



Red Hat Satellite 6.5

Content Management Guide

Red Hat およびカスタムソースのコンテンツ管理ガイド

Red Hat Satellite 6.5 Content Management Guide

Red Hat およびカスタムソースのコンテンツ管理ガイド

Red Hat Satellite Documentation Team

satellite-doc-list@redhat.com

法律上の通知

Copyright © 2019 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, the Red Hat 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 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

本書を使用して、RPM ファイル、ISO イメージ、Puppet モジュールなどの、Satellite 6 のコンテンツを理解し、管理します。Red Hat Satellite 6 は、アプリケーションライフサイクル全体でプロモートされた一連のコンテンツビューを使用してこのコンテンツを管理します。本書では、それぞれの組織に合わせたアプリケーションライフサイクルとライフサイクル環境内でホストの状態に合致するコンテンツビューの作成方法を説明します。これらのコンテンツビューは、最終的に Red Hat Satellite 6 環境でホストをプロビジョニングおよび更新する基礎となります。

目次

第1章 INTRODUCTION	5
1.1. CONTENT MANAGEMENT TYPES OVERVIEW	5
第2章 MANAGING ORGANIZATIONS	6
2.1. CREATING AN ORGANIZATION	6
2.2. SETTING THE ORGANIZATION CONTEXT	7
2.3. CREATING AN ORGANIZATION DEBUG CERTIFICATE	7
2.4. BROWSING REPOSITORY CONTENT USING AN ORGANIZATION DEBUG CERTIFICATE	8
2.5. DELETING AN ORGANIZATION	9
第3章 MANAGING LOCATIONS	10
3.1. CREATING A LOCATION	10
3.2. SETTING THE LOCATION CONTEXT	10
3.3. DELETING A LOCATION	11
第4章 MANAGING SUBSCRIPTIONS	12
4.1. IMPORTING A SUBSCRIPTION MANIFEST INTO SATELLITE SERVER	12
4.2. LOCATING A SUBSCRIPTION IN THE SATELLITE WEB UI	13
4.3. ADDING SUBSCRIPTIONS TO SUBSCRIPTION ALLOCATIONS IN THE SATELLITE WEB UI	13
4.4. REMOVING SUBSCRIPTIONS FROM SUBSCRIPTION ALLOCATIONS IN THE SATELLITE WEB UI	14
4.5. UPDATING AND REFRESHING SUBSCRIPTION MANIFESTS	14
4.6. ATTACHING SUBSCRIPTIONS TO CONTENT HOSTS	15
4.7. BULK UPDATING CONTENT HOSTS' SUBSCRIPTIONS	16
第5章 IMPORTING RED HAT CONTENT	18
5.1. PRODUCTS IN RED HAT SATELLITE	18
5.2. CONTENT SYNCHRONIZATION OVERVIEW	18
5.3. DISABLING THE GLOBAL HTTP PROXY	18
5.4. DOWNLOAD POLICIES OVERVIEW	18
5.5. CHANGING THE DEFAULT DOWNLOAD POLICY	19
5.6. CHANGING THE DOWNLOAD POLICY FOR A REPOSITORY	20
5.7. SELECTING RED HAT REPOSITORIES TO SYNCHRONIZE	20
5.8. SYNCHRONIZING RED HAT REPOSITORIES	22
5.9. RECOVERING A REPOSITORY	23
5.10. LIMITING SYNCHRONIZATION SPEED	24
5.11. CREATING A SYNCHRONIZATION PLAN	25
第6章 IMPORTING CUSTOM CONTENT	27
6.1. USING CUSTOM PRODUCTS IN SATELLITE	27
6.2. IMPORTING CUSTOM SSL CERTIFICATES	27
6.3. IMPORTING A CUSTOM GPG KEY	27
6.4. CREATING A CUSTOM PRODUCT	28
6.5. ADDING A CUSTOM RPM REPOSITORY	29
第7章 CREATING AN APPLICATION LIFE CYCLE	31
7.1. APPLICATION LIFE CYCLE OVERVIEW	31
7.2. PROMOTING CONTENT ACROSS THE APPLICATION LIFE CYCLE	32
7.3. CREATING AN APPLICATION LIFE CYCLE	33
7.4. REMOVING LIFE CYCLE ENVIRONMENTS FROM SATELLITE SERVER	34
7.5. REMOVING LIFE CYCLE ENVIRONMENTS FROM CAPSULE SERVER	35
7.6. ADDING LIFE CYCLE ENVIRONMENTS TO CAPSULE SERVERS	36
第8章 MANAGING CONTENT VIEWS	38

8.1. CREATING A CONTENT VIEW	39
8.2. VIEWING MODULE STREAMS	40
8.3. CREATING A CONTENT VIEW WITH A PUPPET MODULE	40
8.4. PROMOTING A CONTENT VIEW	41
8.5. COMPOSITE CONTENT VIEWS OVERVIEW	42
8.6. CREATING A COMPOSITE CONTENT VIEW	43
8.7. CONTENT FILTER OVERVIEW	44
8.8. CONTENT FILTER EXAMPLES	46
8.9. CREATING A CONTENT FILTER	47
第9章 SYNCHRONIZING CONTENT BETWEEN SATELLITE SERVERS	49
9.1. EXPORTING A CONTENT VIEW VERSION	49
9.2. IMPORTING A CONTENT VIEW VERSION	50
第10章 MANAGING ACTIVATION KEYS	52
10.1. CREATING AN ACTIVATION KEY	52
10.2. UPDATING SUBSCRIPTIONS ASSOCIATED WITH AN ACTIVATION KEY	54
10.3. CSV ファイルを使用してアクティベーションキーに関連付けられたサブスクリプションを更新する手順	55
10.4. USING ACTIVATION KEYS FOR HOST REGISTRATION	57
10.5. ENABLING AUTO-ATTACH	58
10.6. SETTING THE SERVICE LEVEL	58
第11章 MANAGING ERRATA	60
11.1. INSPECTING AVAILABLE ERRATA	60
11.2. SUBSCRIBING TO ERRATA NOTIFICATIONS	62
11.3. LIMITATIONS TO REPOSITORY DEPENDENCY RESOLUTION	62
11.4. CREATING A CONTENT VIEW FILTER FOR ERRATA	63
11.5. ADDING ERRATA TO AN INCREMENTAL CONTENT VIEW	64
11.6. APPLYING ERRATA TO A HOST	65
11.7. APPLYING ERRATA TO MULTIPLE HOSTS	66
第12章 MANAGING OSTREE CONTENT	68
12.1. SELECTING RED HAT OSTREE CONTENT TO SYNCHRONIZE	68
12.2. IMPORTING CUSTOM OSTREE CONTENT	69
12.3. MANAGING OSTREE CONTENT WITH CONTENT VIEWS	70
第13章 MANAGING ISO IMAGES	72
13.1. IMPORTING ISO IMAGES FROM RED HAT	72
13.2. IMPORTING INDIVIDUAL ISO IMAGES AND FILES	73
第14章 MANAGING CUSTOM FILE TYPE CONTENT	75
14.1. CREATING A CUSTOM FILE TYPE REPOSITORY IN RED HAT SATELLITE	75
14.2. CREATING A CUSTOM FILE TYPE REPOSITORY IN A LOCAL DIRECTORY	77
14.3. CREATING A REMOTE FILE TYPE REPOSITORY	79
14.4. UPLOADING FILES TO A CUSTOM FILE TYPE REPOSITORY IN RED HAT SATELLITE	80
14.5. DOWNLOADING FILES TO A HOST FROM A CUSTOM FILE TYPE REPOSITORY IN RED HAT SATELLITE	81
第15章 MANAGING CUSTOM PUPPET CONTENT	83
15.1. CREATING A CUSTOM PUPPET REPOSITORY	83
15.2. MANAGING INDIVIDUAL PUPPET MODULES	83
15.3. SYNCHRONIZING PUPPET REPOSITORIES	84
15.4. SYNCHRONIZING PUPPET MODULES FROM A GIT REPOSITORY	85
付録A USING AN NFS SHARE FOR CONTENT STORAGE	87

付録B IMPORTING CONTENT ISOS INTO A DISCONNECTED SATELLITE	89
付録C IMPORTING CONTENT ISOS INTO A CONNECTED SATELLITE	91
付録D SYNCHRONIZING TEMPLATES WITH GIT	94
D.1. ENABLING THE TEMPLATESYNC PLUG-IN	94
D.2. CONFIGURING THE TEMPLATESYNC PLUG-IN	94
D.3. IMPORTING AND EXPORTING TEMPLATES	95
D.3.1. Synchronizing Templates with a Git repository	96
D.3.2. Synchronizing templates with a local directory	97
D.4. ADVANCED GIT CONFIGURATION	98
D.5. UNINSTALLING THE PLUG-IN	98

第1章 INTRODUCTION

Satellite 6 では、**コンテンツ** は、システムにインストールされたソフトウェアとして定義されます。コンテンツには、ベースオペレーティングシステム、ミドルウェアサービス、エンドユーザーアプリケーションなどが含まれますが、これらに限定はされません。Red Hat Satellite 6 を使用して、ソフトウェアライフサイクルの各段階で Red Hat Enterprise Linux システムのさまざまな種類のコンテンツを管理できます。

Red Hat Satellite 6 manages the following content:

Subscription management

サブスクリプション管理は、Red Hat サブスクリプション情報を管理する方法を組織に提供します。

Content management

コンテンツ管理は、Red Hat コンテンツを保存し、さまざまな方法で整理する方法を組織に提供します。

1.1. CONTENT MANAGEMENT TYPES OVERVIEW

With Red Hat Satellite 6, you can manage the following Red Hat content types:

RPM Packages

RPM ファイルは、Red Hat サブスクリプションに関連するリポジトリからインポートします。Satellite Server を使用して、Red Hat のコンテンツ配信ネットワークから RPM ファイルをダウンロードし、ローカルに保存します。これらのリポジトリと RPM ファイルはコンテンツビューで使用できます。

Kickstart Trees

Import the kickstart trees for creating a system. New systems access these kickstart trees over a network to use as base content for their installation. Red Hat Satellite 6 also contains some predefined kickstart templates as well as the ability to create your own, which are used to provision systems and customize the installation.

Red Hat Satellite 6 では、カスタムコンテンツを管理することもできます。以下に例を示します。

ISO and KVM Images

Download and manage media for installation and provisioning. For example, Satellite downloads, stores and manages ISO images and guest images for specific Red Hat Enterprise Linux and non-Red Hat operating systems.

Puppet Modules

RPM コンテンツとともに Puppet モジュールをアップロードできるため、Puppet は、プロビジョニング後のシステムの状態を設定できます。また、ユーザーはプロビジョニングプロセスの一部として Puppet クラスとパラメーターを管理することもできます。

OSTree

You can import OSTree branches and publish this content to an HTTP location.

この手順を使用して、SSL 証明書や OVAL ファイルなど、必要なタイプのコンテンツのカスタムコンテンツを追加できます。

第2章 MANAGING ORGANIZATIONS

Organizations divide Red Hat Satellite 6 resources into logical groups based on ownership, purpose, content, security level, or other divisions. You can create and manage multiple organizations through Red Hat Satellite 6, then divide and assign your Red Hat subscriptions to each individual organization. This provides a method of managing the content of several individual organizations under one management system. Here are some examples of organization management:

Single Organization

システム管理チェーンが単純な小企業。この場合は、このビジネスに対して組織を1つ作成し、コンテンツをその組織に割り当てることができます。

Multiple Organizations

複数の小規模な事業単位を所有する大企業 (たとえば、独立したシステム管理およびソフトウェア開発グループがある会社)。この場合は、企業に対して、またこの企業が所有する各事業単位に対して組織を作成できます。こうすることで、それぞれのシステムインフラストラクチャーを分けることができます。各組織に、それぞれのニーズに基づいてコンテンツを割り当てることができます。

External Organizations

他の組織の外部システムを管理する会社 (たとえば、クラウドコンピューティングと Web ホスティングリソースを顧客に提供する会社)。この場合は、会社の独自のシステムインフラストラクチャーの組織に加え、外部の各会社に対して組織を作成できます。必要に応じて、各組織にコンテンツを割り当てることができます。

A default installation of Red Hat Satellite 6 has a default organization called **Default_Organization**.

New Users

新しいユーザーに、デフォルトの組織が割り当てられていない場合には、このユーザーのアクセス権限は制限されます。ユーザーにシステムの権限を付与するには、ユーザーをデフォルトの組織に割り当てます。ユーザーが次に Satellite にログオンしたときに、ユーザーのアカウントに適切なシステム権限が付与されます。

2.1. CREATING AN ORGANIZATION

Use this procedure to create an organization.

Procedure

To create an organization, complete the following steps:

1. Satellite Web UI で、**管理 > 組織** に移動します。
2. Click **New Organization**.
3. In the **Name** field, enter a name for the organization.
4. **ラベル** フィールドに、組織の一意的な ID を入力します。これは、コンテンツストレージ用ディレクトリーなどの特定のアセットを作成およびマッピングする場合に使用されます。文字、数字、アンダースコア、およびダッシュを使用し、スペースは使用しないでください。
5. オプション: **説明** フィールドに、組織の説明を入力します。
6. Click **Submit**.
7. If you have hosts with no organization assigned, select the hosts that you want to add to the organization, then click **Proceed to Edit**

- 編集 ページで、組織に追加するインフラストラクチャーリソースを割り当てます。このインフラストラクチャーリソースには、ネットワークリソース、インストールメディア、キックスタートテンプレートなどのパラメーターが含まれます。このページには、**管理 > 組織** に移動し、編集する組織を選択するといつでも戻ることができます。
- Click **Submit**.

For CLI Users

Enter the following command to create an organization:

```
# hammer organization create \  
--name "your_organization_name" \  
--label "your_organization_label" \  
--description "your_organization_description"
```

2.2. SETTING THE ORGANIZATION CONTEXT

An organization context defines the organization to use for a host and its associated resources.

Procedure

組織メニューは、Satellite Web UI の左上にあるメニューバーの最初のメニュー項目です。現在の組織を選択していない場合には、メニューには**任意の組織**と表示されます。**任意の組織**ボタンをクリックして、使用する組織を選択します。

For CLI Users

While using the CLI, include either **--organization "your_organization_name"** or **--organization-label "your_organization_label"** as an option. For example:

```
# hammer subscription list --organization "Default_Organization"
```

This command outputs subscriptions allocated for the Default_Organization.

2.3. CREATING AN ORGANIZATION DEBUG CERTIFICATE

If you require a debug certificate for your organization, use the following procedure.

Procedure

組織のデバッグ証明書を作成するには、以下の手順を実行します。

- Satellite Web UI で、**管理 > 組織** に移動します。
- Select an organization that you want to generate a debug certificate for.
- 生成してダウンロード** をクリックします。
- Save the certificate file in a secure location.

Debug Certificates for Provisioning Templates

Debug Certificates are automatically generated for provisioning template downloads if they do not already exist in the organization for which they are being downloaded.

2.4. BROWSING REPOSITORY CONTENT USING AN ORGANIZATION DEBUG CERTIFICATE

組織のデバッグ証明書がある場合には、Web ブラウザーまたは API を使用して、組織のリポジトリ コンテンツを表示できます。

Prerequisites

1. 「[Creating an Organization Debug Certificate](#)」に記載されている手順で、組織の証明書を作成およびダウンロードします。
2. Open the X.509 certificate, for example, for the default organization:

```
$ vi 'Default Organization-key-cert.pem'
```

3. Copy the contents of the file from **-----BEGIN RSA PRIVATE KEY-----** to **-----END RSA PRIVATE KEY-----**, into a **key.pem** file.
4. このファイルの **-----BEGIN CERTIFICATE-----** から **-----END CERTIFICATE-----** までを **cert.pem** ファイルにコピーします。

Procedure

ブラウザーを使用するには、X.509 証明書をご利用のブラウザーがサポートする形式にまず変換してから、証明書をインポートする必要があります。

For Firefox Users

To use an organization debug certificate in Firefox, complete the following steps:

1. To create a PKCS12 format certificate, enter the following command:

```
$ openssl pkcs12 -keypbe PBE-SHA1-3DES -certpbe PBE-SHA1-3DES -export -in cert.pem  
-inkey key.pem -out organization_label.pfx -name organization_name
```

2. In the Firefox browser, navigate to **Edit > Preferences > Advanced Tab**.
3. **証明書の表示** を選択し、**証明書** タブをクリックします。
4. **インポート** をクリックし、読み込む **.pfx** ファイルを選択します。
5. アドレスバーに、リポジトリの参照先の URL を以下の形式で入力します。

```
http://satellite.example.com/pulp/repos/organization_label
```

Pulp uses the organization label, therefore, you must enter the organization label into the URL.

