



OpenShift Container Platform 4.3

クラスタの更新

OpenShift Container Platform クラスタの更新

OpenShift Container Platform 4.3 クラスターの更新

OpenShift Container Platform クラスターの更新

法律上の通知

Copyright © 2020 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

概要

本書では、OpenShift Container Platform クラスターを更新し、アップグレードする方法を説明します。クラスターの更新を、クラスターをオフラインにする必要のない単純なプロセスで実行できます。

目次

第1章 クラスターのマイナーバージョン間の更新	3
1.1. 前提条件	3
1.2. OPENSIFT CONTAINER PLATFORM の更新サービスについて	3
1.3. OPENSIFT CONTAINER PLATFORM アップグレードチャンネルおよびリリース	4
1.4. WEB コンソールを使用したクラスターの更新	6
第2章 WEB コンソールからのマイナーバージョン内でのクラスターの更新	9
2.1. 前提条件	9
2.2. OPENSIFT CONTAINER PLATFORM の更新サービスについて	9
2.3. OPENSIFT CONTAINER PLATFORM アップグレードチャンネルおよびリリース	10
2.4. WEB コンソールを使用したクラスターの更新	12
第3章 CLI の使用によるマイナーバージョン内でのクラスターの更新	14
3.1. 前提条件	14
3.2. OPENSIFT CONTAINER PLATFORM の更新サービスについて	14
3.3. OPENSIFT CONTAINER PLATFORM アップグレードチャンネルおよびリリース	15
3.4. CLI を使用したクラスターの更新	17
第4章 RHEL コンピュータマシンを含むクラスターの更新	21
4.1. 前提条件	21
4.2. OPENSIFT CONTAINER PLATFORM の更新サービスについて	21
4.3. OPENSIFT CONTAINER PLATFORM アップグレードチャンネルおよびリリース	22
4.4. WEB コンソールを使用したクラスターの更新	24
4.5. (オプション) RHEL マシンで ANSIBLE タスクを実行するためのフックの追加	26
4.6. クラスター内の RHEL コンピュータマシンの更新	28

第1章 クラスターのマイナーバージョン間の更新

マイナーバージョン間で OpenShift Container Platform クラスターの更新またはアップグレードを実行できます。

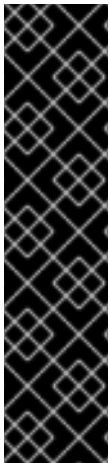


注記

ocを使用して更新チャンネルを変更するのが容易ではないため、Web コンソールを使用して更新チャンネルを変更します。Web コンソール内で更新プロセスを完了することが推奨されます。4.3 チャンネルに変更を加えた後に更新を完了するために、[CLI を使用してマイナーバージョン内でクラスターを更新する手順](#)を実行できます。

1.1. 前提条件

- **admin** 権限を持つユーザーとしてクラスターにアクセスできること。「[Using RBAC to define and apply permissions](#)」を参照してください。
- アップグレードが失敗し、[クラスターを直前の状態に復元する](#)必要がある場合に最新の **etcd バックアップ**があること。



重要

OpenShift Container Platform 4.2 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

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

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



重要

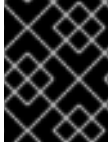
unsupportedConfigOverrides セクションを使用して Operator の設定を変更することはサポートされておらず、クラスターのアップグレードをブロックする可能性があります。クラスターをアップグレードする前に、この設定を削除する必要があります。

1.2. OPENSIFT CONTAINER PLATFORM の更新サービスについて

OpenShift Container Platform の更新サービスとは、OpenShift Container Platform と Red Hat Enterprise Linux CoreOS (RHCOS) の両方に OTA(over-the-air) 更新を提供するホスト型サービスです。コンポーネント Operator のグラフ、または**頂点**とそれらを結ぶ**辺**を含む図表が提示されます。グラフの辺は、どのバージョンであれば安全に更新できるかを示します。頂点は、管理されたクラスターコンポーネントの想定された状態を特定する更新のペイロードです。

クラスター内の Cluster Version Operator (CVO) は、OpenShift Container Platform の更新サービスをチェックして、グラフの現在のコンポーネントバージョンとグラフの情報に基づき、有効な更新および更新パスを確認します。ユーザーが更新をリクエストすると、OpenShift Container Platform CVO はその更新のリリースイメージを使ってクラスターをアップグレードします。リリースアーティファクトは、コンテナイメージとして Quay でホストされます。

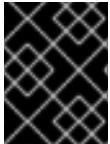
OpenShift Container Platform 更新サービスが互換性のある更新のみを提供できるようにするために、自動化を支援するリリース検証 Pipeline が使用されます。それぞれのリリースアーティファクトについて、他のコンポーネントパッケージだけでなくサポートされているクラウドプラットフォームおよびシステムアーキテクチャーとの互換性の有無が検証されます。Pipeline がリリースの適合性を確認した後に、OpenShift Container Platform 更新サービスは更新が利用可能であることを通知します。



重要

更新サービスが有効な更新をすべて表示するために、更新サービスが表示しないバージョンへの更新を強制することはできません。

連続更新モードでは、2つのコントローラーが実行されます。1つのコントローラーはペイロードマニフェストを絶えず更新し、それらをクラスターに適用し、Operator が利用可能か、アップグレード中か、または失敗しているかに応じて Operator の制御されたロールアウトのステータスを出力します。2つ目のコントローラーは OpenShift Container Platform 更新サービスをポーリングして、更新が利用可能かどうかを判別します。



重要

クラスターを以前のバージョンに戻すこと、つまりロールバックはサポートされていません。サポートされているのは、新規バージョンへのアップグレードのみです。

アップグレードプロセスで、Machine Config Operator (MCO) は新規設定をクラスターマシンに適用します。これは、マシン設定プールの **maxUnavailable** フィールドによって指定されるノードの数を分離し、それらを利用不可としてマークします。デフォルトで、この値は **1** に設定されます。次に、新しい設定を適用して、マシンを再起動します。Red Hat Enterprise Linux (RHEL) マシンをワーカーとして使用する場合、まず OpenShift API をそれらのマシンで更新する必要があるため、MCO はそれらのマシンで kubelet を更新しません。新規バージョンの仕様は古い kubelet に適用されるため、RHEL マシンを **Ready** 状態に戻すことができません。マシンが利用可能になるまで更新を完了することはできません。ただし、利用不可のノードの最大数は、その数のマシンがサービス停止状態のマシンとして分離されても通常のクラスター操作が継続できるようにするために設定されます。

追加リソース

- [管理外の Operator のサポートポリシー](#)

1.3. OPENSIFT CONTAINER PLATFORM アップグレードチャンネルおよびリリース

OpenShift Container Platform 4.1 で、Red Hat はクラスターのアップグレードの適切なリリースバージョンを推奨するためにチャンネルという概念を導入しました。アップグレードのペースを制御することで、これらのアップグレードチャンネルからアップグレードストラテジーを選択することができます。アップグレードチャンネルは OpenShift Container Platform のマイナーバージョンに関連付けられます。たとえば、OpenShift Container Platform 4.3 アップグレードチャンネルには 4.4 リリースへのアップグレードが含まれることはありません。このストラテジーにより、管理者は OpenShift Container Platform の次のマイナーバージョンへのアップグレードに関して明確な決定を行うことができます。アップグレードチャンネルはリリースの選択のみを制御し、インストールするクラスターのバージョンには影響を与えません。OpenShift Container Platform の特定のバージョンの **openshift-install** バイナリファイルは常に該当バージョンをインストールします。

OpenShift Container Platform 4.3 は以下のアップグレードチャンネルを提供します。

- **candidate-4.3**

- **fast-4.3**
- **stable-4.3**

candidate-4.3 チャンネル

candidate-4.3 チャンネルには、z-stream (4.3.z) リリースの候補となるビルドが含まれます。リリース候補には、製品のすべての機能が含まれますが、それらがサポートされる訳ではありません。リリース候補を使用して機能の受け入れテストを実行し、OpenShift Container Platform の次のバージョンへの対応を支援します。リリース候補は、名前に **-rc** を含まないものを含め、候補チャンネルで利用可能なビルドを指します。候補チャンネルでバージョンが利用可能になると、さらに品質のチェックが行われます。品質基準を満たす場合には、これは **fast-4.3** または **stable-4.3** チャンネルにプロモートされます。このストラテジーにより、特定のリリースが **candidate-4.3** チャンネルと **fast-4.3** または **stable-4.3** チャンネルの両方で利用可能な場合、そのリリースは Red Hat でサポートされるバージョンということになります。**candidate-4.3** チャンネルには、いずれのチャンネルでも推奨されている更新のないリリースバージョンを含めることができます。

candidate-4.3 チャンネルを使用して、OpenShift Container Platform の直前のマイナーバージョンからアップグレードできます。



