

vSphere IaaS 制御プレーン の更新

Update 2

VMware vSphere 8.0

VMware vCenter 8.0

VMware ESXi 8.0

VMware by Broadcom の Web サイトで最新の技術ドキュメントを確認できます

<https://docs.vmware.com/jp/>

VMware by Broadcom

3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2024 Broadcom. All Rights Reserved. 「Broadcom」という語表現は、Broadcom Inc. およびその子会社のいずれかまたは両方を指します。詳細については、<https://www.broadcom.com> を参照してください。本書に記載されるすべての商標、製品名、サービス マークおよびロゴは、各社に帰属します。

目次

vSphere IaaS 制御プレーンの更新 4

1 vSphere IaaS control plane 環境の更新 5

vSphere IaaS control plane の更新の仕組み 5

スーパーバイザー の更新 10

Object Missing 12

有効化または更新中のスーパーバイザー制御プレーン仮想マシンの健全性ステータス エラーの解決 12

スーパーバイザー の自動アップグレード 16

kubectl 向けの vSphere プラグイン の更新 17

2 TKG サービスのインストールとアップグレード 18

TKG サービス の使用 18

TKG サービス のステータスの確認 20

新しいバージョンの TKG サービスの登録 21

TKG サービス バージョンのアップグレード 21

TKG サービスのトラブルシューティング 22

3 TKG サービス クラスタの更新 23

TKG サービス クラスタのローリング アップデート モデルについて 23

更新のための TKGS クラスタ互換性の確認 27

TKR バージョンの編集による TKG クラスタの更新 28

ストレージ クラスの編集による TKG クラスタの更新 32

仮想マシン クラスの編集による TKG サービス クラスタの更新 34

Tanzu CLI を使用した TKG クラスタの更新 36

4 vSphere Lifecycle Manager の操作 38

要件 38

vSphere Lifecycle Manager によって管理されているクラスタでの vSphere IaaS control plane の有効化 39

スーパーバイザー のアップグレード 39

スーパーバイザー へのホストの追加 40

スーパーバイザー からのホストの削除 41

スーパーバイザー クラスタの無効化 41

vSphere IaaS 制御プレーンの更新

『vSphere IaaS 制御プレーンの更新』では、最新の製品バージョンと Kubernetes リリースを使用して、vSphere IaaS control plane 環境を最新の状態に保つ方法について説明します。スーパーバイザー および Tanzu Kubernetes Grid の更新の仕組みと、VMware vSphere Lifecycle Manager を使用して環境のライフサイクルを管理する方法について説明します。

対象読者

この情報は、vSphere IaaS control plane 環境を最新の製品バージョンで最新の状態に維持する vSphere 管理者および DevOps エンジニアを対象としています。次の領域の知識が必要です。

- vSphere IaaS control plane
- Kubernetes
- vSphere

vSphere IaaS control plane 環境の更新

1

vSphere IaaS control plane の最新バージョンに更新できます。これには、スーパーバイザー、TKG サービス、Tanzu Kubernetes Grid クラスタをサポートする vSphere インフラストラクチャ、Kubernetes のバージョン、および vSphere 向け Kubernetes CLI Tools が含まれます。

注： vCenter Server を更新する前に、すべてのスーパーバイザーの Kubernetes バージョンがサポートされている最小バージョンであり、Tanzu Kubernetes Grid クラスタの Tanzu Kubernetes リリース バージョンがサポートされている最小バージョンであることを確認します。詳細については、[互換性マトリックス](#)を参照してください。

次のトピックを参照してください。

- [vSphere IaaS control plane の更新の仕組み](#)
- [スーパーバイザー の更新](#)
- [スーパーバイザー の自動アップグレード](#)
- [kubectI 向けの vSphere プラグイン の更新](#)

vSphere IaaS control plane の更新の仕組み

vSphere IaaS control plane 環境内のスーパーバイザー および Tanzu Kubernetes Grid クラスタを更新する方法を確認します。vSphere IaaS control plane は、スーパーバイザー と Tanzu Kubernetes Grid クラスタ、およびこれらのクラスタをサポートするインフラストラクチャのローリング アップデートに対応しています。

注： vSphere Update Manager を使用するスーパーバイザーを vSphere Lifecycle Manager に移行することはできません。vSphere Update Manager から vSphere Lifecycle Manager への移行は、vSphere IaaS control plane が有効になっていないクラスタでのみサポートされます。

vSphere IaaS control plane 環境を更新する場合は、基盤となるインフラストラクチャ コンポーネントおよびサービスとともに、スーパーバイザー および Tanzu Kubernetes Grid クラスタの Kubernetes バージョンを更新します。したがって、このプロセスの説明には、ソフトウェア バージョンの更新に限定されるアップグレードという言葉ではなく、更新という言葉を使用しています。

スーパーバイザー および Tanzu Kubernetes Grid クラスタは、共通の Kubernetes ディストリビューション コアを使用して構築されます。スーパーバイザー クラスタと Tanzu Kubernetes Grid クラスタの Kubernetes バージョンの配信方法は異なります。

- スーパーバイザー Kubernetes リリースは、vCenter Server リリースに付属しており、個人用の Kubernetes バージョンが含まれています。すべての vCenter Server リリース（メジャー リリース、アップデート リリース、またはパッチ リリース）には、スーパーバイザー 用の 3 つのバージョンの Kubernetes が含まれています。最新のスーパーバイザー Kubernetes バージョンと 2 つの以前のバージョン。サポートされているスーパーバイザー Kubernetes バージョンの詳細については、「[VMware vSphere with Tanzu 8.0 リリース ノート](#)」を参照してください。
- Tanzu Kubernetes Grid クラスタは、Tanzu Kubernetes リリース (TKr) を使用して構築されます。TKr は、アップストリームに統合した Kubernetes ソフトウェア配布を、VMware による署名、テスト、およびサポート済みの状態で提供します。TKr は、Tanzu Kubernetes Grid クラスタを対象としています。TKr は、vCenter Server とは別にリリースされます。TKr は、Photon や Ubuntu などのオペレーティング システムと、ポッド機能を提供するために必要な Tanzu Kubernetes Grid コア コンポーネントを組み合わせたものです。TKr には、Tanzu Kubernetes Grid クラスタにデプロイできる Antrea や Calico などのコンポーネントが含まれています。詳細については、[VMware Tanzu Kubernetes リリースのリリース ノート](#)を参照してください。

アップストリーム Kubernetes の更新ポリシーにより、スーパーバイザー および Tanzu Kubernetes Grid クラスタの Kubernetes バージョンのみを順次更新できます。マイナー バージョンはスキップできません。たとえば、スーパーバイザー が 1.24 を実行している場合、1.26 に直接更新することはできません。正しい更新パスは、1.24、1.25、1.26 です。

スーパーバイザー および Tanzu Kubernetes Grid クラスタ Kubernetes バージョンのサポート ポリシー

スーパーバイザー Kubernetes バージョンと TKr の両方で、アップストリーム Kubernetes サポート ポリシーと同様に、Kubernetes リリースに [N-2 サポート ポリシー](#)を実装します。つまり、vSphere IaaS control plane (スーパーバイザー または TKr) の各 Kubernetes リリースは、VMware によってリリースされた時点から少なくとも 12 か月間サポートされます。

ただし、スーパーバイザー および Tanzu Kubernetes Grid クラスタでは互換性のある Kubernetes バージョンを実行する必要があることに注意してください。詳細については、TKr リリースの[互換性マトリックスと #unique_4](#) を参照してください。

vCenter Server のアップグレード パス ルール

アップストリーム Kubernetes には順次アップグレードが必要であるため、スーパーバイザー を実行するときの vCenter Server のアップグレード パスは、含まれている Kubernetes のバージョンによって異なります。サポートされている Kubernetes バージョンを実行するには、vCenter Server の年間約 3 回の更新と、スーパーバイザー の Kubernetes バージョンの更新が必要になる場合があります。

スーパーバイザー が実行されている vCenter Server のアップグレードは、次のいずれかの条件が満たされた場合に可能です。

- ソースおよびターゲットの vCenter Server リリースに、少なくとも 1 つの重複するバージョンの Kubernetes が含まれています。
- ターゲット vCenter Server のバージョンには、ソース vCenter Server リリースに存在する Kubernetes の次のバージョンが含まれている必要があります。

これらの条件のいずれも満たされない場合は、vCenter Server をアップグレードできません。

表 1-1. vCenter Server アップグレードのシナリオの例

例	リリースの更新	vCenter Server リリースでサポートされているスーパーバイザー Kubernetes バージョン				
		移行元	1.22	1.23	[1.24]	アップグレード可能
例 1:	移行元 vCenter Server					
	ターゲット vCenter Server			[1.24]	1.25	1.26
例 2:	移行元 vCenter Server	1.21	1.22	[1.23]	アップグレード可能	
	ターゲット vCenter Server				[1.24]	1.25 1.26
例 3:	移行元 vCenter Server	1.20	1.21	[1.22]	アップグレード不可能	
	ターゲット vCenter Server				[1.24]	1.25 1.26

上記の例は、ソースおよびターゲットの vCenter Server システムに含まれるスーパーバイザー Kubernetes のバージョンに応じて、vCenter Server のアップグレードが可能かどうかを示しています。

- 1 最初の例では、ソースとターゲットの両方の vCenter Server バージョンにスーパーバイザー Kubernetes バージョン 1.24 が含まれているため、アップグレードが可能です。
- 2 2 番目の例では、ターゲットの vCenter Server バージョンに直接の次のバージョンであるスーパーバイザー Kubernetes バージョン 1.24 が含まれているため、こちらもアップグレードは可能です。
- 3 最後の例では、ソースとターゲットの vCenter Server バージョンに、重複するスーパーバイザー Kubernetes バージョンも、直接の次のバージョンも含まれていないため、アップグレードが不可能なシナリオを示しています。

vSphere 名前空間 バージョンと スーパーバイザー バージョンについて

各 vCenter Server バージョンには、1つの新しいスーパーバイザー バージョンと2つの以前サポートされていたバージョンを含む、新しい vSphere 名前空間 バージョンが付属しています。たとえば、vCenter Server 8 Update 3 で提供される vSphere 名前空間 バージョン 0.1.9 には、次の3つのスーパーバイザー バージョンが含まれています。

- v1.26.8+vmware.wcp.1-vsc0.1.9-23708114
- v1.27.5+vmware.wcp.1-vsc0.1.9-23708114
- v1.28.3+vmware.wcp.1-vsc0.1.9-23708114

スーパーバイザー バージョンの更新の詳細については、「[スーパーバイザー の更新](#)」を参照してください。

注： Tanzu Kubernetes Grid 3.0 を使用するには、スーパーバイザー を vSphere 名前空間 0.1.9 に含まれる3つのサポートされているバージョンのいずれかに更新する必要があります。

スーパーバイザー および Tanzu Kubernetes Grid クラスタのローリング アップデート

vSphere IaaS control plane は、スーパーバイザー および Tanzu Kubernetes Grid クラスタのローリング アップデート モデルを使用します。ローリング アップデート モデルを使用すると、更新プロセスでのクラスタのワークロードのダウンタイムが最小限に抑えられます。ローリング アップデートには、Kubernetes ソフトウェア バージョンのアップグレードに加えて、仮想マシンの構成とリソース、vSphere 名前空間、カスタム リソースなどの、Tanzu Kubernetes Grid クラスタをサポートするインフラストラクチャおよびサービスの更新が含まれます。詳細については、「[スーパーバイザー上の TKG クラスタのローリング アップデート モデルについて](#)」を参照してください。

