

vRealize Automation 7.1 以降から 7.4 へのアップ デート

2018 年 10 月 5 日

vRealize Automation 7.4



vmware®

VMware Web サイトで最新の技術ドキュメントをご確認いただけます。

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

VMware の Web サイトでは、最新の製品アップデートを提供しています。

本書に関するご意見、ご要望をお寄せください。フィードバック送信先：

docfeedback@vmware.com

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

ヴィエムウェア株式会社
105-0013 東京都港区浜松町 1-30-5
浜松町スクエア 13F
www.vmware.com/jp

Copyright © 2008–2018 VMware, Inc. All rights reserved. [著作権および商標](#).

内容

更新情報 6

- 1 vRealize Automation 7.1、7.2、または 7.3.x の 7.4 へのアップグレード 7
 - [vRealize Automation のアップグレードの前提条件](#) 7
 - [vRealize Automation のアップグレード チェックリスト](#) 9
 - [vRealize Automation 環境のユーザー インターフェイス](#) 10
- 2 vRealize Automation に統合された VMware 製品のアップグレード 14
 - [vRealize Automation に統合された vRealize Operations Manager のアップグレード](#) 14
 - [vRealize Automation に統合された vRealize Log Insight のアップグレード](#) 15
 - [vRealize Automation に統合された vRealize Business for Cloud のアップグレード](#) 15
- 3 vRealize Automation のアップグレードの準備 16
 - [vRealize Automation のアップグレード前の NSX ネットワークおよびセキュリティ インベントリ データ収集の実行](#) 16
 - [vRealize Automation をアップグレードする場合のバックアップの前提条件](#) 16
 - [既存の vRealize Automation 環境のバックアップ](#) 17
 - [vRealize Automation PostgreSQL レプリケーション モードの非同期設定](#) 19
 - [vRealize Automation アプライアンスの更新のダウンロード](#) 19
 - [CD-ROM ドライブで使用する仮想アプライアンスのアップデートのダウンロード](#) 20
 - [VMware リポジトリからの vRealize Automation アプライアンス更新のダウンロード](#) 20
- 4 vRealize Automation アプライアンスと IaaS コンポーネントのアップデート 22
 - [vRealize Automation アプライアンスと IaaS コンポーネントへのアップデートのインストール](#) 22
- 5 更新プロセスが失敗する場合に IaaS サーバ コンポーネントを個別にアップグレードする 26
 - [vRealize Automation アプライアンスのアップグレード後にアップグレード シェル スクリプトを使用して IaaS コンポーネントをアップグレード](#) 26
 - [vRealize Automation アプライアンスをアップグレードした後に IaaS インストーラ実行可能ファイルを使用して IaaS コンポーネントをアップグレード](#) 28
 - [vRealize Automation アプライアンスのアップグレード後に IaaS コンポーネントをアップグレードするための IaaS インストーラのダウンロード](#) 29
 - [vRealize Automation アプライアンスのアップグレード後の IaaS コンポーネントのアップグレード](#) 30
 - [組み込み vRealize Orchestrator コントロール センターへのアクセスのリストア](#) 34

6	vRealize Automation をアップグレードした後の vRealize Orchestrator のアップグレード	36
	外部 vRealize Orchestrator サーバから vRealize Automation への移行	36
	外部 Orchestrator および組み込み Orchestrator のコントロール センターの違い	37
	外部 vRealize Orchestrator 7.x から vRealize Automation 7.4 への移行	37
	組み込み vRealize Orchestrator サーバの構成	40
	vRealize Automation で使用するスタンドアローン vRealize Orchestrator アプライアンスのアップグレード	41
	デフォルトの VMware リポジトリを使用した Orchestrator Appliance のアップグレード	42
	ISO イメージを使用した Orchestrator Appliance のアップグレード	44
	指定したリポジトリを使用した Orchestrator Appliance のアップグレード	45
	vRealize Automation 7.4 で使用するための vRealize Orchestrator Appliance クラスタのアップグレード	47
7	ロード バランサの有効化	49
8	vRealize Automation のアップグレード後のタスク	50
	ソフトウェア エージェントの TLS 1.2 へのアップグレード	50
	vRealize Automation 仮想マシン テンプレートの更新	50
	ソフトウェア エージェントのアップグレードが必要な仮想マシンの特定	51
	vSphere 上のソフトウェア エージェントのアップグレード	53
	Amazon Web Service または Azure 上のソフトウェア エージェントのアップグレード	55
	vRealize Automation PostgreSQL レプリケーション モードの同期設定	58
	接続テストの実行とアップグレード後のエンドポイントの確認	58
	vRealize Automation からのアップグレード後の NSX ネットワークおよびセキュリティ インベントリ データ収集の実行	59
	レプリカ アプライアンスのクラスタへの参加	59
	高可用性を展開する環境でのポート構成	60
	組み込みの vRealize Orchestrator で高可用性をサポートするための再構成	60
	外部ワークフローのタイムアウト ファイルのリストア	60
	ユーザー用リモート コンソール アクションとの接続の有効化	61
	app.config ファイルに行ったログの変更のリストア	61
	アップグレード後に Manager Service の自動フェイルオーバーを有効にする	61
	Manager Service の自動フェイルオーバーについて	61
9	vRealize Automation アップグレードのトラブルシューティング	63
	Manager Service の自動フェイルオーバーが有効にならない	64
	ロード バランサのタイムアウト エラーでインストールまたはアップグレードに失敗する	66
	IaaS Web サイト コンポーネントのアップグレードに失敗する	66
	実行中の SSL 検証エラーが原因で Manager Service の実行に失敗する	68
	アップグレード後のログインの失敗	69
	vRealize Automation での実体のないノードの削除	69
	高可用性環境アップグレード後にクラスタへの参加コマンドが失敗したように表示される	69
	PostgreSQL データベースのアップグレード マージが成功しない	70

レプリカ vRealize Automation アプライアンスのアップデートに失敗する	71
.xml ファイルのバックアップ コピーによってシステムがタイムアウトする	72
IaaS アップグレードの除外	72
vRealize Automation で新規ディレクトリを作成できない	73
vRealize Automation レプリカ仮想アプライアンスのアップデートがタイムアウトになる	73
アップグレード中に作成された展開が一部の仮想マシンに存在しない	74
信頼されていない証明書に関するエラー	74
vRealize Automation のインストールまたはアップグレードが失敗する	75
DEM および DEO コンポーネントを更新できない	75
更新で管理エージェントのアップグレードに失敗する	75
管理エージェントのアップグレードに失敗する	76
デフォルトのタイムアウト設定が原因で vRealize Automation のアップデートに失敗する	77
高可用性環境での IaaS のアップグレードが失敗する	78
アップグレードの問題の回避	78

更新情報

次の表では、今回の製品リリースに対応する『vRealize Automation 7.1 以降から 7.4 へのアップグレード』の変更点を記載しています。

リビジョン	説明
2018 年 10 月 05 日	マイナー更新。
2018 年 6 月 15 日	<ul style="list-style-type: none">■ 「レプリカ vRealize Automation アプライアンスのアップデートに失敗する」 でログインを明記しました。■ 「vRealize Automation アプライアンスの更新のダウンロード」 にナレッジベース リファレンスを追加しました。
2018 年 5 月 3 日	<ul style="list-style-type: none">■ 「IaaS アップグレードの除外」 を改訂しました。■ 「アップグレードの問題の回避」 を改訂しました。■ 「vRealize Automation で使用するスタンドアローン vRealize Orchestrator アプライアンスのアップグレード」 を改訂しました。
2018 年 4 月 12 日	初版リリース。

vRealize Automation 7.1、7.2、または 7.3.x の 7.4 へのアップグレード

1

現在の vRealize Automation 7.1、7.2、または 7.3.x 環境を 7.4 にアップグレードすることができます。環境のアップグレードには、このバージョンに固有のアップグレード手順を使用します。

インプレース アップグレードは 3 段階のプロセスです。次の順序で、現在の環境のコンポーネントをアップグレードします。

- 1 vRealize Automation アプライアンス
- 2 IaaS Web サーバ
- 3 vRealize Orchestrator

すべての製品コンポーネントを同一バージョンにアップグレードする必要があります。

vRealize Automation 7.2 以降、JFrog Artifactory Pro は vRealize Automation アプライアンスにはバンドルされなくなりました。vRealize Automation の以前のバージョンからアップグレードする場合、アップグレード プロセスで JFrog Artifactory Pro は削除されます。詳細については、[ナレッジベースの記事 KB2147237](#) を参照してください。

この章には、次のトピックが含まれています。

- [vRealize Automation のアップグレードの前提条件](#)
- [vRealize Automation のアップグレード チェックリスト](#)
- [vRealize Automation 環境のユーザー インターフェイス](#)

vRealize Automation のアップグレードの前提条件

vRealize Automation 7.1、7.2、7.3.x 環境の 7.4 へのアップグレードを実行する前に、次の前提条件を確認します。

システム構成要件

アップグレードを開始する前に、次の前提条件が完了していることを確認します。

- 環境内のアプライアンスとサーバのすべてが、最新バージョンのシステム要件を満たしているか確認します。
[VMware vRealize Automation のドキュメント](#)の vRealize Automation サポート マトリックスを参照してください。
- VMware の他の製品との互換性の詳細については、VMware Web サイトの VMware 製品の相互運用性マトリックスを参照してください。

- アップグレードする vRealize Automation の動作状況が安定していることを確認します。アップグレード前にすべての問題を修正します。
- ロード バランサのタイムアウト設定を、デフォルト設定から 10 分以上に変更したことを確認します。

ハードウェア構成要件

お使いの環境のハードウェアが vRealize Automation 7.4 に適していることを確認します。

vRealize Automation ドキュメントのリファレンス アーキテクチャで、「<vRealize Automation のハードウェアの仕様とキャパシティの上限>」を参照してください。

アップグレードを開始する前に、次の前提条件が完了していることを確認します。

- アップグレードを実行するには、少なくとも 18 GB の RAM、4 つの CPU、Disk1 = 50 GB、Disk3=25 GB、Disk4=50 GB が必要です。

仮想マシンが vCloud Networking and Security 上に配置されている場合は、多くの RAM 容量の割り当てが必要になる場合があります。

vCloud Networking and Security の一般的なサポートは終了しましたが、VCNS カスタム プロパティは NSX の用途に対して引き続き有効です。詳細については、[ナレッジベースの記事 KB2144733](#) を参照してください。

- これらのノードには、少なくとも 5 GB の空きディスク容量が必要です。
 - プライマリ IaaS Web サイト
 - Microsoft SQL データベース
 - Model Manager
- Model Manager Data がインストールされているプライマリ IaaS Web サイト ノードに、JAVA SE Runtime Environment 8 Update 161 (64 ビット) 以降がインストールされている必要があります。Java をインストールした後、JAVA_HOME 環境変数に新しいバージョンを設定する必要があります。
- アップグレードをダウンロードして実行するには、次のリソースが必要です。
 - ルート パーティションに少なくとも 5 GB
 - マスター vRealize Automation アプライアンス の **/storage/db** パーティションに 5 GB
 - 各レプリカ仮想アプライアンスのルート パーティションに 5 GB
- **/storage/log** サブフォルダを確認し、以前のアーカイブ ZIP ファイルがあれば削除して容量をクリーンアップします。

一般的な前提条件

アップグレードを開始する前に、次の前提条件が完了していることを確認します。

- vRealize Automation アップグレードの影響を受けるまたはこのアップグレードに参加する、すべてのデータベースおよびロード バランサにアクセスできる。
- アップグレードの実行中、ユーザーがシステムを使用できないようにする。
- vRealize Automation に対してクエリを実行するアプリケーションがあれば、それを無効にする。

- Microsoft 分散トランザクション コーディネーター (MSDTC) が、すべての vRealize Automation および関連する SQL サーバ上で有効であることを確認する。手順については、[ナレッジベースの記事 KB2089503](#) を参照してください。
- 組み込みの PostgreSQL データベースで構成されている分散環境をアップグレードする場合は、次の手順を実行します。
 - a レプリカ ホストをアップグレードする前に、マスター ホストで **pgdata** ディレクトリ内のファイルを調べます。
 - b マスター ホスト上の PostgreSQL データ フォルダ (`/var/vmware/vpostgres/current/pgdata/`) に移動します。
 - c **pgdata** ディレクトリ内で開かれているファイルがあればすべて閉じ、.swp サフィックスを持つすべてのファイルを削除します。
 - d このディレクトリ内のすべてのファイルの所有者 (postgres:users) が正しいことを確認します。

さらに、カスタム プロパティの名前にスペースが使用されていないことを確認します。vRealize Automation のこのリリースにアップグレードする前に、カスタム プロパティ名からスペース文字を削除します。たとえば、スペースをアンダースコア文字で置き換えます。これにより、アップグレードされた vRealize Automation 環境でカスタム プロパティが認識されるようになります。vRealize Automation カスタム プロパティ名にスペースは使用できません。この問題は、以前のリリースの vRealize Automation、vRealize Orchestrator、またはその両方で使用されるカスタム プロパティにスペースが含まれており、この vRealize Orchestrator をアップグレードした環境を使用する場合に影響があります。

vRealize Automation のアップグレード チェックリスト

vRealize Automation 7.1、7.2 または 7.3.x を 7.4 にアップデートする際は、vRealize Automation 内のすべてのコンポーネントを特定の順序で更新します。

アップグレードの順序は、アップグレードする対象が最小環境なのか、それとも複数の vRealize Automation アプライアンスがある分散環境なのかによって異なります。

アップグレードを完了するまでの作業を追跡するため、チェックリストを使用します。タスクは示された順序で行うようにしてください。

表 1-1. vRealize Automation 最小環境をアップグレードする場合のチェックリスト

タスク	方法
 vRealize Automation 7.1、7.2、または 7.3.x を 7.4 にアップデートする前に、NSX ネットワークおよびセキュリティ インベントリ データ収集を実行します。これは vRealize Automation が NSX と統合されている場合のみ必要です。	「vRealize Automation のアップグレード前の NSX ネットワークおよびセキュリティ インベントリ データ収集の実行」 を参照してください。
 現在のインストール環境をバックアップする。これは重要な手順です。	システムのバックアップ方法とリストア方法の詳細については、「 既存の vRealize Automation 環境のバックアップ 」を参照してください。 一般情報については、 http://www.vmware.com/pdf/vrealize-backup-and-restore-netbackup.pdf の「Symantec Netbackup を使用したバックアップとリストアの構成」を参照してください。

表 1-1. vRealize Automation 最小環境をアップグレードする場合のチェックリスト (続き)

タスク	方法
 vRealize Automation アプライアンスにアップデートをダウンロードする。	「vRealize Automation アプライアンスの更新のダウンロード」 を参照してください。
 vRealize Automation アプライアンスと IaaS コンポーネントのアップデートをインストールする。	「vRealize Automation アプライアンスと IaaS コンポーネントへのアップデートのインストール」 を参照してください。

表 1-2. vRealize Automation 分散環境をアップグレードする場合のチェックリスト

タスク	方法
 vRealize Automation 7.1、7.2、または 7.3.x を 7.4 にアップデートする前に、NSX ネットワークおよびセキュリティ インベントリ データ収集を実行します。これは vRealize Automation が NSX と統合されている場合のみ必要です。	「vRealize Automation のアップグレード前の NSX ネットワークおよびセキュリティ インベントリ データ収集の実行」 を参照してください。
 現在のインストールをバックアップする。これは重要な手順です。	システムのバックアップ方法とリストア方法の詳細については、「 既存の vRealize Automation 環境のバックアップ 」を参照してください。 詳細情報については、 http://www.vmware.com/pdf/vrealize-backup-and-restore-netbackup.pdf の「Symantec Netbackup を使用したバックアップとリストアの構成」(Configuring Backup and Restore by Using Symantec Netbackup) を参照してください。
 vRealize Automation 7.3.x からアップデートする場合は、PostgreSQL の自動フェイルオーバーを無効にします。	「vRealize Automation PostgreSQL レプリケーション モードの非同期設定」 を参照してください。
 vRealize Automation アプライアンスにアップデートをダウンロードする。	「vRealize Automation アプライアンスの更新のダウンロード」 を参照してください。
 ロード バランサを無効にする。	ロード バランサのドキュメントを参照してください。
 マスター vRealize Automation アプライアンスと IaaS コンポーネントのアップデートをインストールする。	「vRealize Automation アプライアンスと IaaS コンポーネントへのアップデートのインストール」 を参照してください。
注: 分散環境のマスター アプライアンスにアップデートをインストールする必要があります。	
 ロード バランサを有効にする。	章 7 「ロード バランサの有効化」

vRealize Automation 環境のユーザー インターフェイス

vRealize Automation 環境は、複数のインターフェイスで使用および管理します。

ユーザー インターフェイス

これらの表は、vRealize Automation 環境を管理するために使用するインターフェイスを示しています。

表 1-3. vRealize Automation 管理コンソール

目的	アクセス	必要な認証情報
<p>以下のシステム管理者のタスクには、vRealize Automation コンソールを使用します。</p> <ul style="list-style-type: none"> ■ テナントを追加します。 ■ vRealize Automation ユーザー インターフェイスをカスタマイズします。 ■ メール サーバを構成します。 ■ イベント ログを表示します。 ■ vRealize Orchestrator を構成します。 	<p>1 ブラウザを起動し、仮想アプライアンスの完全修飾ドメイン名を使用して、vRealize Automation アプライアンスのスプラッシュ ページを開きます。</p> <p><code>https://<vra-vd-hostname.domain.name></code></p> <p>2 [vRealize Automation コンソール] をクリックします。</p> <p>vRealize Automation コンソールを開くには、次の URL を使用することもできます : <code>https://<vra-vd-hostname.domain.name>/vcac</code></p> <p>3 ログインします。</p>	<p>システム管理者ロールを持つユーザーである必要があります。</p>

表 1-4. vRealize Automation テナント コンソール。このインターフェイスは、サービスやリソースの作成および管理に使用するプライマリ ユーザー インターフェイスです。

目的	アクセス	必要な認証情報
<p>以下のタスクには、vRealize Automation を使用します。</p> <ul style="list-style-type: none"> ■ 新しい IT サービス ブループリントを申請します。 ■ クラウドおよび IT リソースを作成および管理します。 ■ カスタム グループを作成および管理します。 ■ ビジネス グループを作成、管理します。 ■ ユーザーにロールを割り当てます。 	<p>1 ブラウザを起動し、仮想アプライアンスの完全修飾ドメイン名とテナントの URL 名を使用して、テナントの URL を入力します。</p> <p><code>https://<vra-vd-hostname.domain.name>/vcac/org/<tenant_URL_name></code></p> <p>2 ログインします。</p>	<p>以下の 1 つ以上のロールを持つユーザーである必要があります。</p> <ul style="list-style-type: none"> ■ アプリケーション アーキテクト ■ 承認管理者 ■ カタログ管理者 ■ コンテナ管理者 ■ コンテナ アーキテクト ■ 健全性サービス ユーザー ■ インフラストラクチャ アーキテクト ■ セキュアなエクスポートの利用者 ■ ソフトウェア アーキテクト ■ テナント管理者 ■ XaaS アーキテクト

表 1-5. vRealize Automation アプライアンス管理。このインターフェイスは、仮想アプライアンス管理インターフェイス (VAMI) と呼ばれます。

目的	アクセス	必要な認証情報
<p>以下のタスクには vRealize Automation アプライアンス管理を使用します。</p> <ul style="list-style-type: none"> 登録されているサービスのステータスを表示します。 システム情報を表示、およびアプライアンスを再起動またはシャットダウンします。 カスタム エクスペリエンス改善プログラムへの参加を管理します。 ネットワーク ステータスを表示します。 更新ステータスを表示、およびアップデートをインストールします。 管理設定を管理します。 vRealize Automation ホスト設定を管理します。 SSO の設定を管理します。 製品ライセンスを管理します。 vRealize Automation Postgres データベースを設定します。 vRealize Automation メッセージングを設定します。 vRealize Automation ログを設定します。 IaaS コンポーネントをインストールします。 既存の vRealize Automation 環境から移行します。 IaaS コンポーネントの証明書を管理します。 Xenon サービスを設定します。 	<ol style="list-style-type: none"> ブラウザを起動し、仮想アプライアンスの完全修飾ドメイン名を使用して、vRealize Automation アプライアンスのブラッシュ ページを開きます。 https://<vra-va-hostname.domain.name> [vRealize Automation アプライアンス管理] をクリックします。 次の URL を使用して vRealize Automation アプライアンス管理を開くこともできます : https://<vra-va-hostname.domain.name:5480> ログインします。 	<ul style="list-style-type: none"> ユーザー名 : root パスワード : vRealize Automation アプライアンスを展開したときに 入力したパスワード。