注記

リリース候補は、<https://www.openshift.com/try> サイトにある夜間バッチとは異なります。夜間ビルドは各種機能への早期アクセスのために利用できますが、夜間バッチへのからの更新は推奨されておらず、サポートもされていません。夜間ビルドはいずれのアップグレードチャンネルでも利用できません。

fast-4.3 チャンネル

fast-4.3 チャンネルは、Red Hat が一般公開リリースとして指定のバージョンを宣言するとすぐに新しいバージョンで更新されます。そのため、これらのリリースは完全にサポートされ、実稼働用の品質があり、これらのリリースのプロモート元の **candidate-4.3** チャンネルのリリース候補として利用可能であった間のパフォーマンスにも問題がなかったリリースです。リリースは **fast-4.3** チャンネルに表示されてからしばらくすると、**stable-4.3** チャンネルに追加されます。リリースは **fast-4.3** チャンネルに表示される前に、**stable-4.3** チャンネルに表示されることはありません。

fast-4.3 チャンネルを使用して、OpenShift Container Platform の以前のマイナーバージョンからのアップグレードを実行できます。

stable-4.3 チャンネル

fast-4.3 チャンネルにはエラータの公開後すぐにリリースが組み込まれ、リリースの **stable-4.3** チャンネルへの追加は遅延します。この期間中、接続環境のカスタマープログラム(Connected Customer Program)に関わる Red Hat SRE チーム、Red Hat サポートサービス、および実稼働前および実稼働環境からリリースの安定性についてのデータが収集されます。

stable-4.3 チャンネルを使用して、OpenShift Container Platform の以前のマイナーバージョンからのアップグレードを実行できます。

アップグレードバージョンパス

OpenShift Container Platform では、インストールされた OpenShift Container Platform のバージョンと、次のリリースにアクセスするために選択したチャンネル内のパスの確認を可能にするアップグレード推奨サービスが提供されます。**fast-4.3** チャンネルでは以下を確認できます。

- 4.3.0
- 4.3.1
- 4.3.3

- 4.3.4

このサービスは、テスト済みの重大な問題のないアップグレードのみを推奨します。クラスターが 4.3.1 にあり、OpenShift Container Platform が 4.3.4 を提案している場合、4.3.1 から 4.3.4 に更新しても問題がありません。パッチの連続する番号のみに依存しないようにしてください。たとえば、この例では 4.3.2 はチャンネルで利用可能な状態ではなく、これまで利用可能になったことがありません。更新サービスは、既知の脆弱性を含む OpenShift Container Platform のバージョンへの更新を提案しません。

更新の安定性は、チャンネルによって異なります。**candidate-4.3** チャンネルに更新についての推奨があるからといって、その更新が必ずしもサポートされる訳ではありません。つまり、更新について深刻な問題がまだ検出されていないものの、この更新の安定性についての提案を導くようなトラフィックの安定性はとくに確認されていない可能性があります。**fast-4.3** または **stable-4.3** チャンネルに更新の推奨がある場合は、更新がそれぞれのチャンネルにある限り完全にサポートされることを示します。リリースがチャンネルから削除されることは決してありませんが、深刻な問題を示す更新の推奨はすべてのチャンネルから削除されます。更新の推奨が削除された後に開始された更新はサポートされない可能性があります。

Red Hat は最終的には、**fast-4.3** または **stable-4.3** チャンネルのサポートされるリリースから 4.3.z の最新リリースへのサポートされる更新パスを提供します。ただし、問題のあるリリースからの安全なパスが構築され、検証される間に遅延が生じる可能性があります。

高速かつ安定したチャンネルの使用およびストラテジー

fast-4.3 および **stable-4.3** チャンネルでは、一般公開リリースが利用可能になり次第これを受信するか、または Red Hat がそれらの更新のロールアウトを制御するようにするかを選択することができます。問題がロールアウト時またはロールアウト後に検出される場合、該当バージョンへのアップグレードは **fast-4.3** および **stable-4.3** チャンネルの両方でブロックされ、新たに推奨されるアップグレード先の新規バージョンが導入される可能性があります。

fast-4.3 チャンネルで実稼働前のシステムを設定し、**stable-4.3** チャンネルで実稼働システムを設定してから Red Hat の接続環境のカスタマープログラム(Connected Customer Program)に参加することで、お客様のプロセスを改善することができます。Red Hat はこのプログラムを使用して、ご使用の特定のハードウェアおよびソフトウェア設定に対する更新の影響の有無を確認します。今後のリリースでは、更新が **fast-4.3** から **stable-4.3** チャンネルに移行するペースが改善されるか、変更される可能性があります。

ネットワークが制限された環境のクラスター

OpenShift Container Platform クラスターのコンテナイメージを独自に管理する場合には、製品リリースに関連する Red Hat エラータを確認し、アップグレードへの影響に関するコメントに留意する必要があります。アップグレード時に、インターフェースにこれらのバージョン間の切り替えについての警告が表示される場合があります。そのため、これらの警告を無視するかどうかを決める前に適切なバージョンを選択していることを確認する必要があります。

チャンネル間の切り替え

stable-4.3 チャンネルから **fast-4.3** チャンネルに切り換える場合も、クラスターは引き続きサポートされます。**candidate-4.3** チャンネルに常に切り替えることはできますが、そのチャンネルの一部のリリースはサポートされないリリース候補である可能性があります。現在のリリースが一般利用公開リリースの場合、**candidate-4.3** チャンネルから **fast-4.3** チャンネルに切り換えることができます。**fast-4.3** チャンネルから **stable-4.3** チャンネルに常に切り換えることができますが、現在のリリースが **fast-4.3** に最近プロモートされた場合、リリースが **stable-4.3** にプロモートされるまでに最長 1 日分の遅延が生じる可能性があります。現在のリリースを含まないチャンネルに変更すると、アラートが表示され、更新は推奨されません。ただし、元のチャンネルにいつでも安全に戻ることができます。

1.4. WEB コンソールを使用したクラスターの更新

更新が利用可能な場合、Web コンソールからクラスターを更新できます。

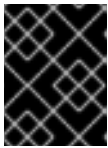
利用可能な OpenShift Container Platform アドバイザリーおよび更新については、カスタマーポータル の [エラー](#) のセクションを参照してください。

前提条件

- **admin** 権限を持つユーザーとして Web コンソールにアクセスできること。

手順

1. Web コンソールから、**Administration** > **Cluster Settings** をクリックし、**Overview** タブの内容を確認します。
2. 実稼働クラスターの場合、**CHANNEL** が **stable-4.3** などの現在のマイナーバージョンの正しいチャンネルに設定されていることを確認します。



重要

実稼働クラスターの場合、stable-* または fast-* チャンネルにサブスクライブする必要があります。

- **UPDATE STATUS** が **Updates Available** ではない場合、クラスターをアップグレードすることはできません。
 - **DESIRED VERSION** は、クラスターが実行されているか、または更新されるクラスターのバージョンを示します。
3. **Updates Available** をクリックし、利用可能な最新バージョンを選択し、**Update** をクリックします。**UPDATE STATUS** は **Updating** に切り替わり、**Cluster Operators** タブで Operator のアップグレードの進捗を確認できます。
 4. OpenShift Container Platform 4.2 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。OpenShift CLI (**oc**) を必要とする以下のコマンドを使用してこれを実行できます。

```
$ for I in $(oc get ns -o jsonpath='{range .items[*]} {.metadata.name}{"\n"} {end}'); \
do oc delete pods --all -n $I; \
sleep 1; \
done
```



注記

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

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

5. 更新が完了し、Cluster Version Operator が利用可能な更新を更新したら、追加の更新が現在のチャンネルで利用可能かどうかを確認します。

- 更新が利用可能な場合は、更新ができなくなるまで、現在のチャンネルでの更新を継続します。
- 利用可能な更新がない場合は、**CHANNEL** を次のマイナーバージョンの stable-* または fast-* チャンネルに切り替え、そのチャンネルで必要なバージョンに更新します。

必要なバージョンに達するまで、いくつかの中間更新を実行する必要がある場合があります。

第2章 WEB コンソールからのマイナーバージョン内でのクラスターの更新

Web コンソールを使用して、OpenShift Container Platform クラスターの更新またはアップグレードを実行できます。

2.1. 前提条件

- **admin** 権限を持つユーザーとしてクラスターにアクセスできること。「[Using RBAC to define and apply permissions](#)」を参照してください。
- アップグレードが失敗し、[クラスターを直前の状態に復元する](#)必要がある場合に最新の [etcd バックアップ](#)があること。

重要

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

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

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

2.2. OPENSIFT CONTAINER PLATFORM の更新サービスについて

OpenShift Container Platform の更新サービスとは、OpenShift Container Platform と Red Hat Enterprise Linux CoreOS (RHCOS) の両方に OTA(over-the-air) 更新を提供するホスト型サービスです。コンポーネント Operator のグラフ、または[頂点](#)とそれらを結ぶ[辺](#)を含む図表が提示されます。グラフの辺は、どのバージョンであれば安全に更新できるかを示します。頂点は、管理されたクラスターコンポーネントの想定された状態を特定する更新のペイロードです。

