



Red Hat Satellite 6.3

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 6.3 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	6
第2章 RED HAT SATELLITE のアップグレード	7
2.1. 自己登録 SATELLITE のマイグレート	7
2.2. RED HAT ENTERPRISE LINUX 6 から RED HAT ENTERPRISE LINUX 7 へのマイグレーション	8
2.2.1. 除外	9
2.2.2. 作業を行う前に	9
2.2.3. Satellite Server 移行の概要	9
2.2.4. Satellite Server のマイグレーション	10
2.2.5. Capsule Server のマイグレーション	13
2.3. SATELLITE SERVER のアップグレード	14
2.3.1. 接続している Satellite Server のアップグレード	15
2.3.2. 切断されている Satellite Server のアップグレード	17
2.4. CAPSULE SERVER のアップグレード	21
2.5. アップグレード後のタスク	24
2.5.1. Discovery のアップグレード	24
2.5.1.1. Satellite Server での Discovery のアップグレード	24
2.5.1.2. Capsule Server での Discovery のアップグレード	25
2.5.2. Satellite クライアントのアップグレード	26
2.5.3. virt-who のアップグレード	27
2.5.4. 以前のバージョンの Satellite Tools リポジトリの削除	27
2.5.5. PXE ローダーオプションのアップグレード	28
2.5.6. テンプレートの更新	28
2.5.6.1. テンプレートの完了	29
2.5.7. 外部の DHCP サービスのアップグレード	29
第3章 PUPPET のアップグレード	30
3.1. アップグレードパス	30
3.2. PUPPET モジュールのテスト	30
3.3. アップグレードの実行	31
第4章 SATELLITE SERVER、CAPSULE SERVER、およびコンテンツホストのアップデート	33
4.1. SATELLITE SERVER のアップデート	33
4.2. CAPSULE SERVER のアップデート	34
4.3. コンテンツホストのアップデート	35

第1章 アップグレードの概要



警告

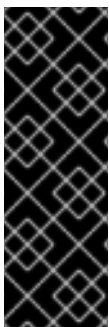
高可用性設定に Satellite 6 をインストールしている場合は、Satellite 6.3 にアップグレードする前に Red Hat サポートにご連絡ください。

本章では、Red Hat Satellite 6.3 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 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」は、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 6.2 から Satellite 6.3)。
- 「アップデート」は、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 6.2.10 から Satellite 6.2.11)。
- 「マイグレーション」は、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での 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 6.2 は Capsule Server 6.3 と互換性がなく、Satellite Server 6.3 は Capsule Server 6.2 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 6.2.11 の Satellite Server は、6.2.10 の Capsule Server と互換性があります。

1.1. 前提条件

Satellite 6.3 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 6.3 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。
- 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 6.3 でストレージ要件が変更になりました。詳細は『インストールガイド』の「[ストレージ要件および推奨事項](#)」を参照してください。
- Satellite Server と、すべての Capsule Server のバックアップを作成します。詳細は『[Red Hat Satellite 6.2 サーバー管理ガイド](#)』の「[バックアップおよび災害復旧](#)」を参照してください。