表 1-6. vRealize Orchestrator クライアント

目的	アクセス	必要な認証情報
<p>以下のタスクには、vRealize Orchestrator クライアントを使用します。</p> <ul style="list-style-type: none"> アクションを作成します。 ワークフローを作成します。 ポリシーを管理します。 パッケージをインストールします。 ユーザーおよびユーザー グループの権限を管理します。 URI オブジェクトにタグを追加します。 インベントリを表示します。 	<ol style="list-style-type: none"> ブラウザを起動し、仮想アプライアンスの完全修飾ドメイン名を使用して、vRealize Automation のブラッシュ ページを開きます。 https://<vra-va-hostname.domain.name> ローカル コンピュータに client.jnlp ファイルをダウンロードするには、[vRealize Orchestrator Client] をクリックします。 client.jnlp ファイルを右クリックして [起動] を選択します。 [続行しますか?] ダイアログボックスで、[続行] をクリックします。 ログインします。 	<p>システム管理者ロールを持つユーザーであるか、または vRealize Orchestrator コントロール センターの認証プロバイダの設定で構成されている vcoadmins グループに属している必要があります。</p>

表 1-7. vRealize Orchestrator コントロール センター

目的	アクセス	必要な認証情報
vRealize Automation に組み込まれているデフォルトの vRealize Orchestrator インスタンスの設定を編集するには、vRealize Orchestrator コントロール センターを使用します。	<ol style="list-style-type: none"> 1 ブラウザを起動し、仮想アプライアンスの完全修飾ドメイン名を使用して、vRealize Automation アプライアンスのスプラッシュ ページを開きます。 https://<vra-virtual-hostname.domain.name> 2 [vRealize Automation アプライアンス管理] をクリックします。 次の URL を使用して vRealize Automation アプライアンス管理を開くこともできます : https://<vra-virtual-hostname.domain.name:5480> 3 ログインします。 4 [vRA 設定] - [Orchestrator] の順にクリックします。 5 [Orchestrator ユーザー インターフェイス] を選択します。 6 [開始] をクリックします。 7 Orchestrator ユーザー インターフェイスの URL をクリックします。 8 ログインします。 	<p>ユーザー名</p> <ul style="list-style-type: none"> ■ ロールベースの認証が設定されていない場合は、root と入力します。 ■ ロールベースの認証で設定されている場合は、vRealize Automation ユーザー名を入力します。 <p>パスワード</p> <ul style="list-style-type: none"> ■ ロールベースの認証が設定されていない場合、vRealize Automation アプライアンスを展開したときに入力したパスワードを入力します。 ■ ロールベースの認証でユーザー名が設定されている場合は、ユーザー名に対するパスワードを入力します。

表 1-8. Linux コマンド プロンプト

目的	アクセス	必要な認証情報
vRealize Automation アプライアンス ホストなどのホストでは、以下のタスクには Linux コマンド プロンプトを使用します。 <ul style="list-style-type: none"> ■ サービスの開始または停止 ■ 構成ファイルの編集 ■ コマンドの実行 ■ データの取得 	<ol style="list-style-type: none"> 1 vRealize Automation アプライアンス ホストで、コマンド プロンプトを開きます。 ローカル コンピュータでコマンド プロンプトを開く方法の 1 つは、PuTTY などのアプリケーションを使用して、ホストでセッションを開始することです。 2 ログインします。 	<ul style="list-style-type: none"> ■ ユーザー名 : root ■ パスワード : vRealize Automation アプライアンスを展開したときに作成したパスワード。

表 1-9. Windows コマンド プロンプト

目的	アクセス	必要な認証情報
laaS ホストなどのホスト上で、Windows コマンド プロンプトを使用してスクリプトを実行できます。	<ol style="list-style-type: none"> 1 laaS ホスト上で、Windows にログインします。 ローカル コンピュータからログインする方法の 1 つは、リモート デスクトップ セッションを開始することです。 2 Windows コマンド プロンプトを開きます。 コマンド プロンプトを開く方法の 1 つは、ホスト上で [スタート] アイコンを右クリックし、[コマンド プロンプト] または [コマンド プロンプト (管理者)] を選択することです。 	<ul style="list-style-type: none"> ■ ユーザー名 : 管理者権限を持つユーザー。 ■ パスワード : ユーザーのパスワード。

vRealize Automation に統合された VMware 製品のアップグレード

2

vRealize Automation をアップグレードする場合、vRealize Automation 環境に統合されているすべての VMware 製品を管理する必要があります。

vRealize Automation 環境が 1 つ以上の他の製品に統合されている場合は、他の製品をアップデートする前に vRealize Automation をアップグレードする必要があります。vRealize Business for Cloud が vRealize Automation に統合されている場合は、vRealize Automation をアップグレードする前に vRealize Business for Cloud を登録解除する必要があります。

vRealize Automation をアップグレードする場合は、統合製品を管理するための推奨ワークフローを実行します。

- 1 vRealize Automation をアップグレードします。
- 2 VMware vRealize Operations Manager をアップグレードします。
- 3 VMware vRealize Log Insight をアップグレードします。
- 4 VMware vRealize Business for Cloud をアップグレードします。

このセクションでは、vRealize Automation 環境に統合する場合の vRealize Business for Cloud の管理について詳細に説明します。

この章には、次のトピックが含まれています。

- [vRealize Automation に統合された vRealize Operations Manager のアップグレード](#)
- [vRealize Automation に統合された vRealize Log Insight のアップグレード](#)
- [vRealize Automation に統合された vRealize Business for Cloud のアップグレード](#)

vRealize Automation に統合された vRealize Operations Manager のアップグレード

vRealize Automation のアップグレード後に vRealize Operations Manager をアップグレードします。

手順

- 1 vRealize Automation をアップグレードします。
- 2 vRealize Operations Manager をアップグレードします。詳細については、VMware vRealize Operations Manager のドキュメントの「Updating Your Software」を参照してください。

vRealize Automation に統合された vRealize Log Insight のアップグレード

vRealize Automation のアップグレード後に vRealize Log Insight をアップグレードします。

手順

- 1 vRealize Automation をアップグレードします。
- 2 vRealize Log Insight をアップグレードします。詳細については、VMware vRealize Log Insight のドキュメントの「vRealize Log Insight のアップグレード」を参照してください。

vRealize Automation に統合された vRealize Business for Cloud のアップグレード

vRealize Automation 環境をアップグレードする場合は、vRealize Business for Cloud への接続を一度登録解除して、再度登録する必要があります。

vRealize Automation 環境をアップグレードする場合は、この手順を実行して vRealize Business for Cloud のサービスの継続を確認します。

手順

- 1 vRealize Automation から vRealize Business for Cloud を登録解除します。vRealize Business for Cloud のドキュメントの「vRealize Business for Cloud の vRealize Automation からの登録解除」を参照してください。
- 2 vRealize Automation をアップグレードします。
- 3 必要に応じて、vRealize Business for Cloud for Cloud をアップグレードします。vRealize Business for Cloud のドキュメントの「vRealize Business for Cloud のアップグレード」を参照してください。
- 4 vRealize Automation に vRealize Business for Cloud を登録します。vRealize Business for Cloud のドキュメントの「vRealize Business for Cloud の vRealize Automation への登録」を参照してください。

vRealize Automation のアップグレードの準備

3

vRealize Automation 7.1、7.2、7.3 を 7.4 にアップグレードする前に、次のタスクを完了します。

これらのタスクはチェックリストに表示されている順序で実行します。[「vRealize Automation のアップグレードチェックリスト」](#)を参照してください。

この章には、次のトピックが含まれています。

- [vRealize Automation のアップグレード前の NSX ネットワークおよびセキュリティ インベントリ データ収集の実行](#)
- [vRealize Automation をアップグレードする場合のバックアップの前提条件](#)
- [既存の vRealize Automation 環境のバックアップ](#)
- [vRealize Automation PostgreSQL レプリケーション モードの非同期設定](#)
- [vRealize Automation アプライアンスの更新のダウンロード](#)

vRealize Automation のアップグレード前の NSX ネットワークおよびセキュリティ インベントリ データ収集の実行

vRealize Automation 7.1、7.2、7.3.x から 7.4 にアップグレードする前に、vRealize Automation 7.1、7.2、7.3.x 環境で NSX ネットワークとセキュリティ インベントリのデータ収集を実行する必要があります。

このデータ収集は、vRealize Automation 7.4 で 7.1、7.2、および 7.3.x の展開用にロード バランサの再構成アクションを行うために必要です。

手順

- ◆ vRealize Automation 7.4 にアップグレードする前に、バージョン 7.1、7.2、7.3.x で NSX ネットワークとセキュリティ インベントリ データの収集を実行します。「[vRealize Automation の管理](#)」の「[手動によるエンドポイント データ収集の開始](#)」を参照してください。

次のステップ

[「vRealize Automation をアップグレードする場合のバックアップの前提条件」](#)。

vRealize Automation をアップグレードする場合のバックアップの前提条件

vRealize Automation 7.1、7.2 または 7.3.x を 7.4 にアップグレードする前に、バックアップの前提条件を満たします。

前提条件

- 移行前の環境が正しくインストールされ、構成されていることを確認します。
- vSphere Client にログインし、移行前の環境の各アプライアンスで、次のディレクトリのすべての vRealize Automation アプライアンス構成ファイルをバックアップします。
 - /etc/vcac/
 - /etc/vco/
 - /etc/apache2/
 - /etc/rabbitmq/
- IaaS Microsoft SQL Server データベースをバックアップします。詳細については、SQL Server データベースの完全バックアップの作成に関する記事 [Microsoft Developer Network](#) を参照してください。
- カスタマイズしたすべてのファイル (**DataCenterLocations.xml** など) をバックアップします。
- 各仮想アプライアンスおよび IaaS サーバのスナップショットを作成します。vRealize Automation のアップグレードが失敗した場合に備えて、システム全体のバックアップに関する基本ガイドラインには必ず従ってください。『vRealize Automation の管理』の「vRealize Automation インストールのバックアップおよびリカバリ」を参照してください。

次のステップ

[「既存の vRealize Automation 環境のバックアップ」](#)。

既存の vRealize Automation 環境のバックアップ

vRealize Automation 7.1、7.2、または 7.3.x から 7.4 へアップデートする前に、各 Windows ノードの vRealize Automation IaaS サーバおよび各 Linux ノードの vRealize Automation アプライアンスのシャットダウンおよびスナップショットの作成を行います。アップグレードが失敗した場合は、スナップショットを使用して以前の正常な構成に戻り、別のアップグレードを試します。

vRealize Automation の起動については、『vRealize Automation の管理』の「vRealize Automation の起動」を参照してください。

前提条件

- [「vRealize Automation をアップグレードする場合のバックアップの前提条件」](#)。
- vRealize Automation 7.0 以降では、PostgreSQL データベースが常に高可用性モードで構成されます。vRealize Automation アプライアンス管理コンソールにログインし、[vRA 設定 > データベース] の順に選択して現在のマスター ノードを特定します。データベース構成が外部データベースとしてリストされている場合は、この外部データベースの手動バックアップを作成します。
- vRealize Automation Microsoft SQL データベースが IaaS サーバ上でホストされていない場合は、データベースバックアップファイルを作成します。
- アップグレードのためのバックアップの前提条件が完了していることを確認します。

- シャットダウン時にシステムのスナップショットを作成したことを確認します。スナップショットを作成するときには、この方法が推奨されます。『vSphere 6.0 のドキュメント』を参照してください。

注: vRealize Automation アプライアンスと IaaS コンポーネントをバックアップする場合、インメモリ スナップショットと静止スナップショットを無効にします。

- **app.config** ファイルを変更した場合は、そのファイルのバックアップを作成します。[\[app.config ファイルに行ったログの変更のリストア\]](#)を参照してください。
- 外部ワークフロー構成 (xmldb) ファイルのバックアップを作成します。[\[外部ワークフローのタイムアウト ファイルのリストア\]](#)を参照してください。
- 現在のフォルダの外にバックアップ ファイルを保存する場所があることを確認します。[\[.xml ファイルのバックアップ コピーによってシステムがタイムアウトする\]](#)を参照してください。

手順

- 1 vSphere クライアントにログインします。
- 2 各 vRealize Automation IaaS Windows マシン、および各 vRealize Automation アプライアンス ノードを見つけてます。
- 3 各マシンで、[ゲストのシャットダウン] を次の順序でクリックします。
 - a IaaS Windows サーバ マシン
 - b vRealize Automation アプライアンス
- 4 各 vRealize Automation マシンのスナップショットを作成します。
- 5 好みのバックアップ方法を使用して、各アプライアンス ノードの完全バックアップを作成します。
- 6 システムをパワーオンします。『vRealize Automation の管理』の「vRealize Automation の起動」を参照してください。

高可用性の環境がある場合は、この手順に従って仮想アプライアンスをパワーオンします。

- a マスター vRealize Automation アプライアンスを起動します。
- b vRealize Automation アプライアンス管理にログインし、[サービス] をクリックして、ライセンスサービスのステータスが [登録済み] になるまで待機します。
- c 残りの vRealize Automation アプライアンスを同時に起動します。
- d プライマリ Web ノードを起動し、起動が完了するまで待機します。
- e プライマリ Manager Service マシンを起動し、2 ～ 5 分間待機します。

実際の所要時間はサイトの構成によって異なります。

注: セカンダリ マシンでは、Manager Service 自動フェイルオーバーが設定されている場合を除いて、Windows サービスを起動または実行しないでください。

- f Distributed Execution Manager の Orchestrator とワーカー、およびすべての vRealize Automation プロキシ エージェントを起動します。

注: これらのコンポーネントは、任意の順序で起動することができます。他のコンポーネントを起動する前に、コンポーネントの起動が完了するまで待つ必要はありません。

- 7 各 vRealize Automation アプライアンス管理コンソールにログインし、システムが完全に機能することを確認します。
 - a [サービス] をクリックします。
 - b 各サービスが登録済みであることを確認します。

次のステップ

[\[vRealize Automation PostgreSQL レプリケーション モードの非同期設定\]](#)。

vRealize Automation PostgreSQL レプリケーション モードの非同期設定

PostgreSQL 同期レプリケーション モードで動作する分散 vRealize Automation 環境からアップグレードを実行する場合、アップグレード前に、モードを非同期に変更する必要があります。

前提条件

- アップグレード対象となる分散 vRealize Automation 環境があります。
- <https://<vra-virtual-hostname.domain.name>:5480> の vRealize Automation アプライアンス管理に **root** としてログインしています。

手順

- 1 [vRA 設定] - [データベース] の順にクリックします。
- 2 [非同期モード] をクリックして、アクションが完了するまで待機します。
- 3 [同期の状態] 列のすべてのノードのステータスが **非同期** と表示されていることを確認します。

次のステップ

[\[vRealize Automation アプライアンスの更新のダウンロード\]](#)

vRealize Automation アプライアンスの更新のダウンロード

アプライアンス管理コンソールでアップデートの有無をチェックし、次の方法のいずれかを使用して、アップデートをダウンロードすることができます。

最適なアップグレード パフォーマンスを得るためには、ISO ファイルによる方法を使用します。

アプライアンスをアップグレードする際の潜在的な問題を回避する場合や、アプライアンスのアップグレード中に問題が発生した場合は、[VMware ナレッジベースの記事](#)「vRealize Automation upgrade fails due to duplicates in the vRealize Orchestrator database (KB54987)」を参照してください。

CD-ROM ドライブで使用する仮想アプライアンスのアップデートのダウンロード

仮想アプライアンスは、アプライアンスが仮想 CD-ROM ドライブから読み取る ISO ファイルからアップデートできます。これが推奨される方法です。

ISO ファイルをダウンロードし、プライマリ アプライアンスを設定したら、このファイルを使用してアプライアンスをアップグレードします。

前提条件

- 既存の vRealize Automation 環境をバックアップします。
- vRealize Automation アプライアンスをアップデートする前に、アップグレードで使用するすべての CD-ROM ドライブが有効になっていることを確認します。vSphere クライアントで仮想マシンに CD-ROM ドライブを追加する際の詳細については、vSphere のドキュメントを参照してください。

手順

- 1 アップデート リポジトリ ISO ファイルをダウンロードします。
 - a ブラウザを起動し、www.vmware.com の [vRealize Automation 製品ページ](#) に移動します。
 - b [vRealize Automation ダウンロード リソース] をクリックして VMware ダウンロード ページに移動します。
 - c 適切なファイルをダウンロードします。
- 2 システム上でダウンロードしたファイルを探し、このサイズが VMware ダウンロード ページ上のファイルと同一であることを確認します。ダウンロード ページに記載されているチェックサムを使用して、ダウンロードしたファイルの整合性を検証します。詳細については、VMware ダウンロード ページの下にあるリンクを参照してください。
- 3 プライマリ仮想アプライアンスが起動していることを確認します。
- 4 プライマリ仮想アプライアンスの CD-ROM ドライブを、ダウンロードした ISO ファイルに接続します。
- 5 プライマリ vRealize Automation アプライアンスで、vRealize Automation アプライアンスを展開したときに入力したパスワードを使用して、**root** として vRealize Automation アプライアンス管理にログインします。
- 6 [アップデート] タブをクリックします。
- 7 [設定] をクリックします。
- 8 [アップデートリポジトリ] で、[CD-ROM アップデートを使用] を選択します。
- 9 [設定の保存] をクリックします。

VMware リポジトリからの vRealize Automation アプライアンス更新のダウンロード

vmware.com Web サイトの公開リポジトリから vRealize Automation アプライアンスのアップデートをダウンロードできます。

前提条件

- 既存の vRealize Automation 環境をバックアップします。
- vRealize Automation アプライアンスが起動していることを確認します。

手順

- 1 プライマリ vRealize Automation アプライアンスで、vRealize Automation アプライアンスを展開したときに入力したパスワードを使用して、**root** として vRealize Automation アプライアンス管理にログインします。
- 2 [アップデート] タブをクリックします。
- 3 [設定] をクリックします。
- 4 (オプション) [自動アップデート] パネルで、アップデートをチェックする頻度を設定します。
- 5 [リポジトリをアップデート] パネルで、[デフォルトリポジトリの使用] を選択します。
デフォルトリポジトリが正しい VMware.com URL に設定されます。
- 6 [設定の保存] をクリックします。

vRealize Automation アプライアンス と IaaS コンポーネントのアップデート

4

アップグレードの前提条件を満たすことを確認し、仮想アプライアンスのアップデートをダウンロードしたら、vRealize Automation 7.1、7.2、7.3.x アプライアンスに更新をインストールして、7.4 にアップグレードします。

最小環境の場合、vRealize Automation アプライアンスにアップデートをインストールします。分散環境の場合、アップデートはマスター アプライアンス ノードにインストールします。アップデートに必要な時間は、環境およびネットワークによって異なります。アップデートが完了すると、変更された項目が vRealize Automation アプライアンス管理の [アップデート ステータス] ページに表示されます。アプライアンスのアップデートが完了したら、アプライアンスを再起動する必要があります。分散環境でマスター アプライアンスを再起動すると、システムによって各レプリカ ノードが再起動されます。

再起動すると、[アップデート ステータス] ページに「**仮想アプライアンスサービスの開始を待機しています**」と表示されます。IaaS のアップデートは、システムが完全に初期化され、すべてのサービスが実行されると開始されます。[アップデート ステータス] ページで、IaaS のアップグレードの進行を確認できます。最初の IaaS サーバ コンポーネントは、アップデートが完了するまで約 30 分かかります。アップデート中は、「**web1-vra.mycompany.com ノードのサーバ コンポーネントをアップグレード中**」のようなメッセージが表示されます。

各 Manager Service ノードのアップグレード プロセスが完了するたびに、「**mgr-vra.mycompany.com ノードに対して ManagerService の自動フェイルオーバー モードを有効にします**」のようなメッセージが表示されます。vRealize Automation 7.3 以降、アクティブな Manager Service ノードからフェイルオーバー サーバを選択する方法を、以前の手動による選択からシステムによる決定に変更しています。この機能は、アップグレード時に有効になります。この機能によって問題が発生する場合は、「[更新で管理エージェントのアップグレードに失敗する](#)」を参照してください。

vRealize Automation アプライアンスと IaaS コンポーネントへのアップデートのインストール

vRealize Automation 7.1、7.2、7.3.x 仮想アプライアンスにアップデートをインストールして vRealize Automation と IaaS コンポーネントをバージョン 7.4 にアップデートします。