更新を正常に行うには、使用環境がいくつかの互換性要件を満たしている必要があります。システムは事前チェック条件を適用してクラスタを更新する準備ができているかを確認し、クラスタのアップグレードが成功しなかった場合はロールバックをサポートします。

スーパーバイザー の更新

スーパーバイザー が実行されている Kubernetes バージョン (Kubernetes 1.23 から Kubernetes 1.24 など) と、スーパーバイザー および Tanzu Kubernetes Grid クラスタをサポートするインフラストラクチャを更新できます。Kubernetes リリースの頻度に合わせて、スーパーバイザー Kubernetes のバージョンの更新が年3回必要になる場合があります。スーパーバイザー Kubernetes バージョンの更新手順は次のとおりです。

- 1 [vCenter Server のアップグレード パス ルール](#) で説明されているルールに従って、vCenter Server をアップグレードします。
- 2 スーパーバイザー Kubernetes のバージョンとインフラストラクチャ コンポーネントを更新します。[スーパーバイザー の更新](#)を参照してください。

スーパーバイザー Kubernetes バージョンの更新を開始すると、更新は次の操作の順序で実行されます。

- 1 システムは新しい制御プレーン仮想マシンを作成し、これを既存のスーパーバイザー 制御プレーンに参加させます。更新のこの段階で新しい更新済みの仮想マシンが追加され、古い期限切れの仮想マシンが削除されるため、vSphere インベントリには4台の制御プレーン仮想マシンが表示されます。

- オブジェクトは、古い制御プレーン仮想マシンのいずれかから新しい仮想マシンに移行され、古い制御プレーン仮想マシンは削除されます。すべての制御プレーン仮想マシンが更新されるまで、このプロセスが1回ずつ繰り返されます。
- すべての制御プレーン仮想マシンが更新されると、同様のローリング アップデート方式でワーカー ノードが更新されます。ワーカー ノードは ESXi ホストであり、各 ESXi ホストの各 spherelet プロセスが1つずつ更新されます。

次の更新の中から選択できます。

- スーパーバイザー Kubernetes バージョンのみを更新します。
- VMware バージョンや Kubernetes バージョンなどを含め、すべてを更新します。

Tanzu Kubernetes Grid クラスタの更新

Tanzu Kubernetes Grid クラスタの更新は、スーパーバイザー で実行されている Tanzu Kubernetes Grid サービスのバージョンと、クラスタが実行されている TKr のバージョンによって異なります。詳細については、『2 章 TKG サービスのインストールとアップグレード』および『3 章 TKG サービス クラスタの更新』を参照してください。

すべての vSphere IaaS control plane コンポーネントの更新

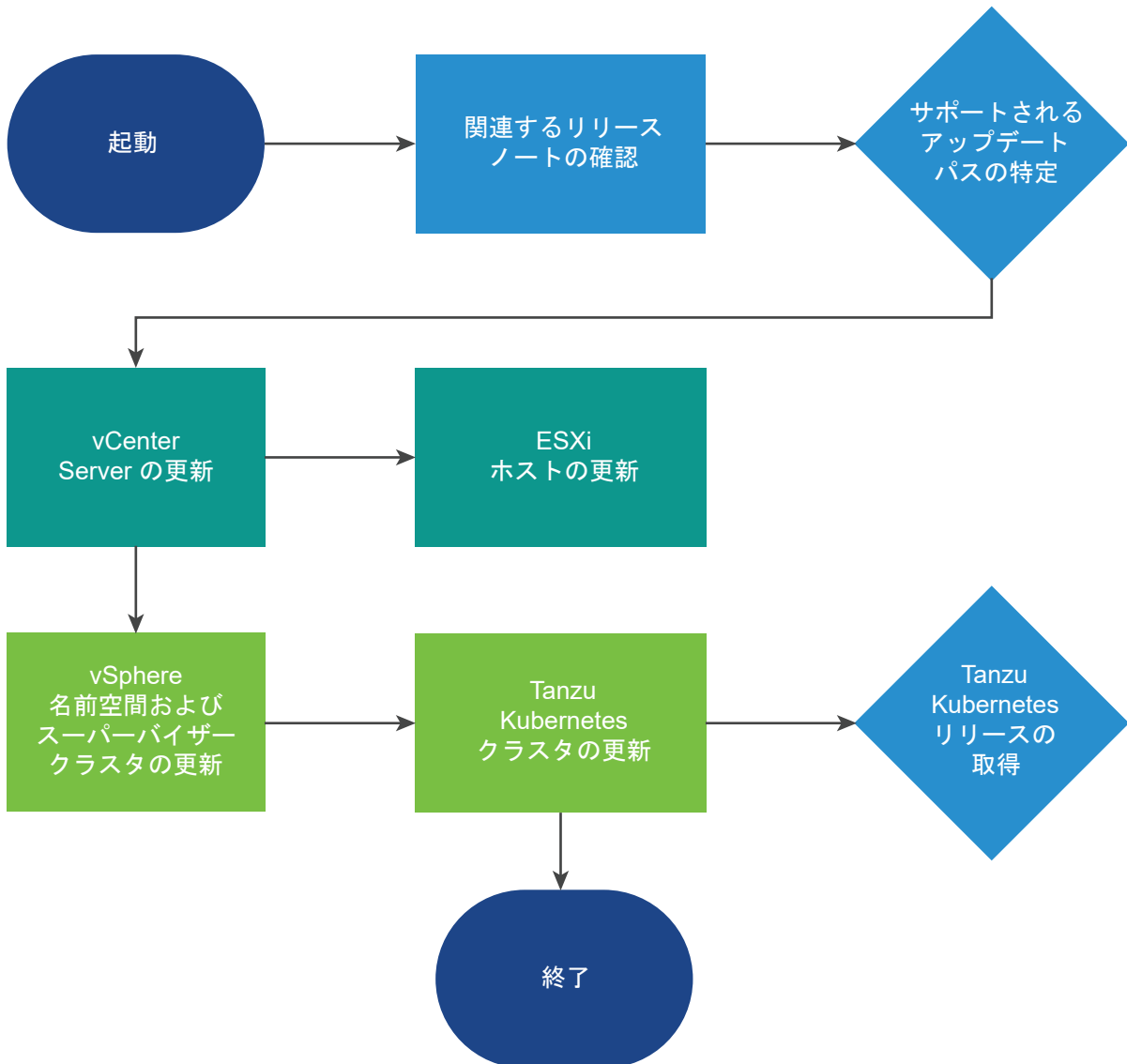
すべての vSphere IaaS control plane コンポーネントを更新するには、すべてを更新するワークフローを使用します。このタイプの更新は、NSX 3.X から 4、vSphere 7.x から 8 など、メジャー リリースを更新する場合があります。

注： vCenter Server 8.0 にアップグレードする前に、すべてのスーパーバイザー の Kubernetes バージョンが 1.22 以上（できればサポートされている最新バージョン）であること、および Tanzu Kubernetes Grid クラスタの Tanzu Kubernetes リリース バージョンが 1.22（できればサポートされている最新バージョン）であることを確認します。

この更新ワークフローは、新しい VMware 製品リリースの公開時期に応じて低い頻度で行われます。すべてを更新する手順は次のとおりです。

- 互換性を判断するには、VMware 製品の相互運用性マトリックス (<https://interopmatrix.vmware.com/Interoperability>) で vCenter Server と NSX を確認してください。vSphere IaaS control plane の機能は、vCenter Server に付属するワークロード制御プレーン (WCP) ソフトウェアによって提供されます。
- NSX に互換性がある場合は、アップグレードします。
- vCenter Server をアップグレードします。
- vSphere Distributed Switch をアップグレードします。
- ESXi ホストをアップグレードします。
- プロビジョニングされた Tanzu Kubernetes Grid クラスタとターゲット スーパーバイザー のバージョンの互換性を確認します。
- vSphere 名前空間 を更新します（スーパーバイザー Kubernetes バージョンを含む）。
- Tanzu Kubernetes Grid クラスタを更新します。

次の図に、vSphere IaaS control plane を更新する一般的なワークフローを示します



スーパーバイザー の更新

スーパーバイザー を更新する方法について、スーパーバイザー が実行されている Kubernetes バージョンを含めて説明します。環境内のスーパーバイザー は、常にサポートされている Kubernetes バージョンを実行する必要があります。

vSphere IaaS control plane のバージョン エンティティがあります。バージョン エンティティは `v1.28.3+vmware.wcp.1-vsc0.1.9-23708114` 形式のセマンティック バージョン文字列です。プリフィックスは Kubernetes バージョン (v1.28.3)、サフィックスは対応する vCenter Server バージョンとビルド番号でリリースされたスーパーバイザー リリース バージョン (vvsc0.1.9-23708114) です。

vCenter Server を新しいバージョンにアップグレードすると、vSphere 名前空間 バージョンも更新されます。各 vSphere 名前空間 バージョンには、新しい スーパーバイザー バージョンと、サポートされている 2 つの以前のバージョンが含まれています。

vCenter Server 8 Update 3 で提供される vSphere 名前空間 バージョン 0.1.9 には、次の 3 つのスーパーバイザー バージョンが含まれています。

- v1.26.8+vmware.wcp.1-vsc0.1.9-23708114
- v1.27.5+vmware.wcp.1-vsc0.1.9-23708114
- v1.28.3+vmware.wcp.1-vsc0.1.9-23708114

注： Tanzu Kubernetes Grid 3.0 を使用するには、スーパーバイザー を vSphere 名前空間 0.1.9 に含まれる 3 つのサポートされているバージョンのいずれかに更新する必要があります。

。

前提条件

- vSphere IaaS control plane リリース ノートで、スーパーバイザー でサポートされている Kubernetes バージョンを確認します。各 vCenter Server リリースには、スーパーバイザー 用の 3 つの Kubernetes バージョン（対応する vCenter Server バージョンでリリースされた スーパーバイザー の最新の Kubernetes バージョンと、以前の 2 つのバージョン）が含まれており、少なくとも 12 か月間サポートされます。
- スーパーバイザー で現在サポートされている Kubernetes バージョンをインストールします。それには、サポートされるバージョンを提供する vCenter Server バージョンに vCenter Server Appliance をアップグレードします。「[vCenter Server Appliance のアップグレード](#)」を参照してください。

注： スーパーバイザー を更新する場合、プロビジョニングされたすべての Tanzu Kubernetes Grid クラスタが vSphere 8 Update 3 の新しい スーパーバイザー K8s バージョンと互換性のある TKr バージョンを使用している必要があります。vSphere Update 3 以降では、Tanzu Kubernetes Grid は独立した スーパーバイザー サービス になるため、TKr バージョンは今後移行する Tanzu Kubernetes Grid サービスのバージョンによって異なります。

注： スーパーバイザー を更新すると、そこにデプロイされている Tanzu Kubernetes クラスタのローリング アップデートがトリガされることがあります。[TKG サービス クラスタのローリング アップデート モデルについて](#)を参照してください。

手順

- 1 vCenter Server に vSphere 管理者としてログインします。
- 2 [メニュー] - [ワークロー管理] の順に選択します。
- 3 [名前空間] - [更新] タブの順に選択します。

- 更新する [使用可能なバージョン] を選択します。

たとえば、バージョン `vv1.28.3+vmware.wcp.1-vsc0.1.9-23708114` を選択します。

注： アップデートは段階的に行う必要があります。1.26 から 1.28 へのアップデートなど、更新をスキップすることはできません。パスは 1.26、1.27、1.28 にする必要があります。

- 更新する スーパーバイザー を選択します。

- [更新の適用] をクリックします。

システムは一連の事前チェックを実行して、更新する スーパーバイザー Kubernetes バージョンに対するさまざまなコンポーネントの互換性を確認します。事前チェックが正常に完了したら、スーパーバイザー を更新できます。

Object Missing

This object is not available in the repository.