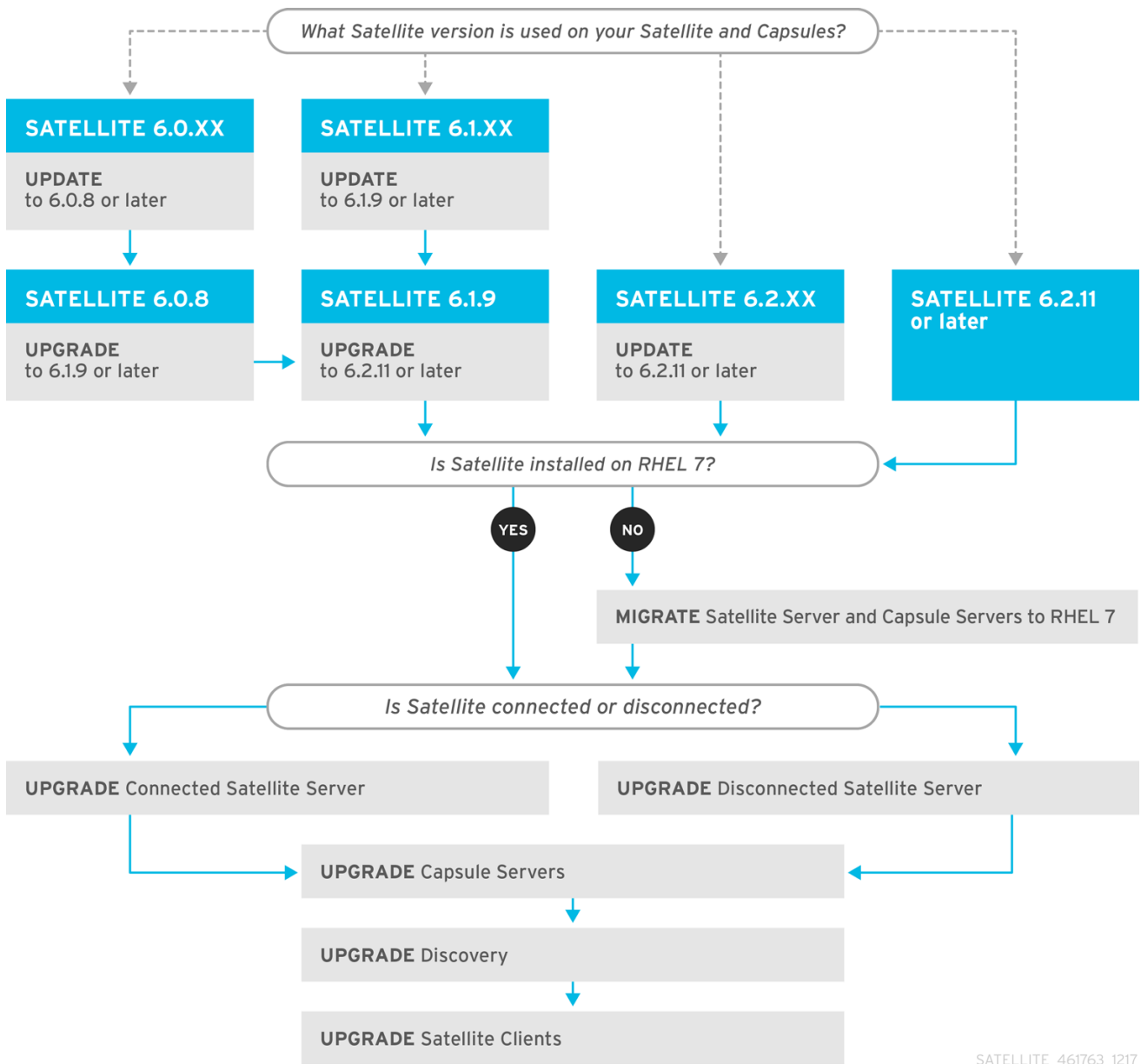
警告

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 へのアップグレードパスは、お使いの環境によって異なります。Satellite 6.3 は、Red Hat Enterprise Linux 7 でのみサポートされます。Satellite Server および Capsule Server を Red Hat Enterprise Linux 6 で使用している場合は、最初に Satellite 6.2.13 以降にアップデートし、Red Hat Enterprise Linux 7 に移行してから、Satellite 6.3 にアップグレードする必要があります。Satellite Server と Capsule Server を Red Hat Enterprise Linux 7 で使用している場合は、Satellite 6.2.11 にアップデートしてから、Satellite 6.3 にアップグレードする必要があります。

図1.1 Satellite 6.3 アップグレードパスの概要



サポート対象のアップグレードパス:

- Satellite 6.2.11 以降 → Satellite 6.3
- Satellite 6.1.9 以降 → Satellite 6.2.11 以降 → Satellite 6.3
- Satellite 6.0.x → Satellite 6.1.9 以降 → Satellite 6.2.11 以降 → Satellite 6.3



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 6.3 にアップグレードする手順の概要です。以下の要件の中で、お使いのインストールが満たしているものについては省くことができます。たとえば、Satellite 6.2.11 がすでにインストールされている場合は、Satellite 6.2.11 にアップデートする手順は必要ありません。

1. Satellite Server と全 Capsule Server を、Satellite 6.0.x から Satellite 6.1.9 にアップグレードします。詳細は『[Red Hat Satellite 6.1 インストールガイド](#)』の「[Red Hat Satellite Server および Capsule Server のアップグレード](#)」を参照してください。
2. Satellite 6.1 がインストールされている場合は、Satellite Server と全 Capsule Server を、Satellite 6.1.9 以降または Satellite 6.2.11 以降にアップグレードします。詳細は、『[Red Hat Satellite 6.2 インストールガイド](#)』の「[Red Hat Satellite Server および Capsule Server のアップグレード](#)」を参照してください。
3. Satellite 6.2.11 より前の Satellite 6.2.x がインストールされている場合は、Satellite Server と全 Capsule Server を Satellite 6.2.11 リリースに更新する必要があります。Red Hat Enterprise Linux 6 の Satellite Server および Capsule Server は Satellite 6.2.13 に更新する必要があります。詳細は『[Red Hat Satellite 6.2 インストールガイド](#)』の「[Satellite Server、Capsule Server、およびコンテンツホストの更新](#)」を参照してください。
4. Satellite Server と全 Capsule Server を Red Hat Enterprise Linux 7 に移行します。詳細は「[Red Hat Enterprise Linux 6 から Red Hat Enterprise Linux 7 へのマイグレーション](#)」を参照してください。
5. Satellite Server と全 Capsule Server を Satellite 6.3 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。
6. 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

自己登録した Satellite は、Red Hat Satellite でサポートされなくなりました。自己登録した Satellite Server は Red Hat コンテンツ配信ネットワークに移行する必要があります。自己登録 Satellite の詳細は「[自己登録 Satellite のマイグレート](#)」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 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 6.3 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 6.3 にアップグレードします。

本章では、「[Red Hat Enterprise Linux 6 から Red Hat Enterprise Linux 7 へのマイグレーション](#)」などの前提条件、「[Satellite Server のアップグレード](#)」のシナリオ、「[Capsule Server のアップグレード](#)」のシナリオ、および「[アップグレード後のタスク](#)」を説明します。

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。

2.1. 自己登録 SATELLITE のマイグレート

Red Hat Satellite 6.3 は、自己登録された Satellite をサポートしません。自己登録された Satellite Server は、Red Hat コンテンツ配信ネットワークに移行する必要があります。