For CURL Users

To use the organization debug certificate with CURL, enter the following command:

```
$ curl -k --cert cert.pem --key key.pem \  
http://satellite.example.com/pulp/repos/Default_Organization/Library/content/dist/rhel/server/7/7Serve  
r/x86_64/sat-tools/6.5/os/
```

cert.pem と **key.pem** へのパスが適切な絶対パスであることを確認します。間違っていると、エラーメッセージなしにコマンドが失敗します。

2.5. DELETING AN ORGANIZATION

組織は、ライフサイクル環境またはホストグループに関連付けられていない場合に削除できます。削除する組織にライフサイクル環境またはホストグループが関連付けられている場合は、**管理 > 組織** に移動して関連の組織をクリックします。インストール中に作成されたデフォルトの組織は、Satellite 環境で関連付けられていないホストへのプレースホルダーであるため、削除しないでください。環境には常に1つ以上の組織が必要です。

Procedure

To delete an organization, complete the following steps:

1. Satellite Web UI で、**管理 > 組織** に移動します。
2. 削除する組織名の右側にあるリストから **削除** を選択します。
3. Click **OK** to delete the organization.

For CLI Users

Enter the following command to delete an organization:

```
# hammer organization delete --organization "your_organization_name"
```

第3章 MANAGING LOCATIONS

ロケーションは、リソースを分類し、ホストを割り当てる方法を提供する点で組織に似ています。組織とロケーションは、以下の点が異なります。

- Locations are based on physical or geographical settings.
- Locations have a hierarchical structure.

3.1. CREATING A LOCATION

Use this procedure to create a location so that you can manage your hosts and resources by location.

Procedure

To create a location, complete the following steps:

1. In the Satellite web UI, navigate to **Administer > Locations**
2. **新規組ロケーション**をクリックします。
3. オプション: **親** リストから、親ロケーションを選択します。これにより、ロケーションの階層が作成されます。
4. **名前** フィールドに、ロケーションの名前を入力します。
5. オプション: **説明** フィールドに、ロケーションの説明を入力します。
6. Click **Submit**.
7. ホストにロケーションが割り当てられていない場合は、新しいロケーションに割り当てるホストを追加し、**Proceed to Edit (編集に進む)**をクリックします。
8. ロケーションに追加するインフラストラクチャーリソースを割り当てます。このインフラストラクチャーリソースには、ネットワークリソース、インストールメディア、キックスタートテンプレートなどのパラメーターが含まれます。**管理 > ロケーション** に移動して、編集するロケーションを選択すると、いつでもこのページに戻ることができます。
9. Click **Submit** to save your changes.

For CLI Users

Enter the following command to create a location:

```
# hammer location create \  
--parent-id "parent_location_id" \  
--name "your_location_name" \  
--description "your_location_description"
```

3.2. SETTING THE LOCATION CONTEXT

A location context defines the location to use for a host and its associated resources.

Procedure

ロケーションメニューは、Satellite Web UI の左上にあるメニューバーの2番目のメニュー項目です。現在のロケーションを選択していない場合には、メニューには **任意のロケーション** と表示されます。**任意のロケーション** をクリックして、使用するロケーションを選択します。

For CLI Users

While using the CLI, include either `--location "your_location_name"` or `--location-id "your_location_id"` as an option. For example:

```
# hammer subscription list --location "Default_Location"
```

This command outputs subscriptions allocated for the **Default_Location**.

3.3. DELETING A LOCATION

ロケーションは、ライフサイクル環境またはホストグループに関連付けられていない場合に削除できます。削除するロケーションにライフサイクル環境またはホストグループが関連付けられている場合は、**管理 > ロケーション** に移動して関連のロケーションをクリックします。インストール中に作成されたデフォルトのロケーションは、Satellite 環境で関連付けられていないホストへのプレースホルダーであるため、削除しないでください。環境には常に1つ以上のロケーションが必要です。

Procedure

To delete a location, complete the following steps:

1. In the Satellite web UI, navigate to **Administer > Locations**.
2. 削除するロケーションの名前の右側にあるリストから**削除**を選択します。
3. Click **OK** to delete the location.

For CLI Users

Enter the following command to delete a location:

```
# hammer location delete --location "your_location_name"
```

第4章 MANAGING SUBSCRIPTIONS

Red Hat Satellite 6 では、Red Hat のコンテンツ配信ネットワーク (CDN) からコンテンツをインポートします。Satellite 6 では、対応のリポジトリからコンテンツの検索、アクセス、ダウンロードを行うサブスクリプションマニフェストが必要です。サブスクリプションマニフェストには、Satellite Server の組織ごとにサブスクリプション割り当てを含める必要があります。サブスクリプション情報はすべて、Red Hat カスタマーポータルのアカウントで確認できます。

Before you can complete the tasks in this chapter, you must create a Subscription Manifest in the Customer Portal.

カスタマーポータルでサブスクリプションマニフェストを作成、管理、エクスポートするには、『[Red Hat Subscription Management の使用](#)』ガイドの「[マニフェストの使用](#)」を参照してください。

Use this chapter to import a Subscription Manifest and manage the manifest within the Satellite web UI.

Subscription Allocations and Organizations

複数のサブスクリプション割り当てがある場合は、複数の組織を管理できます。Satellite 6 では、Satellite Server で設定した組織ごとに1つの割り当てが必要です。この利点は、各組織が独立したサブスクリプションを保持するため、複数の組織をそれぞれ独自の Red Hat アカウントでサポートできることです。

Future-Dated subscriptions

You can use future-dated subscriptions in a subscription allocation. When you add future-dated subscriptions to content hosts before the expiry date of the existing subscriptions, you can have uninterrupted access to repositories.

現在のサブスクリプションの有効期限前に、コンテンツホストに未来の日付のサブスクリプションを手動でアタッチします。自動アタッチ機能は別の目的で設計されており、機能しない可能性があるため、この機能に依存しないでください。詳細は、「[Attaching Subscriptions to Content Hosts](#)」を参照してください。

4.1. IMPORTING A SUBSCRIPTION MANIFEST INTO SATELLITE SERVER

Use the following procedure to import a Subscription Manifest into Satellite Server.

Prerequisite

サブスクリプションマニフェストファイルがカスタマーポータルからエクスポートされていること。詳細は、『[Red Hat Subscription Management の使用](#)』ガイドの「[マニフェストの使用](#)」を参照してください。

Procedure

To import a Subscription Manifest, complete the following steps:

1. In the Satellite web UI, ensure the context is set to the organization you want to use.
2. Navigate to **Content > Subscriptions**.
3. In the Red Hat Subscriptions window, click **Manage Manifest**.
4. マニフェストの管理ウィンドウで、**参照**をクリックします。

5. サブスクリプションマニフェストファイルが保存されている場所へ移動して、**表示** をクリックします。管理マニフェストウィンドウが自動的に終了しない場合は、**終了** をクリックして Red Hat サブスクリプションウィンドウに戻ります。

For CLI Users

1. Copy the Subscription Manifest file to Satellite Server:

```
[user@client ~]$ scp ~/manifest_file.zip root@satellite.example.com:~/.
```

2. Connect to Satellite Server as the root user, and then import the Subscription Manifest file:

```
[root@satellite ~]# hammer subscription upload \  
--file ~/manifest_file.zip \  
--organization "organization_name"
```

リポジトリを有効化して、Red Hat コンテンツをインポートできるようになります。詳細は、[5章 Importing Red Hat Content](#) を参照してください。

4.2. LOCATING A SUBSCRIPTION IN THE SATELLITE WEB UI

When you import a Subscription Manifest into Satellite Server, the subscriptions from your manifest are listed in the Red Hat Subscriptions window. If you have a high volume of subscriptions, you can filter the results to find a specific subscription.

Prerequisite

サブスクリプションマニフェストファイルは、Satellite Server にインポートしておく必要があります。詳細は、「[Importing a Subscription Manifest into Satellite Server](#)」を参照してください。

Procedure

To locate a subscription, complete the following steps:

1. In the Satellite web UI, ensure the context is set to the organization you want to use.
2. Navigate to **Content > Subscriptions**.
3. Red Hat サブスクリプションウィンドウで、**検索** フィールドをクリックし、検索条件の一覧を表示して検索クエリーをビルドします。
4. Select search criteria to display further options.
5. When you have built your search query, click the search icon.

たとえば、**検索** フィールドにカーソルを置き、**expires (期限切れ)** を選択して、スペースキーを押すと、別のリストが表示され、> <または = 文字を選択できます。>を選択してスペースキーを押すと、自動オプションの別のリストが表示されます。独自の条件を入力することも可能です。

4.3. ADDING SUBSCRIPTIONS TO SUBSCRIPTION ALLOCATIONS IN THE SATELLITE WEB UI

以下の手順を使用して、Satellite Web UI でサブスクリプション割り当てにサブスクリプションを追加します。

Prerequisite

サブスクリプションマニフェストファイルは、Satellite Server にインポートしておく必要があります。詳細は、「[Importing a Subscription Manifest into Satellite Server](#)」を参照してください。

Procedure

To add subscriptions to a subscription allocation, complete the following steps:

1. In the Satellite web UI, ensure the context is set to the organization you want to use.
2. Navigate to **Content > Subscriptions**.
3. In the Red Hat Subscriptions window, click **Add Subscriptions**.
4. 追加するサブスクリプションの各行で、**Quantity to Allocate (割り当てる数量)** 列に数量を入力します。
5. Click **Submit**.

4.4. REMOVING SUBSCRIPTIONS FROM SUBSCRIPTION ALLOCATIONS IN THE SATELLITE WEB UI

以下の手順を使用して、Satellite Web UI でサブスクリプション割り当てからサブスクリプションを削除します。



注記

Manifests must not be deleted. If you delete the manifest from the Red Hat Customer Portal or in the Satellite web UI, all of the entitlements for all of your content hosts will be removed.

Prerequisite

サブスクリプションマニフェストファイルは、Satellite Server にインポートしておく必要があります。詳細は、「[Importing a Subscription Manifest into Satellite Server](#)」を参照してください。

Procedure

To remove subscriptions, complete the following steps:

1. In the Satellite web UI, ensure the context is set to the organization you want to use.
2. Navigate to **Content > Subscriptions**.
3. On the row of each subscription you want to remove, select the corresponding check box.
4. **削除**をクリックして、削除を確定します。

4.5. UPDATING AND REFRESHING SUBSCRIPTION MANIFESTS

Every time that you change a subscription allocation, you must refresh the manifest to reflect these changes. For example, you must refresh the manifest if you take any of the following actions:

- Renewing a subscription

- Adjusting subscription quantities
- Purchasing additional subscriptions

You can refresh the manifest directly in the Satellite web UI. Alternatively, you can import an updated manifest that contains the changes.

Procedure

1. In the Satellite web UI, ensure the context is set to the organization you want to use.
2. Navigate to **Content > Subscriptions**.
3. In the Red Hat Subscriptions window, click **Manage Manifest**.
4. マニフェストの管理ウィンドウで、**更新**をクリックします。

4.6. ATTACHING SUBSCRIPTIONS TO CONTENT HOSTS

Using activation keys is the main method to attach subscriptions to content hosts during the provisioning process. However, an activation key cannot update an existing host. If you need to attach new or additional subscriptions, such as future-dated subscriptions, to one host, use the following procedure.

複数ホストの更新に関する詳細情報は、「[Bulk Updating Content Hosts' Subscriptions](#)」を参照してください。

アクティベーションキーに関する詳細情報は、[10章 Managing Activation Keys](#)を参照してください。

Smart Management Subscriptions

Satellite 6 では、管理するすべての Red Hat Enterprise Linux ホストの Red Hat Enterprise Linux Smart Management サブスクリプションを保持する必要があります。

However, you are not required to attach Smart Management subscriptions to each content host. Smart Management subscriptions cannot attach automatically to content hosts in Satellite because they are not associated with any product certificates. Adding a Smart Management subscription to a content host does not provide any content or repository access. If you want, you can add a Smart Management subscription to a manifest for your own recording or tracking purposes.

Prerequisite

Satellite Server にサブスクリプションマニフェストファイルをインポートしておくこと。

Procedure

To attach subscriptions to content hosts, complete the following steps:

1. In the Satellite web UI, ensure the context is set to the organization you want to use.
2. **ホスト > コンテンツホスト**に移動します。
3. On the row of each content host whose subscription you want to change, select the corresponding check box.
4. From the **Select Action** list, select **Manage Subscriptions**.
5. 任意で、**検索**フィールドにキーと値を入力し、表示するサブスクリプションを絞り込みます。

- 追加または削除するサブスクリプションの左側にあるチェックボックスを選択し、必要に応じて **選択した項目を追加** または **選択した項目を削除** をクリックします。
- 完了をクリックして変更を保存します。

For CLI Users

- root ユーザーとして Satellite Server に接続してから、利用可能なサブスクリプションを表示します。

```
# hammer subscription list \
--organization-id 1
```

- Attach a subscription to the host:

```
# hammer host subscription attach \
--host host_name \
--subscription-id subscription_id
```

4.7. BULK UPDATING CONTENT HOSTS' SUBSCRIPTIONS

以下の手順を使用して、インストール後に複数のコンテンツホストを同時に変更します。Satellite Web UI およびフィルター機能を使用して変更するコンテンツホストを選択するか、Hammer CLI ツールの CSV ファイルエクスポート機能を使用して、CSV ファイルで設定を編集し、変更内容をアップロードします。

Procedure

To update multiple content hosts, complete the following steps:

- In the Satellite web UI, ensure the context is set to the organization you want to use.
- ホスト > コンテンツホスト** に移動します。
- On the row of each content host whose subscription you want to change, select the corresponding check box.
- From the **Select Action** list, select **Manage Subscriptions**.
- 任意で、**検索** フィールドにキーと値を入力し、表示するサブスクリプションを絞り込みます。
- 追加または削除するサブスクリプションの左側にあるチェックボックスを選択し、**選択した項目を追加** または **選択した項目を削除** をクリックします。
- 完了をクリックして変更を保存します。

For CLI Users

- Export the current state of content hosts to a CSV file.

```
# hammer --server https://satellite.example.com csv content-hosts --export --file
content_hosts.csv
```

- Make a backup of the file:

```
# cp content_hosts.csv content_hosts.csv.backup
```

3. CSV ファイルで必要な値を変更します。CSV ファイルの文字列を変更するには、以下のよう
に、エディター、CSV プラグイン、または **sed** を使用できます。

```
# sed -i "s/1|RH1234|Red Hat Enterprise Linux Server/1|RH5678|Red Hat Enterprise Linux  
Server/g" content_hosts.csv
```

4. Confirm only the required changes were made. For example:

```
# diff content_hosts.csv content_hosts.csv.backup
```

5. Upload the changed file to Satellite Server:

```
# hammer --server https://satellite.example.com csv content-hosts --file content_hosts.csv
```

第5章 IMPORTING RED HAT CONTENT

このセクションでは、Satellite の製品とリポジトリを使用する方法、Satellite コンテンツが Red Hat コンテンツ配信ネットワーク (CDN) のコンテンツに合わせて最新の状態に保たれるように、同期プランを作成する方法を説明します。

5.1. PRODUCTS IN RED HAT SATELLITE

Satellite では、**製品** は複数のリポジトリをグループ化する組織単位です。たとえば、Satellite では、Red Hat Enterprise Linux Server は **製品** で、この製品のリポジトリは、さまざまなバージョン、アーキテクチャー、アドオンで構成されます。製品を使用すると、相互に依存関係にあるリポジトリが、まとめて同期されます。Red Hat リポジトリの場合は、製品は、リポジトリを有効にした後に自動的に作成されます。

5.2. CONTENT SYNCHRONIZATION OVERVIEW

Satellite Server synchronizes its own repositories with the repositories on the Red Hat CDN. This ensures that Satellite Server retains an exact copy of Red Hat's repositories. Satellite Server fetches this repository information and stores it on Satellite Server's file system. After an initial synchronization, you can create a synchronization plan that checks to ensure the repositories are up to date with the CDN's repositories.

ISO イメージを使用して最初の同期を実行できます。コンテンツ ISO の使用の詳細は、[付録C Importing Content ISOs into a Connected Satellite](#) を参照してください。

5.3. DISABLING THE GLOBAL HTTP PROXY

設定されたプロキシを使用せずにローカルリポジトリを同期する場合には、**satellite-installer** ツールの実行時に設定されるグローバルプロキシを無視できます。

Procedure

To ignore the global HTTP proxy, complete the following steps:

1. In the Satellite web UI, navigate to **Content > Products** and select the product that you want to change.
2. **リポジトリ** タブをクリックして、変更するリポジトリを選択します。
3. **同期設定** エリアで、**グローバル HTTP Capsule を無視** に移動します。編集アイコンをクリックし、チェックボックスを選択して、**保存** をクリックします。

CLI をご利用の場合

```
# hammer repository update --id Repository_ID --ignore-global-proxy yes
```

5.4. DOWNLOAD POLICIES OVERVIEW

Red Hat Satellite には、コンテンツメタデータのみをダウンロードし、実際のコンテンツのダウンロードは後で行うなど、RPM コンテンツの同期に関する複数のダウンロードポリシーが含まれています。

Satellite Server has the following policies:

Immediate

Satellite Server downloads all metadata and packages during synchronization.

