machine Config Operator ImageContentSourcePolicy オブジェクトの機能拡張 25
1.4.14. ノード 26

1.4.14.1. Descheduler operator.openshift.io/v1 API グループが利用可能になる 26
1.4.14.2. Descheduler の Prometheus メトリクス 26
1.4.14.3. Downward API を使用した Huge Page のサポート 26
1.4.14.4. Node Feature Discovery Operator の新規ラベル 26
1.4.14.5. Poison Pill Operatorによる正常ではないノードの修復 27
1.4.14.6. kubelet-ca.crt の自動ローテーションに再起動は不要になる 27
1.4.14.7. Vertical Pod autoscaling が一般に利用可能になる 27
1.4.14.8. Vertical pod autoscaling の最小値が設定可能 27
1.4.14.9. ノードの CPU およびメモリーリソースの自動割り当て 27
1.4.14.10. イメージをプルするための特定のリポジトリーの追加 28
1.4.14.11. Cron ジョブは一般に利用可能 28
1.4.15. Red Hat OpenShift Logging 28
1.4.16. モニタリング 28
1.4.16.1. アラートルールの変更 28
1.4.16.2. 今後のリリースで削除される API が使用される際のアラートおよび情報 29
1.4.16.3. モニタリングスタックコンポーネントおよび依存関係のバージョン更新 29
1.4.16.4. kube-state-metrics はバージョン 2.0.0 にアップグレード 29
1.4.16.5. Grafana および Alertmanager UI リンクの削除 30
1.4.16.6. Web コンソールでのダッシュボードの機能拡張のモニタリング 30
1.4.17. メータリング 30
1.4.18. スケーリング 30
1.4.18.1. 単一ノードクラスターでの実行 30
1.4.18.2. Performance Addon Operator を使用した NIC の削減 31
1.4.18.3. クラスターの最大数 31
1.4.18.4. パフォーマンスプロファイルの作成 31
1.4.18.5. Node Feature Discovery Operator 31
1.4.18.6. ドライバツールキット (テクノロジープレビュー) 31
1.4.19. バックアップおよび復元 32
1.4.19.1. etcd スナップショットの強化 32
1.4.20. Insights Operator 32
1.4.20.1. ネットワークが制限された環境に関する Insights Advisor の推奨事項 32
1.4.20.2. Insights Advisor の改善 32
1.4.20.3. Insights Operator のデータ収集機能の拡張 32
1.4.20.4. 正常でない SAP Pod の Insights Operator の拡張機能 32
1.4.20.5. SAP Pod データを収集するための Insights Operator の拡張機能 33
1.4.21. 認証および認可 33
1.4.21.1. 認証情報の AWS Security Token Service (STS) を使用した OpenShift Container Platform の実行が一般に利用可能になる 33
1.4.22. OpenShift サンドボックスコンテナー 33
1.4.22.1. OpenShift Container Platform での OpenShift サンドボックスコンテナーのサポート (テクノロジープレビュー) 33
1.5. 主な技術上の変更点 33
Kuryr サービスサブネットの作成の変更 33
SHA-256 プレフィックスのない OAuth トークンが使用不可になる 34
Federal Risk and Authorization Management Program (FedRAMP) モードレート制御 34
Ingress コントローラーを HAProxy 2.2.13 にアップグレード 34
CoreDNS がバージョン 1.8.1 に更新される 34
etcd が zap ロガーを使用する 34
LSO 用にマージされた複数のデーモンセット 34
バインドされたサービスアカウントトークンボリュームが有効化されている 34
Operator SDK v1.8.0 35
1.6. 非推奨および削除された機能 35
1.6.1. 非推奨の機能 36

1.6.1.1. Descheduler operator.openshift.io/v1beta1 API グループが非推奨になる 36
1.6.1.2. Red Hat Enterprise Linux CoreOS (RHCOS) での dhclient の使用が非推奨になる 36
1.6.1.3. クラスターローダーが非推奨になる 37
1.6.1.4. ビルドの lastTriggeredImageID パラメーターは非推奨となりました。 37
1.6.1.5. Jenkins Operator (テクノロジープレビュー) が非推奨となりました。 37
1.6.1.6. Red Hat Virtualization (RHV) のinstance_type_idインストール設定パラメーター 37
1.6.2. 削除された機能 37
1.6.2.1. サンプルイメージストリームから削除されたイメージ 37
1.6.2.2. Operator のパッケージマニフェスト形式へのサポートの削除 38
1.6.2.3. Prometheus に基づく HPA カスタムメトリクスアダプターのサポート 38
1.6.2.4. セキュアなトークンストレージアノテーション認識が削除される 38
1.7. バグ修正 38
1.8. テクノロジープレビューの機能 68
1.9. 既知の問題 70
1.10. エラータの非同期更新 76
1.10.1. RHSA-2021:2438 - OpenShift Container Platform 4.8.2 イメージのリリース、バグ修正およびセキュリティー更新アドバイザリー 76
1.10.2. RHBA-2021:2896 - OpenShift Container Platform 4.8.3 バグ修正の更新 77
1.10.2.1. アップグレード 77
1.10.3. RHSA-2021:2983: OpenShift Container Platform 4.8.4 セキュリティーおよびバグ修正の更新 77
1.10.3.1. バグ修正 77
1.10.3.2. アップグレード 78
1.10.4. RHBA-2021:3121 - OpenShift Container Platform 4.8.5 バグ修正の更新 78
1.10.4.1. 特長 78
1.10.4.1.1. Egress IPの強化 78
1.10.4.2. バグ修正 78
1.10.4.3. アップグレード 79
1.10.5. RHBA-2021:3247: OpenShift Container Platform 4.8.9 セキュリティーおよびバグ修正の更新 79
1.10.5.1. バグ修正 80
1.10.5.2. アップグレード 80
1.10.6. RHBA-2021:3299 - OpenShift Container Platform 4.8.10 バグ修正の更新 80
1.10.6.1. アップグレード 80
1.10.7. RHBA-2021:3429 - OpenShift Container Platform 4.8.11 バグ修正の更新 80
1.10.7.1. バグ修正 81
1.10.7.2. アップグレード 81
1.10.8. RHBA-2021:3511 - OpenShift Container Platform 4.8.12 バグ修正の更新 81
1.10.8.1. 特長 81
1.10.8.1.1. クラスタに対する新しい最小ストレージ要件 81
1.10.8.2. バグ修正 81
1.10.8.3. アップグレード 81
1.10.9. RHBA-2021:3632 - OpenShift Container Platform 4.8.13 バグ修正およびセキュリティー更新 82
1.10.9.1. 特長 82
1.10.9.2. バグ修正 82
1.10.9.3. アップグレード 82
1.10.10. RHBA-2021:3682 - OpenShift Container Platform 4.8.14 バグ修正の更新 82
1.10.10.1. OpenShift Container Platformの次期リリースへのアップグレードの準備 82
1.10.10.2. バグ修正 83
1.10.10.3. アップグレード 83
1.10.11. RHBA-2021:3821 - OpenShift Container Platform 4.8.15 バグ修正およびセキュリティー更新 83
1.10.11.1. 既知の問題 83
1.10.11.2. バグ修正 84
1.10.11.3. アップグレード 84
1.10.12. RHBA-2021:3927 - OpenShift Container Platform 4.8.17 バグ修正およびセキュリティー更新 84

1.10.12.1. アップグレード 84
1.10.13. RHBA-2021:4020 - OpenShift Container Platform 4.8.18 バグ修正の更新 84
1.10.13.1. バグ修正 85
1.10.13.2. アップグレード 85
1.10.14. RHBA-2021:4109 - OpenShift Container Platform 4.8.19 バグ修正の更新 85
1.10.14.1. アップグレード 85
1.10.15. RHBA-2021:4574 - OpenShift Container Platform 4.8.20 バグ修正の更新 85
1.10.15.1. 既知の問題 85
1.10.15.2. アップグレード 86

**第2章 OPENSHIFT CONTAINER PLATFORM のバージョン管理ポリシー** ..... **87**

# 第1章 OPENSHIFT CONTAINER PLATFORM 4.8 リリースノート

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

Red Hat Enterprise Linux (RHEL) および Kubernetes にビルドされる OpenShift Container Platform は、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

OpenShift Container Platform (RHSA-2021:2438) が公開されました。本リリースでは、CRI-O ランタイムで Kubernetes 1.21 を使用します。以下では、OpenShift Container Platform 4.8 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 4.8.0 を GA バージョンとしてリリースせず、OpenShift Container Platform 4.8.2 を GA バージョンとしてリリースしています。

OpenShift Container Platform 4.8 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 4.8 は、Red Hat Enterprise Linux (RHEL) 7.9 以降、および Red Hat Enterprise Linux CoreOS (RHCOS) 4.8 でサポートされます。

コントロールプレーンには RHCOS マシンを使用する必要があり、コンピュートマシンには RHCOS または Red Hat Enterprise Linux (RHEL) 7.9 以降のいずれかを使用できます。

**重要**

RHEL 7.9 以降のみがコンピュートマシンでサポートされるため、RHEL コンピュートマシンを RHEL 8 にアップグレードすることはできません。

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

OpenShift Container Platform 4.8 のリリースでは、バージョン 4.5 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. 多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティーにおける配慮に欠ける用語の置き換えに取り組んでいます。

この作業の一環として、本リリースでは以下の変更を実施しています。

- OpenShift Docs GitHub リポジトリー の **master** ブランチの名前が **main** に変更されました。

- 「マスター」から「コントロールプレーン」への用語の置き換えを徐々に開始しています。ドキュメント全体にわたって、両方の用語を使用しているいることにお気づきいただけると思います (「マスター」は括弧で括っています)。たとえば、「... コントロールプレーンノード (別名マスターノード)」のように使用しています。今後のリリースでは、これを「コントロールプレーンノード」に更新します。

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

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

## 1.4. 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。

### 1.4.1. Red Hat Enterprise Linux CoreOS (RHCOS)

#### 1.4.1.1. RHCOS が RHEL 8.4 を使用するように

RHCOS は、OpenShift Container Platform 4.8およびOpenShift Container Platform 4.7.24 以降で、Red Hat Enterprise Linux (RHEL) 8.4 パッケージを使用するようになりました。これにより、最新の修正、機能、拡張機能、および最新のハードウェアサポートおよびドライバー更新を利用できます。OpenShift Container Platform 4.6 は延長アップデートサポート (EUS) リリースです。このリリースは、ライフサイクル全体に対して RHEL 8.2 EUS パッケージを引き続き使用します。

#### 1.4.1.2. ブートイメージの自動化を改善するためのストリームメタデータの使用

ストリームメタデータは、OpenShift Container Platform のインストール時に、メタデータをクラスターに挿入するための標準化された JSON 形式を提供します。自動化を強化するために、新規の **openshift-install coreos print-stream-json** コマンドは、スクリプト可能かつ機械読み取り可能な形式でストリームメタデータを出力する方法を提供します。

ユーザーによってプロビジョニングされるインストールの場合、**openshift-install** バイナリーには、AWS AMI などの OpenShift Container Platform での使用がテストされている RHCOS ブートイメージのバージョンへの参照が含まれます。https://github.com/coreos/stream-metadata-go の公式の **stream-metadata-go** ライブラリーを使用して、Goプログラムからストリームメタデータを解析できるようになりました。

詳細は、「ストリームメタデータを使用した RHCOS AMI へのアクセス」を参照してください。

#### 1.4.1.3. Butane config transpiler がマシン設定の作成を単純化する

OpenShift Container Platform には、マシン設定の生成および検証に役立つ Butane config transpiler が含まれるようになりました。ドキュメントでは、Butane を使用して LUKS ディスク暗号化、ブートディスクのミラーリング、およびカスタムカーネルモジュールのマシン設定を作成することを推奨しています。

詳細は、「Butane を使用したマシン設定の作成」を参照してください。

#### 1.4.1.4. クラウドプラットフォームにおけるカスタム chrony.conf のデフォルトへの変更

クラウド管理者がカスタムの **/etc/chrony.conf** 設定をすでに設定している場合、RHCOS はデフォルトで **PEERNTP=no** オプションをクラウドプラットフォームに設定しなくなりました。これを設定していない場合、**PEERNTP=no** オプションがデフォルトで設定されます。詳細は、BZ#1924869 を参照してください。

#### 1.4.1.5. ベアメタルインストール時のマルチパスの有効化

ベアメタルインストール時のマルチパスの有効化が、OpenShift Container Platform 4.8 以降でプロビジョニングされるノードでサポートされるようになりました。マルチパスを有効にするには、**coreos-installer install** コマンドにカーネル引数を追加して、インストール済みシステム自体が初回起動からマルチパスを使用できるようにします。インストール後のサポートはマシン設定を使用してマルチパスをアクティベートすることで引き続き利用できますが、OpenShift Container Platform 4.8 以降では、インストール中にプロビジョニングされたノードのマルチパスを有効化することをお勧めします。

詳細は、Enabling multipathing with kernel arguments on RHCOS について参照してください。

### 1.4.2. インストールおよびアップグレード

#### 1.4.2.1. クラスターの Azure の既存の空のリソースグループへのインストール

**install-config.yaml** ファイルの **platform.azure.resourceGroupName** フィールドを定義して、既存のリソースグループを定義し、クラスターを Azure にインストールできるようになりました。このリソースグループは空である必要があり、単一のクラスターにのみ使用する必要があります。クラスターのコンポーネントには、リソースグループ内のすべてのリソースの所有権があります。

インストールプログラムのサービスプリンシパルの範囲をこのリソースグループに制限する場合は、環境内でインストールプログラムが使用する他のすべてのリソースに、パブリック DNS ゾーンや仮想ネットワークなどの必要なパーミッションがあることを確認する必要があります。インストールプログラムを使用してクラスターを破棄すると、ユーザー定義のリソースグループが削除されます。

#### 1.4.2.2. AWS でのクラスターの既存 IAM ロールの使用

**install-config.yaml** ファイルに **compute.platform.aws.iamRole** および **controlPlane.platform.aws.iamRole** フィールドを設定して、マシンインスタンスのプロファイルに既存の Amazon Web Services (AWS) IAM ロールを定義できるようになりました。これにより、IAM ロールについて以下を実行できます。

- 命名スキームのマッチング
- 事前に定義されたパーミッション境界の追加

#### 1.4.2.3. AWS での既存の Route53 ホストプライベートゾーンの使用

**install-config.yaml** ファイルに **platform.aws.hostedZone** フィールドを設定して、クラスターの既存の Route 53 プライベートホストゾーンを定義できるようになりました。独自の VPC を指定する場合も、既存のホストゾーンのみを使用できます。

#### 1.4.2.4. マシン CIDR 内の GCP サブネットサイズの拡大

Google Cloud Platform (GCP) の OpenShift Container Platform インストールプログラムは、マシン CIDR 内に可能な限り大きなサブネットを作成するようになりました。これにより、クラスターがマシン CIDR のサイズを、クラスター内のノード数に対応するように設定できます。

#### 1.4.2.5. アップグレード期間の短縮

今回のリリースにより、デーモンセットをすべてのノードにデプロイするクラスター Operator のアップグレード期間が大幅に短縮されるようになりました。たとえば、250 ノードのテストクラスターのアップグレード期間は 7.5 時間から 1.5 時間に短縮されるため、アップグレード期間は 1 ノードの追加につき 1 分未満となります。

**注記**

この変更はマシン設定プールのロールアウト期間には影響しません。

#### 1.4.2.6. MCO は、すべてのマシン設定プールが更新を待ってから、更新の完了を報告します。

更新時に、Machine Config Operator (MCO) は、マシン設定プールの更新が完了していない場合に **Upgradeable=False** の状態を machine-config Cluster Operator に報告するようになりました。このステータスは今後のマイナー更新をブロックしますが、今後のパッチ更新をブロックしたり、現在の更新をブロックしたりすることはありません。以前のバージョンでは、MCO はワーカープールの更新が完了していなくても、コントロールプレーンマシンの設定プールの状態のみに基づいて、**Upgradeable** ステータスを報告していました。

#### 1.4.2.7. ベアメタルノードへのインストールでの Fujitsu iRMC の使用

OpenShift Container Platform 4.8 では、インストーラーでプロビジョニングされるクラスターをベアメタルにデプロイするときに、Fujitsu ハードウェアと Fujitsu iRMC ベースボード管理コントローラープロトコルを使用できます。現在 Fujitsu は、ベアメタルへのインストーラーでプロビジョニングされるインストール用に iRMC S5 ファームウェアバージョン **3.05P** 以降をサポートしています。OpenShift Container Platform 4.8 の機能拡張およびバグ修正には以下が含まれます。

- iRMC ハードウェアにおける通常の電源オフをサポートします。
- インストーラーがベアメタルノードにコントロールプレーンをデプロイすると、プロビジョニングサービスを停止します。詳細は、BZ#1949859 を参照してください。
- ブートストラップ **keepalived** チェックに Ironic ヘルスチェックを追加します。詳細は、BZ#1949859 を参照してください。
- コントロールプレーンノードでユニキャストピアリストが空ではないことを確認します。詳細は、BZ#1957708 を参照してください。
- Bare Metal Operator を更新して、iRMC PowerInterface に合わせます。詳細は、BZ#1957869 を参照してください。
- **pyghmi** ライブラリーバージョンを更新します。詳細は、BZ#1920294 を参照してください。
- Bare Metal Operator を更新して、不足している IPMI 認証情報に対処します。詳細は、BZ#1965182 を参照してください。
- **enabled_bios_interfaces** から iRMC を削除します。詳細は、BZ#1969212 を参照してください。
- **ironicTlsMount** および **inspectorTlsMount** をベアメタル Pod 定義に追加します。詳細は、BZ#1968701 を参照してください。
- iRMC サーバーの RAID 機能を無効化します。詳細は、BZ#1969487 を参照してください。
- すべてのドライバーで RAID を無効にします。詳細は、BZ#1969487 を参照してください。

#### 1.4.2.8. RHOSP 上で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するクラスターに対する SR-IOV ネットワークサポート

コンピュートマシン用に Single Root I/O Virtualization (SR-IOV) ネットワークを使用する RHOSP にクラスターをデプロイすることができるようになりました。

詳細は、「SR-IOV で接続されたコンピュートマシンをサポートする OpenStack へのクラスターのインストール」を参照してください。

#### 1.4.2.9. VLAN インターフェースに対する Ironic Python Agent のサポート

今回の更新により、Ironic Python Agent は、イントロスペクション中にインターフェース一覧で VLAN インターフェースを報告するようになりました。さらに、IP アドレスがインターフェースに含まれているため、CSR を適切に作成できます。これにより、VLAN インターフェースを含むすべてのインターフェースで CSR を取得できます。詳細は、BZ#1888712 を参照してください。

#### 1.4.2.10. OpenShift Update Service を使用した OTA (over-the-air) 更新

OpenShift Update Service (OSUS) は、Red Hat Enterprise Linux CoreOS (RHCOS) を含む OpenShift Container Platform に OTA (over-the-air) 更新を提供します。以前は、パブリック API の背後にある Red Hat ホストサービスとしてのみアクセス可能でしたが、現在はローカルにインストールできます。OpenShift Update Service は Operator および 1 つ以上のアプリケーションインスタンスで構成され、OpenShift Container Platform 4.6 以降で一般に利用可能になりました。

詳細は、「Understanding the OpenShift Update Service 」を参照してください。

### 1.4.3. Web コンソール

#### 1.4.3.1. カスタムコンソールルートは新規の CustomDomains クラスター API を使用する

**console** および **downloads** ルートについて、カスタムルート機能が新規 **ingress** 設定ルート設定 API の **spec.componentRoutes** を使用するようになりました。Console Operator 設定にはカスタムルートのカスタマイズがすでに含まれていますが、**console** ルートのみが対象です。**console-operator** 設定を使用したルート設定は非推奨になりました。そのため、**console** カスタムルートが **ingress** 設定と **console-operator** 設定の両方に設定されている場合、新規の **ingress** 設定のカスタムルート設定が優先されます。

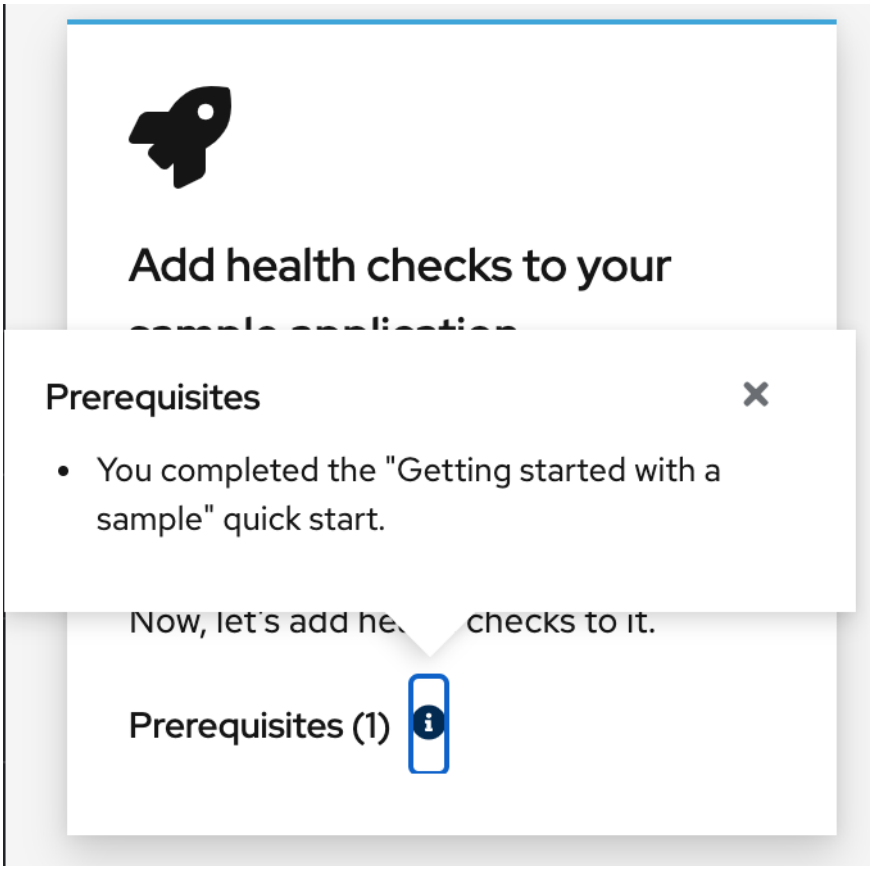
詳細は、「コンソールルートのカスタマイズ」を参照してください。

#### 1.4.3.2. クイックスタートからのコードスニペットへのアクセス

CLI スニペットがクイックスタートに含まれる場合、これを Web コンソールから実行できるようになりました。この機能を使用するには、まず Web ターミナル Operator をインストールする必要があります。Web ターミナルで実行する Web ターミナルおよびコードスニペットの各種アクションは、Web ターミナル Operator をインストールしない場合は表示されません。または、Web ターミナル Operator がインストールされているかどうかに関係なく、コードスニペットをクリップボードにコピーできます。

#### 1.4.3.3. クイックスタートの前提条件の表示についての改善

以前のバージョンでは、クイックスタートの前提条件はクイックスタートカードの一覧ではなく、統合されたテキストとして表示されていました。スケーラビリティーを考慮し、前提条件はカードではなくポップアップで表示されるようになりました。




### 1.4.4. IBM Z および LinuxONE

本リリースでは、IBM Z および LinuxONE は OpenShift Container Platform 4.8 と互換性があります。インストールは z/VM または RHEL KVM で実行できます。インストール手順については、以下のドキュメントを参照してください。

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

主な機能拡張
以下の新機能は、OpenShift Container Platform 4.8 の IBM Z および LinuxONE でサポートされます。

- RHEL 8.3 以降の KVM は、IBM Z および LinuxONE での OpenShift Container Platform 4.8 のユーザーによってプロビジョニングされるインストールのハイパーバイザーとしてサポートされます。静的 IP アドレスを使用したインストールや、ネットワークが制限された環境でのインストールもサポートされます。
- etcd に保存されるデータの暗号化
- 4k FCP ブロックデバイス
- 3 ノードクラスターのサポート

サポートされる機能

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

- マルチパス化
- iSCSI を使用した永続ストレージ
- ローカルボリュームを使用した永続ストレージ (Local Storage Operator)
- hostPath を使用した永続ストレージ
- ファイバーチャネルを使用した永続ストレージ
- Raw Block を使用した永続ストレージ
- OpenShift Container Platform 4.8 の初回インストールを含む OVN-Kubernetes
- SCSI ディスク上の z/VM Emulated FBA デバイス

これらの機能は、4.8 の IBM Z の OpenShift Container Platform についてのみ利用できます。

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

制限
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。
  - Precision Time Protocol (PTP) ハードウェア
- 以下の OpenShift Container Platform 機能はサポートされていません。
  - 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
  - CodeReady Containers (CRC)
  - 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
  - CSI ボリュームのクローン作成
  - CSI ボリュームスナップショット
  - FIPS 暗号
  - Helm コマンドラインインターフェース (CLI) ツール
  - Multus CNI プラグイン
  - NVMe
  - OpenShift Metering
  - OpenShift Virtualization
  - 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 (RHCOS) を実行する必要があります。
- 永続共有ストレージは、NFS またはその他のサポートされるストレージプロトコルを使用してプロビジョニングする必要があります。
- 共有されていない永続ストレージは、iSCSI、FC、DASD、FCP または EDEV/FBA と共に LSO を使用するなど、ローカルストレージを使用してプロビジョニングする必要があります。

### 1.4.5. IBM Power Systems

本リリースでは、IBM Power Systems は OpenShift Container Platform 4.8 と互換性があります。インストール手順については、以下のドキュメントを参照してください。

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

主な機能拡張
以下の新機能は、OpenShift Container Platform 4.8 の IBM Power Systems でサポートされます。

- etcd に保存されるデータの暗号化
- 3 ノードクラスターのサポート
- Multus SR-IOV

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

- 現時点で、5 つの Operator がサポートされています。
  - Cluster-Logging-Operator
  - Cluster-NFD-Operator
  - Elastic Search-Operator
  - Local Storage Operator
  - SR-IOV ネットワーク Operator
- マルチパス化
- iSCSI を使用した永続ストレージ
- ローカルボリュームを使用した永続ストレージ (Local Storage Operator)
- hostPath を使用した永続ストレージ
- ファイバーチャネルを使用した永続ストレージ
- Raw Block を使用した永続ストレージ
- OpenShift Container Platform 4.8 の初回インストールを含む OVN-Kubernetes
- 4K ディスクのサポート
- NVMe

制限
IBM Power Systems 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。
  - Precision Time Protocol (PTP) ハードウェア
- 以下の OpenShift Container Platform 機能はサポートされていません。
  - 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
  - CodeReady Containers (CRC)
  - 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
  - FIPS 暗号
  - Helm コマンドラインインターフェース (CLI) ツール
  - OpenShift Metering
  - OpenShift Virtualization
  - 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 (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)、または Container Storage Interface (CSI) を使用する Filesystem タイプである必要があります。

### 1.4.6. セキュリティーおよびコンプライアンス

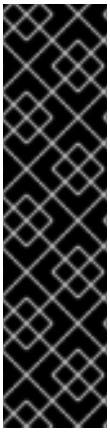
#### 1.4.6.1. OAuth アクセストークンのログアウト要求についての監査ロギング

**Default** 監査ログポリシーは、OAuth アクセストークンの作成（ログイン）および削除（ログアウト）要求の本体をログに記録するようになりました。以前のバージョンでは、削除要求の本体はログに記録されませんでした。

監査ログポリシーについての詳細は、ノード監査ログポリシーの設定について参照してください。

#### 1.4.6.2. ヘッドレスサービスのサービス提供証明書のワイルドカードサブジェクト

ヘッドレスサービスのサービス提供証明書を生成すると、***.<service.name>.<service.namespace>.svc** 形式のワイルドカードサブジェクトが組み込まれるようになりました。これにより、これらの Pod の証明書を手動で生成しなくても、個別のステートフルセット Pod への TLS で保護される接続を使用できます。



**重要**

生成された証明書にはヘッドレスサービスのワイルドカードサブジェクトが含まれるため、クライアントが個別の Pod を区別する必要がある場合はサービス CA を使用しないでください。この場合は、以下のようになります。

- 別の CA を使用して個別の TLS 証明書を生成します。
- サービス CA は、個々の Pod に送信される接続についての信頼される CA として許可することはできず、他の Pod がこの権限を借用することはできません。これらの接続は、個別の TLS 証明書の生成に使用されている CA を信頼するように設定される必要があります。

詳細は、「サービス証明書の追加」を参照してください。

#### 1.4.6.3. oc-compliance プラグインが利用可能になりました。

コンプライアンス Operator は、OpenShift Container Platform クラスターのチェックおよび修復の多くを自動化します。ただし、クラスターをコンプライアンスに準拠させる完全なプロセスでは、多くの場合、管理者がコンプライアンス Operator API や他のコンポーネントと対話する必要があります。**oc-compliance** プラグインが利用可能となり、プロセスが簡単になりました。

詳細は、「**oc-compliance** プラグインの使用」を参照してください。

#### 1.4.6.4. Kubernetes コントロールプレーンの TLS セキュリティープロファイル

Kubernetes API サーバーの TLS セキュリティープロファイル設定は、Kubernetes スケジューラーおよび Kubernetes コントローラーマネージャーでも適用されるようになりました。

