



Red Hat Satellite 6.4

Red Hat Satellite のアップグレードおよびアップ デート

Red Hat Satellite Server および Capsule Server のアップグレードおよびアップデー
ト

Red Hat Satellite 6.4 Red Hat Satellite のアップグレードおよびアップデート

Red Hat Satellite Server および Capsule Server のアップグレードおよびアップデート

Red Hat Satellite Documentation Team
satellite-doc-list@redhat.com

法律上の通知

Copyright © 2018 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

概要

本ガイドでは、Red Hat Satellite Server、Capsule Server、およびホストのアップグレードおよびアップデートについて説明します。

目次

第1章 アップグレードの概要	3
1.1. 前提条件	3
1.2. アップグレードパス	4
1.3. アップグレードの進捗の追跡	5
第2章 SATELLITE SERVER のクローン	6
2.1. クローン作成プロセスの概要	6
2.2. 前提条件	6
2.3. PULP データの考慮事項	7
2.4. SATELLITE SERVER のクローン	7
2.4.1. クローン作成のためのソースサーバーの準備	7
2.4.2. ターゲットサーバーのクローン作成	8
第3章 RED HAT SATELLITE のアップグレード	11
3.1. SATELLITE SERVER のアップグレード	11
3.1.1. 接続している Satellite Server のアップグレード	13
3.1.2. 切断されている Satellite Server のアップグレード	15
3.2. CAPSULE SERVER のアップグレード	19
3.3. SATELLITE クライアントのアップグレード	22
3.4. アップグレード後のタスク	24
3.4.1. Discovery のアップグレード	24
3.4.1.1. Satellite Server での Discovery のアップグレード	24
3.4.1.2. Capsule Server での Discovery のアップグレード	25
3.4.1.3. サブネットにテンプレート Capsule があることの確認	25
3.4.2. virt-who のアップグレード	26
3.4.3. 以前のバージョンの Satellite Tools リポジトリの削除	26
第4章 PUPPET のアップグレード	27
4.1. アップグレードパス	27
4.2. PUPPET モジュールのテスト	28
4.3. アップグレードの準備	28
第5章 SATELLITE SERVER、CAPSULE SERVER、およびコンテンツホストのアップデート	30
5.1. SATELLITE SERVER のアップデート	30
5.2. CAPSULE SERVER のアップデート	31
5.3. コンテンツホストのアップデート	32

第1章 アップグレードの概要

本章では、Red Hat Satellite 6.4 の前提条件、および利用可能なアップグレードパスを説明します。現在の Red Hat Satellite 6.4 インストールをアップグレードする前にお読みください。

このガイドでは、「アップデート」、「アップグレード」、「マイグレーション」を以下の意味で使用します。

アップグレード

y-stream を基準にして、Satellite Server および Capsule Server のインストールを次のリリースに上げるプロセスです (たとえば Satellite 6.3 から Satellite 6.4)。

アップデート

z-stream を基準にして、Satellite Server および Capsule Server のインストールを次のリリースに上げるプロセスです (たとえば Satellite 6.3.0 から Satellite 6.3.1)。

移行

既存の Satellite インストールを、別の Red Hat Enterprise Linux サーバーに移行するプロセスです。

Red Hat カスタマーポータル [の Red Hat Satellite Upgrade Helper](#) では、対話式のアップグレード手順がご利用になれます。このアプリケーションが提供する手順は、お使いのバージョン番号によって異なるため、アップグレードパスに適したアップグレード手順と、既知の問題を回避する手順が提示されます。詳細は、カスタマーポータル [の「Satellite Upgrade Helper」](#) をご覧ください。



重要

Red Hat Satellite Server と Capsule Server で、y-stream バージョンを一致させる必要があります。たとえば、Satellite Server 6.3 は Capsule Server 6.4 と互換性がなく、Satellite Server 6.4 は Capsule Server 6.3 と互換性がありません。Satellite Server と Capsule Server のバージョンが一致しないと Capsule Server が失敗し、メッセージも表示されません。ただし、z-stream バージョンについては、Capsule Server が Satellite Server より 1 つだけ古い場合はサポートされます。たとえば、Satellite 6.3.1 の Satellite Server は、6.3.0 の Capsule Server と互換性があります。

1.1. 前提条件

Satellite 6.4 へのアップグレードは、Satellite インフラストラクチャー全体に影響します。アップグレード前に以下を完了してください。

- [『Red Hat Satellite 6.4 リリースノート』](#) を参照してください。
- このガイドでは、アップグレードプロセスとその影響について説明します。
- アップグレードパスの計画を立てます。詳細は [「アップグレードパス」](#) を参照してください。
- Red Hat Satellite 6.4 は Puppet 5 を使用します。Puppet 4 から Puppet 5 への移行のみがサポートされます。Satellite 6.4 にアップグレードする前に、Puppet 4 をアップグレードする必要があります。Puppet 4 へのアップグレードに関する情報は、[『Satellite 6.3 Red Hat Satellite のアップグレードおよびアップデート』](#) ガイドの [「Puppet のアップグレード」](#) のセクションを参照してください。
- 必要な Satellite ダウンタイムの計画を立てます。アップグレードプロセスの実行時は、Satellite サービスをシャットダウンする必要があります。

- ストレージ要件の計画を立てます。Red Hat Satellite 6.4 では、ストレージ要件が変更になりました。詳細は『[オンラインネットワークからの Satellite Server のインストール](#)』の「[ストレージ要件およびガイドライン](#)」を参照してください。
- Satellite Server および全 Capsule Server をバックアップします。詳細は、『[Red Hat Satellite 6.3 管理](#)』ガイドの「[Satellite Server および Capsule Server のバックアップと復元](#)」を参照してください。



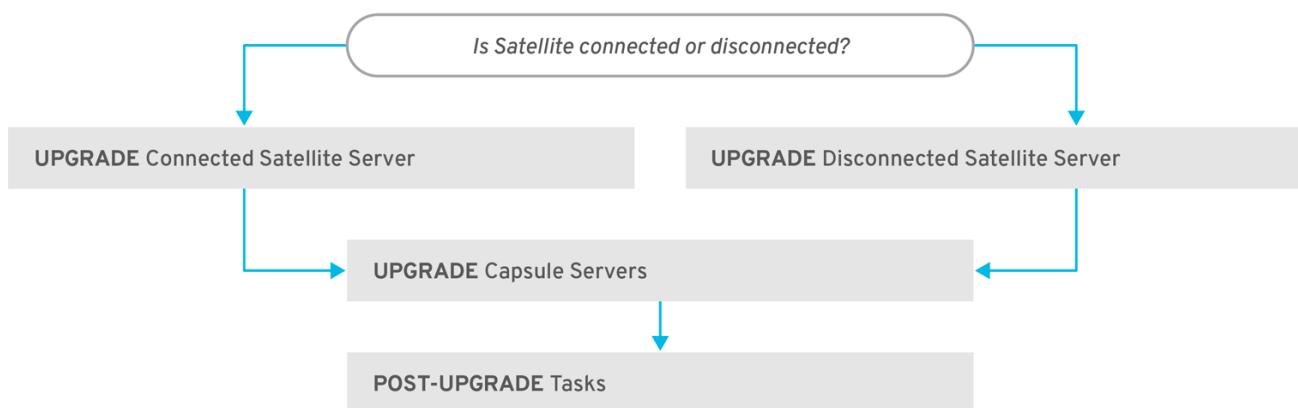
警告

設定ファイルをカスタマイズする場合は、手動または Hiera などのツールを使用します。このようなカスタマイズを行うと、アップグレード時または更新時にインストールスクリプトを実行する際に上書きされます。satellite-installer スクリプトで `--noop` オプションを使用すると、変更をテストできます。詳細は、ナレッジベースソリューションの「[How to use the noop option to check for changes in Satellite config files during an upgrade](#)」を参照してください。

1.2. アップグレードパス

Red Hat Satellite 6.3 から Red Hat Satellite 6.4 にアップグレードできます。以前のバージョンの Satellite Server と Capsule Server は、先に Satellite 6.3 にアップグレードする必要があります。詳細は、Satellite 6.3 『[Red Hat Satellite のアップグレードおよびアップデート](#)』ガイドを参照してください。