On Demand

Satellite Server は同期時にメタデータのみをダウンロードし、Capsule、または直接接続されているクライアントが要求した場合に限り、ファイルシステムにパッケージを取得して保存します。Capsule の対応のリポジトリを **即時** に設定した場合には、Satellite Server が強制的に全パッケージをダウンロードするため、この設定は効果がありません。

Background

Satellite Server creates a background task to download all packages after the initial synchronization.

オンデマンド と 背景 のポリシーは、コンテンツの同期時間を短縮するので、**遅延同期** 機能として動作します。遅延同期機能は **yum** リポジトリにのみ使用してください。通常と同じように、コンテンツビューにパッケージを追加し、通常通りにライフサイクル環境にプロモートすることができます。

Capsule Server offers the following policies:

Immediate

Capsule Server downloads all metadata and packages during synchronization. Do not use this setting if the corresponding repository on Satellite Server is set to **On Demand** as Satellite Server is forced to download all the packages.

On Demand

Capsule Server は、同期時にメタデータのみをダウンロードします。Capsule Server は、直接接続されたクライアントから要求されると、ファイルシステムだけにパッケージを取得して保存します。オンデマンド ダウンロードポリシーを使用すると、Capsule Server でコンテンツを入手できない場合には、コンテンツが Satellite Server からダウンロードされます。

Background

Capsule Server creates a background task to download all packages after the initial synchronization.

Inherit

Capsule Server inherits the download policy for the repository from the corresponding repository on Satellite Server.

These policies are not available if a Capsule was installed or updated with **--enable-foreman-proxy-plugin-pulp** set to false.

5.5. CHANGING THE DEFAULT DOWNLOAD POLICY

Satellite が全組織に作成した新規リポジトリに対して適用するデフォルトのダウンロードポリシーを設定できます。デフォルト値を変更しても、既存の設定は変更されません。

Procedure

To change the default download policy for repositories, complete the following steps:

1. In the Satellite web UI, navigate to **Administer > Settings**.
2. **コンテンツ** タブをクリックし、**デフォルトのリポジトリダウンロードポリシー** を見つけます。
3. **値** フィールドで、**編集** アイコンをクリックします。
4. 必要なダウンロードポリシーを選択し、**保存** をクリックします。

For CLI Users

デフォルトのダウンロードポリシーを **immediate**、**on_demand**、**background** の1つに変更するには、以下のコマンドを入力します。

```
# hammer settings set \
--name default_download_policy \
--value immediate
```

5.6. CHANGING THE DOWNLOAD POLICY FOR A REPOSITORY

You can also set the download policy for a repository.

Procedure

To set the download policy for a repository, complete the following steps:

1. In the web UI, navigate to **Content > Products**, and click the required product name.
2. **リポジトリ** タブで必要なリポジトリ名をクリックし、**ポリシーのダウンロード** フィールドを見つけて、**編集** アイコンをクリックします。
3. リストから、必要なダウンロードポリシーを選択して、**保存** をクリックします。

For CLI Users

1. List the repositories for an organization:

```
# hammer repository list \
--organization-label organization-label
```

2. リポジトリのダウンロードポリシーを **immediate**、**on_demand**、**background** の1つに変更します。

```
# hammer repository update \
--organization-label organization-label \
--product "Red Hat Enterprise Linux Server" \
--name "Red Hat Enterprise Linux 7 Server Kickstart x86_64 7.5" \
--download-policy immediate
```

5.7. SELECTING RED HAT REPOSITORIES TO SYNCHRONIZE

To select the repositories to synchronize, you must first identify the product that contains the repository, and then enable that repository based on the relevant release version and base architecture. For Red Hat Enterprise Linux 8, you must enable both AppStream and BaseOS repositories.

Disconnected Satellite

非接続の Satellite Server を使用している場合は、コンテンツを同期する前に Red Hat Satellite 用のコンテンツ ISO をインポートし、Satellite Server 上で CDN URL を変更する必要があります。詳細は [付録B Importing Content ISOs into a Disconnected Satellite](#) を参照してください。

For Red Hat Enterprise Linux 8

To add the AppStream and BaseOS repositories that you require for Red Hat Enterprise Linux 8, complete the following steps:

1. In the Satellite web UI, navigate to **Content > Red Hat Repositories**.
2. **製品別のフィルター** 一覧から **Red Hat Enterprise Linux x86_64** を選択します。
3. 結果の一覧をフィルタリングするには、**検索** フィールドを使用して名前別にフィルタリングするか、**推奨のリポジトリ** ボタンをオンの位置に切り替えます。
4. 利用可能なリポジトリペインで、**Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMS)** と **Red Hat Enterprise Linux 8 for x86_64 - BaseOS (RPMs)** を展開します。
5. 各 **x86_64 8** エントリーの横にある **有効化** アイコンをクリックして、リポジトリを有効にします。
6. **コンテンツ > 同期の状態** に移動して、Red Hat Enterprise Linux 8 リポジトリを選択し、**今すぐ同期** をクリックします。

Viewing Red Hat Enterprise Linux 8 Modules

When you synchronize Red Hat Enterprise Linux 8 repositories to Satellite, you can view the module streams that are associated with the repositories.

To view the module streams for a repository, complete the following steps:

1. In the Satellite web UI, navigate to **Content > Products**, and select the product that contains the repository you want to view.
2. リポジトリの **コンテンツ** 列で、**58 モジュールストリーム** など、**モジュールストリーム** の数をクリックします。
3. To view information about a specific module stream, click the module.

For Red Hat Enterprise Linux 7

To add repositories, complete the following steps:

1. In the Satellite web UI, navigate to **Content > Red Hat Repositories**.
2. **Red Hat Enterprise Linux 7 Server (RPMs)** リポジトリを検索するには、リポジトリ名を入力するか、**推奨のリポジトリ** ボタンをオンの位置に切り替えて、必要なりポジトリの一覧を表示します。
3. In the Available Repositories pane, click the **Red Hat Enterprise Linux 7 Server (RPMs)** to expand the repository set.
4. **x86_64 7Server** エントリーの場合は、**有効化** アイコンをクリックしてリポジトリを有効にします。これにより、Red Hat Enterprise Linux 7 の最新の RPM ファイルが有効になります。

Repository Versioning

Red Hat Enterprise Linux オペレーティングシステムに 7 Server リポジトリと、7.X リポジトリを関連付けることの相違点は、7 Server リポジトリには最新アップデートがすべて含まれますが、Red Hat Enterprise Linux 7.X リポジトリは次のマイナーバージョンリリース以降のアップデートを取得しなくなる点です。キックスタートリポジトリにはマイナーバージョンのみが含まれることに注意してください。

For CLI Users

1. To search for your product, enter the following command:

```
# hammer product list --organization "My_Organization"
```

- List the repository set for the product:

```
# hammer repository-set list \
--product "Red Hat Enterprise Linux Server" \
--organization "My_Organization"
```

- 名前または ID 番号のいずれかを使用してリポジトリを有効にします。リリースバージョンに **7Server**、基本アーキテクチャーに **x86_64** のように含めます。例を示します。

```
# hammer repository-set enable \
--name "Red Hat Enterprise Linux 7 Server (RPMs)" \
--releasever "7Server" \
--basearch "x86_64" \
--product "Red Hat Enterprise Linux Server" \
--organization "My_Organization"
```

5.8. SYNCHRONIZING RED HAT REPOSITORIES

Synchronize the repositories with the Red Hat CDN's repositories.

For Web UI Users

- In the Satellite web UI, navigate to **Content > Products** and select **Red Hat Enterprise Linux Server**.
- すべてのリポジトリを選択して **今すぐ同期** をクリックします。

Web UI で同期の進捗状況を表示するには、**コンテンツ > 同期の状態** に移動して、対応する製品またはリポジトリツリーを展開します。

For CLI Users

Synchronize the enabled repositories in the Red Hat Enterprise Linux Server product:

```
# hammer product synchronize \
--name "Red Hat Enterprise Linux Server" \
--organization "My_Organization"
```

You can also synchronize each repository individually. List all repositories in the product, then synchronize using the ID number for the corresponding repositories. For example:

```
# hammer repository list \
--product "Red Hat Enterprise Linux Server" \
--organization "My_Organization"
# hammer repository synchronize \
--name "Red Hat Enterprise Linux 7 Server RPMs x86_64 7Server" \
--product "Red Hat Enterprise Linux Server" \
--organization "My_Organization"
```

The synchronization duration depends on the size of each repository and the speed of your network connection. The following table provides estimates of how long it would take to synchronize content, depending on the available Internet bandwidth:

	Single Package (10Mb)	Minor Release (750Mb)	Major Release (6Gb)
256 Kbps	5 Mins 27 Secs	6 Hrs 49 Mins 36 Secs	2 Days 7 Hrs 55 Mins
512 Kbps	2 Mins 43.84 Secs	3 Hrs 24 Mins 48 Secs	1 Day 3 Hrs 57 Mins
T1 (1.5 Mbps)	54.33 Secs	1 Hr 7 Mins 54.78 Secs	9 Hrs 16 Mins 20.57 Secs
10 Mbps	8.39 Secs	10 Mins 29.15 Secs	1 Hr 25 Mins 53.96 Secs
100 Mbps	0.84 Secs	1 Min 2.91 Secs	8 Mins 35.4 Secs
1000 Mbps	0.08 Secs	6.29 Secs	51.54 Secs

定期的に更新されるように同期プランを作成します。

Changing a Download Policy

Red Hat リポジトリのダウンロードポリシーは変更できます。**Red Hat Enterprise Linux Server** 製品内のリポジトリを選択し、**ダウンロードポリシー** フィールドに移動します。CLI を使用している場合は、**hammer repository update** コマンドに **--download-policy** オプションを追加して実行します。

5.9. RECOVERING A REPOSITORY

In the case of repository corruption, you can recover it by using an advanced synchronization, which has three options:

Optimized Sync

Synchronizes the repository bypassing RPMs that have no detected differences from the upstream RPMs.

Complete Sync

Synchronizes all RPMs regardless of detected changes. Use this option if specific RPMs could not be downloaded to the local repository even though they exist in the upstream repository.

コンテンツの同期検証

すべての RPM を同期してから、すべての RPM のチェックサムをローカルで検証します。RPM のチェックサムがアップストリームと異なる場合は、RPM をもう一度ダウンロードします。このオプションは **yum** リポジトリにのみ該当します。以下のいずれかのエラーが発生した場合に限りこのオプションを使用します。

- Specific RPMs cause a **404** error while synchronizing with **yum**.
- **Package does not match intended download** error, which means that specific RPMs are corrupted.

Procedure

To synchronize a specific repository with an advanced option, complete the following steps:

1. Satellite Web UI で、**コンテンツ** > **製品**に移動します。

2. Select the product containing the corrupted repository.
3. Select the name of a repository you want to synchronize.
4. From the **Select Action** menu, select **Advanced Sync**.
5. オプションを選択し、**同期**をクリックします。

CLI をご利用の場合

1. Obtain a list of repository IDs:

```
# hammer repository list --organization "My_Organization"
```

2. Synchronize a corrupted repository using the necessary option:

- For the optimized synchronization:

```
# hammer repository synchronize --incremental true --id 1
```

- For the complete synchronization:

```
# hammer repository synchronize --skip-metadata-check true --id 1
```

- For the validate content synchronization:

```
# hammer repository synchronize --validate-contents true --id 1
```

5.10. LIMITING SYNCHRONIZATION SPEED

同期の速度を制限して、利用可能な帯域幅の枯渇や、その他のパフォーマンスの問題を回避できます。これには、**PULP_CONCURRENCY** パラメーターおよび **max_speed** パラメーターを設定します。この設定はアップグレード時に上書きされる点に注意してください。アップグレード前に、変更したファイルをバックアップし、設定を復元できるようにしてください。

1. To control the number of synchronization jobs that run in parallel, configure the **PULP_CONCURRENCY** parameter in the **/etc/default/pulp_workers** file. For example, to set the number of jobs that run in parallel to 1, change **PULP_CONCURRENCY** to 1:

```
PULP_CONCURRENCY=1
```

By default, on a system with less than 8 CPUs, **PULP_CONCURRENCY** is set to the number of CPUs. On a system with more than 8 CPUs, it is set to 8.

2. 同期するネットワークの最大速度 (バイト毎秒) を設定するには、**max_speed** パラメーターを設定します。このパラメーターは、**/etc/pulp/server/plugins.conf.d/** ディレクトリーで、インポーターごとに設定する必要があります。たとえば、RPM コンテンツを同期する最大速度を毎秒 10 バイトに設定するには、**/etc/pulp/server/plugins.conf.d/yum_importer.json** ファイルの **"max_speed"** パラメーターを 10 に設定します。

```
# cat /etc/pulp/server/plugins.conf.d/yum_importer.json
{
  "proxy_host": null,
  "proxy_port": null,
```

```
"proxy_username": null,
"proxy_password": null,
"max_speed": 10
}
```

3. Verify the syntax of the file after editing:

```
# json_verify < /etc/pulp/server/plugins.conf.d/yum_importer.json
JSON is valid
```

4. Restart Satellite services to apply the changes:

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

5.11. CREATING A SYNCHRONIZATION PLAN

A synchronization plan checks and updates the content at a scheduled date and time. In Red Hat Satellite 6, you can create a synchronization plan and assign products to the plan.

Procedure

To create a synchronization plan, complete the following steps:

1. Satellite Web UI で **コンテンツ** > **同期プラン** に移動して、**新規同期プラン** をクリックします。
2. **名前** フィールドに、プランの名前を入力します。
3. **説明** フィールドに、プランの説明を入力します。
4. **間隔** の一覧から、プランを実行する間隔を選択します。
5. **開始日** と **開始時間** のリストから、同期プランの実行を開始する日時を開始します。
6. **保存** をクリックします。
7. **製品** タブをクリックし、次に **追加** をクリックします。Red Hat Enterprise Linux Server製品を選択し、**選択した項目を追加** をクリックします。

For CLI Users

1. To create the synchronization plan, enter the following command:

```
# hammer sync-plan create \
--name "Red Hat Products 2" \
--description "Example Plan for Red Hat Products" \
--interval daily \
--sync-date "2016-02-01 01:00:00" \
--enabled true \
--organization "My_Organization"
```

2. Assign the Red Hat Enterprise Linux Server product to it:

```
# hammer product set-sync-plan \
--name "Red Hat Enterprise Linux Server" \
--sync-plan "Red Hat Products" \
```

█ --organization "**My_Organization**"

第6章 IMPORTING CUSTOM CONTENT

本章では、さまざまな種類のカスタムコンテンツを Satellite にインポートする方法を概説します。Satellite に ISO、Puppet モジュール、またはさまざまなコンテンツタイプをインポートする場合は、この章とほぼ同じ手順を実行します。

For example, you can use the following chapters for information on specific types of custom content but the underlying procedures are the same:

- [12章 Managing OSTree Content](#)
- [13章 Managing ISO Images](#)
- [14章 Managing Custom File Type Content](#)
- [15章 Managing Custom Puppet Content](#)

6.1. USING CUSTOM PRODUCTS IN SATELLITE

Red Hat Satellite 6 の Red Hat コンテンツとカスタムコンテンツはいずれも、次の点が類似しています。

- The relationship between a product and its repositories is the same and the repositories still require synchronization.
- Custom products require a subscription for clients to access, similar to subscriptions to Red Hat Products. Red Hat Satellite 6 creates a subscription for each custom product you create.

RPM の作成およびパッケージングの詳細は、Red Hat Enterprise Linux ドキュメンテーションの [RPM パッケージングガイド](#) を参照してください。

6.2. IMPORTING CUSTOM SSL CERTIFICATES

Before you create custom content, ensure that you import any Custom SSL certificates that you require.

If you require SSL certificates and keys to download RPMs, you can add them to Satellite.

1. In the Satellite web UI, navigate to **Content** > **Content Credentials**. In the Content Credentials window, click **Create Content Credential**.
2. **名前** フィールドに、SSL 証明書の名前を入力します。
3. **タイプ** の一覧から、**SSL 証明書** を選択します。
4. **コンテンツの認証情報** フィールドに SSL 証明書を貼り付けるか、**参照** をクリックして SSL 証明書をアップロードします。
5. **保存** をクリックします。

6.3. IMPORTING A CUSTOM GPG KEY

Before you create custom content, ensure that you import any GPG keys that you require.