詳細は、「TLS セキュリティープロファイルの設定」を参照してください。

#### 1.4.6.5. サーバーとしての kubelet の TLS セキュリティープロファイル

Kubernetes API サーバーの HTTP サーバーとして機能する場合に、kubelet の TLS セキュリティープロファイルを設定できるようになりました。

詳細は、「TLS セキュリティープロファイルの設定」を参照してください。

#### 1.4.6.6. bcrypt パスワードハッシュのサポート

以前のバージョンでは、**oauth-proxy** コマンドでは、認証に使用される **htpasswd** ファイルでの SHA-1 ハッシュ化されたパスワードの使用のみが許可されていました。**oauth-proxy** には、**bcrypt** パスワードハッシュを使用する **htpasswd** エントリーのサポートが含まれるようになりました。詳細は、BZ#1874322 を参照してください。

#### 1.4.6.7. インストーラーでプロビジョニングされるクラスターでの管理対象 Secure Boot の有効化

OpenShift Container Platform 4.8 は、プロビジョニングされたコントロールプレーンおよびワーカーノードの UEFI セキュアブートモードを自動的に有効にし、ノードの削除時にオフに戻すことをサポートします。この機能を使用するには、ノードの **bootMode** 設定を **install-config.yaml** ファイルで **UEFISecureBoot** に設定します。Red Hat は、第 10 世代 HPE ハードウェアまたは第 13 世代 Dell ハードウェア (ファームウェアバージョン **2.75.75.75** 以上を実行) で、管理対象 Secure Boot を使用したインストーラーでプロビジョニングされるインストールのみをサポートします。詳細は、Configuring managed Secure Boot in the install-config.yaml file を参照してください。

### 1.4.7. ネットワーキング

#### 1.4.7.1. OVN-Kubernetes クラスターネットワークプロバイダーを使用した、インストーラーでプロビジョニングされるベアメタルインフラストラクチャーでのデュアルスタックサポート

インストーラーでプロビジョニングされる ベアメタルインフラストラクチャー 上のクラスターの場合、OVN-Kubernetes クラスターネットワークプロバイダーは、IPv4 アドレスファミリーと IPv6 アドレスファミリーの両方をサポートします。

以前のバージョンの OpenShift Container Platform からアップグレードするインストーラーでプロビジョニングされるベアメタルクラスターの場合、デュアルスタックネットワークをサポートするようにクラスターを変換する必要があります。詳細は、IPv4/IPv6 デュアルスタックネットワークへの変換 を参照してください。

#### 1.4.7.2. 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での OpenShift SDN から OVN-Kubernetes への移行

OpenShift SDN クラスターネットワークプロバイダーの OVN-Kubernetes クラスターネットワークプロバイダーへの移行は、ユーザーによってプロビジョニングされるクラスターでサポートされます。詳細は、OpenShift SDN クラスターネットワークプロバイダーからの移行について参照してください。

#### 1.4.7.3. OpenShift SDN クラスターネットワークプロバイダーの egress IP 機能によるノード全体での分散

OpenShift SDN の egress IP 機能は、namespace に複数の egress IP アドレスが割り当てられている場合、その指定された namespace のノード全体にほぼ均等にネットワークトラフィックを分散するようになりました。それぞれの IP アドレスは異なるノードに存在する必要があります。詳細は、OpenShift SDN の「プロジェクトの egress IP の設定」を参照してください。

#### 1.4.7.4. ネットワークポリシーによるホストネットワーク Ingress コントローラーの選択のサポート

OpenShift SDN または OVN-Kubernetes クラスターネットワークプロバイダーを使用する場合、Ingress コントローラーがクラスターネットワークまたはホストネットワークで実行されるかどうかにかかわらず、ネットワークポリシールールの Ingress コントローラーからトラフィックを選択することができます。ネットワークポリシールールでは、**policy-group.network.openshift.io/ingress=""** namespace セレクターラベルが Ingress コントローラーからのトラフィックと一致します。**network.openshift.io/policy-group: ingress** namespace セレクターラベルは引き続き使用できますが、これはレガシーラベルであり、OpenShift Container Platformの将来のリリースで削除される可能性があります。

OpenShift Container Platform の以前のリリースでは、以下の制限がありました。

- OpenShift SDN クラスターネットワークプロバイダーを使用するクラスターは、**network.openshift.io/policy-group="ingress"** ラベルを **default** namespace に適用することによってのみ、ホストネットワーク上の Ingress コントローラーからトラフィックを選択できます。

- OVN-Kubernetes クラスターネットワークプロバイダーを使用するクラスターは、ホストネットワーク上の Ingress コントローラーからのトラフィックを選択できませんでした。

詳細は、「ネットワークポリシーについて」を参照してください。

#### 1.4.7.5. ネットワークポリシーによるホストネットワークトラフィックの選択のサポート

OVN-Kubernetes クラスターネットワークプロバイダーまたは OpenShift SDN クラスターネットワークプロバイダーのいずれかを使用する場合は、**policy-group.network.openshift.io/host-network: ""** namespace セレクターを使用して、ネットワークポリシールールでホストネットワークトラフィックを選択できます。

#### 1.4.7.6. ネットワークポリシー監査ログ

OVN-Kubernetes クラスターネットワークプロバイダーを使用する場合は、namespace でネットワークポリシーの監査ロギングを有効にできます。ログは syslog と互換性のある形式であり、ローカルに保存したり、UDP 接続を介して送信したり、UNIX ドメインソケットに送信したりできます。許可された接続、ドロップされた接続、または許可された接続とドロップされた接続の両方をログに記録するかどうかを指定できます。詳細は、「ネットワークポリシーイベントのロギング」を参照してください。

#### 1.4.7.7. macvlan 追加ネットワークのネットワークポリシーサポート

**NetworkPolicy** API を実装する **MultiNetworkPolicy** API を使用して、macvlan の追加ネットワークに適用するネットワークポリシーを作成できます。詳細は、「マルチネットワークポリシーの設定」を参照してください。

#### 1.4.7.8. SR-IOV でサポートされるハードウェア

OpenShift Container Platform 4.8 では、追加の Intel ハードウェアおよび Mellanox ハードウェアのサポートを追加します。

- Intel X710 コントローラーおよび XL710 コントローラー
- Mellanox ConnectX-5 Ex

詳細は、サポートされるデバイスについて参照してください。

#### 1.4.7.9. SR-IOV Network Operator の機能拡張

Operator でデプロイされる Network Resources Injector は、Downward API を使用して Huge Page の要求および制限についての情報を公開するように機能強化されました。Pod 仕様に Huge Page 要求または制限が含まれる場合、情報は **/etc/podnetinfo** パスで公開されます。

詳細は、Downward API の Huge Page リソースの挿入 について参照してください。

#### 1.4.7.10. ネットワークフローの追跡

OpenShift Container Platform 4.8 では、Pod ネットワーク上のネットワークフローについてのメタデータをネットワークフローコレクターに送信するためのサポートを追加します。以下のプロトコルがサポートされます。

- NetFlow
- sFlow
- IPFIX

パケットデータはネットワークフローコレクターに送信されません。プロトコル、ソースアドレス、宛先アドレス、ポート番号、バイト数、その他のパケットレベルの情報などのパケットレベルのメタデータはネットワークフローコネクターに送信されます。

詳細は、ネットワークフローの追跡について参照してください。

#### 1.4.7.11. CoreDNS-mDNS が IP アドレスにノード名を解決する際に使用されなくなる

OpenShift Container Platform 4.8 以降のリリースには、クラスターメンバーシップ情報を使用して A/AAAA レコードを生成する機能が含まれます。これにより、ノード名が IP アドレスに解決されます。ノードが API に登録されると、クラスターは CoreDNS-mDNS を使用せずにこれらのノード情報を分散させることができます。これにより、マルチキャスト DNS に関連付けられたネットワークトラフィックがなくなります。

#### 1.4.7.12. OpenShift Container Platform 4.8 へのアップグレードをサポートする HTTP ヘッダー名の変換

OpenShift Container Platform は HAProxy 2.2 に対して更新されます。これはデフォルトで、**Host: xyz.com** を **host: xyz.com** に変更するなど HTTP ヘッダー名を小文字に変換します。HTTP ヘッダー名の大文字/小文字を認識するレガシーアプリケーションの場合、Ingress コントローラーの **spec.httpHeaders.headerNameCaseAdjustments** API フィールドを使用して、レガシーアプリケーションが修正されるまでこれらのレガシーアプリケーションに対応します。HAProxy 2.2 が利用可能な場合、OpenShift Container Platform をアップグレードする前に、**spec.httpHeaders.headerNameCaseAdjustments** を使用して必要な設定を追加するようにしてください。

詳細は、HTTP ヘッダーケースの変換について参照してください。

#### 1.4.7.13. GCP での Ingress コントローラーのグローバルアクセスの設定

OpenShift Container Platform 4.8 では、内部ロードバランサーを使用して GCP で作成された Ingress コントローラーのグローバルアクセスオプションのサポートが追加されました。グローバルアクセスオプションが有効にされる場合、ロードバランサーと同じ VPC ネットワークおよびコンピュートリージョン内の任意のリージョンのクライアントは、クラスターで実行されるワークロードに到達できます。

詳細は、GCP での Ingress コントローラーのグローバルアクセスの設定について参照してください。

#### 1.4.7.14. Ingress コントローラースレッド数の設定

OpenShift Container Platform 4.8 では、スレッド数を設定し、クラスターが処理できる受信接続の量を増やすサポートを追加しています。

詳細は、Ingress コントローラースレッド数の設定について参照してください。

#### 1.4.7.15. Ingress コントローラーの PROXY プロトコルの設定

OpenShift Container Platform 4.8 は、とくに **HostNetwork** または **NodePortService** エンドポイント公開ストラテジーのタイプについて、クラウド以外のプラットフォームでの Ingress コントローラーの PROXY プロトコルの設定をサポートします。

詳細は、Ingress コントローラーの PROXY プロトコルの設定について参照してください。

#### 1.4.7.16. コントロールプレーンノード上の NTP サーバー

OpenShift Container Platform 4.8 では、インストーラーでプロビジョニングされるクラスターは、コントロールプレーンノードの Network Time Protocol (NTP) サーバーおよびワーカーノードの NTP クライアントを設定し、デプロイできます。これにより、ルーティング可能なネットワークから切断されている場合でも、ワーカーはコントロールプレーンノードの NTP サーバーから日時を取得できます。デプロイメント後に NTP サーバーおよび NTP クライアントを設定し、デプロイすることもできます。

#### 1.4.7.17. Kuryr のデフォルト API ロードバランサー管理への変更

Kuryr-Kubernetes を使用した Red Hat OpenStack Platform (RHOSP) への OpenShift Container Platform 4.8 デプロイメントでは、**default/kubernetes** サービスの API ロードバランサーは Cluster Network Operator (CNO) によって管理されなくなり、kuryr-controller 自体により管理されます。これは以下を意味します。

- OpenShift Container Platform 4.8 にアップグレードする場合、**default/kubernetes** サービスにはダウンタイムがあります。

**注記**

Open Virtual Network (OVN) Octavia が利用できないデプロイメントでは、より多くのダウンタイムが予想されます。

- **default/kubernetes** ロードバランサーを Octavia Amphora ドライバーを使用するために使用する必要がなくなります。その代わりに、OVN Octavia は OpenStack クラウドで利用可能な場合に **default/kubernetes** サービスを実装するために使用されます。

#### 1.4.7.18. インストール後のプロビジョニングネットワークの有効化

ベアメタルクラスター用の支援付きインストーラーおよびインストーラーでプロビジョニングされるインストールは、**provisioning** ネットワークなしでクラスターをデプロイする機能を提供します。OpenShift Container Platform 4.8 以降では、Cluster Baremetal Operator (CBO) を使用してインストール後に **provisioning** ネットワークを有効にすることができます。

#### 1.4.7.19. コントロールプレーンで実行されるネットワークコンポーネントの設定

ベアメタルインストールのコントロールプレーンノードで実行する仮想 IP (VIP) アドレスが必要な場合は、コントロールプレーンノードでのみ実行する **apiVIP** および **ingressVIP** VIP アドレスを設定する必要があります。デフォルトで、OpenShift Container Platform は、ワーカーマシン設定プールの任意のノードが **apiVIP** および **ingressVIP** VIP アドレスをホストできるようにします。多くのベアメタル環境では、コントロールプレーンノードとは別のサブネットにワーカーノードをデプロイするため、コントロールプレーンノードで **apiVIP** および **ingressVIP** 仮想 IP アドレスを排他的に実行するように設定すると、ワーカーノードを別のサブネットにデプロイすることに関連して発生する問題を防ぐことができます。詳細は、「コントロールプレーンで実行されるネットワークコンポーネントの設定」を参照してください。

#### 1.4.7.20. apiVIP および ingressVIP トラフィックの外部ロードバランサーの設定

OpenShift Container Platform 4.8 では、インストーラーでプロビジョニングされるクラスターのコントロールプレーンへの **apiVIP** および **ingressVIP** トラフィックを処理するように外部ロードバランサーを設定できます。外部の負荷分散サービスとコントロールプレーンノードは同じ L2 ネットワークで実行する必要があります。また、VLAN を使用して負荷分散サービスとコントロールプレーンノード間のトラフィックをルーティングする際に同じ VLAN で実行する必要があります。

#### 1.4.7.21. デュアルスタックネットワークに対する OVN-Kubernetes IPsec のサポート

OpenShift Container Platform 4.8 では、デュアルスタックネットワークを使用するように設定されているクラスターに対して、OVN-Kubernetes IPsec サポートを追加しています。

#### 1.4.7.22. OVN-Kubernetes の egress ルーター CNI

egress ルーター CNI プラグインは一般に利用可能です。Cluster Network Operator は、**EgressRouter**

API オブジェクトをサポートするように強化されています。OVN-Kubernetes を使用するクラスターに egress ルーターを追加するプロセスは簡素化されます。egress ルーターオブジェクトの作成時に、Operator はネットワーク接続定義とデプロイメントを自動的に追加します。デプロイメントの Pod は egress ルーターとして機能します。

詳細は、「egress ルーター Pod の使用についての考慮事項」を参照してください。

#### 1.4.7.23. OpenShift Container Platform での IP フェイルオーバーのサポート

IP フェイルオーバーがベアメタルの OpenShift Container Platform クラスターでサポートされるようになりました。IP フェイルオーバーは **keepalived** を使用して、一連のホストでの外部からアクセスできる VIP アドレスのセットをホストします。各 VIP は 1 度に 1 つのホストによって提供されます。**keepalived** は Virtual Router Redundancy Protocol (VRRP) を使用して、(一連のホストの) どのホストがどの VIP を提供するかを判別します。ホストが利用不可の場合や **keepalived** が監視しているサービスが応答しない場合は、VIP は一連のホストの別のホストに切り換えられます。したがって、VIP はホストが利用可能である限り常に提供されます。

詳細は、「IP フェイルオーバーの設定」を参照してください。

#### 1.4.7.24. DNS Pod 配置の制御

OpenShift Container Platform 4.8では、カスタムノードセレクターと容認を使用して、CoreDNS を特定のノードで実行または実行しないようにデーモンセットを設定できます。

詳細は、「DNS Pod 配置の制御」を参照してください。

#### 1.4.7.25. RHOSP で実行されるクラスターによるプロバイダーネットワークのサポート

Red Hat OpenStack Platform (RHOSP) の OpenShift Container Platform クラスターは、すべてのデプロイメントタイプのプロバイダーネットワークをサポートするようになりました。

#### 1.4.7.26. HAProxy の設定可能な tune.maxrewrite および tune.bufsize

クラスター管理者は、**headerBufferMaxRewriteByte** および **headerBufferBytes** Ingress コントローラーの調整パラメーターを設定して、Ingress コントローラーごとに **tune.maxrewrite** および **tune.bufsize** HAProxy メモリーオプションを設定できるようになりました。

詳細は、「Ingress コントローラー設定パラメーター」を参照してください。

### 1.4.8. ストレージ

#### 1.4.8.1. GCP PD CSI Driver Operator を使用した永続ストレージは一般に利用可能

Google Cloud Platform (GCP) 永続ディスク (PD) Container Storage Interface (CSI) ドライバーは、GCP 環境に自動的にデプロイされ、管理されるため、ドライバーを手動でインストールしなくてもこれらのボリュームを動的にプロビジョニングできます。この機能は以前は OpenShift Container Platform 4.7 のテクノロジープレビュー機能として導入されましたが、OpenShift Container Platform 4.8 では一般に利用可能となり、デフォルトで有効にされます。

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

#### 1.4.8.2. Azure Disk CSI Driver Operator を使用した永続ストレージ (テクノロジープレビュー)

Azure Disk CSI Driver Operator は、永続ボリューム要求 (PVC) の作成に使用できるデフォルトのストレージクラスを提供します。このドライバーを管理する Azure Disk CSI Driver Operator はテクノロジープレビュー機能としてご利用いただけます。

詳細は、「Azure Disk CSI Driver Operator」を参照してください。

#### 1.4.8.3. vSphere CSI Driver Operator を使用した永続ストレージ (テクノロジープレビュー)

vSphere CSI Driver Operator は、永続ボリューム要求 (PVC) の作成に使用できるデフォルトのストレージクラスを提供します。このドライバーを管理する vSphere CSI Driver Operator はテクノロジープレビュー機能としてご利用いただけます。

詳細は、「vSphere CSI Driver Operator」を参照してください。

#### 1.4.8.4. CSI の自動移行 (テクノロジープレビュー)

OpenShift Container Platform 4.8 以降では、以下の in-tree ボリュームプラグインの同等の CSI ドライバーへの自動移行が、テクノロジープレビュー機能として利用可能になりました。

- Amazon Web Services (AWS) Elastic Block Storage (EBS)
- OpenStack Cinder

詳細は、「CSI の自動移行」を参照してください。

#### 1.4.8.5. AWS EFS (テクノロジープレビュー) 機能の外部プロビジョナーが削除される

Amazon Web Services (AWS) Elastic File System (EFS) テクノロジープレビュー機能が削除され、サポートされなくなりました。

#### 1.4.8.6. RHOSP で実行されるクラスターの Cinder ボリュームアベイラビリティーゾーンの制御の強化

インストール時に Cinder ボリュームのアベイラビリティーゾーンを選択できるようになりました。イメージレジストリー の特定アベイラビリティーゾーンで Cinder ボリュームを使用することもできます。

### 1.4.9. レジストリー

### 1.4.10. Operator ライフサイクル

#### 1.4.10.1. 管理者用のエラーレポートの強化

Operator Lifecycle Manager (OLM) を使用して Operator をインストールするクラスター管理者の場合、現在の API または低レベルの API に関連したエラー状態が生じる可能性があります。以前のバージョンでは、OLM が Operator をインストールしたり、更新したりする要求を満たせないことに関する洞察を得ることがほとんどできませんでした。これらのエラーは、オブジェクトプロパティーの誤字や RBAC が欠落しているなどの簡単な問題から、メタデータの解析により項目をカタログから読み込むことができないなどのより複雑な問題にまで及びます。

管理者は各種の低レベル API 間の対話プロセスへの理解や、これらの問題のデバッグのために OLM Pod ログにアクセスする必要がないため、OpenShift Container Platform 4.8 では OLM に以下の拡張機能を導入し、管理者により包括的なエラーレポートおよびメッセージが提供されるようになりました。

#### 1.4.10.2. インストール計画の再試行

**InstallPlan** オブジェクトで定義されるインストール計画では、API サーバーの可用性や他のライターとの競合などによる一時的なエラーが発生する可能性があります。以前のバージョンでは、これらのエラーにより、手動クリーンアップが必要な部分的に適用されるインストール計画が終了しました。今回の機能拡張により、カタログ Operator は、インストール計画の実行中に最大 1 分間エラーを再試行するようになりました。新規の **.status.message** フィールドには、再試行の実行時に人間が判読できる通知が提供されます。

#### 1.4.10.3. 無効な Operator グループの指定

以前のバージョンでは、Operator グループまたは複数の Operator グループのない namespace でサブスクリプションを作成すると、**phase=Installing** のままになるインストール計画と共に Operator のインストールが停止されました。今回の機能拡張により、インストール計画が **phase=Failed** にすぐに移行し、管理者が無効な Operator グループを修正してからサブスクリプションを再度削除し、作成できるようになりました。

#### 1.4.10.4. Operator 候補が見つからない場合の特定レポート

**ResolutionFailed** イベントは、namespace での依存関係の解決に失敗すると作成されますが、namespace に参照されるカタログソースに存在しないパッケージまたはチャネルを参照するサブスクリプションが含まれる場合に、より具体的なテキストを提供するようになりました。以前のバージョンでは、このメッセージは汎用的なメッセージでした。

```
no candidate operators found matching the spec of subscription '<name>'
```

今回の機能拡張により、メッセージがより具体的になりました。

##### Operator does not exist

```
no operators found in package <name> in the catalog referenced by subscription <name>
```

##### Catalog does not exist

```
no operators found from catalog <name> in namespace openshift-marketplace referenced by subscription <name>
```

##### Channel does not exist

```
no operators found in channel <name> of package <name> in the catalog referenced by subscription <name>
```

##### Cluster service version (CSV) does not exist

```
no operators found with name <name>.<version> in channel <name> of package <name> in the catalog referenced by subscription <name>
```

### 1.4.11. Operator の開発

#### 1.4.11.1. Operator プロジェクトのパッケージマニフェスト形式からバンドル形式への移行

Operator のレガシーパッケージマニフェスト形式のサポートは、OpenShift Container Platform 4.8 以

降で削除されます。バンドル形式は、OpenShift Container Platform 4.6 以降の Operator Lifecycle Manager (OLM) の推奨 Operator パッケージ形式です。非推奨のパッケージマニフェスト形式で最初に作成された Operator プロジェクトがある場合、Operator SDK **pkgman-to-bundle** コマンドを使用してプロジェクトをバンドル形式に移行できます。

詳細は、「パッケージマニフェストプロジェクトのバンドル形式への移行」を参照してください。

#### 1.4.11.2. バンドルされた Operator を含むカタログの公開

Operator をインストールおよび管理するには、Operator Lifecycle Manager (OLM) では、Operator バンドルがクラスターのカタログで参照されるインデックスイメージに一覧表示される必要があります。Operator の作成者は、Operator SDK を使用して Operator のバンドルおよびそれらのすべての依存関係を含むインデックスを作成できます。これは、リモートクラスターでのテストおよびコンテナーレジストリーへの公開に役立ちます。

詳細は、「バンドルされた Operator を含むカタログの公開」を参照してください。

#### 1.4.11.3. Operator のアップグレードテストの強化

Operator SDK の **run bundle-upgrade** サブコマンドは、より新しいバージョンのバンドルイメージを指定することにより、インストールされた Operator をトリガーしてそのバージョンにアップグレードするプロセスを自動化します。以前のバージョンでは、サブコマンドは、**run bundle** サブコマンドを使用して最初にインストールした Operator のみをアップグレードすることができました。今回の機能拡張により、**run bundle-upgrade** は、従来の Operator Lifecycle Manager (OLM) ワークフローで最初にインストールされた Operator でも機能するようになりました。

詳細は、「Operator Lifecycle Manager での Operator アップグレードのテスト」を参照してください。

#### 1.4.11.4. OpenShift Container Platform バージョンとの Operator 互換性の制御

API が OpenShift Container Platform バージョンから削除されると、削除された API を依然として使用しているクラスターバージョンで実行されている Operator が適切に機能しなくなります。Operator の作成者は、Operator ユーザーの中断を回避するために、API の非推奨および削除に対応するように Operator プロジェクトを更新する計画を立てる必要があります。

詳細は、「OpenShift Container Platform バージョンとの Operator 互換性の制御」を参照してください。

### ビルド

#### 1.4.11.5. ストラテジーごとのビルド数の新しい Telemetry メトリクス

Telemetry には新規の **openshift:build_by_strategy:sum** 測定メトリクスが含まれており、このメトリクスは、ストラテジータイプごとのビルド数を Telemeter クライアントに送信します。このメトリクスにより、サイト信頼性エンジニア (SRE) と製品マネージャーは、OpenShift Container Platform クラスターで実行されるビルドの種類を可視化できます。

#### 1.4.11.6. カスタム PKI 認証局のマウント

以前のバージョンでは、ビルドは、企業アーティファクトリポジトリーへのアクセスに必要なことがあるクラスターの PKI 認証局を使用できませんでした。現在では、**mountTrustedCA** を **true** に設定することにより、クラスターのカスタム PKI 認証局をマウントするように **BuildConfig** オブジェクトを設定できるようになりました。

### 1.4.12. イメージ

### 1.4.13. マシン API

#### 1.4.13.1. クラスター Autoscaler を使用した、vSphere で実行されるマシンの 0 への/からのスケーリング

vSphere でマシンを実行している場合、**MachineAutoscaler** リソース定義で **minReplicas** 値を **0** に設定できるようになりました。この値を **0** に設定すると、クラスター Autoscaler は、マシンが使用中であるかどうかに応じてマシンセットを 0 に/からスケーリングします。詳細は、「MachineAutoscaler リソース定義」を参照してください。

#### 1.4.13.2. kubelet-ca.crt の自動ローテーションでは、ノードのドレイン (解放) または再起動は必要ありません。

**/etc/kubernetes/kubelet-ca.crt** 認証局 (CA) の自動ローテーションでは、ノードのドレイン (解放) またはクラスターの再起動において Machine Config Operator (MCO) は不要になりました。

この変更の一環として、以下の変更では MCO によるノードのドレイン (解放) が不要になりました。

- マシン設定の **spec.config.ignition.passwd.users.sshAuthorizedKeys** パラメーターの SSH キーへの変更
- **openshift-config** namespace でのグローバルプルシークレットまたはプルシークレットへの変更

MCO がこれらの変更のいずれかを検出すると、その変更を適用し、ノードの遮断を解除します。

詳細は、「Machine Config Operator について」を参照してください。

#### 1.4.13.3. マシンセットポリシーの強化

以前のバージョンでは、マシンセットを作成するには、ホストのパフォーマンスを向上させるために、ユーザーが CPU ピニング設定、NUMA ピニング設定、および CPU トポロジーの変更を手動で設定する必要がありました。今回の機能拡張により、ユーザーは **MachineSet** リソースのポリシーを選択し、設定を自動的に設定できるようになりました。詳細は、BZ#1941334 を参照してください。

#### 1.4.13.4. マシンセットの hugepage の拡張機能

**hugepages** プロパティーを **MachineSet** リソースに提供できるようになりました。今回の機能拡張により、oVirt のカスタムプロパティーで **MachineSet** リソースのノードが作成され、それらのノードにハイパーバイザーの **hugepages** を使用するように指示されるようになりました。詳細は、BZ#1948963 を参照してください。

#### 1.4.13.5. Machine Config Operator ImageContentSourcePolicy オブジェクトの機能拡張

OpenShift Container Platform 4.8 は、一部の **ImageContentSourcePolicy** オブジェクト変更のワークロードの中断を防ぎます。この機能により、ユーザーおよびチームはワークロードの中断なしにミラーおよびレジストリーを追加することができます。その結果、**/etc/containers/registries.conf** ファイルの以下の変更に対して、ワークロードの中断は発生しなくなりました。

- **mirror-by-digest-only=true** を含むレジストリーの追加
- **mirror-by-digest-only=true** を含むレジストリーへのミラーの追加

- **unqualified-search-registries** 一覧への項目の追加

**/etc/containers/registries.conf** ファイルのその他の変更については、Machine Config Operator はデフォルトでノードをドレイン (解放) して変更を適用します。詳細は、BZ#1943315 を参照してください。

### 1.4.14. ノード

#### 1.4.14.1. Descheduler operator.openshift.io/v1 API グループが利用可能になる

**operator.openshift.io/v1** API グループが Descheduler で利用可能になりました。Descheduler の **operator.openshift.io/v1beta1** API グループのサポートは今後のリリースで削除される可能性があります。

#### 1.4.14.2. Descheduler の Prometheus メトリクス

Descheduler をインストールした **openshift-kube-descheduler-operator** namespace に **openshift.io/cluster-monitoring=true** ラベルを追加することで、 Descheduler の Prometheus メトリクスを有効にできるようになりました。

以下の Descheduler メトリクスを利用できます。

- **descheduler_build_info**: Descheduler に関するビルド情報を提供します。
- **descheduler_pods_evicted:** ストラテジー、namespace、および結果の組み合わせごとにエビクトされた Pod 数を提供します。このメトリクスを表示するには、少なくとも 1 つのエビクトされた Pod が必要です。

#### 1.4.14.3. Downward API を使用した Huge Page のサポート

今回のリリースにより、Pod 仕様の Huge Page の要求および制限を設定する際に、Downward API を使用してコンテナー内から Pod の割り当てを表示できるようになりました。この拡張機能は **DownwardAPIHugePages** 機能ゲートに依存します。OpenShift Container Platform 4.8 は機能ゲートを有効にします。

詳細は、Downward API を使用した Huge Page リソースの消費 について参照してください。

#### 1.4.14.4. Node Feature Discovery Operator の新規ラベル

Node Feature Discovery (NFD) Operator は、OpenShift Container Platform クラスターの各ノードで利用可能なハードウェア機能を検出します。次に、これはノードラベルでノードオブジェクトを変更します。これにより、NFD Operator は特定のノードの機能を公開できます。OpenShift Container Platform 4.8 は、NFD Operator の 3 つの追加のラベルをサポートします。