アップデートのインストール中は管理コンソールを閉じないでください。

アップグレード プロセスで問題が発生する場合は、[章 9 「vRealize Automation アップグレードのトラブルシューティング」](#) を参照してください。

注: IaaS 仮想マシン上の管理エージェントのアップグレード中、VMware パブリック証明書が一時的に Trusted Publishers 証明書ストアにインストールされます。管理エージェントのアップグレード プロセスでは、この証明書を使って署名された PowerShell スクリプトが使用されます。アップグレードが終了すると、この証明書は証明書ストアから削除されます。

前提条件

- ダウンロード方法を選択し、その方法の手順を完了していることを確認します。[「vRealize Automation アプライアンスの更新のダウンロード」](#) を参照してください。
- すべての高可用性環境については、[「既存の vRealize Automation 環境のバックアップ」](#) を参照してください。
- ロード バランサを備えた環境では、すべての冗長ノードが無効になっていて、健全性モニターが削除されていることを確認します。詳細については、ロード バランサのドキュメントを参照してください。
 - vRealize Automation アプライアンス
 - IaaS Web サイト
 - IaaS Manager Service
- ロード バランサを備えた環境では、トラフィックがプライマリ ノードのみに送られていることを確認します。
- 以下の手順を実行することで、Microsoft Internet Information Services (IIS) でホストされている IaaS サービスが稼動していることを確認します。
 - a ブラウザを起動し、**https://<webhostname>/Repository/Data/MetaModel.svc** という URL を入力して、Web リポジトリが実行されていることを確認します。成功した場合、エラーは返されず、XML 形式のモデルのリストが表示されます。
 - b IaaS Web サイトにログインし、**Repository.log** ファイルに記録されたステータスが OK であることを確認します。このファイルは VCAC ホーム フォルダの **/Server/Model Manager Web/Logs/Repository.log** にあります。

注: 分散型 IaaS Web サイトの場合は、MMD なしでセカンダリ Web サイトにログインし、Microsoft IIS を一時的に停止します。ロード バランサのトラフィックがプライマリ Web ノードのみを通過していることを確認するには、MetaModel.svc の接続を確認して、Microsoft IIS を再起動します。

- 以下の手順を実行して、すべての IaaS ノードが健全な状態であることを確認します。
 - a プライマリ仮想アプライアンスで、vRealize Automation アプライアンスを展開したときに入力したパスワードを使用して、**root** として vRealize Automation アプライアンス管理にログインします。
 - b [vRA 設定] - [クラスタ] の順に選択します。
 - c [最終接続] の下で、次のことを確認します。
 - テーブル内の IaaS ノードに 30 秒未満の最終接続時間があること。
 - 仮想アプライアンス ノードに 10 分未満の最終接続時間があること。
 IaaS ノードが vRealize Automation アプライアンスと通信していない場合、アップグレードは失敗します。

管理エージェントと仮想アプライアンス間の接続の問題を診断するには、次の手順を実行します。

- 1 リストにない、または [最終接続] 時間が 30 秒を超える各 IaaS ノードにログインします。
 - 2 管理エージェント ログに何らかのエラーが記録されているかどうかを確認します。
 - 3 管理エージェントが実行されていない場合は、サービス コンソールでエージェントを再起動します。
- d 表に実体のないノードが示されていないことを確認します。実体のないノードとは、ホスト上で報告されているのにそのホストに存在しない重複ノードです。実体のないノードはすべて削除する必要があります。詳細については、[「vRealize Automation での実体のないノードの削除」](#)を参照してください。
- クラスタに属していないレプリカ仮想アプライアンスがある場合は、クラスタ テーブルから削除する必要があります。このアプライアンスを削除しないと、レプリカの更新が失敗したことを示す警告メッセージがアップグレード プロセスで表示されます。
 - アップグレードの前に、すべての保存済みおよび進行中の申請が正常に完了したことを確認します。
 - vRealize Automation 7.1、7.2、または 7.3.x アプライアンスのアップデート後に IaaS コンポーネントを手動でアップグレードする場合は、[「IaaS アップグレードの除外」](#)を参照してください。IaaS を手動でアップグレードする場合は、各 IaaS ノードで、管理エージェントを除くすべての IaaS サービスを停止する必要もあります。

手順

- 1 プライマリ vRealize Automation アプライアンスで、vRealize Automation アプライアンスを展開したときに入力したパスワードを使用して、**root** として vRealize Automation アプライアンス管理にログインします。
分散環境では、マスター アプライアンスで管理コンソールを開きます。
- 2 [サービス] をクリックし、すべてのサービスが登録されていることを確認します。
- 3 [vRA 設定] - [データベース] の順に選択し、このアプライアンスがマスター vRealize Automation アプライアンスであることを確認します。
マスター vRealize Automation アプライアンスのみにアップデートをインストールします。各レプリカ vRealize Automation アプライアンスがマスター アプライアンスによって更新されます。
- 4 [更新] - [ステータス] の順に選択します。
- 5 [アップデートの確認] をクリックし、アップデートが利用可能かどうかを確認します。
- 6 (オプション) vRealize Automation アプライアンスのインスタンスの場合は、[アプライアンスのバージョン] 領域で [詳細] をクリックすると、リリース ノートの場所が表示されます。
- 7 [アップデートをインストール] をクリックします。
- 8 [OK] をクリックします。

アップデート処理が進行していることを示すメッセージが表示されます。アップグレード中に行われた変更がシステムによって [更新のサマリ] ページに表示されます。アップデートに必要な時間は、環境およびネットワークによって異なります。

- 9 (オプション) より詳細に更新を監視するには、ターミナルエミュレータを使用してプライマリ アプライアンスにログインします。`/opt/vmware/var/log/vami/updatecli.log` で `updatecli.log` ファイルを表示します。

その他のアップグレード進行状況の情報は次のファイルでも参照できます。

- `/opt/vmware/var/log/vami/vami.log`
- `/var/log/vmware/horizon/horizon.log`
- `/var/log/bootstrap/*.log`

アップグレード中にログアウトした場合、ログ ファイルで更新の進捗状況を継続することができます。

`updatecli.log` ファイルに、アップグレード前のバージョンの vRealize Automation の情報が表示される場合があります。表示されたバージョンは、アップグレード プロセスの中で適切なバージョンに変わります。

- 10 vRealize Automation アプライアンスの更新が完了したら、管理コンソールで [システム] - [再起動] をクリックします。

分散環境では、マスター アプライアンスを再起動するときに、正常にアップグレードされたすべてのレプリカ アプライアンス ノードが再起動されます。

システムが初期化され、すべてのサービスが稼働していると、IaaS の更新が開始されます。IaaS のアップグレードの進行状況を表示するには、[更新] - [ステータス] をクリックします。

- 11 IaaS の更新が完了したら、アプライアンス管理コンソールで [クラスタ] をクリックして、すべての IaaS ノードとコンポーネントのバージョン番号が最新のバージョンになっていることを確認します。

- 12 アプライアンス管理コンソールで [テレメトリ] をクリックします。カスタマー エクスペリエンス向上プログラム (CEIP) への参加に関する注意を読み、プログラムに参加するかしないかを選択します。

CEIP によって収集されるデータの詳細と、VMware がそのデータを使用する目的については、Trust & Assurance Center (<http://www.vmware.com/trustvmware/ceip.html>) を参照してください。

カスタマ エクスペリエンス改善プログラムの詳細については、vRealize Automation の管理 で「vRealize Automation のカスタマ エクスペリエンス改善プログラムへの参加または脱退」を参照してください。

次のステップ

環境でロード バランサを使用している場合は、次の手順を実行します。

- 1 ロード バランサの vRealize Automation 健全性チェックを有効にします。
- 2 すべての vRealize Automation ノードでロード バランサのトラフィックを再度有効にします。

IaaS コンポーネントのアップグレードに失敗した場合は、[章 5 「更新プロセスが失敗する場合に IaaS サーバ コンポーネントを個別にアップグレードする」](#)を参照してください。

更新プロセスが失敗する場合に laaS サーバ コンポーネントを個別にアップ グレードする

5

自動更新プロセスが失敗した場合、laaS コンポーネントを個別にアップグレードすることができます。

vRealize Automation laaS Web サイトおよび Manager Service が正常にアップグレードされた場合は、アップグレード前に作成したスナップショットに戻さずに再度 laaS アップグレードシェルスクリプトを実行できます。同じ仮想マシンにインストールされた複数の laaS コンポーネントのアップグレード時に、再起動の保留イベントが生成されると、アップグレードが失敗することがあります。この場合は、手動による laaS ノードの再起動と、アップグレードの再実行により問題を解決します。アップグレードの失敗が解消されない場合は、VMware サポートに問い合わせるか、以下の手順で手動アップグレードを実行します。

- 1 vRealize Automation アプライアンスを更新前の状態に戻します。
- 2 laaS コンポーネントを更新プロセスから除外するためのコマンドを実行します。[「laaS アップグレードの除外」](#)を参照してください。
- 3 vRealize Automation アプライアンスで更新プロセスを実行します。
- 4 アップグレードシェルスクリプトまたは vRealize Automation 7.4 laaS インストーラ msi パッケージを使用して、laaS コンポーネントを個別に更新します。

この章には、次のトピックが含まれています。

- [vRealize Automation アプライアンスのアップグレード後にアップグレードシェルスクリプトを使用して laaS コンポーネントをアップグレード](#)
- [vRealize Automation アプライアンスをアップグレードした後に laaS インストーラ実行可能ファイルを使用して laaS コンポーネントをアップグレード](#)
- [組み込み vRealize Orchestrator コントロール センターへのアクセスのリストア](#)

vRealize Automation アプライアンスのアップグレード後にアップグレードシェルスクリプトを使用して laaS コンポーネントをアップグレード

各 vRealize Automation 7.1、7.2、7.3.x アプライアンスを 7.4 にアップデートした後に、アップグレードシェルスクリプトを使用して laaS コンポーネントをアップグレードします。

アップデートした vRealize Automation アプライアンスには、各 laaS ノードおよびコンポーネントのアップグレードに使用するシェルスクリプトが含まれています。

仮想マシンの vSphere コンソールまたは SSH コンソール セッションを使用すると、アップグレード スクリプトを実行できます。vSphere コンソールを使用する場合は、スクリプトの実行が中断する断続的なネットワーク接続の問題を回避します。

コンポーネントのアップグレード中にスクリプトを停止すると、そのスクリプトはコンポーネントのアップグレードを完了したときに停止します。ノード上の別のコンポーネントをアップグレードする必要がある場合は、もう一度スクリプトを実行します。

アップグレードが完了すると、アップグレード ログ ファイル (`/opt/vmware/var/log/vami/upgrade-iaas.log`) を開くことによってアップグレード結果を確認できます。

前提条件

- 章 9 「vRealize Automation アップグレードのトラブルシューティング」を確認します。
- すべての vRealize Automation アプライアンスのアップデートが成功していることを確認します。
- すべての vRealize Automation アプライアンスをアップデートしてから、IaaS コンポーネントをアップグレードする前に IaaS サーバを再起動する場合、管理エージェント サービスを除く、Windows のすべての IaaS サービスを停止します。
- マスター vRealize Automation アプライアンス ノードでアップグレード シェル スクリプトを実行する前に、アプライアンス管理コンソールで [サービス] をクリックします。各サービス (iaas-service を除く) が登録済みであることを確認します。
- 各 IaaS ノードに手動で IaaS 管理エージェントをインストールするには、次の手順を実行します。
 - a ブラウザを開き、アプライアンス上の VMware vRealize Automation IaaS インストール ページ (`https://<virtual_appliance_host_FQDN>:5480/installer`) に移動します。
 - b 管理エージェントのインストーラ (vCAC-iaasManagementAgent-Setup.msi) をダウンロードします。
 - c 各 vRealize Automation IaaS マシンにログインし、管理エージェントのインストーラを使用して管理エージェントをアップグレードします。Windows 管理エージェント サービスを再起動します。
- プライマリ IaaS Web サイトおよび Model Manager ノードに JAVA SE Runtime Environment 8 Update 161 (64 ビット) 以降がインストールされていることを確認します。Java をインストールした後、各サーバ ノード上で環境変数 JAVA_HOME に新しいバージョンを設定する必要があります。
- 各 IaaS Web サイト ノードにログインして、作成日が **web.config** ファイルの変更日よりも前であることを確認します。**web.config** ファイルの作成日が変更日以降である場合は、「[IaaS Web サイト コンポーネントのアップグレードに失敗する](#)」の手順を実行します。
- 各 IaaS ノードにアップグレードされた IaaS 管理エージェントが存在することを確認するには、各 IaaS ノードで次の手順を実行します。
 - a vRealize Automation アプライアンス管理コンソールにログインします。
 - b [vRA 設定] - [クラスタ] の順に選択します。
 - c 各 IaaS ノードですべてのインストール済みコンポーネントのリストを展開して、IaaS 管理エージェントを見つけます。
 - d 管理エージェントのバージョンが最新であることを確認します。

- [「IaaS アップグレードの除外」](#)。
- ロールバックする必要がある場合に備えて、IaaS Microsoft SQL Server データベースのバックアップにアクセスできることを確認します。
- 展開で IaaS サーバのスナップショットが利用できることを確認します。
アップグレードが失敗した場合、スナップショットとデータベース バックアップに戻り、別のアップグレードを試します。

手順

- 1 vRealize Automation アプライアンス ホスト上で新しいコンソール セッションを開きます。root アカウントを使用してログインします。
- 2 ディレクトリを `/usr/lib/vcac/tools/upgrade/` に変更します。

`./upgrade` シェル スクリプトを実行する前に、すべての IaaS 管理エージェントがアップグレードされており、健全な状態にする必要があります。アップグレード シェル スクリプトを実行したときにいずれかの IaaS 管理エージェントに問題がある場合は、[「更新で管理エージェントのアップグレードに失敗する」](#)を参照してください。
- 3 アップグレード スクリプトを実行します。
 - a コマンド プロンプトで `./upgrade` と入力します。
 - b Enter キーを押します。

IaaS アップグレード プロセスの説明については、[章 4 「vRealize Automation アプライアンスと IaaS コンポーネントのアップデート」](#)を参照してください。

アップグレード シェル スクリプトが失敗した場合は、`upgrade-iaas.log` ファイルを確認します。

問題を修正した後、もう一度アップグレード スクリプトを実行できます。

次のステップ

- 1 [「組み込み vRealize Orchestrator コントロール センターへのアクセスのリストア」](#)。
- 2 展開環境でロード バランサを使用している場合、vRealize Automation 健全性モニターとすべてのノードへのトラフィックを再度有効にします。

詳細については、vRealize Automation のロード バランシングを参照してください。

vRealize Automation アプライアンスをアップグレードした後に IaaS インストーラ実行可能ファイルを使用して IaaS コンポーネントをアップグレード

vRealize Automation 7.1、7.2、または 7.3.x アプライアンスを 7.4 にアップグレードした後、この代替方法を使用して IaaS コンポーネントをアップグレードすることができます。

vRealize Automation アプライアンスのアップグレード後に IaaS コンポーネントをアップグレードするための IaaS インストーラのダウンロード

vRealize Automation アプライアンスを 7.4 にアップグレードした後、アップグレードする IaaS コンポーネントがインストールされているマシンに IaaS インストーラをダウンロードします。

この手順の間に証明書の警告が表示された場合は、無視して構いません。

注: アップグレード プロセスでは、Manager Service のパッシブ バックアップ インスタンスを除き、すべてのサービスの起動タイプを [自動] に設定する必要があります。サービスを [手動] に設定すると、アップグレード プロセスが失敗します。

前提条件

- IaaS のインストール マシンに、Microsoft .NET Framework 4.5.2 以降がインストールされていることを確認します。.NET インストーラは、vRealize Automation のインストーラ Web ページからダウンロードできます。サービスをシャットダウンし、インストールの一環としてマシンを再起動した後、.NET を 4.5.2 に更新した場合は、管理エージェントを除くすべての IaaS サービスを手動で停止する必要があります。
- ダウンロードに Internet Explorer を使用する場合、厳密なセキュリティ設定が有効になっていないことを確認します。検索バーに **res://iesetup.dll/SoftAdmin.htm** と入力して、Enter キーを押します。
- アップグレードする IaaS コンポーネントが 1 つ以上インストールされている Windows サーバにローカル管理者としてログインします。

手順

- 1 Web ブラウザを起動します。
- 2 Windows インストーラのダウンロード ページの URL を入力します。
たとえば、**https://<vcac-va-hostname.domain.name>:5480/installer** などとし、ここで <vcac-va-hostname.domain.name> はプライマリ（マスター）vRealize Automation アプライアンス ノードの名前になります。
- 3 [IaaS インストーラ] リンクをクリックします。
- 4 プロンプトが表示されたら、インストーラ ファイル (setup__<vcac-va-hostname.domain.name>@5480.exe) をデスクトップに保存します。

ファイル名は変更しないでください。インストールの vRealize Automation アプライアンス への接続に使用されます。

次のステップ

[\[vRealize Automation アプライアンスのアップグレード後の IaaS コンポーネントのアップグレード\]](#)。

vRealize Automation アプライアンスのアップグレード後の IaaS コンポーネントのアップグレード

vRealize Automation アプライアンスを 7.4 にアップグレードした後、PostgreSQL データベースをアップグレードし、IaaS コンポーネントがインストールされているすべてのシステムを構成する必要があります。この手順は、最小および分散インストールに対して使用できます。

注: IaaS インストーラは、アップグレードする IaaS コンポーネントが含まれているマシンに存在する必要があります。外部の場所からインストーラを実行することはできません。ただし、Microsoft SQL データベースは Web ノードからリモートでアップグレードすることもできます。

展開で IaaS サーバのスナップショットが利用できることを確認します。アップグレードが失敗した場合は、スナップショットに戻り、別のアップグレードを試すことができます。

サービスが次の順序でアップグレードされるようにアップグレードを実行します。

1 IaaS Web サイト

ロード バランサを使用している場合は、プライマリ以外のすべてのノードのトラフィックを無効にします。

1 つのサーバのアップグレードを完了してから、Web サイト サービスを実行している次のサーバをアップグレードします。Model Manager Data コンポーネントがインストールされているサーバからアップグレードします。

外部 Microsoft SQL データベースの手動アップグレードを実行している場合は、外部 SQL をアップグレードしてから、Web ノードをアップグレードする必要があります。外部 SQL は Web ノードからリモートでアップグレードできます。

2 Manager Service

パッシブ Manager Service をアップグレードする前に、アクティブな Manager Service をアップグレードします。

SQL インスタンスで SSL 暗号化が有効にされていない場合には、[アップグレード構成] ダイアログ ボックスの SQL 定義の横にある [SSL 暗号化] チェックボックスのチェックを外します。

3 DEM orchestrator とワーカー

すべての DEM orchestrator とワーカーをアップグレードします。1 台のサーバのアップグレードを完了してから、次のサーバをアップグレードします。

4 エージェント

1 台のサーバのアップグレードを完了してから、エージェントを実行している次のサーバをアップグレードします。

5 管理エージェント

アップグレード手順の一部として自動的にアップデートされます。

あるサーバで異なるサービスを使用している場合は、アップグレードにより、サービスが正しい順序でアップデートされます。たとえば、サイト内に同一のサーバ上に Web サイトと Manager Service がある場合、両方をアップデート対象として選択します。アップグレード インストーラがアップデートを正しい順序で適用します。1 台のサーバのアップグレードを完了してから、別のサーバのアップグレードを開始する必要があります。

注: 環境でロード バランサを使用する場合は、プライマリ アプライアンスがロード バランサに接続されている必要があります。vRealize Automation アプライアンスの他のすべてのインスタンスは、キャッシュ エラーを回避するためにアップグレードを適用する前に、ロード バランサのトラフィックに対して無効にする必要があります。

前提条件

- 既存の vRealize Automation 環境をバックアップします。
- すべての vRealize Automation アプライアンスをアップデートしてから、IaaS コンポーネントをアップグレードする前に IaaS サーバを再起動する場合、管理エージェント サービスを除く、サーバ上のすべての IaaS Windows サービスを停止します。
- [\[vRealize Automation アプライアンスのアップグレード後に IaaS コンポーネントをアップグレードするための IaaS インストーラのダウンロード\]](#)。
- プライマリ IaaS Web サイト、Microsoft SQL データベース、および Model Manager ノードに JAVA SE Runtime Environment 8 Update 161 (64 ビット) 以降がインストールされていることを確認します。Java をインストールした後、各サーバ ノード上で環境変数 JAVA_HOME に新しいバージョンを設定する必要があります。
- 作成日が **web.config** ファイルの変更日よりも前であることを確認します。**web.config** ファイルの作成日が変更日以降である場合は、[\[IaaS Web サイト コンポーネントのアップグレードに失敗する\]](#)の手順を実行します。
- 次の手順を実行して、Microsoft 分散トランザクション コーディネータ (DTC) を再構成します。