有効化または更新中のスーパーバイザー制御プレーン仮想マシンの健全性ステータス エラーの解決

スーパーバイザーを有効にするか、スーパーバイザー Kubernetes バージョンを更新するか、既存のスーパーバイザーの設定を編集すると、指定したすべての設定が検証され、構成が完了するまでスーパーバイザーに適用されます。健全性チェックは、入力したパラメータに対して実行されます。これにより、構成内のエラーが検出されて、スーパーバイザー の健全性ステータスがエラーになることがあります。これらの健全性ステータス エラーを解決して、スーパーバイザーを構成または更新できるようにする必要があります。

表 1-2. vCenter Server 接続エラー

エラー メッセージ	原因	解決方法
制御プレーン仮想マシン <仮想マシン名> に構成された管理 DNS サーバで vCenter Server プライマリ ネットワーク識別子 <FQDN> を解決できません。管理 DNS サーバ <サーバ名> で <ネットワーク名> が解決できることを検証してください。	<ul style="list-style-type: none"> ■ 1 台以上の管理 DNS サーバにアクセスできません。 ■ 1 つの管理 DNS が静的に提供されています。 ■ 管理 DNS サーバには、vCenter Server PNID のホスト名ルックアップ機能がありません。 ■ vCenter Server PNID が固定 IP アドレスではなく、ドメイン名です。 	<ul style="list-style-type: none"> ■ vCenter Server PNID のホスト エントリを管理 DNS サーバに追加します。 ■ 構成された DNS サーバが正しいことを確認します。
制御プレーン仮想マシン <仮想マシン名> の管理ネットワーク上の DHCP を介して取得した DNS サーバで vCenter Server プライマリ ネットワーク識別子 <ネットワーク名> を解決できません。管理 DNS サーバで <ネットワーク名> を解決できることを検証してください。	<ul style="list-style-type: none"> ■ DHCP サーバによって提供される管理 DNS サーバ (1 台以上) にアクセスできません。 ■ 管理 DNS サーバが静的に提供されています。 ■ 管理 DNS サーバには、vCenter Server PNID のホスト名ルックアップ機能がありません。 ■ 管理 DNS サーバには、vCenter Server PNID のホスト名ルックアップ機能がありません。 ■ vCenter Server PNID が固定 IP アドレスではなく、ドメイン名です。 	<ul style="list-style-type: none"> ■ 構成された DHCP サーバによって提供される管理 DNS サーバに、vCenter Server PNID のホスト エントリを追加します。 ■ DHCP サーバによって提供される DNS サーバが正しいことを確認します。
管理 DNS サーバが構成されていないため、制御プレーン仮想マシン <仮想マシン名> でホスト <ホスト名> を解決できません。	<ul style="list-style-type: none"> ■ vCenter Server PNID が固定 IP アドレスではなく、ドメイン名です。 ■ DNS サーバが構成されていません。 	管理 DNS サーバを構成します。
制御プレーン仮想マシン <仮想マシン名> でホスト <ホスト名> を解決できません。ホスト名の末尾のトップ レベル ドメインが「.local」であるため、管理 DNS 検索ドメインに「local」を含める必要があります。	vCenter Server PNID にはトップ レベル ドメイン (TLD) として、 .local が含まれていますが、構成された検索ドメインには local が含まれていません。	管理 DNS 検索ドメインに local を追加します。

表 1-2. vCenter Server 接続エラー（続き）

エラー メッセージ	原因	解決方法
制御プレーン仮想マシン <仮想マシン名> から管理 DNS サーバ <サーバ名> に接続できません。接続は、ワークロード ネットワークを使用して試行されました。	<ul style="list-style-type: none"> ■ 管理 DNS サーバを vCenter Server に接続できません。 ■ 指定した <code>worker_dns</code> の値には、指定した管理 DNS の値が完全に含まれていません。つまり、スーパーバイザー は固定トラフィックをこれらの IP アドレスに送信するためのネットワーク インターフェイスを 1 つ選択する必要があるため、トラフィックはワークロード ネットワークを介してルーティングされます。 	<ul style="list-style-type: none"> ■ ワークロード ネットワークを調べて、構成された管理 DNS サーバにルーティングできることを確認します。 ■ 競合する IP アドレスがないため、ワークロード ネットワーク上の DNS サーバと他のサーバ間で代替ルーティングがトリガされる可能性がないことを確認します。 ■ 構成された DNS サーバが実際には DNS サーバであり、その DNS ポートがポート 53 でホストされていることを確認します。 ■ ワークロード DNS サーバが、制御プレーン仮想マシンの IP アドレス（ワークロード ネットワークによって提供される IP アドレス）からの接続を許可するように構成されていることを確認します。 ■ 管理 DNS サーバのアドレスに誤りがないことを確認します。 ■ 検索ドメインに、不要な「~」が含まれていないことを確認します。これは、ホスト名の誤った解決の原因になる可能性があります。
制御プレーン仮想マシン <仮想マシン名> から管理 DNS サーバ <サーバ名> に接続できません。	DNS サーバに接続できません。	<ul style="list-style-type: none"> ■ 管理ネットワークを調べて、管理 DNS サーバのルートが存在することを確認します。 ■ 競合する IP アドレスがないため、DNS サーバと他のサーバ間で代替ルーティングがトリガされる可能性がないことを確認します。 ■ 構成された DNS サーバが実際には DNS サーバであり、その DNS ポートがポート 53 でホストされていることを確認します。 ■ 管理 DNS サーバが、制御プレーン仮想マシンの IP アドレスからの接続を許可するように構成されていることを確認します。 ■ 管理 DNS サーバのアドレスに誤りがないことを確認します。 ■ 検索ドメインに、不要な「~」が含まれていないことを確認します。これは、ホスト名の誤った解決の原因になる可能性があります。

表 1-2. vCenter Server 接続エラー (続き)

エラー メッセージ	原因	解決方法
制御プレーン仮想マシン <仮想マシン名> から <コンポーネント名> <コンポーネントアドレス> に接続できません。エラー: エラーメッセージテキスト	<ul style="list-style-type: none"> ■ 一般的なネットワーク障害が発生しました。 ■ vCenter Server への実際の接続中にエラーが発生しました。 	<ul style="list-style-type: none"> ■ vCenter Server、HAProxy、NSX Manager、NSX Advanced Load Balancer などの構成済みコンポーネントのホスト名または IP アドレスが正しいことを検証します。 ■ 管理ネットワーク上の競合する IP アドレス、ファイアウォール ルールなどの外部ネットワーク設定を検証します。
制御プレーン仮想マシン <仮想マシン名> が vCenter Server <vCenter Server 名> 証明書を検証できません。vCenter Server 証明書は無効です。	vCenter Server から提供された証明書が無効な形式であるため、信頼できません。	<ul style="list-style-type: none"> ■ wcpssc を再起動して、制御プレーン仮想マシンの信頼されたルートバンドルの最新の vCenter Server ルート証明書を含む最新バンドルであることを確認します。 ■ vCenter Server 証明書が実際に有効な証明書であることを確認します。
制御プレーン仮想マシン <仮想マシン名> は vCenter Server <vCenter Server 名> 証明書を信頼していません。	<ul style="list-style-type: none"> ■ vCenter Server によって提示される vmca.pem 証明書は、制御プレーン仮想マシンに構成されているものとは異なります。 ■ vCenter Server アプライアンスの信頼されたルート証明書が置き換えられましたが、wcpssc は再起動しませんでした。 	<ul style="list-style-type: none"> ■ wcpssc を再起動して、制御プレーン仮想マシンの信頼されたルートバンドルの最新の vCenter Server 証明書のルートを含む最新バンドルであることを確認します。

表 1-3. NSX Manager 接続エラー

制御プレーン仮想マシン <仮想マシン名> が NSX サーバ <NSX サーバ名> の証明書を検証できませんでした。サーバ <NSX-T アドレス> から返されたサムプリントが、vCenter Server <vCenter Server 名> に登録された、想定されるクライアント証明書サムプリントと一致しません。	スーパーバイザーに登録された SSL サンプルが、NSX Manager によって提示された証明書の SHA-1 ハッシュと一致しません。	<ul style="list-style-type: none"> ■ NSX と vCenter Server インスタンス間にある NSX Manager で信頼を再度有効にします。 ■ vCenter Server で wcpssc を再起動します。
制御プレーン仮想マシン <仮想マシン名> から <コンポーネント名> <コンポーネントアドレス> に接続できません。エラー: エラーメッセージテキスト	一般的なネットワーク障害が発生しました。	<ul style="list-style-type: none"> ■ NSX Manager の管理ネットワークで、競合する IP アドレス、ファイアウォールルールなどの外部ネットワーク設定を検証します。 ■ NSX 拡張機能の NSX Manager の IP アドレスが正しいことを確認します。 ■ NSX Manager が実行されていることを確認します。

表 1-4. ロード バランサ エラー

制御プレーン仮想マシン <仮想マシン名> は、ロード バランサの (<ロード バランサ>- <ロード バランサ エンドポイント>) 証明書を信頼していません。	ロード バランサが提示する証明書は、制御プレーン仮想マシンに構成されている証明書とは異なります。	ロード バランサに正しい管理 TLS 証明書が構成されていることを確認します。
制御プレーン仮想マシン <仮想マシン名> がロード バランサの (<ロード バランサ>- <ロード バランサ エンドポイント>) 証明書を検証できませんでした。証明書が無効です。	ロード バランサが提示する証明書の形式が無効であるか、有効期限が切れています。	構成されたロード バランサのサーバ証明書を修正します。
制御プレーン仮想マシン <仮想マシン名> が、ユーザー名 <ユーザー名> と指定のパスワードを使用してロード バランサ (<ロード バランサ>- <ロード バランサ エンドポイント>) に対して認証できませんでした。	ロード バランサのユーザー名またはパスワードが正しくありません。	ロード バランサに構成されたユーザー名とパスワードが正しいかどうかを確認します。
制御プレーン仮想マシン <仮想マシン名> からロード バランサ (<ロード バランサ>- <ロード バランサ エンドポイント>) に接続する際に HTTP エラーが発生しました。	制御プレーン仮想マシンはロード バランサ エンドポイントに接続できますが、エンドポイントが正常な (200) HTTP 応答を返しません	ロード バランサが健全な状態であり、要求を受け入れていることを確認します。
制御プレーン仮想マシン <仮想マシン名> から <ロード バランサ> (<ロード バランサ エンドポイント>) に接続できません。エラー: <エラー テキスト>	<ul style="list-style-type: none"> ■ 一般的なネットワーク障害が発生しました。 ■ 通常は、ロード バランサが機能していないか、ファイアウォールの一部によって接続がブロックされています。 	<ul style="list-style-type: none"> ■ ロード バランサ エンドポイントにアクセスできることを検証します ■ ファイアウォールがロード バランサへの接続をブロックしていないことを検証します。