図1.1 Satellite 6.4 アップグレードパスの概要



SATELLITE_461763_0718



警告

ベータから GA バージョンへのアップグレードはサポートされていません。

以下は、Satellite 6.4 にアップグレードする手順の概要です。

1. 既存の Satellite Server をクローンします。詳細は、「[2章 Satellite Server のクローン](#)」を参照してください。
2. Satellite Server と全 Capsule Server を Satellite 6.4 にアップグレードします。詳細は「[Satellite Server のアップグレード](#)」を参照してください。
3. すべての Satellite クライアントで、Satellite Tools をアップグレードします。詳細は「[Satellite クライアントのアップグレード](#)」を参照してください。

自己登録 Satellite

自己登録の Satellite をアップグレードすることはできません。自己登録の Satellite は、Red Hat コンテンツ配信ネットワーク (CDN) に移行すればアップグレードを実行できます。自己登録の Satellite を CDN に移行する方法は、『[Satellite 6.3 の Red Hat Satellite のアップグレードおよびアップデート](#)』ガイドの「[自己登録 Satellite の移行](#)」を参照してください。

1.3. アップグレードの進捗の追跡

アップグレード時間は長くなるため、通信セッションの中断と再接続を可能にする **screen** などのユーティリティを使用します。これにより、コマンドシェルに接続し続けなくてもアップグレードの進捗が確認できるようになります。screen コマンドの使用方法は、**Red Hat ナレッジベース** の「[How do I use the screen command?](#)」を参照してください。また、**screen** の man ページでも、詳細を確認できます。

アップグレードコマンドを実行しているコマンドシェルへの接続がなくなった場合は、`/var/log/foreman-installer/satellite.log` のログで、プロセスが完全に終了したかどうかを確認できます。

第2章 SATELLITE SERVER のクローン

Satellite Server のアップグレード時に、アップグレード中にデータが損失されないように、オプションで Satellite のクローンを作成することができます。アップグレードが完了したら、Satellite Server の以前のバージョンの使用を停止できます。

以下の手順を使用して、Satellite インスタンスのクローンを作成し、環境を保存してアップグレードの準備を整えます。

Satellite クローンツールは、Capsule Server から Red Hat Enterprise Linux 7 への移行をサポートしません。代わりに、既存の Capsule Server のバックアップを作成し、Red Hat Enterprise Linux 7 に Capsule Server を復元してから再設定します。

用語

次の用語を理解するようにしてください。

ソースサーバー: クローンするサーバー

ターゲットサーバー: ファイルをコピーし、ソースサーバーのクローンを作成する先の新規サーバー

2.1. クローン作成プロセスの概要

1. ソースサーバーをバックアップします。
2. ソースサーバーからターゲットサーバーにクローンを作成します。
3. ソースサーバーの電源を切断します。
4. 新規ホスト名とターゲットサーバーの IP アドレスが一致するように、ターゲットサーバー上のネットワーク設定を更新します。
5. コンテンツホストと Capsule で **goferd** を再起動して、接続を更新します。
6. 新規ターゲットサーバーをテストします。

2.2. 前提条件

Satellite Server のクローンを作成するには、以下のリソースが用意されていることを確認します。

- ターゲットサーバーとして使用する Red Hat Enterprise Linux 7 サーバーの最小インストール。Red Hat Enterprise Linux 7 ソフトウェアグループやサードパーティーのアプリはインストールしないでください。また、『[オンラインネットワークからの Satellite Server のインストール](#)』の「[インストールのために環境を準備](#)」に記載されている仕様すべてに準拠していることを確認してください。
- **katello-backup** スクリプトを使用して作成した Satellite バージョン 6.1、6.2 または 6.3 のバックアップ。バックアップは、Pulp データありでも、なしでも使用できます。
- ターゲットサーバーの Satellite のサブスクリプション

クローンを開始する前に、以下の条件を満たしていることを確認してください。

- ターゲットサーバーが分離ネットワーク上にあること。これにより、Capsule Server とホスト間で必要のない通信を回避できます。

- ソースサーバーからのバックアップファイルすべてを格納できるだけの容量があるターゲットサーバー

2.3. PULP データの考慮事項

Pulp データを含めずに、Satellite Server のクローンを作成できます。ただし、クローンした環境を機能させるためには、Pulp データが必要です。ターゲットサーバーに、Pulp データがない場合には、Satellite は完全に機能しません。

Pulp データをターゲットサーバーに転送するには、2 つのオプションがあります。

- Pulp データを含むバックアップを使用したクローン作成
- Pulp データなしのバックアップを使用してクローンを作成し、ソースサーバーから手動で `/var/lib/pulp` をコピーします。

`pulp_data.tar` ファイルが 500 GB 以上の場合や、`pulp_data.tar` ファイルが 100 GB 以上で、NFS など、速度の遅いストレージシステムを使用している場合には、展開時にメモリーエラーが発生する可能性があるので、バックアップに `pulp_data.tar` を含めないようにしてください。ソースサーバーからターゲットサーバーに `pulp_data.tar` ファイルをコピーするようにしてください。

Pulp データなしでバックアップする方法

「[Satellite Server のクローン](#)」の順の内容に従います。ただし、Pulp データが含まれるクローンを作成する手順を、以下の手順に置き換えてください。

1. MongoDB と PostgreSQL の両データベースをアクティブにし、Pulp データは除外してバックアップを実行します。

```
# katello-service stop
# katello-service start --only mongod,postgresql
# katello-backup --skip-pulp-content /var/backup --assumeyes
```

2. Satellite Service をすべて停止および無効にします。

```
# katello-service stop
# for i in $(katello-service list | awk '{print $1}' | grep service); \
do systemctl disable $i ;done
```

3. Pulp データをターゲットサーバーにコピーします。

```
# rsync --archive --partial --progress --compress \
/var/lib/pulp target_server.example.com:/var/lib/pulp
```

「[ターゲットサーバーのクローン作成](#)」に進みます。

2.4. SATELLITE SERVER のクローン

以下の手順を使用して、Satellite Server のクローンを作成します。この手順の一部として大量のデータをコピーして転送する必要があるため、完了までにかかり時間がかかる可能性がある点に注意してください。

2.4.1. クローン作成のためのソースサーバーの準備

ソースサーバーで、以下の手順を実行します。

1. Satellite サブスクリプションのプール ID を確認します。

```
# subscription-manager list --consumed \
--matches 'Red Hat Satellite'|grep "Pool ID:"|awk '{print $3}'
```

2. Red Hat Satellite のサブスクリプションを削除します。

```
# subscription-manager remove --serial=$(subscription-manager list \
--consumed \
--matches 'Red Hat Satellite'|grep "Serial:"|awk '{print $2}')
```

3. Pulp データのサイズを判断します。

```
# du -sh /var/lib/pulp/
```

4. Pulp データが 500 GB 未満の場合は、MongoDB と PostgreSQL のデータベースをアクティブにし、Pulp データを含めてバックアップを実行します。Pulp データが 500 GB 以上の場合には、以下の手順は省略して、続行する前に「[Pulp データの考慮事項](#)」の手順を実行してください。

```
# katello-service stop
# katello-service start --only mongod,postgresql
# katello-backup /var/backup --assumeeyes
```

5. Satellite サービスをすべて停止および無効にします。

```
# katello-service stop
# for i in $(katello-service list| awk '{print $1}'|grep service); \
do systemctl disable $i ;done
```

「[ターゲットサーバーのクローン作成](#)」に進みます。

2.4.2. ターゲットサーバーのクローン作成

サーバーのクローンを作成するには、ターゲットサーバーで以下の手順を実行してください。

1. **satellite-clone** ツールはデフォルトで、'/backup/' をバックアップフォルダーとして使用します。別のフォルダーにコピーする場合には、**/etc/satellite-clone/satellite-clone-vars.yml** ファイルの **backup_dir** 変数を更新してください。
2. **/backup** フォルダーにバックアップファイルを配置します。共有ストレージをマウントするか、**candlepin.dump**、**config_files.tar.gz**、**foreman.dump** のファイルと、**mongo_dump** フォルダーをコピーすることができます。
3. ソースサーバーの電源を切断します。
4. 以下のコマンドを入力してカスタマーポータルに登録して、サブスクリプションのアタッチし、必要なサブスクリプションだけを有効化します。

```
# subscription-manager register your_customer_portal_credentials
# subscription-manager attach --pool=pool_ID
# subscription-manager repos --disable=*
```

```
# subscription-manager repos \
--enable=rhel-7-server-rpms \
--enable=rhel-server-rhsc1-7-rpms \
--enable=rhel-7-server-satellite-maintenance-6-rpms \
--enable=rhel-7-server-satellite-6.3-rpms
```