- **pstate intel-pstate**: Intel **pstate** ドライバーが有効であり、使用中の場合、**pstate intel-pstate** ラベルには Intel **pstate** ドライバーのステータスが反映されます。ステータスは **active** または **passive** のいずれかです。
- **pstate scaling_governor**: Intel **pstate** ドライバーのステータスが **active** の場合、**pstate scaling_governor** ラベルには scaling governor アルゴリズムが反映されます。アルゴリズムは **powersave** または **performance** のいずれかです。
- **cstate status**: **intel_idle** ドライバーに C-states または idle 状態がある場合、 **cstate status** ラベルは **true** になります。そうでない場合は **false** になります。

#### 1.4.14.5. Poison Pill Operatorによる正常ではないノードの修復

Poison Pill Operatorを使って、正常ではないノードが自動的に再起動するようにできます。これにより、ステートフルアプリケーションとReadWriteOnce（RWO）ボリュームのダウンタイムを最小限に抑え、一時的な障害が発生した場合に計算能力を回復します。

Poison Pill Operatorは、あらゆる種類のクラスタとハードウェアで動作します。

詳しくは、Remediating nodes with the Poison Pill Operator をご覧ください。

#### 1.4.14.6. kubelet-ca.crt の自動ローテーションに再起動は不要になる

**/etc/kubernetes/kubelet-ca.crt** 認証局 (CA) の自動ローテーションでは、ノードのドレイン (解放) またはクラスターの再起動において Machine Config Operator (MCO) は不要になりました。

この変更の一環として、以下の変更では MCO によるノードのドレイン (解放) が不要になりました。

- マシン設定の **spec.config.ignition.passwd.users.sshAuthorizedKeys** パラメーターの SSH キーへの変更
- **openshift-config** namespace でのグローバルプルシークレットまたはプルシークレットへの変更

MCO がこれらの変更のいずれかを検出すると、その変更を適用し、ノードの遮断を解除します。

詳細は、「Machine Config Operator について」を参照してください。

#### 1.4.14.7. Vertical Pod autoscaling が一般に利用可能になる

OpenShift Container Platform vertical pod autoscaler (VPA) が一般に利用可能になりました。VPA は、Pod 内のコンテナーの履歴および現在の CPU とメモリーリソースを自動的に確認し、確認された使用についての値に基づいてリソース制限および要求を更新できます。

以下のように **VerticalPodAutoscalerController** オブジェクトを変更して、1 つのレプリカのみを必要とする Pod で VPA を使用することもできます。以前のバージョンでは、VPA は 2 つ以上のレプリカを必要とする Pod でのみ機能しました。

詳細は、vertical pod autoscaler を使用した Pod リソースレベルの自動調整 について参照してください。

#### 1.4.14.8. Vertical pod autoscaling の最小値が設定可能

デフォルトで、ワークロードオブジェクトは、VPA が Pod を自動的に更新できるようにするためにレプリカを 2 つ以上指定する必要があります。そのため、2 つ未満を指定するワークロードオブジェクトの場合 VPA は機能しません。このクラスター全体の最小値は、**VerticalPodAutoscalerController** オブジェクトを変更して **minReplicas** パラメーターを追加することで変更できます。

詳細は、vertical pod autoscaler を使用した Pod リソースレベルの自動調整 について参照してください。

#### 1.4.14.9. ノードの CPU およびメモリーリソースの自動割り当て

OpenShift Container Platform は、ノードの起動時に **system-reserved** 設定の最適なサイジング値を自動的に判別できます。以前のバージョンでは、**system-reserved** 設定の CPU およびメモリーの割り当ては、手動で決定し、設定する必要のある固定された制限値でした。

リソースの自動割り当てが有効な場合、各ノードのスクリプトにより、ノードにインストールされている CPU およびメモリーの容量に基づいて、予約されたそれぞれのリソースに最適な値が計算されます。

詳細は、ノードのリソースの自動割り当てについて参照してください。

#### 1.4.14.10. イメージをプルするための特定のリポジトリーの追加

イメージのプルおよびプッシュ用に許可され、ブロックされるレジストリーの一覧を作成する際に、レジストリー内に個別のリポジトリーを指定できるようになりました。以前のバージョンでは、レジストリーのみを指定できました。

詳細は、「特定レジストリーの追加」および「特定レジストリーのブロック」を参照してください。

#### 1.4.14.11. Cron ジョブは一般に利用可能

cron ジョブカスタムリソースが一般に利用できるようになりました。この変更の一環として、cron ジョブのパフォーマンスを大幅に向上させる新しいコントローラーが実装されました。cron ジョブの詳細は、「ジョブと Cron ジョブについて」を参照してください。

### 1.4.15. Red Hat OpenShift Logging

OpenShift Container Platform 4.7 では、**Cluster Logging** は **Red Hat OpenShift Logging** になりました。詳細は、「Red Hat OpenShift Logging のリリースノート 」を参照してください。

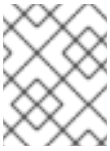
### 1.4.16. モニタリング

#### 1.4.16.1. アラートルールの変更

OpenShift Container Platform 4.8 には、以下のアラートルールの変更が含まれます。

例1.1 アラートルールの変更

- **ThanosSidecarPrometheusDown**アラートの重大度が、重大 から 警告 に更新されました。
- **ThanosSidecarUnhealthy** アラートの重大度が、重大 から 警告 に更新されました。
- **ThanosQueryHttpRequestQueryErrorRateHigh** アラートの重大度が、重大 から 警告 に更新されました。
- **ThanosQueryHttpRequestQueryRangeErrorRateHigh** アラートの重大度が、重大 から 警告 に更新されました。
- **ThanosQueryInstantLatencyHigh** の重大なアラートが削除されました。このアラートは、Thanos Querier のインスタントクエリーのレイテンシーが高い場合に発生しました。
- **ThanosQueryRangeLatencyHigh** の重大なアラートが削除されました。このアラートは、Thanos Querier のレンジクエリーのレイテンシーが高い場合に発生しました。
- すべての Thanos Querier アラートについては、**for** の期間が増え、1時間となりました。
- すべての Thanos サイドカーアラートについては、**for** の期間が増え、1時間となりました。



**注記**

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

#### 1.4.16.2. 今後のリリースで削除される API が使用される際のアラートおよび情報

OpenShift Container Platform 4.8 では、次のリリースで削除される API が使用中の場合に実行される 2 つの新規アラートが導入されました。

- **APIRemovedInNextReleaseInUse**: OpenShift Container Platform の次のリリースで削除される API の場合
- **APIRemovedInNextEUSReleaseInUse**: 次の OpenShift Container Platform Extended Update Support (EUS) リリースで削除される API の場合

新しい **APIRequestCount** API を使用して、非推奨の API を使用する内容を追跡することができます。これにより、次のリリースにアップグレードするためにアクションが必要であるかどうかについて計画できます。

#### 1.4.16.3. モニタリングスタックコンポーネントおよび依存関係のバージョン更新

OpenShift Container Platform 4.8 には、以下のモニタリングスタックコンポーネントおよび依存関係に対するバージョンの更新が含まれます。

- Prometheus Operator: バージョン 0.48.1
- Prometheus: バージョン 2.26.1
- **node-exporter** エージェント: バージョン 1.1.2
- Thanos: バージョン 0.20.2
- Grafana: バージョン 7.5.5

#### 1.4.16.4. kube-state-metrics はバージョン 2.0.0 にアップグレード

**kube-state-metrics** はバージョン 2.0.0 にアップグレードされました。以下のメトリクスは **kube-state-metrics** バージョン 1.9 で非推奨となり、バージョン 2.0.0 で事実上削除されます。

- Pod の非汎用リソースメトリクス:
  - kube_pod_container_resource_requests_cpu_cores
  - kube_pod_container_resource_limits_cpu_cores
  - kube_pod_container_resource_requests_memory_bytes
  - kube_pod_container_resource_limits_memory_bytes
- ノードの非汎用リソースメトリクス:
  - kube_node_status_capacity_pods
  - kube_node_status_capacity_cpu_cores

- kube_node_status_capacity_memory_bytes
- kube_node_status_allocatable_pods
- kube_node_status_allocatable_cpu_cores
- kube_node_status_allocatable_memory_bytes

#### 1.4.16.5. Grafana および Alertmanager UI リンクの削除

サードパーティーの Alertmanager UI へのリンクは、OpenShift Container Platform Web コンソールの **Monitoring → Alerting** ページから削除されます。また、サードパーティーの Grafana UI へのリンクは、**Monitoring → Dashboards** ページから削除されます。**openshift-monitoring** プロジェクトの **Networking → Routes** ページに移動して、**Administrator** パースペクティブの Web コンソールで Grafana UI および Alertmanager UI へのルートに引き続きアクセスできます。

#### 1.4.16.6. Web コンソールでのダッシュボードの機能拡張のモニタリング

OpenShift Container Platform Web コンソールの **Monitoring → Dashboards** ページで新しい拡張機能を利用できます。

- マウスで領域を選択して単一のグラフを拡大すると、他のすべてのグラフが更新され、同じ時間範囲が反映されるようになりました。
- ダッシュボードパネルはグループに分類され、展開したり折りたたんだりできるようになりました。
- 単一値パネルは、値に応じて色を変更できるようになりました。
- ダッシュボードラベルが **Dashboard** のドロップダウンリストに表示されるようになりました。
- **Time Range** ドロップダウンリストで **Custom time range** を選択して、ダッシュボードのカスタム時間範囲を指定できるようになりました。
- 該当する場合は、ダッシュボードフィルタードロップダウンメニューで **All** オプションを選択して、そのフィルター内のすべてのオプションのデータを表示できるようになりました。

### 1.4.17. メータリング

メータリング Operator は OpenShift Container Platform 4.6 で非推奨となり、OpenShift Container Platform の次のリリースで削除される予定です。

### 1.4.18. スケーリング

#### 1.4.18.1. 単一ノードクラスターでの実行

単一ノードクラスターでテストを実行すると、SR-IOV および SCTP テストを含む特定のテストのタイムアウトが長くなり、コントロールプレーンとワーカーノードを必要とするテストは省略されます。ノードの再起動が必要な再設定では、OpenShift コントロールプレーンを含む環境全体が再起動されるため、完了するまでに時間がかかります。コントロールプレーンおよびワーカーノードを必要とするすべての PTP テストは省略されます。テストは起動時にノード数をチェックし、それに応じてテスト動作を調整するため、追加の設定は必要ありません。

PTP テストが検出モードで実行できます。このテストは、クラスター外に設定された PTP マスターを検索します。以下のパラメーターが必要です。

- **ROLE_WORKER_CNF=master**: コントロールプレーン (**master**) は、ノードが所属する唯一のマシンプールであるため必須です。
- **XT_U32TEST_HAS_NON_CNF_WORKERS=false**: モジュールが読み込まれるノードのみが存在するため、**xt_u32** の障害テストを省略するように指示する必要があります。
- **SCTPTEST_HAS_NON_CNF_WORKERS=false**: モジュールが読み込まれるノードのみがあるため、SCTP の障害テストを省略するように指示する必要があります。

#### 1.4.18.2. Performance Addon Operator を使用した NIC の削減

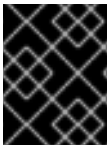
Performance Addon Operator を使用すると、パフォーマンスプロファイルを設定して、各ネットワークデバイスの Network Interface Card (NIC) キュー数を調整できます。デバイスネットワークキューを使用すると、パケットを複数の異なる物理キューに分散でき、各キューはパケット処理用に個別のスレッドを取得します。

Data Plane Development Kit (DPDK) ベースのワークロードの場合、NIC キューを予約済みまたはハウスキーピング CPU の数に制限して、必要な低レイテンシーを実現するために NIC キューを減らすことが重要です。

詳細は、Performance Addon Operator を使用した NIC キューの削減について参照してください。

#### 1.4.18.3. クラスターの最大数

OpenShift Container Platform 4.8 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。



**重要**

本リリースでは、OVN-Kubernetes テストに対してパフォーマンスの大規模なスケーリングテストは実行されません。

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。

#### 1.4.18.4. パフォーマンスプロファイルの作成

Performance Profile Creator (PPC) ツールを使用してパフォーマンスプロファイルを作成できるようになりました。このツールは、クラスターから **must-gather** データおよびいくつかのユーザー指定のプロファイル引数を消費し、この情報を使用して、ハードウェアおよびトポロジーに適したパフォーマンスプロファイルを生成します。

詳細は、コントロールプレーンプロファイルの作成について参照してください。

#### 1.4.18.5. Node Feature Discovery Operator

Node Feature Discovery (NFD) Operator が利用可能になりました。これを使用して、ハードウェア機能およびシステム設定を検出する Node Feature Discovery という Kubernetes アドオンをオーケストレーションしてノードレベルの情報を公開します。

#### 1.4.18.6. ドライバツールキット (テクノロジープレビュー)

Driver Toolkit をドライバーコンテナーのベースイメージとして使用し、Kubernetes で特別なソフトウェアおよびハードウェアデバイスを有効化できるようになりました。これは現在、テクノロジープレビュー機能です。

## 1.4.19. バックアップおよび復元

### 1.4.19.1. etcd スナップショットの強化

新規の拡張機能は、バックアップ後および復元前の etcd スナップショットのステータスを検証します。以前のバージョンでは、バックアッププロセスは、取得されたスナップショットが完了したことを検証しませんでした。また、復元プロセスは、復元されるスナップショットが有効であり、破損していないことを検証しませんでした。バックアップまたは復元中にディスクが破損している場合は、エラーが管理者に明確に報告されるようになりました。詳細は、BZ#1965024 を参照してください。

## 1.4.20. Insights Operator

### 1.4.20.1. ネットワークが制限された環境に関する Insights Advisor の推奨事項

OpenShift Container Platform 4.8 では、ネットワークが制限された環境で操作しているユーザーは、Insights Operator アーカイブを Insights Advisor に収集し、アップロードして潜在的な問題を診断できます。さらに、ユーザーはアップロード前に、Insights Operator アーカイブに含まれる機密データを難読化できます。

詳細は、「限定的なネットワーク環境でのリモートヘルスレポートの使用」を参照してください。

### 1.4.20.2. Insights Advisor の改善

OpenShift Container Platform Web コンソールの Insights Advisor は、0 の問題が検出されたと正しく報告するようになりました。以前のバージョンでは、Insights Advisor は情報を提供していませんでした。

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

OpenShift Container Platform 4.8 では、Insights Operator は以下の追加情報を収集します。

- 既知のセキュリティー問題とバージョンの問題を見つけるための識別できないクラスターワークロード情報。
- **MachineHealthCheck** および **MachineAutoscaler** の定義。
- **virt_platform** および **vsphere_node_hw_version_total** のメトリクス。
- SAP Smart Data Integration のインストールを支援するための正常でない SAP Pod に関する情報。
- SAPクラスターを識別するための **datahubs.installers.datahub.sap.com** リソース。
- ネットワークを強化するための失敗した **PodNetworkConnectivityChecks** の概要。
- **cluster-version** Operator の問題をデバッグするための **openshift-cluster-operator** namespace からの **cluster-version** Pod およびイベントに関する情報。

この追加情報により、Red Hat は Insights Advisor の改善された修復手順を提供できます。

### 1.4.20.4. 正常でない SAP Pod の Insights Operator の拡張機能

Insights Operator は正常でない SAP Pod についてのデータを収集できるようになりました。SDI インストールが失敗すると、どの初期化 Pod が失敗したかを確認して問題を検出できます。Insights Operator は、SAP/SDI namespace で失敗した Pod の情報を収集できるようになりました。詳細

は、BZ#1930393 を参照してください。

#### 1.4.20.5. SAP Pod データを収集するための Insights Operator の拡張機能

Insights Operator は、SAP クラスターから **Datahubs** リソースを収集できるようになりました。このデータにより、SAP クラスターは Insights Operator アーカイブの非 SAP クラスターと区別できます。これは、SAP クラスターのみから収集されたすべてのデータが欠落しており、クラスターに SDI インストールがあるかどうかを判別することができない場合でも同様です。詳細は、BZ#1940432 を参照してください。

### 1.4.21. 認証および認可

#### 1.4.21.1. 認証情報の AWS Security Token Service (STS) を使用した OpenShift Container Platform の実行が一般に利用可能になる

Cloud Credential Operator (CCO) ユーティリティー (**ccoctl**) を、Amazon Web Services Security Token Service (AWS STS) を使用するように設定できるようになりました。CCO が STS を使用するように設定されている場合、短期的で権限に制限のあるセキュリティー認証情報を提供する IAM ロールをコンポーネントに割り当てます。

この機能は以前は OpenShift Container Platform 4.7 のテクノロジープレビュー機能として導入されましたが、OpenShift Container Platform 4.8 では一般に利用可能となりました。

詳細は、STS での手動モードの使用 を参照してください。

### 1.4.22. OpenShift サンドボックスコンテナー

#### 1.4.22.1. OpenShift Container Platform での OpenShift サンドボックスコンテナーのサポート (テクノロジープレビュー)

OpenShift サンドボックスコンテナー 1.0.0 テクノロジープレビューリリースでは、追加のランタイムとして Kata コンテナーを実行するための組み込みサポートが導入されています。OpenShift サンドボックスコンテナーを使用すると、ユーザーは追加のランタイムとして Kata コンテナーを選択して、ワークロードをさらに分離できます。OpenShift サンドボックスコンテナー Operator は、Kata コンテナーのインストール、削除、および更新のタスクを自動化します。**KataConfig** カスタムリソースを記述することにより、これらのタスクの状態を追跡できます。

OpenShift サンドボックスコンテナーは、ベアメタルでのみサポートされます。Red Hat Enterprise Linux CoreOS (RHCOS) は、OpenShift サンドボックスコンテナー 1.0.0 で唯一サポートされているオペレーティングシステムです。非接続環境は、OpenShift Container Platform 4.8ではサポートされていません。

詳細は、「OpenShift サンドボックスコンテナーについて」を参照してください。

## 1.5. 主な技術上の変更点

OpenShift Container Platform 4.8 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。

**Kuryr サービスサブネットの作成の変更**
Kuryr を使用するように設定された Open Virtual Network を使用したRed Hat OpenStack Platform (RHOSP) への OpenShift Container Platform の新規インストールでは、**networking.serviceCIDR** で要求されたサイズの2倍の **services** サブネットが作成されなくなりました。作成されたサブネットは、要求されたサイズと同じになります。詳細は、BZ#1955548 を参照してください。

**SHA-256 プレフィックスのない OAuth トークンが使用不可になる**
OpenShift Container Platform 4.6 よりも前のバージョンでは、OAuth のアクセスおよび認証トークンはオブジェクト名のシークレット情報を使用していました。

OpenShift Container Platform 4.6 以降、OAuth アクセストークンおよび認証トークンオブジェクト名は、SHA-256 プレフィックスを持つ機密ではないオブジェクト名として保存されます。OpenShift Container Platform 4.8 では、SHA-256 プレフィックスが含まれない OAuth トークンは使用されず、作成できなくなります。

**Federal Risk and Authorization Management Program (FedRAMP) モードレート制御**
OpenShift Container Platform 4.8 では、**rhcos4-moderate** プロファイルが完了しています。**ocp4-moderate** プロファイルは今後のリリースで完了されます。

**Ingress コントローラーを HAProxy 2.2.13 にアップグレード**
OpenShift Container Platform Ingress コントローラーが HAProxy version 2.2.13 にアップグレードされます。

**CoreDNS がバージョン 1.8.1 に更新される**
OpenShift Container Platform 4.8 では、CoreDNS はバージョン 1.8.1 を使用します。これには、バグ修正、名前が変更されたメトリクス、およびデュアルスタック IPv6 対応が含まれます。

**etcd が zap ロガーを使用する**
OpenShift Container Platform 4.8 では、etcd は capnslog ではなく、デフォルトのロガーとして zap を使用するようになりました。zap は、マシンで消費可能な JSON ログメッセージを提供する構造化ロガーです。**jq** を使用して、これらのログメッセージを簡単に解析することができます。

capnslog 形式の使用が予想されるログコンシューマーがある場合、zap ロガー形式に合わせてこれを調整する必要がある場合があります。

**サンプル capnslog 形式 (OpenShift Container Platform 4.7)**

```
2021-06-03 22:40:16.984470 W | etcdserver: read-only range request
"key:\"/kubernetes.io/operator.openshift.io/clustercsidrivers/\"
range_end:\"/kubernetes.io/operator.openshift.io/clustercsidrivers0\" count_only:true " with result
"range_response_count:0 size:8" took too long (100.498102ms) to execute
```

**サンプル zap 形式 (OpenShift Container Platform 4.8)**

```
{"level":"warn","ts":"2021-06-14T13:13:23.243Z","caller":"etcdserver/util.go:163","msg":"apply request
took too long","took":"163.262994ms","expected-duration":"100ms","prefix":"read-only range
","request":"key:\"/kubernetes.io/namespaces/default\" serializable:true keys_only:true
","response":"range_response_count:1 size:53"}
```

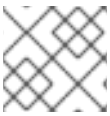
**LSO 用にマージされた複数のデーモンセット**
OpenShift Container Platform 4.8 では、複数のデーモンセットがローカルストレージオブジェクト (LSO) に対してマージされます。ローカルボリュームカスタムリソースを作成すると、**daemonset.apps/diskmaker-manager** のみが作成されます。

**バインドされたサービスアカウントトークンボリュームが有効化されている**
以前のバージョンでは、サービスアカウントトークンは Pod にマウントされたシークレットでした。OpenShift Container Platform 4.8 以降では、代わりに Projected ボリュームを使用します。この変更により、サービスアカウントトークンには基礎となる対応するシークレットが含まれなくなりました。

バインドされたサービスアカウントトークンは、オーディエンスバインドとタイムバインドです。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。

さらに、kubelet はトークンが期間の 80 パーセントに達するとトークンを自動的に更新し、**client-go** はトークンの変更を監視して自動的にリロードします。これら 2 つの動作の組み合わせは、バインドされたトークンのほとんどの使用法が、期限切れにならないレガシートークンの使用法と変わらないことを意味します。**client-go** 以外での非標準的な使用は、問題を引き起こす可能性があります。

**Operator SDK v1.8.0**
OpenShift Container Platform 4.8 は Operator SDK v1.8.0 をサポートします。この最新バージョンにインストールまたは更新するには、「Operator SDK CLI のインストール」を参照してください。



**注記**

Operator SDK v1.8.0 は Kubernetes 1.20 をサポートします。

以前に Operator SDK v1.3.0 で作成または保守された Operator プロジェクトがある場合は、「新しい Operator SDK バージョンのプロジェクトのアップグレード」を参照してプロジェクトをアップグレードし、Operator SDK v1.8.0 との互換性が維持されていることを確認してください。

## 1.6. 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 4.8 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
- **TP**: テクノロジープレビュー
- **DEP**: 非推奨機能
- **REM**: 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 4.6 | OCP 4.7 | OCP 4.8 |
|---|---|---|---|
| **OperatorSource** オブジェクト | REM | REM | REM |
| Package Manifest Format (Operator Framework) | DEP | DEP | REM |
| **oc adm catalog build** | DEP | DEP | REM |
| **oc adm catalog mirror** の **--filter-by-os** フラグ | GA | DEP | REM |
| v1beta1 CRD | DEP | DEP | DEP |
| Docker Registry v1 API | DEP | DEP | DEP |

| 機能 | OCP 4.6 | OCP 4.7 | OCP 4.8 |
|---|---|---|---|
| メータリング Operator | DEP | DEP | DEP |
| スケジューラーポリシー | GA | DEP | DEP |
| Cluster Samples Operator の **ImageChangesInProgress** 状態 | GA | DEP | DEP |
| Cluster Samples Operator の **MigrationInProgress** 状態 | GA | DEP | DEP |
| OpenShift Container Platform リソースの **apiVersion** での **v1** の使用 | GA | DEP | DEP |
| Red Hat Enterprise Linux CoreOS (RHCOS) での **dhclient** の使用 | DEP | DEP | DEP |
| クラスターローダー | GA | GA | DEP |
| 独自の RHEL 7 コンピュートマシンの持ち込み | DEP | DEP | DEP |
| AWS EFS の外部プロビジョナー | REM | REM | REM |
| ビルドの **BuildConfig** 仕様の **lastTriggeredImageID** フィールド | GA | GA | DEP |
| Jenkins Operator | TP | TP | DEP |
| Prometheus に基づく HPA カスタムメトリクスアダプター | TP | TP | REM |
| Red Hat Virtualization (RHV) の**instance_type_id**インストール設定パラメーター | GA | DEP | DEP |

### 1.6.1. 非推奨の機能

#### 1.6.1.1. Descheduler operator.openshift.io/v1beta1 API グループが非推奨になる

Descheduler の **operator.openshift.io/v1beta1** API グループは非推奨となり、今後のリリースで削除される可能性があります。代わりに **operator.openshift.io/v1** API グループを使用します。

#### 1.6.1.2. Red Hat Enterprise Linux CoreOS (RHCOS) での dhclient の使用が非推奨になる

OpenShift Container Platform 4.6 以降、Red Hat Enterprise Linux CoreOS (RHCOS) は **initramfs** で **NetworkManager** を使用し、初回の起動時にネットワークを設定するようになりました。この変更の一環として、DHCP の **dhclient** バイナリーの使用が非推奨になりました。その代わり

に、**NetworkManager** の内部 DHCP クライアントをネットワーク設定に使用します。**dhclient** バイナリーは、今後のリリースで Red Hat Enterprise Linux CoreOS (RHCOS) から削除されます。詳細は、BZ#1908462 を参照してください。

#### 1.6.1.3. クラスターローダーが非推奨になる

クラスターローダーが非推奨になり、今後のリリースで削除されます。

#### 1.6.1.4. ビルドの lastTriggeredImageID パラメーターは非推奨となりました。

このリリースでは、**BuildConfig** 仕様で設定できる **BuildTriggerPolicy** タイプの 1 つである **ImageChangeTrigger** オブジェクトの **lastTriggeredImageID** が非推奨となりました。

OpenShift Container Platform の次のリリースでは、**lastTriggeredImageID** のサポートが削除され、これを無視します。次に、イメージ変更トリガーは、**BuildConfig** 仕様の **lastTriggeredImageID** フィールドへの変更に基づいたビルドは開始しません。その代わりに、ビルドをトリガーするイメージ ID は **BuildConfig** オブジェクトのステータスに記録されます。これは、ほとんどのユーザーが変更することはできません。

したがって、**buildConfig.spec.triggers[i].imageChange.lastTriggeredImageID** を検査するスクリプトとジョブを適切に更新します。(BUILD-213)

#### 1.6.1.5. Jenkins Operator (テクノロジープレビュー) が非推奨となりました。

このリリースでは、テクノロジープレビュー機能であった Jenkins Operator が非推奨となりました。OpenShift Container Platform の今後のバージョンでは、OpenShift Container Platform Web コンソールインターフェースの OperatorHub から Jenkins Operator を削除します。その後、Jenkins Operator のアップグレードは利用できなくなり、Operator はサポートされなくなります。

お客様は、Samples Operator によって提供されるテンプレートを使用して引き続き OpenShift Container Platform に Jenkins をデプロイすることができます。

#### 1.6.1.6. Red Hat Virtualization (RHV) のinstance_type_idインストール設定パラメーター

**instance_type_id**インストール設定パラメータは非推奨になり、今後のリリースで削除される予定です。

### 1.6.2. 削除された機能

#### AWS EFS (テクノロジープレビュー) 機能の外部プロビジョナーが削除される

Amazon Web Services (AWS) Elastic File System (EFS) テクノロジープレビュー機能が削除され、サポートされなくなりました。

#### 1.6.2.1. サンプルイメージストリームから削除されたイメージ

以下のイメージは、OpenShift Container Platform で提供されるサンプルイメージストリームに含まれなくなりました。

```
registry.redhat.io/rhscl/nodejs-10-rhel7
registry.redhat.io/ubi7/nodejs-10
registry.redhat.io/rhscl/perl-526-rhel7
registry.redhat.io/rhscl/postgresql-10-rhel7
registry.redhat.io/rhscl/ruby-25-rhel7
```

```
registry.redhat.io/ubi7/ruby-25
registry.redhat.io/rhdm-7/rhdm-decisioncentral-rhel8:7.9.0
registry.redhat.io/rhdm-7/rhdm-kieserver-rhel8:7.9.0
registry.redhat.io/rhpam-7/rhpam-businesscentral-monitoring-rhel8:7.9.0
registry.redhat.io/rhpam-7/rhpam-businesscentral-rhel8:7.9.0
registry.redhat.io/rhpam-7/rhpam-smartrouter-rhel8:7.9.0
```

#### 1.6.2.2. Operator のパッケージマニフェスト形式へのサポートの削除

Operator のレガシーパッケージマニフェスト形式のサポートは、OpenShift Container Platform 4.8 以降で削除されます。このサポートの削除には、レガシー形式で構築されたカスタムカタログと、OperatorSDK を使用してレガシー形式で最初に作成された Operator プロジェクトが含まれます。バンドル形式は、OpenShift Container Platform 4.6 以降の Operator Lifecycle Manager (OLM) の推奨 Operator パッケージ形式です。

バンドル形式の使用についての詳細は、「カスタムカタログの管理」および「パッケージマニフェストプロジェクトのバンドル形式への移行」を参照してください。

さらに、形式に関連する以下のコマンドが OpenShift CLI (**oc**) および Operator SDK CLI から削除されました。

- **oc adm catalog build**
- **operator-sdk generate packagemanifest**
- **operator-sdk run packagemanifest**