クラスター内の Cluster Version Operator (CVO) は、OpenShift Container Platform の更新サービスをチェックして、グラフの現在のコンポーネントバージョンとグラフの情報に基づき、有効な更新および更新パスを確認します。ユーザーが更新をリクエストすると、OpenShift Container Platform CVO はその更新のリリースイメージを使ってクラスターをアップグレードします。リリースアーティファクトは、コンテナイメージとして Quay でホストされます。

OpenShift Container Platform 更新サービスが互換性のある更新のみを提供できるようにするために、自動化を支援するリリース検証 Pipeline が使用されます。それぞれのリリースアーティファクトについて、他のコンポーネントパッケージだけでなくサポートされているクラウドプラットフォームおよびシステムアーキテクチャーとの互換性の有無が検証されます。Pipeline がリリースの適合性を確認した後に、OpenShift Container Platform 更新サービスは更新が利用可能であることを通知します。

重要

更新サービスが有効な更新をすべて表示するために、更新サービスが表示しないバージョンへの更新を強制することはできません。

連続更新モードでは、2つのコントローラーが実行されます。1つのコントローラーはペイロードマニフェストを絶えず更新し、それらをクラスターに適用し、Operator が利用可能か、アップグレード中か、または失敗しているかに応じて Operator の制御されたロールアウトのステータスを出力します。2つ目のコントローラーは OpenShift Container Platform 更新サービスをポーリングして、更新が利用可能かどうかを判別します。



重要

クラスターを以前のバージョンに戻すこと、つまりロールバックはサポートされていません。サポートされているのは、新規バージョンへのアップグレードのみです。

アップグレードプロセスで、Machine Config Operator (MCO) は新規設定をクラスターマシンに適用します。これは、マシン設定プールの **maxUnavailable** フィールドによって指定されるノードの数を分離し、それらを利用不可としてマークします。デフォルトで、この値は **1** に設定されます。次に、新しい設定を適用して、マシンを再起動します。Red Hat Enterprise Linux (RHEL) マシンをワーカーとして使用する場合、まず OpenShift API をそれらのマシンで更新する必要があるため、MCO はそれらのマシンで kubelet を更新しません。新規バージョンの仕様は古い kubelet に適用されるため、RHEL マシンを **Ready** 状態に戻すことができません。マシンが利用可能になるまで更新を完了することはできません。ただし、利用不可のノードの最大数は、その数のマシンがサービス停止状態のマシンとして分離されても通常のクラスター操作が継続できるようにするために設定されます。

追加リソース

- [管理外の Operator のサポートポリシー](#)

2.3. OPENSIFT CONTAINER PLATFORM アップグレードチャンネルおよびリリース

OpenShift Container Platform 4.1 で、Red Hat はクラスターのアップグレードの適切なリリースバージョンを推奨するためにチャンネルという概念を導入しました。アップグレードのペースを制御することで、これらのアップグレードチャンネルからアップグレード戦略を選択することができます。アップグレードチャンネルは OpenShift Container Platform のマイナーバージョンに関連付けられます。たとえば、OpenShift Container Platform 4.3 アップグレードチャンネルには 4.4 リリースへのアップグレードが含まれることはありません。この戦略により、管理者は OpenShift Container Platform の次のマイナーバージョンへのアップグレードに関して明確な決定を行うことができます。アップグレードチャンネルはリリースの選択のみを制御し、インストールするクラスターのバージョンには影響を与えません。OpenShift Container Platform の特定のバージョンの **openshift-install** バイナリファイルは常に該当バージョンをインストールします。

OpenShift Container Platform 4.3 は以下のアップグレードチャンネルを提供します。

- **candidate-4.3**
- **fast-4.3**
- **stable-4.3**

candidate-4.3 チャンネル

candidate-4.3 チャンネルには、z-stream (4.3.z) リリースの候補となるビルドが含まれます。リリース候補には、製品のすべての機能が含まれますが、それらがサポートされる訳ではありません。リリース候補を使用して機能の受け入れテストを実行し、OpenShift Container Platform の次のバージョンへの対応を支援します。リリース候補は、名前に **-rc** を含まないものを含め、候補チャンネルで利用可能なビルドを指します。候補チャンネルでバージョンが利用可能になると、さらに品質のチェックが行われます。品質基準を満たす場合には、これは **fast-4.3** または **stable-4.3** チャンネルにプロモートされます。このス

トラテジーにより、特定のリリースが **candidate-4.3** チャンネルと **fast-4.3** または **stable-4.3** チャンネルの両方で利用可能な場合、そのリリースは Red Hat でサポートされるバージョンということになります。**candidate-4.3** チャンネルには、いずれのチャンネルでも推奨されている更新のないリリースバージョンを含めることができます。

candidate-4.3 チャンネルを使用して、OpenShift Container Platform の直前のマイナーバージョンからアップグレードできます。



注記

リリース候補は、<https://www.openshift.com/try> サイトにある夜間バッチとは異なります。夜間ビルドは各種機能への早期アクセスのために利用できますが、夜間バッチへの/からの更新は推奨されておらず、サポートもされていません。夜間ビルドはいずれのアップグレードチャンネルでも利用できません。

fast-4.3 チャンネル

fast-4.3 チャンネルは、Red Hat が一般公開リリースとして指定のバージョンを宣言するとすぐに新しいバージョンで更新されます。そのため、これらのリリースは完全にサポートされ、実稼働用の品質があり、これらのリリースのプロモート元の **candidate-4.3** チャンネルのリリース候補として利用可能であった間のパフォーマンスにも問題がなかったリリースです。リリースは **fast-4.3** チャンネルに表示されてからしばらくすると、**stable-4.3** チャンネルに追加されます。リリースは **fast-4.3** チャンネルに表示される前に、**stable-4.3** チャンネルに表示されることはありません。

fast-4.3 チャンネルを使用して、OpenShift Container Platform の以前のマイナーバージョンからのアップグレードを実行できます。

stable-4.3 チャンネル

fast-4.3 チャンネルにはエラータの公開後すぐにリリースが組み込まれ、リリースの **stable-4.3** チャンネルへの追加は遅延します。この期間中、接続環境のカスタマープログラム(Connected Customer Program)に関わる Red Hat SRE チーム、Red Hat サポートサービス、および実稼働前および実稼働環境からリリースの安定性についてのデータが収集されます。

stable-4.3 チャンネルを使用して、OpenShift Container Platform の以前のマイナーバージョンからのアップグレードを実行できます。

アップグレードバージョンパス

OpenShift Container Platform では、インストールされた OpenShift Container Platform のバージョンと、次のリリースにアクセスするために選択したチャンネル内のパスの確認を可能にするアップグレード推奨サービスが提供されます。**fast-4.3** チャンネルでは以下を確認できます。

- 4.3.0
- 4.3.1
- 4.3.3
- 4.3.4

このサービスは、テスト済みの重大な問題のないアップグレードのみを推奨します。クラスターが 4.3.1 にあり、OpenShift Container Platform が 4.3.4 を提案している場合、4.3.1 から 4.3.4 に更新しても問題がありません。パッチの連続する番号のみに依存しないようにしてください。たとえば、この例では 4.3.2 はチャンネルで利用可能な状態ではなく、これまで利用可能になったことがありません。更新サービスは、既知の脆弱性を含む OpenShift Container Platform のバージョンへの更新を提案しません。

更新の安定性は、チャンネルによって異なります。**candidate-4.3** チャンネルに更新についての推奨があるからといって、その更新が必ずしもサポートされる訳ではありません。つまり、更新について深刻な問

題がまだ検出されていないものの、この更新の安定性についての提案を導くようなトラフィックの安定性はとくに確認されていない可能性があります。**fast-4.3** または **stable-4.3** チャンネルに更新の推奨がある場合は、更新がそれぞれのチャンネルにある限り完全にサポートされることを示します。リリースがチャンネルから削除されることは決してありませんが、深刻な問題を示す更新の推奨はすべてのチャンネルから削除されます。更新の推奨が削除された後に開始された更新はサポートされない可能性があります。

Red Hat は最終的には、**fast-4.3** または **stable-4.3** チャンネルのサポートされるリリースから 4.3.z の最新リリースへのサポートされる更新パスを提供します。ただし、問題のあるリリースからの安全なパスが構築され、検証される間に遅延が生じる可能性があります。