5. **satellite-clone** パッケージをインストールします。

```
# yum install satellite-clone
```

satellite-clone ツールをインストールした後に、独自のデプロイメントに合わせて **/etc/satellite-clone/satellite-clone-vars.yml** ファイルで設定を調節します。

6. **satellite-clone** ツールを実行します。

```
# satellite-clone
```

7. DHCP、DNS、TFTP、およびリモート実行サービスを再設定します。ソースの Satellite Server と競合しないように、クローンプロセスにより、ターゲットの Satellite Server でこのサービスが無効になります。
8. Satellite web UI で DHCP、DNS、TFTP を再設定し、有効にします。詳細は『オンラインネットワークからの **Satellite Server** のインストール』の「[Satellite Server での DNS、DHCP および TFTP の設定](#)」を参照してください。
9. リモート実行を有効にします。

```
# satellite-installer --scenario satellite \
--enable-foreman-plugin-remote-execution \
--enable-foreman-proxy-plugin-remote-execution-ssh
```

10. ユーザー名 **admin** とパスワード **changeme** で、Satellite Web UI にログインします。すぐに、管理者パスワードを変更して認証情報のセキュリティを確保します。
11. 正しい組織が選択されていることを確認します。
12. **コンテンツ > サブスクリプション** に移動して、**マニフェストの管理**をクリックします。
13. **更新** ボタンをクリックして **終了** をクリックし、サブスクリプションの一覧に戻ります。
14. 利用可能なサブスクリプションが正しいことを確認します。
15. **/usr/share/satellite-clone/logs/reassociate_capsules.txt** ファイルの説明に従い、Capsules とライフサイクル環境の間の関係を復元します。
16. ターゲットサーバーの IP アドレスと新規ホスト名が一致するように、DNS など、ネットワーク設定を更新します。**satellite-clone** ツールにより、ホスト名をソースサーバーのホスト名に変更します。ホスト名を別のものに変更する場合には、**satellite-change-hostname** ツールを使用してください。詳細は、『**Red Hat Satellite の管理**』の「[Satellite または Capsule Server の名前変更](#)」を参照してください。
17. ソースサーバーが **virt-who** デーモンを使用する場合は、ターゲットサーバーにインストールして設定します。ソースサーバーのディレクトリーにある **virt-who** 設定ファイルをすべて、ターゲットサーバーにある同じディレクトリーに移動します。詳細情報は、『**仮想インスタンスガイド**』の「[Virt-who インストールと設定の概要](#)」を参照してください。

18. 登録したすべてのコンテンツホストと Capsule で、**goferd** を再起動します。

```
systemctl restart goferd
```

以下の章を使用してアップグレードを行った後に、安全にソースサーバーの使用を停止することができます。

第3章 RED HAT SATELLITE のアップグレード



警告

高可用性設定に Satellite 6 をインストールしている場合は、Satellite 6.4 にアップグレードする前に Red Hat サポートにご連絡ください。

本章を参照して、既存の Red Hat Satellite 環境を Red Hat Satellite 6.4 にアップグレードします。

本章には以下が含まれます。

- [「Satellite Server のアップグレード」](#)
- [「Capsule Server のアップグレード」](#)
- [「アップグレード後のタスク」](#)

アップグレードを行う前に [「前提条件」](#) を参照してください。

3.1. SATELLITE SERVER のアップグレード

このセクションでは、Satellite Server を 6.3 から 6.4 にアップグレードする方法を説明します。Red Hat Satellite Server 6.3 であれば、どのマイナーバージョンからでもアップグレードできます。

作業を開始する前に

- Satellite Server をアップグレードする前に、ファイアウォールの設定を確認して更新してください。詳細は『[オンラインネットワークからの Satellite Server のインストール](#)』の [「ポートおよびファイアウォールの要件」](#) を参照してください。
- カスタマーポータルまたは Satellite Web UI からマニフェストを削除しないでください。削除すると、コンテンツホストからエンタイトルメントがすべて削除されます。
- アップグレードする前に、全 Foreman フックのバックアップを作成して、その後フックを削除します。アップグレードが完了し、Satellite が動作しているのを確認したら、フックを復元します。
- デフォルトテンプレートを変更する場合は、そのテンプレートのクローンを作成するか、エクスポートを行って、ファイルをバックアップします。クローンを作成することで、今後の更新またはアップグレードが上書きされなくなるため、推奨されるのはクローン作成です。テンプレートの変更の有無を確認するには、アップグレード前に [履歴](#) を確認するか、アップグレード後に監査ログで変更を表示できます。Web UI で [モニター](#) > [監査](#) に移動し、テンプレートを検索すると、変更履歴を確認できます。エクスポートを使用する場合は、エクスポートしたテンプレートと、デフォルトテンプレートを比較し、手動で変更を適用して変更を復元します。
- `/root/.hammer/defaults.yml` ファイルが Satellite Server にある場合には、アップグレード前にこのファイルのバックアップを作成してから、`/root/.hammer/defaults.yml` を削除してください。このファイルは、アップグレードの完了後に Satellite が機能していることが分かるまで復元しないでください。Red Hat の [バグ 1632768](#) が解決されるまで、この回避策を適用する必要があります。

Capsule に関する留意事項

- Red Hat Satellite Capsule 6.4 リポジトリが Satellite Server で有効になっており、同期されているのを確認します。[Red Hat のバグ 1305040](#) が解決され、Capsules にこれらの変更が同期されるまでしばらくお待ちください。
- Capsule Server のベースオペレーティングシステム、または Capsule Server リポジトリへの更新をコンテンツビューで管理する場合は、更新したコンテンツビューを公開する必要があります。



警告

カスタムの証明書を実装している場合は、`/root/ssl-build` ディレクトリーと、カスタム証明書に関連するソースファイルを作成したディレクトリーのコンテンツを保持する必要があります。

アップグレード時にこのファイルを保持できないと、アップグレードは失敗します。ファイルを削除してしまった場合は、アップグレードを進めるためにバックアップから復元する必要があります。

BASH シェルの設定

BASH シェルには、ハッシュテーブルのバイナリーの場所が保存されます。アップグレード時に **katello-service** スクリプトの場所が変更になりますが、BASH にはこの変更が登録されないため、変更後にこのスクリプトを呼び出すと、**foreman-maintain** が失敗します。

- オプション: インストーラーの完了後に **katello-service** が有効になるように、アップグレードする前に、BASH シェルで **checkhash** オプションを一時的に設定できます。BASH シェルで以下のようなコマンドを実行します。

```
# shopt -s checkhash
```

- アップグレードに成功または失敗した後に、現在実行しているすべての BASH シェルで、以下のコマンドを実行します。

```
# hash -d katello-service 2> /dev/null
```

アップグレードシナリオ

- Red Hat コンテンツ配信ネットワークに接続している Satellite Server をアップグレードする場合は「[接続している Satellite Server のアップグレード](#)」に進みます。
- Red Hat コンテンツ配信ネットワークに接続していない Satellite Server をアップグレードする場合は「[切断されている Satellite Server のアップグレード](#)」に進みます。

自己登録の Satellite をアップグレードすることはできません。自己登録の Satellite は、Red Hat コンテンツ配信ネットワーク (CDN) に移行すればアップグレードを実行できます。自己登録の Satellite を CDN に移行する方法は、『[Satellite 6.3 の Red Hat Satellite のアップグレードおよびアップデート](#)』ガイドの「[自己登録 Satellite の移行](#)」を参照してください。

3.1.1. 接続している **Satellite Server** のアップグレード

Satellite Server を Red Hat 配信ネットワークに接続している場合は、以下の手順を行います。



警告

設定ファイルをカスタマイズする場合は、手動または Hiera などのツールを使用します。このような変更を行うと、アップグレード時または更新時にインストールスクリプトを実行する際に上書きされます。satellite-installer スクリプトで **--noop** オプションを使用すると、変更をテストできます。詳細は、ナレッジベースソリューションの「[How to use the noop option to check for changes in Satellite config files during an upgrade](#)」を参照してください。

