

vRealize Automation 7.4 リリース ノート

更新日: 2019 年 2 月 19 日

vRealize Automation | 2018 年 4 月 12 日 | ビルド 8229492

リリース ノートを定期的に確認して、最新の追加情報や更新情報を入手してください。

リリース ノートの概要

本リリース ノートでは、次のトピックについて説明します。

- [新機能](#)
- [国際化](#)
- [システム要件](#)
- [ドキュメント](#)
- [vRealize Code Stream の使用](#)
- [解決した問題](#)
- [既知の問題](#)

新機能

vRealize Automation 7.4 リリースでは問題の修正、および次の新機能が含まれています。

カスタム申請フォーム デザイナ

新しいカスタム申請フォーム デザイナの導入により、vRealize Automation 7.4 はサービス アーキテクト向けに大幅に最適化されました。新しいカスタム申請フォーム デザイナは、インフラストラクチャおよびアプリケーションのカタログ アイテムの設計に一貫した操作性を提供します。さまざまな業務部門に適応可能な、簡素化された機能豊富なプレゼンテーション層を持つ、汎用性の高いブループリントを容易に作成することができます。すぐに使用可能なカスタム申請フォームにより、インフラストラクチャおよび PaaS ブループリントを XaaS ブループリントにラッピングする必要がないため、過剰なブループリントの作成を抑え、所有コストを削減します。

カスタム申請フォーム デザイナのブループリントを使用することで、アーキテクトは以下のロジックをブループリント申請フォームに適用できます。

- キャンバスでのドラッグアンドドロップによる操作およびカスタム プロパティ
- ブループリント スキーマの使用: ブループリント プロパティ、カスタム プロパティおよびプロファイル
- 生成済みフォームの使用
- カスタマイズ済みフォームの保存、クリア、および取り消し
- カスタムの条件ロジックに基づいてフィールドを動的に表示または非表示
- 外部または内部ロジックに基づいて入力値をオートフィルまたは動的に入力
- vRealize Orchestrator を利用した内部依存関係および外部呼び出しの使用

- 入力値に制限を適用
- 正規表現を使用したカスタム検証の適用
- カスタムのヘルプ テキストとエラー メッセージの適用
- vRealize Orchestrator インベントリ オブジェクトの選択
- ディスク ボリュームや vRealize Orchestrator を組み合わせるなど、複雑なタイプのサポート
- 高度なフォーマットを使用して、カスタム CSS をブループリント申請フォームに適用
- ブループリント定義に対する自動フォーム検証を設計時に実行
- GUI と CLI によるカスタム フォームのエクスポートとインポート

詳細については、「[サービス ブループリントのユーザーへの提供](#)」を参照してください。

OVF からの展開

- OVF または OVA から vSphere ブループリントを展開するための新たなプロビジョニング オプション
- 認証およびプロキシ オプションを維持した OVF の配置場所の URL 指定
- OVF に特化したカスタム プロパティ フォームでの高度な構成オプションをサポート
- イメージ コンポーネント プロファイルを使用したパラメータ化をサポート

詳細については、「[OVF からプロビジョニングするブループリントの構成](#)」を参照してください。

ワークロード配置のため vRealize Operations との連携の強化

vRealize Operations との連携を強化し、展開時のワークロード配置を最適化します。ワークロード配置では vRealize Automation で設定されたストレージ予約ポリシーを考慮するようになりました。

詳細については、[vRealize Automation ストレージ予約ポリシーの機能](#)を参照してください。

インストール、アップグレード、および移行の強化

- vRealize Automation アプライアンス管理の共通ページにおいて証明書管理を統合
- Infrastructure as a Service (IaaS) のサービス ステータスを vRealize Automation アプライアンス管理の [クラスタ] タブで表示
- vRealize Automation のホスト名の変更をサポート
- Python での vra-command の実装をサポート
- ログのローテーションの迅速化、RabbitMQ および HealthBroker サービスでのログのローテーションの追加
- ログ ファイル形式を、ローカル タイムスタンプおよび UTC タイムスタンプを表示するように更新

- vRealize Automation アプライアンス管理に LogInsight エージェントの詳細設定を追加
- vRealize Automation アプライアンス管理から vRealize Orchestrator 構成の起動、停止、リセット オプションをデフォルトで有効化 (CLI は無効)
- vRealize Automation アプライアンス管理タブでのソフトウェア エージェントのアップグレード
- 組み込みの vRealize Orchestrator 向けの自動移行
- 移行完了後に移行パッケージをクリーンアップ
- デフォルトのテナント フィールドまたは管理者フィールドの編集を無効化