Prerequisites

1. Download a copy of the version specific repository package to your client system.

```
$ wget http://yum.postgresql.org/9.5/redhat/rhel-7-x86_64/pgdg-redhat95-9.5-2.noarch.rpm
```

2. Extract the RPM file without installing it:

```
$ rpm2cpio pgdg-redhat95-9.5-2.noarch.rpm | cpio -idmv
```

The GPG key is located relative to the extraction at **etc/pki/rpm-gpg/RPM-GPG-KEY-PGDG-95**.

Procedure

To import a GPG key, complete the following procedure:

1. In the Satellite web UI, navigate to **Content > Content Credentials** and in the upper-right of the window, click **Create Content Credential**.
2. リポジトリの名前を入力し、**タイプ**の一覧から **GPG キー** を選択します。
3. GPG キーを **Content Credential Contents** フィールドに貼り付けるか、**参照**をクリックし、インポートする GPG キーファイルを選択します。
4. **保存**をクリックします。

For CLI Users

1. Copy the GPG key to your Satellite Server:

```
$ scp ~/etc/pki/rpm-gpg/RPM-GPG-KEY-PGDG-95 root@satellite.example.com:~/.
```

2. Upload the GPG key to Satellite:

```
# hammer gpg create \  
--key ~/RPM-GPG-KEY-PGDG-95 \  
--name "My_Repository" \  
--organization "My_Organization"
```

6.4. CREATING A CUSTOM PRODUCT

Use this procedure to create a custom product that you can then add repositories to.

Procedure

To create a custom product, complete the following procedure:

1. In the Satellite web UI, navigate to **Content > Products**, click **New Product**.
2. **名前** フィールドで、製品の名前を入力します。Red Hat Satellite 6 では、**名前** に入力した内容に基づいて **ラベル** フィールドに自動的に入力されます。
3. オプション: **GPG キー** の一覧から、製品の GPG キーを選択します。
4. オプション: **SSL CA 証明書** の一覧から、製品の SSL CA 証明書を選択します。

5. オプション: **SSL クライアント証明書** の一覧から、製品の SSL クライアント 証明書を選択します。
6. オプション: **SSL クライアントキー** の一覧から、製品の SSL クライアントキーを選択します。
7. オプション: **同期プラン** の一覧から、既存の同期プランを選択するか、**同期プランの作成** をクリックし、製品要件の同期プランを作成します。
8. **説明** フィールドには、製品の説明を入力します。
9. **保存** をクリックします。

For CLI Users

To create the product, enter the following command:

```
# hammer product create \
--name "My_Product" \
--sync-plan "Example Plan" \
--description "Content from My Repositories" \
--organization "My_Organization"
```

6.5. ADDING A CUSTOM RPM REPOSITORY

Use this procedure to add a custom RPM repository in Satellite.

The Products window in the Satellite web UI also provides a **Repo Discovery** function that finds all repositories from a URL and you can select which ones to add to your custom product. For example, you can use the **Repo Discovery** to search, for example, <http://yum.postgresql.org/9.5/redhat/> and list all repositories for different Red Hat Enterprise Linux versions and architectures. This helps users save time importing multiple repositories from a single source.

カスタム RPM のサポート

Red Hat では、サードパーティーのサイトに、直接接続されているアップストリーム RPM はサポートしていません。これらの RPM は同期プロセスのデモに使用されます。これらの RPM について問題がある場合は、サードパーティーの開発者に連絡してください。

Procedure

1. Satellite Web UI で、**コンテンツ** > **製品** に移動し、使用する製品を選択して、**リポジトリの作成** をクリックします。
2. **名前** フィールドで、リポジトリの名前を入力します。Red Hat Satellite 6 では、**名前** に入力した内容に基づいて **ラベル** フィールドに値が自動的に入力されます。
3. From the **Type** list, select the type of repository. You can select either a repository for RPM files (**yum**), Puppet modules (**puppet**), or Docker images (**docker**).
4. In the **URL** field, enter the URL of the external repository to use as a source.
5. From the **Download Policy** list, select the type of synchronization Satellite Server performs.
6. **同期時のミラー** チェックボックスが選択されていることを確認します。アップストリームのリポジトリにないコンテンツが同期中に削除されるようにします。

7. **チェックサム** の一覧から、リポジトリのチェックサムタイプを選択します。
8. オプション: 必要に応じて、**HTTP 経由で公開** チェックボックスの選択を解除して、このリポジトリの HTTP 経由での公開を無効にできます。
9. オプション: **GPG キー** の一覧から、製品の GPG キーを選択します。
10. **保存** をクリックします。

製品のウィンドウで即時同期を実行する場合は、**今すぐ同期** をクリックします。

For CLI Users

1. Enter the following command to create the repository:

```
# hammer repository create \  
--name "My_Repository" \  
--content-type "yum" \  
--publish-via-http true \  
--url http://yum.postgresql.org/9.5/redhat/rhel-7-x86_64/ \  
--gpg-key "My_Repository" \  
--product "My_Product" \  
--organization "My_Organization"
```

2. Synchronize the repository:

```
# hammer repository synchronize \  
--name "My_Repository" \  
--product "My_Product" \  
--organization "My_Organization"
```

第7章 CREATING AN APPLICATION LIFE CYCLE

This chapter outlines the application life cycle in Satellite and how to create and remove application life cycles for Satellite and Capsule.

7.1. APPLICATION LIFE CYCLE OVERVIEW

アプリケーションライフサイクル は、Red Hat Satellite 6 のコンテンツ管理機能の中心となる概念です。アプリケーションライフサイクルは、特定の段階で特定のシステムとソフトウェアがどのように見えるかを定義します。たとえば、アプリケーションライフサイクルが単純な場合には、開発段階と実稼働段階のみになります。このような場合に、アプリケーションライフサイクルは以下のようになります。

- Development
- Production

However, a more complex application life cycle might have further stages, such as a phase for testing or a beta release. This adds extra stages to the application life cycle:

- Development
- Testing
- Beta Release
- Production

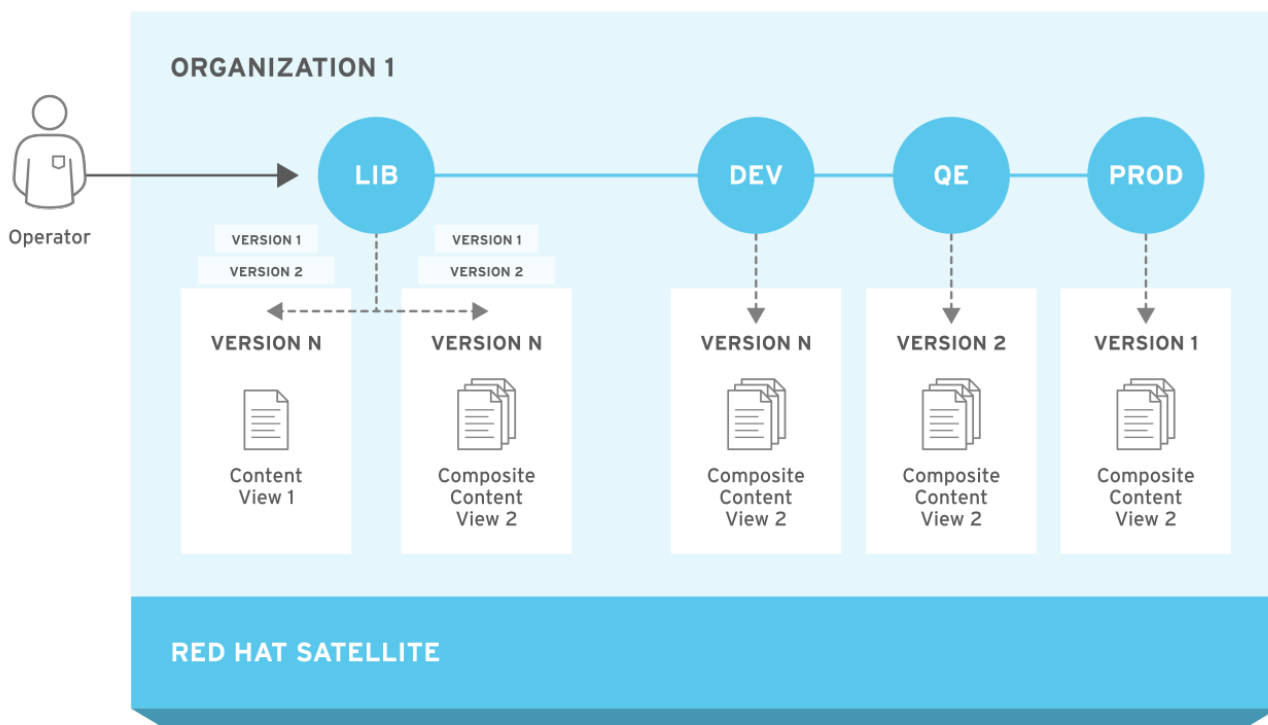
Red Hat Satellite 6 provides methods to customize each application life cycle stage so that it suits your specifications.

Each stage in the application life cycle is called an **environment** in Red Hat Satellite 6. Each environment uses a specific collection of content. Red Hat Satellite 6 defines these content collections as a Content View. Each Content View acts as a filter where you can define what repositories, packages, and Puppet modules to include in a particular environment. This provides a method for you to define specific sets of content to designate to each environment.

たとえば、メールサーバーの場合は、実際に使用する実稼働レベルのサーバーと、最新のメールサーバーパッケージをテストするテストサーバーという、単純なアプリケーションライフサイクルのみを必要とします。テストサーバーが初期段階をパスすると、実稼働レベルのサーバーで新パッケージを使用するように設定できます。

別の例としては、ソフトウェア製品の開発ライフサイクルがあります。開発環境でソフトウェアの新しい部分を開発するには、品質保証環境でソフトウェアをテストしてベータ版としてプレリリースした後、実稼働レベルのアプリケーションとしてソフトウェアをリリースします。

図7.1 The Red Hat Satellite 6 Application Life Cycle



7.2. PROMOTING CONTENT ACROSS THE APPLICATION LIFE CYCLE

In the application life cycle chain, when content moves from one environment to the next, this is called **promotion**.

Example Content Promotion Across Satellite Life Cycle Environments

各環境には、Red Hat Satellite 6 に登録したシステムが含まれ、そのシステムがアクセスできるのは、各環境に関連するリポジトリに限られます。別の環境にパッケージをプロモートすると、プロモート先の環境のリポジトリはパッケージの新バージョンを受け取り、その結果、プロモート先の環境にある各システムはパッケージを新バージョンに更新できます。

Development	Testing	Production
example_software-1.1-0.noarch.rpm	example_software-1.0-0.noarch.rpm	example_software-1.0-0.noarch.rpm

パッチ開発の完了後には、RPM をテスト環境にプロモートして、品質保証エンジニアチームがパッチをレビューできるようにします。この時点では、アプリケーションライフサイクルに以下のパッケージバージョンが各環境に含まれます。

Development	Testing	Production
example_software-1.1-0.noarch.rpm	example_software-1.1-0.noarch.rpm	example_software-1.0-0.noarch.rpm

品質保証エンジニアチームがパッチのレビューを行う間、開発チームは **example_software 2.0** の作業に着手します。このため、アプリケーションライフサイクルは以下のようになります。

Development	Testing	Production
exampleware-2.0-0.noarch.rpm	exampleware-1.1-0.noarch.rpm	exampleware-1.0-0.noarch.rpm

品質保証エンジニアチームがパッチのレビューを完了します。これで、`example_software 1.1` をリリースする準備が完了します。1.1 を **実稼働環境** にプロモートします。

Development	Testing	Production
example_software-2.0-0.noarch.rpm	example_software-1.1-0.noarch.rpm	example_software-1.1-0.noarch.rpm

開発チームが `example_software 2.0` の作業を完了し、テスト環境にプロモートします。

Development	Testing	Production
example_software-2.0-0.noarch.rpm	example_software-2.0-0.noarch.rpm	example_software-1.1-0.noarch.rpm

Finally, the Quality Engineering team reviews the package. After a successful review, promote the package to the **Production** environment:

Development	Testing	Production
exampleware-2.0-0.noarch.rpm	exampleware-2.0-0.noarch.rpm	exampleware-2.0-0.noarch.rpm

For more information, see [「Promoting a Content View」](#) .

7.3. CREATING AN APPLICATION LIFE CYCLE

ソフトウェアを開発してリリースする単純なアプリケーションライフサイクルを作成するには、初期環境として **ライブラリー** 環境を使用し、その後に **開発**、**テスト**、および **実稼働** の順に、チェーンにこの3つの環境を追加します。

Procedure

To create an application life cycle, complete the following steps:

1. In the Satellite web UI, navigate to **Content > Lifecycle Environments**.
2. **新規環境パス** をクリックして、新しいアプリケーションライフサイクルを開始します。
3. **名前** フィールドに、環境の名前を入力します。
4. In the **Description** field, enter a description for your environment.
5. **保存** をクリックします。

For CLI Users

Use the **hammer lifecycle-environment create** command to create each environment. Use the **--prior** option to specify the previous environment. For example:

```
# hammer lifecycle-environment create \
--name "Development" \
--description "Development Environment" \
--prior "Library" \
--organization "My_Organization"
# hammer lifecycle-environment create \
--name "Testing" \
--description "Testing Environment" \
--prior "Development" \
--organization "My_Organization"
# hammer lifecycle-environment create \
--name "Production" \
--description "Production Environment" \
--prior "Testing" \
--organization "My_Organization"
```

To view the chain of the life cycle environment, enter the **hammer lifecycle-environment paths** command:

```
# hammer lifecycle-environment paths --organization "_My_Organization_"
-----
LIFECYCLE PATH
-----
Library >> Development >> Testing >> Production
-----
```

7.4. REMOVING LIFE CYCLE ENVIRONMENTS FROM SATELLITE SERVER

Use this procedure to remove a life cycle environment.

Procedure

ライフサイクル環境を削除するには、以下の手順を行います。

1. Satellite Web UI で、**コンテンツ > ライフサイクル環境** に移動します。
2. 削除するライフサイクル環境の名前をクリックし、**環境の削除**をクリックします。
3. **削除** をクリックして環境を削除します。

For CLI Users

1. List the life cycle environments for your organization and note the name of the life cycle environment you want to remove:

```
# hammer lifecycle-environment list --organization "My_Organization"
```

2. Use the **hammer lifecycle-environment delete** command to remove an environment:

```
# hammer lifecycle-environment delete \
--name "your_environment" \
--organization "My_Organization"
```

7.5. REMOVING LIFE CYCLE ENVIRONMENTS FROM CAPSULE SERVER

When life cycle environments are no longer relevant to the host system or environments are added incorrectly to Capsule Server, you can remove the life cycle environments from Capsule Server.

Capsule からライフサイクル環境を削除するには、Satellite Web UI と Hammer の両方を使用できません。

Procedure

To remove a life cycle environment from Capsule Server, complete the following step:

1. In the Satellite web UI, navigate to **Infrastructure** > **Capsules**, and select the Capsule that you want to remove a life cycle from.
2. **Edit** をクリックしてから、**Life Cycle Environments** タブをクリックします。
3. 右のメニューから、Capsule から削除するライフサイクル環境を選択し、**送信** をクリックします。
4. Capsule のコンテンツを同期するには、**概要**タブをクリックしてから**同期**ボタンをクリックします。
5. **最適化された同期**または **完全な同期**を選択します。

For CLI Users

To remove a life cycle environment from Capsule Server, complete the following steps:

1. Select the Capsule Server that you want from the list and take note of its **id**:

```
# hammer capsule list
```

2. To verify the Capsule Server's details, enter the following command:

```
# hammer capsule info --id capsule_id
```

3. Verify the list of life cycle environments currently attached to the Capsule Server and take note of the **environment id**:

```
# hammer capsule content lifecycle-environments \
--id capsule_id
```

4. Remove the life cycle environment from Capsule Server:

```
# hammer capsule content remove-lifecycle-environment \
--id capsule_id \
--environment-id environment_id
```

Repeat this step for every life cycle environment that you want to remove from Capsule Server.

5. Synchronize the content from Satellite Server's environment to Capsule Server:

```
# hammer capsule content synchronize \  
--id capsule_id
```

7.6. ADDING LIFE CYCLE ENVIRONMENTS TO CAPSULE SERVERS

If your Capsule Server has the content functionality enabled, you must add an environment so that Capsule can synchronize content from Satellite Server and provide content to host systems.