注: 分散トランザクション コーディネータが有効になっていても、ファイアウォールがオンになっている場合は分散トランザクションが失敗する可能性があります。

- a vRealize Automation アプライアンスで、[スタート] - [管理ツール] - [コンポーネント サービス] の順に選択します。
- b [コンポーネント サービス] - [コンピューター] - [マイ コンピューター] - [分散トランザクション コーディネーター] の順に展開します。
- c 該当するタスクを選択します。
 - ローカル スタンドアロン DTC の場合、[ローカル DTC] を右クリックして [プロパティ] を選択します。
 - クラスタ化された DTC の場合、[クラスタ化された DTC] を展開し、名前付きのクラスタ化された DTC を右クリックして [プロパティ] を選択します。
- d [セキュリティ] をクリックします。
- e 次のすべてのオプションを選択します。
 - [ネットワーク DTC アクセス]
 - [リモート クライアントを許可する]

- [受信を許可する]
- [送信を許可する]
- [相互認証を必要とする]

f [OK] をクリックします。

手順

- 1 ロード バランサを使用している場合は、使用環境で次の準備を行います。
 - a Model Manager Data が含まれている IaaS Web サイト ノードが、ロード バランサのトラフィックに対し有効になっていることを確認します。
`<vCAC Folder>\Server\ConfigTool` フォルダがあれば、このノードを特定できます。
 - b ロード バランサのトラフィックに対し、その他すべての IaaS Web サイトと、プライマリ以外の Manager Service を無効にします。
- 2 セットアップ ファイル `setup__<vrealize-automation-appliance-FQDN>@5480.exe` を右クリックして、[管理者として実行] を選択します。
- 3 [次へ] をクリックします。
- 4 使用許諾契約に同意し、[次へ] をクリックします。
- 5 [ログイン] ページで現在の導入環境の管理者認証情報を入力します。
 ユーザー名は **root** で、パスワードはアプライアンスを展開したときに指定したパスワードです。
- 6 [証明書の受け入れ] を選択します。
- 7 [インストール タイプ] ページで、[アップグレード] が選択されていることを確認します。
 [アップグレード] が選択されていない場合は、このシステム上のコンポーネントがすでにこのバージョンにアップグレードされています。
- 8 [次へ] をクリックします。
- 9 アップグレード設定を構成します。

オプション	アクション
Model Manager Data をアップグレードする場合	vCAC Server セクションで [Model Manager Data] チェック ボックスを選択します。 デフォルトではこのチェック ボックスは選択されています。Model Manager Data のアップグレードは 1 度のみです。複数のマシンで設定ファイルを実行し、分散インストールをアップグレードする場合、Web サーバは、Web サーバと Model Manager Data とのバージョンが一致しないと機能が停止します。Model Manager Data とすべての Web サーバをアップグレードしたら、すべての Web サーバが機能します。
Model Manager Data をアップグレードしない場合	vCAC Server セクションで [Model Manager Data] を選択解除します。

オプション	アクション
Model Manager Data でカスタマイズされたワークフローを最新バージョンで保存するには	<p>Model Manager Data をアップグレードする場合は、拡張性ワークフロー セクションで [ワークフローを最新バージョンに維持する] チェック ボックスを選択します。</p> <p>デフォルトではこのチェック ボックスは選択されています。カスタマイズされたワークフローが常に維持されます。このチェック ボックスはバージョン順のみを決定します。vRealize Automation Designer を使用して Model Manager でワークフローをカスタマイズする場合は、このオプションを選択して、アップグレード後に最新バージョンとなるようにアップグレードする前に、カスタマイズされた各ワークフローを最新バージョンに維持します。</p> <p>このオプションを選択しない場合、vRealize Automation Designer で提供される各ワークフローはアップグレード後に最新バージョンとなり、アップグレード前の最新バージョンは 2 番目に新しいバージョンとなります。</p> <p>vRealize Automation Designer の詳細については、『ライフ サイクルの拡張性』を参照してください。</p>
Distributed Execution Manager または プロキシ エージェント をアップグレードする場合	<p>サービス アカウント セクションで管理者アカウントの認証情報を入力します。</p> <p>アップグレードするすべてのサービスがこのアカウントで実行されます。</p>
Microsoft SQL Server データベースを指定するには	<p>Model Manager Data をアップグレードする場合、Microsoft SQL Server データベース インストール情報セクションの [サーバ] テキスト ボックスにデータベース サーバとデータベース インスタンスの名前を入力します。[データベース名] にデータベース サーバ名の完全修飾ドメイン名 (FQDN) を入力します。</p> <p>データベース インスタンスがデフォルト以外の SQL ポートにある場合、サーバ インスタンス仕様にポート番号を含めます。Microsoft SQL のデフォルト ポート番号は 1433 です。</p> <p>マネージャ ノードをアップグレードする場合、MSSQL SSL オプションはデフォルトで選択されています。データベースで SSL を使用しない場合には、[データベース接続に SSL を使用] を選択解除します。</p>

- 10 [次へ] をクリックします。
 - 11 アップグレードするすべてのサービスが [アップグレードの準備完了] ページに表示されていることを確認し、[アップグレード] をクリックします。
- [アップグレード] ページおよび進行状況インジケータが表示されます。アップグレード手順を完了すると、[次へ] ボタンが有効になります。
- 12 [次へ] をクリックします。
 - 13 [完了] をクリックします。
 - 14 すべてのサービスが再起動されたことを確認します。
 - 15 導入環境内の各 IaaS サーバに対し、推奨されている順序でこの手順を繰り返します。
 - 16 すべてのコンポーネントをアップグレードしたら、アプライアンスの管理コンソールにログインし、IaaS を含むすべてのサービスが登録されていることを確認します。
 - 17 (オプション) Manager Service の自動フェイルオーバーを有効にします。『vRealize Automation のインストール』の「Manager Service 自動フェイルオーバーを有効にする」を参照してください。

選択したすべてのコンポーネントが新しいリリースにアップグレードされました。

次のステップ

- 1 [「組み込み vRealize Orchestrator コントロール センターへのアクセスのリストア」](#)。

- 2 展開環境でロード バランサを使用している場合、vRealize Automation 健全性モニターとすべてのノードへのトラフィックを再度有効にします。

詳細については、vRealize Automation のロード バランシングを参照してください。

組み込み vRealize Orchestrator コントロール センターへのアクセスのリストア

IaaS サーバ コンポーネントをアップグレードした後、vRealize Orchestrator へのアクセスをリストアする必要があります。

vRealize Automation 7.3 以前を 7.4 にアップグレードする場合、新しいロール ベースのアクセス コントロール機能に対応するには、この手順を実行する必要があります。ここには、高可用性環境での手順を記載しています。

前提条件

vRealize Automation 環境のスナップショットを作成します。

手順

- 1 アプライアンス ホストの完全修飾ドメイン名を使用して (`https://<va-hostname.domain.name>:5480`)、vRealize Automation アプライアンス 管理コンソールに root としてログインします。
- 2 [vRA 設定] - [データベース] の順に選択します。
- 3 マスター ノードとレプリカ ノードを特定します。
- 4 各レプリカ ノードで SSH セッションを開き、管理者としてログインして、次のコマンドを実行します。
service vco-server stop && service vco-configurator stop
- 5 マスター ノードで SSH セッションを開き、管理者としてログインして、次のコマンドを実行します。
rm /etc/vco/app-server/vco-registration-id
- 6 マスター ノードで、`/etc/vco/app-server/` ディレクトリに移動します。
- 7 **sso.properties** ファイルを開きます。
- 8 プロパティ名 **com.vmware.o11n.sso.admin.group.name** にスペースや、Bash コマンドで特殊文字として使用できる他の Bash 関連文字（ハイフン (-) やドル記号 (\$) など）が含まれる場合は、次の手順を実行します。
 - a **com.vmware.o11n.sso.admin.group.name** プロパティが含まれる行をコピーし、値に **AdminGroup** を入力します。
 - b **com.vmware.o11n.sso.admin.group.name** プロパティが含まれる元の行の先頭に # を追加して、この行をコメントアウトします。
 - c **sso.properties** ファイルを保存して閉じます。
- 9 次のコマンドを実行します。
vcac-vami vco-service-reconfigure

10 **sso.properties** ファイルを開きます。ファイルが変更されている場合は、次の手順を実行します。

- a **com.vmware.o11n.sso.admin.group.name** プロパティが含まれる元の行の先頭にある # を削除して、この行をコメント解除します。
- b **com.vmware.o11n.sso.admin.group.name** プロパティが表示されている行のコピーを削除します。
- c **sso.properties** ファイルを保存して閉じます。

11 次のコマンドを実行して、vco-server サービスを再起動します。

```
service vco-server restart
```

12 次のコマンドを実行して、vco-configurator サービスを再起動します。

```
service vco-configurator restart
```

13 vRealize Automation アプライアンス 管理コンソールで、[サービス] をクリックし、マスター ノードのすべてのサービスが [登録済み] になるまで待機します。

14 すべてのサービスが [登録済み] になったら、vRealize Automation レプリカ ノードを vRealize Automation クラスタに参加させ、vRealize Orchestrator 構成を同期します。詳細については、[「組み込みの vRealize Orchestrator で高可用性をサポートするための再構成」](#)を参照してください。

次のステップ

[章 6 「vRealize Automation をアップグレードした後の vRealize Orchestrator のアップグレード」](#)。

vRealize Automation をアップグレードした後の vRealize Orchestrator のアップグレード

6

vRealize Automation 7.1、7.2、または 7.3.x から 7.4 にアップデートする場合は、vRealize Orchestrator インスタンスをアップグレードする必要があります。

vRealize Orchestrator 7.4 リリースでは、vRealize Automation 7.4 にアップデートする場合に vRealize Orchestrator をアップグレードする方法が 2 つあります。

- 既存の外部 vRealize Orchestrator サーバを vRealize Automation 7.4 に組み込まれた vRealize Orchestrator に移行できます。
- 既存のスタンドアロンまたはクラスタ化された vRealize Orchestrator サーバをアップグレードして、vRealize Automation 7.4 と連携することができます。

この章には、次のトピックが含まれています。

- [外部 vRealize Orchestrator サーバから vRealize Automation への移行](#)
- [vRealize Automation で使用するスタンドアローン vRealize Orchestrator アプライアンスのアップグレード](#)
- [vRealize Automation 7.4 で使用するための vRealize Orchestrator Appliance クラスタのアップグレード](#)

外部 vRealize Orchestrator サーバから vRealize Automation への移行

既存の外部 vRealize Orchestrator サーバは、vRealize Automation 7.4 に組み込まれている vRealize Orchestrator インスタンスに移行することができます。

vRealize Orchestrator を外部サーバ インスタンスとして導入し、その外部インスタンスと連携するように vRealize Automation を構成することができます。または、vRealize Automation アプライアンスに含まれている vRealize Orchestrator サーバを構成して使用することもできます。

VMware では、外部 vRealize Orchestrator を、vRealize Automation に組み込まれた Orchestrator サーバに移行することをお勧めします。外部 Orchestrator から組み込み Orchestrator への移行には、次の利点があります。

- 総所有コストが削減されます。
- デプロイ モデルが簡素化されます。

- 運用効率が向上します。

注: 外部 vRealize Orchestrator の使用は、次の場合に考慮します。

- vRealize Automation 環境内の複数のテナント
- 物理的に分散した環境
- ワークロードの処理
- 特定のプラグイン（古いバージョンの Site Recovery Manager プラグインなど）の使用

外部 Orchestrator および組み込み Orchestrator のコントロール センターの違い

外部 vRealize Orchestrator のコントロール センターで使用可能な一部のメニュー項目は、組み込み Orchestrator インスタンスのデフォルトのコントロール センター ビューに含まれていません。

組み込み Orchestrator サーバのコントロール センターでは、いくつかのオプションがデフォルトで非表示になっています。

メニュー項目	詳細
[ライセンス]	組み込み Orchestrator では vRealize Automation をライセンス プロバイダとして使用するよう事前構成されています。
[設定をエクスポート/インポート]	エクスポートされた vRealize Automation コンポーネントに、組み込み Orchestrator の構成が含まれています。
[データベースを構成]	組み込み Orchestrator では、vRealize Automation で使用されているデータベースを使用します。
[カスタム エクスペリエンス改善プログラム]	カスタム エクスペリエンス改善プログラム (CEIP) には vRealize Automation アプライアンス管理インターフェイスから参加できます。 『vRealize Automation の管理』の「カスタム エクスペリエンス改善プログラム」を参照してください。

デフォルトのコントロール センター ビューでは非表示になっている別のオプションには、[認証プロバイダを設定] ページの [ホスト アドレス] テキスト ボックスや [登録解除] ボタンがあります。

注: vRealize Automation に組み込まれている vRealize Orchestrator でコントロール センターのオプションをすべて表示するには、Orchestrator 管理の詳細ページ (https://<vra-vahostname.domain.name_or_load_balancer_address>:8283/vco-controlcenter/#/?advanced) にアクセスし、キーボードの [F5] ボタンを押してページを更新する必要があります。

外部 vRealize Orchestrator 7.x から vRealize Automation 7.4 への移行

既存の外部 Orchestrator インスタンスから構成をエクスポートし、これを vRealize Automation に組み込まれている Orchestrator サーバにインポートすることができます。

注: 複数の vRealize Automation アプライアンス ノードがある場合は、プライマリの vRealize Automation ノードに対してのみ移行手順を実行します。

前提条件

- vRealize Automation のバージョンを 7.4 にアップグレードまたは移行します。詳細については、『vRealize Automation のインストールまたはアップグレード』の「vRealize Automation のアップグレード」を参照してください。
- 外部 Orchestrator の Orchestrator サーバ サービスを停止します。
- 外部 Orchestrator サーバのデータベースを、データベース スキーマを含めバックアップします。

手順

- 1 外部 Orchestrator サーバから構成をエクスポートします。
 - a 移行元のバージョンに応じて、**root** または**管理者**として外部 Orchestrator サーバのコントロール センターにログインします。
 - b [起動オプション] ページから Orchestrator サーバ サービスを停止して、データベースに不要な変更が加えられないようにします。
 - c [設定のエクスポート/インポート] ページに移動します。
 - d [設定をエクスポート] ページで、[サーバ設定をエクスポート]、[バンドル プラグイン]、[プラグイン設定をエクスポート] を選択します。

- 2 エクスポートした設定を組み込み Orchestrator インスタンスに移行します。
 - a エクスポートした Orchestrator 構成ファイルを vRealize Automation アプライアンスの `/usr/lib/vco/tools/configuration-cli/bin` ディレクトリにアップロードします。
 - b SSH を使用して vRealize Automation アプライアンス に **root** としてログインします。
 - c 組み込み vRealize Orchestrator サーバの Orchestrator サーバ サービスとコントロール センター サービスを停止します。

```
service vco-server stop && service vco-configurator stop
```

- d **import** コマンドを使用して **vro-configure** スクリプトを実行し、Orchestrator 構成ファイルを組み込み vRealize Orchestrator サーバにインポートします。

```
./vro-configure.sh import --type embedded --path orchestrator-config-export-  
<orchestrator_appliance_ip>--<date>_<hour>.zip
```

- 3 移行元とする外部 Orchestrator サーバが組み込み PostgreSQL データベースを使用している場合は、データベース構成ファイルを編集します。
 - a `/var/vmware/vpostgres/current/pgdata/postgresql.conf` ファイルで、**listen_addresses** 行をコメント解除します。
 - b **listen_addresses** の値をワイルドカード (*) に設定します。

```
listen_addresses = '*'
```

- c `/var/vmware/vpostgres/current/pgdata/pg_hba.conf` ファイルに行を追加します。

```
host all all <vra-va-ip-address>/32 md5
```

注: `pg_hba.conf` ファイルでは、IP アドレスとサブネット マスクの代わりに CIDR プリフィックス形式を使用する必要があります。

- d PostgreSQL サーバ サービスを再起動します。

```
service vpostgres restart
```

- 4 `db-migrate` コマンドを使用して `vro-configure` スクリプトを実行し、データベースを内部 PostgreSQL データベースに移行します。

```
./vro-configure.sh db-migrate --sourceJdbcUrl <JDBC_connection_URL> --sourceDbUsername <database_user> --sourceDbPassword <database_user_password>
```

注: 特殊文字を含むパスワードは一重引用符で囲んでください。

<JDBC_connection_URL> は、使用するデータベースのタイプによって異なります。

```
PostgreSQL: jdbc:postgresql://<host>:<port>/<database_name>
```

```
MSSQL: jdbc:jtds:sqlserver://<host>:<port>/<database_name>; if using SQL authentication and MSSQL:
jdbc:jtds:sqlserver://<host>:<port>/<database_name>;domain=<domain>;useNTLMv2=TRUE if using Windows authentication.
```

```
Oracle: jdbc:oracle:thin:@<host>:<port>:<database_name>
```

デフォルトのデータベースのログイン情報は以下のとおりです。

<database_name>	vmware
<database_user>	vmware
<database_user_password>	vmware

- 5 データベースのキーストアからすべての証明書を削除します。

```
./vro-configuration.sh untrust --reset-db
```

- 6 Orchestrator プラグインを再インストールします。

- コントロール センターに **root** としてログインします。
- [トラブルシューティング] をクリックします。
- [プラグインを強制的に再インストール] をクリックします。

7 Orchestrator サーバ サービスを開始します。

8 `postgresql.conf` および `pg_hba.conf` ファイルをデフォルトの構成に戻します。

a PostgreSQL サーバ サービスを再起動します。

外部 Orchestrator サーバ インスタンスが vRealize Automation に組み込まれている vRealize Orchestrator インスタンスに正常に移行されました。

次のステップ

組み込み vRealize Orchestrator サーバを設定します。[組み込み vRealize Orchestrator サーバの構成] を参照してください。

組み込み vRealize Orchestrator サーバの構成

外部 Orchestrator サーバの構成をエクスポートして vRealize Automation 7.4 にインポートしたら、vRealize Automation に組み込まれた Orchestrator サーバを構成する必要があります。

前提条件

設定を外部から内部 vRealize Orchestrator に移行します。

手順

- 1 SSH を使用して vRealize Automation アプライアンス に **root** としてログインします。
- 2 コントロール センター サービスと、組み込み vRealize Orchestrator サーバの Orchestrator サーバ サービスを開始します。

```
service vco-configurator start && service vco-server start
```

- 3 組み込み Orchestrator サーバのコントロール センターに**管理者**としてログインします。

注: 外部 vRealize Orchestrator 7.4 インスタンスから移行する場合は、手順 5 に進んでください。

- 4 コントロール センターの [設定を検証] ページで、Orchestrator が正しく設定されていることを確認します。
- 5 外部 Orchestrator がクラスタ モードで動作するように構成されている場合は、vRealize Automation で Orchestrator クラスタを再構成します。

a 詳細な [Orchestrator クラスタ管理] ページ (https://<vra-vahostname.domain.name_or_load_balancer_address>:8283/vco-controlcenter/#/control-app/ha?remove-nodes) に移動します。

注: クラスタの既存のノードの横に [削除] チェック ボックスが表示されない場合は、キーボードの [F5] ボタンをクリックして、ブラウザ ページを更新する必要があります。

b 外部 Orchestrator ノードの横にあるチェック ボックスをオンにし、[削除] をクリックしてクラスタから削除します。

- c 詳細なクラスタ管理ページを終了するには、URL から **remove-nodes** 文字列を削除し、キーボードの [F5] ボタンをクリックしてブラウザ ページを更新します。
- d コントロール センターの [設定を検証] ページで、Orchestrator が適切に設定されていることを確認します。
- 6 (オプション) [証明書] ページの [パッケージ署名証明書] タブの下で、新しいパッケージ署名証明書を生成します。
- 7 (オプション) [認証プロバイダを設定] ページの [デフォルト テナント] と [管理グループ] の値を変更します。
- 8 vRealize Automation アプライアンス 管理コンソールの [サービス] タブの下で **vco-server** サービスが REGISTERED と表示されていることを確認します。
- 9 外部 Orchestrator サーバの **vco** サービスを選択し、[登録解除] をクリックします。