Satellite Server のアップグレード

1. バックアップを作成します。
 - 仮想マシンで、スナップショットを作成します。
 - 物理マシンで、バックアップを作成します。
バックアップに関する詳細は、『**Red Hat Satellite 6.3 管理**』ガイドの「[Satellite Server および Capsule Server のバックアップと復元](#)」を参照してください。
2. DNS と DHCP の設定ファイルである `/etc/zones.conf` と `/etc/dhcp/dhcpd.conf` のバックアップをそれぞれ作成します。インストーラーでは、ドメインまたはサブネットが 1 つしかサポートされないため、バックアップからの変更の復元が必要になる場合があります。
3. DNS または DHCP の設定ファイルを手動で編集し、変更を上書きしない場合は、以下のコマンドを実行します。

```
# satellite-installer --foreman-proxy-dns-managed=false \  
--foreman-proxy-dhcp-managed=false
```

4. Satellite Web UI で、**ホスト > 検出された** ホストに移動します。検出されたホストページで、検出したホストの電源を切って削除します。**組織の選択** メニューで、組織を順番に選択し、検出したホストの電源を切って削除するプロセスを繰り返します。アップグレードが完了したら、検出されたホストを再起動します。
5. Satellite Web UI で、**コンテンツ > Red Hat サブスクリプション** に移動して、**マニフェストの管理** をクリックします。サブスクリプションマニフェストペインで、**アクション** タブをクリックし、**マニフェストの更新** をクリックして、サブスクリプションマニフェストの最新コピーをダウンロードします。
6. リポジトリを設定します。
 - a. Satellite Web UI で、**コンテンツ > Red Hat リポジトリ** に移動して、**RPM** タブを選択します。
 - b. **製品** 一覧から **Red Hat Enterprise Linux Server** を探して、展開します。

- c. リポジトリセット 一覧から **Red Hat Satellite Tools 6.4 (for RHEL7 Server) (RPMs)** を探して、展開します。
 - d. **Red Hat Satellite Tools 6.4 for RHEL 7 Server RPMs x86_64** を選択します。
7. 新たに有効にしたリポジトリを同期します。
- a. Satellite Web UI で、**コンテンツ > 同期の状態** に移動します。
 - b. 製品の横にある矢印をクリックして、利用可能なリポジトリを表示します。
 - c. 6.4 のリポジトリを選択します。
 - d. **今すぐ同期** をクリックします。
リポジトリのアップデート時にエラーが発生した場合は、カスタマーポータルまたは Satellite Web UI からマニフェストを削除していないことを確認してください。マニフェストを削除すると、コンテンツホストからエンタイトルメントがすべて削除されます。マニフェストを更新しても問題が解決しない場合は、サポートケースを作成してください。
8. 6.3 バージョンのリポジトリを使用する既存のコンテンツビューを、6.4 用の新しいバージョンで更新します。更新したコンテンツビューのバージョンを公開してプロモートすると、リポジトリのバージョンが 6.4 になります。
9. サブスクリプションを更新します。

```
# subscription-manager refresh
```

10. Satellite Maintenance と Red Hat Enterprise Linux Ansible のリポジトリを有効にします。

```
# subscription-manager repos \
--enable rhel-7-server-satellite-maintenance-6-rpms \
--enable rhel-7-server-ansible-2.6-rpms
```

11. 以下のコマンドを入力して、**foreman-maintain** をインストールするか、最新バージョンに更新します。

```
# yum install rubygem-foreman_maintain
```

12. 利用可能なバージョンを確認して、希望のバージョンが表示されていることを確認します。

```
# foreman-maintain upgrade list-versions
```

13. ヘルスチェックオプションを使用して、システムをアップグレードする準備が完了しているかどうかを確認します。プロンプトが表示されたら、hammer の管理者ユーザー認証情報を入力して **foreman-maintain** を設定します。この変更は、**/etc/foreman-maintain/foreman-maintain-hammer.yml** ファイルに適用されます。

```
# foreman-maintain upgrade check --target-version 6.4
```

結果を確認し、アップグレードを実行する前に、強調表示されているエラー状態に対応します。

14. アップグレード時間は長くなるため、通信セッションの中断と再接続を可能にする **screen** などのユーティリティを使用します。これにより、コマンドシェルに接続し続けなくてもアップグレードの進捗が確認できるようになります。screen コマンドの使用方法は、**Red Hat ナ**

レジベースの「[How do I use the screen command?](#)」を参照してください。
 アップグレードコマンドを実行しているコマンドシェルへの接続がなくなった場合は、`/var/log/foreman-installer/satellite.log` のログで、プロセスが完全に終了したかどうかを確認できます。

15. アップグレードを実行します。

```
# foreman-maintain upgrade run --target-version 6.4
```

16. BASH シェルを使用している場合は、アップグレードに成功または失敗した後に、以下を入力します。

```
# hash -d foreman-maintain service 2> /dev/null
```

17. 作成したバックアップを使用して、DNS と DHCP の設定ファイルに必要なすべての変更を確認し、復元します。

18. 以前の手順で変更を加えた場合には、Satellite サービスを再起動します。

```
# foreman-maintain service restart
```

19. OpenSCAP プラグインがインストールされているにもかかわらず、デフォルトの OpenSCAP コンテンツが利用できない場合は、以下のコマンドを実行します。

```
# foreman-rake foreman_openscap:bulk_upload:default
```

3.1.2. 切断されている **Satellite Server** のアップグレード

Red Hat Content Delivery Network に接続していない Satellite Server には、この手順を使用します。



警告

設定ファイルをカスタマイズする場合は、手動または Hiera などのツールを使用します。このようなカスタマイズを行うと、アップグレード時または更新時にインストールスクリプトを実行する際に上書きされます。satellite-installer スクリプトで `--noop` オプションを使用すると、変更をテストできます。詳細は、ナレッジベースソリューションの「[How to use the noop option to check for changes in Satellite config files during an upgrade](#)」を参照してください。

作業を開始する前に

- Satellite Server をアップグレードする前に、ファイアウォールの設定を確認して更新してください。詳細は『[オンラインネットワークからの Satellite Server のインストール](#)』の「[ポートおよびファイアウォールの要件](#)」を参照してください。
- カスタマーポータルまたは Satellite Web UI からマニフェストを削除しないでください。削除すると、コンテンツホストからエンタイトルメントがすべて削除されます。

- アップグレードする前に、全 Foreman フックのバックアップを作成して、その後フックを削除します。アップグレードが完了し、Satellite が動作しているのを確認できるまで、フックを元に戻さないでください。



警告

カスタムの証明書を実装している場合は、`/root/ssl-build` ディレクトリーと、カスタム証明書に関連するソースファイルを作成したディレクトリーのコンテンツを保持する必要があります。

アップグレード時にこのファイルを保持できないと、アップグレードは失敗します。ファイルを削除してしまった場合は、アップグレードを進めるためにバックアップから復元する必要があります。

切断されている Satellite Server のアップグレード

1. バックアップを作成します。
 - 仮想マシンで、スナップショットを作成します。
 - 物理マシンで、バックアップを作成します。
2. アップグレード前スクリプトは競合を検出し、アップグレード後に登録解除および削除できる Satellite Server の重複エントリーがあるホストをリストできます。また、組織に割り当てられていないホストを検出します。**ホスト > すべてのホスト** で、組織の関連付けがないホストがリストされ、同じ名前のコンテンツホストに組織がすでに関連付けられている場合、コンテンツホストは自動的に登録解除されます。これは、アップグレード前にこのようなホストを組織に関連付けることによって回避できます。
アップグレードの前に、アップグレード前チェックスクリプトを実行して、削除できるホストのリストを取得します。関連付けられていないホストが検出された場合は、アップグレードの前に組織にそれらのホストを関連付けることが推奨されます。

```
# foreman-rake katello:upgrade_check
```

アップグレードチェックで、タスクが実行中であることが原因の障害が報告された場合は、タスクが完了するまで待機することが推奨されます。一部のタスクはキャンセルすることができますが、Red Hat ナレッジベースソリューション「[Red Hat Satellite 6 で一時停止したタスクを管理する](#)」に目を通して、安全にキャンセルできるタスクとできないタスクについて理解する必要があります。

3. DNS と DHCP の設定ファイルである `/etc/zones.conf` と `/etc/dhcp/dhcpd.conf` のバックアップをそれぞれ作成します。インストーラーでは、ドメインまたはサブネットが 1 つしかサポートされないため、バックアップからの変更の復元が必要になる場合があります。
4. DNS または DHCP の設定ファイルを手動で編集し、変更を上書きしない場合は、以下のよう
にインストーラースクリプトを実行します。

```
# satellite-installer --foreman-proxy-dns-managed=false \
--foreman-proxy-dhcp-managed=false
```