リポジトリが CDN から更新されるたびに自動で Capsule が同期されるようになるので、**ライブライフサイクル環境**を Capsule Server に割り当てないでください。自動的に同期されると、Capsule 上の複数のシステムリソースや Satellite と Capsule 間のネットワーク帯域幅、および Capsule 上の利用可能なディスク領域が消費される可能性があります。

You can use Hammer CLI on Satellite Server or the Satellite web UI.

Procedure

To add a life cycle environment to Capsule Server, complete the following step:

1. In the Satellite web UI, navigate to **Infrastructure > Capsules**, and select the Capsule that you want to add a life cycle to.
2. **Edit** をクリックしてから、**Life Cycle Environments** タブをクリックします。
3. From the left menu, select the life cycle environments that you want to add to Capsule, and then click **Submit**.
4. Capsule のコンテンツを同期するには、**概要**タブをクリックしてから**同期**ボタンをクリックします。
5. **最適化された同期**または **完全な同期**を選択します。

For CLI Users

1. Capsule Server の全一覧を表示するには、以下のコマンドを入力します。

```
# hammer capsule list
```

Note the ID that returns.

2. Using the ID, verify the details of your Capsule Server:

```
# hammer capsule info --id capsule_id
```

3. Verify the life cycle environments available and note the environment ID:

```
# hammer capsule content available-lifecycle-environments \  
--id capsule_id
```

4. To view the life cycle environments available for your Capsule Server, enter the following command and note the ID and the organization name:


```
# hammer capsule content available-lifecycle-environments --id capsule_id
```

5. Add the life cycle environment to your Capsule Server:

```
# hammer capsule content add-lifecycle-environment \  
--id capsule_id --organization "My_Organization" \  
--environment-id environment_id
```

Repeat for each life cycle environment you want to add to Capsule Server.

To synchronize all content from your Satellite Server environment to Capsule Server, enter the following command:

```
# hammer capsule content synchronize --id capsule_id
```

To synchronize a specific life cycle environment from your Satellite Server to Capsule Server, enter the following command:

```
# hammer capsule content synchronize --id external_capsule_id \  
--environment-id environment_id
```

第8章 MANAGING CONTENT VIEWS

Red Hat Satellite 6 uses Content Views to create customized repositories from the repositories. To do this, you must define which repositories to use and then apply certain filters to the content. These filters include both package filters, package group filters, and errata filters. You can use Content Views to define which software versions a particular environment uses. For example, a **Production** environment might use a Content View containing older package versions, while a **Development** environment might use a Content View containing newer package versions.

コンテンツビューは各環境でリポジトリセットを作成し、Satellite Server が保存して管理します。アプリケーションライフサイクルの次の環境にコンテンツビューをプロモートすると、それに対応する Satellite Server のリポジトリがパッケージを更新して公開します。

	Development	Testing	Production
Content View Version and Contents	Version 2 - example_software-1.1-0.noarch.rpm	Version 1 - example_software-1.0-0.noarch.rpm	Version 1 - example_software-1.0-0.noarch.rpm

The repositories for Testing and Production contain the **example_software-1.0-0.noarch.rpm** package. If you promote Version 2 of the Content View from Development to Testing, the repository for Testing regenerates and then contains the **example_software-1.1-0.noarch.rpm** package:

	Development	Testing	Production
Content View Version and Contents	Version 2 - example_software-1.1-0.noarch.rpm	Version 2 - example_software-1.1-0.noarch.rpm	Version 1 - example_software-1.0-0.noarch.rpm

This ensures systems are designated to a specific environment but receive updates when that environment uses a new version of the Content View.

スナップショットのフィルタリングと作成に使用するコンテンツビューを作成する一般的なワークフローは、以下の通りです。

1. Create a Content View.
2. Add the repository and the Puppet modules that you want to the Content View.
3. Optionally, create one or more filters to refine the content of the Content View.
4. Publish the Content View.
5. Optionally, promote the Content View to another environment.
6. Attach the content host to the Content View.

If a repository is not associated with the Content View, the file `/etc/yum.repos.d/redhat.repo` remains empty and systems registered to it cannot receive updates.

ホストを関連付けるコンテンツビューは1つだけにすることができます。複数のコンテンツビューにホストを関連付けるには、複合コンテンツビューを作成します。詳細は「[Creating a Composite Content View](#)」を参照してください。

8.1. CREATING A CONTENT VIEW

Use this procedure to create a simple Content View.

Procedure

To create a content view complete the following steps:

1. In the Satellite web UI, navigate to **Content** > **Content Views** and click **Create New View**.
2. **名前** フィールドで、ビューの名前を入力します。Red Hat Satellite 6 では、入力した名前から、**ラベル** フィールドに自動的に入力されます。
3. **説明** フィールドに、ビューの説明を入力します。
4. Click **Save** to create the Content View.
5. In the **Repository Selection** area, select the repositories that you want to add to your Content View, then click **Add Repositories**.
6. **新規バージョンの公開** をクリックし、**説明** フィールドに、変更をログに記録するバージョンについての情報を入力します。
7. **保存** をクリックします。
8. オプション: Yum リポジトリでメタデータを再生成させるには、コンテンツビューバージョンの **アクション** リストから、**リポジトリメタデータの再生成** を選択します。

You can view the Content View in the Content Views window. To view more information about the Content View, click the Content View name.

コンテンツビューにホストを登録するには、『**ホストの管理**』ガイドの「**ホストの登録**」を参照してください。

For CLI Users

1. Obtain a list of repository IDs:

```
# hammer repository list --organization "My_Organization"
```

2. Create the Content View and add repositories:

```
# hammer content-view create \  
--name "Example_Content_View" \  
--description "Example Content View" \  
--repository-ids 1,2 \  
--organization "My_Organization"
```

--repository-ids オプションを使用すると、**hammer repository list** コマンドの出力で ID が分かります。

3. Publish the view:

```
# hammer content-view publish \  
--name "Example_Content_View" \  
--description "Example Content View" \  
--organization "My_Organization"
```

```
--organization "My_Organization"
```

Satellite Server creates the new version of the view and publishes it to the Library environment.

8.2. VIEWING MODULE STREAMS

In Satellite, you can view the module streams of the repositories in your Content Views.

Procedure

To view module streams for the repositories in your content view, complete the following steps:

1. In the Satellite web UI, navigate to **Content > Content Views**, and select the Content View that contains the modules you want to view.
2. **バージョン** タブをクリックし、表示するコンテンツビューバージョンを選択します。
3. **モジュールストリーム** タブをクリックして、コンテンツビューに利用できるモジュールストリームを表示します。
4. **フィルター** フィールドを使用して、モジュールのリストを絞り込みます。
5. To view the information about the module, click the module.

8.3. CREATING A CONTENT VIEW WITH A PUPPET MODULE

以下の手順を使用して、フィルターなしで、リポジトリを1つ使用するコンテンツビューを作成します。

Procedure

To create a Content View with a Puppet module, complete the following steps:

1. In the Satellite web UI, navigate to **Content > Content Views** and click **Create New View**.
2. **名前** フィールドで、ビューの名前を入力します。Red Hat Satellite 6 では、入力した名前から、**ラベル** フィールドに自動的に入力されます。
3. **説明** フィールドに、ビューの説明を入力します。
4. Click **Save** to complete.
5. In the **Repository Selection** area, select the repositories that you want to add to your Content View, then click **Add Repositories**.
6. **Puppet モジュール** タブで **新規モジュールの追加** をクリックします。
7. 追加するモジュールを探し、**バージョンの選択** をクリックします。
8. Navigate to the entry for **Use Latest** and click **Select Version** in the **Actions** column.
9. 公開するには、**バージョン** タブをクリックし、**新規バージョンの公開** をクリックします。説明フィールドに、変更をログに記録する説明を入力し、**保存** をクリックします。

コンテンツビューにホストを登録するには、『**ホストの管理**』ガイドの「**ホストの登録**」を参照してください。

8.4. PROMOTING A CONTENT VIEW

以下の手順を使用して、異なるライフサイクル環境全体に、コンテンツビューをプロモートします。

Permission Requirements for Content View Promotion

Non-administrator users require two permissions to promote a Content View to an environment:

1. `promote_or_remove_content_views`
2. `promote_or_remove_content_views_to_environment`.

`promote_or_remove_content_views` パーMISSIONで、ユーザーがプロモートできるコンテンツビューを制限します。

`promote_or_remove_content_views_to_environment` パーMISSIONで、コンテンツビューのプロモート先となる環境を制限します。

上記のパーMISSIONを使用すると、指定の環境に指定のコンテンツビューをプロモートし、それ以外の環境にはプロモートできないように、ユーザーパーMISSIONを割り当てることができます。たとえば、テスト環境へのプロモーションはできるが、実稼働環境にはできないようにユーザーを制限できます。

コンテンツビューをプロモートできるようにするには、ユーザーに両パーMISSIONを割り当てる必要があります。

Procedure

To promote a Content View, complete the following steps:

1. In the Satellite web UI, navigate to **Content > Content Views** and select the Content View that you want to promote.
2. コンテンツビューの **バージョン** タブをクリックします。
3. プロモートするバージョンを選択し、**アクション** 列で、**プロモート** をクリックします。
4. コンテンツビューをプロモートする環境を選択し、**バージョンのプロモート** をクリックします。
5. **プロモート** ボタンを再度クリックします。今度は **テスト** 環境を選択して **バージョンのプロモート** をクリックします。
6. Finally click on the **Promote** button again. Select the **Production** environment and click **Promote Version**.

Now the repository for the Content View appears in all environments.

For CLI Users

- Promote the Content View using the **hammer content-view version promote** each time:

```
# hammer content-view version promote \  
--content-view "Database" \  
--version 1 \  
--to-lifecycle-environment "Development" \  
--organization "My_Organization"
```

```
# hammer content-view version promote \
--content-view "Database" \
--version 1 \
--to-lifecycle-environment "Testing" \
--organization "My_Organization"
# hammer content-view version promote \
--content-view "Database" \
--version 1 \
--to-lifecycle-environment "Production" \
--organization "My_Organization"
```

Now the database content is available in all environments.

コンテンツビューにホストを登録するには、『[ホストの管理](#)』ガイドの「[ホストの登録](#)」を参照してください。

8.5. COMPOSITE CONTENT VIEWS OVERVIEW

A Composite Content View combines the content from several Content Views. For example, you might have separate Content Views to manage an operating system and an application individually. You can use a Composite Content View to merge the contents of both Content Views into a new repository. The repositories for the original Content Views still exist but a new repository also exists for the combined content.

さまざまなデータベースサーバーをサポートするアプリケーションを開発する場合には、**example_application** は次のように表示されます。

example_software
Application
Database
Operating System

Example of four separate Content Views:

- Red Hat Enterprise Linux (Operating System)
- PostgreSQL (Database)
- MariaDB (Database)
- **example_software** (Application)

From the previous Content Views, you can create two Composite Content Views.

Example Composite Content View for a PostgreSQL database:

複合コンテンツビュー 1: PostgreSQL の example_software
example_software (Application)

複合コンテンツビュー 1: PostgreSQL の example_software

PostgreSQL (Database)

Red Hat Enterprise Linux (Operating System)

Example Composite Content View for a MariaDB:

Composite Content View 2 - example_software on MariaDB

example_software (Application)

MariaDB (Database)

Red Hat Enterprise Linux (Operating System)

これで各コンテンツビューは別個に管理して公開されます。アプリケーションのバージョンを作成すると、複合コンテンツビューの新バージョンを公開することになります。複合コンテンツビューの作成時に、**自動公開** オプションを選択することも可能です。自動公開オプションを選択して、複合コンテンツビューに含まれるコンテンツビューが再公開されると、自動的に複合コンテンツビューが再公開されません。

Repository Restrictions

複合コンテンツビューは、各リポジトリで1つしか許可されません。たとえば、同じリポジトリを使用したコンテンツビューを2つ追加しようとする、Satellite Server はエラーをレポートします。

8.6. CREATING A COMPOSITE CONTENT VIEW**Procedure**

To create a Composite Content View, complete the following steps:

1. In the Satellite web UI, navigate to **Content > Content Views** and click **Create New View**.
2. **名前** フィールドで、ビューの名前を入力します。Red Hat Satellite 6 では、入力した名前から、**ラベル** フィールドに自動的に入力されます。
3. **説明** フィールドに、ビューの説明を入力します。
4. Select the **Composite View?** check box to create a Composite Content View.
5. オプション: コンテンツビューが再公開されると複合コンテンツビューが自動的に再公開されるようにするには、**自動公開** チェックボックスを選択します。
6. **保存** をクリックします。
7. **コンテンツビューの追加** エリアで、複合コンテンツビューに追加するコンテンツビューを選択して、**Add Content Views** をクリックします。
8. **Publish New Version** をクリックして、複合コンテンツビューを公開します。**説明** フィールドに説明を入力し、**保存** をクリックします。

9. **プロモート** をクリックし、複合コンテンツビューをプロモートするライフサイクル環境を選択し、説明を入力して、**バージョンのプロモート** をクリックします。

For CLI Users

1. Before you create the Composite Content Views, list the version IDs for our existing Content Views:

```
# hammer content-view version list \
--organization "My_Organization"
```

2. 新しい複合コンテンツビューを作成し、**--component-ids** オプションを使用して、コンテンツビューバージョン ID のコンマ区切りのリストを追加します。**--auto-publish** オプションを **yes** に設定すると、複合コンテンツビューに含まれるコンテンツビューが再公開されると、自動的に複合コンテンツビューが再公開されます。

```
# hammer content-view create \
--composite \
--auto-publish yes \
--name "Example_Composite_Content_View" \
--description "Example Composite Content View" \
--component-ids Content_View_Version_IDs \
--organization "My_Organization"
```

3. Publish the Composite Content View:

```
# hammer content-view publish \
--name "Example_Composite_Content_View" \
--description "Initial version of Composite Content View" \
--organization "My_Organization"
```

4. Promote the Composite Content View across all environments:

```
# hammer content-view version promote \
--content-view "Example_Composite_Content_View" \
--version 1 \
--to-lifecycle-environment "Development" \
--organization "My_Organization"
# hammer content-view version promote \
--content-view "Example_Composite_Content_View" \
--version 1 \
--to-lifecycle-environment "Testing" \
--organization "My_Organization"
# hammer content-view version promote \
--content-view "Example_Composite_Content_View" \
--version 1 \
--to-lifecycle-environment "Production" \
--organization "My_Organization"
```

8.7. CONTENT FILTER OVERVIEW

コンテンツビューでは、フィルターを使って特定の RPM コンテンツを追加したり制限したりします。フィルターを使用しないと、選択したリポジトリからのすべてのコンテンツが含まれてしまいます。

There are two types of content filters:

表8.1 Filter Types

Filter Type	Description
Include	コンテンツなしの状態から開始し、選択したリポジトリから追加するコンテンツを選択します。このフィルターを使用して、複数のコンテンツアイテムを組み合わせます。
Exclude	You start with all content from selected repositories, then select which content to remove. Use this filter when you want to use most of a particular content repository but exclude certain packages, such as blacklisted packages. The filter uses all content in the repository except for the content you select.

包含と除外のフィルターの組み合わせ

包含と除外のフィルターの組み合わせを使用してコンテンツビューを公開すると、最初に包含フィルターが適用され、次に除外フィルターが適用されます。この場合には、先に、含めるコンテンツを選択してから、このサブセットから除外するコンテンツを選択することになります。

Content Types

There are also four types of content to filter:

表8.2 Content Types

Content Type	Description
Package	Filter packages based on their name and version number.
Package Group	パッケージグループ別に、パッケージをフィルタリングします。パッケージグループの一覧は、コンテンツビューに追加されたりポジトリ別になっています。
Erratum (by ID)	フィルターに追加する特定のエラータを選択します。エラーター一覧は、コンテンツビューに追加されたりポジトリ別になっています。
Erratum (by Date and Type)	Select a issued or updated date range and errata type (Bugfix, Enhancement, or Security) to add to the filter.

パッケージの依存関係の解決とフィルター

Filters do not resolve any dependencies of the packages listed in the filters. Ensure to add package dependencies to the filter. This might require some level of testing to determine what dependencies are required.

8.8. CONTENT FILTER EXAMPLES

Use any of the following examples with the procedure that follows to build custom content filters.

Example 1

Create a repository with the base Red Hat Enterprise Linux packages. This filter requires a Red Hat Enterprise Linux repository added to the Content View.