高速かつ安定したチャンネルの使用およびストラテジー

fast-4.3 および **stable-4.3** チャンネルでは、一般公開リリースが利用可能になり次第これを受信するか、または Red Hat がそれらの更新のロールアウトを制御するようにするかを選択することができます。問題がロールアウト時またはロールアウト後に検出される場合、該当バージョンへのアップグレードは **fast-4.3** および **stable-4.3** チャンネルの両方でブロックされ、新たに推奨されるアップグレード先の新規バージョンが導入される可能性があります。

fast-4.3 チャンネルで実稼働前のシステムを設定し、**stable-4.3** チャンネルで実稼働システムを設定してから Red Hat の接続環境のカスタマープログラム(Connected Customer Program)に参加することで、お客様のプロセスを改善することができます。Red Hat はこのプログラムを使用して、ご使用の特定のハードウェアおよびソフトウェア設定に対する更新の影響の有無を確認します。今後のリリースでは、更新が **fast-4.3** から **stable-4.3** チャンネルに移行するペースが改善されるか、変更される可能性があります。

ネットワークが制限された環境のクラスター

OpenShift Container Platform クラスターのコンテナイメージを独自に管理する場合には、製品リリースに関連する Red Hat エラータを確認し、アップグレードへの影響に関するコメントに留意する必要があります。アップグレード時に、インターフェースにこれらのバージョン間の切り替えについての警告が表示される場合があります。そのため、これらの警告を無視するかどうかを決める前に適切なバージョンを選択していることを確認する必要があります。

チャンネル間の切り替え

stable-4.3 チャンネルから **fast-4.3** チャンネルに切り換える場合も、クラスターは引き続きサポートされます。**candidate-4.3** チャンネルに常に切り替えることはできませんが、そのチャンネルの一部のリリースはサポートされないリリース候補である可能性があります。現在のリリースが一般利用公開リリースの場合、**candidate-4.3** チャンネルから **fast-4.3** チャンネルに切り換えることができます。**fast-4.3** チャンネルから **stable-4.3** チャンネルに常に切り換えることができますが、現在のリリースが **fast-4.3** に最近プロモートされた場合、リリースが **stable-4.3** にプロモートされるまでに最長1日分の遅延が生じる可能性があります。現在のリリースを含まないチャンネルに変更すると、アラートが表示され、更新は推奨されません。ただし、元のチャンネルにいつでも安全に戻ることができます。

2.4. WEB コンソールを使用したクラスターの更新

更新が利用可能な場合、Web コンソールからクラスターを更新できます。

利用可能な OpenShift Container Platform アドバイザリーおよび更新については、カスタマーポータル [の エラータ](#) のセクションを参照してください。

前提条件

- **admin** 権限を持つユーザーとして Web コンソールにアクセスできること。

手順

1. Web コンソールから、**Administration > Cluster Settings** をクリックし、**Overview** タブの内容を確認します。
2. 実稼働クラスターの場合、**CHANNEL** が **stable-4.3** などの現在のマイナーバージョンの、更新する必要のあるバージョンの正しいチャンネルに設定されていることを確認します。



重要

実稼働クラスターの場合、stable-* または fast-* チャンネルにサブスクライブする必要があります。

- **UPDATE STATUS** が **Updates Available** ではない場合、クラスターをアップグレードすることはできません。
 - **DESIRED VERSION** は、クラスターが実行されているか、または更新されるクラスターのバージョンを示します。
3. **Updates Available** をクリックし、更新するバージョンで、利用可能な最新バージョンを選択し、**Update** をクリックします。**UPDATE STATUS** は **Updating** に切り替わり、**Cluster Operators** タブで Operator のアップグレードの進捗を確認できます。
 4. OpenShift Container Platform 4.2 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。OpenShift CLI (**oc**) を必要とする以下のコマンドを使用してこれを実行できます。

```
$ for I in $(oc get ns -o jsonpath='{range .items[*]} {.metadata.name}{"\n"} {end}'); \
do oc delete pods --all -n $I; \
sleep 1; \
done
```



注記

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

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

5. 更新が完了し、Cluster Version Operator が利用可能な更新を更新したら、追加の更新が現在のチャンネルで利用可能かどうかを確認します。
 - 更新が利用可能な場合は、更新ができなくなるまで、現在のチャンネルでの更新を続けます。
 - 利用可能な更新がない場合は、**CHANNEL** を次のマイナーバージョンの stable-* または fast-* チャンネルに切り替え、そのチャンネルに必要なバージョンに更新します。

必要なバージョンに達するまで、いくつかの中間更新を実行する必要がある場合があります。

第3章 CLI の使用によるマイナーバージョン内でのクラスターの更新

OpenShift CLI (**oc**) を使用して OpenShift Container Platform クラスターをマイナーバージョン内で更新するか、またはアップグレードすることができます。

3.1. 前提条件

- **admin** 権限を持つユーザーとしてクラスターにアクセスできること。「[Using RBAC to define and apply permissions](#)」を参照してください。
- アップグレードが失敗し、[クラスターを直前の状態に復元する](#)必要がある場合に最新の **etcd** [バックアップ](#)があること。



重要

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

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

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

3.2. OPENSIFT CONTAINER PLATFORM の更新サービスについて

OpenShift Container Platform の更新サービスとは、OpenShift Container Platform と Red Hat Enterprise Linux CoreOS (RHCOS) の両方に OTA(over-the-air) 更新を提供するホスト型サービスです。コンポーネント Operator のグラフ、または**頂点**とそれらを結ぶ**辺**を含む図表が提示されます。グラフの辺は、どのバージョンであれば安全に更新できるかを示します。頂点は、管理されたクラスターコンポーネントの想定された状態を特定する更新のペイロードです。

クラスター内の Cluster Version Operator (CVO) は、OpenShift Container Platform の更新サービスをチェックして、グラフの現在のコンポーネントバージョンとグラフの情報に基づき、有効な更新および更新パスを確認します。ユーザーが更新をリクエストすると、OpenShift Container Platform CVO はその更新のリリースイメージを使ってクラスターをアップグレードします。リリースアーティファクトは、コンテナイメージとして Quay でホストされます。

OpenShift Container Platform 更新サービスが互換性のある更新のみを提供できるようにするために、自動化を支援するリリース検証 Pipeline が使用されます。それぞれのリリースアーティファクトについて、他のコンポーネントパッケージだけでなくサポートされているクラウドプラットフォームおよびシステムアーキテクチャーとの互換性の有無が検証されます。Pipeline がリリースの適合性を確認した後に、OpenShift Container Platform 更新サービスは更新が利用可能であることを通知します。



重要

更新サービスが有効な更新をすべて表示するために、更新サービスが表示しないバージョンへの更新を強制することはできません。

連続更新モードでは、2つのコントローラーが実行されます。1つのコントローラーはペイロードマニフェストを絶えず更新し、それらをクラスタに適用し、Operatorが利用可能か、アップグレード中か、または失敗しているかに応じてOperatorの制御されたロールアウトのステータスを出力します。2つ目のコントローラーはOpenShift Container Platform更新サービスをポーリングして、更新が利用可能かどうかを判別します。



重要

クラスタを以前のバージョンに戻すこと、つまりロールバックはサポートされていません。サポートされているのは、新規バージョンへのアップグレードのみです。

アップグレードプロセスで、Machine Config Operator (MCO) は新規設定をクラスタマシンに適用します。これは、マシン設定プールの **maxUnavailable** フィールドによって指定されるノードの数を分離し、それらを利用不可としてマークします。デフォルトで、この値は **1** に設定されます。次に、新しい設定を適用して、マシンを再起動します。Red Hat Enterprise Linux (RHEL) マシンをワーカーとして使用する場合、まず OpenShift API をそれらのマシンで更新する必要があるため、MCO はそれらのマシンで kubelet を更新しません。新規バージョンの仕様は古い kubelet に適用されるため、RHEL マシンを **Ready** 状態に戻すことができません。マシンが利用可能になるまで更新を完了することはできません。ただし、利用不可のノードの最大数は、その数のマシンがサービス停止状態のマシンとして分離されても通常のクラスタ操作が継続できるようにするために設定されます。

追加リソース

- [管理外の Operator のサポートポリシー](#)

3.3. OPENSIFT CONTAINER PLATFORM アップグレードチャネルおよびリリース

OpenShift Container Platform 4.1 で、Red Hat はクラスタのアップグレードの適切なリリースバージョンを推奨するためにチャネルという概念を導入しました。アップグレードのペースを制御することで、これらのアップグレードチャネルからアップグレード戦略を選択することができます。アップグレードチャネルは OpenShift Container Platform のマイナーバージョンに関連付けられます。たとえば、OpenShift Container Platform 4.3 アップグレードチャネルには 4.4 リリースへのアップグレードが含まれることはありません。この戦略により、管理者は OpenShift Container Platform の次のマイナーバージョンへのアップグレードに関して明確な決定を行うことができます。アップグレードチャネルはリリースの選択のみを制御し、インストールするクラスタのバージョンには影響を与えません。OpenShift Container Platform の特定のバージョンの **openshift-install** バイナリファイルは常に該当バージョンをインストールします。