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](#)を参照してください。

vRealize Automation のパッチ管理

累積パッチ向けのサポートの追加

- 配信済みおよびデプロイ済みホットパッチの追跡
- クラスタ化された環境における追加ノード上でインストールが必要なパッチに関する警告の表示
- インストール失敗時の再試行
- 過去のパッチへのロールバック
- パッチ履歴の参照

詳細については、[パッチ管理のアクセス](#)を参照してください。

ジャストインタイムのユーザー プロビジョニング

vRealize Automation ではジャストインタイム (JIT) プロビジョニング機能が提供されます。JIT を設定すると、初回ログイン時にユーザーが自動作成され、2 回目以降のログイン時にユーザー属性を同期します。また、vIDM 動的グループ メンバーシップ機能も提供されます。vIDM 動的グループは vRealize Automation に公開され、高度なグループ メンバーシップを使用してカスタム グループ メンバーシップを作成、更新、または削除することができます。カスタム グループは移行することもできます。この機能は API でも提供されます。

詳細については、[ジャストインタイムのユーザー プロビジョニングの設定](#)を参照してください。

停止した申請処理の向上

- 「処理中」で停止した申請を API または CloudClient でキャンセルし、キャンセルされた申請に関連付けられたプロビジョニング済みリソースをクリーンアップ可能
- [申請] タブの新しいフィルタで失敗またはキャンセルした申請を非表示

詳細については、[vRealize CloudClient 4.5](#) を参照してください。

セキュリティ

- メッセージ ボード ポートレットのセキュリティ強化
 - メッセージ ボードに表示可能な URL のホワイトリストを導入
- TLS 1.2 プロトコルは、セキュアな通信を確保するためデフォルトのプロトコルとなりました

詳細については、「[メッセージ ボード ポートレット URL のホワイトリストの作成](#)」を参照してください。

VMware vRealize Orchestrator におけるマルチテナント

vRealize Orchestrator 7.4 ではマルチテナント アーキテクチャが導入されます。

詳細については、『[VMware vRealize Orchestrator におけるマルチテナント](#)』を参照してください。

組み込み vIDM インスタンスを使用した Microsoft NT LAN Manager (NTLM) 認証のサポートは vRealize Automation 7.4 で廃止されました

注：このリリースでは、vRealize Automation 7.3.1 で修正されたすべての問題が含まれています。詳細については、「[vRealize Automation 7.3.1 のリリース ノート](#)」を参照してください。

国際化

vRealize Automation 7.4 は、次の言語で使用できます。

- 英語
- フランス語
- ドイツ語
- スペイン語
- 日本語
- 韓国語
- 簡体字中国語
- 繁体字中国語
- イタリア語
- Russian
- オランダ語
- ブラジルポルトガル語

各製品でサポートされている言語については、『[VMware Product Globalization Guide](#)』を参照してください。

システム要件

サポート対象のホスト オペレーティング システム、データベース、および Web サーバについては、『[vRealize Automation Support Matrix](#)』を参照してください。

ドキュメント

vRealize Automation 7.4 のドキュメントは、VMware Docs の「[VMware vRealize Automation](#)」を参照してください。

インストール

前提条件とインストール手順については、VMware のドキュメントにある「[vRealize Automation のインストール](#)」を参照してください。

アップグレード

一般的なガイダンスについては、VMware のドキュメントにある「[vRealize Automation のアップグレード](#)」を参照してください。

vRealize Automation 6.2.x からのアップグレードの準備

vRealize Production Test Upgrade Assist Tool は、vRealize Automation 6.2.x 環境を分析してアップグレードに関する問題を引き起こす可能性がある機能構成を特定し、使用環境のアップグレード準備ができていることを確認します。このツールおよび関連ドキュメントをダウンロードするには、VMware vRealize Production Test Tool の[製品のダウンロード](#) ページに移動します。

vRealize Code Stream の使用

vRealize Automation 環境で vRealize Code Stream を使用するには、vRealize Code Stream ライセンスを保有している必要があります。

vRealize Automation インストール ウィザードまたは vRealize Automation アプライアンス管理インターフェイスでライセンスを入力することができます。

詳細については、次を参照してください。

- vRealize Code Stream 情報センターの「[vRealize Code Stream Installation](#)」および「[Apply a vRealize Code Stream License to an Appliance](#)」
- vRealize Automation 情報センターの「[Licensing vRealize Code Stream](#)」