5. Satellite Web UI で、**ホスト > 検出されたホスト** に移動します。検出されたホストが利用可能

な場合は無効にし、**検出されたホスト** ページにあるエントリーをすべて削除します。必要に応じて、組織設定メニューから、その他の組織を順番に選択し、すべてのエントリーを削除します。アップグレードが完了したら、検出されたホストを再起動します。

6. すべての外部 Capsule Server が組織に割り当てられていることを確認します。割り当てられていない場合、これらのサーバーは、ホスト統合の変更により登録解除された可能性があります。

7. Puppet 4 リポジトリを無効にします。

```
# subscription-manager repos \
--disable=rhel-7-server-satellite-6.3-puppet4-rpms
```

8. Satellite サービスを停止します。

```
# katello-service stop
```

9. 『オフラインネットワークからの **Satellite SERVER** のインストール』の「[切断されたネットワークからのダウンロードおよびインストール](#)」の手順に従って、最新の ISO ファイルを取得し、マウントしてから、パッケージをインストールします。

10. カスタム Apache サーバー設定がある場合は、次の手順でインストールデフォルト値に戻ります。**アップグレードの実行時**に変更する内容を確認する場合は、**--noop** (no operation) オプションとともにアップグレードコマンドを入力し、次の手順でアップグレードコマンドを入力するときに適用される変更内容を確認できます。このテストを行わない場合は、次の手順に進みます。テストを行う場合は、以下の手順を行います。

- a. 次の行を **/etc/httpd/conf/httpd.conf** 設定ファイルに追加します。

```
Include /etc/httpd/conf.modules.d/*.conf
```

- b. **httpd** サービスを再起動します。

```
# systemctl restart httpd
```

- c. **postgresql** データベースサービスおよび **mongod** データベースサービスを起動します。

```
# systemctl start postgresql
# systemctl start mongod
```

- d. 以下のように、**--noop** オプションを付けて、インストーラスクリプトを実行します。

```
# satellite-installer --scenario satellite --upgrade --verbose --
noop
```

/var/log/foreman-installer/satellite.log を参照して、**--noop** オプションを付けずに実行した場合に **適用される変更** を確認します。設定ファイルへの変更を示す **+++** と **---** の記号を探します。この "no operation" オプションにより、実際にはファイルは作成されませんが、モジュール内の一部の Puppet リソースではファイルが作成されていることが期待されるため、エラーメッセージが表示されるはずです。

- e. Satellite サービスを停止します。

```
# katello-service stop
```

-
11. アップグレード時間は長くなるため、通信セッションの中断と再接続を可能にする **screen** などのユーティリティを使用します。これにより、コマンドシェルに接続し続けなくてもアップグレードの進捗が確認できるようになります。screen コマンドの使用方法は、**Red Hat ナレッジベース** の「[How do I use the screen command?](#)」を参照してください。
アップグレードコマンドを実行しているコマンドシェルへの接続がなくなった場合は、`/var/log/foreman-installer/satellite.log` のログで、プロセスが完全に終了したかどうかを確認できます。

12. 以下のコマンドを入力して、**foreman-maintain** をインストールするか、最新バージョンに更新します。

```
# yum install rubygem-foreman_maintain
```

13. 利用可能なバージョンを確認して、希望のバージョンが表示されていることを確認します。

```
# foreman-maintain upgrade list-versions
```

14. ヘルスチェックオプションを使用して、システムをアップグレードする準備が完了しているかどうかを確認します。プロンプトが表示されたら、hammer の管理者ユーザー認証情報を入力して **foreman-maintain** を設定します。この変更は、`/etc/foreman-maintain/foreman-maintain-hammer.yml` ファイルに適用されます。

```
# foreman-maintain upgrade check --target-version 6.4
```

結果を確認し、アップグレードを実行する前に、強調表示されているエラー状態に対応します。

15. アップグレードを実行します。

```
# foreman-maintain upgrade run --target-version 6.4
```

16. BASH シェルを使用している場合は、アップグレードに成功または失敗した後に、以下を入力します。

```
# hash -d foreman-maintain service 2> /dev/null
```



警告

config サブディレクトリーを含むディレクトリーからコマンドを実行すると、以下のエラーが発生します。

```
ERROR: Scenario (config/satellite.yaml) was not found, can not continue.
```

このような場合は、**root** ユーザーのホームディレクトリーに移動し、コマンドを再実行します。

17. 作成したバックアップを使用して、DNS と DHCP の設定ファイルに必要なすべての変更を確認し、復元します。
18. 以前の手順で変更を加えた場合には、Satellite サービスを再起動します。

```
# foreman-maintain service restart
```

19. OpenSCAP プラグインがインストールされているにもかかわらず、デフォルトの OpenSCAP コンテンツが利用できない場合は、以下のコマンドを実行します。

```
# foreman-rake foreman_openscap:bulk_upload:default
```

20. Satellite Web UI で **設定 > 検出ルール** に移動し、選択した組織および場所を検出ルールに関連付けます。

3.2. CAPSULE SERVER のアップグレード

このセクションは、Capsule Server を 6.3 から 6.4 にアップグレードする方法を説明します。

作業を開始する前に

- Capsule Server をアップグレードする前に、Satellite Server がアップグレードされている必要があります。
- Capsule Server は Puppet 4 にアップグレードする必要があります。詳細は、『**Red Hat Satellite 6.3 アップグレードおよびアップデートガイド**』の「[Puppet のアップグレード](#)」の章を参照してください。
- Red Hat Satellite Capsule 6.4 リポジトリが Satellite Server で有効になっており、同期されているのを確認します。
- Capsule Server のベースオペレーティングシステム、または Capsule Server リポジトリへの更新をコンテンツビューで管理する場合は、更新したコンテンツビューを公開する必要があります。
- 新たにアップグレードした Satellite Server に、Capsule のベースシステムが登録されていることを確認します。
- 新たにアップグレードした Satellite Server で、Capsule の組織と場所が正しく設定されていることを確認します。
- Capsule Server をアップグレードする前に、ファイアウォールの設定を確認して更新してください。詳細は『**Capsule Server のインストール**』の「[ポートおよびファイアウォールの要件](#)」を参照してください。

**警告**

カスタムの証明書を実装している場合は、**/root/ssl-build** ディレクトリーと、カスタム証明書に関連するソースファイルを作成したディレクトリーのコンテンツを保持する必要があります。

アップグレード時にこのファイルを保持できないと、アップグレードは失敗します。ファイルを削除してしまった場合は、アップグレードを進めるためにバックアップから復元する必要があります。

Capsule Server のアップグレード

1. バックアップを作成します。

- 仮想マシンで、スナップショットを作成します。
- 物理マシンで、バックアップを作成します。
バックアップに関する詳細は、『**Red Hat Satellite 6.3 管理**』ガイドの「[Satellite Server および Capsule Server のバックアップと復元](#)」を参照してください。

2. DNS と DHCP の設定ファイルである **/etc/zones.conf** と **/etc/dhcp/dhcpd.conf** のバックアップをそれぞれ作成します。インストーラーでは、ドメインまたはサブネットが 1 つしかサポートされないため、バックアップからの変更の復元が必要になる場合があります。

3. DNS または DHCP の設定ファイルを手動で編集し、変更を上書きしない場合は、以下のコマンドを実行します。

```
# satellite-installer --foreman-proxy-dns-managed=false \
--foreman-proxy-dhcp-managed=false
```

4. Red Hat Satellite 6.3 のリポジトリを無効にします。

```
# subscription-manager repos \
--disable rhel-7-server-satellite-capsule-6.3-rpms
```

5. Puppet 4 リポジトリを無効にします。

```
# subscription-manager repos \
--disable=rhel-7-server-satellite-capsule-6.3-puppet4-rpms
```

6. 新しいリポジトリを有効にします。

Red Hat Software Collections リポジトリは、リモート実行機能を含む一部の Red Hat Satellite 機能に必要な、新しいバージョンの Ruby を提供します。Satellite ツールリポジトリでは、エラーを管理するための通信サービスを提供する **gofer** および **katello-agent** が提供されます。