自己登録された Satellite を Red Hat コンテンツ配信ネットワークに移行するには、以下の手順を行います。

1. Satellite Server が Foreman で管理されているかどうかを確認します。
 - a. Satellite web UI で、**ホスト > すべてのホスト** に移動します。
 - b. 移行するホストの上で **編集** をクリックします。
 - c. 画面右上に **ホストの管理を解除** ボタンが表示されたら、そのボタンをクリックします。
2. 移行するベースシステムで、以下のコマンドを実行してサブスクリプションを削除し、ホストの登録を解除します。

```
# subscription-manager remove --all
# subscription-manager unregister
```

3. **katello-ca-consumer** パッケージと、そのパッケージのその他のバージョンを削除するには、以下のコマンドを実行します。

```
# yum remove 'katello-ca-consumer*'
```

4. Red Hat に登録するベースシステムを指定するには、以下のコマンドを実行して、バックアップ設定を復元します。

```
# cd /etc/rhsm
# mv rhsm.conf rhsm.satellite.back
# mv rhsm.conf.kat-backup rhsm.conf
```

5. Satellite Server を Red Hat カスタマーポータルに登録します。



注記

Red Hat Satellite 6 サブスクリプションが Satellite マニフェストに割り当てられている場合は、このサブスクリプションを削除して、必要なサーバーに Red Hat Satellite 6 サブスクリプションをアタッチできるようにする必要があります。マニフェストを検索するには、カスタマーポータルにログインし、[サブスクリプション](#)、[Satellite 組織](#) の順にクリックし、[全サブスクリプション管理アプリケーション](#) タブをクリックします。詳細は『[コンテンツ管理ガイド](#)』の「[サブスクリプションの管理](#)」を参照してください。

```
# subscription-manager register your_customer_portal_credentials
```

6. Satellite サブスクリプションをシステムに割り当てます。

```
# subscription-manager list --available --matches '*Red Hat  
Satellite*'  
# subscription-manager attach --pool=pool_id
```

7. サードパーティのリポジトリ、または間違った Red Hat リポジトリで問題が発生しないように、すべてのリポジトリを無効にします。

```
# subscription-manager repos --disable "*" 
```

8. Red Hat Satellite 6.2 でサポートされるリポジトリを有効にするには、以下のコマンドを実行します。

- Red Hat Enterprise Linux 6 の場合:

```
# subscription-manager repos --enable=rhel-6-server-rpms \  
--enable=rhel-server-rhsc1-6-rpms \  
--enable=rhel-6-server-satellite-6.2-rpms
```

- Red Hat Enterprise Linux 7 の場合:

```
# subscription-manager repos --enable=rhel-7-server-rpms \  
--enable=rhel-server-rhsc1-7-rpms \  
--enable=rhel-7-server-satellite-6.2-rpms
```

9. Red Hat コンテンツ配信ネットワーク以外の **yum** リポジトリからメタデータを削除します。

```
# yum clean all
```

Red Hat Enterprise Linux 6 を使用している場合は、[「Red Hat Enterprise Linux 6 から Red Hat Enterprise Linux 7 へのマイグレーション」](#)に進みます。

Red Hat Enterprise Linux 7 を使用している場合は、[「Satellite Server のアップグレード」](#)に進みます。

2.2. RED HAT ENTERPRISE LINUX 6 から RED HAT ENTERPRISE LINUX 7 へのマイグレーション

Red Hat Satellite 6.3 は、Red Hat Enterprise Linux 7 でのみサポートされます。Red Hat Enterprise Linux 6 で Red Hat Satellite 6 を実行している場合は、最初に Red Hat Enterprise Linux 7 に移行する必要があります。移行プロセスは、Red Hat Satellite 6.3 で導入された Satellite クローンツールを使用して、Satellite バックアップから既存の Satellite Server のクローンを作成します。

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 で Capsule Server を復元してから再設定します。

2.2.1. 除外

Satellite マイグレーションプロセスには、Red Hat Satellite 環境だけが含まれます。手動で設定した cron ジョブなど、Satellite 環境外で行ったカスタマイズについては、このガイドでは説明していません。

2.2.2. 作業を行う前に

用語

以下の手順では、次の用語を理解することが重要になります。

- 「ソースサーバー」は、既存の Satellite Server または Capsule Server を指します。
- 「ターゲットサーバー」は、Satellite Server または Capsule Server の移行先となる新しいサーバーです。

要件

Satellite Server または Capsule Server を移行する場合は、以下の要件が適用されます。

- Red Hat Enterprise Linux 7 の最小インスタンスのみをインストールします。Red Hat Enterprise Linux 7 ソフトウェアグループや、サードパーティーアプリケーションはインストールしないでください。『インストールガイド』の「[インストールのために環境を準備](#)」に記載されている前提条件を完了してください。
- ターゲットサーバーには、ソースサーバーからターゲットサーバーに移したバックアップファイルと、復元する際のバックアップファイルを保存するのに十分な容量が必要です。Pulp データが 500 GB を超える場合は、ソースサーバーからターゲットサーバーに Pulp データを直接コピーすることが推奨されます。通常は、Pulp データをコピーした方が、このデータから圧縮したアーカイブファイルを作成し、ターゲットサーバーで解凍するよりも時間がかかりません。いずれの方法も、詳しい情報は「[Satellite Server のマイグレーション](#)」を参照してください。