#### 1.6.2.3. Prometheus に基づく HPA カスタムメトリクスアダプターのサポート

本リリースでは、テクノロジープレビューであった Prometheus アダプターが削除されます。

#### 1.6.2.4. セキュアなトークンストレージアノテーション認識が削除される

**authentication** および **openshift-apiserver** Operatorは、クラスターの監査ポリシーを選択する際に **oauth-apiserver.openshift.io/secure-token-storage** アノテーションを無視するようになりました。監査ポリシーは、デフォルトで **secure-** を使用するようになりました。詳細は、BZ#1879182 を参照してください。

## 1.7. バグ修正

assisted-installer

- 以前のバージョンでは、**assisted-service** コンテナーは **postgres** が起動し、接続を受け入れる準備ができるのを待ちませんでした。**assisted-service** コンテナーはデータベース接続を確立しようとして失敗し、そして **assisted-service** コンテナーが失敗して再起動していました。この問題は、**assisted-service** コンテナーが、最大 10 秒間データベースに接続しようとすることで修正されました。**postgres** が起動して 10 秒以内に接続を受け入れる準備ができると、**assisted-service** コンテナーはエラー状態に進むことなく接続します。**assisted-service** コンテナーが 10 秒以内に **postgres** に接続できない場合、エラー状態になり、再起動し、再度試みます。(BZ#1941859)

ベアメタルハードウェアのプロビジョニング

- 以前のバージョンでは、Ironic はデフォルトで HTTPS を使用し、正しい証明書バンドルを利用できなかったため、インストール用のイメージのダウンロードに失敗していました。この問題

は、イメージのダウンロードを **Insecure** に設定して、証明書なしで転送を要求することで修正されています。(BZ#1953795)

- 以前のバージョンでは、デュアルスタックネットワークを使用する場合、ワーカーノードのホスト名は、デプロイメントの前に Ironic が検査するホスト名と一致しないことがありました。そのため、ノードを手動で承認する必要がありました。これは修正されています。(BZ#1955114)

- 以前のバージョンでは、UEFI モードでは、RHCOS イメージのダウンロード後に **ironic-python-agent** が UEFI ブートローダーエントリーを作成していました。RHEL 8.4 に基づく RHCOS イメージを使用する場合、このエントリーを使用してイメージを起動できない可能性があります。イメージの起動時に Ironic によりインストールされたエントリーが使用された場合、起動に失敗し、BIOS エラー画面が出力される可能性があります。これは、固定のブートエントリーを使用する代わりに、イメージにある CSV ファイルに基づいてブートエントリーを設定する **ironic-python-agent** により修正されています。イメージはエラーなしで正常に起動します。(BZ#1972213)

- 以前のバージョンでは、ノードは、起動時に誤った IP バージョンを選択することがありました (IPv4 ではなく IPv6 を選択したり、またはその逆の場合もあり)。ノードは IP アドレスを受信しなかったために起動できませんでした。これは、IP オプションを downloader (**ip=dhcp** or **ip=dhcp6**) に渡す Cluster Bare Metal Operator によって修正され、これにより、起動時に正しく設定され、ノードは予想どおりに起動します。(BZ#1946079)

- 以前のバージョンでは、Ironic のイメージキャッシュメカニズムが無効化され、virtualmedial iso をホストする HTTP サーバーへの直接接続を有効化し、ローカルストレージの問題を阻止していました。標準に準拠しない HTTP クライアントおよび redfish 実装により、BMC 接続でで障害が発生しました。これは、virtualmedia iso がキャッシュされ、Ironic conductor ノードから提供されるデフォルトの Ironic の動作に戻すことで修正されています。標準に準拠しない HTTP クライアントおよび redfish 実装によって生じる問題が修正されました。(BZ#1962905)

- 以前のバージョンでは、マシンインスタンスの **state** アノテーションは設定されていませんでした。その結果、**STATE** 列は空でした。今回の更新により、マシンインスタンスの **state** アノテーションが設定され、**STATE** 列の情報が自動的に入力されるようになりました。(BZ#1857008)

- 新しい ipmitool パッケージはデフォルトで暗号化スイート 17 を使用するため、暗号化スイート 17 をサポートしない古いハードウェアはデプロイメント中に失敗します。暗号化スイート 17 がハードウェアでサポートされない場合は、ipmitool を使用する古いハードウェアで正常にデプロイメントされるように、Ironic は暗号化スイート 3 を使用するようになりました。(BZ#1897415)

- 以前のバージョンでは、イメージキャッシュの設定前に導入が実行されたため、永続的な導入に失敗し、再試行されることはありませんでした。これにより、コントロールプレーンのベアメタルホストが **adoption failed** を報告しました。この更新により、外部でプロビジョニングされるホストの導入は、導入の失敗後にコントロールプレーンホストが正しく導入されるまで自動的に再試行されます。(BZ#1905577)

- 以前のバージョンでは、カスタムリソース (CR) には、ベースボード管理コントローラー (BMC) の詳細が必要でした。ただし、アシスト付きインストーラーの場合は、この情報が提供されませんでした。この更新により、Operator がノードを作成していないときに、CR が BMC の詳細をバイパスできるようになります。(BZ#1913112)

- イメージをノードにプロビジョニングする場合、qemu-image は 1G の RAM に制限されていたため、qemu-img がクラッシュする可能性がありました。この修正により、制限が 2G に引き上げられ、qemu-img がプロビジョニングを確実に完了するようになりました。(BZ#1917482)

- redfish/v1/SessionService URL には認証が必要なため、Ironic はサイトにアクセスする際に認証エラーを生成します。Ironic がこのエラーメッセージを報告した際、機能上の問題はなかったため、削除されました。(BZ#1924816)

- 一部のドライブでは、**/dev/sda1** などのパーティションには読み取り専用ファイルがありませんでした。ただし、**/dev/sda** などのベースデバイスにはこのファイルがあります。そのため、Ironic はパーティションが読み取り専用であることを判断できず、そのドライブでメタデータのクリーニングが失敗する可能性がありました。この更新により、パーティションが読み取り専用として検出され、ベースデバイスの追加チェックが含まれるようになります。その結果、メタデータのクリーニングは読み取り専用パーティションで実行されず、メタデータのクリーニングが失敗しなくなりました。(BZ#1935419)

- プロキシーが設定された状態で Baremetal IPI がデプロイされると、内部の machine-os イメージのダウンロードはプロキシーを介して送信されました。これによりイメージが破損し、ダウンロードできなくなりました。この更新により、内部イメージトラフィックが **no_proxy** に修正され、イメージのダウンロードでプロキシーが使用されなくなります。(BZ#1962592)

- 以前のバージョンでは、Ironic と RAM ディスク間の大規模なパケット転送によって接続障害が発生した場合、ベアメタルのデプロイメントは失敗していました。今回の更新により、Ironic は、接続エラーを回避するための情報を RAM ディスクにクエリーし、デプロイメントを正常に実行できるようになりました。(BZ#1957976)

ビルド

- 以前のバージョンでは、CVE-2021-3344 が修正された後、ビルドは OpenShift Container Platform ノードにエンタイトルメントキーを自動的にマウントしませんでした。その結果、エンタイトルメント証明書がホストまたはノードに保存されていた場合は、修正により、エンタイトルメントのあるビルドがシームレスに機能しなくなりました。ホストまたはノードに保存されているエンタイトルメント証明書の取り込みの失敗は、4.7.z では BZ#1945692 で修正され、4.6.z では BZ#1946363 で修正されました。ただし、これらの修正では、Red Hat Enterprise Linux CoreOS (RHCOS) ワーカーノードで実行されているビルドに無害な警告メッセージが導入されました。現在のリリースでは、ビルドが RHEL ワーカーノードにのみエンタイトルメントを自動的にマウントできるようにし、RHCOS ワーカーノードでのマウント試行を回避することで、この問題を修正しています。これで、RHCOS ノードでビルドを実行するときに、エンタイトルメントのマウントに関する無害な警告メッセージが表示されなくなります。(BZ#1951084)

- Docker Hub からイメージをプルする一部のユーザーには、以下のエラーが発生する可能性があります。

  > container image registry lookup failed...toomanyrequests: You have reached your pull rate limit

  このエラーは、**oc new-app** の呼び出しに使用した **docker.io** ログインに **docker.io** の有料サポートが十分にないために発生します。結果として得られるアプリケーションは、イメージプルスロットリングの対象となり、障害が発生する可能性があります。現在のリリースでは、**oc new-app** ヘルプが更新され、イメージレジストリーおよびリポジトリー仕様でデフォルトがどのように機能するかをユーザーに通知します。これによりユーザーは、可能な場合、デフォルト以外のイメージ参照を使用して、同様のエラーを回避できます。(BZ#1928850)

- 以前のバージョンでは、ビルドはイメージのプッシュが失敗したかどうかを確認するためのエラーチェックを実行しませんでした。その結果、ビルドは常に **Successfully pushed** のメッセージをログに記録します。現在、ビルドはエラーが発生したかどうかをチェックし、イメージのプッシュが成功した後にのみ、**Successfully pushed** のメッセージをログに記録します。(BZ#1947164)

- 以前のバージョンでは、ドキュメントおよび **oc explain** ヘルプテキストは、**BuildConfig** オブ

ジェクト内の **buildArgs** フィールドが、その基礎となる Kubernetes **EnvVar** タイプの **valueFrom** フィールドをサポートしていないことを伝えていませんでした。その結果、ユーザーはそれがサポートされていると信じて使用しようとしました。現在のリリースでは、ドキュメントとヘルプテキストが更新されているため、**BuildConfig** オブジェクトの **buildArgs** フィールドが **valueFrom** フィールドをサポートしていないことがより明確になっています。(BZ#1956826)

- ビルドがベースイメージのプルなどのイメージレジストリーと対話する場合、断続的な通信の問題によりビルドが失敗する可能性があります。現在のリリースでは、これらの対話に対する再試行回数が増えています。現在の OpenShift Container Platform ビルドは、イメージレジストリーとの断続的な通信の問題が発生した場合、以前よりも回復力があります。(BZ#1937535)

クラウドコンピュート

- 以前のバージョンでは、**Cluster Image Registry Operator** は **user_domain_name** を不変フィールドと見なし、インストール後に変更しませんでした。これにより、**user_domain_name** および生成される認証情報への変更を受け入れることが拒否されました。今回の更新により、**user_domain_name** が変更可能としてマークされ、イメージレジストリー設定に保存されなくなりました。これにより、**user_domain_name** およびその他すべての **auth** パラメーターをインストール後に変更できます。(BZ#1937464)

- 以前のバージョンでは、プロキシーの更新により、継続的インテグレーション (CI) の実行中に、API サーバーの再起動を含む完全なクラスター設定の更新が発生していました。その結果、Machine API Operator の一部のクラスターは、予期しない API サーバーの停止が原因でタイムアウトになっていました。この更新により、プロキシーテストが分離され、事後条件が追加されるため、CI の実行中は Machine API Operator のクラスターが再び安定します。(BZ#1913341)

- 以前のバージョンでは、さまざまな vCenter タスクタイプの区別がなかったため、**Insufficient disk space on datastore** 状態のマシンを削除するには、想定外の時間がかかりました。この更新により、マシンコントローラーの削除手順で vCenter タスクタイプがチェックされたため、マシンコントローラーの削除がブロックされなくなりました。その結果、マシンコントローラーはすぐに削除されます。(BZ#1918101)

- 以前のバージョンでは、インスタンスタイプがない場合でも、ゼロアノテーションからのスケーリングが再びキューに入れられていました。その結果、MachineSet コントローラーログに一定の再キューおよびエラースペースメッセージがありました。この更新では、インスタンスタイプが自動的に解決されない場合、ユーザーはアノテーションを手動で設定することができます。その結果、ユーザーが手動でアノテーションを提供した場合、不明なインスタンスタイプのゼロからのスケーリングが機能します。(BZ#1918910)

- 以前のバージョンでは、HTTP 応答は Machine API 終了ハンドラーによって適切に閉じられていませんでした。その結果、goroutine が **net.http** の読み取りおよび書き込みループでリークされ、メモリ使用量が高くなりました。今回の更新で、HTTP 応答が常に正常に閉じられるようになりました。その結果、メモリー使用量が安定するようになりました。(BZ#1934021)

- 以前のバージョンでは、MachineSet コントローラー内で作成された複数のクライアントセットにより起動時間が遅くなり、一部の大規模なクラスターで Pod が readiness チェックに失敗していました。その結果、MachineSet コントローラーは無限ループでスタックします。この更新により、MachineSet コントローラーが修正され、単一のクライアントが使用されるようになります。その結果、MachineSet コントローラーは想定どおりに動作します。(BZ#1934216)

- 以前のバージョンでは、最初の起動時に Machine Config Daemon によってアップグレードが実行されたときに、インスタンスの起動に時間がかかりました。その結果、ワーカーノードが再起動ループでスタックし、ワーカーノードが適切に起動しなかったため、マシンヘルスチェッ

ク (MCH) はワーカーノードを削除しました。この更新により、MHC は正しく開始されていないノードを削除しなくなりました。代わりに、MHC は、明示的に要求された場合にのみノードを削除します。(BZ#1939054)

- 以前のバージョンでは、不明な理由で証明書署名要求 (CSR) の承認が遅れていました。その結果、インストール中にクラスターに表示される新しいマシンはすぐには承認されず、クラスターのインストールに時間がかかりました。インストールの初期段階で API サーバーがたまに使用できなくなることを減らすために、この更新により、キャッシュの再同期期間が 10 時間から 10 分に変更されます。その結果、コントロールプレーンマシンがより迅速に承認されるようになり、クラスターのインストールが長引くことはなくなりました。(BZ#1940972)

- 以前のバージョンでは、デフォルトの Google Cloud Platform (GCP) イメージは古く、新しい Ignition バージョンをサポートしない OpenShift Container Platform 4.6 リリースからのバージョンを参照していました。その結果、デフォルトの GCP イメージを使用するクラスター内の新しいマシンは、OpenShift Container Platform 4.7 以降を起動できませんでした。この更新により、GCP イメージがリリースバージョンと一致するように更新されます。その結果、新しいマシンはデフォルトの GCP イメージで起動できるようになりました。(BZ#1954597)

- 以前のバージョンでは、仮想マシン (VM) の ProvisioningState 値が厳密にチェックされていたため、存在チェック中に仮想マシンが失敗することがありました。この更新により、チェックがより寛容になり、削除されたマシンのみが存在チェック中に **Failed** フェーズに入ります。(BZ#1957349)

- 以前のバージョンでは、AWS クラスターで **oc delete machine** を使用してコントロールプレーンマシンを削除した場合、そのマシンはロードバランサーから削除されていませんでした。その結果、ロードバランサーは、削除されたコントロールプレーンマシンへの要求を引き続き処理しました。この修正により、コントロールプレーンマシンを削除すると、ロードバランサーはマシンへの要求を処理しなくなります。(BZ#1880757)

- 以前のバージョンでは、到達不能なマシンを削除すると、永続ボリューム用に作成され、ノードに接続されている vSphere 仮想マシンディスク (VMDK) が誤って削除されていました。その結果、VMDK 上のデータは回復できませんでした。この修正により、vSphere クラウドプロバイダーは、kubelet に到達できない場合に、これらのディスクをチェックしてノードからデタッチします。その結果、VMDK を失うことなく、到達不能なマシンを削除できます。(BZ#1883993)

- 以前のバージョンでは、生成された AWS インスタンスタイプのリストが古くなっていたため、レプリカがゼロの Cluster Autoscaler とマシンセットを使用すると、一部の新しい Amazon Web Services (AWS) インスタンスタイプをゼロからスケーリングできませんでした。AWS インスタンスタイプの一覧が更新され、新しいインスタンスタイプが含まれるようになりました。この修正により、レプリカがゼロからスケーリングするために、より多くのインスタンスタイプを Cluster Autoscaler Operator で使用できるようになります。(BZ#1896321)

- 以前のバージョンでは、Pod の Disruption Budget では、アップストリームのエビクション API 機能がないため、到達不能なノードで Pod をドレインしませんでした。その結果、到達不能ノード上のマシンは、削除された後に取り除かれるまでに膨大な時間がかかる可能性があります。現在は、到達不能ノード上のマシンを削除するときの猶予期間のタイムアウトが 1 秒に変更されました。この修正により、Machine API は、到達不能なノードを正常にドレインおよび削除できます。(BZ#1905709)

### Cloud Credential Operator

- 以前のバージョンでは、Cloud Credential Operator は、ベアメタルプラットフォームで **unsupported platform type: BareMetal** 警告を繰り返していました。この更新により、ベアメタルプラットフォームは不明のプラットフォームとして扱われなくなりました。その結果、誤解を招くロギングメッセージが削減されます。(BZ#1864116)

- 以前のバージョンでは、Cloud Credential Operator の **credentialsRequest** カスタムリソース (CR) に保存されたエラーメッセージが繰り返し発生するため、CPU 使用率が過度に高くなり、Amazon Web Services (AWS) のレート制限などの一部のエラーシナリオにログインしていました。この更新により、クラウドプロバイダーから返されるリクエスト ID が削除され、エラーメッセージはユーザーが見つけやすい状態で保存され、**credentialsRequest** CR で繰り返し発生するエラーメッセージが削除されます。(BZ#1910396)

- 以前のバージョンでは、CCO デプロイメントが正常ではない場合に、Cloud Credential Operator (CCO) および Cluster Version Operator (CVO) の両方が報告しました。これにより、問題がある場合に 2 つのレポートが作成されました。今回のリリースにより、CCO はデプロイメントが正常ではない場合に報告しなくなりました。(BZ#1957424)

### Cluster Version Operator

- 以前のバージョンでは、Cluster Version Operator は **cluster_operator_up** メトリクスの設定時に **Available** パラメーターおよび **Degraded** パラメーターの両方を評価していました。これにより、アラートに記載された「has not been available」が **Available=True** に一致しなかった場合でも、**ClusterOperatorDown** アラートが **Available=True** または **Degraded=True** の Operator に対して表示されました。今回の修正により、Cluster Version Operator は **cluster_operator_up** メトリクスの設定時に **Degraded** パラメーターを無視するようになりました。(BZ#1834551)

- 以前のバージョンでは、Prometheus がクラスターにインストールされている場合、重要なプラットフォームトポロジーメトリクスは使用できず、呼び出し元で生成されたインストーラーメトリクスが "" に設定されていると、CI エラーが発生していました。エラーの原因となっているメトリクスが提供される前にインフォーマーが同期されなかった潜在的な競合状態が修正されました。(BZ#1871303)

- 以前のバージョンでは、Cluster Version Operator の独自デプロイメントなど、同じキーの複数の容認を持つマニフェストは、最後に読み取られたエントリーのみを受け入れ、前のエントリーを上書きしていました。これにより、**in-cluster tolerations** がマニフェストに記載されている容認から分離されました。この更新により、Cluster Version Operator は、容認が完全に等しい場合に、容認が一致すると見なすようになりました。これにより、Cluster Version Operator は **in-cluster resource** のマニフェストに存在するすべての容認を保持することができます。(BZ#1941901)

- 以前のバージョンでは、Cluster Version Operator は、**env** および **envFrom** を設定しなかったマニフェストのこれらのプロパティーを調整しませんでした。そのため、Cluster Version Operator はコンテナー環境を適切に管理しませんでした。今回の更新により、Cluster Version Operator が改善され、**env** および **envFrom** がマニフェストで未設定の場合は、これらを消去できるようになりました。これにより、クラスターは、これらのプロパティーに対する **cluster-admin** の無効な変更から自動的に回復できます。(BZ#1951339)

- 以前のバージョンでは、**cluster-version-operator** デプロイメントオブジェクトなど、同じキーの複数の容認を持つマニフェストは、最後に読み取られたエントリーのみを受け入れ、前のエントリーを上書きしていました。これにより、クラスター内の容認がマニフェストに記載されている容認から分離されました。この更新により、Cluster Version Operator は、容認が等しい場合に、容認が一致すると見なすようになりました。これにより、Cluster Version Operator は、クラスター内リソースのマニフェストに存在するすべての容認を保持することができます。(BZ#1941901)

- 以前のバージョンでは、Cluster Version Operator は、**ClusterOperator** リソースのパフォーマンスが 10 分間低下すると、**ClusterOperatorDegraded** アラートを報告していました。このアラートは、リソースがまだ作成中だったため、インストールの途中で発生したことがありました。今回の更新により、期間が 10 分から 30 分に変更され、**ClusterOperatorDegraded** アラートが途中で発生することなく、十分な時間を提供することで、インストールが進行するようになりました。(BZ#1957991)

コンプライアンス Operator

- 以前のバージョンでは、ユーザーがコンプライアンスチェックを実行し、**NON-COMPLIANT** の結果が表示された場合に、ユーザーが対応するために必要な修正手順は示されていませんでした。このリリースでは、ユーザーがルールの検証に必要な手順を確認できるように **instructions** キーが提供されています。これにより、ユーザーおよび監査担当者は Operator が正しい値をチェックしていることを確認できます。(BZ#1919367)

コンソール Kubevirt プラグイン

- 以前のバージョンでは、ユーザーが仮想化テンプレートにブートソースを追加する際に役立つ Web コンソールフォームでは、テンプレートが使用するオペレーティングシステムに関係なく、説明テキストは Fedora に対してのみ情報を提供していました。この更新により、テンプレートのオペレーティングシステムに固有の例を提供する修正が追加され、ユーザーに関連するガイダンスを提供します。(BZ#1908169)

- 以前のバージョンでは、ユーザーの仮想マシンテンプレートの作成をサポートする Web コンソールウィザードの説明が曖昧だったことから、操作がテンプレートに適用されるのか、または仮想マシンに適用されるのか、わかりにくい点がありました。今回の修正により説明が明確になり、ユーザーは十分な情報に基づいて決定を下すことができます。(BZ#1922063)

- 以前のバージョンでは、Web コンソールの曖昧なエラーメッセージにより、テンプレートから作成している仮想マシンにネットワークインターフェイスを追加しようとした一部のユーザーに不必要な混乱が生じていました。今回の更新でエラーメッセージに詳細が追加され、ユーザーはより簡単にエラーをトラブルシューティングできるようになりました。(BZ#1924788)

- 以前のバージョンでは、Web コンソールの Red Hat Enterprise Linux (RHEL) 6 テンプレートから仮想マシンを作成しようとすると、RHEL 6 がサポートされていない場合でも、ポップアップウィンドウにサポートレベルの定義方法に関する情報が表示されていました今回の修正により、このウィンドウのテキストが変更され、RHEL 6 はサポートされていないことを明確にしました。(BZ#1926776)

- 以前のバージョンでは、Web コンソールのドロップダウンリストがボタン要素によって隠されていたため、ユーザーは仮想マシンの作成時に特定のオペレーティングシステムを選択できませんでした。今回の修正にはボタン要素の **z-index** 値の調整が含まれ、エラーが修正されたため、ユーザーは使用可能なオペレーティングシステムを選択できるようになります。(BZ#1930015)

- 以前のバージョンでは、ストレージクラスが定義されていないクラスターで Web コンソールの新しい仮想マシンウィザードを使用した場合、Web コンソールが無限ループに陥ってクラッシュしていました。今回の修正により、ストレージクラスが定義されていないインスタンスのストレージクラスのドロップダウンリストが削除されました。その結果、Web コンソールはクラッシュしません。(BZ#1930064)

- 以前のバージョンでは、お気に入りリストから仮想マシンテンプレートを削除するというボタンの機能について、ボタン要素のテキストは明確に説明していませんでした。この修正でテキストが更新され、ボタンの機能が明確になりました。(BZ#1937941)

- 以前のバージョンでは、**RerunOnFailure** 実行ストラテジーを持つ仮想マシンの場合は、仮想マシンを停止すると、複数の UI 要素が応答しなくなり、ユーザーがステータス情報を読み取ったり、仮想マシンを再起動したりできなくなりました。今回の更新で、応答しない要素が修正され、ユーザーがそれらの機能を使用できるようになりました。(BZ#1951209)

- 以前のバージョンでは、別の **/var** パーティションを持つように設定されたクラスターの場合、ファイルシステムにクエリーを実行すると、**/var** パーティションのサイズを除く、root ディレクトリーにマウントされたディスクのサイズのみが返されていました。今回の修正により、ク

エリーの実行方法が変更され、ユーザーはクラスター上のファイルシステムの合計サイズを判断できるようになりました。(BZ#1960612)

コンソールストレージプラグイン

- 以前のバージョンでは、正しいストレージクラスが利用できない場合に OpenShift Container Storage Operator はエラーメッセージを表示していました。今回の更新によりエラーメッセージが削除され、正しいストレージクラスが利用可能になるまで **Next** ボタンが無効になります。(BZ#1924641)

- 以前のバージョンでは、内部接続されたストレージクラスターの作成中にユーザーがブラウザーの「戻る」ボタンをクリックすると、インストールウィザードがプロセスを再開しました。今回の更新でこの問題が修正されています。(BZ#1928008)

- ノードをローカルボリューム検出に追加すると、既存のノードの一覧を表示できるようになりました。これにより、不要なナビゲーションが削減されます。(BZ#1947311)

- 以前のバージョンでは、**Create Storage Cluster** ウィザードを使用すると、未定義の値を持つ Arbiter ゾーンを有効化できました。この更新の修正では、未定義の値がフィルターに掛けられて除外され、Arbiter ゾーンは定義された値でのみ作成できるようになりました。(BZ#1926798)

- 以前のバージョンでは、製品タイトルの書き方や登録商標記号の使用方法に一貫性がなかったため、Web コンソールでクイックスタートカードが誤って表示されていました。今回の更新では、製品名が正しく表記され、登録商標記号は最初のカードでのみ一貫して表示されます。(BZ#1931760)

DNS

- 以前のバージョンでは、BZ#1936587 はグローバル CoreDNS キャッシュの最大 TTL を 900 秒に設定しました。そのため、アップストリームリゾルバーから受信される NXDOMAIN レコードが 900 秒間キャッシュされました。今回の更新により、最大 30 秒間、ネガティブな DNS 応答レコードが明示的にキャッシュされるようになりました。その結果、NXDOMAINs の解決は 900 秒間キャッシュされなくなりました。(BZ#1943578)

- BZ#1953097 の修正は、サイズが 1232 バイトの CoreDNS **bufsize** プラグインを有効化しました。一部のプリミティブ DNS リゾルバーは、512 バイトを超える UDP 経由で DNS 応答メッセージを受信できません。その結果、Go の内部 DNS ライブラリーなどの一部の DNS リゾルバーは、DNS Operator から詳細な DNS 応答を受け取ることができません。今回の更新で、すべてのサーバーで CoreDNS **bufsize** が 512 バイトに設定されました。その結果、UDP DNS メッセージが適切に受信されるようになりました。(BZ#1966116)

- 以前のバージョンでは、クラスターのアップストリームリゾルバーは、UDP 経由で 512 バイトを超える DNS 応答を返しました。そのため、coreDNS は **SERVFAIL** または他のエラーメッセージを返し、クライアントに TCP 上で再試行するように強制しました。今回の更新で、UDP バッファーサイズ 1232 バイトの coreDNS bufisze プラグインが有効になりました。(BZ#1949361)

etcd

- 以前のバージョンでは、etcd Operator でトランスポートリークが発生したため、時間の経過とともにメモリー使用量が増加していました。メモリーリークは修正されました。(BZ#1925586)

- 以前のバージョンでは、**etcdInsufficientMembers** アラートが誤って実行されていました。このリリースでは、アラートが更新され、インスタンスラベルに加えて Pod ラベルが含まれるようになり、クォーラムが失われた場合にのみアラートが実行されるようになりました。

(BZ#1929944)

- 以前のバージョンでは、SO_REUSEADDR ソケットオプションの導入により readiness プローブが正しい readiness (準備状態) を報告しませんでした。そのため、etcd-quorum-guard が失敗した場合でも etcd Pod は準備完了と表示されていました。readiness プローブチェックはこれらのオプションを考慮して更新され、etcd readiness プローブはオペランドの readiness を適切に反映するようになりました。(BZ#1946607)

- 以前のバージョンでは、**spec.loglevel** フィールドが etcd オペランドに **log-level** フラグを設定していなかったため、ユーザーは etcd ログレベルを変更できませんでした。ユーザーは、以下のようにログレベルを設定できるようになりました。

  - **Debug**、**Trace**、および **TraceAll** ログレベルは、etcd **debug** ログレベルにマップします。

  - **Default** または **Normal** ログレベルは、etcd **info** ログレベルにマップします。

  詳細は、BZ#1948553 を参照してください。

- 以前のバージョンでは、etcd プロセスに従うと、次のプロセスは関連するポートがリリースされるまで開始されませんでした。このプロセスに **SO_REUSEADDR** を追加すると、ポートをすぐに再利用できます。詳細は、BZ#1927942 を参照してください。

- 以前のバージョンでは、**network.Status.ServiceNetwork** フィールドが設定されていないと、etcd-endpoint の ConfigMap は空のままになっていました。その結果、etcd Operator はスケールアップできませんでした。OpenShift Container Platform 4.8 の新規機能により、etcd Operator は **network.Status.ServiceNetwork** フィールドが設定されていなくてもスケールアップできるようになりました。(BZ#1902247)