スーパーバイザー の自動アップグレード

vCenter Server Appliance をアップグレードすると、スーパーバイザー のアップグレードが自動的にトリガされます。

vSphere IaaS control plane コンポーネントには、vCenter Server 内のコンポーネント、Kubernetes コンポーネント、および ESXi コンポーネントが含まれています。vCenter Server をアップグレードすると、vCenter Server の vSphere IaaS control plane コンポーネントのみがアップグレードされます。Kubernetes コンポーネントと ESXi コンポーネントは手動でアップグレードする必要があります。

自動アップグレード機能を使用すると、vCenter Server をアップグレードするときに、スーパーバイザー のアップグレードがトリガされます。ただし、vSphere 8 U3 の時点では、自動アップグレードでは TKG サービス はアップグレードされません。TKG サービス の使用を参照してください。

注： 自動アップグレードは構成できません。アップグレードは、スーパーバイザー と vCenter Server の Kubernetes バージョン間の互換性に応じて自動的にトリガされます。

vCenter Server をアップグレードしようとする、スーパーバイザー と vCenter Server の Kubernetes バージョン間の互換性を確認するための事前チェックが実行されます。次のシナリオでは、警告が表示されます。

- ターゲット vCenter Server の Kubernetes コンポーネントは、スーパーバイザー の Kubernetes バージョンよりも 1 つ高いバージョンです。このシナリオでは、vCenter Server のアップグレードを続行すると、スーパーバイザー の Kubernetes バージョンが、vCenter Server の Kubernetes バージョンと一致するように自動的にアップグレードされます。

たとえば、アクティブなスーパーバイザー バージョンの Kubernetes バージョンは 1.24 で、ターゲット vCenter Server でサポートされている Kubernetes バージョンは 1.25、1.26、および 1.27 です。この場合、vCenter Server アップグレードすると、スーパーバイザー も 1.25 にアップグレードされます。

- ターゲット vCenter Server の Kubernetes コンポーネントは、スーパーバイザー の Kubernetes バージョンよりも 2 つ以上高いバージョンです。このシナリオでは、vCenter Server をアップグレードできません。vCenter Server をアップグレードできるようにするには、まず、スーパーバイザー の Kubernetes バージョンを、vCenter Server の Kubernetes バージョンの少なくとも 1 つ低いバージョンに手動でアップグレードする必要があります。

たとえば、アクティブなスーパーバイザー の Kubernetes バージョンが 1.24 以下で、ターゲット vCenter Server でサポートされている Kubernetes バージョンは 1.26、1.27、および 1.28 です。vCenter Server をアップグレードするには、最初にスーパーバイザー を 1.25 以降、理想的には 1.26 にアップグレードする必要があります。

- スーパーバイザー のライセンスの期限が切れている。スーパーバイザー に評価版ライセンス以外のライセンスがある場合、ライセンスのステータス（期限切れまたはアクティブ）に関係なく、スーパーバイザー をアップグレードできます。すべての操作はそのまま実行できます。スーパーバイザー に評価版ライセンスがあり、有効期限が切れている場合、スーパーバイザー をアップグレードまたは更新することはできません。

vCenter Server リリースのリストについては、ナレッジベースの記事 <https://kb.vmware.com/s/article/2143838> を参照してください。

kubectl 向けの vSphere プラグインの更新

vSphere 名前空間のアップデートを実行し、スーパーバイザー をアップグレードしたら、kubectl 向けの vSphere プラグイン をアップデートします。

最新バージョンの kubectl 向けの vSphere プラグイン は、Tanzu Kubernetes クラスターのルート CA 証明書を `TANZU-KUBERNETES-NAME-ca` という名前の Kubernetes シークレットにダウンロードしてインストールします。プラグインは、この証明書を使用して、対応するクラスターの認証局データストアの CA 情報を入力します。

kubectl 向けの vSphere プラグイン をダウンロードしてインストールするには、[Kubernetes CLI Tools for vSphere のダウンロードとインストール](#)を参照してください。 `TANZU-KUBERNETES-CLUSTER-NAME-ca` シークレットの詳細については、[Kubectl を使用した TKG 2 クラスター シークレットの取得](#)を参照してください。

TKG サービスのインストールとアップグレード

2

このセクションでは、TKG サービスのインストールとアップグレードについて説明します。

次のトピックを参照してください。

- [TKG サービス の使用](#)
- [TKG サービス のステータスの確認](#)
- [新しいバージョンの TKG サービスの登録](#)
- [TKG サービス バージョンのアップグレード](#)
- [TKG サービスのトラブルシューティング](#)

TKG サービス の使用

VMware Tanzu Kubernetes Grid サービス (TKG サービス) を使用すると、vSphere IaaS control plane に Kubernetes ワークロード クラスタをデプロイできます。TKG サービス が提供するリリースは独立しており、ワークロードを中断することなく非同期アップグレードを行うことができます。

TKG サービス の概要

vSphere 8.0 Update 3 以降では、Tanzu Kubernetes Grid は スーパーバイザー サービス としてインストールされます。このアーキテクチャの変更により、TKG は vSphere IaaS control plane リリースから分離され、vCenter Server や スーパーバイザー に関係なく TKG サービス をアップグレードできます。

TKG サービス 3.0 がインストールされ、スーパーバイザー 制御プレーン ノードで実行されます。TKG サービス は、Carvel パッケージをネストしたコレクションとして提供されます。TKG サービス は、インターネットが制限されている環境でもコア スーパーバイザー サービス としてアップグレードできますが、アンインストールやダウングレードを実行することはできません。[ワークロード管理] - [サービス] タブで TKG サービス を監視および管理できます。TKG サービス バージョンのアップグレードを参照してください。

TKG サービス バージョン 3.1 は、ユーザーのアップグレード先となる最初の独立したリリースです。TKG サービス vCenter Server への新しい TKG サービス バージョンの登録と TKG サービス バージョンのアップグレードは個別のプロセスです。

TKG サービス 3.0 のインストール

vSphere IaaS control plane コンポーネントを必要なバージョンにアップグレードすると、TKG サービス のインストールが自動的に実行されます。詳細については、TKG サービス の [リリース ノート](#) を参照してください。

vCenter Server への新しい TKG サービス バージョンの登録

TKG サービス パッケージは vCenter Server によって公開され、VMware パブリック レジストリにプッシュされます。TKG サービス の登録は、vCenter Server レベルで実行されます。TKG サービス の新しいバージョンを登録するには、同期と非同期の 2 つのオプションがあります。

表 2-1. TKG サービス バージョンの登録オプション

登録方法	説明
同期	最新の vCenter Server リリースへの更新により、TKG サービス の新しいバージョンが自動登録されるまで待機してから、スーパーバイザー を更新し、組み込みレジストリに新しいバージョンをポピュレートします。
非同期	パブリック レジストリから新しい TKG サービス バージョンの定義をダウンロードし、vCenter Server に手動で登録します。

同期登録を行うは、システム更新が必要です。vCenter Server を更新すると、新しい TKG サービス バージョンがスーパーバイザー に自動的に登録されます。ただし、自動登録された（新しい）バージョンを使用するには、スーパーバイザー を、この vCenter Server によって提供される vSphere 名前空間 リリースに付属するバージョンに更新する必要があります。スーパーバイザー を更新すると、スーパーバイザー 組み込みレジストリで TKG サービス の Carvel パッケージ バンドルが使用可能になり、展開する準備が整います。スーパーバイザー をアップグレードしても、TKG サービス は自動的にアップグレードされません。必要なバージョンを選択してデプロイする必要があります。

非同期登録では、現在のスーパーバイザー バージョンがサポート ウィンドウ内にある場合、vCenter Server およびスーパーバイザー の更新は必要ありません。非同期登録のワークフローは、次のとおりです。

- 1 スーパーバイザー サービス のパブリック レジストリ サイトからサービス定義 YAML ファイルをダウンロードします。
- 2 サービス定義を vCenter Server にアップロードして、TKG サービス の新しいバージョンを登録します。

次の表に、TKG サービス の登録の詳細を示します。

表 2-2. TKG サービス バージョンの登録

TKG サービスのプロパティ	バンドルされている vCenter Server	パブリック レジストリ
新しいバージョンの登録	自動登録	手動登録
新しく登録されたバージョンの削除	許可されていません	許可
イメージの場所	スーパーバイザー制御プレーンの組み込みレジストリ	パブリック レジストリ

TKG サービス バージョンのアップグレード

TKG サービス バージョンのアップグレードは、スーパーバイザー レベルで実行されます。TKG サービス が登録されたら、TKG サービス をターゲット スーパーバイザー にスーパーバイザー サービス としてデプロイして、アップグレードします。TKG サービス バージョンのアップグレードを参照してください。

インターネットが制限されている（エアギャップ）環境で TKG サービス をアップグレードするには、vCenter Server をアップデートして新しい TKG サービス バージョンを同期的に登録します。インストールするバージョンを選択すると、ローカル レジストリを使用して新しい TKG サービス バージョンがインストールされます。[vCenter Server への新しい TKG サービス バージョンの登録](#)を参照してください。

TKG サービス バージョンをアップグレードすると、システムは事前チェックを実行し、次の 2 段階の重要度を報告します。

- 警告（非ブロック）
- エラー（ブロック）

Kubernetes バージョン チェックは、ブロックを伴わない警告チェックの一例です。スーパーバイザー バージョン チェックは、ブロックを伴うエラーの一例です。詳細については、スーパーバイザー サービスのドキュメントを参照してください。

TKG サービス のステータスの確認

TKG サービス のステータスを確認するには、このトピックを参照してください。

このタスクを完了して、TKG サービス がコア スーパーバイザー サービス としてインストールされていることを確認し、そのステータスを確認します。

ステータスを確認する手段として、vSphere Client を使用する方法と kubectl を使用する方法の 2 つがあります。vSphere Client を使用してステータスを確認するには、vCenter Server にログインし、[ワークロード管理] - [サービス] の順に選択します。

kubectl を使用してステータスを確認するには、次の手順を実行します。

前提条件

このタスクでは、すべてのシステム コンポーネントをアップグレード済みであること、および TKG サービス 3.0 をインストール済みであることを前提にしています。[「#unique_12/unique_12_Connect_42_TABLE_CED10728-0714-4D00-BF58-D4BBAA2A4D8E」](#)を参照してください。

手順

- 1 kubectl を使用して スーパーバイザー にログインします。

```
kubectl vsphere login --server=<SUPERVISOR-IP-or-FQDN> --vsphere-username <VCENTER-SSO-USER>
```

- 2 次のコマンドを実行します。

```
kubectl get packageinstall --namespace vmware-system-supervisor-services
```

TKG サービス がインストールされていることが表示されます。

NAMESPACE	NAME	PACKAGE NAME
PACKAGE VERSION	DESCRIPTION	AGE
vmware-system-supervisor-services	svc-	
tkg.vsphere.vmware.com	tkg.vsphere.vmware.com	0.0.1-b836be7 Reconcile succeeded 17h