2.2.3. Satellite Server 移行の概要

Satellite Server または Capsule Server を移行する手順の概要は以下のようになります。

1. ソースの Satellite Server または Capsule Server のバックアップを作成します。
2. すべての Satellite サービスを停止します。
3. Satellite データをターゲットサーバーにコピーします。
4. ソースの Satellite Server または Capsule Server をシャットダウンします。
5. ソースの Satellite Server または Capsule Server サーバーを移行します。
6. ターゲットの Satellite Server または Capsule Server の設定を終了します。

7. ソースの Satellite Server または Capsule Server の使用を停止します。



警告

Capsule Server やホストとの不要な接続を行わないように、ターゲットサーバーは内部ネットワークから切断します。ソースサーバーの使用を停止したら、ターゲットサーバーを再接続します。

ISO ファイルからインストールする場合を除き、サブスクリプションをアクティブにし、Satellite をインストールするには、Red Hat コンテンツ配信ネットワーク (CDN) への接続が必要です。

2.2.4. Satellite Server のマイグレーション

以下の手順は、既存の Satellite Server インストールをターゲットサーバーに移行する方法の概要を説明します。

1. ソースサーバーで、現在の Red Hat Satellite サブスクリプションを確認します。

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

後で必要になるため、Red Hat Satellite サブスクリプションプール ID を書き留めます。

2. ソースサーバーで Red Hat Satellite 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. ソースサーバーでバックアップを実行し、Pulp データをターゲットサーバーにコピーします。Satellite サーバーで Pulp データのサイズを推計して、バックアップに Pulp データが含まれているかどうかを確認します。

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

Pulp データが 500 GB に満たない場合

- a. MongoDB と PostgreSQL の両データベースをアクティブにし、Pulp データとともにオンラインバックアップを実行します。

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

この例では、バックアップファイルは **/backup** ディレクトリーに出力されます。

- b. 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}'); do
chkconfig $i off;done
```

- c. Pulp データをターゲットサーバーにコピーします。

```
# scp /backup/pulp_data.tar \
root@target_server.example.com:/backup/
# rsync --archive --partial --progress --compress \
/var/lib/qpidd target_server.example.com:/var/lib/qpidd
```

Pulp データが 500 GB を超える場合

- a. MongoDB と PostgreSQL の両データベースをアクティブにし、Pulp データは除外してオンラインバックアップを実行します。

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

この例では、バックアップファイルは **/backup** ディレクトリーに出力されます。

- b. 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}'); do
chkconfig $i off;done
```

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

4. リモート実行がソースサーバーで有効になっている場合は、**/usr/share/foreman-proxy/.ssh** ディレクトリーのコンテンツを、ターゲットサーバーにある同じディレクトリーにコピーします。ホストへの Satellite Server リモートアクセスを有効にするには、このディレクトリーに保存されている SSH 秘密鍵が必要になります。詳細は **Red Hat Satellite の『ホストの管理』** の「**ホストでのジョブの実行**」を参照してください。
5. ターゲットサーバーで、共有ストレージをマウントするか、ソースの Satellite Server から以下のファイルおよびディレクトリーをコピーします。

- **candlepin.dump**
- **config_files.tar.gz**
- **foreman.dump**
- **mongo_dump** folder

6. ソースサーバーのホストになっている Red Hat Enterprise Linux サーバーをシャットダウンします。
7. カスタマーポータルにターゲットサーバーを登録し、必要なサブスクリプションをアタッチし、Satellite リポジトリを有効にします。『[Red Hat Satellite 6.2 インストールガイド](#)』の「[接続済みネットワークからの Satellite Server のインストール](#)」を参照してください。

- a. カスタマーポータルにターゲットサーバーを登録します。

```
# subscription-manager register your_customer_portal_credentials
```

- b. ターゲットサーバーにサブスクリプションをアタッチします。

```
# subscription-manager attach --pool=pool_ID
```

Red Hat Satellite サブスクリプションのプール ID は、前の手順で確認しました。

- c. 必要なりポジトリ **だけ** が有効になっていることを確認します。

```
# 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-6.2-rpms
```

8. ターゲットサーバーに **satellite-clone** パッケージをインストールします。

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

9. ターゲットサーバーで、**satellite-clone** 設定ファイルをカスタマイズします。
/etc/satellite-clone/satellite-clone-vars.yml 設定ファイルを編集します。デフォルトから項目を変更するには、コメント記号 (#) を削除して、値を変更します。

- a. **backup_dir** 値を、バックアップファイルが含まれるパスに変更します。

- b. **rhel_migration** の値を **true** に変更します。

- c. その他の値を、要件に合わせて変更します。

10. ターゲットサーバーで、**satellite-clone** ツールを実行します。

```
# satellite-clone
```

11. ターゲットサーバーで、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。
Satellite web UI で DHCP、DNS、TFTP を再設定し、有効にします。詳細は『[インストールガイド](#)』の「[Satellite Server での DNS、DHCP、および TFTP の設定](#)」を参照してください。