OpenShift Container Platform 4.3 は以下のアップグレードチャネルを提供します。

- **candidate-4.3**
- **fast-4.3**
- **stable-4.3**

candidate-4.3 チャネル

candidate-4.3 チャネルには、z-stream (4.3.z) リリースの候補となるビルドが含まれます。リリース候補には、製品のすべての機能が含まれますが、それらがサポートされる訳ではありません。リリース候補を使用して機能の受け入れテストを実行し、OpenShift Container Platform の次のバージョンへの対応を支援します。リリース候補は、名前に **-rc** を含まないものを含め、候補チャネルで利用可能なビルドを指します。候補チャネルでバージョンが利用可能になると、さらに品質のチェックが行われます。品質基準を満たす場合には、これは **fast-4.3** または **stable-4.3** チャネルにプロモートされます。このス

トラジェジーにより、特定のリリースが **candidate-4.3** チャンネルと **fast-4.3** または **stable-4.3** チャンネルの両方で利用可能な場合、そのリリースは Red Hat でサポートされるバージョンということになります。**candidate-4.3** チャンネルには、いずれのチャンネルでも推奨されている更新のないリリースバージョンを含めることができます。

candidate-4.3 チャンネルを使用して、OpenShift Container Platform の直前のマイナーバージョンからアップグレードできます。



注記

リリース候補は、<https://www.openshift.com/try> サイトにある夜間バッチとは異なります。夜間ビルドは各種機能への早期アクセスのために利用できますが、夜間バッチへの/からの更新は推奨されておらず、サポートもされていません。夜間ビルドはいずれのアップグレードチャンネルでも利用できません。

fast-4.3 チャンネル

fast-4.3 チャンネルは、Red Hat が一般公開リリースとして指定のバージョンを宣言するとすぐに新しいバージョンで更新されます。そのため、これらのリリースは完全にサポートされ、実稼働用の品質があり、これらのリリースのプロモート元の **candidate-4.3** チャンネルのリリース候補として利用可能であった間のパフォーマンスにも問題がなかったリリースです。リリースは **fast-4.3** チャンネルに表示されてからしばらくすると、**stable-4.3** チャンネルに追加されます。リリースは **fast-4.3** チャンネルに表示される前に、**stable-4.3** チャンネルに表示されることはありません。

fast-4.3 チャンネルを使用して、OpenShift Container Platform の以前のマイナーバージョンからのアップグレードを実行できます。

stable-4.3 チャンネル

fast-4.3 チャンネルにはエラータの公開後すぐにリリースが組み込まれ、リリースの **stable-4.3** チャンネルへの追加は遅延します。この期間中、接続環境のカスタマープログラム(Connected Customer Program)に関わる Red Hat SRE チーム、Red Hat サポートサービス、および実稼働前および実稼働環境からリリースの安定性についてのデータが収集されます。

stable-4.3 チャンネルを使用して、OpenShift Container Platform の以前のマイナーバージョンからのアップグレードを実行できます。

アップグレードバージョンパス

OpenShift Container Platform では、インストールされた OpenShift Container Platform のバージョンと、次のリリースにアクセスするために選択したチャンネル内のパスの確認を可能にするアップグレード推奨サービスが提供されます。**fast-4.3** チャンネルでは以下を確認できます。

- 4.3.0
- 4.3.1
- 4.3.3
- 4.3.4

このサービスは、テスト済みの重大な問題のないアップグレードのみを推奨します。クラスターが 4.3.1 にあり、OpenShift Container Platform が 4.3.4 を提案している場合、4.3.1 から 4.3.4 に更新しても問題がありません。パッチの連続する番号のみに依存しないようにしてください。たとえば、この例では 4.3.2 はチャンネルで利用可能な状態ではなく、これまで利用可能になったことがありません。更新サービスは、既知の脆弱性を含む OpenShift Container Platform のバージョンへの更新を提案しません。

更新の安定性は、チャンネルによって異なります。**candidate-4.3** チャンネルに更新についての推奨があるからといって、その更新が必ずしもサポートされる訳ではありません。つまり、更新について深刻な問

題がまだ検出されていないものの、この更新の安定性についての提案を導くようなトラフィックの安定性はとくに確認されていない可能性があります。**fast-4.3** または **stable-4.3** チャンネルに更新の推奨がある場合は、更新がそれぞれのチャンネルにある限り完全にサポートされることを示します。リリースがチャンネルから削除されることは決してありませんが、深刻な問題を示す更新の推奨はすべてのチャンネルから削除されます。更新の推奨が削除された後に開始された更新はサポートされない可能性があります。

Red Hat は最終的には、**fast-4.3** または **stable-4.3** チャンネルのサポートされるリリースから 4.3.z の最新リリースへのサポートされる更新パスを提供します。ただし、問題のあるリリースからの安全なパスが構築され、検証される間に遅延が生じる可能性があります。

高速かつ安定したチャンネルの使用およびストラテジー

fast-4.3 および **stable-4.3** チャンネルでは、一般公開リリースが利用可能になり次第これを受信するか、または Red Hat がそれらの更新のロールアウトを制御するようにするかを選択することができます。問題がロールアウト時またはロールアウト後に検出される場合、該当バージョンへのアップグレードは **fast-4.3** および **stable-4.3** チャンネルの両方でブロックされ、新たに推奨されるアップグレード先の新規バージョンが導入される可能性があります。

fast-4.3 チャンネルで実稼働前のシステムを設定し、**stable-4.3** チャンネルで実稼働システムを設定してから Red Hat の接続環境のカスタマープログラム(Connected Customer Program)に参加することで、お客様のプロセスを改善することができます。Red Hat はこのプログラムを使用して、ご使用の特定のハードウェアおよびソフトウェア設定に対する更新の影響の有無を確認します。今後のリリースでは、更新が **fast-4.3** から **stable-4.3** チャンネルに移行するペースが改善されるか、変更される可能性があります。

ネットワークが制限された環境のクラスタ

OpenShift Container Platform クラスタのコンテナイメージを独自に管理する場合には、製品リリースに関連する Red Hat エラータを確認し、アップグレードへの影響に関するコメントに留意する必要があります。アップグレード時に、インターフェースにこれらのバージョン間の切り替えについての警告が表示される場合があります。そのため、これらの警告を無視するかどうかを決める前に適切なバージョンを選択していることを確認する必要があります。

チャンネル間の切り替え

stable-4.3 チャンネルから **fast-4.3** チャンネルに切り換える場合も、クラスタは引き続きサポートされます。**candidate-4.3** チャンネルに常に切り替えることはできますが、そのチャンネルの一部のリリースはサポートされないリリース候補である可能性があります。現在のリリースが一般利用公開リリースの場合、**candidate-4.3** チャンネルから **fast-4.3** チャンネルに切り換えることができます。**fast-4.3** チャンネルから **stable-4.3** チャンネルに常に切り換えることができますが、現在のリリースが **fast-4.3** に最近プロモートされた場合、リリースが **stable-4.3** にプロモートされるまでに最長1日分の遅延が生じる可能性があります。現在のリリースを含まないチャンネルに変更すると、アラートが表示され、更新は推奨されません。ただし、元のチャンネルにいつでも安全に戻ることができます。

3.4. CLI を使用したクラスタの更新

更新が利用可能な場合、OpenShift CLI (**oc**) を使用してクラスタを更新できます。

利用可能な OpenShift Container Platform アドバイザリーおよび更新については、カスタマーポータル [の エラータ](#) のセクションを参照してください。

前提条件

- お使いの更新バージョンのバージョンに一致する OpenShift Command-line Interface (CLI) (**oc** として一般に知られる) のバージョンをインストールします。
- **cluster-admin** 権限を持つユーザーとしてクラスタにログインします。

- **jq** パッケージをインストールします。

手順

1. クラスターが利用可能であることを確認します。

```
$ oc get clusterversion

NAME     VERSION  AVAILABLE  PROGRESSING  SINCE   STATUS
version  4.3.0    True       False        158m   Cluster version is 4.3.0
```

2. 現在の更新チャンネル情報を確認し、チャンネルが **stable-4.3** に設定されていることを確認します。

```
$ oc get clusterversion -o json|jq ".items[0].spec"

{
  "channel": "stable-4.3",
  "clusterID": "990f7ab8-109b-4c95-8480-2bd1deec55ff",
  "upstream": "https://api.openshift.com/api/upgrades_info/v1/graph"
}
```



重要

実稼働クラスターの場合、**stable-*** チャンネルにサブスクライブする必要があります。

3. 利用可能な更新を確認し、適用する必要がある更新のバージョン番号をメモします。

```
$ oc adm upgrade

Cluster version is 4.1.0

Updates:

VERSION IMAGE
4.1.2 quay.io/openshift-release-dev/ocp-
release@sha256:9c5f0df8b192a0d7b46cd5f6a4da2289c155fd5302dec7954f8f06c878160b8b
```