解決した問題

- **NEW:** リソース アクションが、vCloud:VM および VC:VirtualMachine 含む外部パラメータをマッピングできない
カスタム アクションの vCloud:VM 入力パラメータが灰色で表示され、XaaS フォーム エディタでスクリプト アクションの入力として選択できません。

- **vRealize Automation 7.3.x へのアップデートを実行しても、Distributed Execution Manager (DEM) または Distributed Execution Manager Orchestrator (DEO) がアップデートされない**

vRealize Automation 7.3.x へのアップデートを実行する際、DEM または DEO の IaaS コンポーネントはデフォルトの格納場所である C:\program files (x86)\vmware\vcac にインストールされている必要があります。これらのコンポーネントがデフォルトの場所にインストールされていない場合、製品のアップデート中にアップデートされません。

- **「Guest and Software Agent Installers」ページにある Java Runtime Environment for Linux のダウンロード リンクに誤りがある**

これらのリンクは、「Linux Software Installers」セクションに表示されています。

- vmware-jre-1.8.0_121-fcs.i586.rpm
- vmware-jre-1.8.0_121-fcs.x86_64.rpm

いずれかのリンクをクリックすると、新しいページが開き、「HTTP Status 404 – Not Found」というエラーが表示されます。

回避策: これらの RPM ファイルをダウンロードするには:

1. リンクをクリックした後に、ブラウザのアドレス フィールドに表示される URL 内のファイル名を次のように置き換えます。

- vmware-jre-1.8.0_121-fcs.i586.rpm を vmware-jre-1.8.0_121-fcs_b31.i586.rpm で置き換えます。
- vmware-jre-1.8.0_121-fcs.x86_64.rpm を vmware-jre-1.8.0_121-fcs_b31.x86_64.rpm で置き換えます。

例:

- https://va-hostname.domain.name/software/download/vmware-jre-1.8.0_121-fcs_b31.x86_64.rpm
- https://va-hostname.domain.name/software/download/vmware-jre-1.8.0_121-fcs_b31.i586.rpm

2. Enter キーを押します。

ブラウザにエラー メッセージが残る場合にも、ファイルは正常にダウンロードされます。

- **サードパーティ製 IP アドレス管理プロバイダに関連付けられた展開済みのオンデマンド NAT ネットワークに NAT ポート転送ルールを追加できない**

サードパーティ製 IP アドレス管理プロバイダに関連付けられた展開済みのオンデマンド NAT ネットワークに対し、[NAT ルールの変更] プロビジョニング後アクションを使用して、NAT ポート転送ルールを追加する際、**【コンポーネント】** フィールドのドロップダウンメニューにデータが表示されず、新規データを受け入れることができません。そのため、新しいルールを追加できません。

- **「仮想サーバの配布設定の定義」の手順にサポートされていない HTTPS トラフィック パターンが含まれている**

[「仮想サーバの配布設定の定義」](#)に次の手順が含まれています。

SSL セッション ID を選択して、次のサポートされている HTTPS トラフィック パターンのいずれかをサポートします。

- SSL パススルー - クライアント -> HTTPS -> LB (SSL を終了) -> HTTPS -> サーバ
- クライアント - HTTP -> LB -> HTTP -> サーバ

「クライアント - HTTP」のパターンを選択しても、システムでは「SSL パススルー - クライアント」トラフィック パターンが使用されます。vRealize Automation では、「クライアント - HTTP」トラフィック パターンはサポートされていません。

- **YAML からインポートされたブループリントに、プロビジョニング後アクションとして [NAT ルールの変更] を実行すると失敗する**

展開上で [NAT ルールの変更] プロビジョニング後アクションを呼び出すと、次のエラーが発生して失敗します: 「Failed to invoke deployment update request [{Could not determine current component state for nat1}]」。これは、展開に関連付けられたブループリントのインポート元の YAML ファイルに、名前フィールドと ID フィールドに異なる値が設定されているオンデマンド NAT ネットワークが含まれている場合に発生します。

- **エンドポイントに特定の vRealize Orchestrator プロパティを追加している場合、vRealize Automation 7.3 または 7.3.1 にアップデートするとエンドポイントが表示されない**

エンドポイントに固有の vRealize Orchestrator カスタム プロパティを設定していると、エンドポイントのアップデートに失敗します。

- **アドバンスド サービス オプションが管理メニューから削除された**
[管理] > [アドバンスド サービス] > [デフォルトの vRO フォルダ] から、テナントのデフォルト ワークフロー フォルダを構成できなくなりました。このオプションは削除されました。