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
```

12. **admin** ユーザー (パスワードは **changeme**) で Satellite Web UI にログインし、**admin** ユーザーのパスワードを変更します。
13. Satellite マニフェストを更新します。
 - a. Satellite web UI にログインします。
 - b. 正しい組織が選択されていることを確認します。
 - c. **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。
 - d. **マニフェストの更新** をクリックします。
 - e. **コンテンツ > Red Hat サブスクリプション** に移動して、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。
14. ライフサイクルと Capsule の関連を編集します。
 既存のインフラストラクチャーが中断されるのを回避するため、クローンプロセスにより、Capsule Server とそのライフサイクル環境との関連が壊れます。**/usr/share/satellite-clone/logs/reassociate_capsules.txt** ファイルの手順に従って、ターゲットの Satellite Server でこの変更を元に戻します。
15. ネットワーク設定 (DNS など) を、ターゲットサーバーの IP アドレスとホスト名に合わせて更新します。
16. 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。
 - a. virt-who デーモンに必要な手順を完了します。詳細は『[仮想インスタンスガイド](#)』の「[前提条件](#)」を参照してください。
 - b. ソースサーバーの **/etc/virt-who.d/** ディレクトリーにある virt-who 設定ファイルを、ターゲットサーバーの同じディレクトリーにすべてコピーします。
 - c. virt-who デーモンを設定して起動します。詳細は『[仮想インスタンスガイド](#)』の「[virt-who サービスの設定および起動](#)」を参照してください。
17. 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
```

18. ソースサーバーの使用を停止します。

2.2.5. Capsule Server のマイグレーション

以下の手順は、既存の Capsule Server インストールをターゲットサーバーに移行する方法の概要を説明します。

1. Red Hat Enterprise Linux 7 サーバーをインストールします。これは、新しい Capsule Server になります。

2. Capsule Server を新たにインストールしますが、**satellite-installer** は実行しないでください。詳細は『インストールガイド』の「[Capsule Server のインストール](#)」を参照してください。「**Capsule Server のインストール**」が終了したら、その先には進まないでください。
3. ソースの Capsule Server のバックアップを取得します。詳細は『**Red Hat Satellite 6.2 サーバー管理ガイド**』の「[バックアップおよび災害復旧](#)」を参照してください。
4. ソースの Capsule Server にある Satellite サービスをすべて停止します。

```
# katello-service stop
```

5. ソースサーバーからターゲットサーバーに、バックアップファイルをすべてコピーします。
6. ターゲットサーバーで Capsule Server を復元します。
7. ソースの Capsule Server をシャットダウンします。
8. ネットワーク設定 (DNS など) を、ターゲットサーバーの IP アドレスとホスト名に合わせて更新します。
9. ソースの Capsule Server の使用を停止します。

2.3. SATELLITE SERVER のアップグレード

このセクションは、Satellite Server をアップグレードする方法を説明します。

Red Hat Satellite 6.3 に **foreman-maintain** が追加されました。このツールは、Satellite をアップデートまたはアップグレードし、実行し続けるための様々な機能を提供します。このツールは、複数のバージョンの Satellite および Capsule をサポートします。

作業を行う前に

- Red Hat Enterprise Linux 7 では、Red Hat Satellite Server 6.2 のマイナーバージョンである 6.2.11 以降にアップグレードする必要があります。Satellite Server および Capsule Server が Red Hat Enterprise Linux 6 で実行している場合は、6.2.13 以降にする必要があります。Satellite 6.3 にアップグレードする場合は、Red Hat Satellite のバージョンを上記のバージョンにアップグレードする必要はありません。上記のバージョンより前のマイナーバージョンから直接アップグレードすることはサポートされていません。詳細は『**Red Hat Satellite 6.2 インストールガイド**』の「[Satellite Server と Capsule Server のアップグレード](#)」を参照してください。
- 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『**Red Hat Satellite インストールガイド**』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、バックアップを作成します。推奨されるのはクローン作成です。クローンを作成することで、将来の更新またはアップグレードが上書きされなくなります。テンプレートを変更したことがあるかどうかを確認するには、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。



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あと、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。

自己登録された Satellite をアップグレードすることはできません。自己登録された 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。自己登録された Satellite を CDN に移行する方法は「[自己登録 Satellite のマイグレート](#)」を参照してください。

2.3.1. 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。

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。
バックアップの詳細は、『Red Hat Satellite 6.2 サーバー管理ガイド』の「[バックアップおよび災害復旧](#)」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. 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。
4. 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。
5. Satellite Web UI に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 6.3 (for RHEL7 Server) (RPMs)** を探して、展開します。
 - d. **Red Hat Satellite Tools 6.3 for RHEL 7 Server RPMs x86_64** を選択します。
6. 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. 6.3 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。
7. 6.2 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、6.3 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 6.3 になります。
8. サブスクリプションを更新します。


```
# subscription-manager refresh
```
9. Satellite Maintenance リポジトリを有効にします。

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

10. **foreman-maintain** をインストールします。

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

11. 利用可能なバージョンを確認して、6.3 が記載されているのを確認します。

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

12. 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 6.3
```

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。

13. 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。

14. アップグレードを実行します。

```
# foreman-maintain upgrade run --target-version 6.3
```

15. 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あと、以下を入力します。

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

16. 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。

17. 変更を加えた場合は、Katello サービスを再起動します。

```
# katello-service restart
```

18. 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
```

2.3.2. 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。

作業を行う前に

- Red Hat Enterprise Linux 7 では、Red Hat Satellite Server 6.2 のマイナーバージョンである 6.2.11 以降にアップグレードする必要があります。Satellite Server および Capsule Server が Red Hat Enterprise Linux 6 で実行している場合は、6.2.13 以降にする必要があります。Satellite 6.3 にアップグレードする場合は、Red Hat Satellite のバージョンを上記のバージョンにアップグレードする必要はありません。上記のバージョンより前のマイナーバージョンから直接アップグレードすることはサポートされていません。詳細は『[Red Hat Satellite 6.2 インストールガイド](#)』の「[Satellite Server と Capsule Server のアップグレード](#)」を参照してください。
- 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『[Red Hat Satellite インストールガイド](#)』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、変更を上書きしたくない場合は、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
```

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. Katello サービスを停止します。