新しいバージョンの TKG サービスの登録

TKG サービスを非同期にアップグレードする目的で、新しいバージョンの TKG サービスを vCenter Server に手動で登録するには、このトピックを参照してください。

このタスクは、TKG サービス バージョンを非同期にアップグレードする場合にのみ必要です。vCenter Server への新しい TKG サービス バージョンの登録を参照してください。

前提条件

このタスクでは、すべてのシステム コンポーネントをアップグレード済みであること、および TKG サービス 3.0 をインストール済みであることを前提にしています。[#unique_12/unique_12_Connect_42_TABLE_CED10728-0714-4D00-BF58-D4BBAA2A4D8E]を参照してください。

手順

- 1 ブラウザを使用して、スーパーバイザー サービスの配布サイト (<https://www.vmware.com/go/supervisor-service>) に移動します。
- 2 サイトから TKG サービスの `package.yaml` ファイルをダウンロードします。
- 3 vSphere Client を使用して、vCenter Server にログインします。
- 4 [ワークロード管理] - [サービス] の順に移動します。
- 5 [Tanzu Kubernetes Grid Service] という名前のサービス タイルを見つけます。
- 6 [アクション] - [新規バージョンの追加] の順に選択します。
- 7 [アップロード] をクリックします。
- 8 ダウンロードした `package.yaml` ファイルを選択します。
- 9 [終了] をクリックします。

結果

新しいサービス定義を登録すると、TKG サービス タイルで使用可能な TKG サービスのバージョンが複数表示されることがあります。TKG サービスをアップグレードするときに、ターゲットバージョンを選択します。

次のステップ

[TKG サービス バージョンのアップグレード](#)。

TKG サービス バージョンのアップグレード

TKG サービス のバージョンをアップグレードするには、このトピックを参照してください。

次の手順を完了して、TKG サービス バージョンをアップグレードします。

TKG サービス バージョンのアップグレードは、スーパーバイザー レベルで実行されます。vCenter Server に複数のスーパーバイザー がホストされている場合は、ターゲットスーパーバイザー を選択する必要があります。

前提条件

このタスクでは、すべてのシステム コンポーネントをアップグレード済みであること、および TKG サービス 3.0 をインストール済みであることを前提としています。[#unique_12/unique_12_Connect_42_TABLE_CED10728-0714-4D00-BF58-D4BBAA2A4D8E]を参照してください。

このタスクでは、TKG サービス の新しいバージョンを新しいバージョンの TKG サービスの登録で、または vCenter Server への新しい TKG サービス バージョンの登録登録していることが前提となります。

手順

- 1 vSphere Client を使用して、vCenter Server にログインします。
- 2 [ワークロード管理] - [サービス] の順に移動します。
- 3 [Tanzu Kubernetes Grid Service] タイルを特定します。
- 4 [アクション] - [スーパーバイザーへのインストール] の順に選択します。
- 5 アップグレード先となる TKG サービス のターゲット バージョンを選択します。
- 6 アップグレードする TKG サービス をホストしているターゲット スーパーバイザー を選択します。
- 7 サービスの互換性を選択し、[OK] をクリックします。
- 8 TKG サービス がアップグレードされていることを確認します。

[ワークロード管理] - [サービス] 画面の [Tanzu Kubernetes Grid Service] タイルには、バージョンとステータスが反映されます。kubect1 get tkr を使用してステータスを確認することもできます。

TKG サービスのトラブルシューティング

TKG サービスをトラブルシューティングするには、このトピックを参照してください。

TKG サービス サポート バンドル

TKG サービス サポート バンドルは、スーパーバイザー サポート バンドルに含まれています。ガイダンスについては、「スーパーバイザーのサポート バンドルの収集」を参照してください。

スーパーバイザー サポート バンドル内では、TKG サービス ログは var/logs/tkg-svs フォルダにあります。

スーパーバイザー サービスはコア サービス コントローラによって管理されます。コア サービス コントローラのログは /var/log/vmware/wcp/ にあります。

アプリケーション プラットフォームのログは、次のコマンドを使用して取得できます。

```
kubect1 logs vmware-system-appplatform-lifecycle-xxx -n vmware-system-appplatform-operator-system
```

TKG サービス クラスターの更新

3

このセクションでは、TKG サービス クラスターの更新について説明します。

次のトピックを参照してください。

- TKG サービス クラスターのローリング アップデート モデルについて
- 更新のための TKGS クラスター互換性の確認
- TKR バージョンの編集による TKG クラスターの更新
- ストレージ クラスの編集による TKG クラスターの更新
- 仮想マシン クラスの編集による TKG サービス クラスターの更新
- Tanzu CLI を使用した TKG クラスターの更新

TKG サービス クラスターのローリング アップデート モデルについて

TKG サービス クラスターでは、ローリング アップデート モデルがサポートされます。クラスター仕様を変更すると、ローリング アップデートを開始できます。一部のシステム操作でローリング アップデートが開始される場合があります。環境をアップデートする前に、ローリング アップデート プロセスについて理解しておく必要があります。

TKG サービス 3.0 以降の TKGS クラスターのローリング アップデート モデル

TKG サービス 3.0 以降では、TKG コントローラが vCenter Server とスーパーバイザーから独立しています。[TKG サービスの使用](#)を参照してください。これらのコンポーネントをアップグレードしても、TKGS クラスターのローリング アップデートは開始されません。

TKG サービス バージョンをアップグレードすると、TKGS クラスターのローリング アップデートがトリガされる場合があります。

TKG サービス 3.0 より前の TKGS クラスターのローリング アップデート モデル

TKG コントローラは、スーパーバイザー で実行されます。スーパーバイザー をアップデートすると、アップデートが可能な場合は TKG コントローラが自動的にアップデートされます。各 TKG コントローラのアップデートには、CNI、CSI、CPI などのサポート サービスのアップデートと、クラスターの構成のアップデートが含まれる可能性があります。互換性を維持するために、システムは事前チェックを実行し、コンプライアンスを適用します。

vSphere IaaS control plane では、ローリング アップデート モデルを使用して、スーパーバイザー 上の TKG クラスタをアップデートします。ローリング アップデート モデルを使用すると、クラスタをアップデートしている間のダウンタイムを最小限に抑えられます。ローリング アップデートには、Kubernetes ソフトウェアのバージョンのアップグレードに加えて、仮想マシンの構成とリソース、サービスと名前空間、カスタム リソースなどの、クラスタをサポートするインフラストラクチャおよびサービスの更新が含まれます。アップデートが正常に実行されるためには、構成がいくつかの互換性要件を満たしている必要があります。そのため、システムは再チェック条件を適用して、クラスタの更新の準備ができていることを確認し、クラスタのアップグレードが失敗した場合のロールバックをサポートします。

クラスタ マニフェストの特定の要素を変更することで、TKG クラスタのローリング アップデートを開始できます。ローリング クラスタ アップデートは、システムによって開始することもできます。たとえば、vSphere 名前空間の更新を実行すると、システムは更新された構成を直ちにすべてのワークロード クラスタに伝達します。これらの更新によってクラスタ ノードのローリング アップデートをトリガすることができます。また、いずれかの構成要素に対する変更によってローリング アップデートを開始することもできます。たとえば、ディストリビューションバージョンに対応する `VirtualMachineImage` を名前変更または置換すると、システムが新しいイメージで実行されているすべてのノードの取得を試行するため、ローリング アップデートが開始されます。スーパーバイザー を更新した場合も、そこにデプロイされているワークロード クラスタのローリング アップデートがトリガされることがあります。たとえば、`vmware-system-tkg-controller-manager` が更新された場合、システムは新しい値をマニフェスト ジェネレータに導入し、コントローラはこれらの値をデプロイするローリング アップデートを開始します。

クラスタ ノードを置き換えるためのローリング アップデート プロセスは、Kubernetes 環境でのポッドのローリング アップデートと同様です。ワークロード クラスタのローリング アップデートを実行するコントローラは 2 つあります。アドオン コントローラとクラスタ コントローラです。これらの 2 つのコントローラでは、ローリング アップデートに、アドオンの更新、制御プレーンの更新、およびワーカー ノードの更新の 3 つの主要なステージがあります。これらのステージは順番に実行されますが、前の手順が十分に進行するまで次のステップの開始を防ぐ事前チェックが実行されます。不要と判断された場合、ステップはスキップされることがあります。たとえば、更新がワーカー ノードのみに影響する場合、アドオンや制御プレーンの更新は必要なくなります。

更新プロセスでは、システムは新しいクラスタ ノードを追加し、ノードがターゲットの Kubernetes バージョンでオンラインになるまで待機します。その後、古いノードを削除対象としてマークし、次のノードに移動して、プロセスを繰り返します。すべてのポッドが削除されるまで、古いノードは削除されません。たとえば、ノードの完全なドレーンを妨げる `PodDisruptionBudgets` でポッドが定義されている場合、ノードは遮断されますが、それらのポッドが消去できるようになるまで削除されません。システムは、最初にすべての制御プレーン ノード、次にワーカー ノードをアップグレードします。更新中は、クラスタのステータスが「更新中」に変わります。ローリング アップデート プロセスが完了すると、クラスタのステータスが「実行中」に変わります。

レプリケーション コントローラによって管理されていないクラスタで実行されているポッドは、クラスタの更新中にワーカー ノードのドレーンの一環として、Kubernetes バージョンのアップグレードで削除されます。クラスタの更新が手動でトリガされた場合や、vSphere 名前空間またはスーパーバイザー の更新によって自動実行された場合がこれに該当します。レプリケーション コントローラによって管理されていないポッドには、`Deployment` または `ReplicaSet` 仕様の一部として作成されていないポッドなどがあります。詳細については、Kubernetes ドキュメントの [Pod Lifecycle: Pod lifetime](#) を参照してください。

ユーザーが開始するローリング アップデート

スーパーバイザー で TKG クラスタのローリング アップデートを開始するには、Tanzu Kubernetes リリース バージョン、仮想マシン クラス、ストレージ クラスをそれぞれアップデートします。詳細については、次のいずれかのトピックを参照してください。

- [TKR バージョンの編集による TKG クラスタの更新](#)
- [ストレージ クラスの編集による TKG クラスタの更新](#)
- [仮想マシン クラスの編集による TKG サービス クラスタの更新](#)
- [Tanzu CLI を使用した TKG クラスタの更新](#)

システムが開始するローリング アップデート

スーパーバイザー のリリースごとに、次の1つ以上のオブジェクトを変更できます。

- kubeadmcontrolplanetemplate/kubeadmcontrolplane
- kubeadmconfigtemplate/kubeadmconfig
- vspheremachinetemplate/vspheremachine (vSphere 8.x の場合)
- wcpmachinetemplate/wcpmachine (vSphere 7.x の場合)

スーパーバイザー がアップグレードされると、コア クラスタ API (CAPI) コントローラは TKG ワークロード クラスタへのアップデート ロールアウトをトリガし、上記のオブジェクトの目的の状態を、実行中のワークロード クラスタと一致させます。

vSphere IaaS control plane では、スーパーバイザー で実行されている TKG コントローラによってこれらのオブジェクトが生成され、システム コードとの同期が維持されます。これにより、コントローラが新しいコードに更新された場合、上記のオブジェクトのいずれかを変更すると、既存の TKG クラスタがローリング アップデートされます。つまり、スーパーバイザー に影響する変更をシステム コードに加えると、TKG クラスタがローリング アップデートされます。