イメージレジストリー

- 以前のバージョンでは、イメージプルーナーはイメージの削除に失敗した際に停止しました。その結果、2 つのイメージプルーナーがイメージの同時削除を試行する場合、それらのいずれかが **not found** エラーを出して失敗していました。今回の更新により、**not found** エラーは無視され、イメージプルーナーは同時削除を容認できるようになりました。(BZ#1890828)

- 以前のバージョンでは、イメージレジストリー Operator のステータス評価中にルートステータスが含まれていなかったため、ルートが **degraded** 状態であっても、イメージレジストリー Operator のパフォーマンスは低下しませんでした。今回の修正により、イメージレジストリー Operator は設定済みのすべてのルートを取得し、自身のステータスを評価する際にそれらのステータスを評価するようになりました。今回の更新により、ルートのいずれかが **degraded** の場合、イメージレジストリー Operator はエラーメッセージと共に自身を **degrade** として報告します。(BZ#1902076)

- 以前のバージョンでは、自動作成された Docker 設定シークレットには、統合された内部レジストリールートの認証情報が含まれませんでした。いずれかのルートを介してレジストリーにアクセスするための認証情報がなかったため、レジストリーへのアクセスを試行した Pod は認証情報がないために失敗しました。今回の修正では、デフォルトの Docker 認証情報シークレットに設定されたすべてのレジストリールートが含まれています。現在は、認証情報に各ルートのエントリーが含まれるようになったため、Pod は任意のルートで統合レジストリーに到達できるようになりました。(BZ#1911470)

- 以前のバージョンでは、イメージレジストリーはクラスター全体の **ImageContentSourcePolicy** (ICSP) ルールを無視していました。プルスルー中にイメージのミラーは無視され、非接続クラスターでプルの失敗が生じました。今回の更新により、ICSP ルールがターゲットリポジトリーに存在する場合、レジストリーはミラーからプルするようになりました。その結果、設定されたミラーからイメージをプルしても失敗しなくなりました。(BZ#1918376)

- 以前のバージョンでは、イメージレジストリー Operator は設定リソースの **.status.readyReplicas** フィールドを更新しなかったため、その値は常に **0** でした。今回の修正により、イメージレジストリー Operator は準備状態にあるイメージレジストリーのレプリカ数をデプロイメントから設定に書き込むようになりました。現在、このフィールドには、準備ができているイメージレジストリーレプリカの数が表示されます。(BZ#1923811)

- Azure では、**v1** ではなく、ストレージアカウント **v2** を使用するようユーザーに勧めています。特定のセキュリティープロファイルでは、管理者は Azure に対して **v1** ストレージアカウントの作成を許可しないように強制できます。イメージレジストリーは **v1** のストレージアカウントに依存するため、クラスターのインストールはこのような環境で失敗します。今回の修正により、クラスターのブートストラップ中に、イメージレジストリー Operator は **V2** ストレージアカウントの作成および使用を試みるようになりました。**v1** で実行されているクラスターは、引き続き **V1** ストレージアカウントを使用します。インストールは成功し、イメージレジストリー Operator が **Available** を報告するようになりました。(BZ#1929654)

### ImageStreams

- 以前のバージョンでは、ストリームから複数のイメージをインポートする場合にパフォーマンスが遅くなることがありました。今回のリリースにより、イメージレジストリーへの同時要求の数が 5 から 50 に増え、パフォーマンスが向上します。(BZ#1954715)

### Insights Operator

- 以前のバージョンでは、Insights Operator は **openshift-cluster-version** namespace で Cluster Version Operator (CVO) Pod またはイベントを収集しませんでした。その結果、Insights Operator は、CVO で発生する可能性のある問題に関する情報を表示せず、ユーザーは CVO に関する診断情報を取得できませんでした。Insights Operator が更新され、**openshift-cluster-operator** namespace から CVO Pod とイベントが収集されることで、CVO の問題が Insights Operator によって報告されるようになりました。(BZ#1942271)

### Installer

- 以前は、IPv6 ネットワークが /64 以外になっている場合は、DNSmasq はプレフィックス長を指定する必要がありました。その結果、コントロールプレーンのホストは PXE ブートに失敗しました。この更新には、DNSmasq 設定のサブネットプレフィックス長が含まれます。その結果、コントロールプレーンホストは、任意のプレフィックス長の IPv6 ネットワークで DHCP および PXE ブートを実行します。(BZ#1927068)

- vSphere にインストールする場合、ブートストラップマシンは **/etc/resolv.conf** ファイルのネームサーバーを正しく更新しないことがありました。その結果、ブートストラップマシンは一時的なコントロールプレーンにアクセスできず、インストールは失敗しました。この修正には、更新する適切な行の検索の信頼性を高める変更が含まれています。ブートストラップマネージャーが一時的なコントロールプレーンにアクセスできるようになったため、正常にインストールすることができます。(BZ#1967355)

- 以前のバージョンでは、インストーラーはその URL の生成時にブートストラップ Ignition 設定を配置するリージョンを考慮しませんでした。その結果、ブートストラップマシンは、提供された URL が正しくないため、設定を取得できませんでした。今回の更新により、URL の生成時にユーザーのリージョンを考慮し、正しいパブリックエンドポイントを選択するようになりました。その結果、インストーラーは常に正しいブートストラップ Ignition URL を生成します。(BZ#1934123)

- 以前のバージョンでは、ストレージアカウントを作成する際、Azure の最小 TLS のデフォルトバージョンは 1.0 でした。その結果、ポリシーチェックは失敗しました。今回の更新では、ストレージアカウントの作成時に最小 TLS バージョンを 1.2 に設定するように openshift-installer Azure クライアントを設定します。その結果、ポリシーチェックに合格するようになりました。(BZ#1943157)

- 以前のバージョンでは、Azure で IPI を使用してデプロイされたプライベートクラスターには、ブートストラップノードへの SSH を許可するインバウンド NSG ルールがありました。これにより、Azure のセキュリティーポリシーがトリガーされる可能性があります。今回の更新で、NSG ルールが削除されました。(BZ#1943219)

- 以前のバージョンでは、インストーラーは **ap-northeast-3** AWS リージョンを認識しませんでした。今回の更新により、インストーラーは既知のパーティションのパターンに適合する不明なリージョンへのインストールを許可します。これにより、ユーザーは **ap-northeast-3** AWS リージョンでインフラストラクチャーを作成できるようになりました。(BZ#1944268)

- 以前のバージョンでは、オンプレミスプラットフォームには内部ロードバランサーを作成する機能がありませんでした。今回の更新により、ユーザーがマニフェストを作成する際にチェックが追加され、このストラテジーが AWS、Azure、GCP などのクラウドプラットフォームでのみ使用されるようになりました。(BZ#1953035)

- 以前のバージョンでは、Google Cloud Platform リソースに名前を付ける際、フィルターによって、**Google** という単語を使用する特定の名前を使用できませんでした。この更新により、クラスター名のインストーラーにチェックが追加され、名前を設定するときに **Google** という単語のいくつかのバリエーションを使用できるようになりました。(BZ#1955336)

- 以前のバージョンでは、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たベアメタルのインストールでは、インストーラープロセスでプロビジョニングネットワークと通信できる必要がありました。現在、インストーラープロセスは、API サーバーの仮想 IP と通信できるようになりました。この変更により、プロビジョニングネットワークがルーティング可能ではなく、インストーラープロセスが Red Hat OpenStack Platform (RHOSP) または Red Hat Advanced Cluster Management 向けの Hive などのリモートのロケーションから実行されるケースが可能になります。API サーバーの仮想 IP で TCP ポート **6385** および **5050** との通信を許可するように、ファイアウォールルールを調整しなければならない場合があります。(BZ#1932799)

- 以前のリリースでは、Red Hat OpenStack Platform (RHOSP) へのインストールで、**platform.openstack.machinesSubnet** フィールドに存在しないサブネットの ID が提供されると、**openshift-install** コマンドは SIGSEGV およびバックトレースを生成していました。**openshift-install** コマンドが修正され、以下のメッセージのようなエラーが生成されるようになりました。

  ```
  FATAL failed to fetch Metadata: failed to load asset "Install Config":
  platform.openstack.machinesSubnet: Not found: "<network-ID>"
  ```

  (BZ#1957809)

- 以前のリリースでは、RHOSP HTTPS 証明書がホスティングデバイスにインポートされない限り、Red Hat OpenStack Platform (RHOSP) へのインストールは失敗していました。**cloud.yaml** の **cacert** 値が RHOSP HTTPS 証明書に設定されている場合に、インストールが正常に実行されるようになりました。ホストへの証明書のインポートは不要になりました。(BZ#1786314)

- 以前のバージョンでは、**proxy.config.openshift.io** の外部ネットワークエントリーが正しくないため、インストールが失敗する可能性がありました。検証チェックにより、これらの間違いが識別され、修正が可能になりました。(BZ#1873649)

- 以前は曖昧または紛らわしかった Terraform コンポーネントの説明が、より明確な情報に置き換えられました。(BZ#1880758)

- gophercloud/utils に対する以前の変更により、自己署名証明書を使用するカスタム HTTP クライアントが導入されました。この変更により、プロキシー環境変数の設定を含む

**DefaultTransport** の設定が削除されたため、自己署名証明書とプロキシーの両方を使用するインストールで失敗が発生しました。この更新では、カスタム HTTP クライアントが **DefaultTransport** の設定を継承するため、自己署名証明書およびプロキシーを使用して OpenShift Container Platform をインストールできるようになりました。(BZ#1925216)

- 以前のバージョンでは、インストーラーは検証中にインストール設定の **defaultMachineSet** 値を考慮していなかったため、インストーラーが失敗していました。この更新により、デフォルト値がインストール設定に適用され、空のフィールドの検証が開始されます。(BZ#1903055)

- 以前のバージョンでは、**soft-anti-affinity** は、クライアントが最小の Nova マイクロバージョンを設定する必要がありました。Ansible OS サーバーモジュールのほとんどのバージョンでは、クライアントが最小値を自動的に設定する必要はありませんでした。その結果、soft-anti-affinity コマンドは失敗する可能性がありました。この更新では、soft-anti-affinity を処理する際に、Python OpenStack クライアントを使用して Novaマイクロバージョンを設定する方法が修正されています。(BZ#1910067)

- 以前のバージョンでは、OpenStack UPI Playbook は作成されたすべてのリソースにタグを付けていませんでした。その結果、**openshift-install destroy** コマンドは、すべてのクラスターリソースを適切に識別できず、タイムアウトに達するまでリソースの削除をループし、これによりリソースが残されました。この更新により、欠落していたタグ付けに関する指示が OpenStack UPI Playbook に追加されました。(BZ#1916593)

- 以前のバージョンでは、Python パッケージエラーが原因で **e2e-gcp-upi** が成功せず、失敗していました。この更新により、gsutil の正しい Python バージョン (pip バージョン) および **CLOUDSDK_PYTHON** を設定して、パッケージエラーを解決できます。(BZ#1917931)

- 以前のバージョンでは、pip バージョン 21 はインストールされた Python バージョン 2 をサポートしていませんでした。その結果、コンテナーのセットアップに必要なすべての依存パッケージの解決中にエラーが発生しました。この更新では、問題を回避するために、pip バージョンが 21 未満の値に修正されています。(BZ#1922235)

- 以前のバージョンでは、インストーラーはクラウドに関する情報を 2 回収集していました。その結果、OpenStack API へのリクエスト数が 2 倍になり、クラウドに追加の負荷がかかり、インストール時間が長くなりました。この更新では、クォータをチェックする前にクラウドに関する情報を収集し、同じ情報を検証に再利用することで、問題を修正しています。(BZ#1923038)

- 以前のバージョンでは、/64 以外のサブネットを持つ IPv6 プロビジョニングネットワークでデプロイする場合、DNSmasq はプレフィックス長を指定する必要がありました。そのため、/64 以外のネットワークを使用している場合、ホストは PXE ブートに失敗していました。この更新には、DNSmasq 設定のプレフィックス長が含まれています。その結果、ホストは DHCP で成功し、任意のプレフィックス長の IPv6 ネットワークでPXE ブートを実行します。(BZ#1925291)

- 以前のバージョンでは、OpenShift Container Platform インストーラーは、**Shared Subnet** タグを削除する際に、このタグは削除されているとロギングが示していても、IAM パーミッションの問題を報告していませんでした。この更新により、タグ付け解除およびロギングエラーの結果がチェックされます。ログには、共有リソースのタグ解除のステータスが表示されるようになりました。(BZ#1926547)

- 以前のバージョンでは、Azure クラスターは Premium_LRS のディスクタイプと、クラスターが失敗する原因となった PremiumIO 機能をサポートしていないインスタンスタイプで作成されていました。この更新では、ディスクタイプがデフォルトのディスクタイプである Premium_LRS である場合にのみ、選択されたインスタンスタイプに PremiumIO 機能があるかどうかを確認します。コードは、Azure サブスクリプションとリージョンにクエリーを実行して必要な情報を取得し、条件が満たされない場合はエラーを返します。(BZ#1931115)

- 以前のバージョンでは、API サーバーの再起動時に API VIP がブートストラップで利用できなくなり、これにより、プロビジョニングサービスが利用できなってプロビジョニングに失敗していました。プロビジョニングサービス (Ironic) が VIP ヘルスチェックに含まれるようになり、API VIP は引き続き利用可能となります。(BZ#1949859)

kube-apiserver

- 以前のバージョンでは、Google Cloud Platform (GCP) ロードバランサーのヘルスチェッカーがホストに古い conntrack エントリーを残していたため、GCP ロードバランサーを使用する API サーバートラフィックにネットワークの中断が発生していました。ヘルスチェックトラフィックがホストをループしなくなったため、API サーバーに対するネットワークの中断がなくなりました。(BZ#1925698)

Machine Config Operator

- 以前のバージョンでは、**drain timeout** とプールのパフォーマンスの低下期間が短すぎたため、より多くの時間を必要とする通常のクラスターで、アラートが早期に発生していました。この更新により、タイムアウトが障害を報告するまでに必要な時間が延長されました。これにより、クラスター Operator には、通常のクラスターのパフォーマンスを早期に低下させることなく、より現実的で有用なアラートが提供されます。(BZ#1968019)

- 以前のバージョンでは、OpenShift 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 (IPI) を使用して VMware vSphere から新しい仮想マシンを作成中に、ノードがクラスターに参加できませんでした。これは、Dynamic Host Configuration Protocol (DHCP)が、IDI によって提供された名前の代わりにホスト名を入力したときに発生しました。これは解決されています。(BZ#1920807)

- 以前のバージョンでは、ホスト名が設定される前にネットワークが有効化されると、インストールが失敗する可能性がありました。これにより、ノードがクラスターに参加できなくなり、もう一度試行するまでに 5 分間の遅延が強制されました。この問題は解決され、ノードは最初の試行時に自動的にクラスターに参加します。(BZ#1899187)

- 以前のバージョンでは、ユーザーはコアユーザーおよび関連する SSH キーを削除できましたが、キーは残っていました。この更新により、ユーザーはコアユーザーを削除できなくなります。(BZ#1885186)

- 4.6 から 4.7 にアップグレードする場合、**vsphere-hostname** サービスで設定したホスト名は、ノードのインストール時にのみ適用されました。アップグレード前にホスト名が静的に設定されていない場合には、ホスト名が失われる可能性があります。今回の更新により、ノードがインストールされている場合にのみ、**vsphere-hostname** サービスの実行を許可していた条件が削除されます。その結果、vSphere ホスト名はアップグレード時に失われなくなりました。(BZ#1942207)

- **keepalived** 2.0.10 のバグにより、liveness プローブが **keepalived** コンテナーを終了すると、システムに割り当てられた仮想 IP アドレス (VIP) はそのまま残り、**keepalived** の再起動時にクリーンアップされませんでした。その結果、複数のノードが同じ VIP を保持する可能性がありました。現在は、**keepalived** の起動時に VIP が削除されるようになりました。その結果、VIP は単一ノードで保持されます。(BZ#1931505)

- 以前のバージョンでは、rpm-ostree 関連の操作は、Red Hat Enterprise Linux CoreOS (RHCOS) などの CoreOS 以外のノードで適切に処理されませんでした。その結果、カーネルの切り替えなどの操作が RHEL ノードを含むプールに適用されると、RHEL ノードのパフォーマンスは低下していました。今回の更新により、Machine Config Daemon は、CoreOS 以外のノードでサポート対象外の操作が実行されるたびに、メッセージをログに記録します。メッセージのロギング後、エラーではなく nil を返します。プール内の RHEL ノードは、サポート対象外の操作が Machine Config Daemon によって実行された場合、想定どおりに続行されるようになりました。(BZ#1952368)

- 以前のバージョンでは、空の静的 Pod ファイルは **/etc/kubernetes/manifests** ディレクトリーに書き込まれていました。その結果、kubelet ログは、一部のユーザーとの混乱を引き起こす可能性のあるエラーを報告していました。空のマニフェストは、不要な場合には別の場所に移動されるようになりました。その結果、エラーは kubelet ログに表示されなくなりました。(BZ#1927042)

メータリング Operator

- 以前のバージョンでは、レポート Operator は、イベントの調整時にユーザーが指定する保持期間を含む **Report** カスタムリソース (CR) を誤って処理していました。その結果、**Report** CR の有効期限が切れると、影響を受けるカスタムリソースは無限に再びキューに入れられるため、レポート Operator が継続的にループしました。今回の更新により、保持期間が指定された期限切れの **Report** CR が再びキューに入れられることはなくなります。その結果、レポート Operator は期限切れの **Report** CR のイベントを正しく処理します。(BZ#1926984)

モニタリング

- 以前のバージョンでは、**node-exporter** デーモンセットの **mountstats** コレクターにより、NFS マウントポイントが設定されたノードでのメモリー使用量が高くなっていました。今回の更新で、ユーザーは **mountstats** コレクターを無効にして、メモリー使用量を軽減できるようになりました。(BZ#1955467)

ネットワーク

- 以前のバージョンでは、**keepalived** の誤った設定により、間違ったシステムに VIP が配置される場合があり、正しいシステムに戻れませんでした。この更新では、VIP が正しいシステムに配置されるように、**keepalived** の誤った設定が削除されます。(BZ#1916890)

- iptables の書き換えルールにより、サービス IP と Pod IP の両方を介してサービスに接続するために固定ソースポートを使用するクライアントでは、ポートの競合による問題が発生する可能性があります。今回の更新により、追加の OVS ルールが挿入され、ポートの競合が発生したときに通知し、その競合を回避するために追加の SNAT を実行します。その結果、サービスへの接続時にポートの競合が発生しなくなりました。(BZ#1910378)

- 以前のバージョンでは、コントロールプレーンノードと egress が割り当てられたノード間の IP ポート 9 は、内部ファイアウォールによってブロックされていました。これにより、egress ノードへの IP アドレスの割り当てが失敗していました。この更新により、IP ポート 9 を介したコントロールプレーンと egress ノード間のアクセスが可能になります。その結果、egress ノードへの IP アドレスの割り当てが正常に許可されるようになりました。(BZ#1942856)

- 以前のバージョンでは、無効になった古い接続追跡エントリーが原因で、UDP サービストラフィックがブロックされる可能性がありました。これにより、サーバー Pod へのアクセスは、**NodePort** サービスに切り替えられた後に妨げられました。この更新により、**NodePort** サービス切り替えの場合は、接続追跡エントリーが削除され、これにより、新しいネットワークトラフィックが切り替えられたエンドポイントに到達できるようになります。(BZ#1949063)

- 以前のバージョンでは、OVN-Kubernetes ネットワークプロバイダーは、複数の **ipBlocks** を持つネットワークポリシーを無視していました。最初の ipBlock 後のすべての ipBlock が無視され、これにより、Pod は設定されたすべての IP アドレスに到達できなくなりました。Kubernetes ネットワークポリシーから OVN ACL を生成するためのコードが修正されました。その結果、複数の **ipBlocks** を持つネットワークポリシーが正しく機能するようになりました。(BZ#1953680)

- 以前のバージョンでは、OVN-Kubernetes クラスターネットワークプロバイダーを使用すると、エンドポイントのない Kubernetes サービスが誤って接続を受け入れていました。この更新により、エンドポイントのないサービス用にロードバランサーが作成されなくなったため、ト

ラフィックは受け入れられなくなりました。(BZ#1918442)

- 以前のバージョンでは、Multus の Container Network Interface (CNI) プラグインは、ゼロで始まる数値の IPv6 アドレスを理解していませんでした。この更新により、CNI プラグインはゼロより大きい値で始まる IPv6 で動作します。(BZ#1919048)

- 以前は、マシン設定ポリシーの変更によって再起動がトリガーされる際に、SR-IOV ネットワーク Operator が再起動を開始した場合、競合状態がトリガーされる可能性がありました。競合状態が発生した場合、ノードは不確定な状態のままになりました。この更新により、そのような状況が回避されます。(BZ#1921321)

- 以前は、Kuryr クラスターネットワークプロバイダーを使用して、ユーザーによってプロビジョニングされる新しいクラスターを作成すると、クラスターノードによって使用される OpenStack サブセットが検出されず、クラスターのインストールがタイムアウトになることがありました。この更新により、サブネットが正しく検出され、ユーザーによってプロビジョニングされるインストールが成功します。(BZ#1927244)

- 以前は、OpenShift Container Platform 4.6 から OpenShift Container Platform 4.7 にアップグレードする際に、Cluster Network Operator (CNO) は次のバージョンへのアップグレードを完了したと誤ってマークしていました。後でアップグレードが失敗した場合、CNO は自身のパフォーマンスを **degraded** と報告しましたが、バージョンは 4.7 であると誤って報告していました。この更新により、CNO は、クラスターネットワークプロバイダーイメージが正常にアップグレードされるのを待ってから、CNO のアップグレードが成功したと報告します。(BZ#1928157)

- 以前は、OVN-Kubernetes クラスターネットワークプロバイダーを使用する際に、Kubernetes バージョンに数字以外の文字を含むマイナーバージョンが含まれていると、エンドポイントスライスコントローラーが実行されないことがありました。この更新により、エンドポイントスライスコントローラーはデフォルトで有効化されました。(BZ#1929314)

- Kuryr クラスターネットワークプロバイダーを使用する場合、インストール後に作成された Neutron ポートは、インストール中に作成された Neutron ポートとは異なるパターンで名前が付けられました。その結果、インストール後に作成された Neutron ポートは、デフォルトのロードバランサーに追加されませんでした。この更新により、Kuryr はどちらの命名規則で作成されていても Neutron ポートを検出します。(BZ#1933269)

- 以前は、Open Virtual Network (OVN) がヘアピントラフィックパケットのソース IP アドレスをロードバランサーの IP アドレスに変更していました。これにより、ネットワークポリシーの使用中にトラフィックがブロックされることがありました。この更新により、Kuryr はネットワークポリシーの namespace 内のすべてのサービスの IP アドレスへのトラフィックを開き、ヘアピントラフィックはブロックされなくなりました。(BZ#1920532)

- 以前は、IPv4 アドレスを持つノードでシングルスタック IPv6 クラスターを開始すると、kubelet はノード IP に IPv6 IP ではなく IPv4 IP を使用していた可能性がありました。その結果、ホストネットワーク Pod には IPv6 IP ではなく IPv4 IP が含まれるため、IPv6 のみの Pod からは到達できなくなっていました。この更新により、ノード IP ピッキングコードが修正され、kubelet が IPv6 IP を使用するようになりました。(BZ#1939740)

- 以前のバージョンでは、不明な理由により、kubelet はノードに誤った IP アドレスを登録する可能性がありました。その結果、ノードは再起動されるまで **NotReady** 状態になります。systemd マネージャーの設定は環境変数として有効な IP アドレスで再読み込みされるようになりました。つまり、kubelet が誤った IP アドレスを登録することによりノードが **NotReady** 状態になることはなくなりました。(BZ#1940939)

- 以前のリリースでは、シャドウ変数のリファクタリングにより、チェックポイントファイルの使用に関連するリグレッションが生じ、SR-IOV Pod サンドボックスが起動しませんでした。kubelet ソケットのパスの確認は、リファクタリング時に適切に考慮されませんでした。この修

正により、kubelet ソケットパスの確認が適切に復元され、SR-IOV Pod サンドボックスが適切に作成されるようになりました。(BZ#1968625)

ノード

- 以前は、Reliable Autonomic Distributed Object Store (RADOS) Block Devices (RBD) は、**lsblk** を実行している特権なしのコンテナー Pod に表示されていました。これは修正され、**lsblk** を実行している特権なしのコンテナー Pod に RBD が表示されなくなりました。(BZ#1772993)

- 以前は、クラスターのアップグレード中に CRI-O が **/etc/hostname** ファイルを変更していたため、ノードに障害が発生し、再起動時に元に戻りました。この更新により、アップグレード中は **/etc/hosts** ファイルをそのままにするように CRI-O に特別な処理が加えられ、アップグレードされたノードは問題なく起動できるようになりました(BZ#1921937)

- 以前は、ネットワークがプロビジョニングされた後の Pod の作成に CRI-O は時間をかけすぎていました。これにより、ネットワーククリーンアップコードにバグが発生し、ネットワークリソースがプロビジョニングされた後にネットワークリソースが適切にクリーンアップされなくなっていました。この更新により、コマンドがタイムアウトした場合でも、ネットワークリソースが適切にクリーンアップされるようにコードが変更されます。これにより、Pod の作成に時間がかかりすぎた場合でも、クラスターは通常のネットワーク操作を続行できます。(BZ#1957224)

- 以前は、**CNI** プラグインを使用したノードの再起動は、正常に完了しませんでした。CRI-O は、再起動前に実行されていたすべてのコンテナーで、**CNI DEL** を呼び出すように変更されました。この更新により、**CNI** リソースがクリーンアップされ、正常に再起動できるようになります。(BZ#1948137)

- 以前は、**CNI** クリーンアップ操作はクリーンアップの失敗をチェックしなかったため、**CNI DEL** 要求が失敗した場合は、再度呼び出されませんでした。現在は、CRI-O は成功するまで **CNI DEL** 要求を再度呼び出し、**CNI** リソースを正しくクリーンアップします。(BZ#1948047)

- 以前は、コンテナーまたはイメージがディスクにコミットされているときに再起動が起きた場合、コンテナーまたはイメージへの再起動要求が失敗を引き起こす可能性がありました。これにより、コンテナーのストレージが明らかに破損し、イメージのプルまたはイメージからのコンテナーの再作成に失敗しました。この更新は、ノードが再起動されたことを検出し、true の場合はコンテナーストレージをクリアにします。(BZ#1942536)

- 以前は、**runc** は、自身を実行したエンティティーのパーミッションを引き継ぎました。ただし、**workdir** のパーミッションは、**container** ユーザーによって設定されます。これらのパーミッションが異なると、コンテナー作成エラーが発生し、コンテナの起動に失敗しました。このパッチは、1 回だけ失敗した場合に備えて、**runc** を **chdir** から **workdir** に複数回更新します。これにより、コンテナーの作成は成功します。(BZ#1934177)

- 以前のバージョンでは、CRI-O ログには、イメージのプル元のソースについての情報が含まれていませんでした。この修正により、ログプルソースが CRI-O ログの情報レベルに追加されます。(BZ#1881694)

- 以前のバージョンでは、Pod が迅速に作成および削除された場合、Pod が削除を開始するまでに、Pod サンドボックスの作成を完了するための十分な時間がない場合がありました。その結果、「ErrCreatePodSandbox」エラーが表示され、Pod の削除が失敗する可能性がありました。このエラーは、Pod が終了している場合に無視されるようになりました。その結果、Pod が Pod サンドボックスの作成を完了できなくても、Pod の終了が失敗することはなくなりました。(BZ#1908378)

- 以前のバージョンでは、Machine Config Operator (MCO) は トレース を有効なログレベルとして許可しませんでした。その結果、CRI-O がサポートしていても、MCO はトレースレベルのロギングを有効にする方法を提供できませんでした。MCO が トレース ログレベルをサポー

トするように更新されました。その結果、ユーザーは MCO 設定を通じてトレースログレベルを確認できます。(BZ#1930620)

- 以前のバージョンでは、kubelet は完全にプルされていないイメージのステータスを取得しようとしました。その結果、**crictl** はこれらのイメージの **error locating item named "manifest"** エラーを報告しました。CRI-O は、マニフェストのないイメージを一覧表示しないように更新されました。その結果、**crictl** はこれらのエラーを報告しなくなりました。(BZ#1942608)

- 以前のバージョンでは、古いステータスメッセージが削除されませんでした。そのため、Machine Config Operator (MCO) は適切なマシン設定プールを見つけることができない場合がありました。今回のリリースにより、クリーンアップ機能が追加され、ステータスの数を制限できるようになりました。その結果、MCO は最大 3 つの異なる kubeletConfig ステータスを保持します。(BZ#1950133)

- 以前のバージョンでは、OpenShift Container Platform バージョン 4.6.25 からアップグレードする際に、複数の **kubeletconfig** CR または **ContainerRuntimeConfig** CR を持つクラスターで、Machine Config Operator (MCO) が同じ設定に対して、重複するマシン設定を生成する可能性がありました。その結果、MCO は古いコントローラーバージョン (IGNITIONVERSION 3.1.0) を使用するため、アップグレードに失敗していました。今回の更新により、古い重複マシン設定がクリーンアップされ、バージョン 4.6.25 から適切にアップグレードできるようになります。(BZ#1955517)

oauth-apiserver

- 以前のバージョンでは、一部の OAuth サーバーメトリクスは適切に初期化されず、Prometheus UI での検索には表示されませんでした。不足している OAuth サーバーメトリクスが適切に初期化され、Prometheus UI メトリクスの検索に表示されるようになりました。(BZ#1892642)