Filter:

- **Inclusion Type:** Include
- **Content Type:** Package Group
- **Filter:** Select only the **Base** package group

Example 2

セキュリティーアップデートを除く、特定日以降の全エラータを除外するリポジトリを作成します。これは、重要なセキュリティーアップデートは即座に適用する必要があるが、その他のシステムアップデートは定期的に行う場合に便利です。このフィルターでは、Red Hat Enterprise Linux リポジトリをコンテンツビューに追加しておく必要があります。

Filter:

- **Inclusion Type:** Exclude
- **Content Type:** Erratum (by Date and Type)
- **フィルター:** バグ修正と機能強化のエラータイプのみを選択し、セキュリティーの選択は解除します。日付タイプを更新日に設定します。エラータを制限する日付を開始日に設定します。終了日は空白にして、セキュリティー以外の新たなエラータにフィルターが適用されないようにします。

Example 3

例1と例2の組み合わせで、オペレーティングシステムパッケージのみが必要で、最近のバグ修正と機能強化エラータを除外します。この場合には、同一のコンテンツビューにアタッチされた2つのフィルターが必要です。コンテンツビューは組み込みフィルターを最初に適用してから、除外フィルターを適用します。

Filter 1:

- **Inclusion Type:** Include
- **Content Type:** Package Group
- **Filter:** Select only the **Base** package group

Filter 2:

- **Inclusion Type:** Exclude
- **Content Type:** Erratum (by Date and Type)
- **フィルター:** バグ修正と機能強化のエラータイプのみを選択し、セキュリティーの選択は解除します。日付タイプを更新日に設定します。エラータを制限する日付を開始日に設定しま

す。終了日は空白にして、セキュリティー以外の新たなエラータにフィルターが適用されないようにします。

コンテンツフィルターの機能例については、[How do content filters work in Satellite 6](#) を参照してください。

8.9. CREATING A CONTENT FILTER

以下の手順を使用して、コンテンツフィルターを作成します。フィルターをビルドする方法の例については、「[Content Filter Examples](#)」を参照してください。

Procedure

To create a content filter, complete the following steps:

1. In the Satellite web UI, navigate to **Content > Content Views** and select a Content View.
2. **Yum コンテンツ > フィルター**に移動し、**新規フィルター**をクリックします。
3. **名前** フィールドに、フィルターの名前を入力します。
4. **コンテンツタイプ**の一覧から、絞り込むコンテンツタイプを選択します。新しいフィルターのコンテンツタイプに選択した内容に応じて、異なるオプションが表示されます。
5. **包含タイプ** リストから、**包含** または **除外** を選択します。
6. **説明** フィールドに、フィルターの説明を入力し、**保存** をクリックします。
7. **コンテンツタイプ** に入力した内容に応じて、必要なフィルターを作成するルールを追加します。
8. **影響のあるリポジトリ** タブをクリックして、どのリポジトリがこのフィルターを使用するかを選択します。
9. **新規バージョンの公開** をクリックして、フィルタリングされたリポジトリを公開します。**説明** フィールドに変更の説明を入力し、**保存** をクリックします。

You can promote this Content View across all environments.

For CLI Users

1. Add a filter to the Content View. Use the **--inclusion false** option to set the filter to an Exclude filter:

```
# hammer content-view filter create \
--name "Errata Filter" \
--type erratum --content-view "Example_Content_View" \
--description "My latest filter" \
--inclusion false \
--organization "My_Organization"
```

2. Add a rule to the filter:

```
# hammer content-view filter rule create \
--content-view "Example_Content_View" \
--content-view-filter "Errata Filter" \
```

```
--start-date "YYYY-MM-DD" \  
--types enhancement,bugfix \  
--date-type updated \  
--organization "My_Organization"
```

3. Publish the Content View:

```
# hammer content-view publish \  
--name "Example_Content_View" \  
--description "Adding errata filter" \  
--organization "My_Organization"
```

4. Promote the view across all environments:

```
# hammer content-view version promote \  
--content-view "Example_Content_View" \  
--version 1 \  
--to-lifecycle-environment "Development" \  
--organization "My_Organization"  
# hammer content-view version promote \  
--content-view "Example_Content_View" \  
--version 1 \  
--to-lifecycle-environment "Testing" \  
--organization "My_Organization"  
# hammer content-view version promote \  
--content-view "Example_Content_View" \  
--version 1 \  
--to-lifecycle-environment "Production" \  
--organization "My_Organization"
```

第9章 SYNCHRONIZING CONTENT BETWEEN SATELLITE SERVERS

Red Hat Satellite 6.5 は Inter-Satellite Synchronization (ISS) を使用して、コンテンツの Satellite Server 間、または Satellite Server の組織間を同期します。

You can use ISS in the following scenarios:

- If you have both connected and disconnected Satellite Servers, and want to copy content from the connected servers to the disconnected servers. For example, you require complete isolation of management infrastructure for security or other purposes.
- Satellite Server の一部のコンテンツのみを他の Satellite Server にコピーする場合。たとえば、IT 部門が検証するコンテンツビューが Satellite Server であり、このコンテンツビューのコンテンツを他の Satellite Server へコピーする場合などです。
- If you want to clone a Content View from one organization to another organization on Satellite Server.

ISS は、Satellite Server から Capsule Server へのコンテンツの同期には使用できません。Capsule Server はネイティブで同期をサポートします。詳細は、『Red Hat Satellite 6 のプランニング』の「[Capsule Server の概要](#)」を参照してください。

9.1. EXPORTING A CONTENT VIEW VERSION

コンテンツビューのバージョンを、Satellite Server からアーカイブファイルにエクスポートして、このアーカイブファイルを使用し、別の Satellite Server か、別の Satellite Server の組織に同じコンテンツビューバージョンを作成します。Satellite では、複合コンテンツビューはエクスポートされません。エクスポートしたアーカイブファイルには、以下のデータが含まれます。

- A JSON file containing Content View version metadata
- An archive file containing all the repositories included into the Content View version

Satellite Server は、コンテンツビューのバージョンに追加された RPM およびキックスタートファイルのみをエクスポートします。Satellite では、以下の内容はエクスポートされません。

- Puppet content
- Docker content
- OSTree content
- Content View definitions and metadata, such as package filters.

hammer content-view version export コマンドの変更

hammer content-view version export および **hammer content-view version import** の新しいコマンドは、以前の Satellite のバージョンのコマンドと機能の仕方が違います。以前の機能は、**hammer content-view version export-legacy** コマンドで利用でき、新しい機能には存在しない機能が、以前の機能には含まれています。

1. オンラインの Satellite Server からオフラインの Satellite Server に直接パッチを適用できません。**hammer content-view version export-legacy** は、CDN の構造をエクスポートするので、Red Hat カスタマーポータルからの DVD ISO を使用する必要はありません。

2. yum 以外のコンテンツを含むコンテンツビューをエクスポートする場合は、**hammer content-view version export-legacy** を使用すると、yum 以外のコンテンツをスキップしてコンテンツビューをエクスポートしますが、**hammer content-view version export** は、yum 以外のリポジトリを削除するようにプロンプトを表示して失敗します。

以前の機能に関する詳細情報は、『Satellite 6.4 コンテンツ管理ガイド』の「[Satellite Server 間のコンテンツの同期](#)」を参照してください。

Prerequisites

To export a Content View, ensure that the Satellite Server where you want to export meets the following conditions:

- Ensure that the export directory has free storage space to accommodate the export.
- Ensure that the `/var/lib/pulp/` directory has free storage space equivalent to the size of the repositories being exported for temporary files created during the export process.
- `/var/cache/pulp` ディレクトリーに、エクスポートプロセス中に作成される一時ファイルに、エクスポートされるリポジトリのサイズの 2 倍の空き容量があることを確認します。
- エクスポートするコンテンツビュー内の全リポジトリでダウンロードポリシーを**即時**に設定していることを確認します。詳細については、「[Download Policies Overview](#)」を参照してください。
- リポジトリ設定ページで、インポートするリポジトリの **Mirror on Sync** チェックボックスの選択が解除されていることを確認します。
- エクスポートする製品が、必要な日付に同期されることを確認します。

To Export a Content View Version:

1. List Content Views to determine the ID of a Content View version you want to export:

```
# hammer content-view version list \
--organization "Default Organization"
```

2. Export the version of a Content View. Specify the directory where to store the export with the **--export-dir** option and the ID of the Content View version that you export with the **--id** option. The **pulp_export_destination** setting does not work for this procedure.

```
# hammer content-view version export --export-dir export_directory \
--id content_view_version_ID
```

3. エクスポートしたコンテンツビューバージョンが含まれるアーカイブが、エクスポートディレクトリーにあることを確認します。

```
# ls export_directory
export-1.tar
```

9.2. IMPORTING A CONTENT VIEW VERSION

You can use the archive that the **hammer content-view version export** command outputs to create a version of a Content View with the same content as the exported Content View version. For more information about exporting a Content View version, see [「Exporting a Content View Version」](#).

コンテンツビューバージョンをインポートすると、メジャーバージョン番号、マイナーバージョン番号が同じで、同じパッケージ、エラータを含む同じリポジトリが含まれます。エクスポートされたアーカイブ内の **json** ファイルの **major** および **minor** 設定を変更して、バージョン番号をカスタマイズできます。

Prerequisites

コンテンツビューをインポートするには、インポート先の Satellite Server が、以下の条件を満たしていることを確認します。

- エクスポートしたコンテンツビューバージョンに含まれる同じ Yum コンテンツを追加することを確認します。
- コンテンツビューをオフラインの Satellite にインポートする場合には、Red Hat カスタマーポータルからコンテンツ ISO をダウンロードして、Satellite Server にインポートする日付町があります。詳細は [付録B Importing Content ISOs into a Disconnected Satellite](#) を参照してください。
- エクスポートする製品が、必要な日付に同期されることを確認します。
- エクスポートするコンテンツビュー内の全リポジトリでダウンロードポリシーを**即時**に設定していることを確認します。詳細については、「[Download Policies Overview](#)」を参照してください。
- リポジトリ設定ページで、インポートするリポジトリの **Mirror on Sync** チェックボックスの選択が解除されていることを確認します。

Procedure

1. インポート先の Satellite Server の `/var/lib/pulp/katello-export` ディレクトリーに、エクスポートしたコンテンツビューバージョンが含まれるアーカイブファイルをコピーします。
2. インポート先の Satellite Server で、エクスポートしたコンテンツビューと、名前とラベルが同じコンテンツビューを作成します。
3. エクスポートしたコンテンツビューバージョンのアーカイブを Satellite Server にインポートします。

```
# hammer content-view version import \  
--export-tar /var/lib/pulp/katello-export/exported_CV_archive \  
--organization-id Your_Organization_ID
```

4. List Content Views for your organization to verify that you import the Content View version successfully:

```
# hammer content-view version list --organization "Your_Organization"
```

第10章 MANAGING ACTIVATION KEYS

Activation keys provide a method to automate system registration and subscription attachment. You can create multiple keys and associate them with different environments and Content Views. For example, you might create a basic activation key with a subscription for Red Hat Enterprise Linux workstations and associate it with Content Views from a particular environment.

コンテンツホストの登録時にアクティベーションキーを使用して、プロセスのスピードアップ、単純化、一貫性の向上を図ることができます。

アクティベーションキーを使用して、コンテンツホストの次のプロパティを定義できます。

- 関連付けるサブスクリプションおよびサブスクリプションのアタッチの動作。
- Available products and repositories.
- A life cycle environment and a Content View.
- Host collection membership.

サブスクリプションが十分にある場合には、同じアクティベーションキーを複数のコンテンツホストに適用できます。ただし、アクティベーションキーはコンテンツホストの初期設定のみを行います。コンテンツホストを組織に登録した後は、組織のコンテンツをコンテンツホストに手動でアタッチできます。

A content host can be associated with multiple activation keys that are combined to define the host settings. In case of conflicting settings, the last specified activation key takes precedence.

アクティベーションキーは、ホストが登録されている場合にのみ使用できます。アクティベーションキーに変更が加えられた場合には、それ以降、改訂されたアクティベーションキーで登録されるホストにのみ、変更が適用され、既存のホストには加えられません。

10.1. CREATING AN ACTIVATION KEY

With activation keys, you can define how the content host is subscribed during registration. The subscription behavior defined by the activation key depends on two factors:

- Are there any subscriptions associated with the activation key?
- Is the auto-attach option enabled?

上記の要因をもとに、アクティベーションキーを使用してサブスクライブするシナリオを3つ想定できます。

1. Activation key with no subscriptions specified.

With no subscriptions specified and auto-attach enabled, hosts using the activation key search for the best fitting subscription from the ones provided by Satellite Server. This is similar to entering the **subscription-manager --auto-attach** command.

2. 自動アタッチ用にカスタムサブスクリプションプールを指定するアクティベーションキー。

If there are subscriptions specified and auto-attach is enabled, hosts using the activation key select the best fitting subscription from the list specified in the activation key.

3. Activation key with the exact set of subscriptions.

If there are subscriptions specified and auto-attach is disabled, hosts using the activation key are associated with all subscriptions specified in the activation key.

Custom Products

カスタム製品 (通常は Red Hat が提供しないコンテンツを含む製品) がアクティベーションキーに割り当てられている場合には、この製品は、自動アタッチの設定の有無にかかわらず、登録されたコンテンツホストに対して常に有効になります。

Procedure

To create an activation key, complete the following steps:

1. In the Satellite web UI, navigate to **Content > Activation keys** and click **Create Activation Key**.
2. **名前** フィールドに、アクティベーションキーの名前を入力します。
3. If you want to set a limit, clear the **Unlimited hosts** check box, and in the **Limit** field, enter the maximum number of systems you can register with the activation key. If you want unlimited hosts to register with the activation key, ensure the **Unlimited Hosts** check box is selected.
4. In the **Description** field, enter a description for the activation key.
5. **環境** 一覧から、使用する環境を選択します。
6. **コンテンツビュー** 一覧から、使用するコンテンツビューを選択します。
7. **保存** をクリックし、アクティベーションキーウィンドウに新しいアクティベーションキーが表示されたら、編集する名前をクリックします。
8. アタッチする製品と、登録時に有効にするリポジトリを定義するには、**サブスクリプション** タブをクリックし、追加するサブスクリプションを選択して、**追加** をクリックします。
9. Satellite Web UI で、**製品コンテンツ** ページに移動します。たとえば、Red Hat Satellite Tools リポジトリを有効にするには、**Red Hat Satellite Tools 6.5 (for RHEL 7 Server) (RPMs) - Enabled** で、**はいに上書き** を選択します。
10. **保存** をクリックします。

For CLI Users

1. Create the activation key:

```
# hammer activation-key create \
--name "My_Activation_Key" \
--unlimited-hosts \
--description "Example Stack in the Development Environment" \
--lifecycle-environment "Development" \
--content-view "Stack" \
--organization "My_Organization"
```

2. Obtain a list of your subscription IDs:

```
# hammer subscription list --organization "My_Organization"
```

3. Attach the Red Hat Enterprise Linux subscription UUID to the activation key:

```
# hammer activation-key add-subscription \
--name "My_Activation_Key" \
```

```
--subscription-id ff808181533518d50152354246e901aa \  
--organization "My_Organization"
```

- List the product content associated with the activation key:

```
# hammer activation-key product-content \  
--name "My_Activation_Key" \  
--organization "My_Organization"
```

- Override the default auto-enable status for the Red Hat Satellite Tools 6.5 repository. The default status is set to disabled. To enable, enter the following command:

```
# hammer activation-key content-override \  
--name "My_Activation_Key" \  
--content-label rhel-7-server-satellite-tools-6.5-rpms \  
--value 1 \  
--organization "My_Organization"
```

10.2. UPDATING SUBSCRIPTIONS ASSOCIATED WITH AN ACTIVATION KEY

You can change the subscriptions associated with an activation key using the web UI or using the Hammer command-line tool.

アクティベーションキーへの変更は、変更後にプロビジョニングしたマシンにのみ適用されます。既存のコンテンツホストでサブスクリプションを更新する方法は「[Bulk Updating Content Hosts' Subscriptions](#)」を参照してください。

Procedure

To update the subscriptions associated with an activation key, complete the following steps:

- Satellite Web UI で、**コンテンツ > アクティベーションキー** に移動し、アクティベーションキーの名前をクリックします。
- Click the **Subscriptions** tab.
- To remove subscriptions, select **List/Remove**, and then select the check boxes to the left of the subscriptions to be removed and then click **Remove Selected**.
- サブスクリプションを追加するには、**追加** を選択してから、追加するサブスクリプションの左側のチェックボックスを選択し、**選択した項目を追加** をクリックします。
- リポジトリセット** タブをクリックし、リポジトリのステータス設定を確認します。
- To enable or disable a repository, select the check box for a repository and then change the status using the **Select Action** list.
- Click the **Details** tab, select a Content View for this activation key, and then click **Save**.