次の表に、スーパーバイザー がアップグレードされたときにワークロード クラスタのローリング アップデートが自動実行される条件を示します。

アップグレードのシナリオ	説明
任意の vCenter Server 7.x リリースから任意の vCenter Server リリースへのアップグレード	<p>すべての Tanzu Kubernetes クラスタのローリング アップデートがトリガされる場合があります。</p> <p>ローリング アップデートは、vCenter Server のアップグレードに続くスーパーバイザーの最初のアップグレードによってトリガされます。通常、ローリング アップデートは、同じ vCenter Server でのスーパーバイザーのアップグレードによってトリガされることはありません。</p> <p>詳細については、リリース ノートを参照してください。</p>
任意の vCenter Server リリースから任意の vCenter Server 8.x リリースへのアップグレード	<p>次のコード変更を反映させる必要があるため、すべての TKG クラスタのローリング アップデートがトリガされます。</p> <ul style="list-style-type: none"> ■ 基盤となる CAPI プロバイダを CAPW から CAPV に移動する ■ クラスタをクラスレス CAPI クラスタから上位の CAPI クラスタに移行する

アップグレードのシナリオ	説明
vCenter Server 8.0 GA リリース (8.0.0) から vCenter Server 8.0.0b または 8.0.0c リリースへのアップグレード	次のいずれかのケースが当てはまる場合は、指定した TKG クラスタのローリング アップデートがトリガされます。 <ul style="list-style-type: none"> ■ 空でない noProxy リストを持つプロキシ設定が使用されていた TKG クラスタがある場合は、この TKG クラスタ。 ■ スーパーバイザーで組み込みの Harbor レジストリ サービスが有効になっていた場合は、すべての TKG クラスタ。
vSphere 8.0.0b リリースから vSphere 8.0.0c リリースへのアップグレード	ワークロード クラスタの自動ロールアウトなし
vSphere 8.0.0c リリースから vSphere 8.0 Update 1 リリース (8.0.1) へのアップグレード	ワークロード クラスタの自動ロールアウトなし
任意の vSphere 8.x バージョンから 8.0 U2 リリース (8.0.2) へのアップグレード	以下の変更を行う必要があるため、すべての TKC に対するローリング アップグレードが実行されます。 <ul style="list-style-type: none"> ■ vSphere 8.0 U2 には、ClusterClass の一部として、GCM 内の TKG 1.0 と TKG 2.0 の両方の TKR に対する Kubernetes レベルの STIG の変更が含まれています。 ■ 1.23 以降の TKC には 8.0 U2 に対する互換性があるため、すべてのクラスタでローリング アップグレードが実行されます。
8.0 U2 (8.0.2) 未満の任意の vSphere 8.x バージョンから 8.0 U2c リリースへのアップグレード	以下の変更を行う必要があるため、すべての TKC に対するローリング アップグレードが実行されます。 <ul style="list-style-type: none"> ■ 8.0 U2 には、ClusterClass の一部として、GCM 内の TKG 1.0 と TKG 2.0 の両方の TKR に対する k8s レベルの STIG の変更が含まれています。 ■ 1.23 以降の TKC には 8.0 P03 に対する互換性があるため、すべてのクラスタでローリング アップグレードが実行されます。

また、TKR イメージをホストしているコンテンツ ライブラリを変更したときに、TKG クラスタのローリング アップデートがトリガされることがあります。サブスクリプションを介して、または手動で新しいイメージを追加しても、TKG クラスタのローリング アップデートはトリガされません。ただし、コンテンツ ライブラリを変更して、異なる名前のイメージを追加すると、すべての TKG クラスタのローリング アップデートがトリガされます。

たとえば、システム定義の OVA 名を自動的に使用するサブスクライブ済みコンテンツ ライブラリを使用しているシナリオについて考えます。ここでは、ローカル コンテンツ ライブラリに切り替え、同じ OVA をポピュレートしてそれぞれ異なる名前を指定します。これにより、すべての TKG クラスタのローリング アップデートがトリガされます。置換後のコンテンツ ライブラリには同じ OVA がありますが、ユーザー定義の異なる名前を持つためです。

複数のノード プールを持つクラスタのローリング アップデートに関する考慮事項

複数のノード プールを持つ TKG クラスタを使用している場合は、ローリング アップデートに関する次の情報を考慮してください。

ワーカー ノード プール

ワーカー ノード プールは、vSphere 7 U3 でリリースされた TKGS v1alpha2 API で導入されました。クラスタ API MachineDeployments は、ワーカー ノード プールの基盤となる Kubernetes プリミティブです。

ClusterClass は、vSphere 8 リリースの TKGS で導入されました。v1alpha3 API と v1beta1 API の両方が ClusterClass に基づいています (v1alpha3 は ClusterClass 上の抽象化レイヤーです)。

ローリング アップデート中に複数のノード プールを更新する方法

複数のノード プールを使用してプロビジョニングされた TKGS ワークロード クラスタを更新する場合は、使用されている vSphere のバージョンによってローリング アップデート モデルが異なります。

vSphere	TKGS API	アップグレードの動作
vSphere 7 TKGS	v1alpha2 API	同じクラスタ内の複数のノード プールが同時に更新されます
vSphere 8 TKGS	v1alpha3 API および v1beta1 API	同じクラスタ内の複数のノード プールが論理的な順序に従って (順番に) 更新されます

ベスト プラクティスに関する考慮事項

同一の複数のノード プールを使用して vSphere 8 TKGS クラスタをプロビジョニングしても、サイジングの観点からは意味がありません。ノード プールは、サイズ、仮想マシン クラス、TKr バージョンなどが異なる場合に使用する必要があります。複数の同一のノード プールを使用してクラスタを迅速にアップグレードしても効果がないため、この処理は行わないでください。

Pod Disruption Budget は、アップグレードが実行中のアプリケーションに干渉しないようにするための適切な方法です。これに対処する最善の方法は、ワークロードに PodDisruptionBudgets を設定することです (<https://kubernetes.io/docs/tasks/run-application/configure-pdb/> を参照)。クラスタ API はこれらの設定を考慮し、しきい値を超えるとマシンを終了しません。

vSphere 8 TKGS クラスタのローリング アップデートの詳細

vSphere 8 TKGS クラスタ バージョンの更新中：

- 制御プレーン ノードが最初に更新され、続いて Zone-A ノード プールから一度に 1 つのワーカー ノードがロールアウトされます。2 つのノード プールが使用されている場合、一度にロールアウトされるワーカーは 1 つのみです。

クラスタ構成変数の更新中：

- 制御プレーン ノードが最初に更新され、続いてノード プールごとに 1 つのワーカー ノードがロールアウトされます。たとえば、2 つのノード プールが使用されている場合は、一度に 2 つのワーカーがロールアウトされます。

更新のための TKGS クラスタ 互換性の確認

TKGS ワークロード クラスタをアップグレードする前に、クラスタがアップグレードに対する互換性を備えているか確認する必要があります。TKG サービス に対する互換性を再確認する必要があります。

TKG サービス による互換性の確認

ワークロード クラスタをアップグレードする前に、クラスタがアップグレードに対する互換性を備えているか確認する必要があります。クラスタに TKG サービス との互換性がない場合は、Tanzu Kubernetes リリース をアップグレードします。使用可能な TKr の詳細については、[リリース ノード](#)を参照してください。[オンライン相互運用性マトリックス](#)も参照してください。

次のコマンドを使用して、Tanzu Kubernetes リリース の一覧および互換性を表示できます。

```
kubectl get tkr
```

COMPATIBLE 列は、その Tanzu Kubernetes リリース がインストール済みの TKG サービス と互換性があるかどうかを示します。TKG サービス 3.1 リリース以降では、TYPE 列も互換性ステータスを返します。

TKGS クラスタを指定した場合は、使用可能な TKr のアップデートが表示されます。

v1alpha3 API を使用する場合：

```
kubectl get tkc <tkgs-cluster-name>
```

または v1beta1 API を使用する場合：

```
kubectl get cc <tkgs-cluster-name>
```

UPDATES AVAILABLE 列は、Kubernetes の利用可能なアップグレードがあるかどうかと、使用する次の推奨 Tanzu Kubernetes リリース を示します。例：

```
kubectl get tkc tkg2-cluster-11-tkc
NAME                CONTROL PLANE  WORKER  TKR NAME                AGE
READY  TKR COMPATIBLE  UPDATES AVAILABLE
tkg2-cluster-11-tkc  3              3       v1.25.7---vmware.3-fips.1-tkg.1  13d
True    True             [v1.26.5+vmware.2-fips.1-tkg.1]
```

TKr 形式には、非レガシーとレガシーの 2 種類があります。

- 非レガシー TKr は vSphere 8.x 専用であり、vSphere 8.x とのみ互換性があります
- レガシー TKr は vSphere 7.x および vSphere 8.x と互換性があるレガシー形式を使用していますが、アップグレードのみを目的としています。

非レガシー TKr を一覧表示する場合：

```
kubectl get -l !run.tanzu.vmware.com/legacy-tkr
```

レガシー TKr を一覧表示する場合：

```
kubectl get -l !run.tanzu.vmware.com/legacy-tkr
```

TKR バージョンの編集による TKG クラスタの更新

このタスクでは、TKG クラスタ マニフェストを編集して TKG クラスタの Tanzu Kubernetes リリース バージョンを更新する方法について説明します。

kubectl edit コマンドを使用して Tanzu Kubernetes リリース バージョンをアップグレードすることで、TKGS クラスタのローリング アップデートを開始できます。

注： kubectl apply コマンドを使用して、デプロイされたクラスタの TKR バージョンを更新することはできません。

前提条件

このタスクでは、kubectl edit コマンドを使用する必要があります。このコマンドを実行すると、KUBE_EDITOR または EDITOR 環境変数によって定義されたテキスト エディタで、クラスタ マニフェストが開かれます。ファイルを保存すると、変更が反映されてクラスタが更新されます。kubectl edit コマンドを実行できるように kubectl のエディタを構成するには、「[#unique_23](#)」を参照してください。

手順

- 1 スーパーバイザー で認証します。

```
kubectl vsphere login --server=IP-ADDRESS --vsphere-username USERNAME
```

- 2 ターゲット ワークロード クラスタがプロビジョニングされている vSphere 名前空間 にコンテキストを切り替えます。

```
kubectl config use-context SUPERVISOR-NAMESPACE
```

- 3 ターゲット TKG クラスタおよびバージョンを取得します。

v1alpha3 クラスタ :

```
kubectl get tanzukubernetescluster
```

v1beta1 クラスタ :

```
kubectl get cluster
```

- 4 使用できる Tanzu Kubernetes リリースをリストします。

```
kubectl get tanzukubernetesreleases
```