- 以前のバージョンでは、カスタム SCC (Security Context Constraints) に **defaultAllowPrivilegeEscalation: false** フィールドおよび **allowPrivilegedContainer: true** フィールドの組み合わせが含まれる場合、セキュリティーコンテキストミューテーターは特権付きの **openshift-apiserver** Pod および **oauth-apiserver** Pod を API 検証に失敗した状態に変更しました。Pod は起動に失敗し、OpenShift API が停止することがありました。セキュリティーコンテキストミューテーターは、すでに特権のあるコンテナーの **defaultAllowPrivilegeEscalation** フィールドを無視し、これらのフィールドを含むカスタム SCC は Pod の起動を妨げなくなりました。(BZ#1934400)

oc

- 以前は、**oc explain** コマンドを実行するときに、リソースグループ名がリソース文字列の一部として提供された場合、リソースグループ名は自動的に検出されませんでした。異なるグループの 2 つのリソースが同じリソース名を持っていた場合、グループが **--api-version** パラメーターで指定されていない限り、最も優先度の高い定義が返されていました。現在は、**--api-version** パラメーターが含まれていない場合、リソース文字列に対してプレフィックスチェックが実行され、グループ名が検出されます。コマンドによって返される説明は、指定されたグループの一致するリソースに関連しています。(BZ#1725981)

- 以前は、**oc image extract** コマンドは、イメージのルートディレクトリーからファイルを抽出しませんでした。コマンドが更新され、イメージのルートディレクトリーからファイルを抽出するために使用できるようになりました。(BZ#1919032)

- 以前は、**oc apply**コマンドは、呼び出しごとに OpenAPI 仕様を取得していました。コマンドが最初に実行されたときに、OpenAPI 仕様がキャッシュされるようになりました。キャッシュされた OpenAPI 仕様は、**oc apply** コマンドが複数回実行され、ネットワーク負荷が軽減されたときに再利用されます。(BZ#1921885)

- 以前は、イメージミラーリング中に作成された承認ヘッダーが、一部のレジストリーのヘッダーサイズ制限を超える可能性がありました。これにより、ミラーリング操作中にエラーが発生します。現在は、**oc adm catalog mirror** コマンドの **--skip-multiple-scopes** オプションが **true** に設定され、承認ヘッダーがヘッダーサイズの制限を超えないようになりました。(BZ#1946839)

- 以前は、**oc volume set** コマンドに **--claim-class** オプションが含まれている場合は、**storageClassName** 属性は **PersistentVolumeClaim** オブジェクトに追加されませんでした。代わりに、**--claim-class** オプションの値が **volume.beta.kubernetes.io/storage-class** アノテーションに追加されました。これにより、**storageClassName** 属性の依存関係が原因で、ボリュームのスナップショットが失敗していました。現在は、**oc volume set** コマンドは **--claim-class** オプションの値を **PersistentVolumeClaim** オブジェクトの **storageClassName** 属性に適用し、ボリュームスナップショットは属性値を参照することができます。(BZ#1954124)

- 以前は、**oc adm top --help** の出力には、**oc adm top** コマンドは Pod とノードの CPU、メモリー、およびストレージリソースの使用状況を表示できると記載されていました。**oc adm top** コマンドは、ストレージリソースの使用状況を表示しません。現在は、ストレージ参照は **oc adm top --help** の出力に含まれていません。(BZ#1959648)

### Operator Lifecycle Manager (OLM)

- 以前のバージョンでは、Operator のインストールの一部として適用される **CustomResourceDefinition** (CRD) オブジェクトが、同じ Operator の新規バージョンのインストール要件を満たす場合がありました。その結果、Operator のアップグレード中に、置き換えられるバージョンが途中で削除される可能性がありました。場合によっては、アップグレードが停止することがありました。今回の更新により、Operator バンドルインストールの一部として作成または更新される CRD には、元のバンドルを示すアノテーションが付けられるようになりました。これらのアノテーションは、**ClusterServiceVersion** (CSV) オブジェクトによって使用され、既存の CRD と同じバンドルの CRD を区別します。その結果、現行バージョンの CRD が適用されるまでアップグレードは完了しません。(BZ#1947946)

- 以前のバージョンでは、**CatalogSource** オブジェクトによって参照されるインデックスを実行する Pod には、**securityContext** フィールドに明示的に設定された **readOnlyRootFileSystem: false** がありませんでした。そのため、**readOnlyRootFileSystem: true** を実施し、その Pod の **securityContext** に一致する SCC (Security Context Constraints) が存在する場合、SCC はその Pod に割り当てられ、繰り返し失敗します。今回の更新により、**securityContext** フィールドに **readOnlyRootFileSystem: false** が明示的に設定されるようになりました。その結果、**CatalogSource** オブジェクトによって参照される Pod は、読み取り専用のルートファイルシステムを実施する SCC と一致しなくなり、失敗しなくなりました。(BZ#1961472)

- 以前は、Operator Lifecycle Manager (OLM) は、初期インストール時に **startingCSV** フィールドでバージョンが指定されていた場合、スキップされたバージョンのインストールを許可していませんでした。これにより、ユーザーがスキップされたバージョンをインストールしたい場合でも、スキップされた理由に関係なく、それらをインストールできなくなりました。この修正により OLM が更新され、**Subscription** オブジェクトの **startingCSV** 仕様を使用して、初期インストール時にのみ、ユーザーはスキップされたバージョンをインストールできるようになります。ユーザーがスキップされたバージョンにアップグレードできないのは、想定どおりです。(BZ#1906056)

- **k8s.io/apiserver** は Webhook 承認者のコンテキストエラーを処理していなかったため、タイムアウトなどのコンテキストエラーにより、承認者はパニックに陥りました。この修正により、API サーバーのバージョンが増分され、問題のアップストリーム修正が含まれるようになります。その結果、承認者はコンテキストエラーを適切に処理できます。(BZ#1913525)

- 以前は、エアギャップ環境全体で Operator カタログをミラーリングするために、**oc adm catalog mirror** コマンドを使用することは簡単ではありませんでした。この機能拡張により、

カタログの内容をファイルシステムにミラーリングし、リムーバブルメディアに配置してから、エアギャップクラスターで使用するためにファイルシステムからレジストリーにミラーリングして戻すことができます。(BZ#1919168)

- 以前のバージョンでは、カタログ Operator は、タイムアウトを設定せずに、インストールプランのバンドルアンパックジョブを作成していました。これにより、バンドルイメージが存在しないか削除された場合は、ジョブが永久に実行され、ジョブの Pod がイメージの解決に失敗したことを示すことなく、インストールプランは **Installing** フェーズにとどまりました。この修正により、カタログ Operator は、バンドルアンパックジョブにデフォルトの **10m** タイムアウトを設定するようになりました。これは、**--bundle-unpack-timeout** フラグを使用して設定できます。その結果、バンドルアンパックジョブが設定されたタイムアウト後に失敗し、**status.conditions** プロパティーと **status.bundleLookups.conditions** プロパティーに表示される理由により、インストールも **Failed** フェーズに移行します。(BZ#1921264)

- 以前のバージョンでは、OpenShift Container Platform 4.6 より前のクラスターにインストールされた Operator は、依存関係の解決とアップグレードの選択の目的で、特定の Operator パッケージからのものとして識別されていませんでした。これにより、既存の Operator インストールが独自のサブスクリプションの基準と競合し、namespace 内のアップグレードと依存関係の解決がブロックされました。この修正により OLM が更新され、サブスクリプションによって参照される Operator のパッケージ名とバージョンを推測するようになりました。その結果、アップグレードと依存関係の解決は想定どおりに進行します。(BZ#1921953)

- 一時的なエラーに使用される **Info** ログレベルにより、デフォルト設定の詳細な OLM Operator ログが発生しました。この修正により、一時的なエラーログレベルが **debug** に変更されます。その結果、**debug** 設定での重要ではないログの表示が減りました。(BZ#1925614)

- 以前のバージョンでは、**Subscription** オブジェクトの **spec.config.resources** セクションは、設定されていないか空の場合でも、インストールされたデプロイメントに常に適用されていました。これにより、クラスターサービスバージョンで (CSV) で定義されたリソースが無視され、**Subscription** オブジェクトの **spec.config.resources** セクションで定義されたリソースのみが使用されました。この修正により、**spec.config.resources** セクションが nil 以外または空でない値に設定されている場合にのみ、デプロイメント固有のリソースをオーバーライドするように OLM が更新されます。(BZ#1926893)

- 以前のバージョンでは、依存関係とアップグレードの解決中のサブスクリプションの一意性は、サブスクライブされたパッケージ名に基づいていました。namespace 内の 2 つのサブスクリプションが同じパッケージをサブスクライブする場合、それらは内部的には単一のサブスクリプションとして扱われ、想定外の動作が発生していました。この修正により、サブスクリプションは内部的に **.spec.name** ではなく **.metadata.name** で、namespace 内で一意に識別されるようになりました。その結果、アップグレードと依存関係の解決の動作は、同じ **.spec.name.** 持つ複数の **Subscription** オブジェクトを含む namespace で一貫しています。(BZ#1932001)

- 次のカタログ更新ポーリングの試行までの時間が 1 分未満の場合、間隔ジッター関数は再同期の間隔をゼロまで切り捨てます。これにより、Operator カタログがホットループに入り、CPU サイクルが無駄になりました。この修正により、再同期の遅延の計算に使用されるジッター関数の精度が向上します。その結果、カタログ Operator は、次のカタログ更新ポーリングまでほとんどアイドル状態のままになります。(BZ#1932182)

- Operator のアップグレード中に、関連する **ServiceAccount** オブジェクトの所有者参照が更新され、古いオブジェクトではなく、新しい**ClusterServiceVersion** (CSV) オブジェクトを指すようになりました。これにより、CSV を調整する OLM Operator と、インストールプランを実行するカタログ Operator との間で競合状態が発生し、サービスアカウントの所有権の変更により古い CSV が **Pending**/**RequirementsNotMet** としてマークされる可能性があります。これにより、古い CSV が正常なステータスを示すことを新しい CSV が無期限に待機している間に、アップグレードの完了がブロックされました。この修正により、所有者参照を 1 つの手順で更

新する代わりに、2 番目の所有者が既存の所有者に追加されるようになりました。その結果、同じサービスアカウントで、古い CSV と新しい CSV の両方の要件を満たすことができます。(BZ#1934080)

- 以前のバージョンでは、クラスターサービスバージョン (CSV) では、関連するサービスアカウントが **ownerReferences** 値を設定しないか、または関連する CSV に **ownerReferences** 値を設定する必要がありました。これにより、Operator のインストールの一部として作成されていない **default** のサービスアカウントは、**metadata.ownerReferences** フィールドが空でない場合、CSV 要件として満たされませんでした。この修正により、CSV では、関連付けられたサービスアカウントが、CSV に **ownerReferences** 値を設定しないか、または関連する CSV に **ownerReference** 値を設定することが必要となりました。その結果、CSV 以外の **ownerReferences** 値のみのサービスアカウントは、CSV の要件を満たすことができます。(BZ#1935909)

- OpenShift Container Platform 4.5 より前は、**openshift-marketplace** namespace で Marketplace Operator によってデプロイおよび管理されたデフォルトのカタログは、Marketplace Operator によって公開された API である **OperatorSource** オブジェクトによって作成されていました。Operator ソースで発生したエラーを示すために、適切なメトリクスとアラートがインストルメント化されました。OpenShift Container Platform 4.6 では、**OperatorSource** リソースはいくつかのリリースで非推奨になった後に削除され、代わりに Marketplace Operator は OLM の **CatalogSource** リソースを直接作成しました。ただし、**openshift-marketplace** namespace にデプロイされたカタログソースに対しては、同じメトリクスとアラートのインストルメンテーションは実行されませんでした。したがって、デフォルトのカタログソースで発生したエラーは、Prometheus アラートでは強調されませんでした。この修正により、OLM に新しい **catalogsource_ready** メトリクスが導入されます。これは、カタログソースが Unready (準備が未完了) 状態にあることをデフォルトのカタログソースのメトリクスが示すたびに、アラートを発生させるために Marketplace Operator が使用します。その結果、**openshift-marketplace** namespace の Unready (準備が未完了) 状態のデフォルトのカタログソースに対して、Prometheus アラートが提供されるようになりました。(BZ#1936585)

- 以前のバージョンでは、候補の Operator 依存関係がデフォルトのチャネルとデフォルト以外のチャネルから利用可能であった場合、Operator Lifecycle Manager (OLM) は、2 つのチャネルのいずれかを任意に指定するサブスクリプションを生成することができました。現在、Operator の依存関係は、最初にデフォルトチャネルからの候補によって満たされ、続いて他のチャネルからの候補によって満たされます。(BZ#1945261)

- 以前のバージョンでは、クラスターサービスバージョン (CSV) が複数の Operator のコンポーネントとしてコピーされる可能性がありました。これは、Operator のインストール後に namespace が Operator グループに追加された場合に発生する可能性があります。この動作は、メモリー使用量と CPU 負荷に影響を及ぼしていました。現在、CSV は、**Copied** の理由で Operator の **status.components** フィールドに表示されず、パフォーマンスへの影響はありません。(BZ#1946838)

### Operator SDK

- 以前のバージョンでは、**ManagedFields** が調整中に処理されていたため、一部のリソースが無限ループに陥っていました。この修正により、**operator-lib** が更新され、 **ManagedFields** が無視されるようになり、ループが一貫して調整されます。(BZ#1856714)

- コマンドラインインターフェース (CLI) で **--help** が渡された際に、デフォルトのプラグインが呼び出されなかったため、Operator SDK の出力されるヘルプメッセージは最小限となっていました。この修正により、デフォルトのプラグインが呼び出され、ユーザーが **operator-sdk init --help** コマンドを実行すると、より有用なヘルプメッセージが出力されます。(BZ#166222)

- 以前のバージョンでは、オプションのバリデーターがない状態で実行すると、警告は表示されずに、**operator-sdk bundle** が失敗していました。これは修正されています。(BZ#1921727)

openshift-apiserver

- 以前のリリースでは、カスタム SCC (Security Context Constraints) は、デフォルトセットの他の SCC よりも優先度を高くすることができました。その結果、これらの SSC は **openshift-apiserver** Pod と一致することがあり、これにより、ルートファイルシステムへの書き込みができなくなりました。また、このバグにより、一部の OpenShift API が停止しました。この修正では、ルートファイルシステムが書き込み可能でなければならないことが、**openshift-apiserver** Pod に明示的に示されています。その結果、カスタム SCC は **openshift-apiserver** Pod の実行を阻止することはできません。(BZ#1942725)

Performance Addon Operator

- 以前のバージョンでは、低レイテンシーの応答を提供するようにコンテナーを設定する場合、CRI-O を使用した動的割り込みマスクが **irqbalance** システムサービスによって設定された割り込みマスクと一致しませんでした。それぞれが異なるマスクを設定し、コンテナーのレイテンシーを損なっていました。この更新では、**irqbalance** システムサービスに一致するように CRI-O を設定することにより、割り込みマスクセットが変更されます。その結果、動的割り込みマスク処理が想定どおりに機能するようになりました。(BZ#1934630)

RHCOS

- 以前のリリースでは、起動プロセスの非常に遅い段階でマルチパスが有効化されていました。その結果、Red Hat Enterprise Linux CoreOS (RHCOS) は、一部のマルチパス環境で I/O エラーを返しました。今回の更新で、起動プロセスの早い段階でマルチパスが有効化されるようになりました。その結果、RHCOS は一部のマルチパス環境で I/O エラーを返さなくなりました。(BZ#1954025)

- 以前のバージョンでは、潜在的な競合状態により、一部の環境で Red Hat Enterprise Linux CoreOS (RHCOS) PXE デプロイメントの rootfs の取得が失敗する可能性がありました。この修正により、rootfs のプルを試みる前に再試行する接続チェックが追加され、**coreos-livepxe-rootfs** スクリプトが失敗することがあるポイントに進む前に、リモートサーバーと rootfs ファイルへのアクセスが検証されるようになりました。(BZ#1871303)

- 以前のバージョンでは、**MachineConfig** のユーザーによる事前設定は無視されていました。これは、ユーザーが **kdump.service** の設定を変更できないことを意味していました。現在は、デフォルトの事前設定の優先度はユーザー設定のデフォルトよりも低いため、ユーザー設定はベンダー設定を適切に上書きできます。(BZ#1969208)

- 以前のバージョンでは、**coreos-installer** は、インストールイメージで上書きする前にターゲットディスクの GUID Partition Table (GPT) を読み取ろうとするため、GPT が破損しているディスクへのインストールを拒否していました。この修正により、**coreos-installer** は、既存のパーティションを保持するように指示される場合にのみターゲットディスクの GPT を読み取ることで、GPT が破損しているディスクに正常にインストールできるようになりました。(BZ#1914976)

- 以前のバージョンでは、フォーマットされていない直接アクセス記憶装置 (DASD) にクラスターをインストールすると、**coreos-installer** によって誤って書き込まれたディスクセクターが作成されていました。現在は、**coreos-installer** は、フォーマットされていない新しい DASD ドライブを 4096 バイトセクターに正しくフォーマットします。これにより、**coreos-installer** は OS イメージのディスクドライブへのインストールを完了することができます。(BZ#1905159)

- 以前のバージョンでは、s390x z15 システムでのハードウェア支援の **zlib** 展開により、RHEL rootfs イメージのマウントが失敗し、RHEL 8.3 カーネルを使用する REHL s390x z15 ノードのブートが失敗していました。ハードウェア支援の **zlib** 圧縮が利用可能な場合に、**zlib** で圧縮された squashfs ファイルを正しく処理するようにカーネルが更新されました。(BZ#1903383)

- 以前のバージョンでは、**zipl** コマンドは 512 バイトのセクターサイズを想定して、ディスクジオメトリーを設定していました。その結果、4k セクターの SCSI ディスクでは、**zipl** ブートローダー設定に誤ったオフセットが含まれ、zVM を起動できませんでした。この修正により、**zipl** は、zVM が正常に起動するようにディスクセクターサイズを考慮するようになりました。(BZ#1918723)

- 以前のバージョンでは、**chrony.config** は自動的に複数回実行され、最初を除いて毎回失敗する可能性がありました。**chrony.config** の設定は初回実行時に設定され、変更できないため、これにより問題が発生しました。これらのエラーは、設定のセットアッププロセスを**chrony.config** の初回実行時に限定することで、回避されるようになりました。(BZ#1924869)

- 以前のバージョンでは、ワークロードの高い期間中は、ノードは正常ではないように見え、想定どおりに動作しませんでした。これは、メモリーが回収されるよりも速く、ワークロードがメモリーを使用することが原因でした。この更新により、メモリーの回収とメモリー不足の状況が解決され、これらの状況はワークロードが高い状況では発生しなくなりました。(BZ#1931467)

- 以前のバージョンでは、カーネル引数を使用するボンドインターフェースの Maximum transmission unit (MTU) 仕様が適切に割り当てられていませんでした。これは修正されています。(BZ#1932502)

- 以前のバージョンでは、**clevis-luks-askpass.path** ユニットはデフォルトで有効になっていませんでした。これにより、root 以外 **LUKS Clevis** デバイスが、再起動時に自動ロック解除に失敗しました。この更新により、デフォルトで **clevis-luks-askpass.path** ユニットが有効化され、root 以外の **LUKS Clevis** デバイスが再起動時に自動ロック解除できるようになりました。(BZ#1947490)

- 以前のバージョンでは、systemd は **mountinfo** を過度に読み取り、CPU リソースを過剰に消費していたため、コンテナーの起動に失敗していました。この更新により、**systemd** が **mountinfo** を読み取る際に制限することが可能となり、コンテナーを正常に開始できるようになりました。(BZ#1957726)

- 以前のバージョンでは、Machine Config Operator (MCO) が起動時に Ignition を呼び出して、Ignition のバージョンを確認すると、Ignition がクラッシュしていました。そのため、MCO は起動に失敗しました。この更新により、MCO は Ignition バージョンをクエリーしなくなり、MCO は正常に起動します。(BZ#1927731)

ルーティング

- 以前のバージョンでは、HAProxyDown のアラートメッセージはあいまいでした。その結果、エンドユーザーは、アラートの意味として、HAProxy Pod ではなくルーター Pod を利用できないものと考えていました。この更新により、HAProxyDown のアラートメッセージがより明確になりました。(BZ#1941592)

- 以前のバージョンでは、ホワイトリスト IP のファイルを生成する HAProxy のヘルパー関数テンプレートは、間違った引数タイプを想定していました。その結果、長い IP リストのバックエンドにホワイトリスト ACL が適用されませんでした。今回の更新により、ヘルパー関数テンプレートの引数タイプが変更され、ホワイトリスト ACL が長い IP リストのバックエンドに適用されるようになりました。(BZ#1964486)

- 以前のバージョンでは、カスタムドメインを使用して Ingress を作成する場合、正規ルーターのホスト名を使用して OpenShift Container Platform Ingress コントローラーによって Ingress のステータスが更新され、**external-dns** を使用して Route 53 と同期していました。問題は、正規ルーターのホスト名が DNS に存在せず、OpenShift Container Platform によって作成されなかったことです。OpenShift Container Platform は、**apps.<cluster_name>.<base_domain>** DNS レコードではなく、***.apps.<cluster_name>.<base_domain>** DNS レ

コードを作成します。そのため、正規ルーターのホスト名が正しくありませんでした。この修正により、正規ルーターのホスト名が **router-default.apps.<cluster_name>.<base_domain>** に設定されます。正規のホスト名を取得してワイルドカードまたはサブドメインを先頭に追加する自動化機能を備えたクラスター管理者は、正規の Ingress ホスト名が **<ingress-controller-name>.apps.<cluster_name>.<base_domain>** として設定されていることに留意する必要があります。(BZ#1901648)

- 以前のバージョンでは、BZ#1932401 の修正により、デフォルトの Go HTTP クライアントトランスポートが上書きされていました。その結果、クラスター全体のプロキシー設定が Ingress Operator Pod に組み込まれなかったため、クラスター全体の egress プロキシーのクラスターでのカナリアチェックが失敗しました。この更新により、カナリアクライアントの HTTP トランスポートにプロキシー設定が明示的に設定されます。その結果、カナリアチェックはすべてのクラスター全体のプロキシーで機能します。(BZ#1935528)

- 以前のバージョンでは、カナリア DaemonSet はノードセレクターを指定していなかったため、カナリア namespace にデフォルトのノードセレクターを使用していました。その結果、カナリア DaemonSet はインフラストラクチャーノードにスケジュールできず、場合によってはアラートをスローしていました。この更新により、カナリア DaemonSet がインフラストラクチャーノードに明示的にスケジュールされ、テイントのマークが付けられたインフラストラクチャーノードが容認されます。これにより、カナリア DaemonSet は、問題やアラートなしにワーカーノードとインフラストラクチャーノードに安全にロールアウトできます。(BZ#1933102)

- 以前のバージョンでは、アイドル状態にされたワークロードを使用してクラスターを以前のバージョンからアップグレードする場合、アイドル状態にされたワークロードは、**oc idle** 機能の修正と再作業により、OpenShift Container Platform 4.6 または 4.7 にアップグレードした後は HTTP 要求で起動しませんでした。今回の更新により、アイドリングの変更が Ingress Operator の起動時にエンドポイントからサービスにミラーリングされるようになりました。その結果、アップグレード後のワークロードのアイドリング解除が予想通りに機能するようになりました。(BZ#1925245)

- 以前のバージョンでは、すべての HTTP トラフィックを HTTPS にリダイレクトする外部ロードバランサーを介してデフォルトの Ingress コントローラーを公開すると、Ingress Operator によって実行される Ingress カナリアエンドポイントチェックが失敗し、最終的に Ingress Operator のパフォーマンスが **degraded** となりました。この修正により、クリアテキストカナリアルートが edge 暗号化ルートに変換されます。安全でないトラフィックがロードバランサーによってリダイレクトされると、カナリアルートは HTTPS のみのロードバランサーを介し機能するようになりました。(BZ#1932401)

- 以前のバージョンでは、Ingress Operator カナリアチェッククライアントは、HTTP トラフィックをドロップしたロードバランサーに HTTP 経由でカナリアリクエストを送信していました。これにより、カナリアチェックが失敗した後、Ingress Operator のパフォーマンスが **degraded** となりました。この修正により、ルーターからのリダイレクトに依存する代わりに、Ingress Operator カナリアチェッククライアントは、最初から HTTPS 経由でカナリアチェック要求を送信します。現在、カナリアチェックは、安全でない HTTP トラフィックをドロップするロードバランサーを介してデフォルトの Ingress Controller を公開するクラスターに対して機能します。(BZ#1934773)

- 以前のバージョンでは、**openshift-router** で使用される HAProxy テンプレートは、**firstMatch()** 関数を繰り返し呼び出していました。その関数は、毎回正規表現を解析して再コンパイルします。**firstMatch()** を呼び出すたびに正規表現を解析して再コンパイルすると、特に何千ものルートがある設定の場合、コストがかかります。この修正により、**firstMatch()** の呼び出しで正規表現がすでに表示されている場合、コンパイル済みのバージョンが再利用され、キャッシュされます。現在は、**haproxy-config.template** を解析および評価する際のランタイムが 60 パーセント短縮されています。(BZ#1937972)

- 以前のバージョンでは、ユーザーはオーバーライドアノテーションを使用して、無効なホスト名でルートに名前を付けることができました。今回の更新でこの問題が修正されています。(BZ#1925697)

- 以前のバージョンでは、ルートを介して公開されたサービスから **selector** を削除すると、サービスの Pod 用に作成される **endpointslices** が重複し、サーバーエントリーの重複が原因で HAProxy の再読み込みエラーが発生していました。この更新により、HAProxy 設定ファイルを書き出すときに誤って重複するサーバー行が除外されるため、サービスからセレクターを削除してもルーターが失敗することはなくなりました。(BZ#1961550)

サンプル

- 以前のバージョンでは、Cluster Samples Operator は監視するオブジェクトのコントローラーのキャッシュに変更を加える可能性がありました。これにより、Kubernetes がコントローラーキャッシュを管理する際にエラーが生じました。今回の更新により、Cluster Samples Operator がコントローラーキャッシュの情報を使用する方法が修正されました。そのため、Cluster Samples Operator はコントローラーキャッシュを変更してもエラーが生じなくなりました。(BZ#1949481)

service-ca

- OpenShift Container Platform 4.8 を使用すると、ユーザーは、root 以外のユーザーとして **service-ca-operator** Podを実行し、組織のニーズに合わせることができます。root 以外のユーザーとして実行する場合、**service-ca-operator** は以下の UID および GID として実行されます。

  ```
  uid=1001(1001) gid=1001 groups=1001
  ```

  (BZ#1914446)

ストレージ

- 以前のバージョンでは、**capacity breakdown** を要求する際に、**block type PVC** ファイルシステムのメトリクスが報告されていませんでした。これは、ユーザーがすべてのファイルシステムにわたって、メトリクスの不正確なレポートを受け取っていたことを意味しました。今回の更新で、Kubelet から要求される場合は、**block type PVC** が含まれるようになりました。これにより、すべてのファイルシステムメトリクスの正確なレポートが提供されます。(BZ#1927359)

- 以前のバージョンでは、**/var/lib/kubelet** が、**Cinder CSI Node Controller** コンテナーに 2 回マウントされていました。これにより、**CSI Node Controller** が起動に失敗し、**/var/lib/kubelet/pods** の空き領域が不足していることを示すエラーが発生しました。この修正により、**/var/lib/kubelet** および **/var/lib/kubelet/pods** の重複マウントが削除され、**CSI Node Controller** を正常に実行できるようになりました。(BZ#1952211)

- 以前のバージョンでは、永続ボリューム (PV) の Cinder CSI ドライバーのサイズ変更中に、**findmnt** コマンドが複数のボリュームマウントを受信し、正しいマウントを選択できなかったため、サイズ変更が停止していました。その結果、ユーザーはファイルシステムを手動で拡張する必要があります。この修正では、PV とともにファイルシステムのサイズが変更されるように、コマンドは最初のマウントを使用するようになりました。(BZ#1919291)

- Cinder CSI ドライバー Operator は、**VolumeSnapshotClass** オブジェクトを手動で作成するのではなく、デフォルトのストレージクラスを作成する際に、Cinder CSI のデフォルトの **VolumeSnapshotClass** オブジェクトを自動的にプロビジョニングするようになりました。(BZ#1905849)

- 以前のバージョンでは、recycler-pod テンプレートが kubelet 静的マニフェストディレクト

リーに誤って配置されていました。この誤った場所では、recycler の静的 Pod の起動が失敗したことを示す静的 Pod ログメッセージが生成されました。この更新により、誤って配置された recycler-pod テンプレートが静的 Pod マニフェストディレクトリーから削除されました。その結果、エラーメッセージは表示されなくなります。(BZ#1896226)

- 以前のバージョンでは、ローカルストレージ Operator (LSO) は、ビジー状態のディスクが誤って空きディスクとして検出されたため、他のプロビジョナーに属するディスクを要求できました。LSO がそれらのディスクを要求できないように、ディスクのバインドマウントがチェックされるようになりました。(BZ#1929175)