4. 更新を適用します。

- 最新バージョンに更新するには、以下を実行します。

```
$ oc adm upgrade --to-latest=true ①
```

- 特定のバージョンに更新するには、以下を実行します。

```
$ oc adm upgrade --to=<version> ①
```

① ① **<version>** は、直前のコマンドの出力から得られる更新バージョンです。

5. クラスターバージョン Operator を確認します。

```
$ oc get clusterversion -o json|jq ".items[0].spec"
{
  "channel": "stable-4.3",
  "clusterID": "990f7ab8-109b-4c95-8480-2bd1deec55ff",
  "desiredUpdate": {
    "force": false,
    "image": "quay.io/openshift-release-dev/ocp-
release@sha256:9c5f0df8b192a0d7b46cd5f6a4da2289c155fd5302dec7954f8f06c878160b8b",

    "version": "4.3.1" ❶
  },
  "upstream": "https://api.openshift.com/api/upgrades_info/v1/graph"
}
```

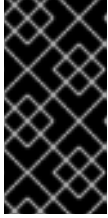
- ❶ **desiredUpdate** スタンザの **version** 番号が指定した値と一致する場合、更新は進行中です。

6. クラスタバージョン履歴で、更新のステータスをモニターします。すべてのオブジェクトの更新が終了するまでに時間がかかる可能性があります。

```
$ oc get clusterversion -o json|jq ".items[0].status.history"
[
  {
    "completionTime": null,
    "image": "quay.io/openshift-release-dev/ocp-
release@sha256:9c5f0df8b192a0d7b46cd5f6a4da2289c155fd5302dec7954f8f06c878160b8b",

    "startedTime": "2019-06-19T20:30:50Z",
    "state": "Partial",
    "verified": true,
    "version": "4.1.2"
  },
  {
    "completionTime": "2019-06-19T20:30:50Z",
    "image": "quay.io/openshift-release-dev/ocp-
release@sha256:b8307ac0f3ec4ac86c3f3b52846425205022da52c16f56ec31cbe428501001d6
",
    "startedTime": "2019-06-19T17:38:10Z",
    "state": "Completed",
    "verified": false,
    "version": "4.1.0"
  }
]
```

履歴には、クラスタに適用された最新バージョンの一覧が含まれます。この値は、CVO が更新を適用する際に更新されます。この一覧は日付順に表示され、最新の更新は一覧の先頭に表示されます。履歴の更新には、ロールアウトが完了した場合には **Completed** と表示され、更新が失敗したか、または完了しなかった場合には **Partial** と表示されます。



重要

アップグレードに失敗する場合、Operator は停止し、失敗しているコンポーネントのステータスを報告します。クラスターの以前のバージョンへのロールバックはサポートされていません。アップグレードできない場合は、Red Hat サポートにお問い合わせください。

- 更新が完了したら、クラスターのバージョンが新たなバージョンに更新されていることを確認できます。

```
$ oc get clusterversion
```

```
NAME      VERSION  AVAILABLE  PROGRESSING  SINCE   STATUS
version  4.1.2    True       False        2m     Cluster version is 4.1.2
```

- OpenShift Container Platform 4.3.3 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。これは以下のコマンドで確認できます。

```
$ for I in $(oc get ns -o jsonpath='{range .items[*]} {.metadata.name}{"\n"} {end}'); \
do oc delete pods --all -n $I; \
sleep 1; \
done
```



注記

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

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

第4章 RHEL コンピュータマシンを含むクラスタの更新

OpenShift Container Platform クラスタの更新またはアップグレードを実行できます。クラスタに Red Hat Enterprise Linux (RHEL) マシンが含まれる場合は、それらのマシンを更新するために追加の手順を実行する必要があります。

4.1. 前提条件

- **admin** 権限を持つユーザーとしてクラスタにアクセスできること。「[Using RBAC to define and apply permissions](#)」を参照してください。
- アップグレードが失敗し、[クラスタを直前の状態に復元する](#)必要がある場合に最新の `etcd` [バックアップ](#)があること。

重要

OpenShift Container Platform 4.2 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。

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

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

4.2. OPENSIFT CONTAINER PLATFORM の更新サービスについて

OpenShift Container Platform の更新サービスとは、OpenShift Container Platform と Red Hat Enterprise Linux CoreOS (RHCOS) の両方に OTA(over-the-air) 更新を提供するホスト型サービスです。コンポーネント Operator のグラフ、または[頂点とそれらを結ぶ辺](#)を含む図表が提示されます。グラフの辺は、どのバージョンであれば安全に更新できるかを示します。頂点は、管理されたクラスタコンポーネントの想定された状態を特定する更新のペイロードです。

クラスタ内の Cluster Version Operator (CVO) は、OpenShift Container Platform の更新サービスをチェックして、グラフの現在のコンポーネントバージョンとグラフの情報に基づき、有効な更新および更新パスを確認します。ユーザーが更新をリクエストすると、OpenShift Container Platform CVO はその更新のリリースイメージを使ってクラスタをアップグレードします。リリースアーティファクトは、コンテナイメージとして Quay でホストされます。

OpenShift Container Platform 更新サービスが互換性のある更新のみを提供できるようにするために、自動化を支援するリリース検証 Pipeline が使用されます。それぞれのリリースアーティファクトについて、他のコンポーネントパッケージだけでなくサポートされているクラウドプラットフォームおよびシステムアーキテクチャとの互換性の有無が検証されます。Pipeline がリリースの適合性を確認した後、OpenShift Container Platform 更新サービスは更新が利用可能であることを通知します。

重要

更新サービスが有効な更新をすべて表示するために、更新サービスが表示しないバージョンへの更新を強制することはできません。

連続更新モードでは、2つのコントローラーが実行されます。1つのコントローラーはペイロードマニ

フェストを絶えず更新し、それらをクラスターに適用し、Operator が利用可能か、アップグレード中か、または失敗しているかに応じて Operator の制御されたロールアウトのステータスを出力します。2 つ目のコントローラーは OpenShift Container Platform 更新サービスをポーリングして、更新が利用可能かどうかを判別します。



重要

クラスターを以前のバージョンに戻すこと、つまりロールバックはサポートされていません。サポートされているのは、新規バージョンへのアップグレードのみです。

アップグレードプロセスで、Machine Config Operator (MCO) は新規設定をクラスターマシンに適用します。これは、マシン設定プールの **maxUnavailable** フィールドによって指定されるノードの数を分離し、それらを利用不可としてマークします。デフォルトで、この値は **1** に設定されます。次に、新しい設定を適用して、マシンを再起動します。Red Hat Enterprise Linux (RHEL) マシンをワーカーとして使用する場合、まず OpenShift API をそれらのマシンで更新する必要があるため、MCO はそれらのマシンで kubelet を更新しません。新規バージョンの仕様は古い kubelet に適用されるため、RHEL マシンを **Ready** 状態に戻すことができません。マシンが利用可能になるまで更新を完了することはできません。ただし、利用不可のノードの最大数は、その数のマシンがサービス停止状態のマシンとして分離されても通常のクラスター操作が継続できるようにするために設定されます。

追加リソース

- [管理外の Operator のサポートポリシー](#)

4.3. OPENSIFT CONTAINER PLATFORM アップグレードチャンネルおよびリリース

OpenShift Container Platform 4.1 で、Red Hat はクラスターのアップグレードの適切なリリースバージョンを推奨するためにチャンネルという概念を導入しました。アップグレードのペースを制御することで、これらのアップグレードチャンネルからアップグレードストラテジーを選択することができます。アップグレードチャンネルは OpenShift Container Platform のマイナーバージョンに関連付けられます。たとえば、OpenShift Container Platform 4.3 アップグレードチャンネルには 4.4 リリースへのアップグレードが含まれることはありません。このストラテジーにより、管理者は OpenShift Container Platform の次のマイナーバージョンへのアップグレードに関して明確な決定を行うことができます。アップグレードチャンネルはリリースの選択のみを制御し、インストールするクラスターのバージョンには影響を与えません。OpenShift Container Platform の特定のバージョンの **openshift-install** バイナリーファイルは常に該当バージョンをインストールします。

OpenShift Container Platform 4.3 は以下のアップグレードチャンネルを提供します。

- **candidate-4.3**
- **fast-4.3**
- **stable-4.3**

candidate-4.3 チャンネル

candidate-4.3 チャンネルには、z-stream (4.3.z) リリースの候補となるビルドが含まれます。リリース候補には、製品のすべての機能が含まれますが、それらがサポートされる訳ではありません。リリース候補を使用して機能の受け入れテストを実行し、OpenShift Container Platform の次のバージョンへの対応を支援します。リリース候補は、名前に **-rc** を含まないものを含め、候補チャンネルで利用可能なビルドを指します。候補チャンネルでバージョンが利用可能になると、さらに品質のチェックが行われます。品質基準を満たす場合には、これは **fast-4.3** または **stable-4.3** チャンネルにプロモートされます。このストラテジーにより、特定のリリースが **candidate-4.3** チャンネルと **fast-4.3** または **stable-4.3** チャンネルの