- 5 次のコマンドを実行して、クラスタのマニフェストを編集します。

v1alpha3 クラスタ :

```
kubectl edit tanzukubernetescluster/CLUSTER-NAME
```

v1beta1 クラスタ :

```
kubectl edit cluster/CLUSTER-NAME
```

6 Tanzu Kubernetes リリース の文字列を更新してマニフェストを編集します。

たとえば、v1alpha3 クラスタの場合、TKR v1.25.7 を次のように変更します。

```
topology:
  controlPlane:
    replicas: 1
    storageClass: vsan-default-storage-policy
    tkr:
      reference:
        name: v1.25.7---vmware.3-fips.1-tkg.1
      vmClass: guaranteed-large
  nodePools:
  - name: worker-tkg-pool01
    replicas: 3
    storageClass: vsan-default-storage-policy
    tkr:
      reference:
        name: v1.25.7---vmware.3-fips.1-tkg.1
      vmClass: guaranteed-large
    volumes:
    - capacity:
        storage: 128Gi
      mountPath: /var/lib/containerd
      name: containerd
```

TKR v1.26.5 に変更：

```
topology:
  controlPlane:
    replicas: 1
    storageClass: vsan-default-storage-policy
    tkr:
      reference:
        name: v1.26.5---vmware.2-fips.1-tkg.1
      vmClass: guaranteed-large
  nodePools:
  - name: worker-tkg-pool01
    replicas: 3
    storageClass: vsan-default-storage-policy
    tkr:
      reference:
        name: v1.26.5---vmware.2-fips.1-tkg.1
      vmClass: guaranteed-large
    volumes:
    - capacity:
        storage: 128Gi
      mountPath: /var/lib/containerd
      name: containerd
```

注： 制御プレーン ノードとワーカー ノードの TKR バージョンが同じである必要があります。すべての TKR インスタンスを更新するか、制御プレーンのバージョンを更新して、ワーカー ノードから TKR 名を削除できません。

たとえば、v1beta1 クラスタの場合、TKR v1.25.7 を次のように変更します。

```
apiVersion: cluster.x-k8s.io/v1beta1
...
topology:
  class: tanzukubernetescluster
  version: v1.25.7---vmware.3-fips.1-tkg.1
  controlPlane:
    replicas: 3
  workers:
    ...
  variables:
    ...
```

TKR v1.26.5 に変更：

```
apiVersion: cluster.x-k8s.io/v1beta1
...
topology:
  class: tanzukubernetescluster
  version: v1.26.5---vmware.2-fips.1-tkg.1
  controlPlane:
    replicas: 3
  workers:
    ...
  variables:
    ...
```

7 マニフェスト ファイルに行った変更内容を保存します。

ファイルを保存すると、`kubectl` によって変更がクラスタに適用されます。バックグラウンドで、スーパーバイザーの仮想マシン サービスによって新しいワーカー ノードがプロビジョニングされます。

8 `kubectl` から、マニフェストの編集が正常に記録されたことが報告されているかを確認します。

```
kubectl edit tanzukubernetescluster/tkg-cluster-1
tanzukubernetescluster.run.tanzu.vmware.com/tkg-cluster-1 edited
```

注： エラーが表示された場合、またはクラスタ マニフェストが正常に編集されたことが `kubectl` から報告されない場合は、`KUBE_EDITOR` 環境変数を使用して、デフォルトのテキスト エディタが適切に構成されていることを確認してください。[#unique_23](#) を参照してください。

9 クラスタが更新されていることを確認します。

```
kubectl get tanzukubernetescluster
NAME                CONTROL PLANE  WORKER  DISTRIBUTION                AGE   PHASE
tkgs-cluster-1     3              3      v1.26.5---vmware.2-fips.1-tkg.1  21h  updating
```

10 クラスタが更新されたことを確認します。

```
kubectl get tanzukubernetescluster
NAME                CONTROL PLANE  WORKER  DISTRIBUTION                AGE  PHASE
tkgs-cluster-1     3              3      v1.26.5---vmware.2-fips.1-tkg.1  22h  running
```

ストレージ クラスの編集による TKG クラスタの更新

クラスタ ノードで使用されるストレージ クラスを変更して、TKG クラスタを更新できます。

`kubectl edit` コマンドを使用してクラスタ仕様の `storageClass` パラメータの値を編集することで、TKG クラスタのローリング アップデートを開始できます。

注： `kubectl apply` コマンドを使用して、デプロイされた TKG クラスタを更新することはできません。

前提条件

このタスクでは、`kubectl edit` コマンドを使用する必要があります。このコマンドを実行すると、`KUBE_EDITOR` または `EDITOR` 環境変数によって定義されたテキスト エディタで、クラスタ マニフェストが開かれます。ファイルを保存すると、変更が反映されてクラスタが更新されます。`kubectl` のエディタを構成するには、[#unique_23](#) を参照してください。

手順

1 スーパーバイザー で認証します。

```
kubectl vsphere login --server=IP-ADDRESS --vsphere-username USERNAME
```

2 ターゲット ワークロード クラスタがプロビジョニングされている vSphere 名前空間 にコンテキストを切り替えます。

```
kubectl config use-context SUPERVISOR-NAMESPACE
```

3 使用可能なストレージ クラスを判別して、使用するクラスを決定するには、次のコマンドを実行します。

```
kubectl describe tanzukubernetescluster CLUSTER-NAME
```

4 次のコマンドを実行して、クラスタのマニフェストを編集します。

v1alpha3 クラスタ :

```
kubectl edit tanzukubernetescluster/CLUSTER-NAME
```

v1beta1 クラスタ :

```
kubectl edit cluster/CLUSTER-NAME
```


- 5 storageClass の値を変更して、マニフェストを編集します。

たとえば、v1alpha3 クラスタの場合、制御プレーンおよびワーカー ノードに silver-storage-class クラスを使用しないようにクラスタのマニフェストを変更します。

```
spec:
  topology:
    controlPlane:
      ...
      storageClass: silver-storage-class
    workers:
      ...
      storageClass: silver-storage-class
```

制御プレーンおよびワーカー ノードに gold-storage-class クラスを使用するには、次の手順を実行します。

```
spec:
  topology:
    controlPlane:
      ...
      storageClass: gold-storage-class
    workers:
      ...
      storageClass: gold-storage-class
```

同様に、v1beta1 クラスタをプロビジョニングした場合、クラスタ仕様の variables.storageclass の値をストレージ クラスの名前で更新します。

- 6 マニフェスト ファイルに行った変更内容を保存します。

ファイルを保存すると、kubectl によって変更がクラスタに適用されます。バックグラウンドで、Tanzu Kubernetes Grid は新しいノード仮想マシンをプロビジョニングし、古い仮想マシンをスピンドアウンします。

- 7 kubectl から、マニフェストの編集が正常に記録されたことが報告されているかを確認します。

```
kubectl edit tanzukubernetescluster/tkgs-cluster-1
tanzukubernetescluster.run.tanzu.vmware.com/tkgs-cluster-1 edited
```

注： エラーが表示された場合、またはクラスタ マニフェストが正常に編集されたことが kubectl から報告されない場合は、KUBE_EDITOR 環境変数を使用して、デフォルトのテキスト エディタが適切に構成されていることを確認してください。[#unique_23](#) を参照してください。

- 8 クラスタが更新されたことを確認します。

v1alpha3 クラスタ :

```
kubectl get tanzukubernetescluster
```

v1beta1 クラスタ :

```
kubectl get cluster
```

仮想マシン クラスの編集による TKG サービス クラスターの更新

クラスター ノードをホストするために使用されている仮想マシンのクラスを変更することで、TKG サービス クラスターを更新できます。

`kubectl edit` コマンド を使用して `vmClass` の定義を編集することで、TKG サービス クラスターのローリング アップデートを開始できます。変更されたクラスに基づく新しいノードがロールアウトされ、古いノードが停止します。

注： `kubectl apply` コマンドを使用して、デプロイされた TKG クラスターを更新することはできません。

前提条件

このタスクでは、`kubectl edit` コマンドを使用する必要があります。このコマンドを実行すると、`KUBE_EDITOR` または `EDITOR` 環境変数によって定義されたテキスト エディタで、クラスター マニフェストが開かれます。ファイルを保存すると、変更が反映されてクラスターが更新されます。`kubectl` のエディタを構成するには、`#unique_23` を参照してください。

手順

- 1 スーパーバイザー で認証します。

```
kubectl vsphere login --server=IP-ADDRESS --vsphere-username USERNAME
```

- 2 ターゲット TKG クラスターがプロビジョニングされている vSphere 名前空間 にコンテキストを切り替えます。

```
kubectl config use-context SUPERVISOR-NAMESPACE
```

- 3 ターゲット TKG クラスターを記述し、仮想マシンのクラスを確認します。

v1alpha3 クラスター :

```
kubectl describe tanzukubernetescluster CLUSTER-NAME
```

v1beta1 クラスター :

```
kubectl describe cluster CLUSTER-NAME
```

- 4 クラスターがプロビジョニングされている vSphere 名前空間 で使用可能な仮想マシン クラスを一覧表示して、記述します。

```
kubectl get virtualmachineclass
```

注： ターゲット仮想マシン クラスは、TKG クラスターがプロビジョニングされている vSphere 名前空間 に関連付けられている必要があります。仮想マシン クラスを vSphere 名前空間 にバインドする方法の詳細については、TKG サービスまたは仮想マシン サービスのドキュメントを参照してください。

- 5 次のコマンドを実行して、クラスターのマニフェストを編集します。

v1alpha3 クラスター :

```
kubectl edit tanzukubernetescluster/CLUSTER-NAME
```

v1beta1 クラスタ :

```
kubectl edit cluster/CLUSTER-NAME
```

6 仮想マシン クラス文字列を変更して、マニフェストを編集します。

たとえば、v1alpha3 クラスタを使用している場合、ワーカー ノードに `guaranteed-medium` 仮想マシン クラスを使用しないようにクラスタのマニフェストを変更します。

```
topology:
  controlPlane:
    replicas: 3
    storageClass: vwk-storage-policy
    tkr:
      reference:
        name: v1.27.11---vmware.1-fips.1-tkg.2
    vmClass: guaranteed-medium
  nodePools:
  - name: worker-nodepool-a1
    replicas: 3
    storageClass: vwk-storage-policy
    tkr:
      reference:
        name: v1.27.11---vmware.1-fips.1-tkg.2
    vmClass: guaranteed-medium
```

ワーカーノードに `guaranteed-large` 仮想マシン クラスを使用するには、次の手順を実行します。

```
topology:
  controlPlane:
    replicas: 3
    storageClass: vwk-storage-policy
    tkr:
      reference:
        name: v1.27.11---vmware.1-fips.1-tkg.2
    vmClass: guaranteed-medium
  nodePools:
  - name: worker-nodepool-a1
    replicas: 3
    storageClass: vwk-storage-policy
    tkr:
      reference:
        name: v1.27.11---vmware.1-fips.1-tkg.2
    vmClass: guaranteed-large
```

同様に、v1beta1 クラスタをプロビジョニングした場合、`variables.vmclass` の値をターゲット仮想マシン クラスに更新します。

7 マニフェスト ファイルに行った変更内容を保存します。

ファイルを保存すると、`kubectl` によって変更がクラスタに適用されます。バックグラウンドで、TKG コントローラが新しいノード仮想マシンをプロビジョニングし、古い仮想マシンをスピンドウンします。

- 8 kubectl から、マニフェストの編集が正常に記録されたことが報告されているかを確認します。

```
kubectl edit tanzukubernetescluster/tkgs-cluster-1
tanzukubernetescluster.run.tanzu.vmware.com/tkgs-cluster-1 edited
```