- 以下のコマンドを実行します。

```
# subscription-manager repos \
--enable rhel-7-server-satellite-capsule-6.4-rpms \
```



```
--enable rhel-server-rhsc1-7-rpms \  
--enable rhel-7-server-satellite-tools-6.4-rpms \  
--enable rhel-7-server-satellite-maintenance-6-rpms \  
--enable rhel-7-server-ansible-2.6-rpms
```

7. Satellite web UI で、**ホスト > 検出されたホスト** に移動します。検出されたホストが利用できる場合は、そのホストの電源を切り、**検出されたホスト** ページに表示されているすべてのエントリーを削除します。必要に応じて、組織設定メニューから、その他のすべての組織を順番に選択して、すべてのエントリーを削除します。アップグレードが完了したら、検出されたホストを再起動します。

8. リポジトリキャッシュを削除します。

```
# yum clean all
```

9. Satellite サービスを停止します。

```
# katello-service stop
```

10. すべてのパッケージを更新します。

```
# yum update
```

11. 検出されたホストのプロキシとして Capsule Server を使用する場合は、検出プラグインをインストールします。

```
# yum install rubygem-smart_proxy_discovery.noarch
```

12. Capsule Server で **foreman_url** 設定が正しいことを確認します。

```
# grep foreman_url /etc/foreman-proxy/settings.yml
```

Satellite Server の完全修飾ドメイン名が表示されます。

13. **--upgrade** オプションを使用してインストーラスクリプトを実行することによりアップグレードを実行します。

```
# satellite-installer --scenario capsule --upgrade
```



警告

config サブディレクトリーを含むディレクトリーからコマンドを実行すると、以下のエラーが発生します。

```
ERROR: Scenario (config/capsule.yaml) was not found,  
can not continue.
```

このような場合は、**root** ユーザーのホームディレクトリーに移動し、コマンドを再実行します。

14. 作成しておいたバックアップを使用して、DNS と DHCP の設定ファイルに必要なすべての変更を確認し、復元します。
15. Satellite Server で foreman-discovery パッケージをアップグレードし、アップグレード前にシャットダウンしたホストを有効にします。

3.3. SATELLITE クライアントのアップグレード

現時点では、Satellite Tools リポジトリに含まれる、Satellite 6.3 バージョンの **katello-agent** などのクライアントライブラリーは、Satellite 6.4 で正式にテストされていないため、サポート対象外となります。

Satellite Server との完全互換性を持たせるために、すべてのクライアントを、速やかに新しいバージョンの **katello-agent** にアップグレードします。このため、Satellite Tools リポジトリを 6.3 から 6.4 に変更する必要があります。これは手動で行うか、**satellite-6.4-tools-upgrade** パッケージをインストールすることで行えます。このパッケージには、Satellite Tools リポジトリのバージョンを変更するポストインストールスクリプトだけが含まれます。

前提条件

- Satellite Server がアップグレードされている必要があります。
- Satellite で、新しい Satellite Tools リポジトリを有効にしておく必要があります。
- Satellite で、新しいリポジトリを同期しておく必要があります。
- クライアントに **katello-agent** がインストールされていない場合は、手動で作業を行います。



警告

カスタムの証明書を実装している場合は、`/root/ssl-build` ディレクトリーと、カスタム証明書に関連するソースファイルを作成したディレクトリーのコンテンツを保持する必要があります。

アップグレード時にこのファイルを保持できないと、アップグレードは失敗します。ファイルを削除してしまった場合は、アップグレードを進めるためにバックアップから復元する必要があります。

一括リポジトリ設定 UI を使用した Satellite クライアントのアップグレード

1. Satellite Web UI で、**ホスト > コンテンツホスト** に移動し、アップグレードするコンテンツホストを選択します。
2. **アクションの選択** 一覧から **リポジトリセットの管理** を選択します。
3. **リポジトリセットの管理** の一覧から、**Red Hat Satellite Tools 6.3** のチェックボックスを選択します。
4. **アクションの選択** 一覧から **Override to Disabled** (「無効」に上書き) を選択し、**Done (完了)** をクリックします。
5. プロセスが完了したら、以前の手順で使用した同じホストセットで、**リポジトリセットの管理** を選択します。
6. **リポジトリセットの管理** の一覧から、**Red Hat Satellite Tools 6.4** のチェックボックスを選択します。
7. **アクションの選択** 一覧から **Override to Enabled** (「有効」に上書き) を選択し、**Done (完了)** をクリックします。
8. プロセスが完了したら、以前の手順で使用した同じホストセットで、**リポジトリセットの管理** を選択します。
9. **アクションの選択** 一覧から **パッケージの管理** を選択します。
10. **パッケージ** 検索フィールドで、**katello-agent** と入力します。
11. **更新** 一覧から、希望の更新方法を選択します。
12. 更新が完了したことを確認してから、**Done (完了)** をクリックします。

Satellite クライアントの手動アップグレード

1. クライアントシステムにログインします。
2. 以前のバージョンの Satellite のリポジトリを無効にします。

```
# subscription-manager repos \
--disable rhel-7-server-satellite-tools-6.3-rpms
```

3. 現行バージョンの Satellite の Satellite Tools リポジトリを有効にします。

```
# subscription-manager repos \
--enable=rhel-7-server-satellite-tools-6.4-rpms
```

4. 以下の **Katello**、**Pulp**、および **qpidd** パッケージをアップグレードします。

```
# yum upgrade katello-agent katello-host-tools katello-host-tools-
fact-plugin pulp-rpm-handlers qpidd-proton-c
```

5. **goferd** を再起動します。

```
# systemctl restart goferd
```

3.4. アップグレード後のタスク

本セクションで紹介する手順の一部はオプションです。お使いのインストールに関連する手順のみを選択できます。

PXE ベースの検出プロセスを使用する場合には、Satellite の **ホスト > 検出されたホスト** ページに表示させるホストで、Satellite または Capsule Server 上で Discovery アップグレードの手順を実行する必要があります。

Satellite または Capsule のバックアップおよび復元のスクリプトがある場合には、**katello-backup** と **katello-restore** を **foreman-maintain backup** と **foreman-maintain restore** に置き換えるように、このスクリプトを更新するようにしてください。

3.4.1. Discovery のアップグレード

このセクションでは、PXE ブートを使用して Satellite Server に登録するホストに渡した PXELinux テンプレートとブートイメージをアップデートする方法を説明します。

Satellite 6.4 以降、プロビジョニングテンプレートには別途サブネットが関連付けられるので、対象のサブネットに対して TFTP Capsule を使用するように初期設定しないようにしてください。アップグレード後にサブネットを作成する場合には、特に Satellite または Capsule が Discovery テンプレートにプロキシサービスを提供できるようにしてから、**テンプレート Capsule** を使用するように、検出されたホストで全ホストを設定する必要があります。

アップグレード中は、TFTP プロキシが設定された各サブネットが有効化されている場合には、テンプレート Capsule を TFTP Capsule と同じに設定してください。アップグレード後には、すべてのサブネットが正しく設定されていることを確認してください。

ホストで PXE ブートを使用しない場合には、Satellite に新規ホストを検出させるために、これらの手順は、必要ありません。

3.4.1.1. Satellite Server での Discovery のアップグレード

1. Satellite web UI で Discovery テンプレートをアップデートします。
 - a. **ホスト > プロビジョニングテンプレート** に移動します。
 - b. **PXELinux global default** 行で **クローン** をクリックします。

- c. **名前** フィールドに、テンプレートの新しい名前を入力します (例: **ACME PXE global default**)。
 - d. テンプレートエディターフィールドで、**ONTIMEOUT local** 行を **ONTIMEOUT discovery** に変更し、**送信** をクリックします。
 - e. **管理 > 設定** に移動します。
 - f. **Global default PXELinux template** の **値** をクリックします。
 - g. 新しく作成したテンプレートの名前を選択し、**チェックボタン** をクリックします。
 - h. **ホスト > プロビジョニングテンプレート** に移動します。
 - i. **PXE デフォルトのビルド** をクリックして、**OK** をクリックします。
2. Satellite Web UI で **設定 > 検出ルール** に移動し、選択した組織および場所を検出ルールに関連付けます。

3.4.1.2. Capsule Server での Discovery のアップグレード

1. Satellite Server で、Foreman Discovery パッケージが最新であることを確認します。

```
# yum upgrade tfm-rubygem-foreman_discovery
```

2. 以前の手順で更新があれば、Satellite サービスを再起動します。

```
# katello-service restart
```

3. 検出されたホストでプロビジョニングネットワークに接続した Satellite Capsule の Discovery イメージ、または検出されたホストに TFTP サービスを提供する Satellite Capsule の Discovery イメージをアップグレードします。

```
# yum upgrade foreman-discovery-image
```