既知の問題

既知の問題には、次のトピックが含まれます。

- [アップグレード](#)
- [構成とプロビジョニング](#)
- [以前の既知の問題](#)

アップグレード

- **vRealize Automation アプライアンスをアップグレードすると、vRealize Orchestrator サービスのデータベース内に重複に関するエラーが発生することがある**

ユーザー インターフェイスに表示されるエラーは、次のようになります。

- アップデートのインストールに失敗しました (インストール後のスクリプトの実行時エラー)
- 仮想アプライアンスのチェック: 完了
- インストール前: 完了
- インストール後: 完了
- アップデートに失敗しました (コード 0-2) 。/opt/vmware/var/log/vami のログを確認するか、後でアップデートを再試行してください。

/var/log/bootstrap/postupdate.log に表示されるエラーには、次のものが含まれます。

不要なアイテムを削除して重複を解決します。

Orchestrator データベースに重複するエントリが見つかりました:

リソース要素の重複:

- ID が「<UUID>」、名前が「ko.properties」の 1 つのアイテム
 - ID が「<UUID>」、名前が「fr_FR.properties」の 1 つのアイテム
 - ID が「<UUID>」、名前が「zh_CN.properties」の 1 つのアイテム
- (その他多数)

回避策: 7.4 にアップグレードする前に、回避策を実行してください。ナレッジベースの記事 [KB54982](#) を参照してください。

- **NEW: vRO 用の setenv.sh ファイルによるカスタム アップデートがアップグレード後にオーバーライドされる**
アップグレード後に必要に応じて値を手動で更新し、VCO サーバを再起動して変更を適用します。
- **NEW: アップグレードまたは SSO の admin パスワードを変更すると vRA ポータルが停止し、サービスを利用できなくなる**
ユーザーが vRO コントロール センターでカスタム vRO 認証プロバイダを設定している場合、アップグレードまたは SSO の admin パスワードを変更すると、値がデフォルト値 (vsphere.local\vcadmins) でオーバーライドされます。

回避策: ナレッジベースの記事 [KB56627](#) を参照してください。

構成とプロビジョニング

- **401 Unauthorized エラーが表示される**
vRealize Automation API は、VMware Identity Manager (vIDM) API を呼び出します。vIDM は外部/サードパーティ ID プロバイダまたはサードパーティ ID プロバイダに対する API 認証をサポートしていないため、サードパーティ ID プロバイダが使用されていると認証に失敗します。しかし、サードパーティ ID プロバイダは、vIDM の Just-in-Time (JIT) ユーザー プロビジョニング機能を有効にし設定するための、前提条件となっています。そのため、JIT ユーザーは vRealize Automation API を使用した認証を実行できません。

回避策: OAuth2 パスワードの認証タイプを使用する API 認証には、vIDM に以下のパスワード認証方法のいずれかが必要です。

- コネクタ パスワード認証
- コネクタ (送信) パスワード認証
- ローカル ユーザーのパスワード
- ACC のパスワード

認証にサードパーティ ID プロバイダが設定されている場合でも、これらのパスワードのいずれかが存在する必要があります。この問題を回避するために、ローカル ユーザーは vRealize Automation API を使用して認証することができます。

- **再開要求に失敗する**
以下の場合、再開要求に失敗することがあります:
 1. マシンの割り当てに成功したものの、プロビジョニングに失敗した場合、コンポーネント申請で再開要求が失敗します。この問題は、無効な割り当て情報でマシンを再プロビジョニングするときに発生します。
 2. ネストされたブループリントで再開要求が失敗します。コンポーネント申請を再作成するときに、再開要求の操作で内部ブループリントの申請を正しく初期化できません。

ん。
回避策: なし

- **_asd.requestInfo_~requestedBy または _asd.requestInfo_~requestedFor にバインドされた XaaS フィールドがコンポーネント ブループリントに存在する場合、正しく計算されない**
asd.requestInfo~requestedFor または _asd.requestInfo_~requestedBy にバインドされた XaaS フィールドで入力値に制約がある場合、XaaS ブループリントを編集および保存した最終ユーザーで計算されます。

回避策:

1. バインドされた XaaS フィールドから入力値の制約を解除します。
 2. XaaS フィールドをデフォルト値に設定し、
asd.requestInfo~requestedBy~principalId にバインドします。
 3. XaaS コンポーネントを削除し、複合ブループリント キャンバスにドラッグし直します。
 4. 複合ブループリントを保存します。
- **カタログ アイテム申請を送信した直後にキャンセルすると、処理がキャンセル中の状態で停止して表示される**
システムが申請完了イベントを呼び出していないため、申請がキャンセル中の状態で停止します。