注： エラーが表示された場合、またはクラスター マニフェストが正常に編集されたことが kubectl から報告されない場合は、KUBE_EDITOR 環境変数を使用して、デフォルトのテキスト エディタが適切に構成されていることを確認してください。#unique_23 を参照してください。

- 9 クラスターが更新されたことを確認します。

v1alpha3 クラスター :

```
kubectl get tanzukubernetescluster
```

v1beta1 クラスター :

```
kubectl get cluster
```

Tanzu CLI を使用した TKG クラスターの更新

TKG クラスターを更新するには、Tanzu CLI を使用して Tanzu Kubernetes リリース バージョンをアップグレードします。

Tanzu CLI を使用して Tanzu Kubernetes リリース バージョンをアップグレードすることで、TKGS クラスターのローリング アップデートを開始できます。

使用方法の詳細については、『Tanzu CLI リファレンス ガイド』を参照してください。

前提条件

#unique_24。

手順

- 1 スーパーバイザー で認証します。
- 2 TKG クラスターを一覧表示します。

```
tanzu cluster list
```

- 3 TKG クラスターを更新します。

```
tanzu cluster upgrade CLUSTER-NAME --tkr TKR-NAME -n VSPHERE-NAMESPACE
```

ここで、

- CLUSTER-NAME は、アップグレードのターゲットとしている TKG クラスターの名前です。
- TKR-NAME は、TKR バージョンの文字列です
- VSPHERE-NAMESPACE は、TKG クラスターがプロビジョニングされている vSphere 名前空間 の名前です

例 :

```
tanzu cluster upgrade tkg-cluster-1 --tkr v1.23.8---vmware.2-tkg.2-zshippable -n tkg2-  
cluster-ns
```

4 クラスタ アップグレードを確認します。

クラスタがアップグレードされると、次のようなメッセージが表示されます。

```
Cluster 'tkg-cluster-1' successfully upgraded to kubernetes version 'v1.23.8+vmware.2-  
tkg.2-zshippable'
```

vSphere Lifecycle Manager の操作

4

vSphere 管理者は、単一の VMware vSphere Lifecycle Manager イメージを使用して管理している vSphere クラスタで vSphere IaaS control plane を有効にできます。その後、スーパーバイザー が vSphere Lifecycle Manager によって管理されている間、このクラスタを使用できます。

vSphere Lifecycle Manager には、環境内の ESXi ホストおよびクラスタを管理する機能があります。スーパーバイザー を最新バージョンの vSphere IaaS control plane にアップグレードできます。スーパーバイザー 内のホストの ESXi バージョンをアップグレードすることもできます。

vSphere Lifecycle Manager は vCenter Server で実行されるサービスです。vCenter Server をデプロイすると、HTML5 ベースの vSphere Client で vSphere Lifecycle Manager ユーザー インターフェイスが有効になります。

vSphere Lifecycle Manager の詳細については、ホストとクラスタのライフサイクルの管理ドキュメントを参照してください。

次のトピックを参照してください。

- 要件
- vSphere Lifecycle Manager によって管理されているクラスタでの vSphere IaaS control plane の有効化
- スーパーバイザー のアップグレード
- スーパーバイザー へのホストの追加
- スーパーバイザー からのホストの削除
- スーパーバイザー クラスタの無効化

要件

vSphere Lifecycle Manager によって管理されている vSphere クラスタで vSphere IaaS control plane を構成するには、環境が特定の要件を満たしている必要があります。

システム要件

vSphere IaaS control plane を有効にするには、vSphere クラスタのコンポーネントが次の要件を満たしていることを確認します。

- NSX を使用する場合は、vCenter Server と ESXi がバージョン 7.0 Update 2 以降であることを確認します。
 - vSphere ネットワークを使用する場合は、vCenter Server と ESXi がバージョン 7.0 Update 1 以降であることを確認します。
 - vSphere クラスタで HA と DRS が有効であることを確認します。
 - vSphere Distributed Switch バージョン 7.0 Update 2 以降が構成されていることを確認します。
 - クラスタで vSphere ネットワークまたは NSX 3.1 以降のバージョンが構成されていることを確認します。それより前のバージョンの NSX で構成されているクラスタは、vSphere Lifecycle Manager イメージを使用して管理することはできません。
-
- **注：** vSphere Update Manager を使用する スーパーバイザー を vSphere vSphere Lifecycle Manager に移行することはできません。vSphere Update Manager から vSphere Lifecycle Manager に移行 vSphere IaaS control plane が有効になっていないクラスタでのみサポートされます。

vSphere Lifecycle Manager によって管理されているクラスタでの vSphere IaaS control plane の有効化

単一の vSphere Lifecycle Manager イメージを使用して管理しているクラスタで vSphere IaaS control plane を有効にすることで、Kubernetes ワークロードを実行できます。このクラスタを有効にすると、vSphere Lifecycle Manager を使用して スーパーバイザー を管理できるようになります。

NSX を使用する vSphere IaaS control plane でクラスタを有効にすると、vSphere Lifecycle Manager によってクラスタ内のすべての ESXi ホストに Spherelet vSphere インストール バンドル (VIB) がインストールされます。クラスタを有効にすると、vCenter Server に付属するバージョンの Kubernetes が割り当てられます。インストールが完了すると、WCP サービスは、Spherelet の起動や構成などのインストール後のタスクを実行します。

クラスタを有効にする手順については、[vSphere ネットワークを使用する 1 ゾーン スーパーバイザーの有効化](#)を参照してください。

スーパーバイザー のアップグレード

vSphere IaaS control plane クラスタを支えている vSphere インフラストラクチャ、Kubernetes のバージョン、単一の vSphere Lifecycle Manager イメージを使用するクラスタ上の vSphere 向け Kubernetes CLI Tools などを含め、vSphere IaaS control plane を最新バージョンに更新できます。

スーパーバイザー 内のホストの ESXi バージョンをアップグレードします。アップグレード中に、すべての ESXi ホスト上の Spherelet VIB がアップグレードされます。

vSphere Lifecycle Manager は DRS を使用して、ホストをメンテナンス モードにしてから修正します。DRS は、修正が正常に行われるようにするために、vCenter Server が実行されている仮想マシンを別のホストに移行するよう試みます (ホストにアフィニティ化された仮想マシンやホストのローカル ストレージ上で実行されている仮想マシンなど)。また、vSphere ポッド を含むワークロードを他のホストに移行します。

注： vSphere Lifecycle Manager を使用すると、単一の vSphere Lifecycle Manager イメージを使用するクラスタでのみ スーパーバイザー をアップグレードできます。

手順

- 1 vSphere Client メニューから、[ワークロード管理] を選択します。
- 2 [更新] タブを選択します。
- 3 更新する [使用可能なバージョン] を選択します。
たとえば、バージョン `v1.17.4-vsc0.0.2-16293900` を選択します。
- 4 アップデートを適用する対象の スーパーバイザー を選択します。
- 5 アップデートを開始するには [更新の適用] をクリックします。
- 6 更新の状態を監視するには、[最近のタスク] ペインを使用します。

スーパーバイザー へのホストの追加

実行するワークロードを増やすには、vSphere 管理者が スーパーバイザー をスケールアウトする必要があります。クラスタに容量を追加するには、単一の vSphere Lifecycle Manager イメージを使用するクラスタに ESXi ホストを追加します。

NSX で構成されている スーパーバイザー にホストを追加すると、vSphere Lifecycle Manager によって Spherelet VIB とイメージがホストにインストールされます。インストール後、vSphere IaaS control plane は、新たに追加されたホスト上で Spherelet プロセスを構成します。これにより、コンテナを ESXi 上でネイティブに実行できます。

前提条件

- ホストの root ユーザー アカウントのユーザー名およびパスワードを取得します。
- ファイアウォールの背後にあるホストが vCenter Server と通信できることを確認します。

手順

- 1 vSphere Client メニューから、[ワークロード管理] を選択します。
- 2 データセンター、クラスタ、またはフォルダを右クリックし、[ホストの追加] を選択します。
- 3 ホストの IP アドレスまたは名前を入力し、[次へ] をクリックします。
- 4 管理者の認証情報を入力し、[次へ] をクリックします。
- 5 ホストの概要を確認し、[次へ] をクリックします。
- 6 ホストにライセンスを割り当て、[終了] をクリックします。

- 7 [ホストの追加] ウィザードで [次へ] をクリックします。
- 8 概要を確認し、[終了] をクリックします。

注： ホストが同じデータセンター内にある場合は、スーパーバイザー 内に移動することができます。ホストを移動するには、ホストをメンテナンス モードにしてから、クラスタにドラッグしてください。

スーパーバイザー からのホストの削除

vSphere 管理者として、コストを節約するためにスーパーバイザー のスケール インが必要になる場合があります。スーパーバイザー のキャパシティを削減するために、単一の vSphere Lifecycle Manager イメージを使用するクラスタから ESXi ホストを削除できます。

NSX を使用して構成されるスーパーバイザー からホストを削除すると、vSphere IaaS control plane によって Spherelet の構成がクリアされ、ESXi ホストの Spherelet プロセスが停止します。その後、vSphere Lifecycle Manager によって Spherelet VIB およびイメージがホストからアンインストールされ、vSphere IaaS control plane によってクラスタ制御プレーンからホスト メタデータが削除されます。

前提条件

クラスタからホストを削除する前に、そのホストで実行されているすべての仮想マシンをパワーオフするか、新しいホストに仮想マシンを移行する必要があります。

手順

- 1 vSphere Client で、ホストを削除するクラスタに移動します。
- 2 ホストを右クリックし、ポップアップ メニューで [メンテナンス モードへの切り替え] を選択します。
- 3 表示される確認のダイアログ ボックスで、[はい] をクリックします。

確認のダイアログ ボックスでは、パワーオフ状態の仮想マシンを他のホストに移動するかどうか尋ねられます。それらの仮想マシンをクラスタ内のホストに登録したままにする場合は、このオプションをオンにします。

ホストのアイコンが変わり、名前に「メンテナンス モード」と括弧書きで追加されます。

- 4 インベントリからホストのアイコンを選択し、新しい場所にドラッグします。

ホストは、別のクラスタまたは別のデータセンターに移動できます。

vCenter Server により、ホストが新しい場所に移動されます。

- 5 ホストを右クリックし、ポップアップ メニューから [メンテナンス モードの終了] を選択します。
- 6 (オプション) 必要に応じて仮想マシンを再起動します。

スーパーバイザー クラスタの無効化

単一の vSphere Lifecycle Manager イメージを使用する vSphere クラスタから vSphere IaaS control plane を無効にして、従来のワークロードに使用できるようにすることができます。

クラスタで vSphere IaaS control plane を無効にすると、vSphere Lifecycle Manager によって各 ESXi ホストから Spherelet の VIB とイメージがアンインストールされるとともに、WCP サービスによってクラスタ内のすべてのワークロードが停止し、削除されます。

手順

- 1 vSphere Client メニューから、[ワークロード管理] を選択します。
- 2 [クラスタ] タブを選択します。
- 3 vSphere IaaS control plane を無効にするクラスタを選択します。
- 4 [無効化] をクリックします。

[クラスタの無効化] ダイアログ ボックスが開き、クラスタですべての Kubernetes ワークロードと NSX 構成が無効になることを示すメッセージが表示されます。

- 5 [無効化] をクリックします。