- 以前のバージョンでは、device-id に **:** などのサポートされていない文字が含まれていたため、ローカルストレージ Operator (LSO) は、無効なラベル値で永続ボリューム (PV) を作成しようとしていました。これは、デバイス情報をラベルからアノテーションに移動することで修正されました。(BZ#1933630)

- 以前のバージョンでは、削除プログラムが正しくエンキューされていなかったため、ローカルストレージ Operator (LSO) は永続ボリューム (PV) をクリーンアップしていませんでした。これにより、PV は **released** 状態のままになりました。PV が正しくエンキューされるようになり、正しく削除されるようになりました。(BZ#1937145)

- 以前のバージョンでは、Pod が削除されると、ファイバーチャネルボリュームがノードから誤ってアンマウントされていました。これは、ノード上の kubelet が実行されていないときに、ボリュームを使用する別の Pod が API サーバーで削除されたときに発生しました。この更新により、新しい kubelet が起動すると、ファイバーチャネルボリュームは正しくアンマウントされます。さらに、新しい kubelet が完全に起動し、ボリュームがアンマウントされていることを確認するまで、ボリュームを複数のノードにマウントすることはできません。これにより、ファイバーチャネルボリュームが破損しなくなります。(BZ#1954509)

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、開発者モードでコンソール UI の CNV namespace 内のカスタムリソースを削除しようとする場合、**Delete** をクリックすると、**Delete** ボタンがスタック状態でハングしていました。さらに、CLI で同じアクションを実行する際に表示されるエラーメッセージが表示されませんでした。今回の更新により、エラーメッセージが予想通りに表示され、**Delete** ボタンはスタックしなくなりました。(BZ#1939753)

- 以前のバージョンでは、OperatorHub プロバイダータイプ **filter** プロパティーには、**CatalogSource** との関係が明確に表示されませんでした。この問題により、ユーザーは **filter** の基準の意味を理解できませんでした。このパッチにより、プロバイダータイプ **filter** は **Source** へ更新されます。これにより、**filter** と **CatalogSource** の関係がより明確に表示されます。(BZ#1919406)

- 以前のバージョンでは、**Resources** メニューの **ResourceListDropdown** コンポーネントは、一部の言語では国際化されていませんでした。今回の更新により、**Resources** メニューが更新され、英語以外のスピーカーのユーザーエクスペリエンスが改善されました。(BZ#1921267)

- 以前のバージョンでは、**Delete Persistent Volume Claim**などの一部のメニュー項目は、正しく国際化されていませんでした。現在は、より多くのメニュー項目が正しく国際化されています。(BZ#1926126)

- 以前のバージョンでは、**Add HorizontalPodAutoscaler** ページのテキストおよび警告メッセージの一部が国際化されていませんでした。現在、テキストは国際化されています。(BZ#1926131)

- 以前のバージョンでは、ユーザーが Operator SDK で Operator を作成し、**xDescriptors={"urn:alm:<...>:hidden"}** などのアノテーションを指定して、Operator インスタンス作成ページからフィールドを非表示にすると、フィールドがページに表示されたままになる場合があり

ました。現在、非表示フィールドは、Operator インスタンス作成ページから省略されています。(BZ#1966077)

- 以前のバージョンでは、モバイルデバイスでテーブルが正しく表示されませんでした。今回の更新で、テーブルが正しく表示されるようになりました。(BZ#1927013)
- 以前のバージョンでは、OpenShift Container Platform Web コンソールの起動に時間がかかる場合がありました。今回の更新により、Web コンソールがより迅速に起動するようになりました。(BZ#1927310)
- 以前のバージョンでは、OpenShift Container Platform 管理者に対する国際化された通知がなかったため、ユーザーエクスペリエンスが損なわれていました。現在は、国際化が可能になりました。(BZ#1927898)
- 以前のバージョンでは、**Cluster Utilization** ダッシュボードで国際化された期間がなかったため、ユーザーエクスペリエンスが損なわれていました。現在は、国際化が可能になりました。(BZ#1927902)
- 以前のバージョンでは、OpenShift Container Platform Web コンソールの Operator Lifecycle Manager (OLM) ステータス記述子に互換性のないデータタイプが割り当てられるとエラーが生じました。検証が追加され、互換性のないデータタイプが処理から排除され、エラーが回避されました。ログに記録された警告は、互換性のないステータスタイプも識別します。(BZ#1927941)
- 以下の OpenShift Container Platform Web コンソールビューは、マルチファセットフィルタリングをサポートするようになりました。
  - **Home → Search** (**Resources** タブ)
  - **Home → Events** (**Resources** タブ)
  - **Workloads → Pods** (**Filter** タブ)

  詳細は、BZ#1930007 を参照してください。
- 以下のバグ修正は、OpenShift Container Platform Web コンソールのさまざまな翻訳の問題に対応します。
  - BZ#1921780
  - BZ#1921781
  - BZ#1922992
  - BZ#1924585
  - BZ#1924747
  - BZ#1925083
- 以前のバージョンでは、Web コンソールはハードコーディングされたチャネル文字列に依存して、チャネルモーダルドロップダウンを設定していました。その結果、ユーザーには、現在のバージョンに対して正しくない可能性のあるチャネル値が表示されていました。Cluster Version Operator が特定バージョンの正しいチャネルを提供しない場合は、チャネルモーダルドロップダウンはテキスト入力フィールドに変わり、ユーザーにチャネルとヘルプテキストを提案するようになりました。コンソールはハードコーディングされたチャネル文字列に依存しなくなりました。(BZ#1932281)

- 以前のバージョンでは、タイムスタンプは中国語または日本語用に正しくフォーマットされていませんでした。その結果、タイムスタンプが読みにくく、ユーザーエクスペリエンスが低下していました。今回の更新で、中国語と日本語用にタイムスタンプ形式 **Moment.js** がデフォルトで使用され、ユーザーエクスペリエンスが向上します。(BZ#1932453)

- 以前のバージョンでは、FilterToolbar コンポーネントの **rowFilters** プロパティーは、**null** 値を許可しませんでした。このため、**rowFilters** プロパティーが定義されていない場合、キャッチされない例外がスローされました。FilterToolbar コンポーネントで **rowFilters** プロパティーが参照されると、**null** 値が許可されるようになりました。その結果、**rowFilters** プロパティーが定義されていない場合は、FilterToolbar は例外をスローしません。(BZ#1937018)

- 以前のバージョンでは、間違ったスタイルのヘルプテキストが、フィールドレベルのヘルプインスタンスに適用されていました。現在は、フィールドレベルのヘルプインスタンスに対して正しいスタイルのヘルプテキストが表示され、コンソール全体で一貫性が保たれています。(BZ#1942749).

- 以前のバージョンでは、Operator Lifecycle Manamgent (OLM) ステータス条件記述子は、リソースの詳細ページで通常の詳細項目としてレンダリングされていました。その結果、**Conditions** テーブルは半分の幅でレンダリングされていました。今回の更新により、条件記述子は、オペランド の詳細ページの通常の **Conditions** テーブルの下にある全幅テーブルとして、レンダリングされるようになりました。(BZ#1943238)

- 以前のバージョンでは、「Ingresses」という単語は中国語ユーザーに翻訳されていましたが、ユーザーエクスペリエンスは良くありませんでした。現在は、「Ingress」という単語は翻訳されていません。(BZ#1945816)

- 以前のバージョンでは、「Operators」という単語は中国語ユーザーに翻訳されていましたが、複数形での翻訳は、ユーザーエクスペリエンスを悪くしていました。現在は、「Operators」という単語は翻訳されていません。(BZ#1945818)

- 以前のバージョンでは、コードが正しくないと、**User** および **Group** の詳細に、関係のないサブジェクトが表示されていました。現在は、**User** または **Group** でフィルタリングするコードが追加され、**User** および **Group** の詳細には関連するサブジェクトが表示されるようになりました。(BZ#1951212)

- 以前のバージョンでは、Pod コンテナーのテキストは国際化されていなかったため、ユーザーエクスペリエンスが低下していました。Pod コンテナーのテキストが国際化されたため、ユーザーエクスペリエンスが改善されました。(BZ#1937102)

- 以前のバージョンでは、**PackageManifest** リストページのアイテムは、詳細ページにリンクされていなかったため、ユーザーはリストページから個別の **PackageManifest** アイテムに簡単にドリルダウンできませんでした。**PackageManifest** アイテムはそれぞれ、他のリストページの規則に一致する詳細ページにリンクされるようになりました。ユーザーは、リストページから PackageManifest の詳細ページに簡単にアクセスできます。(BZ#1938321)

- 成功した 完了数の代わりに、必要な 完了数によってソートされた ジョブ テーブルの 完了 列。データは **# Succeeded of # Desired** と表示されるため、その列でソートすると、データが 2 番目の数値でソートされるため、わかりにくい結果となっていました。ジョブ 完了 列は、わかりやすくするために **# Succeeded** でソートされるようになりました。(BZ#1902003)

- **Manage Columns** モーダルの入力ラベルは、クリック可能なボタンではなかったため、クリックして列を管理することはできませんでした。このバグ修正により、ラベルは列の管理に使用できるクリック可能なボタンになりました。(BZ#1908343)

- CSI プロビジョナーは、Google Cloud Platform でストレージクラスを作成する際に一覧表示されませんでした。今回のバグ修正により、この問題は解決されています。(BZ#1910500)

- 以前のバージョンでは、ユーザーが **User Management → Roles** リストビューから **Cluster Role** をクリックする場合、詳細ページのバックリンクは、**Cluster Roles** であり、**Cluster Roles** の一般的なリストビューを提供していました。これにより、Web コンソールの後方ナビゲーションが誤ったページにリダイレクトされていました。このリリースでは、バックリンクにより、ユーザーは **Cluster Role/RoleBinding** の詳細ページから **Role/Bindings** リストビューに移動できます。これにより、ユーザーは Web コンソールで逆方向に適切にナビゲートできます。(BZ#1915971)

- 以前のバージョンでは、**Created date time** はユーザーが読み取れる形式で表示されませんでした。そのため、UTC で表示される時刻を理解して使用することが困難でした。今回のリリースにより、表示される時刻が再フォーマットされ、UTC が読み取り可能になり、理解できるようになりました。(BZ#1917241)

- 以前のバージョンでは、Web コンソールでの Pod 要求および制限の計算は正しくありませんでした。これは、完了した Pod または init コンテナーを除外しないために生じていました。今回のリリースにより、計算で必要のない Pod が除外され、Pod 要求の Web コンソール計算の結果の精度が改善されました。(BZ#1918785)

- 以前のバージョンでは、未定義の値を解析すると、Not a Number (NaN) 例外となり、**Chart** のツールチップのボックスには値が表示されませんでした。今回のリリースにより、データの取得時に開始日が指定され、**Chart** ツールチップに正しい値が表示されるようになりました。この変更により、結果が同期され、未定義の値が解析されなくなります。(BZ#1906304)

- 以前のバグ修正中に、Pod ログのダウンロードリンクが、空のダウンロード属性を持つ標準の HTML アンカー要素に変更されました。その結果、ダウンロードファイルはデフォルトのファイル名形式を失いました。この更新により、アンカー要素のダウンロード属性にファイル名が追加され、Pod ログのダウンロード時に **<pod-name>-<container-name>.log** 形式のデフォルトのファイル名が使用されるようになりました。(BZ#1945630)

- 以前のバージョンでは、ユーザーにリソースの作成パーミッションがあるものの、編集パーミッションがない場合は、Web コンソールの YAML エディターは誤って読み取り専用モードに設定されていました。エディターのコンテンツは、リソースの作成アクセスのあるユーザーが編集できるようになりました。(BZ#1824911)

- 以前のバージョンでは、Web コンソールはほとんどの場合に 12 時間形式で時間を表示し、24 時間形式で表示する場合もありました。また、過去の 1 年を上回る日数について年が表示されませんでした。今回のリリースにより、日付と時間が一貫性のある方法でフォーマットされ、ユーザーのロケールと言語設定と一致するようになり、過去の 1 年を上回る日数について年が表示されるようになりました。(BZ#1862084)

- 以前のバージョンでは、Web コンソールは、イベントを表示する権限を持たないユーザーの **ClusterVersion** リソースをポーリングしていました。これにより、コンソール Pod ログに多数のエラーが出力されます。これを回避するには、リソースをポーリングする前にユーザーのパーミッションを確認する必要があります。これにより、コンソール Pod ログの不要なエラーが排除されます。(BZ#1848151)

- 以前のバージョンでは、YAML エディターのキーボードユーザーは、エディターを終了できませんでした。エディター外の **view shortcuts** ポップオーバーへは、ユーザーはエディター内からアクセスできませんでした。この更新により、ユーザーは **opt + F1** キーストロークを使用して、エディターの上に**Accessibility help** を表示できます。この変更により、YAML エディターのキーボードユーザーは、正しいキーストロークを使用してエディターを終了できます。(BZ#1874931)

- OpenShift Container Platform (OCP) の 4.x リリース後、OCP 4 Web コンソールにアップロードされたバイナリーシークレットファイルをロードできませんでした。これにより、インストールが失敗していました。OpenShift Container Platform 4.8 では、この機能が OCP 4 Web

コンソールに復元されました。必要なシークレットの入力は、バイナリーファイル形式を使用して実行できるようになりました。(BZ#1879638)

- 以前のバージョンでは、RoleBinding のリンクを適切に作成するための BZ#1871996 の修正により、namespace が選択された際のバインディングタイプの選択ができなくなりました。その結果、アクティブな namespace を持つユーザーは、アクティブな namespace を **All namespaces** に変更しないと、クラスター RoleBinding を作成できませんでした。この更新により、BZ#1871996 への変更の一部が元に戻され、ユーザーはアクティブな namespace に関係なくクラスターロールバインディングを作成できます。(BZ#1927882)

**Web コンソール (Developer パースペクティブ)**

- 以前のバージョンでは、開発者コンソールでサービスクラスターをローカルにするためにラベルが変更された場合、ユーザーは Knative サービスを作成できませんでした。Knative サービスの更新では、ユーザーが開発者コンソールから cluster-local として Knative サービスを作成できるようにするために、**cluster-local** でサポートされている最新のラベルを使用します。(BZ#1969951)

- 以前のバージョンでは、Image Manifest Vulnerabilities (IMV) の重大度が 低 および 中 の問題の色が (Quay.io) インターフェースに表示される色と一致していませんでした。その結果、ユーザーが脆弱性の重大度を 高 に変更すると、IMV は誤った重大度を付けていました。この結果、IMV を確認する際に混乱が生じました。現在のリリースではこの問題は修正されています。(BZ#1942716)

- 以前のバージョンでは、Samples Operator がインストールされていないために OpenShift namespace テンプレートを使用できない場合は、Developer パースペクティブの トポロジービューがロードされませんでした。今回の更新でこの問題が修正されています。(BZ#1949810)

- 以前のバージョンでは、devfile をインポートする際に、Web コンソールは、環境変数、ポートおよび制限の設定を提供する **build guidance** プレースホルダーコンテナーを無視していました。新規のデプロイメントには、プレースホルダーイメージを取得できず、必要な設定が欠落しているために、起動できない 2 つ目のコンテナーが含まれました。これで、**build guidance** コンテナーが新規のデプロイメントから削除され、コンテナーは環境変数、ポート、および制限設定を追加します。(BZ#1952214)

- 以前のバージョンでは、別のタブで **Developer** パースペクティブを切り替え、プロジェクトの詳細を再読み込みする際に、パースペクティブに関連するルートはレンダリングされず、**404** エラーが発生しました。今回の更新では、すべての非アクティブなルートが読み込まれ、正しいパースペクティブに切り替わるようになりました。(BZ#1929769)

- 以前のバージョンでは、ユーザーが namespace に必要なアクセス権を持たないためにエラーが発生すると、**Monitoring** ダッシュボードページの **Workload** ドロップダウンメニューに継続的にロード進行中のアイコンが表示されていました。現在のリリースではこの問題は修正されています。**Monitoring** ダッシュボードページには、**Forbidden** エラーが発生したことを示すエラーメッセージが表示されます。(BZ#1930546)

- 以前のバージョンでは、API サーバーはリソースの作成に失敗し、これにより、**resource quota** リソースの更新時に競合が発生すると、409 ステータスコードを返す可能性がありました。そのため、リソースは作成に失敗し、API 要求を再試行する必要があった可能性があります。今回の更新により、**OpenShift Console** Web アプリケーションは 409 のステータスコードを受信する際に要求を 3 回試行します。多くの場合、この回数は要求を実行するのに十分な回数です。409 ステータスコードが継続的に発生する場合、コンソールにエラーが表示されます。(BZ#1920699)

- 以前のバージョンでは、**YAML** タブを選択しても、**metadata.managedFields** セクションはすぐに折りたたまれませんでした。これは、**Pipeline Builder** および **Edit HorizontalPodAutoscaler** (HPA) などのページの **Form** または **YAML** スイッチャーの問題が原

因でした。その結果、入力しようとしたドキュメントの部分が折りたたまれました。**metadata.managedFields** セクションはそのままとなり、カーソルは YAML エディターの左上の開始位置にリセットされました。現在のリリースではこの問題は修正されています。現在は、YAML をロードすると、**metadata.managedFields** セクションがすぐに折りたたまれます。(BZ#1932472)

- 以前のバージョンでは、プライベートリポジトリーの **Git Import** フローで作成されたパイプラインは実行できませんでした。これは、パイプライン **ServiceAccount** オブジェクトが、プライベート Git リポジトリーの **Git** インポートフローで作成されたシークレットを使用しないために生じました。今回の更新により、シークレット名をパイプライン **ServiceAccount** オブジェクトのアノテーションに追加し、パイプライン固有のアノテーションを指定されるシークレットに追加できるようになりました。その結果、プライベート Git リポジトリーのパイプラインは正常に実行されます。(BZ#1970470)

- 以前のバージョンでは、ユーザーが YAML エディターにフォーマットされた YAML スニペットを挿入すると、新しい選択がスニペットの新しいコンテンツと一致しませんでした。インデントが削除され、選択範囲にランダムな文字がいくつか表示されました。現在のリリースではこの問題は修正されています。これで、カーソルは開始位置に留まり、カーソルの終了位置に欠落しているインデントが追加されます。YAML スニペットの挿入後に、新しい選択が新しいコンテンツと一致します。(BZ#1952545)

- 以前のバージョンでは、アノテーションは Knative サービスの仕様およびメタデータに渡されていました。その結果、デコレーターは **Topology** の Knative サービスの関連するリビジョンに表示されました。このリリースでは、アノテーションを Knative サービスメタデータにのみ渡すことで、この問題を修正しています。現在、デコレーターは **Topology** の Knative サービスに対してのみ表示され、関連するリビジョンには表示されません。(BZ#1954959)

- 以前のバージョンでは、" などの空の文字列を持つパラメーターを使用してパイプラインを作成した場合、OpenShift Container Platform Web コンソールのフィールドは空の文字列を受け入れませんでした。現在のリリースではこの問題は修正されています。現在は、" は、パラメーターセクション内の有効なデフォルトプロパティーとしてサポートされています。(BZ#1951043)

- 以前のバージョンでは、ユーザーは **Developer** パースペクティブからプライベートサービスとして Knative サービスを作成できませんでした。この問題は、「**networking.knative.dev/visibility': 'cluster-local**」 ラベルを更新して修正されています。(BZ#1970796)

- 以前のバージョンでは、シンクタイプの Kamelets は、ソースタイプと共にイベントソースのカタログに表示されていました。この問題は、Kameletsのリソースをフィルタリングして、ソースタイプのみをリストアップすることで修正されました。(BZ#1972258)

### Windows Containers

- 以前のバージョンでは、ユーザーが追加の Windows ノードをスケーリングすると、ロードバランサーサービスが不安定になりました。この更新により、ロードバランサーサービスが安定化され、ユーザーはパフォーマンスを不安定にすることなく複数の Windows ノードを追加できるようになります。(BZ#1905950)

- 以前のバージョンでは、ロードバランサーが作成され、それが Windows Pod の実行後に作成された場合は、**kube-proxy** サービスが予期せずクラッシュしていました。この更新により、kube-proxy サービスは、ロードバランサーサービスを再作成する際にクラッシュしなくなりました。(BZ#1939968)

- 以前のバージョンでは、ロードバランサーの Ingress の空の IP アドレス値が、データパスを壊していました。その結果、Windows サービスにアクセスできませんでした。今回の更新により、IP アドレスの値が空の場合でも、Windows サービスにアクセスできるようになります。

(BZ#1952914)

- 以前のバージョンでは、ユーザーが Projected ボリュームで Windows Pod を作成した場合、Pod は **ContainerCreating** フェーズでスタックしたままでした。この更新により、Windows Pod の作成は正常に **Running** フェーズに進みます。(BZ#1973580)

## 1.8. 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
- **DEP**: 非推奨機能

表1.2 テクノロジープレビュートラッカー

| 機能 | OCP 4.6 | OCP 4.7 | OCP 4.8 |
|---|---|---|---|
| Precision Time Protocol (PTP) | TP | TP | TP |
| **oc** CLI プラグイン | TP | TP | GA |
| Descheduler | TP | GA | GA |
| OVN-Kubernetes Pod ネットワークプロバイダー | GA | GA | GA |
| メモリー使用率のための HPA | TP | GA | GA |
| サービスバインディング | TP | TP | TP |
| ログ転送 | GA | GA | GA |
| ユーザー定義プロジェクトのモニタリング | GA | GA | GA |
| Cinder での raw ブロック | TP | TP | GA |
| CSI ボリュームスナップショット | TP | GA | GA |
| CSI ボリュームのクローン作成 | GA | GA | GA |
| CSI ボリューム拡張 | TP | TP | TP |

| 機能 | OCP 4.6 | OCP 4.7 | OCP 4.8 |
|---|---|---|---|
| vSphere Problem Detector Operator | - | GA | GA |
| CSI Azure Disk Driver Operator | - | - | TP |
| CSI GCP PD Driver Operator | - | TP | GA |
| CSI OpenStack Cinder Driver Operator | - | TP | TP |
| CSI AWS EBS Driver Operator | TP | TP | TP |
| CSI の自動移行 | - | - | TP |
| Red Hat Virtualization (oVirt) CSI ドライバー Operator | GA | GA | GA |
| CSI インラインの一時ボリューム | TP | TP | TP |
| CSI vSphere Driver Operator | - | - | TP |
| Local Storage Operator を使用した自動デバイス検出およびプロビジョニング | TP | TP | TP |
| OpenShift Pipeline | TP | GA | GA |
| OpenShift GitOps | TP | GA | GA |
| OpenShift サンドボックスコンテナー | - | - | TP |
| Vertical Pod Autoscaler | TP | TP | GA |
| Cron ジョブ | TP | TP | GA |
| PodDisruptionBudget | TP | TP | GA |
| Operator API | GA | GA | GA |
| kvc を使用したノードへのカーネルモジュールの追加 | TP | TP | TP |
| egress ルーター CNI プラグイン | - | TP | GA |
| スケジューラーのプロファイル | - | TP | TP |
| プリエンプションを実行しない優先順位クラス | - | TP | TP |
| Kubernetes NMState Operator | - | TP | TP |

| 機能 | OCP 4.6 | OCP 4.7 | OCP 4.8 |
|---|---|---|---|
| 支援付きインストーラー | - | TP | TP |
| AWS Security Token Service (STS) | - | TP | GA |
| Kdump | - | TP | TP |
| OpenShift Serverless | - | - | GA |
| Serverless functions | - | - | TP |
| Jenkins Operator | TP | TP | DEP |
| ドライバツールキット | - | - | TP |

## 1.9. 既知の問題

- OpenShift Container Platform 4.1 では、匿名ユーザーは検出エンドポイントにアクセスできました。後のリリースでは、一部の検出エンドポイントは集約された API サーバーに転送されるため、このアクセスを無効にして、セキュリティーの脆弱性の可能性を減らすことができます。ただし、既存のユースケースに支障がで出ないように、認証されていないアクセスはアップグレードされたクラスターで保持されます。
  OpenShift Container Platform 4.1 から 4.8 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)

- **oc annotate** コマンドは、等号 (**=**) が含まれる LDAP グループ名では機能しません。これは、コマンドがアノテーション名と値の間に等号を区切り文字として使用するためです。回避策として、**oc patch** または **oc edit** を使用してアノテーションを追加します。(BZ#1917280)
- 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で vSphere 上の仮想マシンの電源をオンにすると、ノードのスケールアッププロセスは予想通りに機能しない可能性があります。ハイパーバイザー設定の既知の問題により、ハイパーバイザー内にマシンが作成されますが、電源がオンになりません。マシンセットをスケールアップした後にノードが **Provisioning** 状態のままである場合、vSphere インスタンス自体で仮想マシンのステータスを調査できます。VMware コマンド **govc tasks** および **govc events** を使用して、仮想マシンのステータスを判別します。以下のようなエラーメッセージがあるかどうかを確認します。

```
Invalid memory setting: memory reservation (sched.mem.min) should be equal to
memsize(8192).
```

この VMware KBase の記事 にある手順に従って、問題の解決を試行できます。詳細は、Red Hat ナレッジベースのソリューションの「UPI vSphere Node scale-up doesn't work as expected」を参照してください。(BZ#1918383)

- ECKD タイプの DASD を VirtIO ブロックデバイスとして使用すると、IBM Z での RHEL KVM インストールへの RHCOS のインストールに失敗します。(BZ#1960485)
- Open Virtual Network (OVN) のバグにより、Octavia ロードバランサーで永続的な接続の問題が発生します。Octavia ロードバランサーが作成されると、OVN はそれらを一部の Neutron サブネットにプラグインしない可能性があります。これらのロードバランサーは、Neutron サブネットの一部では到達できなくなる可能性があります。この問題は、Kuryr の設定時に各 OpenShift namespace に作成される Neutron サブネットに影響を与えます。その結果、この問題が発生すると、OpenShift **Service** オブジェクトを実装するロードバランサーが問題の影響を受ける OpenShift namespace から到達できなくなります。このバグにより、Kuryr SDN を使用する OpenShift Container Platform 4.8 デプロイメントの使用は、バグが修正されるまで OVN および OVN Octavia が設定された Red Hat OpenStack Platform (RHOSP) 16.1 では推奨されません。(BZ#1937392)
- Console Operator は、コンソールのルート (**console** または **downloads**) のいずれかの **componentRoutes** 条件で **Ingress** リソースを適切に更新しません。(BZ#1954148)

- OpenShift サンドボックスコンテナーを使用している場合、OpenShift Container Platformクラスターの **hostPath** ボリュームを使用して、ホストノードのファイルシステムから Pod にファイルまたはディレクトリーをマウントすることはできません。(BZ#1904609)

- OpenShift サンドボックスコンテナーで Fedora を実行している場合は、いくつかのパッケージをインストールするための回避策が必要です。**iputils** などの一部のパッケージでは、OpenShift Container Platform がデフォルトでコンテナーに付与しないファイルアクセス許可の変更が必要です。このような特別なアクセス許可を必要とするコンテナーを実行するには、ワークロードを説明するアノテーションを YAML ファイルに追加する必要があります。これは、そのワークロードに対してこのようなファイルのアクセス許可を受け入れるように **virtiofsd** に指示します。必要なアノテーションは以下のとおりです。

  ```
  io.katacontainers.config.hypervisor.virtio_fs_extra_args: [ "-o", "modcaps=+sys_admin", "-o", "xattr" ]
  ```

  (BZ#1915377)

- 4.8 リリースでは、OpenShift Container Platform Webコンソールを使用して **kataConfgPoolSelector** に値を追加すると、**scheduling.nodeSelector** が空の値で設定される原因となります。**kata** の値で **RuntimeClass** を使用する Pod は、Kata コンテナーランタイムがインストールされていないノードにスケジュールされる場合があります。
  この問題を回避するには、以下のコマンドを実行して、**RuntimeClass kata** に手動で **nodeSelector** 値を指定します。

  ```
  $ oc edit runtimeclass kata
  ```

  以下は、正しい **nodeSelector** ステートメントを持つ **RuntimeClass** の例です。

  ```
  apiVersion: node.k8s.io/v1
  handler: kata
  kind: RuntimeClass
  metadata:
    creationTimestamp: "2021-06-14T12:54:19Z"
    name: kata
  overhead:
    podFixed:
      cpu: 250m
      memory: 350Mi
  scheduling:
    nodeSelector:
      custom-kata-pool: "true"
  ```

  (KATA-764)

- Operator Hub の OpenShift サンドボックスコンテナー Operator の詳細ページでは、いくつかのフィールドがありません。フィールドがない場合でも、4.8 で OpenShift サンドボックスコンテナー Operator をインストールできます。
  (KATA-826)

- 複数の **KataConfig** カスタムリソースを作成すると、警告なしで失敗します。OpenShift Container Platform Web コンソールは、複数のカスタムリソースの作成が失敗したことをユーザーに通知するプロンプトを提供しません。
  (KATA-725)

- OpenShift Container Platform Web コンソールのOperator Hub に、Operator のアイコンが表示されない場合があります。
(KATA-804)

- OVN-Kubernetes ネットワークプロバイダーは、**NodePort** タイプサービスおよび **LoadBalancer** タイプサービスの **externalTrafficPolicy** 機能をサポートしていません。**service.spec.externalTrafficPolicy** フィールドは、サービスのトラフィックをノードローカルまたはクラスター全体のエンドポイントにルーティングするかどうかを決定します。現在、このようなトラフィックはデフォルトでクラスター全体のエンドポイントにルーティングされており、トラフィックをノードローカルエンドポイントに制限する方法はありません。この問題は、今後のリリースで解決される予定です。(BZ#1903408)