回避策: カatalog申請を送信した直後にキャンセルしないようにします。申請の状態が処理中になるまで待機してください。

- **コネクタの認証アダプタの編集にログインが要求される場合がある**
管理者は、vRealize Automation コンソールへログインしてから 30 分以内であれば、コンソールを使用してディレクトリに対応するコネクタに認証アダプタを設定することができます。ログインから 30 分経過した後で管理者が設定を行うと、ログイン ページが表示され、認証が要求されます。

回避策: 管理者の認証情報で、再度コンソールにログインします。

- **vRealize Automation アプライアンス管理に正常にログインした後、再度ログインを求められる**
vRealize Automation アプライアンス管理で [パッチ管理] をクリックすると、認証情報を再度入力するよう求められます。

回避策: root ユーザーとして再度認証し、パッチ管理ページを開きます。

- **プライマリ ドメイン コントローラが利用できない場合、ログインに非常に時間がかかる、またはログインに失敗する**
プライマリ ドメイン コントローラへ接続しようとして失敗すると、vIDM はセカンダリ ドメイン コントローラに接続します。vIDM はセカンダリ ドメイン コントローラに接続する前に必ずプライマリ ドメイン コントローラに接続するため、ログイン要求の処理に遅延が生じます。これによってリクエストが停滞し、システムが遅くなります。

回避策: ナレッジベースの記事 [KB52840](#) を参照してください。

- **vRealize Automation 7.3 から 7.4 へ正常に移行した後で、Azure リソース上での一部の操作においてエラー メッセージを受信する**

vRealize Automation 7.3 から 7.4 へ正常移行した後、移行後の Azure リソースで再起動などの一部の操作が断続的に失敗します。vRealize Orchestrator ワークフローは正常に動作しているにもかかわらず、このような操作の失敗が vRealize Automation で報告されます。

回避策: 新たにコマンド プロンプトを開き、次のコマンドを実行してオーケストレーション ゲートウェイと shindig-ui プロパティのタイムアウトの値を増加するための編集を行い、vcac-server を再起動します。

```
1.# cd /var/lib/vcac/server/webapps/vcac/WEB-INF/classes/
```

```
2.# cp shindig.properties shindig.properties.`date +%m%d%Y`
```

```
3.# vi shindig.properties
```

```
4. 次のように編集: shindig.http.client.read-timeout-ms=150000
```

```
5.# cd /usr/lib/vcac/server/webapps/o11n-gateway-service/WEB-INF/classes/META-INF/spring/root
```

```
6.# cp o11n-gateway-service-context.xml o11n-gateway-service-context.xml.`date +%m%d%Y`
```

```
7.# vi o11n-gateway-service-context.xml
```

```
8 次のように編集: to 150000
```

```
9.# service vcac-server restart
```

- **仮想アプライアンスが 1 台以上利用できないと vRealize Automation 健全性サービスで複数のエラーが表示される**

仮想アプライアンスが 1 台以上利用できないと、健全性サービスにエラーが生じます。一部のエラーのせいで、その他に発生しているエラーが覆い隠されてしまう可能性があります。

回避策: 機能しないノードをリストアするか、クラスタからノードを削除し、隠れていたエラーを特定します。

- **vRealize Automation アプライアンス管理の [Xenon] タブで [起動]、[停止]、[再起動] ボタンをクリックしても、サービスに影響を及ぼすことができない**

クラスタ化された環境の場合、vRealize Automation アプライアンス管理の [Xenon] タブにおいて、開始、停止、再起動の操作をレプリカ ノードで実行すると、サービスに影響を及ぼすことができません。

回避策: Xenon サービスの操作は、マスター ノードのみで実行する必要があります。

- **ブラウザを立ち上げ vRealize Automation アプライアンス管理を開くと、自己署名証明書に関するエラー メッセージが表示され、次の処理に進めなくなる**

HTTP Strict Transport Security (HSTS) が有効なブラウザでは、自己署名証明書が使用されたサイトへのアクセスができません。

回避策: ナレッジベースの記事 [KB53533](#) を参照してください。

- **vRealize Automation 7.4 への IaaS 自動アップグレードが実行された後で、Manager Service の自動フェイルオーバー モードが有効になる**

vRealize Automation 7.3 または 7.3.1 から 7.4 へアップデートまたは移行する際に、アップデートまたは移行前に自動フェイルオーバーを意図的に無効にしていた場合、vRealize Automation 7.4 への IaaS 自動アップグレード実行中に自動フェイルオーバー機能が有効になります。