For CLI Users

- 現在アクティベーションキーが含まれているサブスクリプションを一覧表示します。

```
# hammer activation-key subscriptions \
--name My_Activation_Key \
--organization "My_Organization"
```

2. Remove the required subscription from the activation key:

```
# hammer activation-key remove-subscription \
--name "My_Activation_Key" \
--subscription-id ff808181533518d50152354246e901aa \
--organization "My_Organization"
```

For the **--subscription-id** option, you can use either the UUID or the ID of the subscription.

3. Attach new subscription to the activation key:

```
# hammer activation-key add-subscription \
--name "My_Activation_Key" \
--subscription-id ff808181533518d50152354246e901aa \
--organization "My_Organization"
```

For the **--subscription-id** option, you can use either the UUID or the ID of the subscription.

4. List the product content associated with the activation key:

```
# hammer activation-key product-content \
--name "My_Activation_Key" \
--organization "My_Organization"
```

5. Override the default auto-enable status for the required repository:

```
# hammer activation-key content-override \
--name "My_Activation_Key" \
--content-label content_label \
--value 1 \
--organization "My_Organization"
```

For the **--value** option, enter **1** for enable, **0** for disable.

10.3. CSV ファイルを使用してアクティベーションキーに関連付けられたサブスクリプションを更新する手順

You can update the subscriptions associated with an activation key by downloading a CSV file, making changes to the Activation Key settings, and then uploading the changed CSV file.

Subscription information in the CSV File

CSV ファイルの最後の列にサブスクリプション情報が含まれます。サブスクリプション情報は、CSV ファイル内の1つのフィールドであり、引用符で囲まれたテキストにコンマを含めることができます。

The entries in the CSV file have the following format:

```
Name Organization Description Limit Environment Content View Host Collections Auto-Attach
Service Level Release Version Subscriptions
```

サブスクリプション フィールドに含まれる例を示します。

```
Automatic|RH1234|Red Hat Enterprise Linux Server, Standard (Physical or Virtual Nodes)|11223344|55667788
```

```
1|MCT0369|Red Hat Satellite Capsule Server|11223344|55667788
```

The **Subscriptions** field has the following format:

- 割り当てられるサブスクリプションの数。 **自動** に設定できます。
- The subscription's identification number.
- The subscription's name.
- The contract number.
- The account number.

Procedure

To update the subscription list using a CSV file, complete the following procedure:

1. Export the subscriptions from Satellite Server to a CSV file. For this example, **a_keys.csv**:

```
# hammer --server https://satellite.example.com csv activation-keys \  
--export --file a_keys.csv --organization "My_Organization"
```

2. To view the columns of the file, enter the following command:

```
# column -s, -t < a_keys.csv | less -S
```

3. Change the required values in the CSV file. You can use an editor, with a CSV plug-in, or **sed** to change strings in the **Subscriptions** field.
4. Make a backup of the file.

```
# cp a_keys.csv a_keys.csv.backup
```

5. Edit the string you want to change. For example:

```
# sed -i "s/Automatic|RH1234|Red Hat Enterprise Linux Server/Automatic|RH4567|Red Hat Enterprise Linux Server/g" a_keys.csv
```

6. Confirm only the required changes were made. For example:

```
# diff a_keys.csv a_keys.csv.backup
```

7. Upload the changed file to Satellite Server.

```
# hammer --server https://satellite.example.com csv activation-keys \  
--file a_keys.csv
```

10.4. USING ACTIVATION KEYS FOR HOST REGISTRATION

You can use activation keys to complete the following tasks:

- Registering new hosts during provisioning through Red Hat Satellite 6. The kickstart provisioning templates in Red Hat Satellite 6 contain commands to register the host using an activation key that is defined when creating a host.
- 既存の Red Hat Enterprise Linux ホストを登録する。Red Hat サブスクリプションマネージャーが登録に Satellite Server を使用するように設定し、**subscription-manager register** コマンドの実行時にアクティベーションキーを指定します。

Procedure

アクティベーションキーを使用して、既存の Red Hat Enterprise Linux 7 ホストを、Satellite Server に登録するには、以下の手順を行います。

1. Satellite Server 用のコンシューマー RPM をダウンロードします。これは、ホストの Web サーバーの **pub** ディレクトリーに配置されています。たとえば、ホスト名が **satellite.example.com** の Satellite Server の場合は、登録するホストで以下のコマンドを実行します。

```
# rpm -Uvh http://satellite.example.com/pub/katello-ca-consumer-latest.noarch.rpm
```

This RPM installs the necessary certificates for accessing repositories on Satellite Server and configures Red Hat Subscription Manager to use the server's URL.

2. On the host, enter the following command to register the host to Satellite using the activation key:

```
# subscription-manager register --activationkey="My_Activation_Key" \
--org="My_Organization"
```

3. To view a list of hosts for an organization, on Satellite Server, enter the following command:

```
# hammer host list --organization "My_Organization"
```

4. Satellite Server にホストを登録したら、ホストに **katello-agent** パッケージをインストールして、ホストから Satellite Server にレポートできるようにします。

```
# yum install katello-agent
```

このパッケージは、Red Hat Satellite Tools 6.5 リポジトリーに含まれます。

Multiple Activation Keys

コンテンツホストの登録時に複数のアクティベーションキーを使用できます。特定のサブスクリプションセット用にアクティベーションキーを作成し、コンテンツホストの要件に合わせて、これらのアクティベーションキーを組み合わせることができます。たとえば、以下のコマンドは VDC と OpenShift の両方のサブスクリプションでコンテンツホストを組織に登録します。

```
# subscription-manager register --org="My_Organization" \
--activationkey="ak-VDC,ak-OpenShift"
```

Settings Conflicts

If there are conflicting settings in activation keys, the rightmost key takes precedence.

- Settings that conflict: **Service Level**, **Release Version**, **Environment**, **Content View**, and **Product Content**.
- Settings that do not conflict and the host gets the union of them: **Subscriptions** and **Host Collections**.
- Settings that influence the behavior of the key itself and not the host configuration: **Content Host Limit** and **Auto-Attach**.

10.5. ENABLING AUTO-ATTACH

When auto-attach is enabled on an activation key and there are subscriptions associated with the key, the subscription management service selects and attaches the best-matched associated subscriptions based on a set of criteria like currently-installed products, architecture, and preferences like service level.

You can enable auto-attach and have no subscriptions associated with the key. This type of key is commonly used to register virtual machines when you do not want the virtual machine to consume a RHEL subscription but to inherit a RHEL Virtual Data Center (VDC) subscription from the hypervisor.

Auto-attach is enabled by default. Disable the option if you want to force attach all subscriptions associated with the activation key.

Procedure

To enable auto-attach, complete the following steps:

1. Satellite Web UI で、**コンテンツ > アクティベーションキー** に移動します。
2. Click the activation key name that you want to edit.
3. Click the **Subscriptions** tab.
4. **自動アタッチ**の隣にある編集アイコンをクリックします。
5. Select or clear the check box to enable or disable auto-attach.
6. **保存**をクリックします。

自動アタッチアクティベーションキーを使用して仮想コンテンツホストを Satellite Server に登録するには、まず **virt-who** ユーティリティを使用して、この仮想コンテンツホストを Virtual Datacenter (VDC) サブスクリプションのエンタイトルメントに含まれるハイパーバイザーにマッピングします。この前提条件が満たされない場合には、仮想ホストは、24 時間のみ、一時的に仮想サブスクリプションに登録されます。詳細は、『[仮想インスタンスガイド](#)』の「[仮想ゲストのサブスクリプションの適用](#)」を参照してください。

For CLI Users

To enable auto-attach on the activation key:

```
# hammer activation-key update --name "My_Activation_Key" \
--organization "My_Organization" --auto-attach true
```

10.6. SETTING THE SERVICE LEVEL

アクティベーションキーで作成した新規ホストのデフォルトのサービスレベルを定義するように、アクティベーションキーを設定できます。デフォルトのサービスレベルを設定すると、ホストにアタッチするのに適したサブスクリプションのみが選択されます。たとえば、アクティベーションキーのデフォルトのサービスレベルが Premium に設定されている場合には、Premium サービスレベルのサブスクリプションのみが、登録時にホストに割り当てられます。

Procedure

To set the service level, complete the following steps:

1. Satellite Web UI で、**コンテンツ > アクティベーションキー** に移動します。
2. Click the activation key name you want to edit.
3. **サービスレベル**の隣にある編集アイコンをクリックします。
4. Select the required service level from the list. The list only contains service levels available to the activation key.
5. **保存**をクリックします。

For CLI Users

To set a default service level to Premium on the activation key:

```
# hammer activation-key update --name "My_Activation_Key" \  
--organization "My_Organization" --service-level premium
```

第11章 MANAGING ERRATA

Red Hat では、品質管理およびリリースプロセスの一部として、お客様に Red Hat RPM の公式リリースのアップデートを提供しています。Red Hat では、アップデートを説明するアドバイザリーと共に、関連パッケージのグループをエラータにコンパイルします。アドバイザリーには以下の 3 種類があります (重要度の高い順)。

Security Advisory

Describes fixed security issues found in the package. The security impact of the issue can be Low, Moderate, Important, or Critical.

Bug Fix Advisory

Describes bug fixes for the package.

Product Enhancement Advisory

Describes enhancements and new features added to the package.

Red Hat Satellite 6 imports this errata information when synchronizing repositories with Red Hat's Content Delivery Network (CDN). Red Hat Satellite 6 also provides tools to inspect and filter errata, allowing for precise update management. This way, you can select relevant updates and propagate them through Content Views to selected content hosts.

Errata are labeled according to the most important advisory type they contain. Therefore, errata labeled as **Product Enhancement Advisory** can contain only enhancement updates, while **Bug Fix Advisory** errata can contain both bug fixes and enhancements, and **Security Advisory** can contain all three types.

In Red Hat Satellite, there are two keywords that describe an erratum's relationship to the available content hosts:

Applicable

Erratum applies to one or more content hosts, which means it updates packages present on the content host. Applicable errata are not yet accessible by the content host.

Installable

Erratum applies to one or more content hosts and it has been made available to the content host. Installable errata are present in the content host's life cycle environment and Content View, but are not yet installed. This way, errata can be installed by users who have permissions to manage content hosts, but are not entitled for errata management at higher levels.

本章では、エラータの管理方法と 1 つのホストまたは複数のホストへの適用方法を説明します。

11.1. INSPECTING AVAILABLE ERRATA

The following procedure describes how to view and filter the available errata and how to display metadata of the selected advisory.

1. Navigate to **Content > Errata** to view the list of available errata.
2. Use the filtering tools at the top of the page to limit the number of displayed errata:
 - Select the repository to be inspected from the list. **All Repositories** is selected by default.
 - The **Applicable** check box is selected by default to view only errata applicable to the selected repository. Select the **Installable** check box to view only errata marked as installable.

- To search the table of errata, type the query in the **Search** field in the form of:

parameter operator value

検索に使用できるパラメーターの一覧は、[表11.1「Parameters Available for Errata Search」](#)を参照してください。適用可能な演算子の一覧は、『Red Hat Satellite の管理』の「[詳細な検索に対してサポートされる演算子](#)」を参照してください。入力時に自動サジェスト機能が利用できます。**and** 演算子と **or** 演算子を使用してクエリーを組み合わせることもできます。たとえば、**kernel** パッケージに関するセキュリティーアドバイザリーのみを表示するには、以下を入力します。

```
type = security and package_name = kernel
```

Press **Enter** to start the search.

- Click the **Errata ID** of the erratum you want to inspect:
 - The **Details** tab contains the description of the updated package as well as documentation of important fixes and enhancements provided by the update.
 - On the **Content Hosts** tab, you can apply the erratum to selected content hosts as described in [「Applying Errata to Multiple Hosts」](#).
 - The **Repositories** tab lists repositories that already contain the erratum. You can filter repositories by the environment and Content View, and search for them by the repository name.

表11.1 Parameters Available for Errata Search

Parameter	Description	Example
bug	Search by the Bugzilla number.	bug = 1172165
cve	Search by the CVE number.	cve = CVE-2015-0235
id	Search by the errata ID. The auto-suggest system displays a list of available IDs as you type.	id = RHBA-2014:2004
issued	Search by the issue date. You can specify the exact date, like "Feb16,2015", or use keywords, for example "Yesterday", or "1 hour ago". The time range can be specified with the use of the "<" and ">" operators.	issued < "Jan 12,2015"
package	Search by the full package build name. The auto-suggest system displays a list of available packages as you type.	package = glib2-2.22.5-6.el6.i686

Parameter	Description	Example
package_name	Search by the package name. The auto-suggest system displays a list of available packages as you type.	<code>package_name = glib2</code>
severity	Search by the severity of the issue fixed by the security update. Specify Critical , Important , or Moderate .	<code>severity = Critical</code>
title	Search by the advisory title.	<code>title ~ openssl</code>
type	Search by the advisory type. Specify security , bugfix , or enhancement .	<code>type = bugfix</code>
updated	Search by the date of the last update. You can use the same formats as with the issued parameter.	<code>updated = "6 days ago"</code>

11.2. SUBSCRIBING TO ERRATA NOTIFICATIONS

Satellite ユーザー向けにメール通知を設定することができます。ユーザーには、リポジトリの同期後に、適用可能かつインストール可能なエラータのまとめ、通知が、コンテンツビュープロモーションで送信されます。詳しい情報は、『Red Hat Satellite の管理』ガイドの「[電子メール通知の設定](#)」を参照してください。

11.3. LIMITATIONS TO REPOSITORY DEPENDENCY RESOLUTION

Satellite 6 には、リポジトリ依存関係の解決が必要な問題が複数あり、これは既知の問題です。詳細は、[BZ#1508169](#)、[BZ#1640420](#)、[BZ#1508169](#)、[BZ#1629462](#) を参照してください。Satellite でコンテンツビューの増分更新を使用すると、リポジトリ依存関係の問題がいくつか解決しますが、リポジトリレベルの依存関係の解決で問題が残る場合があります。

When a repository update becomes available with a new dependency, Satellite retrieves the newest version of the package to solve the dependency, even if there are older versions available in the existing repository package. This can create further dependency resolution problems when installing packages.

Example scenario

A repository on your client has the package **example_repository-1.0** with the dependency **example_repository-libs-1.0**. The repository also has another package **example_tools-1.0**.

A security erratum becomes available with the package **example_tools-1.1**. The **example_tools-1.1** package requires the **example_repository-libs-1.1** package as a dependency.

After an incremental Content View update, the **example_tools-1.1**, **example_tools-1.0**, and **example_repository-libs-1.1** are now in the repository. The repository also has the packages **example_repository-1.0** and **example_repository-libs-1.0**. Note that the incremental update to the

Content View did not add the package **example_repository-1.1**. Because you can install all these packages using yum, no potential problem is detected. However, when the client installs the **example_tools-1.1** package, a dependency resolution problem occurs because both **example_repository-libs-1.0** and **example_repository-libs-1.1** cannot be installed.

現在、この問題の回避策はありません。RPMの基本セットから、適用されるエラータまでのメジャー Y リリースの期間が長いほど、依存関係の解決の問題が発生する可能性が高くなります。

11.4. CREATING A CONTENT VIEW FILTER FOR ERRATA

You can use content filters to limit errata. Such filters include:

- **ID** - Select specific erratum to allow into your resulting repositories.
- **Date Range** - Define a date range and include a set of errata released during that date range.
- **Type** - Select the type of errata to include such as bug fixes, enhancements, and security updates.

Create a content filter to exclude errata after a certain date. This ensures your production systems in the application life cycle are kept up to date to a certain point. Then you can modify the filter's start date to introduce new errata into your testing environment to test the compatibility of new packages into your application life cycle.

Prerequisites

- A Content View with the repositories that contain required errata is created. For more information, see [「Creating a Content View」](#).

Procedure

1. In the Satellite web UI, navigate to **Content > Content Views** and select a Content View that you want to use for applying errata.
2. **Yum コンテンツ > フィルター**に移動し、**新規フィルター**をクリックします。
3. In the **Name** field, enter **Errata Filter**.
4. From the **Content Type** list, select **Erratum - Date and Type**
5. From the **Inclusion Type** list, select **Exclude**.
6. In the **Description** field, enter **Exclude errata items from YYYY-MM-DD**.
7. **保存**をクリックします。
8. For **Errata Type**, select the check boxes of errata types you want to exclude. For example, select the **Enhancement** and **Bugfix** check boxes and clear the **Security** check box to exclude enhancement and bugfix errata after certain date, but include all the security errata.
9. For **Date Type**, select one of two check boxes:
 - **Issued On** for the issued date of the erratum.
 - **Updated On** for the date of the erratum's last update.