4. 同じインスタンスに、Proxy サービスを提供するパッケージをインストールして、**foreman-proxy** サービスを再起動します。

```
# yum install rubygem-smart_proxy_discovery
# service foreman-proxy restart
```

5. Satellite web UI で、**インフラストラクチャー > Capsule** に移動して、関連する Capsule の機能列に **Discovery** が表示されていることを確認します。必要に応じて、**アクション** ドロップダウンメニューから **更新** を選択します。
6. **インフラストラクチャー > サブネット** に移動し、検出を使用する各サブネットで行います。
 - a. サブネット名をクリックします。
 - b. **Capsule** タブで、上で設定した Capsule に **Discovery Capsule** が設定されているのを確認します。

3.4.1.3. サブネットにテンプレート Capsule があることの確認

検出されたホストが含まれる全サブネットに、テンプレート **Capsule** が設定されていることを確認します。

1. Satellite Web UI で、インフラストラクチャー > **Capsule** に移動します。
2. 確認するサブネットを選択します。
3. **Capsule** タブで、テンプレート **Capsule** が、このサブネットに設定されていることを確認します。

テンプレート Capsule を使用したサブネットの設定に関する詳細は、『**Red Hat Satellite ホスト管理**』ガイドの「[Discovery サブネットの設定](#)」を参照してください。

3.4.2. virt-who のアップグレード

Satellite または Capsule Server に virt-who がインストールされている場合は、Satellite または Capsule Server のアップグレード時に一緒にアップグレードされるため、追加の作業は必要ありません。virt-who を他の場所にインストールしている場合は、手動でアップグレードする必要があります。

作業を開始する前に

Satellite または Capsule Server に登録しているホストに virt-who がインストールされている場合は、最初にホストを、Satellite Tools リポジトリで利用可能な最新パッケージにアップグレードします。ホストのアップグレードに関する詳細は「[Satellite クライアントのアップグレード](#)」を参照してください。

virt-who の手動アップグレード

1. virt-who をアップグレードします。
2. virt-who サービスを再起動して、新しいバージョンを有効にします。

```
# yum upgrade virt-who
```

```
# systemctl restart virt-who.service
```

3.4.3. 以前のバージョンの **Satellite Tools** リポジトリの削除

Satellite 6.4 へのアップグレードが完了したら、コンテンツホストから Red Hat Satellite Tools 6.3 リポジトリを削除して、無効にすることができます。

バージョン 6.3 の Satellite Tools リポジトリを無効にします。

1. Satellite Web UI で、コンテンツ > **Red Hat サブスクリプション** に移動します。
2. 有効されたりポジトリ エリアで、**Red Hat Satellite Tools 6.3 for RHEL 7 Server RPMs x86_64** を探し出します。
3. 右側の **無効化** アイコンをクリックします。

リポジトリがまだコンテンツビューに含まれている場合には、無効にできません。スケジュールされているタスクにより、無効にされたりポジトリからパッケージが自動的に削除されます。

第4章 PUPPET のアップグレード

Red Hat Satellite 6.4 は Puppet 5 のみをサポートします。Puppet 5 は、Puppet 4 から Puppet 5 への移行しかサポートしません。そのため、3 以前のバージョンを使用している場合は、Satellite 6.4 にアップグレードする前に、Puppet 4 にアップグレードする必要があります。Satellite 6.3 には Puppet 4 へのアップグレードパスが含まれています。詳細は、**Satellite 6.3 『Red Hat Satellite のアップグレードおよびアップデート』** ガイドの「[Puppet のアップグレード](#)」セクションを参照してください。

Satellite のアップグレードプロセス時に、アップグレードプロセスの一部として、Puppet は 4 から 5 にアップグレードされます。本章では、Puppet 4 から Puppet 5 へのアップグレードの準備プロセスについて説明します。

4.1. アップグレードパス

Satellite 6.4 にアップグレードする前に、Puppet モジュールを確認し、変更を加えて Puppet 5 との互換性を確保する必要があります。Puppet 5 での変更に関する情報は、Puppet のウェブサイトの [Puppet 5.0 Release Notes](#) を参照してください。

Red Hat Satellite と Capsule を、同時にアップグレードする必要はありません。Puppet モジュールのアップグレードおよびテストに伴い、Capsules をアップグレードすることができます。もしくは、新しい Capsule をインストールして、テストしてからホストを移動することもできます。Satellite web UI、Hammer CLI、または **bootstrap** スクリプトを使用して、Puppet 4 Capsule から Puppet 5 Capsule へホストを移動できます。

また、アップグレードした Puppet モジュールの最終テストに使用するために、既存の Capsule のクローンを作成することが推奨されます。リポジトリのサイズによっては、既存の Capsule のクローンを作成するのに時間がかかります。

Puppet バージョンを判断するためのパラメーターの使用

デフォルトでは、Satellite 6.4 はグローバルパラメーター **enable-puppet5** を設定します。キックスタートファイルなどのプロビジョニングファイルは、**enable-puppet5** を使用して Puppet 5 エージェントがある場合は、新規ホストをプロビジョニングして Puppet 5 エージェントを使用します。ただし、Puppet 4 エージェントがコンテンツビューで、ホストにアタッチされている場合には、Puppet 5 の代わりにこれが使用されます。ホストで Puppet 5 を使用する場合には、このホストに関連付けられているコンテンツビューを Puppet 5 エージェントが含まれるように更新する必要があります。

Puppet 4 を使用するために新規ホストをプロビジョニングする必要がある場合には、ホストのグループレベルでグローバルパラメーターを上書きできます。

1. Web UI で、**設定 > ホストグループ** に移動し、設定するホストグループ名を選択します。
2. **パラメーター** タブをクリックして、**ホストのグループパラメーター** のエリアで、**パラメーターの追加** をクリックします。
3. **名前** フィールドで、**enable-puppet4** と入力し、**値** フィールドで、**true** と入力します。
4. **送信** をクリックして変更を保存します。

Satellite 6.3 で **enable-puppet4** のパラメーターを設定した場合には、これで安全に削除できます。

1. Web UI で、**設定 > ホストグループ** に移動し、設定するホストグループ名を選択します。
2. **パラメーター** タブをクリックして、**ホストのグループパラメーター** のエリアで、**enable-puppet4** を特定します。

3. **アクション** コラムで、**削除** をクリックします。

4. **送信** をクリックして変更を保存します。

Hammer コマンドを使用して削除することもできます。以下に例を示します。

```
# hammer hostgroup delete-parameter --hostgroup my_hostgroup --name  
enable-puppet4
```

4.2. PUPPET モジュールのテスト

Puppet ウェブサイトの [Puppet 5.0 Release Notes](#) と以下のガイドラインを使用して、アップグレードを開始する前に Puppet モジュールをテストしてください。

- Puppet モジュールは、最初に、Satellite に接続していないワークステーションでテストします。
- **puppet parser validate** コマンドを実行して、コマンドの構文を確認します。
- サポートされなくなっているため、クラス参照名に大文字がないかを確認してください。**Class[MyClass]** から **Class[myclass]** に変更するなど、クラス名を小文字に変更します。
- Ruby 形式のカスタムのバージョン比較コードがある場合は、SemVer の範囲の一致を処理する **Range#intersection** コードは削除されている点に注意してください。
- カスタムの Ruby コードで、文字列のフォーマット方法として、非推奨の **TypeCalculator** クラスが使用されている場合には、**TypeCalculator** の文字列フォーマット方法が削除されたため、**TypeFormatter** クラスが使用されるように、コードを変更してください。詳細は、Puppet ウェブサイトで [Type mismatch describer should expand aliases](#) の問題を参照してください。
- 以前のバージョンでは、クラスが複数回定義されている場合には、ボディがマージされていました。--strict フラグの状態によって、警告またはエラーが表示されていましたが、'' で指定されている上位のスコープクラス以外は、エラーが常に表示されるようになりました ('main' と呼ばれる)。

4.3. アップグレードの準備

このセクションを使用して、Puppet 5 にアップグレードするためにホストを準備してください。Satellite Server と Capsule Server は、適切なアップグレード手順の一部として Puppet 5 にアップグレードされます。