次のステップ

- 外部 Orchestrator サーバにある信頼された証明書を、組み込み Orchestrator のトラスト ストアにインポートします。
- vRealize Automation レプリカ ノードを vRealize Automation クラスタに参加させて、Orchestrator 構成を同期します。

詳細については、『vRealize Automation のインストールまたはアップグレード』の「Reconfigure the Target Embedded vRealize Orchestrator to Support High Availability」を参照してください。

注: vRealize Orchestrator のインスタンスは自動的にクラスタ化されており、使用可能です。

- クラスタ内のすべてのノードで **vco-configurator** サービスを再起動します。
- 移行した組み込み Orchestrator サーバをポイントするように、vRealize Orchestrator エンドポイントを更新します。
- vRealize Automation ホストと IaaS ホストを vRealize Automation プラグインのインベントリに追加します。これは vRA ホスト ワークフローの vRA ホストの追加と IaaS ホストの追加を実行して行います。

vRealize Automation で使用するスタンドアローン vRealize Orchestrator アプライアンスのアップグレード

vRealize Automation で使用する vRealize Orchestrator のスタンドアローンの外部インスタンスを維持している場合は、vRealize Automation を 7.1、7.2、または 7.3.x から 7.4 にアップデートするときに vRealize Orchestrator をアップグレードする必要があります。

vRealize Orchestrator の組み込みインスタンスは、vRealize Automation アプライアンスのアップグレードの一部としてアップグレードされます。組み込みインスタンスの場合、追加のアクションは必要ありません。

vRealize Orchestrator アプライアンス クラスタをアップグレードする場合は、[「vRealize Automation 7.4 で使用するための vRealize Orchestrator Appliance クラスタのアップグレード」](#)を参照してください。

前提条件

- [「vRealize Automation アプライアンスと IaaS コンポーネントへのアップデートのインストール」](#)。

- すべてのネットワーク ファイル システムをマウント解除します。vSphere のドキュメントの『vSphere 仮想マシン管理』を参照してください。
- vSphere Orchestrator Appliance のメモリを 6 GB 以上に増やします。vSphere のドキュメントの『vSphere 仮想マシン管理』を参照してください。
- vSphere Orchestrator 仮想マシンのスナップショットを作成します。vSphere のドキュメントの『vSphere 仮想マシン管理』を参照してください。
- 外部データベースを使用する場合は、データベースをバックアップします。
- vSphere Orchestrator の事前構成された PostgreSQL データベースを使用する場合は、vSphere コントロールセンターの [データベースをエクスポート] メニューを使用して、データベースをバックアップします。

手順

- ◆ 次のいずれかの方法を使用してスタンドアロン vRealize Orchestrator をアップグレードします。
 - 「デフォルトの VMware リポジトリを使用した Orchestrator Appliance のアップグレード」。
 - 「ISO イメージを使用した Orchestrator Appliance のアップグレード」。
 - 「指定したリポジトリを使用した Orchestrator Appliance のアップグレード」。

デフォルトの VMware リポジトリを使用した Orchestrator Appliance のアップグレード

Orchestrator を構成して、デフォルトの VMware リポジトリからアップグレード パッケージをダウンロードできます。

前提条件

- すべてのネットワーク ファイル システムをマウント解除します。詳細については、『vSphere 仮想マシン管理』ドキュメントを参照してください。
- Orchestrator Appliance のメモリを 6 GB 以上増やします。詳細については、『vSphere 仮想マシン管理』ドキュメントを参照してください。
- vRealize Orchestrator 仮想マシンのディスク サイズを、Disk1=7 GB、Disk2=10 GB に増やします。
- Orchestrator Appliance の root パーティションに 3 GB 以上の使用可能な空き容量があることを確認してください。ディスク パーティションのサイズを増やす方法については、KB1004071 (<http://kb.vmware.com/kb/1004071>) を参照してください。
- Orchestrator 仮想マシンのスナップショットを作成します。詳細については、『vSphere 仮想マシン管理』ドキュメントを参照してください。
- 外部データベースを使用する場合は、データベースをバックアップします。
- 事前構成された Orchestrator PostgreSQL データベースを使用する場合は、コントロールセンターの [データベースをエクスポート] メニューを使用して、データベースをバックアップします。

手順

- 1 仮想アプライアンス管理インターフェイス (VAMI : https://<orchestrator_server>:5480) にアクセスし、**root** としてログインします。
- 2 [更新] タブで、[設定] をクリックします。
[デフォルト リポジトリの使用] オプションの横にあるラジオ ボタンが選択されます。
- 3 [ステータス] ページで、[更新チェック] をクリックします。
- 4 アップデートが利用可能な場合は、[アップデートのインストール] をクリックします。
- 5 VMware エンドユーザー使用許諾契約に同意し、アップデートをインストールすることを確認します。
- 6 アップデートを完了するには、Orchestrator Appliance を再起動します。
 - a 仮想アプライアンス管理インターフェイス (VAMI) に **root** として再度ログインします。
- 7 (オプション) [更新] タブで、最新バージョンの Orchestrator Appliance が正常にインストールされていることを確認します。
- 8 コントロール センターに **root** としてログインします。
- 9 Orchestrator インスタンスのクラスタを作成する予定の場合は、ホスト設定を再構成します。
 - a コントロール センターの [ホスト設定] ページで [変更] をクリックします。
 - b vRealize Orchestrator のアプライアンス名ではなく、ロード バランサ サーバのホスト名を入力します。
- 10 認証を再構成します。
 - a アップグレードする前に、Orchestrator サーバが [LDAP] または [SSO (レガシー)] を認証方式として使用するように構成されていた場合は、[vSphere] または [vRealize Automation] を認証プロバイダとして構成します。
 - b 認証がすでに [vSphere] または [vRealize Automation] に設定されている場合は、設定を登録解除し、それらを再度登録します。

注: アップグレード前に、Orchestrator が認証プロバイダとして [vSphere] を使用し、vCenter Server の完全修飾ドメイン名または IP アドレスに接続するように構成されており、外部に Platform Services Controller がある場合、アップグレード後に、Orchestrator を vCenter Single Sign-On を含む Platform Services Controller インスタンスの完全修飾ドメイン名または IP アドレスに接続するように構成する必要があります。同一の vCenter Single Sign-On ドメインを共有するすべての Platform Services Controller の証明書を手動で Orchestrator にインポートする必要もあります。

これで、Orchestrator Appliance が正常にアップグレードされました。

次のステップ

コントロール センターの [設定を検証] ページで、Orchestrator が正しく設定されていることを確認します。

ISO イメージを使用した Orchestrator Appliance のアップグレード

Orchestrator を構成することにより、アプライアンスの CD-ROM ドライブにマウントされている ISO イメージ ファイルからアップグレード パッケージをダウンロードできます。

前提条件

- すべてのネットワーク ファイル システムをマウント解除します。詳細については、『vSphere 仮想マシン管理』ドキュメントを参照してください。
- Orchestrator Appliance のメモリを 6 GB 以上増やします。詳細については、『vSphere 仮想マシン管理』ドキュメントを参照してください。
- vRealize Orchestrator 仮想マシンのディスク サイズを、Disk1=7 GB、Disk2=10 GB に増やします。
- Orchestrator Appliance の root パーティションに 3 GB 以上の使用可能な空き容量があることを確認してください。ディスク パーティションのサイズを増やす方法については、KB1004071 (<http://kb.vmware.com/kb/1004071>) を参照してください。
- Orchestrator 仮想マシンのスナップショットを作成します。詳細については、『vSphere 仮想マシン管理』ドキュメントを参照してください。
- 外部データベースを使用する場合は、データベースをバックアップします。
- 事前構成された Orchestrator PostgreSQL データベースを使用する場合は、コントロール センターの[データベースをエクスポート]メニューを使用して、データベースをバックアップします。

手順

- 1 VMware の公式なダウンロード サイトで **VMware-vRO-Appliance-<version>-<build_number>-updaterepo.iso** アーカイブをダウンロードします。
- 2 Orchestrator Appliance 仮想マシンの CD-ROM ドライブを接続します。詳細については、『vSphere 仮想マシン管理』ドキュメントを参照してください。
- 3 ISO イメージ ファイルをアプライアンスの CD-ROM ドライブにマウントします。詳細については、『vSphere 仮想マシン管理』ドキュメントを参照してください。
- 4 仮想アプライアンス管理インターフェイス (VAMI : https://<orchestrator_server>:5480) にアクセスし、**root** としてログインします。
- 5 [更新] タブで、[設定] をクリックします。
- 6 [CD-ROM アップデートの使用] オプションの横にあるラジオ ボタンを選択します。
- 7 [ステータス] ページに戻ります。
使用可能なアップグレードのバージョンが表示されます。
- 8 [アップデートをインストール] をクリックします。
- 9 VMware エンドユーザー使用許諾契約に同意し、アップデートをインストールすることを確認します。

- 10 アップデートを完了するには、Orchestrator Appliance を再起動します。
 - a 仮想アプライアンス管理インターフェイス (VAMI) に **root** として再度ログインします。
- 11 (オプション) [更新] タブで、最新バージョンの Orchestrator Appliance が正常にインストールされていることを確認します。
- 12 コントロール センターに **root** としてログインします。
- 13 Orchestrator インスタンスのクラスタを作成する予定の場合は、ホスト設定を再構成します。
 - a コントロール センターの [ホスト設定] ページで [変更] をクリックします。
 - b vRealize Orchestrator のアプライアンス名ではなく、ロード バランサ サーバのホスト名を入力します。
- 14 認証を再構成します。
 - a アップグレードする前に、Orchestrator サーバが [LDAP] または [SSO (レガシー)] を認証方式として使用するように構成されていた場合は、[vSphere] または [vRealize Automation] を認証プロバイダとして構成します。
 - b 認証がすでに [vSphere] または [vRealize Automation] に設定されている場合は、設定を登録解除し、それらを再度登録します。

注: アップグレード前に、Orchestrator が認証プロバイダとして [vSphere] を使用し、vCenter Server の完全修飾ドメイン名または IP アドレスに接続するように構成されており、外部に Platform Services Controller がある場合、アップグレード後に、Orchestrator を vCenter Single Sign-On を含む Platform Services Controller インスタンスの完全修飾ドメイン名または IP アドレスに接続するように構成する必要があります。同一の vCenter Single Sign-On ドメインを共有するすべての Platform Services Controller の証明書を手動で Orchestrator にインポートする必要があります。

これで、Orchestrator Appliance が正常にアップグレードされました。

次のステップ

コントロール センターの [設定を検証] ページで、Orchestrator が正しく設定されていることを確認します。

指定したリポジトリを使用した Orchestrator Appliance のアップグレード

Orchestrator を構成して、アップグレード アーカイブのアップロード先であるローカル リポジトリを使用できます。

前提条件

- すべてのネットワーク ファイル システムをマウント解除します。詳細については、『vSphere 仮想マシン管理』ドキュメントを参照してください。
- Orchestrator Appliance のメモリを 6 GB 以上増やします。詳細については、『vSphere 仮想マシン管理』ドキュメントを参照してください。
- vRealize Orchestrator 仮想マシンのディスク サイズを、Disk1=7 GB、Disk2=10 GB に増やします。

- Orchestrator Appliance の root パーティションに 3 GB 以上の使用可能な空き容量があることを確認してください。ディスク パーティションのサイズを増やす方法については、KB1004071 (<http://kb.vmware.com/kb/1004071>) を参照してください。
- Orchestrator 仮想マシンのスナップショットを作成します。詳細については、『vSphere 仮想マシン管理』ドキュメントを参照してください。
- 外部データベースを使用する場合は、データベースをバックアップします。
- 事前構成された Orchestrator PostgreSQL データベースを使用する場合は、コントロール センターの[データベースをエクスポート]メニューを使用して、データベースをバックアップします。

手順

- 1 アップグレード用のローカル リポジトリを準備します。
 - a ローカル Web サーバをインストールして構成します。
 - b VMware の公式なダウンロード サイトで **VMware-vR0-Appliance-<version>-<build_number>-updaterepo.zip** アーカイブをダウンロードします。
 - c **.ZIP** アーカイブをローカル リポジトリに抽出します。
- 2 仮想アプライアンス管理インターフェイス (VAMI : https://<orchestrator_server>:5480) にアクセスし、**root** としてログインします。
- 3 [更新] タブで、[設定] をクリックします。
- 4 [指定したリポジトリを使用] オプションの横にあるラジオ ボタンを選択します。
- 5 ローカル リポジトリの URL アドレスを **Update_Repo** ディレクトリをポイントして入力します。
http://<local_web_server>:<port>/build/mts/release/bora-<build_number>/publish/exports/Update_Repo
- 6 ローカル リポジトリで認証が必要になる場合は、ユーザー名とパスワードを入力します。
- 7 [設定の保存] をクリックします。
- 8 [ステータス] ページで、[更新チェック] をクリックします。
- 9 アップデートが利用可能な場合は、[アップデートのインストール] をクリックします。
- 10 VMware エンドユーザー使用許諾契約に同意し、アップデートをインストールすることを確認します。
- 11 アップデートを完了するには、Orchestrator Appliance を再起動します。
 - a 仮想アプライアンス管理インターフェイス (VAMI) に **root** として再度ログインします。
- 12 (オプション) [更新] タブで、最新バージョンの Orchestrator Appliance が正常にインストールされていることを確認します。
- 13 コントロール センターに **root** としてログインします。

14 Orchestrator インスタンスのクラスタを作成する予定の場合は、ホスト設定を再構成します。

- a コントロール センターの [ホスト設定] ページで [変更] をクリックします。
- b vRealize Orchestrator のアプライアンス名ではなく、ロード バランサ サーバのホスト名を入力します。

15 認証を再構成します。

- a アップグレードする前に、Orchestrator サーバが [LDAP] または [SSO (レガシー)] を認証方式として使用するように構成されていた場合は、[vSphere] または [vRealize Automation] を認証プロバイダとして構成します。
- b 認証がすでに [vSphere] または [vRealize Automation] に設定されている場合は、設定を登録解除し、それらを再度登録します。

注: アップグレード前に、Orchestrator が認証プロバイダとして [vSphere] を使用し、vCenter Server の完全修飾ドメイン名または IP アドレスに接続するように構成されており、外部に Platform Services Controller がある場合、アップグレード後に、Orchestrator を vCenter Single Sign-On を含む Platform Services Controller インスタンスの完全修飾ドメイン名または IP アドレスに接続するように構成する必要があります。同一の vCenter Single Sign-On ドメインを共有するすべての Platform Services Controller の証明書を手動で Orchestrator にインポートする必要もあります。

これで、Orchestrator Appliance が正常にアップグレードされました。

次のステップ

コントロール センターの [設定を検証] ページで、Orchestrator が正しく設定されていることを確認します。

vRealize Automation 7.4 で使用するための vRealize Orchestrator Appliance クラスタのアップグレード

vRealize Automation で vRealize Orchestrator Appliance クラスタを使用する場合は、いずれか 1 つのインスタンスをアップグレードした後に、新しくインストールした 7.4 ノードをアップグレード済みのインスタンスに参加させることで Orchestrator Appliance クラスタをバージョン 7.4 にアップグレードする必要があります。

vRealize Orchestrator のインスタンスを 1 つのみアップグレードする場合は、[\[vRealize Automation で使用するスタンドアローン vRealize Orchestrator アプライアンスのアップグレード\]](#) を参照してください。

前提条件

- [\[vRealize Automation アプライアンスと IaaS コンポーネントへのアップデートのインストール\]](#)。
- vRealize Orchestrator の複数のインスタンス間でトラフィックを分散するロード バランサを設定します。[vRealize Orchestrator ロード バランシング構成ガイド](#)を参照してください。
- すべての vRealize Orchestrator サーバ ノードのスナップショットを作成します。
- vRealize Orchestrator の共有データベースをバックアップします。

手順

- 1 すべてのクラスタ ノードで **vco-server** および **vco-configurator** Orchestrator サービスを停止します。

- 2 ドキュメントで説明されている手順のいずれかを使用して、クラスタ内の Orchestrator サーバ インスタンスのいずれか 1 つのみをアップグレードします。
- 3 バージョン 7.3 で新しい Orchestrator Appliance を展開します。
 - a クラスタに含まれているアップグレード済みインスタンスではなく、既存のネットワーク設定を使用して新しいノードを構成します。
- 4 コントロール センターの 2 番目のノードにアクセスして、構成ウィザードを開始します。
 - a `https://<your_orchestrator_server_ip_or_dns_name>:8283/vco-controlcenter` に移動します。
 - b OVA の展開時に入力したパスワードを使用して **root** としてログインします。

- 5 [クラスタ化された Orchestrator] デプロイ タイプを選択します。

このタイプを選択することで、ノードが既存の Orchestrator クラスタに参加することを指定できます。

- 6 [ホスト名] テキスト ボックスに、最初の Orchestrator サーバ インスタンスのホスト名または IP アドレスを入力します。

注: これは、2 番目のノードを参加させている Orchestrator インスタンスのローカル IP アドレスまたはホスト名にする必要があります。ロード バランサのアドレスを使用することはできません。

- 7 [ユーザー名] および [パスワード] テキスト ボックスに、最初の Orchestrator サーバ インスタンスの root 認証情報を入力します。
- 8 [参加] をクリックします。Orchestrator インスタンスが参加先のノードの構成をクローン作成します。

両方のノードの Orchestrator サーバ サービスが自動的に再起動されます。
- 9 ロード バランサのアドレスおよび**管理者**としてのログインを介してアップグレードされた Orchestrator クラスタのコントロール センターにアクセスします。
- 10 [Orchestrator クラスタ管理] ページで、[アクティブな設定フィンガー プリント] の文字列と [保留中の設定フィンガー プリント] の文字列が、クラスタのすべてのノードで一致することを確認します。

注: 2 つの文字列が一致するまでページを数回更新する必要があることがあります。

- 11 コントロール センターの [設定を検証] ページを開いて、vRealize Orchestrator クラスタが適切に構成されていることを確認します。
- 12 (オプション) クラスタ内の追加ノードごとに手順 3 から手順 8 を繰り返します。

Orchestrator クラスタが正常にアップグレードされました。

次のステップ

章 7 「ロード バランサの有効化」。

ロード バランサの有効化

環境内でロード バランサを使用する場合は、セカンダリ ノードと健全性チェックを再度有効にして、ロード バランサのタイムアウト設定を元に戻します。

vRealize Automation の健全性チェックは、バージョンによって異なります。詳細については、vRealize Automation ドキュメントの『vRealize Automation Load Balancing Configuration Guide』を参照してください。

ロード バランサのタイムアウト設定を変更して、10 分からデフォルトに戻します。

vRealize Automation のアップグレード後のタスク

8

vRealize Automation 7.1、7.2、7.3.x から 7.4 へのアップデート後、アップグレード後のタスクを実行する必要があります。

この章には、次のトピックが含まれています。

- ソフトウェア エージェントの TLS 1.2 へのアップグレード
- vRealize Automation PostgreSQL レプリケーション モードの同期設定
- 接続テストの実行とアップグレード後のエンドポイントの確認
- vRealize Automation からのアップグレード後の NSX ネットワークおよびセキュリティ インベントリ データ収集の実行
- レプリカ アプライアンスのクラスタへの参加
- 高可用性を展開する環境でのポート構成
- 組み込みの vRealize Orchestrator で高可用性をサポートするための再構成
- 外部ワークフローのタイムアウト ファイルのリストア
- ユーザー用リモート コンソール アクションとの接続の有効化
- app.config ファイルに行ったログの変更のリストア
- アップグレード後に Manager Service の自動フェイルオーバーを有効にする

ソフトウェア エージェントの TLS 1.2 へのアップグレード

vRealize Automation 7.4 へのアップグレード後、ソフトウェア エージェントを vRealize Automation 7.1、7.2、7.3、または 7.3.1 環境から TLS 1.2 にアップグレードするには、いくつかのタスクを実行する必要があります。

vRealize Automation 7.4 以降、トランスポート レイヤー セキュリティ (TLS) 1.2 は、vRealize Automation とブラウザの間のデータ通信で唯一サポートされる TLS プロトコルです。

移行後、既存のすべての仮想マシンと、vRealize Automation 7.1、7.2、7.3、または 7.3.1 環境からの既存の仮想マシン テンプレートをアップグレードする必要があります。

vRealize Automation 仮想マシン テンプレートの更新

ソフトウェア エージェントが TLS 1.2 プロトコルを使用できるように vRealize Automation 7.4 への更新が完了した後、既存のテンプレートを更新する必要があります。