```
# katello-service stop
```

8. 『[Red Hat Satellite インストールガイド](#)』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。
9. 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. Katello サービスを停止します。

```
# katello-service stop
```

10. 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。
11. **--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 satellite --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/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。

12. 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。
13. 前の手順で変更を行った場合は、Katello サービスを再起動します。

```
# katello-service restart
```

14. 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
```

15. 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。

2.4. CAPSULE SERVER のアップグレード

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

作業を行う前に

- 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 のマイナーバージョンは 6.2.11 以降にする必要があります。Capsule Server が Red Hat Enterprise Linux 6 で実行している場合は、6.2.13 以降にする必要があります。以前のマイナーバージョンから直接アップグレードすることはサポートされていません。詳細は『[Red Hat Satellite 6.2 インストールガイド](#)』の「[Satellite のマイナーリリース間のアップグレード](#)」を参照してください。
- Red Hat Satellite Capsule 6.3 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『[Red Hat Satellite インストールガイド](#)』の「[ポートと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。
バックアップの詳細は、『Red Hat Satellite 6.2 Server 管理ガイド』の「[バックアップおよび災害復旧](#)」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、変更を上書きしたくない場合は、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 6.2 のリポジトリを無効にします。

```
# subscription-manager repos \
--disable rhel-7-server-satellite-capsule-6.2-rpms
```

5. 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-6.3-rpms \
--enable rhel-server-rhsc1-7-rpms \
--enable rhel-7-server-satellite-tools-6.3-rpms
```

6. 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。

7. リポジトリキャッシュを削除します。

```
# yum clean all
```

8. Katello サービスを停止します。

```
# katello-service stop
```

9. すべてのパッケージを更新します。

```
# yum update
```

10. Satellite Server で、新しい証明書でアーカイブを生成し、証明書バンドルを更新します。

● Satellite 証明書を使用する場合は、以下のコマンドを実行します

- Satellite 証明書を使用する場合、以下のコメントを実行します。

```
# capsule-certs-generate \
--foreman-proxy-fqdn mycapsule.example.com \
--certs-tar mycapsule.example.com-certs.tar \
--certs-update-all
```

mycapsule.example.com を、Capsule Server の完全修飾ドメイン名に置き換える必要があります。

- カスタム証明書を追加する場合は、以下のコマンドを実行します。

```
# capsule-certs-generate \
--foreman-proxy-fqdn mycapsule.example.com \ ①
--certs-tar /root/capsule_cert/capsule_certs.tar \ ②
--server-cert /root/capsule_cert/capsule_cert.pem \ ③
--server-cert-req /root/capsule_cert/capsule_cert_csr.pem \ ④
--server-key /root/capsule_cert/capsule_cert_key.pem \ ⑤
--server-ca-cert /root/sat_cert/ca_cert_bundle.pem \ ⑥
--certs-update-server --certs-update-server-ca
```

- ① Capsule Server の完全修飾ドメイン名。
- ② このコマンドが生成する証明書アーカイブが含まれる tar ファイルのパス。
- ③ Capsule Server 証明書ファイルのパス。
- ④ Capsule Server の証明書署名要求ファイルのパス。
- ⑤ Capsule Server の秘密鍵ファイルのパス。
- ⑥ 認証局バンドルのパス。

11. アーカイブファイルを Capsule Server にコピーします。

```
# scp mycapsule.example.com-certs.tar mycapsule.example.com:~/
```



警告

アップデートが終了しても証明書のアーカイブファイルは削除しないでください。将来のアップデートで必要になります。

12. 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
```

13. 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。

14. **--upgrade** オプションを付けてインストーラスクリプトを実行してアップグレードを実行し、Satellite Server で作成しておいた証明書アーカイブへのパスを指定します。また、上記で証明書のアーカイブを生成する際に **--certs-update-all** オプションを使用しているため、NSS データベースを再生成して、アップデートした証明書バンドルをデプロイするのに、**--certs-regenerate true**、**--certs-deploy true**、および **--certs-update-all** の 3 つのコマンドオプションが必要になります。

```
# satellite-installer --scenario capsule --upgrade \
--foreman-proxy-content-certs-tar mycapsule.example.com-certs.tar \
--certs-update-all --certs-regenerate true --certs-deploy true
```



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。

15. 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。
16. 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。

2.5. 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。

2.5.1. 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。この手順は、ホストの PXE 起動を使用して Satellite が新しいホストを検出するようにしていない場合は必要ありません。

2.5.1.1. 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。

2.5.1.2. 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. 更新が行われたら、Katello 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。

2.5.2. 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 6.2 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 6.3 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。このパッケージをクライアントにインストールすることはできますが、**katello-agent** をアップデートしなければエラーをインストールすることができません。

クライアントに Satellite Server との完全互換性を持たせるために、速やかに、すべてのクライアントの **katello-agent** を新しいバージョンにアップグレードします。このため、Satellite Tools リポジトリを 6.2 から 6.3 に変更する必要があります。これは手動で行うか、**satellite-6.3-tools-upgrade** パッケージをインストールすることで完了します。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。