Manager Service の自動フェイルオーバー モードを無効にするには、次のいずれかのタスクを実行します。

- Manager Service の自動フェイルオーバーの無効化

詳細については、vRealize Automation のインストールの「[Manager Service 自動フェイルオーバーの有効化](#)」を参照してください。

- レガシー インストーラを使用した手動による IaaS のアップグレード

詳細については、vRealize Automation 6.2.5 から 7.4 へのアップグレードの「[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](#)」を参照してください。

- **Azure 仮想マシン向けのプロビジョニング後処理のパブリック IP アドレス管理がタイムアウトになる**

vRealize Orchestrator から Azure 仮想マシンの現在のパブリック IP アドレスおよび利用可能なパブリック IP アドレスを取得するのに非常に長い時間がかかります。vRealize Automation で次のエラー メッセージが表示され、処理がタイムアウトになります。
「vCenter Orchestrator サーバへの接続でタイムアウトが発生しました。」

回避策:

vRealize Automation で次の手順を実行し、タイムアウト設定の時間を増やします。

1. それぞれの vRealize Automation アプライアンス ホストで SSH を使用してコマンドプロンプトを開き、root としてログインします。
2. 次のコマンドを実行して、すべてのノードで vRealize Automation サービスを停止します: `service vcac-server stop`
3. ディレクトリを `/etc/vcac/` に変更し、テキスト エディタで `vcac.properties` ファイルを開きます。
4. `vco.socket.timeout.millis` プロパティのタイムアウト値を 300000 に増やします。
例: `vco.socket.timeout.millis= 300000`. 設定の単位はミリ秒です。
5. `vcac.properties` ファイルを保存して閉じます。
6. ディレクトリを `/var/lib/vcac/server/webapps/vcac/WEB-INF/classes/` に変更します。
7. 次のコマンドを実行して、`shindig.properties` ファイルのバックアップを取得します: `cp shindig.properties shindig.properties.bak`
8. `shindig.properties` ファイルをテキスト エディタで開き、ファイルで `shindig.http.client.read-timeout-ms=70000` と記述された行を特定します。
9. `shindig.http.client.read-timeout-ms` の値を 300000 に増やします。例: `shindig.http.client.read-timeout-ms=300000`.
10. `shindig.properties` ファイルを保存して閉じます。
11. ディレクトリを `/etc/vcac/` に変更し、テキスト エディタで `setenv-user` ファイルを開きます。

12. ファイルに次の行を追加します。VCAC_OPTS="\$VCAC_OPTS - Dclient.system.socket.timeout=300000"

13. setenv-user ファイルを保存して閉じます。

14. 次のコマンドを実行して、すべてのノードで vRealize Automation サービスを開始します: `service vcac-server start`

- **クラスタ化された vRealize Automation 環境でレプリカ アプライアンスで CPU 使用率が 100% になることがある**

クラスタ化された vRealize Automation 環境では、複数の socat 処理によりレプリカ アプライアンスで CPU 使用率が 100% になることがあります。

回避策: ナレッジベースの記事 [KB54143](#) を参照してください。

- **Active Directory の同期に失敗する**

1.Active Directory に 200,000 を超えるユーザーと、60,000 を超えるグループが含まれている。

2.サブドメイン (例: subdomain1.abc.com) ではなくトップ レベル ドメイン (例: abc.com) が同期に使用されている。

症状:

コネクタ (場所: CAFE の /var/log/vmware/horizon フォルダ) ログに以下のエラーが記録されます。

2018-03-23 18:01:22,122 ERROR (SimpleAsyncTaskExecutor-168)

[3259@JNJ;local@JNJ;127.0.0.1] com.vmware.horizon.directory ldap.LdapConnector - Problem reading from LDAP directory:

javax.naming.OperationNotSupportedException: [LDAP: error code 12 - 00002040: SvcErr: DSID-03140395, problem 5010 (UNAVAIL_EXTENSION), data 0

1.各組織単位 (OU) での Active Directory の同期は、1 OU あたり最大 120,000 ユーザーおよび 40,000 グループを許可して実行する必要があります。

2.セーフガードは [同期設定] > [セーフガード] ページで無効にする必要があります。

以前の既知の問題

以前の既知の問題のリストを表示するには、[ここ](#)をクリックします。