ゲスト エージェントおよびエージェント ブートストラップ コードを、vRealize Automation 7.1、7.2、7.3 または 7.3.1 のテンプレートで更新する必要があります。リンク クローン オプションを使用している場合、新規作成した仮想マシンおよびそれらのスナップショットを使用したテンプレートの再マッピングが必要になることがあります。

テンプレートをアップグレードするには、次のタスクを実行します。

- 1 vSphere にログインします。
- 2 vRealize Automation 7.1、7.2、7.3 または 7.3.1 の各テンプレートを仮想マシンに変換し、そのマシンをオンにします。
- 3 適切なソフトウェアのインストーラをインポートし、各仮想マシンでソフトウェアのインストーラを実行します。
- 4 各仮想マシンを再度テンプレートに変換します。

この手順を使用すると、Linux または Windows 用のソフトウェア インストーラを特定できます。

前提条件

vRealize Automation 7.4 への正常なアップグレード

手順

- 1 ブラウザを起動し、仮想アプライアンスの完全修飾ドメイン名 (<https://<vra-virtual-machine-hostname.domain.name>>) を使用して、vRealize Automation 7.4 アプライアンスのスプラッシュ ページを開きます。
- 2 [ゲストおよびソフトウェア エージェント] ページをクリックします。
- 3 Linux または Windows ソフトウェアのインストーラの手順を実行してください。

次のステップ

[「ソフトウェア エージェントのアップグレードが必要な仮想マシンの特定」](#)。

ソフトウェア エージェントのアップグレードが必要な仮想マシンの特定

vRealize Automation の健全性サービスを使用して、TLS 1.2 へのソフトウェア エージェントの更新が必要な仮想マシンを特定できます。

健全性サービスを使用して、TLS 1.2 へのソフトウェア エージェントの更新が必要な仮想マシンを特定できます。ブラウザと vRealize Automation の間の安全な通信を必要とするプロビジョニング後の手順を実行できるように、vRealize Automation 7.4 環境のすべてのソフトウェア エージェントを更新する必要があります。

前提条件

- vRealize Automation 7.4 に正常にアップグレードしている。
- テナント管理者としてプライマリ仮想アプライアンスで vRealize Automation 7.4 にログインしている。

手順

- 1 [管理] - [健全性] の順にクリックします。
- 2 [新しい構成] をクリックします。

3 [構成の詳細] ページで、必要な情報を提供します。

オプション	コメント
名前	SW Agent verification と入力します。
説明	Locate software agents for upgrade to TLS 1.2 などの、オプションの説明を追加します。
製品	vRealize Automation 7.4.0 を選択します。
スケジュール	[なし] を選択します。

4 [次へ] をクリックします。

5 [テストスイートの選択] ページで、[vRealize Automation のシステム テスト] と [vRealize Automation のテナント テスト] を選択します。

6 [次へ] をクリックします。

7 [パラメータの構成] ページで、必要な情報を提供します。

表 8-1. vRealize Automation 仮想アプライアンス

オプション	説明
公開 Web サーバのアドレス	<ul style="list-style-type: none"> ■ 最小インストールの場合、vRealize Automation アプライアンス ホストのベース URL。例：https://<va-host.domain>/ ■ 高可用性展開の場合、vRealize Automation ロード バランサのベース URL。例：https://<load-balancer-host.domain>/
SSH コンソールのアドレス	vRealize Automation アプライアンスの完全修飾ドメイン名。例：<va-host.domain>
SSH コンソール ユーザー	root
SSH コンソール パスワード	root のパスワード。
サービスの応答の最大時間（ミリ秒）	デフォルトを受け入れ：2000

表 8-2. vRealize Automation システム テナント

オプション	説明
システム テナント管理者	管理者
システム テナント パスワード	管理者のパスワード。

表 8-3. vRealize Automation ディスク容量の監視

オプション	説明
警告レベルしきい値の割合	デフォルトを受け入れ：75
重大レベルしきい値の割合	デフォルトを受け入れ：90

表 8-4. vRealize Automation テナント

オプション	説明
テスト対象テナント	テスト対象として選択されたテナント。
ファブリック管理者のユーザー名	ファブリック管理者のユーザー名。たとえば、admin@va-host.local。 <small>注: すべてのテストを実行するために、このファブリック管理者にはテナント管理者と laaS 管理者のロールも必要です。</small>
ファブリック管理者パスワード	ファブリック管理者のパスワード。

- 8 [次へ] をクリックします。
- 9 [サマリ] ページで情報を確認し、[完了] をクリックします。
ソフトウェア エージェントの検証設定が完了しました。
- 10 SW Agent verification カードで、[実行] をクリックします。
- 11 テストが完了したら、SW Agent verification カードの中央をクリックします。
- 12 SW Agent verification の結果ページで、テスト結果を参照して、[名前] 列の [ソフトウェア エージェントのバージョン チェック] テストを見つけます。テスト結果が失敗の場合は、[原因] 列の [原因] リンクをクリックして、ソフトウェア エージェントが古い仮想マシンを表示します。

次のステップ

ソフトウェア エージェントが古い仮想マシンがある場合は、[「vSphere 上のソフトウェア エージェントのアップグレード」](#) を参照してください。

vSphere 上のソフトウェア エージェントのアップグレード

vRealize Automation アプライアンス管理を使用して、アップグレード後に vSphere の古いソフトウェア エージェントを TLS 1.2 にアップグレードできます。

この手順は、アップグレードされた環境内の仮想マシンで、古いソフトウェア エージェントを TLS 1.2 にアップグレードします。vRealize Automation 7.4 へのアップグレードで必要になります。

前提条件

- vRealize Automation 7.4 への正常なアップグレード
- 健全性サービスを使用して、古いソフトウェア エージェントを持つ仮想アプライアンスを識別している。

手順

- 1 プライマリ vRealize Automation アプライアンスで、vRealize Automation アプライアンスを展開したときに入力したパスワードを使用して、**root** として vRealize Automation アプライアンス管理にログインします。
高可用性環境の場合、マスター アプライアンスでアプライアンス管理を開きます。
- 2 [vRA 設定] - [ソフトウェア エージェント] の順にクリックします。

- 3 [Toggle TLS 1.0、1.1] をクリックします。

[TLS v1.0, v1.1 Status] が有効になります。

- 4 テナント認証情報には、vRealize Automation 7.4 アプライアンスに要求される情報を入力します。

オプション	説明
テナント名	アップグレードされた vRealize Automation アプライアンス上のテナントの名前。 <small>注: テナントユーザーにはソフトウェアアーキテクトロールが割り当てられている必要があります。</small>
ユーザー名	vRealize Automation アプライアンス上のテナント管理者のユーザー名。
パスワード	テナント管理者のパスワード。

- 5 [接続をテスト] をクリックします。

接続が確立されると、成功のメッセージが表示されます。

- 6 [バッチの一覧表示] をクリックします。

バッチ選択肢リストテーブルが表示されます。

- 7 [表示] をクリックします。

古いソフトウェアエージェントを持つ仮想マシンのリストがテーブルに表示されます。

- 8 アップグレード可能状態の仮想マシンのソフトウェアエージェントをアップグレードします。

- 個々の仮想マシンのソフトウェアエージェントをアップグレードするには、仮想マシンのグループの [表示] をクリックして、アップグレードする仮想マシンを識別します。それから [実行] をクリックし、アップグレードプロセスを開始します。
- 仮想マシンのバッチのソフトウェアエージェントをアップグレードするには、アップグレードするグループを識別し、[実行] をクリックしてアップグレードプロセスを開始します。

アップグレードする仮想マシンが 200 台以上ある場合は、次のパラメータの値を入力して、バッチアップグレードプロセスの速度を制御できます。

オプション	説明
バッチ サイズ	バッチアップグレードで選択した仮想マシンの数。この数を変えると、アップグレードの速度を調整することができます。
キュー深度	同時に実行されるアップグレードの平行実行の数。たとえば、20 と指定します。この数を変えると、アップグレードの速度を調整することができます。
バッチ エラー	バッチアップグレードの低速化の原因となった REST エラーの数。たとえば、アップグレードの安定性を向上するために 5 回エラーが発生したときに現在のバッチアップグレードを停止する場合は、テキストフィールドに 5 と入力します。

オプション	説明
バッチの失敗	バッチ アップグレードの低速化の原因となった、失敗したソフトウェア エージェント アップグレードの数。たとえば、アップグレードの安定性を向上するために 5 回エラーが発生したときに現在のバッチ アップグレードを停止する場合は、テキスト フィールドに 5 と入力します。
バッチ ポーリング	アップグレード プロセスをチェックするためにアップグレード プロセスをポーリングする頻度。この数を変えると、アップグレードの速度を調整することができます。

アップグレード プロセスが低速すぎるか、失敗したアップグレードが多すぎる場合は、これらのパラメータを調整してアップグレードのパフォーマンスを向上できます。

注: [更新] をクリックすると、バッチのリストがクリアされます。アップグレード プロセスには影響しません。TLS 1.2 が設定されるかどうかに関する情報も更新されます。また、[更新] をクリックすると vRealize Automation サービスの健全性チェックも実行されます。サービスが実行されていない場合、エラー メッセージが表示され、他のすべてのアクション ボタンが無効になります。

9 [Toggle TLS 1.0、1.1] をクリックします。

[TLS v1.0, v1.1 Status] が無効になります。

Amazon Web Service または Azure 上のソフトウェア エージェントのアップグレード

Amazon Web Service (AWS) または Azure 上の仮想マシンの古いソフトウェア エージェントは手動でアップグレードすることができます。

前提条件

- vRealize Automation 7.4 への正常なアップグレード
- ソフトウェア トンネルが存在し、トンネル仮想マシンの IP アドレスが既知である。

手順

1 アップグレードする必要がある各ノードのノード ファイルを作成します。

```
/usr/lib/vcac/server/webapps/ROOT/software/initializeUpdateSoftwareAgents.py -a <
$DestinationVRAServer> -t <$Tenant> -tu <$TenantUser> -S <$SourceVRAServer>
```

注: インブレース アップグレードでは、\$DestinationVRAServer は \$SourceVRAServer と同一です。

2 Linux または Windows 仮想マシン上のソフトウェア エージェントをアップグレードするプラン ファイルを作成します。

- /var/log/vcac/agentupdate/{tenant}/{subtenant-UUID} にある migrate params ファイルを変更し、AWS または Azure エンドポイントに対応するプライベート IP アドレスの値を含めます。

```
"key": "ipAddress",
  "value": {
    "type": "string",
    "value": "<$PrivateIp:$PrivatePort>"
  }
}
```

- Linux マシンをアップデートする場合は、このコマンドを使用します。

```
/usr/lib/vcac/server/webapps/ROOT/software/updateSoftwareAgents.py -a <
$DestinationVRAServer> -t <$Tenant> -S <$SourceVRAServer> -tu <$TenantUser> -CL
Software.LinuxAgentUpdate74 --source_cloud_provider azure
```

- Windows マシンをアップデートする場合は、このコマンドを使用します。

```
/usr/lib/vcac/server/webapps/ROOT/software/updateSoftwareAgents.py -a <
$DestinationVRAServer> -t <$Tenant> -S <$SourceVRAServer> -tu <$TenantUser> -CW
Software.WindowsAgentUpdate74 --source_cloud_provider azure
```

- このコマンドは、プラン ファイルを実行します。

```
/usr/lib/vcac/server/webapps/ROOT/software/updateSoftwareAgents.py -a <
$DestinationVRAServer> -t <$Tenant> -tu <$TenantUser> --
plan_file /usr/lib/vcac/server/webapps/ROOT/software/plan
```


- 3 手順 1 のノード ファイルと手順 2 のプラン ファイルを使用してソフトウェア エージェントをアップデートするには、このコマンドを使用します。

```
/usr/lib/vcac/server/webapps/ROOT/software/updateSoftwareAgents.py -a <
$DestinationVRAServer> -t <$tenant> -tu <$TenantUser> --component_windows
Software.WindowsAgentUpdate74 --component_linux Software.LinuxAgentUpdate74 --
plan_file /usr/lib/vcac/server/webapps/ROOT/software/plan --plan_index 0 --
node_file /usr/lib/vcac/server/webapps/ROOT/software/node --source_cloud_provider azure
--action plan_batch -S <$SourceVRAServer>
```

代わりに、ノードのインデックスを指定することでノード ファイルから一度に 1 台のノードを実行するには、このコマンドを使用します。

```
/usr/lib/vcac/server/webapps/ROOT/software/updateSoftwareAgents.py -a <
$DestinationVRAServer> -t <$tenant> -tu <$TenantUser> --component_windows
Software.WindowsAgentUpdate74 --component_linux Software.LinuxAgentUpdate74 --
plan_file /usr/lib/vcac/server/webapps/ROOT/software/plan --plan_index 0 --
node_file /usr/lib/vcac/server/webapps/ROOT/software/node --source_cloud_provider azure
--action execute_node -S <$SourceVRAServer> --node_index <0 through n-1>
```

この手順を実行するとき、vRealize Automation 仮想アプライアンスおよびホスト マシンからログの末尾を監視してサーバ エージェントのアップグレード プロセスを表示することができます。

アップグレード後、アップグレード プロセスにより、Windows または Linux 用のソフトウェア アップデート スクリプトは vRealize Automation 7.4 仮想アプライアンスにインポートされます。vRealize Automation 仮想アプライアンス ホストにログインすると、ソフトウェア コンポーネントが正常にインポートされていることを確認できます。コンポーネントをインポートした後、ソフトウェア アップデートは古いイベント ブローカ サービス (EBS) に送信され、指定された仮想マシンにソフトウェア アップデート スクリプトが中継されます。アップグレードが完了し、新しいソフトウェア エージェントが稼動すると、ping 要求を送信して新しい vRealize Automation 仮想アプライアンスにバインドします。

注: 有用なログ ファイル

- ソース vRealize Automation の Catalina 出力 : /var/log/vcac/catalina.out。このファイルでは、エージェントの移行が行われるときに行われるアップグレード要求を確認できます。このアクティビティは、ソフトウェア プロビジョニング要求を実行する場合と同様です。
- ターゲット vRealize Automation の Catalina 出力 : /var/log/vcac/catalina.out。このファイルでは、ping 要求をレポートする移行された仮想マシンが確認できます。バージョン番号 7.4.0-SNAPSHOT が含まれます。これらは、sw-agent-UUID などの EBS トピック名を比較することで照合できます。
- ターゲット vRealize Automation マシン マスター アップグレード ログ ファイルのエージェント アップデート フォルダ : var/log/vmware/vcac/agentupdate/updateSoftwareAgents.log。このファイルの末尾を監視すると、どのアップグレード処理が進行中なのかを確認できます。
- テナント フォルダ下で利用可能な個々のログ : /var/log/vcac/agentupdate/{tenant}/{subtenant-UUID}。個々のノードがエラーや進行中の拡張機能がある多数のファイルとしてここに表示されます。

- 移行された仮想マシン : /opt/vmware-appdirector/agent/logs/darwin*.log。この場所をスポットチェックすると、受信されているソフトウェア アップデート要求、および agent_bootstrap + ソフトウェア エージェントの最後に発生する再起動のリストを確認できます。

vRealize Automation PostgreSQL レプリケーションモードの同期設定

アップグレード前に PostgreSQL レプリケーション モードを非同期に設定していても、分散 vRealize Automation 環境のアップグレード後に、PostgreSQL レプリケーション モードを同期に設定することができます。

前提条件

- 分散 vRealize Automation 環境をアップグレードしています。
- <https://vra-va-hostname.domain.name:5480> の適切な vRealize Automation アプライアンス管理に **root** としてログインしています。

手順

- 1 [vRA 設定] - [データベース] の順にクリックします。
- 2 [同期モード] をクリックして、アクションが完了するまで待機します。
- 3 [同期の状態] 列のすべてのノードのステータスが **同期** と表示されていることを確認します。

次のステップ

[「接続テストの実行とアップグレード後のエンドポイントの確認」](#)。

接続テストの実行とアップグレード後のエンドポイントの確認

vRealize Automation 7.3 以前から 7.4 にアップグレードすると、ターゲット環境のエンドポイントが変更されます。

vRealize Automation 7.4 へのアップグレード後には、該当するすべてのエンドポイントに対して [接続をテスト] アクションを実行する必要があります。また、アップグレード後の一部のエンドポイントで調整が必要になる場合があります。詳細については、『vRealize Automation の構成』の「アップグレードまたは移行後のエンドポイントを使用する場合の考慮事項」を参照してください。

アップグレードまたは移行されたエンドポイントのデフォルトのセキュリティ設定では、信頼されていない証明書を受け入れません。

信頼されていない証明書を使用していた場合、以前の vRealize Automation インストール環境からアップグレードまたは移行した後、vSphere と NSX のすべてのエンドポイントに対して、次の手順を実行して証明書の検証を有効にする必要があります。そうしないと、エンドポイントの操作が証明書のエラーで失敗します。詳細については、VMware ナレッジベースの記事「Endpoint communication is broken after upgrade to vRA 7.3 (KB2150230)」(<http://kb.vmware.com/kb/2150230>) および「How to download and install vCenter Server root certificates to avoid Web Browser certificate warnings (KB2108294)」(<http://kb.vmware.com/kb/2108294>) を参照してください。

- 1 アップグレード後または移行後に、vRealize Automation vSphere エージェント マシンにログインし、[サービス] タブを使用して vSphere エージェントを再起動します。

移行ではすべてのエージェントが再起動されない場合があるため、必要に応じて手動で再起動します。

- 2 少なくとも 1 つの ping レポートが終了するのを待ちます。ping レポートの完了には 1、2 分かかります。
- 3 vSphere エージェントがデータ収集を開始したら、IaaS 管理者として vRealize Automation にログインします。
- 4 [インフラストラクチャ] - [エンドポイント] - [エンドポイント] の順にクリックします。
- 5 vSphere エンドポイントを編集し、[接続をテスト] をクリックします。
- 6 証明書のプロンプトが表示されたら、[OK] をクリックして証明書を受け入れます。

証明書のプロンプトが表示されない場合、証明書が現在、プロキシ エージェント マシンや DEM マシンなどのエンドポイントのサービスをホストする Windows マシンの信頼されたルート認証局に正しく保存されていない可能性があります。
- 7 [OK] をクリックして、証明書の承認を適用し、エンドポイントを保存します。
- 8 vSphere エンドポイントごとにこの手順を繰り返します。
- 9 NSX エンドポイントごとにこの手順を繰り返します。

[接続をテスト] 操作は成功したもの、一部のデータ収集やプロビジョニング操作が失敗した場合、エンドポイントとして機能するすべてのエージェント マシンとすべての DEM マシンに同じ証明書をインストールできます。または、既存のマシンから証明書をアンインストールして、問題のあるエンドポイントに対して上記の手順を繰り返します。

vRealize Automation からのアップグレード後の NSX ネットワークおよびセキュリティ インベントリ データ収集の実行

vRealize Automation 7.1、7.2、7.3 から 7.4 にアップグレードした後は、vRealize Automation 7.4 環境内で NSX ネットワークおよびセキュリティ インベントリ データ収集を実行する必要があります。

このデータ収集は、vRealize Automation 7.4 で 7.1、7.2、および 7.3.x の展開用にロード バランサの再構成アクションを行うために必要です。

前提条件

- [「vRealize Automation のアップグレード前の NSX ネットワークおよびセキュリティ インベントリ データ収集の実行」](#)。
- vRealize Automation 7.4 への正常なアップグレード

手順

- ◆ vRealize Automation 7.4 に移行する前に、ソースの vRealize Automation 環境で NSX ネットワークとセキュリティ インベントリのデータ収集を実行します。『vRealize Automation の管理』の「手動によるエンドポイント データ収集の開始」を参照してください。

レプリカ アプライアンスのクラスタへの参加

マスター vRealize Automation アプライアンスのアップデートが完了すると、更新されたレプリカ ノードはそれぞれ自動的にマスター ノードに参加します。レプリカ ノードを個別に更新する必要がある場合は、次の手順を使用して手動でレプリカ ノードをクラスタに参加させます。

クラスタに参加していないレプリカ ノードのアプライアンス管理コンソールにアクセスして、次の手順を実行します。

手順

- 1 [vRA 設定] - [クラスタ] の順に選択します。
- 2 [クラスタに参加] をクリックします。

高可用性を展開する環境でのポート構成

リモート コンソール機能をサポートするには、高可用性を展開している環境をアップグレードした後、ポート 8444 のトラフィックが vRealize Automation アプライアンスに渡されるようにロード バランサを構成する必要があります。