作業を行う前に

- 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。

satellite-6.3-tools-upgrade パッケージを使用した 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 **パッケージの管理** を選択します。
3. 検索フィールドに、パッケージ名 **satellite-6.3-tools-upgrade** を入力します。
4. **インストール** 一覧から、希望するインストール方法を選択します。この操作が適切にスケジュールされたことを示すメッセージが表示されたら、次の手順に進みます。
5. 検索フィールドに、パッケージ名 **katello-agent** が入力されます。
6. **更新** 一覧から、希望する更新方法を選択します。この操作が適切にスケジュールされたことを示すメッセージが表示されたら、次の手順に進みます。

7. **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-6.2-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-6.3-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
```

2.5.3. 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。

作業を行う前に

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。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。

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

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。

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

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

Satellite 6.3 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 6.2 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 6.2 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 リポジトリ** に移動します。
2. **RPM** タブの **製品** で Red Hat Enterprise Linux Server を探して、展開します。
3. **リポジトリセット** で Red Hat Satellite Tools 6.2 (for Red Hat Enterprise Linux 7 Server) (RPM) を探して展開します。
4. Red Hat Enterprise Linux 7 Server RPM x86_64 用の Red Hat Satellite Tools 6.2 の選択を解除します。

チェックボックスが選択できない場合は、リポジトリがコンテンツビューに追加されたままになっています。リポジトリの孤立したパッケージは、スケジュールされているタスク (cron ジョブ) により自動的に削除されます。

2.5.5. PXE ローターオプションのアップグレード

Red Hat Satellite 6.3 に、UEFI システムの PXE ブートのサポートが追加されました。Satellite 6.3 にアップグレードすると、プロビジョニングしたホストに **PXE ローター** オプションが新たに追加されます。アップグレードしたホストでは、**PXE ローター** オプションが **none** にデフォルト設定され、PXE ベースのプロビジョニングが停止します。ホストが引き続き動作するようにするには、ホストに対して **PXE loader** オプションの **PXELinux BIOS** または **PXELinux UEFI** のいずれかを選択する必要があります。

システムの PXE ブート設定をアップデートするには、以下を行います。

1. Satellite web UI で、ホスト > **すべてのホスト** に移動します。
2. アップデートするホストの上で、**編集** をクリックします。
3. **編集** ページの右上で、**ホストの管理** を選択します。
4. **オペレーティングシステム** タブを選択します。
5. **PXE ローター** の一覧から、PXE を起動する方法を選択します。
6. **送信** をクリックして保存します。

2.5.6. テンプレートの更新

Ruby の **to_proc** 構文を含む 6.2 で作成したクローンテンプレートまたはカスタムテンプレート (例: **(1..3).collect(&:to_s)**) は、セキュリティが強化された Satellite 6.3 では動作しません。同じコードを Ruby ブロック全体 (**(1..3).collect {|num| num.to_s}**) として記述する必要があります。

1. 影響を受ける、**&:** が含まれている行を検索します。
2. **your_code(&:your_code)** になっている文字列を **your_code{|i| i.your_code}** に置き換えます。

以下は例になります。

6.2 のクローンテンプレートにおける Ruby 構文:

```
<% host_param('ssh_authorized_keys').split(',').map(&:strip).each do
|ssh_key| -%>
```


Satellite 6.3 に更新した Ruby 構文:

```
<% host_param('ssh_authorized_keys').split(',').map{ |item| item.strip
}.each do |ssh_key| -%>
```

6.2 のクローンテンプレートにおける Ruby 構文:

```
nameserver=#{[subnet.dns_primary,
subnet.dns_secondary].select(&:present?).join(',')}
```

Satellite 6.3 に更新した Ruby 構文:

```
nameserver=#{[subnet.dns_primary, subnet.dns_secondary].select{ |item|
item.present? }.join(',')}
```

2.5.6.1. テンプレートの完了

以前は、テンプレートは `<%= foreman_url %>` を使用して、ビルドが完了したことを Satellite に通知していました。6.3 のテンプレートでは `<%= foreman_url('built') %>` を使用して、ビルドテンプレートを明示的に呼び出します。

カスタムのプロビジョニング完了テンプレートを 6.2 でクローンを作成または作成した場合は、`<%= foreman_url %>` のテンプレートを探して `<%= foreman_url('built') %>` に変更します。

2.5.7. 外部の DHCP サービスのアップグレード

外部の DHCP サーバーを使用している場合は、Satellite 6.2 で使用していた **isc** DHCP プロバイダーではなく、**remote_isc** DHCP プロバイダーを使用するように **satellite-installer** を設定する必要があります。以下のセクションでは、Satellite 6.3 で外部の DHCP サービスを設定する方法を説明します。

- 『インストールガイド』の「[Satellite Server での外部 DHCP の設定](#)」
- 『インストールガイド』の「[Capsule Server での外部 DHCP の設定](#)」

第3章 PUPPET のアップグレード

Red Hat Satellite 6.3 はデフォルトで Puppet 3 を使用しますが、Puppet 4 にアップグレードすることもできます。Satellite 6.4 では Puppet 5 だけがサポートされます。Puppet 5 へのマイグレーションパスは、Puppet 4 から Puppet 5 へのマイグレーションだけがサポートされます。したがって、バージョン 4 以前の Puppet を使用している場合は、Puppet 4 にアップグレードしてから Satellite 6.4 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るため、Satellite 6.3 で Puppet 4 を行ってから次のステップに進むことを Red Hat は推奨しています。本章では、Puppet 4 へのアップグレードプロセスを説明します。

3.1. アップグレードパス

Puppet をアップグレードする前に、Puppet モジュールが Puppet 4 と互換性があるのを確認してから、Puppet モジュールを変更します。たとえば、Puppet エージェントを実行した後に、クライアントに空白の設定があると、Puppet モジュールのアップデートが指定されます。

Red Hat Satellite と Capsule を、同時に Puppet 4 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、テストを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 3 Capsule から Puppet 4 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。

完全に機能させるには、ホストに Puppet 4 エージェントを使用する必要がありますが、Capsule を Puppet 4 にアップグレードしてから、ホストを Puppet 4 エージェントにアップデートする必要があります。

ホストのプロビジョニング時に、Puppet 3 エージェントの代わりに Puppet 4 エージェントをインストールするように Satellite を設定できます。キックスタートのテンプレートファイルに **enable-puppet4** パラメーターを追加します。**true** に設定すると、プロビジョニングテンプレートが Puppet 4 エージェントをインストールして、**/etc/puppetlabs/puppet/** に **puppet.conf** を設定します。パラメーターを定義しないと、Puppet 3 エージェントがインストールされます。

ホストで強制的に Puppet 4 を使用するには、ホストグループレベルで **enable-puppet4** パラメーターを作成し、値を **true** に設定します。

3.2. PUPPET モジュールのテスト

アップグレードを行う前に、以下のガイドラインに沿って Puppet モジュールをテストします。

1. 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。
2. **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。
3. Puppet 3 を実行している Satellite Capsule で、将来のパarserを使用するように設定し、再度テストします。この機能を有効にするには、**--puppet-server-parser future** オプションで Satellite インストーラーを実行します。

Puppet 3.8 における将来のパarserは、Puppet 4.0 で使用している Puppet 言語と同一の機能を有効にします。それを有効にしたあと、**/var/log/puppet/** ログでエラーを確認します。エラーを見つけたら修正して、再度テストします。将来のパarserの情報は「[Puppet 3.8 Reference Manual](#)」を参照して

ください。

3.3. アップグレードの実行

ここでは、Satellite または Capsule を Puppet 4 にアップグレードする方法を説明します。新規にインストールする場合は『[Red Hat Satellite インストールガイド](#)』を参照してください。

作業を行う前に

- Satellite 6.3 へのアップグレードが完了したことを確認します。
- Puppet モジュールを確認して、必要に応じて Puppet 4 にアップグレードします。
- バージョン 4 ではなく、最新の Puppet 3 エージェントにアップグレードするシステムにすべてのホストを接続していることを確認します。ホストを Puppet 4 エージェントにアップグレードしてから、Capsule をアップグレードしてください。Satellite Tools リポジトリは Puppet パッケージを提供します。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。
- システムのバックアップ、またはスナップショットを作成します。詳細は『[Red Hat Satellite の管理](#)』の「[Satellite Server および Capsule Server のバックアップと復元](#)」を作成します。

Satellite または Capsule の Puppet 4 へのアップデート

1. Puppet 4 リポジトリを有効にします。
 - a. Satellite で、以下のコマンドを使用します。

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

- b. Capsule で、以下のコマンドを使用します。

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

2. Puppet をアップグレードします。

```
# satellite-installer --upgrade-puppet
```

ホストの Puppet 4 へのアップグレード

ホストをアップグレードする Puppet リポジトリは、Red Hat リリースまたはプラットフォームによって異なります。アーキテクチャーによって異なる場合もあります。

- 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** はたとえば **arm**、**power**、**hpc-node**、または **system-z** になります。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 |grep puppet4
```

2. 以下のように、ホストに適切な Puppet 4 リポジトリを有効にします。

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

4. Red Hat [Bugzilla 1517624](#) が解決するまでは、以下を実行してください。

```
# cp -rp /var/lib/puppet/ssl /etc/puppetlabs/puppet/
# cp /etc/puppet/puppet.conf.rpmsave
/etc/puppetlabs/puppet/puppet.conf
# sed -i 's|/var/lib/puppet|/opt/puppetlabs/puppet/cache|'
/etc/puppetlabs/puppet/puppet.conf
# sed -i 's|/var/log/puppet|/var/log/puppetlabs/puppet|'
/etc/puppetlabs/puppet/puppet.conf
# sed -i 's|/var/run/puppet|/var/run/puppetlabs|'
/etc/puppetlabs/puppet/puppet.conf
# sed -i 's|$vardir/ssl|etc/puppetlabs/puppet/ssl|'
/etc/puppetlabs/puppet/puppet.conf
```

第4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「[4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](#)』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。

4.1. 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。
- 設定ファイルに変更を加える場合は、手動または Hiera などのツールを使用します。変更した内容は、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設定が上書きされます。**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 6.3.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 6.3.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
```

4.2. 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-6.3-rpms
rhel-server-rhsc1-7-rpms
```

```
# rhel-7-server-satellite-tools-6.3-rpms
```

リポジトリを無効または無効にする方法は、『Red Hat Satellite インストールガイド』の「[リポジトリの設定](#)」を参照してください。**rhel-7-server-satellite-tools-6.3-rpms** リポジトリは Katello エージェントを提供します。詳細は『Red Hat Satellite インストールガイド』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が、リモート実行機能を使用する場合は必要になります。

2. Katello 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
```

4.3. 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
```