作業を開始する前に

- Satellite Server は Satellite 6.4 へのアップグレードを済ませておくようにしてください。
- アタッチされているホストで、アップグレードするホストで、Capsule Server のアップグレードを完了してください。
- Puppet モジュールをレビューし、Puppet 5 にアップグレードする必要がある箇所でアップグレードが行われていることを確認します。

- 関連のあるコンテンツビュー が更新され、Puppet 5 エージェントが含まれていることを確認します。
- グローバルの **enable-puppet5** パラメーターを上書きする **enable-puppet4** パラメーターがホストとホストグループに設定されていないことを確認してください。

Puppet 5 用のホストの準備

1. Puppet 4 リポジトリを無効にします。

- Puppet 4 リポジトリは、デスクトップ、サーバー、およびワークステーションから以下を取得します。

```
rhel-X-platform-satellite-tools-6.3-puppet4-rpms
```

ここで、**X** は使用してるメジャーリリースのバージョンで、**platform** は **desktop**、**server**、または **workstation** になります。

- Puppet 4 リポジトリは、選択したアーキテクチャーから以下を取得します。

```
rhel-X-for-architecture-satellite-tools-6.3-puppet4-rpms
```

ここで、**X** はメジャーリリースのバージョン、**architecture** は **arm**、**power**、**hpc-node**、または **system-z** などです。リポジトリの中には、延長サポートが含まれているサブスクリプション向けのものもあります。たとえば、延長アップデートサポート (EUS) の場合は、アーキテクチャーの後に **eus** と続きます。

延長サポートの種類については、Red Hat ナレッジベースの記事「[Red Hat Enterprise Linux のサポートポリシーについてよく利用されるナレッジ](#)」を参照してください。

1. コンテンツホストで、Puppet 4 RPM リポジトリを検索します。

```
# subscription-manager repos --list-enabled |grep puppet4
```

2. 以下のように、ホストで見つかった Puppet 4 リポジトリを無効にします。

```
# subscription-manager repos \
--disable=rhel-7-server-satellite-tools-6.3-puppet4-rpms
```

3. すべてのパッケージをアップデートします。

```
# yum update
```

第5章 SATELLITE SERVER、CAPSULE SERVER、およびコンテンツホストのアップデート

本章を参照して、既存の Red Hat Satellite 環境を、新しい Red Hat Satellite マイナーバージョンにアップデートします (たとえば 6.3.0 から 6.3.1 へのアップデート)。詳細は「[5章 Satellite Server、Capsule Server、およびコンテンツホストのアップデート](#)」を参照してください。

本章では、Satellite Server、Capsule Server、およびコンテンツホストをアップデートする手順を説明します。

Satellite のマイナーバージョン間の更新

アップデートは、Satellite Server、Capsule Server、およびコンテンツホストを、新しいマイナーバージョンに移行するプロセスです。アップデートでは、通常、コードがリリースされてから見つかったセキュリティ脆弱性にパッチを当て、深刻でない問題を修正します。一般的に、アップデートにかかる時間は短く、オペレーティング環境を破壊することはありません。

アップデートの前に、『[Red Hat Satellite リリースノート](#)』で潜在的な競合を確認し、Satellite Server と全 Capsule Server のバックアップを作成してください。詳細は『[Red Hat Satellite の管理](#)』ガイドの「[Satellite Server および Capsule Server のバックアップと復元](#)」を参照してください。

以下の手順に従って、マイナーバージョン間のアップデート (たとえば 6.3.0 から 6.3.1 へのアップデート) を行います。

5.1. SATELLITE SERVER のアップデート

前提条件

- Satellite、Capsule、および Satellite Tools の Satellite Server リポジトリが同期されていることを確認します。
- 外部の Capsule およびコンテンツホストのアップデートは、アップデートしたリポジトリに関連するすべてのコンテンツビューにそれぞれプロモートすることで行います。



警告

設定ファイルをカスタマイズする場合は、手動または Hiera などのツールを使用します。このようなカスタマイズを行うと、アップグレード時または更新時にインストールスクリプトを実行する際に上書きされます。satellite-installer スクリプトで **--noop** オプションを使用すると、変更をテストできます。詳細は、ナレッジベースソリューションの「[How to use the noop option to check for changes in Satellite config files during an upgrade](#)」を参照してください。

Satellite Server を次のマイナーバージョンにアップデート

Satellite Server のアップデート手順:

1. Satellite Maintenance リポジトリが有効になっているのを確認します。

```
# subscription-manager repos --enable \
```

```
rhel-7-server-satellite-maintenance-6-rpms
```

2. **foreman-maintain** がインストールされており、最新になっているのを確認します。

```
# yum install rubygem-foreman_maintain
```

3. 利用可能なバージョンを確認して、次のマイナーバージョンが一覧に追加されているのを確認します。

```
# foreman-maintain upgrade list-versions
```

4. ヘルスチェックオプションを使用して、システムをアップグレードする準備が完了しているかどうかを確認します。このコマンドを最初に使用したときに、**foreman-maintain** により **hammer** 管理者ユーザー認証情報を入力して、**/etc/foreman-maintain/foreman-maintain-hammer.yml** ファイルに保存します。

```
# foreman-maintain upgrade check --target-version 6.4.z
```

結果を確認し、アップグレードを実行する前に、強調表示されているエラー状態に対応します。

5. 更新時間は長くなるため、通信セッションの中断と再接続を可能にする **screen** などのユーティリティを使用します。これにより、コマンドシェルに接続し続けなくてもアップグレードの進捗が確認できるようになります。screen コマンドの使用方法は、**Red Hat ナレッジベース** の「[How do I use the screen command?](#)」を参照してください。
アップグレードコマンドを実行しているコマンドシェルへの接続がなくなった場合は、**/var/log/foreman-installer/satellite.log** のログで、プロセスが完全に終了したかどうかを確認できます。

6. アップグレードを実行します。

```
# foreman-maintain upgrade run --target-version 6.4.z
```

7. 最後にシステムを再起動してから、カーネルパッケージが更新されたかを確認します。

```
# rpm -qa --last | grep kernel
```

8. 最後にシステムを再起動してからカーネルが更新されている場合は、システムを再起動します。

```
# reboot
```

5.2. CAPSULE SERVER のアップデート

Capsule Server を次のマイナーバージョンに更新

Capsule Server のアップデート手順:

1. 適切なりポジトリだけが有効になっていることを確認します。
 - a. 有効なりポジトリを一覧表示します。

```
# subscription-manager repos --list-enabled
```

- b. 以下のリポジトリだけが有効になっていることを確認します。

```
rhel-7-server-rpms
rhel-7-server-satellite-capsule-6.4-rpms
rhel-server-rhsc1-7-rpms
rhel-7-server-satellite-tools-6.4-rpms
rhel-7-server-satellite-maintenance-6-rpms
```

リポジトリの無効化および有効化に関する詳細は、『**Capsule Server のインストール**』の「[リポジトリの設定](#)」を参照してください。**rhel-7-server-satellite-tools-6.4-rpms** リポジトリでは、Katello エージェントが提供されます。詳しい情報は、『**Capsule Server のインストール**』の「[katello エージェントのインストール](#)」を参照してください。Red Hat Software Collections リポジトリはオプションです。これは、リモート実行機能に必要です。

2. Satellite サービスを停止します。

```
# katello-service stop
```

3. すべてのパッケージをアップデートします。

```
# yum update
```

カーネルアップデートが発生したら、アップグレードが完了後にシステムを再起動する必要があります。この時点ではシステムを再起動しないでください。

4. インストーラスクリプトに **--upgrade** オプションを付けて実行し、アップデートを実行します。

```
# satellite-installer --scenario capsule --upgrade
```

5. **yum update** の最中にカーネルアップデートが行われたら、システムを再起動します。

```
# reboot
```

6. 前の手順でシステムを再起動しなかった場合は、**goferd** を再起動します。

```
# systemctl restart goferd
```

5.3. コンテンツホストのアップデート

コンテンツホストを次のマイナーバージョンにアップデート

コンテンツホストのアップデート手順:

1. すべてのパッケージをアップデートします。

```
# yum update
```

2. カーネルが更新されたら、システムを再起動します。

```
# reboot
```

3. 前の手順でシステムを再起動しなかった場合は、**goferd** を再起動します。

```
# systemctl restart goferd
```