- 現在、Kubernetes ポートの衝突の問題により、Pod が再デプロイされた後でも、Pod 間の通信が機能しなくなる可能性があります。詳細および回避策については、Red Hat ナレッジベースソリューションの「Port collisions between pod and cluster IPs on OpenShift 4 with OVN-Kubernetes」を参照してください。(BZ#1939676、BZ#1939045)

- OVN-Kubernetes ネットワークプロバイダーを使用し、コンピューティングノードが RHEL 7.9 を実行するクラスターの場合、OpenShift Container Platform 4.7 から OpenShift Container Platform 4.8 へのアップグレードは BZ#1976232 によってブロックされます。リリース 4.8 にアップグレードするには、このバグの修正が含まれる 4.8 パッチを待つ必要があります。
(BZ#1976232)

- OVN-Kubernetes ネットワークプロバイダーを使用し、OpenShift Container Platform 4.7 から OpenShift Container Platform 4.8 へアップグレードするクラスターの場合、OVN-Kubernetes のバグが原因で、Pod IP アドレスが古くなることがあります。このバグは、めったに発生しない競合状態です。その結果、4.8 リリースへのアップグレード中に、ノードはドレインに失敗し、一部の Operator は **Degraded** のステータスを報告します。回避策として、**CrashLoopBackOff** 状態のままでアップグレードを完了しなかった Pod を特定します。**oc delete <pod-name>** コマンドで各 Pod を削除します。(BZ#1974403)

- **kubeletconfig** リソースの **tlsSecurityProfile** フィールドの説明 (例: **oc explain** コマンドを使用する場合など) には、TLS セキュリティープロファイルの正しい暗号が記載されていません。回避策として、影響を受けるノードの **/etc/kubernetes/kubelet.conf** ファイルで暗号の一覧を確認してください。(BZ#1971899)

- 単一ノードで通常モードで CNF テストを実行する場合、クラスターの準備ができているかどうかを理解するためのロジックに詳細情報がありません。具体的には、SR-IOV ネットワークを作成しても、少なくとも 1 分経過するまで、ネットワーク接続定義は作成されません。すべての DPDK テストはカスケードで失敗します。**-ginkgo.skip** パラメーターを使用して、単一ノードのインストールに対して実行する場合は、DPDK 機能をスキップして通常モードで CNF テストを実行します。ディスカバリーモードで CNF テストを実行して、単一ノードのインストールに対してテストを実行します。(BZ#1970409)

- 現在、CNF テストは、SR-IOV および DPDK テスト用の MLX NIC を使用したセキュアブートをサポートしていません。**-ginkgo.skip** パラメーターを使用して、通常モードでセキュアブートが有効な環境に対して実行する場合、SR-IOV 機能をスキップして CNF テストを実行できます。検出モードで実行することは、MLX カードを使用してセキュアブートが有効な環境に対してテストを実行する際の推奨される方法です。この問題は、今後のリリースで解決される予定です。(BZ#1975708)

- **ArgoCD** Operator がサブスクライブされ、ArgoCD と AppProject が開始されると、より制限の厳しい OpenShift Container Platform 環境でイメージが機能しないため、**guestbook** という名前のサンプルアプリケーションの起動に失敗します。一時的な回避策として、ユーザーは以下の例をデプロイすることで、**ArgoCD** Operator が正しく機能することを確認できます。

```
PROJ=younamespace
```

```
cat > $PROJ-app.yaml <<EOF
apiVersion: argoproj.io/v1alpha1
kind: Application
metadata:
  name: simple-restricted-webserver
  namespace: $PROJ
spec:
  destination:
    namespace: $PROJ
    server: https://kubernetes.default.svc
  project: default
  source:
    path: basic-nginx
    repoURL: 'https://github.com/opdev/argocd-example-restricted-apps.git'
    targetRevision: HEAD
EOF
oc create -f $PROJ-app.yaml
```

詳細は、BZ#1812212 を参照してください。

- 複数のタブでコンソールを開いている場合、**Developer** パースペクティブの一部のサイドバーのリンクがプロジェクトに直接リンクされず、選択されたプロジェクトで予期しない変更が生じます。この問題は、今後のリリースで解決される予定です。(BZ#1839101)

- **pathType: Prefix** を使用すると、Ingress を使用したパススルールートの作成が失敗します。代わりに、**pathType** を **ImplementationSpecific** に設定し、**path** を **''** に設定することで、パススルールートを作成できます。

  **Ingress YAML ファイルのサンプル**

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: ingress7
  namespace: test-ingress
  annotations:
    route.openshift.io/termination: passthrough
spec:
  rules:
  - host: <ingress-psql-example-test-ingress.apps>
    http:
      paths:
      - path: ''
        pathType: ImplementationSpecific
        backend:
          service:
            name: <ingress-psql-example>
            port:
              number: 8080
```

詳細は、BZ#1878685 を参照してください。

- 現時点では、**Search** ページの **Pipelines** リソーステーブルは、**Name** フィルターを適用または削除した直後に更新されません。ただし、ページを更新して **Pipelines** セクションを展開すると、**Name** フィルターが適用されます。**Name** フィルターを削除すると同じ動作が確認されます。この問題は、今後のリリースで解決される予定です。(BZ#1901207)

- ドキュメントには、**Provisioning** カスタムリソースの **ProvisioningNetworkCIDR** 値が記載されています。これにより、IPv6 プロビジョニングネットワークが **dnsmasq** によって /64 に制限されます。(BZ#1947293)

- トラブルシューティングをサポートするために、インストーラーによってブートストラップの失敗で収集されたログに、コントロールプレーンとブートストラップホストの IP アドレスとルートが含まれるようになりました。(BZ#1956079)

- 自己署名の Amazon Commercial Cloud Services 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、S3 ストレージから直接ではなく、イメージレジストリーからイメージレイヤーをプルできます。(BZ#1924568)

- 以前のバージョンでは、OpenShift Container Platform Web コンソールで Bitbucket リポジトリーを使用してデプロイメント用に作成されたトポロジー URL は、スラッシュ文字を含むブランチ名が含まれている場合は機能しませんでした。これは Bitbucket API (BCLOUD-9969) の問題が原因でした。現在のリリースではこの問題は軽減されています。ブランチ名にスラッシュが含まれている場合、トポロジー URL はリポジトリーのデフォルトのブランチページを指します。この問題は OpenShift Container Platform の今後のリリースで修正されます。(BZ#1969535)

- OpenShift Container Platform (OCP) バージョン 4.6 を Red Hat Virtualization (RHV) にインストールするには、RHV バージョン 4.4 が必要です。RHV 4.3 で以前のバージョンの OCP を実行している場合は、これを OCP バージョン 4.6 に更新しないでください。Red Hat は、RHV バージョン 4.3 での OCP バージョン 4.6 の実行をテストしていないため、この組み合わせをサポートしません。テスト済み統合の詳細は、「OpenShift Container Platform 4.x Tested Integrations (for x86_x64)」を参照してください。

- **operator-sdk pkgman-to-bundle** コマンドは、**--build-cmd** フラグを指定して実行するとエラーを出して終了します。詳細は、(BZ#1967369) を参照してください。

- 現時点で、Web コンソールのクイックスタートカードの前提条件は、一覧ではなく段落で表示されます。この問題は、今後のリリースで解決される予定です。(BZ#1905147)

- OpenShift Container Platformのシングルノード構成では、リアルタイムカーネル（kernel-rt）を使用した場合、非リアルタイムカーネルを使用した場合に比べてPodの作成時間が2倍以上遅くなります。kernel-rtを使用した場合、ノードの再起動後のリカバリータイムが影響を受けるため、作成時間が遅いことで、サポートされるPodの最大数に影響が出ます。kernel-rtを使用している場合の回避策として、**rcupdate.rcu_normal_after_boot=0**のカーネル引数を指定して起動することで、影響を受けた回復時間を改善することができます。この場合、リアルタイムカーネルのバージョンは、**kernel-rt-4.18.0-305.16.1.rt7.88.el8_4**以降でなければなりません。この既知の問題は、OpenShift Container Platformのバージョン4.8.15以降に該当します。(BZ#1975356)

- OpenShift Container Platformのシングルノードのリブートに続いて、すべての Pod が再起動します。これにより、大きな負荷が発生し、通常の Pod 作成時間が長くなります。これは、Container Network Interface (CNI)が **pod add** イベントを素早く処理できないために発生します。**timed out waiting for OVS port binding** エラーメッセージが表示されます。OpenShift Container Platform の単一ノードインスタンスは最終的は復帰しますが、想定よりも遅くなります。この既知の問題は、OpenShift Container Platformのバージョン4.8.15以降に該当します。(BZ#1986216)

- OpenShift Container Platform 4.8 以前のデフォルトのロードバランシングアルゴリズムは**leastconn** でした。パススルーでないルートの場合、OpenShift Container Platform 4.8.0ではデフォルトが**random**に変更されました。**random**に切り替えると、長時間のウェブソケット接続を使用する必要がある環境では、メモリ消費量が大幅に増加するため、互換性がありませ

ん。この大幅なメモリ消費を軽減するために、OpenShift Container Platform 4.8では、デフォルトのロードバランシングアルゴリズムが**leastconn**に戻されました。大幅なメモリ使用量を発生させないソリューションが開発されれば、OpenShift Container Platformの将来のリリースでデフォルトが**random**に変更される予定です。
以下のコマンドを入力することで、デフォルトの設定を確認することができます。

```
$ oc get deployment -n openshift-ingress router-default -o yaml | grep -A 2
ROUTER_LOAD_BALANCE_ALGORITHM
      - name: ROUTER_LOAD_BALANCE_ALGORITHM
        value: leastconn
```

**random**のオプションはまだ利用可能です。しかし、このアルゴリズムの選択の恩恵を受けたいルートは、以下のコマンドを入力して、ルートごとにアノテーションでそのオプションを明示的に設定する必要があります。

```
$ oc annotate -n <NAMESPACE> route/<ROUTE-NAME>
"haproxy.router.openshift.io/balance=random"
```

(BZ#2017708)

## 1.10. エラータの非同期更新

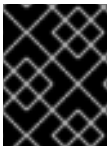
OpenShift Container Platform 4.8 のセキュリティー、バグ修正、拡張機能の更新は、Red Hat Network 経由で非同期エラータとして発表されます。OpenShift Container Platform 4.8 のすべてのエラータは Red Hat カスタマーポータルから入手できます。非同期エラータについては、OpenShift Container Platform ライフサイクルを参照してください。

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 4.8 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 4.8.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。

### 1.10.1. RHSA-2021:2438 - OpenShift Container Platform 4.8.2 イメージのリリース、バグ修正およびセキュリティー更新アドバイザリー

発行日: 2021-07-27

セキュリティー更新を含む OpenShift Container Platform リリース 4.8.2 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-2021:2438 アドバイザリーに一覧表示されます。この更新に含まれる RPM パッケージは、RHSA-2021:2437 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 4.8.2 コンテナーイメージの一覧

### 1.10.2. RHBA-2021:2896 - OpenShift Container Platform 4.8.3 バグ修正の更新

発行日: 2021-08-02

OpenShift Container Platform リリース 4.8.3 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-2021:2896 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-2021:2899 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 4.8.3 コンテナーイメージの一覧

#### 1.10.2.1. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.3. RHSA-2021:2983: OpenShift Container Platform 4.8.4 セキュリティーおよびバグ修正の更新

発行日: 2021-08-09

OpenShift Container Platform リリース 4.8.4 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、RHSA-2021:2983 アドバイザリーに一覧表示されます。この更新に含まれる RPM パッケージは、RHSA-2021:2984 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 4.8.4 コンテナーイメージの一覧

#### 1.10.3.1. バグ修正

- 以前は BZ#1954309 および BZ#1960446 は、OpenShift Container Platform 4.8.3 のリリースノートに修正済みのバグとして記載されていましたが、バージョン 4.8.3 のリリースから削除されました。このリリースでは、**BZ#1960446**のバグフィックスサマリーがOpenShift Container Platform 4.8.4のリリースノートの「バグ修正」セクションに移動し、**BZ#1954309**のバグフィックスサマリーが削除されています。

- これまでは、**nmstate-handler** Podの許容範囲の設定に誤りがあり、**nmstate** Operatorを搭載したノードのネットワーク設定ができませんでした。今回の更新では、ハンドラーPodがすべてのノードで許容範囲を許可しました。(BZ#1960446)

- これまでは、コピーに失敗した**ClusterServiceVersion**オブジェクト（CSV）に対して、Webコンソールに**The operator is running in openshift-operators but is managing this**

**namespace**が表示されていました。このメッセージは具体的ではなく、ユーザーが失敗したCSVをトラブルシューティングするのに役立ちませんでした。今回のリリースでは、コピーされたCSVのメッセージで、失敗の原因を探すために元のCSVに誘導し、元のCSVへのリンクを提供します。(BZ#1972478)

- 以前は、レジストリがカスタム許容値を使用するかどうかをチェックするOperatorは、**spec.tolerations**の代わりに**spec.nodeSelector**をチェックしていましたが、**spec.tolerations**のカスタム許容値は、**spec.nodeSelector**が設定されている場合にのみ適用されます。このリリースでは、**spec.tolerations**がチェックされ、**spec.tolerations**が設定されている場合は、Operatorはカスタムの許容値を使用します。(BZ#1973662)

- これまでは、イメージストリームと**image.openshift.io/triggers**アノテーションを使用せずにデプロイメントを作成すると、デプロイメントコントローラが無限ループでレプリカセットを作成していました。この問題は本リリースでは解決されています。(BZ#1981770)

- 今回のリリースでは、Manila CSIのログが**must-gather**負荷に追加されました。(BZ#1986026)

- 以前は、仮想マシンに自動ピン留めを使用すると、プロパティの名前は**disabled**、**existing**、または**adjust**されていました。このリリースでは、名前が各ポリシーをより適切に説明するようになり、**existing**はoVirtでブロックされているため削除されました。新しいプロパティ名は**none**と**resize_and_pin**で、oVirtユーザーインターフェイスと一致します。(BZ#1987182)

#### 1.10.3.2. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.4. RHBA-2021:3121 - OpenShift Container Platform 4.8.5 バグ修正の更新

発行日: 2021-08-16

OpenShift Container Platform リリース 4.8.5 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-2021:3121 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-2021:3122 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 4.8.5 コンテナーイメージの一覧

#### 1.10.4.1. 特長

##### 1.10.4.1.1. Egress IPの強化

新しい機能拡張により、OpenShift Container Platform 4.8 Anonymizerにegress IPアドレスのサポートが追加されました。詳細は、BZ#1974877 を参照してください。

#### 1.10.4.2. バグ修正

- これまでは、**JenkinsPipelineStrategy**が定義された**BuildConfig**オブジェクトに対して、**oc logs**が機能しませんでした。今回の更新では、パイプラインビルドで**oc logs**が動作するようになりました。(BZ#1974267)

- これまでは、仮想IP（VIP）を保持する**Keepalived**コンテナが**SIGTERMed**と表記されると、VRRPのプロアクティブメッセージが送信されませんでした。その結果、VIPはタイムアウトし

た後、別のノードに移行しました。今回の更新では、**SIGTERMed**と示されたVIPを保持する**Keepalived**コンテナが、**VRRP priority 0**のアドバタイズメントメッセージを送信します。その結果、今ではVIPの移行が早くなっています。(BZ#1920670)

- これまでは、**Kameletbinding**を使ってアクションとシンクのKameletを作成することができましたが、ソースタイプのKameletのみがリストアップされるべきでした。今回の更新により、シンクタイプとアクションタイプのKameletsを選択するオプションが削除されました。その結果、イベントソースのカタログに表示されるのはソースKameletsだけになりました。(BZ#1972258)

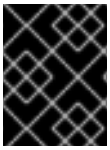
#### 1.10.4.3. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.5. RHBA-2021:3247: OpenShift Container Platform 4.8.9 セキュリティーおよびバグ修正の更新

発行日: 2021-08-31

OpenShift Container Platform リリース 4.8.9 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-2021:3247 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-2021:3248 アドバイザリーで提供されています。



**重要**

RHBA-2021:3247アドバイザリーに記載されているSHA-256イメージダイジェスト情報に誤りがあります。正しい情報は以下のとおりです。

リリースイメージのメタデータを確認するには、**oc**ツールをダウンロードし、以下のコマンドを実行します。

- x86_64アーキテクチャの場合:

```
$ oc adm release info quay.io/openshift-release-dev/ocp-release:4.8.9-x86_64
```

イメージダイジェストは、**sha256:5fb4b4225498912357294785b96cde6b185eaed20bbf7a4d008c462134a4edfd**です。

- s390xアーキテクチャの場合:

```
$ oc adm release info quay.io/openshift-release-dev/ocp-release:4.8.9-s390x
```

イメージダイジェストは**sha256:2665dcca917890b3d06c339bb03dac65b84485fef36c90f219f2773393ba291d**です。

- ppc64leアーキテクチャの場合:

```
$ oc adm release info quay.io/openshift-release-dev/ocp-release:4.8.9-ppc64le
```

イメージダイジェストは**sha256:ded5e8d61915f74d938668cf58cdc9f37eb4172bc24e80c16c7fe1a6f84eff43**です。

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 4.8.9 コンテナーイメージの一覧

#### 1.10.5.1. バグ修正

- 今回のリリースでは、中国語、日本語、韓国語のローカライズコンテンツが追加されています。(BZ#1972987)
- OpenShift Container Platform 4.5 の OVN-Kubernetes で使用されるアドレスセットの命名規則が OpenShift Container Platform 4.6 で変更されたにも拘らず、アップグレードの一部として、既存のアドレスセットから新規命名規則への移行は処理されませんでした。ingress または egress セクションの namespace セレクターの条件を使用してバージョン 4.5 で作成されたネットワークポリシーは、この namespace 内の Pod IP アドレスの最新の情報に更新されていない以前のアドレスセットに依存していました。これらのポリシーは 4.6 以降のリリースでは正しく機能せず、予期しないトラフィックを許可または拒否する可能性があります。
以前のバージョンでは、回避策はこれらのポリシーを削除し、再作成する必要がありました。今回のリリースにより、命名規則が古いアドレスセットが削除され、以前のアドレスセットを参照するポリシー ACL が OVN-Kubernetes のアップグレード時に新規命名規則に従ったアドレスセットを参照するように更新されました。バージョン 4.5 で作成した影響を受けるネットワークポリシーは、アップグレード後に再度機能するようになります。(BZ#1976241)

#### 1.10.5.2. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.6. RHBA-2021:3299 - OpenShift Container Platform 4.8.10 バグ修正の更新

発行日: 2021-09-06

OpenShift Container Platform リリース 4.8.10 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-2021:3299 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-2021:3300 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 4.8.10 コンテナーイメージの一覧

#### 1.10.6.1. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.7. RHBA-2021:3429 - OpenShift Container Platform 4.8.11 バグ修正の更新

発行日: 2021-09-14

OpenShift Container Platform リリース 4.8.11 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-2021:3429 アドバイザリーにまとめられています。

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 4.8.11 コンテナーイメージの一覧

#### 1.10.7.1. バグ修正

- 従来、**Event Sources**は**Developer Catalog Group**に存在していました。今回の更新では、**Serverless**の追加グループが**Eventing**に名称変更され、**Event Sources**が**Eventing**の追加グループに存在するようになりました。(BZ#1999931)

#### 1.10.7.2. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.8. RHBA-2021:3511 - OpenShift Container Platform 4.8.12 バグ修正の更新

発行日: 2021-09-21

OpenShift Container Platform リリース 4.8.12 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-2021:3511 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-2021:3512 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 4.8.12 コンテナーイメージの一覧

#### 1.10.8.1. 特長

##### 1.10.8.1.1. クラスタに対する新しい最小ストレージ要件

OpenShift Container Platformのクラスタをインストールするために必要な最小ストレージが、120GBから100GBに減少しました。このアップデートは、サポート対応のすべてのプラットフォームに適用されます。

#### 1.10.8.2. バグ修正

- 以前は、一部のレジストリにとって大きすぎるヘッダーを**oc**ツールが送信していたため、それらのレジストリが大きなミラーリング要求を拒否していました。この更新により、**oc adm catalog mirror**コマンドのヘッダサイズに制限が設けられ、ミラーリングが期待どおりに動作するようになりました。(BZ#1874106)

- 今回の更新以前は、クラスターオートスケーラーが**csidrivers.storage.k8s.io** または **csistoragecapacities.storage.k8s.io**リソースにアクセスできなかったため、パーミッションエラーが発生していました。この修正により、クラスターオートスケーラーに割り当てられたロールが更新され、これらのリソースへのパーミッションが含まれるようになります。(BZ#1995595)

#### 1.10.8.3. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.9. RHBA-2021:3632 - OpenShift Container Platform 4.8.13 バグ修正およびセキュリティー更新

発行日: 2021-09-27

セキュリティー更新を含む OpenShift Container Platform リリース 4.8.13 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、RHBA-2021:3632 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-2021:3631 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 4.8.13 コンテナーイメージの一覧

#### 1.10.9.1. 特長

- Kubernetes 1.21.4 が利用可能になりました。より詳しい情報は、1.21.4、1.21.3、および 1.21.2のチェンジログをご覧ください。

#### 1.10.9.2. バグ修正

- 以前は、**--max components**引数を使用すると、スライスに対するインデックス操作がチェックされないことがありました。その結果、**oc**クライアントはパニックエラーを返し、クラッシュしてしまいました。今回の更新では、範囲外のインデックスに対して値が要求されないようにするためのチェックが追加されました。その結果、**--max-components**引数を使用しても、**oc**クライアントがクラッシュしなくなりました。(**BZ#2004193**)

#### 1.10.9.3. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.10. RHBA-2021:3682 - OpenShift Container Platform 4.8.14 バグ修正の更新

発行日: 2021-10-11

OpenShift Container Platform リリース 4.8.14 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-2021:3682 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-2021:3865 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 4.8.14 コンテナーイメージの一覧

#### 1.10.10.1. OpenShift Container Platformの次期リリースへのアップグレードの準備

OpenShift Container Platform 4.8.14では、OpenShift Container Platformの次のリリース（現在はOpenShift Container Platform 4.9を予定）へのアップグレードに影響するチェックが導入されています。これは、OpenShift Container Platform 4.9が使用する予定のKubernetes 1.22が、非推奨になった多くの**v1beta1** APIを削除したためです。

このチェックでは、クラスターをOpenShift Container Platform 4.8から4.9にアップグレードする前に、管理者が手動で承認する必要があります。削除されたAPIが、クラスターによって引き続き使用されるOpenShift Container Platform 4.9にアップグレードした後の問題を防ぐ上で役立ちます。管理者

は、削除されたAPIが使用されていないかどうかクラスタを評価し、適切な新しいAPIバージョンを使用するように移行する必要があります。この評価および移行が完了したら、管理者は確認応答を提供できます。

すべてのクラスターでは、OpenShift Container Platform 4.9にアップグレードする前に、この管理者の承認が必要になります。

削除されたKubernetes APIのリスト、削除されたAPIが使用されているかどうかクラスタを評価する方法のヒント、および管理者承認を提供する方法の詳細については、Preparing to upgrade to OpenShift Container Platform 4.9を参照してください。

#### 1.10.10.2. バグ修正

- これまでは、**provisioningHostIP**を設定すると、プロビジョニングネットワークを無効にしている場合でも、Metal3 Podに割り当てられていました。この問題が起こらなくなりました。(BZ#1975711)

- 以前のリリースでは、IPv6 DHCP を使用する場合、ノードインターフェースアドレスは /**128** 接頭辞でリースされる可能性がありました。その結果、OVN-Kubernetesは同じプレフィックスを使用してノードのネットワークを推測し、他のクラスターノードへのトラフィックを含む他のアドレストラフィックをゲートウェイ経由でルーティングします。今回の更新により、OVN-Kubernetesはノードのルーティングテーブルを検査し、ノードのインターフェースアドレスのより広いルーティングエントリをチェックし、そのプレフィックスを使用してノードのネットワークを推測します。その結果、他のクラスターノードへのトラフィックはゲートウェイ経由でルーティングされなくなりました。(BZ#1994624)

#### 1.10.10.3. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.11. RHBA-2021:3821 - OpenShift Container Platform 4.8.15 バグ修正およびセキュリティー更新

発行日: 2021-10-19

セキュリティー更新を含む OpenShift Container Platform リリース 4.8.15 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、RHBA-2021:3821 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-2021:3820 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 4.8.15 コンテナーイメージの一覧

#### 1.10.11.1. 既知の問題

- OpenShift Container Platformのシングルノード構成では、リアルタイムカーネル（kernel-rt）を使用した場合、非リアルタイムカーネルを使用した場合に比べてPodの作成時間が2倍以上遅くなります。kernel-rtを使用した場合、ノードの再起動後のリカバリータイムが影響を受けるため、作成時間が遅いことで、サポートされるPodの最大数に影響が出ます。kernel-rtを使用している場合の回避策として、**rcupdate.rcu_normal_after_boot=0**のカーネル引数を指定して起動することで、影響を受けた回復時間を改善することができます。この場合、リアルタイム

カーネルのバージョンは、**kernel-rt-4.18.0-305.16.1.rt7.88.el8_4**以降でなければなりません。この既知の問題は、OpenShift Container Platformのバージョン4.8.15以降に該当します。(BZ#1975356)

- OpenShift Container Platformのシングルノードのリブートに続いて、すべての Pod が再起動します。これにより、大きな負荷が発生し、通常の Pod 作成時間が長くなります。これは、Container Network Interface (CNI)が **pod add** イベントを素早く処理できないために発生します。**timed out waiting for OVS port binding** エラーメッセージが表示されます。OpenShift Container Platform の単一ノードインスタンスは最終的は復帰しますが、想定よりも遅くなります。この既知の問題は、OpenShift Container Platformのバージョン4.8.15以降に該当します。(BZ#1986216)

#### 1.10.11.2. バグ修正

- これまでは、Local Storage Operatorが孤立した永続ボリューム（PV）を削除する際に、PVを削除してから次のPVを削除するまでに10秒間待機するという問題がありました。多くのPVを削除しなければならない環境では、この10秒の待ち時間が不要な遅延を引き起こし、新しい Persistent Volume Claim (永続ボリューム要求、PVC)に時間がかかりました。このバグ修正により、10秒間の待ち時間がなくなりました。新たに発生したPersistent Volume Claim (永続ボリューム要求、PVC)は、タイムリーに処理されます。(BZ#2008088)

- これまでは、**provisioningNetwork**が無効な場合でも、ベアメタルデプロイメントの構成設定に**provisioningHostIP**の値が含まれていると、Metal3 Podが維持されないプロビジョニング IP アドレスで起動していました。Ironicは、開始時にこのプロビジョニングIPアドレスを取り、そのアドレスが使えなくなると失敗しました。このバグ修正により、**provisioningNetwork**が無効な場合に、システムが**provisioningHostIP**を無視するようになりました。Ironicは、適切に設定された外部IPアドレスで起動します。(BZ#1975711)

#### 1.10.11.3. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.12. RHBA-2021:3927 - OpenShift Container Platform 4.8.17 バグ修正およびセキュリティー更新

発行日: 2021-10-27

セキュリティー更新を含む OpenShift Container Platform リリース 4.8.17 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、RHBA-2021:3927 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-2021:3926 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 4.8.17 コンテナーイメージの一覧

#### 1.10.12.1. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.13. RHBA-2021:4020 - OpenShift Container Platform 4.8.18 バグ修正の更新

発行日: 2021-11-02

OpenShift Container Platform リリース 4.8.18 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-2021:4020 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-2021:4019 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 4.8.18 コンテナーイメージの一覧

#### 1.10.13.1. バグ修正

- ビルド設定の**lastTriggeredImageID**フィールドが非推奨なったため、イメージ変更トリガーコントローラーがビルドを開始する前に ID フィールドをチェックしなくなりました。その結果、クラスタがOpenShift Container Platform 4.7以前を実行しているときに、ビルド設定が作成され、イメージ変更のトリガー開始があった場合、継続してビルドのトリガーを試みていました。今回の更新により、これらの不要なビルド起動の試みは発生しなくなりました。(**BZ#2006793**)

#### 1.10.13.2. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.14. RHBA-2021:4109 - OpenShift Container Platform 4.8.19 バグ修正の更新

発行日: 2021-11-11

OpenShift Container Platform リリース 4.8.19 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-2021:4109 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-2021:4108 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 4.8.19 コンテナーイメージの一覧

#### 1.10.14.1. アップグレード

既存の OpenShift Container Platform 4.8 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、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。

### 1.10.15. RHBA-2021:4574 - OpenShift Container Platform 4.8.20 バグ修正の更新

発行日: 2021-11-16

OpenShift Container Platform リリース 4.8.20 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-2021:4574 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-2021:4571 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 4.8.20 コンテナーイメージの一覧

#### 1.10.15.1. 既知の問題

- 現在のopt-in難読化は、現在**hostsubnets.network.openshift.io**がOVNクラスターにないため、OVNを使用するクラスターでは機能しません。(BZ#2009322)

#### 1.10.15.2. アップグレード

既存の OpenShift Container Platform 4.8 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、Updating a cluster within a minor version by using the CLI について参照してください。

# 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、4.8 クラスターではすべてのマスターは 4.8 で、すべてのノードが 4.8 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 4.8 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。