両方で利用可能な場合、そのリリースは Red Hat でサポートされるバージョンということになります。**candidate-4.3** チャンネルには、いずれのチャンネルでも推奨されている更新のないリリースバージョンを含めることができます。

candidate-4.3 チャンネルを使用して、OpenShift Container Platform の直前のマイナーバージョンからアップグレードできます。



注記

リリース候補は、<https://www.openshift.com/try> サイトにある夜間バッチとは異なります。夜間ビルドは各種機能への早期アクセスのために利用できますが、夜間バッチへの/からの更新は推奨されておらず、サポートもされていません。夜間ビルドはいずれのアップグレードチャンネルでも利用できません。

fast-4.3 チャンネル

fast-4.3 チャンネルは、Red Hat が一般公開リリースとして指定のバージョンを宣言するとすぐに新しいバージョンで更新されます。そのため、これらのリリースは完全にサポートされ、実稼働用の品質があり、これらのリリースのプロモート元の **candidate-4.3** チャンネルのリリース候補として利用可能であった間のパフォーマンスにも問題がなかったリリースです。リリースは **fast-4.3** チャンネルに表示されてからしばらくすると、**stable-4.3** チャンネルに追加されます。リリースは **fast-4.3** チャンネルに表示される前に、**stable-4.3** チャンネルに表示されることはありません。

fast-4.3 チャンネルを使用して、OpenShift Container Platform の以前のマイナーバージョンからのアップグレードを実行できます。

stable-4.3 チャンネル

fast-4.3 チャンネルにはエラータの公開後すぐにリリースが組み込まれ、リリースの **stable-4.3** チャンネルへの追加は遅延します。この期間中、接続環境のカスタマープログラム(Connected Customer Program)に関わる Red Hat SRE チーム、Red Hat サポートサービス、および実稼働前および実稼働環境からリリースの安定性についてのデータが収集されます。

stable-4.3 チャンネルを使用して、OpenShift Container Platform の以前のマイナーバージョンからのアップグレードを実行できます。

アップグレードバージョンパス

OpenShift Container Platform では、インストールされた OpenShift Container Platform のバージョンと、次のリリースにアクセスするために選択したチャンネル内のパスの確認を可能にするアップグレード推奨サービスが提供されます。**fast-4.3** チャンネルでは以下を確認できます。

- 4.3.0
- 4.3.1
- 4.3.3
- 4.3.4

このサービスは、テスト済みの重大な問題のないアップグレードのみを推奨します。クラスタが 4.3.1 にあり、OpenShift Container Platform が 4.3.4 を提案している場合、4.3.1 から 4.3.4 に更新しても問題がありません。パッチの連続する番号のみに依存しないようにしてください。たとえば、この例では 4.3.2 はチャンネルで利用可能な状態ではなく、これまで利用可能になったことがありません。更新サービスは、既知の脆弱性を含む OpenShift Container Platform のバージョンへの更新を提案しません。

更新の安定性は、チャンネルによって異なります。**candidate-4.3** チャンネルに更新についての推奨があるからといって、その更新が必ずしもサポートされる訳ではありません。つまり、更新について深刻な問題がまだ検出されていないものの、この更新の安定性についての提案を導くようなトラフィックの安定

性はとくに確認されていない可能性があります。**fast-4.3** または **stable-4.3** チャンネルに更新の推奨がある場合は、更新がそれぞれのチャンネルにある限り完全にサポートされることを示します。リリースがチャンネルから削除されることは決してありませんが、深刻な問題を示す更新の推奨はすべてのチャンネルから削除されます。更新の推奨が削除された後に開始された更新はサポートされない可能性があります。

Red Hat は最終的には、**fast-4.3** または **stable-4.3** チャンネルのサポートされるリリースから 4.3.z の最新リリースへのサポートされる更新パスを提供します。ただし、問題のあるリリースからの安全なパスが構築され、検証される間に遅延が生じる可能性があります。

高速かつ安定したチャンネルの使用およびストラテジー

fast-4.3 および **stable-4.3** チャンネルでは、一般公開リリースが利用可能になり次第これを受信するか、または Red Hat がそれらの更新のロールアウトを制御するようにするかを選択することができます。問題がロールアウト時またはロールアウト後に検出される場合、該当バージョンへのアップグレードは **fast-4.3** および **stable-4.3** チャンネルの両方でブロックされ、新たに推奨されるアップグレード先の新規バージョンが導入される可能性があります。

fast-4.3 チャンネルで実稼働前のシステムを設定し、**stable-4.3** チャンネルで実稼働システムを設定してから Red Hat の接続環境のカスタマープログラム(Connected Customer Program)に参加することで、お客様のプロセスを改善することができます。Red Hat はこのプログラムを使用して、ご使用の特定のハードウェアおよびソフトウェア設定に対する更新の影響の有無を確認します。今後のリリースでは、更新が **fast-4.3** から **stable-4.3** チャンネルに移行するペースが改善されるか、変更される可能性があります。

ネットワークが制限された環境のクラスター

OpenShift Container Platform クラスターのコンテナイメージを独自に管理する場合には、製品リリースに関連する Red Hat エラータを確認し、アップグレードへの影響に関するコメントに留意する必要があります。アップグレード時に、インターフェースにこれらのバージョン間の切り替えについての警告が表示される場合があります。そのため、これらの警告を無視するかどうかを決める前に適切なバージョンを選択していることを確認する必要があります。

チャンネル間の切り替え

stable-4.3 チャンネルから **fast-4.3** チャンネルに切り換える場合も、クラスターは引き続きサポートされます。**candidate-4.3** チャンネルに常に切り替えることはできますが、そのチャンネルの一部のリリースはサポートされないリリース候補である可能性があります。現在のリリースが一般利用公開リリースの場合、**candidate-4.3** チャンネルから **fast-4.3** チャンネルに切り換えることができます。**fast-4.3** チャンネルから **stable-4.3** チャンネルに常に切り換えることができますが、現在のリリースが **fast-4.3** に最近プロモートされた場合、リリースが **stable-4.3** にプロモートされるまでに最長1日分の遅延が生じる可能性があります。現在のリリースを含まないチャンネルに変更すると、アラートが表示され、更新は推奨されません。ただし、元のチャンネルにいつでも安全に戻ることができます。

4.4. WEB コンソールを使用したクラスターの更新

更新が利用可能な場合、Web コンソールからクラスターを更新できます。

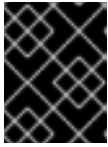
利用可能な OpenShift Container Platform アドバイザリーおよび更新については、カスタマーポータル の [エラータ](#) のセクションを参照してください。

前提条件

- **admin** 権限を持つユーザーとして Web コンソールにアクセスできること。

手順

1. Web コンソールから、**Administration** > **Cluster Settings** をクリックし、**Overview** タブの内容を確認します。
2. 実稼働クラスターの場合、**CHANNEL** が **stable-4.3** などの現在のマイナーバージョンの、更新する必要のあるバージョンの正しいチャンネルに設定されていることを確認します。



重要

実稼働クラスターの場合、stable-* または fast-* チャンネルにサブスクライブする必要があります。

- **UPDATE STATUS** が **Updates Available** ではない場合、クラスターをアップグレードすることはできません。
 - **DESIRED VERSION** は、クラスターが実行されているか、または更新されるクラスターのバージョンを示します。
3. **Updates Available** をクリックし、更新するバージョンで、利用可能な最新バージョンを選択し、**Update** をクリックします。**UPDATE STATUS** は **Updating** に切り替わり、**Cluster Operators** タブで Operator のアップグレードの進捗を確認できます。
 4. OpenShift Container Platform 4.2 以前から本リリースにアップグレードする場合、アップグレードの完了後にすべての Pod を再起動する必要があります。OpenShift CLI (**oc**) を必要とする以下のコマンドを使用してこれを実行できます。

```
$ for I in $(oc get ns -o jsonpath='{range .items[*]} {.metadata.name}{"\n"} {end}'); \
do oc delete pods --all -n $I; \
sleep 1; \
done
```



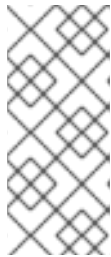
注記

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

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

5. 更新が完了し、Cluster Version Operator が利用可能な更新を更新したら、追加の更新が現在のチャンネルで利用可能かどうかを確認します。
 - 更新が利用可能な場合は、更新ができなくなるまで、現在のチャンネルでの更新を続けます。
 - 利用可能な更新がない場合は、**CHANNEL** を次のマイナーバージョンの stable-* または fast-* チャンネルに切り替え、そのチャンネルに必要なバージョンに更新します。

必要なバージョンに達するまで、いくつかの中間更新を実行する必要がある場合があります。



注記

Red Hat Enterprise Linux (RHEL) ワーカーマシンを含むクラスターを更新する場合、それらのワーカーは、更新プロセス時に一時的に使用できなくなります。クラスターの更新の終了において各 RHEL マシンの状態が **NotReady** になる際に、アップグレード Playbook を各 RHEL マシンに対して実行する必要があります。