詳細については、vRealize Automation ドキュメントの『vRealize Automation Load Balancing Configuration Guide』を参照してください。

組み込みの vRealize Orchestrator で高可用性をサポートするための再構成

高可用性環境では、各ターゲット レプリカ vRealize Automation アプライアンスを手動でクラスタに再参加させて、組み込みの vRealize Orchestrator で高可用性のサポートを有効にする必要があります。

前提条件

ターゲット レプリカ vRealize Automation アプライアンス管理コンソールにログインします。

- 1 ブラウザを起動し、ターゲット レプリカ仮想アプライアンスの完全修飾ドメイン名 (FQDN) `https://<vra-vr-hostname.domain.name>:5480` を使用して、ターゲット レプリカ vRealize Automation 管理コンソールを開きます。
- 2 ユーザー名 **root** と、ターゲット レプリカ vRealize Automation アプライアンスの展開時に入力したパスワードを使用してログインします。

手順

- 1 [vRA 設定] - [クラスタ] の順に選択します。
- 2 [先頭のクラスタ ノード] テキスト ボックスに、ターゲット マスター vRealize Automation アプライアンスの FQDN を入力します。
- 3 [パスワード] テキスト ボックスに root パスワードを入力します。
- 4 [クラスタに参加] をクリックします。

証明書の警告を無視して続行します。システムによってクラスタのサービスが再起動されます。

- 5 サービスが実行されていることを確認します。
 - a 最上部のタブ バーで、[サービス] をクリックします。
 - b サービスの起動の進行状況を監視するには、[更新] をクリックします。

外部ワークフローのタイムアウト ファイルのリストア

アップグレード プロセスによって xmlldb ファイルが上書きされてしまうため、vRealize Automation の外部ワークフローのタイムアウト ファイルを再構成する必要があります。

手順

- 1 次のディレクトリから、システム上の外部ワークフロー構成 (xmldb) ファイルを開きます。
`\\VMware\\vCAC\\Server\\ExternalWorkflows\\xmldb\\`
- 2 xmldb ファイルを移行の前にバックアップしたファイルに置き換えます。バックアップ ファイルがない場合は、外部ワークフローのタイムアウト設定を再構成します。
- 3 設定を保存します。

ユーザー用リモート コンソール アクションとの接続の有効化

ユーザー用リモート コンソール アクションは、vRealize Automation で vSphere によってプロビジョニングされるアプライアンスでサポートされています。

このリリースをアップグレードしたらブループリントを編集し、[アクション] タブの [リモート コンソールに接続] アクションを選択します。

詳細については、[ナレッジ ベースの記事 2109706](#) を参照してください。

app.config ファイルに行ったログの変更のリストア

アップグレード プロセスでは、構成ファイルのログへの変更が上書きされます。アップグレードが終了した後、アップグレード前に **app.config** ファイルに行った変更をリストアする必要があります。

アップグレード後に Manager Service の自動フェイルオーバーを有効にする

vRealize Automation をアップグレードすると、Manager Service の自動フェイルオーバーがデフォルトで無効になります。

アップグレード後に Manager Service の自動フェイルオーバーを有効にするには、次の手順を実行します。

手順

- 1 vRealize Automation アプライアンスで、root ユーザーとしてコマンド プロンプトを開きます。
- 2 ディレクトリを `/usr/lib/vcac/tools/vami/commands` に変更します。
- 3 Manager Service の自動フェイルオーバーを有効にするには、次のコマンドを実行します。

```
python ./manager-service-automatic-failover ENABLE
```

laaS 環境全体で自動フェイルオーバーを無効にするには、次のコマンドを実行します。

```
python ./manager-service-automatic-failover DISABLE
```

Manager Service の自動フェイルオーバーについて

vRealize Automation laaS Manager Service は、プライマリの Manager Service が停止した場合、バックアップへのフェイルオーバーを自動的に実行するように設定できます。

vRealize Automation 7.3 以降では、サーバをプライマリまたはバックアップに設定する際、Windows サーバごとに手動で Manager Service を開始または停止する必要がなくなりました。アップグレードシェルスクリプトまたは IaaS インストーラ実行可能ファイルを使用して IaaS をアップグレードするときに、自動の Manager Service のフェイルオーバーはデフォルトで無効になります。

自動フェイルオーバーを有効にすると、バックアップを含むすべての Manager Service ホストで自動的に Manager Service が開始されます。自動フェイルオーバー機能により、ホストは透過的に互いを監視し、必要に応じてフェイルオーバーを実行しますが、すべてのホストで Windows サービスが実行されている必要があります。

注: 自動フェイルオーバーの使用は必須ではありません。自動フェイルオーバーを無効にして、引き続き手動で Windows サービスを開始および停止し、プライマリまたはバックアップ ホストの設定を制御することもできます。手動でフェイルオーバーを行う際は、一度に 1 台ずつホストのサービスを開始してください。自動フェイルオーバーが無効の場合、同時に複数の IaaS サーバでサービスを実行すると、vRealize Automation を使用できなくなります。

自動フェイルオーバーをホストごとに有効または無効に設定しないでください。IaaS 環境では、自動フェイルオーバーを有効または無効にする設定は、すべての Manager Service ホストで同一にする必要があります。

vRealize Automation アップグレード のトラブルシューティング

9

アップグレードのトラブルシューティングに関するトピックでは、vRealize Automation 7.1、7.2、7.3.x から 7.4 へのアップグレード時に発生する可能性のある問題の解決策を示します。

この章には、次のトピックが含まれています。

- Manager Service の自動フェイルオーバーが有効にならない
- ロード バランサのタイムアウト エラーでインストールまたはアップグレードに失敗する
- IaaS Web サイト コンポーネントのアップグレードに失敗する
- 実行中の SSL 検証エラーが原因で Manager Service の実行に失敗する
- アップグレード後のログインの失敗
- vRealize Automation での実体のないノードの削除
- 高可用性環境アップグレード後にクラスタへの参加コマンドが失敗したように表示される
- PostgreSQL データベースのアップグレード マージが成功しない
- レプリカ vRealize Automation アプライアンスのアップデートに失敗する
- .xml ファイルのバックアップ コピーによってシステムがタイムアウトする
- IaaS アップグレードの除外
- vRealize Automation で新規ディレクトリを作成できない
- vRealize Automation レプリカ仮想アプライアンスのアップデートがタイムアウトになる
- アップグレード中に作成された展開が一部の仮想マシンに存在しない
- 信頼されていない証明書に関するエラー
- vRealize Automation のインストールまたはアップグレードが失敗する
- DEM および DEO コンポーネントを更新できない
- 更新で管理エージェントのアップグレードに失敗する
- 管理エージェントのアップグレードに失敗する
- デフォルトのタイムアウト設定が原因で vRealize Automation のアップデートに失敗する
- 高可用性環境での IaaS のアップグレードが失敗する

■ アップグレードの問題の回避

Manager Service の自動フェイルオーバーが有効にならない

manager-service-automatic-failover コマンドのトラブルシューティングに関する推奨事項。

ソリューション

- manager-service-automatic-failover コマンドが失敗するか、2 分間以上次のメッセージが表示されます：**ノードで Manager Service の自動フェイルオーバー モードが有効になっています：<IAAS_MANAGER_SERVICE_NODEID>。**
 - a アプライアンスを展開したときに入力したユーザー名 **host** とパスワードを使用して、https://<va-hostname.domain.name>:5480 で vRealize Automation アプライアンス管理にログインします。
 - b [vRA 設定] - [クラスタ] の順に選択します。
 - c すべての Manager Service ホストで管理エージェント サービスが実行されていることを確認します。
 - d すべての IaaS Manager Service ノードの最終接続時間が 30 秒未満であることを確認します。

管理エージェント接続の問題が見つかった場合は、手動で解決し、Manager Service の自動フェイルオーバーを有効にするコマンドをもう一度実行してください。
- Manager-service-automatic-failover コマンドが、Manager Service ノード上でのフェイルオーバーを有効にするのに失敗します。この問題を解決するためにコマンドを再実行するのは問題ありません。
- IaaS 展開の一部の Manager Service ホストでフェイルオーバーが有効になっていますが、他のホストでは有効になっていません。IaaS 展開のすべての Manager Service ホストの機能は有効になっていますが、機能していません。この問題を修正するには、次のいずれかを実行します。
 - すべての Manager Service ノードのフェイルオーバーを無効にし、代わりに手動のフェイルオーバーのアプローチを使用します。一度に 1 台のホストでのみフェイルオーバーを実行します。
 - Manager Service ノードで何度かこの機能を有効にしようとして失敗した場合は、このノード上の Windows の VMware vCloud Automation Center サービスを停止し、問題を解決するまでノードの起動タイプを手動に設定します。
- Python を使用して、各 Manager Service ノードでフェイルオーバーが有効であることを検証します。
 - a SSH を使用して、マスター vRealize Automation アプライアンス ノードに **root** としてログインします。
 - b **python /usr/lib/vcac/tools/vami/commands/manager-service-automatic-failover ENABLE** を実行します。
 - c 次のメッセージが返されることを確認します。**ノードで Manager Service 自動フェイルオーバー モードが有効になっています：<IAAS_MANAGER_SERVICE_NODEID> 完了。**
- Manager Service 構成ファイルを調べることによって各 Manager Service ノードでフェイルオーバーが有効になっていることを確認します。
 - a Manager Service ノードでコマンド プロンプトを開きます。

- b vRealize Automation インストール フォルダに移動し、
VMware\VCAC\Server\ManagerService.exe.config で Manager Service 構成ファイルを開きます。
- c 次の要素が <appSettings> セクションに存在することを確認します。
 - <add key="FailoverModeEnabled" value="True" />
 - <add key="FailoverPingIntervalMilliseconds" value="30000" />
 - <add key="FailoverNodeState" value="active" />
 - <add key="FailoverMaxFailedDatabasePingAttempts" value="5" />
 - <add key="FailoverMaxFailedRepositoryPingAttempts" value="5" />
- Windows の VMware vCloud Automation Center サービスのステータスが起動し、起動タイプが自動であることを確認します。
- Python を使用して、各 Manager Service ノードでフェイルオーバーが無効であることを検証します。
 - a SSH を使用して、マスター vRealize Automation アプライアンス ノードに **root** としてログインします。
 - b **python /usr/lib/vcac/tools/vami/commands/manager-service-automatic-failover DISABLE** を実行します。
 - c 次のメッセージが返されることを確認します。**ノードで Manager Service 自動フェイルオーバー モードが無効になっています:<IAAS_MANAGER_SERVICE_NODEID> 完了。**
- Manager Service 構成ファイルを調べることによって各 Manager Service ノードでフェイルオーバーが無効になっていることを確認します。
 - a Manager Service ノードでコマンド プロンプトを開きます。
 - b vRealize Automation インストール フォルダに移動し、
VMware\VCAC\Server\ManagerService.exe.config で Manager Service 構成ファイルを開きます。
 - c 次の要素が <appSettings> セクションに存在することを確認します。
 - <add key="FailoverModeEnabled" value="False" />
- コールド スタンバイの Manager Service ノードを作成するには、ノードの Windows の VMware vCloud Automation Center サービスのステータスを停止にして、起動タイプを手動に設定します。
- アクティブな Manager Service ノードでは、ノードの Windows の VMware vCloud Automation Center サービスのステータスを開始にして、起動タイプを自動にする必要があります。
- **manager-service-automatic-failover** コマンドでは、Manager Service ノードの内部 ID (<IAAS_MANAGER_SERVICE_NODEID>) を使用します。この内部 ID に対応するホスト名を検索するには、コマンド **vra-command list-nodes** を実行し、ノード ID <IAAS_MANAGER_SERVICE_NODEID> を使用して Manager Service ホストを探します。
- 自動で現在アクティブになるように選択されている Manager Service を見つけるには、次の手順を実行します。
 - a SSH を使用して、マスター vRealize Automation アプライアンス ノードに **root** としてログインします。

b `vra-command list-nodes --components` を実行します。

- フェイルオーバーが有効になっている場合は、ステータスがアクティブの Manager Service ノードを検索します。
- フェイルオーバーが無効になっている場合は、ステータスが開始済みの Manager Service ノードを検索します。

ロード バランサのタイムアウト エラーでインストールまたはアップグレードに失敗する

ロード バランサを使用した分散環境を実現するための vRealize Automation のインストールまたはアップグレードが、503 サービス利用不能エラーで失敗します。

問題

ロード バランサ タイムアウトの設定が原因でタスクを完了するための十分な時間が確保できないため、インストールまたはアップグレードに失敗します。

原因

ロード バランサ タイムアウトの設定値が小さいとエラーになる可能性があります。この問題を修正するには、ロード バランサ タイムアウトの設定値を 100 秒以上に増やしてタスクを再実行します。

ソリューション

- 1 ロード バランサ タイムアウト値を最低でも 100 秒に増やします。
- 2 インストールまたはアップグレードを再実行します。

laaS Web サイト コンポーネントのアップグレードに失敗する

laaS のアップグレードに失敗し、アップグレードを続行できません。

問題

Web サイト コンポーネントの laaS アップグレードに失敗します。インストーラ ログ ファイルに次のエラー メッセージが表示されます。

- System.Data.Services.Client.DataServiceQueryException:
An error occurred while processing this request. --->
System.Data.Services.Client.DataServiceClientException: <!DOCTYPE html>
■ Description: An application error
occurred on the server. The current custom error settings for this application
prevent the details of the application error from being viewed remotely (for
security reasons). It could, however, be viewed by browsers running on the
local server machine.

- Warning: Non-zero return code. Command failed.
- Done Building Project "C:\Program Files
(x86)\VMware\vCAC\Server\Model Manager Data\DeployRepository.xml"
(InstallRepoModel target(s)) -- FAILED.

リポジトリ ログ ファイルに次のエラー メッセージが表示されます。

- [Error]: [sub-thread-Id="20"
context="" token=""] Failed to start repository service. Reason:
System.InvalidOperationException: Configuration section encryptionKey is not
protected
at
DynamicOps.Common.Utils.EncryptionHelpers.ReadKeyFromConfiguration(Configuration
config)
at DynamicOps.Common.Utils.EncryptionHelpers.Decrypt(String value)
at DynamicOps.Repository.Runtime.CoreModel.GlobalPropertyItem.Decrypt(Func`2
decryptFunc)
at
DynamicOps.Common.Entity.ContextHelpers.OnObjectMaterializedCallbackEncryptable(Object
sender, ObjectMaterializedEventArgs e)
at
System.Data.Common.Internal.Materialization.Shaper.RaiseMaterializedEvents()
at
System.Data.Common.Internal.Materialization.Shaper`1.SimpleEnumerator.MoveNext()
System.Linq.Enumerable.FirstOrDefault[TSource](IEnumerable`1 source)
System.Linq.Queryable.FirstOrDefault[TSource](IQueryable`1 source)
at
DynamicOps.Repository.Runtime.Common.GlobalPropertyHelper.GetGlobalPropertyItemValue(Core
ModelEntities
coreModelContext, String propertyName, Boolean throwIfPropertyNotFound)
at
DynamicOps.Repository.Runtime.CafeClientAbstractFactory.LoadSolutionUserCertificate()
at
DynamicOps.Repository.Runtime.CafeClientAbstractFactory.InitializeFromDb(String

```
coreModelConnectionString)
```

```
DynamicOps.Repository.Runtime.Common.RepositoryRuntime.Initialize();
```

原因

laaS アップグレードは、**web.config** ファイルの作成日が、変更日と同じまたはそれ以降の日付になっている場合に失敗します。

ソリューション

- 1 laaS ホスト上で、Windows にログインします。
- 2 Windows コマンド プロンプトを開きます。
- 3 vRealize Automation インストール フォルダに移動します。
- 4 [管理者として実行] オプションで任意のテキスト エディタを起動します。
- 5 **web.config** ファイルの場所を特定して選択し、ファイルを保存し直すことで、このファイルの変更日を作成日より後に変更できます。
- 6 **web.config** ファイルのプロパティを調べて、ファイル変更日が作成日より後であることを確認します。
- 7 laaS をアップグレードします。

実行中の SSL 検証エラーが原因で Manager Service の実行に失敗する

SSL 検証エラーが原因で、Manager Service の実行に失敗します。

問題

Manager Service が失敗し、ログに次のエラー メッセージが記録されます。

```
[Info]: Thread-Id="6" - context="" token="" Failed to connect to the core database, will retry in 00:00:05, error details: A connection was successfully established with the server, but then an error occurred during the login process. (provider: SSL Provider, error: 0 - The certificate chain was issued by an authority that is not trusted.)
```

原因

実行時、SSL 検証エラーが原因で、Manager Service の実行に失敗します。

ソリューション

- 1 **ManagerService.config** 構成ファイルを開きます。

- 2 次の行で **Encrypt=False** に更新します。

```
<add name="vcac-repository" providerName="System.Data.SqlClient"
connectionString="Data Source=iaas-db.sqa.local;Initial
Catalog=vcac;Integrated Security=True;Pooling=True;Max Pool
Size=200;MultipleActiveResultSets=True;Connect Timeout=200,
Encrypt=True" />
```

アップグレード後のログインの失敗

同期されていないユーザー アカウントを使用するセッションでは、アップグレード後にブラウザを終了し、もう一度ログインする必要があります。

問題

vRealize Automation をアップグレードすると、ログイン時に同期されていないユーザー アカウントへのアクセスは拒否されます。

ソリューション

ブラウザを終了し、vRealize Automation を再起動します。

vRealize Automation での実体のないノードの削除

実体のないノードとは、ホスト上で報告されているのにそのホストに存在しない重複ノードです。

問題

各 IaaS ノードおよび仮想アプライアンス ノードが健全な状態にあることを確認するとき、あるホストに実体のないノードが 1 つ以上あることに気付くことがあります。実体のないノードはすべて削除する必要があります。

ソリューション

- 1 プライマリ vRealize Automation アプライアンスで、vRealize Automation アプライアンスを展開したときに入力したパスワードを使用して、**root** として vRealize Automation アプライアンス管理にログインします。
- 2 [vRA 設定] - [クラスタ] の順に選択します。
- 3 表内のそれぞれの実体のないノードに対して、[削除] をクリックします。

高可用性環境アップグレード後にクラスタへの参加コマンドが失敗したように表示される

セカンダリ クラスタ ノードの管理コンソールで [クラスタに参加] をクリックした後、進行状況インジケータが表示されなくなります。

問題

アップグレード後に vRealize Automation アプライアンス管理コンソールを使用してセカンダリ クラスタ ノードをプライマリ ノードに参加させると、進行状況インジケータが消え、エラー メッセージも正常完了メッセージも表示されなくなります。この挙動は断続的に確認される問題です。

原因

進行状況インジケータが消えるのは、一部のブラウザがサーバからの応答待ちを中止するためです。この挙動によってクラスタへの参加プロセスが停止することはありません。正常にクラスタへの参加が完了した場合、ログ ファイル (/var/log/vmware/vcac/vcac-config.log) を表示することで確認できます。

PostgreSQL データベースのアップグレード マージが成功しない

外部の PostgreSQL データベースは組み込みの PostgreSQL データベースと正常にマージされません。

問題

PostgreSQL データベースと正常にアップグレード マージが行われない場合は、手動によるマージを実行できます。

ソリューション

- 1 vRealize Automation 仮想アプライアンスを、アップグレード前に作成したスナップショットの状態に戻します。
- 2 vRealize Automation 仮想アプライアンスにログインし、次のコマンドを実行して、データベースが正常にマージされなくてもアップグレードを完了するようにします。

touch /tmp/allow-external-db

このコマンドは自動マージを無効にしません。

- 3 リモート PostgreSQL データベース ホストで、psql ツールを使用して PostgreSQL データベースに接続し、以下のコマンドを実行します。

```
CREATE EXTENSION IF NOT EXISTS "hstore";
```

```
CREATE EXTENSION IF NOT EXISTS "uuid-osspl";
```

```
CREATE SCHEMA saas AUTHORIZATION vcac;
```

このコマンドではユーザーが vcac になっています。vRealize Automation が別のユーザーによって外部データベースに接続している場合は、このコマンド内の vcac をそのユーザーの名前で置き換えます。

```
CREATE EXTENSION IF NOT EXISTS "citext" SCHEMA saas;
```

- 4 アップグレードを実行します。

アップグレードが正常に完了した場合、システムは外部の PostgreSQL データベースを使用して期待どおりに動作します。外部 PostgreSQL データベースが適切に動作していることを確認します。