4.5. (オプション) RHEL マシンで ANSIBLE タスクを実行するためのフックの追加

OpenShift Container Platform の更新時にフックを使用し、RHEL コンピュータマシンで Ansible タスクを実行できます。

4.5.1. アップグレード用の Ansible Hook について

OpenShift Container Platform の更新時にフックを使用し、特定操作の実行中に Red Hat Enterprise Linux (RHEL) ノードでカスタムタスクを実行できます。フックを使用して、特定の更新タスクの前後に実行するタスクを定義するファイルを指定できます。OpenShift Container Platform クラスターで RHEL コンピュータノードを更新する際に、フックを使用してカスタムインフラストラクチャーを検証したり、変更したりすることができます。

フックが失敗すると操作も失敗するため、フックはべき等性があるか、または複数回実行でき、同じ結果を出せるように設計する必要があります。

フックには以下のような重要な制限があります。まず、フックには定義された、またはバージョン付けされたインターフェースがありません。フックは内部の **openshift-ansible** 変数を使用できますが、これらの変数は今後の OpenShift Container Platform のリリースで変更されるか、または削除される予定です。次に、フックにはエラー処理機能がないため、フックにエラーが生じると更新プロセスが中止されます。エラーの発生時には、まず問題に対応してからアップグレードを再び開始する必要があります。

4.5.2. Ansible インベントリーファイルでのフックを使用する設定

Red Hat Enterprise Linux (RHEL) コンピュータマシン (ワーカーマシンとしても知られている) の更新時に使用するフックを、**all:vars** セクションの下にある **hosts** インベントリーファイルで定義します。

前提条件

- RHEL コンピュータマシンクラスターの追加に使用したマシンへのアクセスがあること。RHEL マシンを定義する **hosts** Ansible インベントリーファイルにアクセスできる必要があります。

手順

1. フックの設計後に、フック用に Ansible タスクを定義する YAML ファイルを作成します。このファイルは、以下に示すように一連のタスクで構成される必要があり、Playbook にすることはできません。

```
---
# Trivial example forcing an operator to acknowledge the start of an upgrade
# file=/home/user/openshift-ansible/hooks/pre_compute.yml

- name: note the start of a compute machine update
  debug:
```

```
msg: "Compute machine upgrade of {{ inventory_hostname }} is about to start"
```

```
- name: require the user agree to start an upgrade
```

```
pause:
```

```
prompt: "Press Enter to start the compute machine update"
```

2. **hosts** Ansible インベントリーファイルを変更してフックファイルを指定します。フックファイルは、以下に示すように **[all:vars]** セクションのパラメーター値として指定されます。

インベントリーファイルのフック定義の例

```
[all:vars]
```

```
openshift_node_pre_upgrade_hook=/home/user/openshift-ansible/hooks/pre_node.yml
```

```
openshift_node_post_upgrade_hook=/home/user/openshift-ansible/hooks/post_node.yml
```

フックへのパスでの曖昧さを避けるために、それらの定義では相対パスの代わりに絶対パスを使用します。

4.5.3. RHEL コンピュータマシンで利用できるフック

Red Hat Enterprise Linux (RHEL) コンピュータマシンを OpenShift Container Platform クラスターで更新する際に、以下のフックを使用できます。

フック名	説明
openshift_node_pre_cordon_hook	<ul style="list-style-type: none"> 各ノードの遮断 (cordon) 前 に実行されます。 このフックは 各ノード に対して連続して実行されます。 タスクが異なるホストに対して実行される必要がある場合、そのタスクは delegate_to または local_action を使用する必要があります。
openshift_node_pre_upgrade_hook	<ul style="list-style-type: none"> 各ノードの遮断 (cordon) 後、更新 前 に実行されます。 このフックは 各ノード に対して連続して実行されます。 タスクが異なるホストに対して実行される必要がある場合、そのタスクは delegate_to または local_action を使用する必要があります。

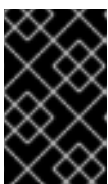
フック名	説明
openshift_node_pre_uncordon_hook	<ul style="list-style-type: none"> ● 各ノードの更新後、遮断の解除 (uncordon) 前に実行されます。 ● このフックは各ノードに対して連続して実行されます。 ● タスクが異なるホストに対して実行される必要がある場合、そのタスクは delegate_to または local_action を使用する必要があります。
openshift_node_post_upgrade_hook	<ul style="list-style-type: none"> ● 各ノードの遮断の解除 (uncordon) 後に実行されます。これは、最後のノード更新アクションになります。 ● このフックは各ノードに対して連続して実行されます。 ● タスクが異なるホストに対して実行される必要がある場合、そのタスクは delegate_to または local_action を使用する必要があります。

4.6. クラスター内の RHEL コンピュータマシンの更新

クラスターの更新後は、クラスター内の Red Hat Enterprise Linux (RHEL) コンピュータマシンを更新する必要があります。

前提条件

- クラスターが更新されていること。



重要

RHEL マシンには、更新プロセスを完了するためにクラスターで生成されるアセットが必要になるため、クラスターを更新してから、クラスター内の RHEL コンピュータマシンを更新する必要があります。

- RHEL コンピュータマシンクラスターの追加に使用したマシンへのアクセスがあること。RHEL マシンを定義する **hosts** Ansible インベントリーファイルおよび **upgrade** Playbook にアクセスできる必要があります。

手順

1. ホストで `firewalld` を停止し、無効にします。

```
# systemctl disable --now firewalld.service
```




注記

firewalld は、後で有効にすることはできません。これを実行する場合、ワーカー上の OpenShift Container Platform ログにはアクセスできません。

2. OpenShift Container Platform 4.3 で必要なりポジトリーを有効にします。

- a. Ansible Playbook を実行するマシンで、必要なりポジトリーを更新します。

```
# subscription-manager repos --disable=rhel-7-server-ansible-2.7-rpms \
--disable=rhel-7-server-ose-4.2-rpms \
--enable=rhel-7-server-ansible-2.8-rpms \
--enable=rhel-7-server-ose-4.3-rpms
```

- b. Ansible Playbook を実行するマシンで、**openshift-ansible**を含む必要なパッケージを更新します。

```
# yum update openshift-ansible openshift-clients
```

- c. 各 RHEL コンピュータノードで、必要なりポジトリーを更新します。

```
# subscription-manager repos --disable=rhel-7-server-ose-4.2-rpms \
--enable=rhel-7-server-ose-4.3-rpms
```

3. RHEL ワーカーマシンを更新します。

- a. 現在のノードステータスを確認し、更新する RHEL ワーカーを判別します。

```
# oc get node
NAME                STATUS              ROLES    AGE   VERSION
mycluster-control-plane-0 Ready              master   145m v1.16.2
mycluster-control-plane-1 Ready              master   145m v1.16.2
mycluster-control-plane-2 Ready              master   145m v1.16.2
mycluster-rhel7-0   NotReady,SchedulingDisabled worker   98m
v1.14.6+97c81d00e
mycluster-rhel7-1   Ready              worker   98m  v1.14.6+97c81d00e
mycluster-rhel7-2   Ready              worker   98m  v1.14.6+97c81d00e
mycluster-rhel7-3   Ready              worker   98m  v1.14.6+97c81d00e
```

ステータスが **NotReady,SchedulingDisabled** のマシンに留意してください。

- b. `/<path>/inventory/hosts` で Ansible インベントリーファイルを確認し、以下の例に示されるように、ステータスが **NotReady,SchedulingDisabled** のマシンのみが **[workers]** セクションに一覧表示されるようにそのコンテンツを更新します。

```
[all:vars]
ansible_user=root
#ansible_become=True

openshift_kubeconfig_path=~/.kube/config"

[workers]
mycluster-rhel7-0.example.com
```

- c. **openshift-ansible** ディレクトリーに切り替え、**upgrade** Playbook を実行します。

```
$ cd /usr/share/ansible/openshift-ansible  
$ ansible-playbook -i <path>/inventory/hosts playbooks/upgrade.yml 1
```

- 1 <path> については、作成した Ansible インベントリーファイルへのパスを指定します。

4. 直前の手順で実行したプロセスに従って、クラスター内の各 RHEL ワーカーマシンを更新します。
5. すべてのワーカーを更新したら、すべてのクラスターノードが新規バージョンに更新されていることを確認します。

```
# oc get node  
NAME                STATUS                ROLES  AGE  VERSION  
mycluster-control-plane-0  Ready                master  145m  v1.16.2  
mycluster-control-plane-1  Ready                master  145m  v1.16.2  
mycluster-control-plane-2  Ready                master  145m  v1.16.2  
mycluster-rhel7-0        NotReady,SchedulingDisabled  worker  98m  v1.16.2  
mycluster-rhel7-1        Ready                worker  98m  v1.16.2  
mycluster-rhel7-2        Ready                worker  98m  v1.16.2  
mycluster-rhel7-3        Ready                worker  98m  v1.16.2
```