- 5 vRealize Automation 仮想アプライアンスにログインし、以下のコマンドを実行します。

```
/etc/bootstrap/postupdate.d/00-20-db-merge-external
```

```
/etc/bootstrap/postupdate.d/11-db-merge-external
```

レプリカ vRealize Automation アプライアンスのアップデートに失敗する

マスター アプライアンスのアップデート中に、レプリカ vRealize Automation アプライアンスのアップデートに失敗します。

原因

接続の問題やその他の障害により、レプリカ アプライアンスがアップデートに失敗する場合があります。その場合、マスター vRealize Automation アプライアンスの [更新] タブに警告メッセージが表示され、アップデートに失敗したレプリカが強調表示されます。

ソリューション

- 1 レプリカの仮想アプライアンスのスナップショットまたはバックアップをアップデート前の状態に戻して、パワーオンします。
- 2 レプリカの vRealize Automation アプライアンスの管理インターフェイスに root としてログインします。
`https://<vrealize-automation-appliance-FQDN>:5480`
- 3 [更新] - [設定] の順にクリックします。
- 4 [リポジトリをアップデート] セクションで、アップデートを VMware リポジトリからダウンロードするか CD-ROM から取得するかを選択します。
- 5 [ステータス] をクリックします。
- 6 [アップデートの確認] をクリックし、アップデートが利用可能かどうかを確認します。
- 7 [アップデートをインストール] をクリックします。
- 8 [OK] をクリックします。

アップデート処理が進行していることを示すメッセージが表示されます。

- 9 ログ ファイルを開いて、アップグレードが正常に進行していることを確認します。

- `/opt/vmware/var/log/vami/vami.log`
- `/var/log/vmware/horizon/horizon.log`

アップグレード中にログアウトし、アップグレードの完了前に再度ログインした場合は、ログ ファイルで更新の進行状況を継続して確認できます。**updatecli.log** ファイルに、アップグレード前のバージョンの vRealize Automation の情報が表示される場合があります。表示されたバージョンは、アップグレード プロセスの中で適切なバージョンに変わります。

アップデートが終了するまでの時間は、環境によって異なります。

- 10 アップデートが完了したら、仮想アプライアンスを再起動します。
 - a [システム] をクリックします。
 - b [再起動] をクリックし、選択を確認します。
- 11 [vRA 設定] - [クラスタ] の順に選択します。

12 マスター vRealize Automation アプライアンスの FQDN を入力し、[クラスタに参加] をクリックします。

.xml ファイルのバックアップ コピーによってシステムがタイムアウトする

vRealize Automation は、\VMware\VCAC\Server\ExternalWorkflows\xmlldb\ ディレクトリにある拡張子が .xml であるすべてのファイルを登録します。このディレクトリに拡張子が .xml のバックアップ ファイルが含まれていると、システムは重複するワークフローを実行するため、システムがタイムアウトします。

ソリューション

回避策：このディレクトリのファイルをバックアップするときは、バックアップ ファイルを別のディレクトリに移動するか、バックアップ ファイルの拡張子を .xml 以外の拡張子に変更します。

laaS アップグレードの除外

laaS コンポーネントをアップグレードせずに vRealize Automation アプライアンスをアップデートできます。

laaS コンポーネントをアップグレードせずに vRealize Automation アプライアンスをアップデートする場合は、この手順を使用します。この手順では、

- laaS サービスは停止しません。
- 管理エージェントの更新をスキップします。
- vRealize Automation アプライアンスの更新後に、laaS コンポーネントの自動更新が行われないようにします。

手順

- 1 プライマリ vRealize Automation アプライアンス ノードへのセキュアなシェル接続を開きます。
- 2 コマンド プロンプトで、以下のコマンドを実行してグル ファイルを作成します。

```
touch /tmp/disable-iaas-upgrade
```

- 3 laaS サービスを手動で停止します。
 - a laaS Windows サーバにログインします。
 - b [スタート] - [管理ツール] - [サービス] を選択します。
 - c 次の順序でサービスを停止します。

注： laaS Windows サーバはシャットダウンしないでください。

- 1 各 VMware vRealize Automation プロキシ エージェント。
 - 2 各 VMware DEM ワーカー。
 - 3 VMware DEM orchestrator。
 - 4 VMware vCloud Automation Center サービス。
- 4 プライマリ vRealize Automation アプライアンス管理コンソールにアクセスして、プライマリ vRealize Automation アプライアンスをアップデートします。

vRealize Automation で新規ディレクトリを作成できない

最初の同期コネクタを使用して新規ディレクトリを追加しようとすると失敗します。

問題

この問題は、`usr/local/horizon/conf/states/VSPHERE.LOCAL/3001/`にある、正しくない `config-state.json` ファイルが原因で発生します。

この問題の解決方法については、[ナレッジベースの記事 KB2145438](#) を参照してください。

vRealize Automation レプリカ仮想アプライアンスのアップデートがタイムアウトになる

マスター仮想アプライアンスをアップデートする際、vRealize Automation レプリカ仮想アプライアンスのアップデートがタイムアウトになります。

問題

マスター仮想アプライアンスをアップデートする際、マスター vRealize Automation 管理コンソールの更新タブには、アップデートのタイムアウト制限を超えたレプリカ仮想アプライアンスが強調表示されます。

原因

アップデートがタイムアウトするのは、パフォーマンスまたはインフラストラクチャの問題が原因です。

ソリューション

- 1 レプリカ仮想アプライアンスのアップデート状況を確認します。
 - a 完全修飾ドメイン名 (FQDN) `https://<va-hostname.domain.name>:5480` を使用してレプリカ仮想アプライアンスの管理コンソールに移動します。
 - b ユーザー名 **root** と、アプライアンスを展開したときに入力したパスワードを使用してログインします。
 - c [更新] - [ステータス] の順に選択し、アップデート状況を確認します。

次のいずれかの操作を行います。

 - アップデートに失敗した場合は、トラブルシューティングトピック「[レプリカ vRealize Automation アプライアンスのアップデートに失敗する](#)」の手順を実行します。
 - レプリカ仮想アプライアンスのアップグレードが進行中の場合は、アップグレードが完了するまで待機し、手順 2 に進みます。
- 2 仮想アプライアンスを再起動します。
 - a [システム] をクリックします。
 - b [再起動] をクリックし、選択を確認します。
- 3 [vRA 設定] - [クラスタ] の順に選択します。
- 4 マスター vRealize Automation 仮想アプライアンスの FQDN を入力し、[クラスタに参加] をクリックします。

アップグレード中に作成された展開が一部の仮想マシンに存在しない

アップグレード時に仮想マシンの状態が指定なしの場合、ターゲット環境で対応する展開が作成されません。

問題

アップグレード時に仮想マシンの状態がソース環境で指定なしの場合、対応する展開がターゲット環境で作成されません。アップグレード後に仮想マシンの状態が指定なしから変わると、バルク インポートを使用してターゲット展開にマシンをインポートすることができます。

信頼されていない証明書に関するエラー

vRealize Automation アプライアンス コンソールの [ログ ビューア] ページでインフラストラクチャを表示すると、**Certificate is not trusted** のような文言を含むエンドポイント接続障害レポートが表示されることがあります。

問題

vRealize Automation アプライアンス コンソールで、[インフラストラクチャ] - [監視] - [ログ] の順に選択します。[ログ ビューア] ページに、次のようなレポートが表示される場合があります。

エンドポイントへの接続に失敗しました。このエンドポイントへの安全な接続を確立できることを検証するには、[エンドポイント] ページで vSphere エンドポイントに移動し、[テスト接続] ボタンをクリックします。

内部例外: 証明書が信頼されていません (RemoteCertificateChainErrors)。件名: C=US, CN=vc6.mycompany.com
サムプリント: DC5A8816231698F4C9013C42692B0AF93D7E35F1

原因

vRealize Automation 7.3 以前のバージョンから 7.4 にアップグレードすると、元の環境にあったエンドポイントに変更が加えられます。最近 vRealize Automation 7.4 にアップグレードした環境では、安全な https 接続を使用するそれぞれの既存のエンドポイントを IaaS 管理者が確認する必要があります。「**Certificate is not trusted**」エラーが発生したエンドポイントは、適切に機能しません。

ソリューション

- 1 インフラストラクチャ管理者として vRealize Automation コンソールにログインします。
- 2 [インフラストラクチャ] - [エンドポイント] - [エンドポイント] を選択します。
- 3 安全な接続が使用されている各エンドポイントについて、次の手順を実行します。
 - a [編集] をクリックします。
 - b [テスト接続] をクリックします。
 - c 証明書の詳細を確認し、この証明書を信頼する場合は、[OK] をクリックします。
 - d このエンドポイントによって使用されるすべての IaaS プロキシ エージェントの Windows サービスを再起動します。

- 4 インフラストラクチャの [ログ ビューア] ページに「**Certificate is not trusted**」エラーが表示されなくなったことを確認します。

vRealize Automation のインストールまたはアップグレードが失敗する

vRealize Automation のインストールまたはアップグレードが失敗し、エラー メッセージがログ ファイルに表示されます。

問題

vRealize Automation のインストールまたはアップグレードが失敗します。これは通常、インストールまたはアップグレード中に適用される修正に失敗した場合に発生します。ログ ファイルに次のようなエラー メッセージが表示されます。**Security error. Applying automatic fix for FIREWALL prerequisite failed. RPM Status 1: Pre install script failed, package test and installation skipped**

原因

Windows 環境のグループ ポリシーで、PowerShell スクリプトの実行が [有効] に設定されています。

ソリューション

- 1 Windows ホスト マシンで **gpedit.msc** を実行してローカル グループ ポリシー エディターを開きます。
- 2 [コンピューターの構成] の下にある左側のペインで、展開ボタンをクリックして [管理用テンプレート] - [Windows コンポーネント] - [Windows PowerShell] の順に開きます。
- 3 [スクリプトの実行を有効にする] で、状態を **Enabled** から **Not Configured** に変更します。

DEM および DEO コンポーネントを更新できない

vRealize Automation 7.2 から 7.3.x へのアップデート中に、DEM および DEO コンポーネントを更新できません

問題

vRealize Automation 7.2 から 7.3.x へのアップデート後、D: ドライブなどのカスタム パスにインストールされている DEM および DEO コンポーネントが更新されません。

[ナレッジベースの記事 KB2150517](#) を参照してください。

更新で管理エージェントのアップグレードに失敗する

vRealize Automation アプライアンス 管理コンソールの [ステータスの更新] ページで [更新のインストール] をクリックすると、管理エージェントに関するエラー メッセージが表示されます。

問題

アップグレード プロセスが成功しない。「**ノード x で管理エージェントをアップグレードできません**」というメッセージが表示されます。メッセージに複数のノードがリストされる場合もあります。

原因

この問題はさまざまな条件によって発生します。エラー メッセージには、影響を受けるマシンのノード ID のみが表示されます。コマンドが失敗したマシン上の管理エージェントについての詳細情報は、**All.log** ファイルに含まれています。

状況に応じて、影響を受けるノードで次のタスクを実行します。

ソリューション

- 管理エージェント サービスが実行されていない場合は、このサービスを開始し、仮想アプライアンスでアップグレードを再開します。
- 管理エージェント サービスが実行されており、管理エージェントがアップグレードされている場合は、仮想アプライアンスでアップグレードを再開します。
- 管理エージェント サービスが実行されているものの、管理エージェントがアップグレードされていない場合は、手動でアップグレードを実行します。
 - a ブラウザを開き、vRealize Automation アプライアンス上の [vRealize Automation IaaS のインストール] ページ (<https://<va-hostname.domain.name>:5480/install>) に移動します。
 - b 管理エージェントのインストーラをダウンロードして、実行します。
 - c 管理エージェントのマシンを再起動します。
 - d 仮想アプライアンスでアップグレードを再開します。

管理エージェントのアップグレードに失敗する

vRealize Automation から 7.2- 7.3.x へのアップグレード時に、管理エージェントのアップグレードに失敗します。

問題

フェイルオーバーの発生によってプライマリとセカンダリの管理エージェントのホストが入れ替わった場合、自動化されたアップグレード プロセスによって想定されるホストを見つけることができないため、アップグレードに失敗します。管理エージェントがアップグレードされていない各 IaaS ノードで、この手順を実行します。

ソリューション

- 1 管理エージェント ログ フォルダ (**C:\Program Files (x86)\VMware\VCAC\Management Agent\Logs**) にある All.log を開きます。

インストール フォルダの場所は、デフォルトの場所とは異なる場合があります。

- 2 ログ ファイル内を検索して、古くなった仮想アプライアンスやパワーオフされている仮想アプライアンスに関するメッセージを探します。

たとえば、次のようなものです。INNER EXCEPTION: System.Net.WebException: Unable to connect to the remote server ----> System.Net.Sockets.SocketException: A connection attempt failed because the connected party did not properly respond after a period of time, or established connection failed because connected host has failed to respond <IP_Address>:5480

- 管理エージェントの構成ファイル (C:\Program Files (x86)\VMware\vCAC\Management Agent\VMware.IaaS.Management.Agent.exe.config) を編集して、既存の alternativeEndpointaddress の値をプライマリ仮想アプライアンス エンドポイントの URL で置き換えます。

インストール フォルダの場所は、デフォルトの場所とは異なる場合があります。

VMware.IaaS.Management.Agent.exe.config での alternativeEndpointaddress の例を次に示します。

```
<alternativeEndpoint address="https://<FQDN>:5480/"
thumbprint="<thumbprint number>" />
```

- 管理エージェント Windows サービスを再起動し、All.log ファイルを参照してこのサービスが機能していることを確認します。
- プライマリ vRealize Automation アプライアンスでアップグレード手順を実行します。

デフォルトのタイムアウト設定が原因で vRealize Automation のアップデートに失敗する

使用環境で、データベース同期のデフォルトの時間設定が短すぎる場合は、アップデートの時間設定を長くすることができます。

問題

データベースの同期にデフォルトのタイムアウト設定値である 3,600 秒よりも長い時間を要する一部の環境では、Vcac-Config SynchronizeDatabases コマンドのタイムアウト設定は不十分です。

API と Vcac-config.exe ユーティリティ ツール間の通信は、Vcac-Config.exe.config ファイルの cafeTimeoutInSeconds および cafeRequestPageSize プロパティの値によって制御されています。このファイルのパスは、<IaaS installation location>\VMware\vCAC\Server\Model Manager Data\Cafe\Vcac-Config.exe.config です。

これらのオプションパラメータに値を指定して、SynchronizeDatabases コマンドのみに対して、デフォルトのタイムアウト値をオーバーライドすることができます。

パラメータ	短縮名	説明
--DatabaseSyncTimeout	-dstm	SynchronizeDatabases に対してのみ HTTP 要求のタイムアウト値を秒単位で設定します。
--DatabaseSyncPageSize	-dsps	予約または予約ポリシーの同期に対してのみ、同期要求のページサイズを設定します。デフォルトは 10 です。

これらのパラメータが Vcac-Config.exe.config ファイルに設定されていない場合は、デフォルトのタイムアウト値が使用されます。

高可用性環境での IaaS のアップグレードが失敗する

プライマリ Web サーバ ノードでロード バランシングを有効にして IaaS アップグレード処理を実行すると失敗します。次のエラー メッセージが表示される可能性があります。「System.Net.WebException: 処理がタイムアウトしました」または「401 - Unauthorized: 無効な認証情報のため、アクセスは拒否されました。」

問題

ロード バランシングを有効にして IaaS をアップグレードすると、断続的に障害が発生する可能性があります。この場合、ロード バランシングを無効にして、再度 vRealize Automation アップグレードを実行する必要があります。

ソリューション

- 1 環境を更新前のスナップショットに戻します。
- 2 プライマリ IaaS Web サーバ ノードへのリモート デスクトップ接続を開きます。
- 3 C:\windows\system32\drivers\etc の Windows の hosts ファイルに移動します。
- 4 hosts ファイルを開き、Web サーバ ロード バランサをバイパスするように次の行を追加します。
<IP_address_of_primary_iaas_website_node vrealizeautomation_iaas_website_lb_fqdn>
例：
10.10.10.5 vra-iaas-web-lb.domain.com
- 5 hosts ファイルを保存して vRealize Automation の更新を再度実行します。
- 6 vRealize Automation の更新が完了したら、hosts ファイルを開いて、手順 4 で追加した行を削除します。

アップグレードの問題の回避

アップグレード プロセスを変更してアップグレードの問題を回避することができます。

ソリューション

vRealize Automation 環境のアップグレードで問題が発生した場合は、この手順を使用して、使用可能なフラグのいずれかを選択することによりアップグレード プロセスを変更します。

手順

- 1 プライマリ vRealize Automation アプライアンス ノードへのセキュアなシェル接続を開きます。

2 コマンド プロンプトで、以下のコマンドを実行してグル ファイルを作成します。

touch <available_flag>

例：**touch /tmp/disable-iaas-upgrade**

表 9-1. 使用可能なフラグ

フラグ	説明
/tmp/disable-iaas-upgrade	<ul style="list-style-type: none"> ■ 仮想アプライアンスの再起動後の IaaS アップグレード プロセスが実行されないようにします。 ■ 管理エージェントがアップグレードされないようにします。 ■ 自動の前提条件チェックおよび修正が実行されないようにします。 ■ IaaS サービスが停止しないようにします。
/tmp/do-not-upgrade-ma	管理エージェントがアップグレードされないようにします。このフラグは、管理エージェントを手動でアップグレードする場合に適しています。
/tmp/skip-prereq-checks	自動の前提条件チェックおよび修正が実行されないようにします。このフラグは、前提条件の自動修正に問題があるために手動で修正を適用した場合に適しています。
/tmp/do-not-stop-services	IaaS サービスが停止しないようにします。アップグレードで、Manager Service、DEM、エージェントなどの IaaS Windows サービスを停止しません。
/tmp/do-not-upgrade-servers	<p>データベース、Web サイト、WAPI、リポジトリ、モデル Mfrontanager データ、Manager Service など、すべてのサーバ IaaS コンポーネントの自動アップグレードが実行されないようにします。</p> <p>注: また、このフラグは、Manager Service 自動フェイルオーバー モードが有効にされないようにします。</p>
/tmp/do-not-upgrade-dems	DEM がアップグレードされないようにします。
/tmp/do-not-upgrade-agents	IaaS プロキシ エージェントがアップグレードされないようにします。

3 選択したフラグのタスクを実行します。

表 9-2. 追加のタスク

フラグ	タスク
/tmp/disable-iaas-upgrade	<ul style="list-style-type: none"> ■ 管理エージェントを手動でアップグレードします。 ■ 必要なすべての IaaS 前提条件を手動で適用します。 ■ IaaS サービスを手動で停止します。 <ol style="list-style-type: none"> a IaaS Windows サーバにログインします。 b [スタート] - [管理ツール] - [サービス] を選択します。 c 次の順序でサービスを停止します。 <p>注: IaaS Windows サーバはシャットダウンしないでください。</p> <ol style="list-style-type: none"> a 各 VMware vRealize Automation プロキシ エージェント。 b 各 VMware DEM ワーカー。 c VMware DEM orchestrator。 d VMware vCloud Automation Center サービス。 ■ 仮想アプライアンス アップグレードが完了した後、IaaS アップグレードを手動で開始します。
/tmp/do-not-upgrade-ma	管理エージェントを手動でアップグレードします。
/tmp/skip-prereq-checks	必要なすべての IaaS 前提条件を手動で適用します。
/tmp/do-not-stop-services	<p>IaaS サービスを手動で停止します。</p> <ol style="list-style-type: none"> 1 IaaS Windows サーバにログインします。 2 [スタート] - [管理ツール] - [サービス] を選択します。 3 次の順序でサービスを停止します。 <p>注: IaaS Windows サーバはシャットダウンしないでください。</p> <ol style="list-style-type: none"> a 各 VMware vRealize Automation プロキシ エージェント。 b 各 VMware DEM ワーカー。 c VMware DEM orchestrator。 d VMware vCloud Automation Center サービス。
/tmp/do-not-upgrade-servers	
/tmp/do-not-upgrade-dems	
/tmp/do-not-upgrade-agents	

4 プライマリ vRealize Automation アプライアンス管理コンソールにアクセスして、プライマリ vRealize Automation アプライアンスをアップデートします。

注: 各フラグは削除されるまでアクティブなままになるため、アップグレード後は次のコマンド **rm /<flag_path>/<flag_name>** を実行して選択したフラグを削除してください。例：
rm /tmp/disable-iaas-upgrade