10. Select the **Start Date** to exclude all errata on or after the selected date.
11. Leave the **End Date** field blank.
12. **保存**をクリックします。
13. Click **Publish New Version** to publish the resulting repository.
14. Enter **Adding errata filter** in the **Description** field.
15. **保存**をクリックします。
When the Content View completes publication, notice the **Content** column reports a reduced number of packages and errata from the initial repository. This means the filter successfully excluded the all non-security errata from the last year.
16. Click the **Versions** tab.
17. Click **Promote** to the right of the published version.
18. Select the environments you want to promote the Content View version to.
19. In the **Description** field, enter the description for promoting.
20. Click **Promote Version** to promote this Content View version across the required environments.

For CLI Users

1. Create a filter for the errata:

```
# hammer content-view filter create --name "Filter Name" \
--description "Exclude errata items from the YYYY-MM-DD" \
--content-view "CV Name" --organization "Default Organization" \
--type "erratum"
```

2. Create a filter rule to exclude all errata on or after the **Start Date** that you want to set:

```
# hammer content-view filter rule create --start-date "YYYY-MM-DD" \
--content-view "CV Name" --content-view-filter="Filter Name" \
--organization "Default Organization" --types=security,enhancement,bugfix
```

3. Publish the Content View:

```
# hammer content-view publish --name "CV Name" \
--organization "Default Organization"
```

4. Promote the Content View to the lifecycle environment so that the included errata are available to that lifecycle environment:

```
# hammer content-view version promote \
--content-view "CV Name" \
--organization "Default Organization" \
--to-lifecycle-environment "Lifecycle Environment Name"
```

11.5. ADDING ERRATA TO AN INCREMENTAL CONTENT VIEW

If errata are available but not installable, you can create an incremental Content View version to add the errata to your content hosts. For example, if the Content View is version 1.0, it becomes Content View version 1.1, and when you publish, it becomes Content View version 2.0.

1. Satellite Web UI で、**コンテンツ > エラータ** に移動します。
2. **エラータ** の一覧から、適用するエラータの名前をクリックします。
3. エラータを適用するコンテンツホストを選択し、**ホストに適用** をクリックします。これにより、コンテンツビューの増分更新が作成されます。
4. エラータをコンテンツホストに適用する場合は、**公開直後にコンテンツホストにエラータを適用する** チェックボックスを選択します。
5. **確認** をクリックして、エラータを適用します。

For CLI Users

1. List the errata and its corresponding IDs:

```
# hammer erratum list
```

2. List the different content-view versions and the corresponding IDs:

```
# hammer content-view version list
```

3. コンテンツビューバージョンに単一のエラータを適用します。コンマ区切りのリストとして、さらに ID を追加できます。

```
# hammer content-view version incremental-update \
--content-view-version-id 319 --errata-ids 34068b
```

11.6. APPLYING ERRATA TO A HOST

Use these procedures to review and apply errata to a host.

Prerequisites

- Synchronize Red Hat Satellite repositories with the latest errata available from Red Hat. For more information, see [「Synchronizing Red Hat Repositories」](#).
- Satellite Server で、環境とコンテンツビューにホストを登録します。詳細は、『[ホストの管理](#)』ガイドの「[ホストの登録](#)」を参照してください。
- RHEL 7 ホストに、**katello-agent** パッケージをインストールしてください。詳細は、[ホストの管理ガイドの Katello エージェントのインストールセクション](#)を参照してください。

For Red Hat Enterprise Linux 8

RHEL 8 ホストにエラータを適用するには、Satellite Server でリモート実行ジョブを実行するか、ホストを更新できます。リモート実行ジョブの実行の詳細は、『[ホストの管理ガイドの「ホストでのジョブの実行」](#)』を参照してください。

To apply an erratum to a RHEL 8 host, complete the following steps:

1. On Satellite, list all errata for the host:

```
# hammer host errata list \
--host client.example.com
```

2. エラータが含まれるモジュールのストリームを検索します。

```
# hammer erratum info --id ERRATUM_ID
```

3. On the host, update the module stream:

```
# yum update Module_Stream_Name
```

For Red Hat Enterprise Linux 7

To apply an erratum to a RHEL 7 host, complete the following steps:

1. In the Satellite web UI, navigate to **Hosts** > **Content Hosts** and select the host you want to apply errata to.
2. Navigate to the **Errata** tab to see the list of errata.
3. 適用するエラータを選択し、**Apply Selected (選択した項目を適用)** をクリックします。確認画面で、**適用** をクリックします。
4. After the task to update all packages associated with the selected errata completes, click the **Details** tab to view the updated packages.

For CLI Users

To apply an erratum to a RHEL 7 host, complete the following steps:

1. List all errata for the host:

```
# hammer host errata list \
--host client.example.com
```

2. ホストに最新のエラータを適用します。エラータ ID を使用して適用するエラータを特定します。

```
# hammer host errata apply --host "Host Name" \
--errata-ids ERRATUM_ID1,ERRATUM_ID2...
```

11.7. APPLYING ERRATA TO MULTIPLE HOSTS

Use these procedures to review and apply errata to multiple RHEL 7 hosts.

Prerequisites

- Synchronize Red Hat Satellite repositories with the latest errata available from Red Hat. For more information, see [「Synchronizing Red Hat Repositories」](#).
- Register the hosts to an environment and Content View on Satellite Server. For more information, see [Registering a Host](#) in the **Managing Hosts** guide.

- ホストに **katello-agent** パッケージをインストールします。詳細は、『[ホストの管理](#)』ガイドの「[Katello エージェントのインストール](#)」セクションを参照してください。

Procedure

1. Navigate to **Content** > **Errata**.
2. Click the name of an erratum you want to apply.
3. Click to **Content Hosts** tab.
4. エラータの適用先のホストを選択し、**ホストへの適用** をクリックします。
5. Click **Confirm**.

For CLI Users

Although the CLI does not have the same tools as the Web UI, you can replicate a similar procedure with CLI commands.

1. List all installable errata:

```
# hammer erratum list \
--errata-restrict-installable true \
--organization "Default Organization"
```

2. Select the erratum you want to use and list the hosts that this erratum is applicable to:

```
# hammer host list \
--search "applicable_errata = ERRATUM_ID" \
--organization "Default Organization"
```

3. Apply the errata to a single host:

```
# hammer host errata apply \
--host client.example.com \
--organization "Default Organization" \
--errata-ids ERRATUM_ID1,ERRATUM_ID2...
```

4. Enter the following command for each host and replace **\$HOST** with the name of the host for each execution.

```
# for HOST in `hammer \
--csv --csv-separator "|" host list \
--search "applicable_errata = ERRATUM_ID" \
--organization "Default Organization" | tail -n+2 | awk \
-F "|" '{ print $2 }'` ; do echo \
"== Applying to $HOST ==" ; hammer host errata apply \
--host $HOST --errata-ids ERRATUM_ID1,ERRATUM_ID2 ; done
```

This command identifies all hosts with **erratum_IDs** as an applicable erratum and then applies the erratum to each host.

第12章 MANAGING OSTREE CONTENT

OSTree is a tool to manage bootable, immutable, versioned file system trees. You can use a custom OSTree content on a build system, then export an OSTree repository to a static HTTP. Red Hat Enterprise Linux Atomic Server uses OSTree content composed from RPM files as a method to keep the operating system up to date.

You can use Red Hat Satellite 6 to synchronize and manage OSTree branches from an OSTree repository.

In Satellite Server 6.5, OSTree management tools are enabled by default. If you ever have a reason to enable the tool, enter the following command:

```
# satellite-installer --katello-enable-ostree=true
```

12.1. SELECTING RED HAT OSTREE CONTENT TO SYNCHRONIZE

Red Hat CDN provides OSTree Content for you to select and synchronize.

Procedure

To find and synchronize OSTree content, complete the following steps:

1. In the Satellite web UI, navigate to **Content** > **Red Hat Repositories**.
2. From the list, select the **OSTree** content type.
3. In the Available Repositories pane, locate the OSTree repository set you want to use, for example, the **Red Hat Enterprise Linux Atomic Host Treesset** from the **Red Hat Enterprise Linux Atomic Host** product group.
4. Click the **Enable** icon to enable the repository you want to use.
5. Navigate to **Content** > **Products** and click the product that you want to use, for example **Red Hat Enterprise Linux Atomic Host**.
6. Select the upstream synchronization policy for this repository. By default, Satellite synchronizes only the latest OSTree branch.
 - a. Click the repository you want to synchronize.
 - b. From the **Upstream Sync Policy** menu, select one of the following policies to synchronize OSTree branches for this repository:
 - **最新のみ**: 最新の OSTree ブランチのみを同期します。
 - **すべての履歴**: すべての OSTree ブランチを同期します。
 - **カスタム**: 指定した数の OSTree ブランチを同期します。下のフィールドに必要な数を入力します。
 - c. **保存**をクリックします。
7. From the **Select Action** menu, select **Sync Now**.

To view the Synchronization Status

- In the Satellite web UI, navigate to **Content** > **Sync Status** and expand, for example, **Red Hat Enterprise Linux Atomic Host**.

For CLI Users

1. Search the Red Hat Enterprise Linux Server product for **ostree** repositories:

```
# hammer repository-set list \
--product "Red Hat Enterprise Linux Atomic Host" \
--organization "My_Organization" | grep "ostree"
```

2. Enable the **ostree** repository for Red Hat Enterprise Linux Atomic Host or any product that you want to use:

```
# hammer repository-set enable \
--product "Red Hat Enterprise Linux Atomic Host" \
--name "Red Hat Enterprise Linux Atomic Host (Trees)" \
--organization "My_Organization"
```

3. Locate and synchronize the repository for the product:

```
# hammer repository list \
--product "Red Hat Enterprise Linux Atomic Host" \
--organization "My_Organization"
# hammer repository synchronize \
--name "Red Hat Enterprise Linux Atomic Host Trees" \
--product "Red Hat Enterprise Linux Atomic Host" \
--organization "My_Organization"
```

12.2. IMPORTING CUSTOM OSTREE CONTENT

In addition to importing OSTree content from Red Hat CDN, you can also import content from other sources. This requires a published HTTP location for the OSTree to import.

Procedure

To import custom OSTree content, complete the following steps:

1. Satellite Web UI で **コンテンツ** > **製品** に移動して、**製品の作成** をクリックします。
2. In the **Name** field, enter a name for your OSTree content. This automatically populates the **Label** field.
3. Optional: In the **GPG Key** field, enter a GPG Key for the entire product.
4. **同期プラン** メニューから製品に関連付ける同期プランを選択します。
5. In the **Description** field, enter a description of the product and click **Save**.
6. When the product creation completes, click **Create Repository**.
7. In the **Name** field, enter a name for the repository. This automatically populates the **Label** field.
8. From the **Type** list, select **ostree**.

9. In the **URL** field, enter the URL of the registry to use as a source. For example <http://www.example.com/rpm-ostree/>.
10. From the **Upstream Sync Policy** menu, select one of the following policies to synchronize OSTree branches for this repository:
 - **最新のみ**: 最新の OSTree ブランチのみを同期します。
 - **すべての履歴**: すべての OSTree ブランチを同期します。
 - **カスタム**: 指定した数の OSTree ブランチを同期します。下のフィールドに必要な数を入力します。
11. **保存**をクリックします。
12. When the repository creation completes, select the new repository and click **Sync Now** to start the synchronization process.

To view the Synchronization Status:

- In the Satellite web UI, navigate to **Content > Sync Status** and expand the entry that you want to view.

For CLI Users

1. Create the custom **OSTree Content** product:

```
# hammer product create \
--name "Custom OSTree Content" \
--sync-plan "Example_Plan" \
--description "OSTree Content" \
--organization "My_Organization"
```

2. Create the repository for the OSTree:

```
# hammer repository create \
--name "Custom OSTree" \
--content-type "ostree" \
--url "http://www.example.com/rpm-ostree/" \
--product "OSTree Content" \
--organization "My_Organization"
```

3. Synchronize the repository:

```
# hammer repository synchronize \
--name "Custom OSTree" \
--product "OSTree Content" \
--organization "My_Organization"
```

12.3. MANAGING OSTREE CONTENT WITH CONTENT VIEWS

Use Content Views to manage OSTree branches across the application life cycle. This process uses the same publication and promotion method that RPMs and Puppet modules use.

Procedure

To create a content view for your OSTree and add a repository, complete the following steps:

1. In the Satellite web UI, navigate to **Content** > **Content Views** and click **Create New View**.
2. In the **Name** field, enter a plain text name for the view. This automatically populates the **Label** field.
3. In the **Description** field, enter a description of the OSTree Content View.
4. If you want to use a Composite Content View, select the **Composite View** check box.
5. Click **Save** to complete.
6. Navigate to the **OSTree Content** tab, then click **Add**.
7. Select the OSTree repository for that you want to use. Click **Add Repository** to add the OSTree content from this repository to the Content View.
8. Navigate to **Versions** and click **Publish New Version**.
9. In the **Description** field, enter a description for the version, and click **Save**.

You can also click **Promote** to promote this Content View across environments in the application life cycle.

For CLI Users

1. Obtain a list of repository IDs:

```
# hammer repository list --organization "_My_Organization_"
```

2. Create the Content View and add the repository:

```
# hammer content-view create \  
--name "OSTree" \  
--description "OSTree for Red Hat Enterprise Linux Atomic Host" \  
--repository-ids 5 \  
--organization "My_Organization"
```

3. Publish the view:

```
# hammer content-view publish \  
--name "OSTree" \  
--description "Example Content View for the OSTree" \  
--organization "My_Organization"
```

第13章 MANAGING ISO IMAGES

You can use Red Hat Satellite 6 to store ISO images, either from Red Hat's Content Delivery Network or other sources. You can also upload other files, such as virtual machine images, and publish them in repositories.

13.1. IMPORTING ISO IMAGES FROM RED HAT

The Red Hat Content Delivery Network provides ISO images for certain products. The procedure for importing this content is similar to the procedure for enabling repositories for RPM content.

Procedure

To import Red Hat ISO images, complete the following steps:

1. In the Satellite web UI, navigate to **Content** > **Red Hat Repositories**.
2. In the **Search** field, enter an image name, for example, **Red Hat Enterprise Linux 7 Server (ISOs)**.
3. In the Available Repositories window, expand **Red Hat Enterprise Linux 7 Server (ISOs)**
4. For the **x86_64 7.2** entry, click the **Enable** icon to enable the repositories for the image.
5. Navigate to **Content** > **Products** and click **Red Hat Enterprise Linux Server**.
6. Click the **Repositories** tab of the Red Hat Enterprise Linux Server window, and click **Red Hat Enterprise Linux 7 Server ISOs x86_64 7.2**.
7. In the upper right of the Red Hat Enterprise Linux 7 Server ISOs x86_64 7.2 window, click **Select Action** and select **Sync Now**.

To view the Synchronization Status

- In the web UI, navigate to **Content** > **Sync Status** and expand **Red Hat Enterprise Linux Server**.

For CLI Users

1. Locate the Red Hat Enterprise Linux Server product for **file** repositories:

```
# hammer repository-set list \  
--product "Red Hat Enterprise Linux Server" \  
--organization "My_Organization" | grep "file"
```

2. Enable the **file** repository for Red Hat Enterprise Linux 7.2 Server ISO:

```
# hammer repository-set enable \  
--product "Red Hat Enterprise Linux Server" \  
--name "Red Hat Enterprise Linux 7 Server (ISOs)" \  
--releasever 7.2 \  
--basearch x86_64 \  
--organization "My_Organization"
```

3. Locate and synchronize the repository in the product:

```
# hammer repository list \
--product "Red Hat Enterprise Linux Server" \
--organization "My_Organization"
# hammer repository synchronize \
--name "Red Hat Enterprise Linux 7 Server ISOs x86_64 7.2" \
--product "Red Hat Enterprise Linux Server" \
--organization "My_Organization"
```

13.2. IMPORTING INDIVIDUAL ISO IMAGES AND FILES

以下の手順を使用して、ISO コンテンツとその他のファイルを Satellite Server に手動でインポートします。カスタムファイルをインポートするには、Web UI または Hammer CLI を使用して、以下の手順を実行してください。ただし、アップロードするファイルのサイズが 15 MB よりも大きい場合は、Hammer CLI を使用してリポジトリにアップロードする必要があります。

1. Create a custom product.
2. Add a repository for files to the product.
3. Upload a file to the repository.

Procedure

To import custom ISO images, complete the following steps:

1. In the Satellite web UI, navigate to **Content > Products**, and in the Products window, click **New Product**.
2. In the **Name** field, enter a name to identify the product. This name populates the **Label** field.
3. In the **GPG Key** field, enter a GPG Key for the product.
4. From the **Sync Plan** list, select a synchronization plan for the product.
5. **説明**フィールドには、製品の説明を入力します。
6. **保存**をクリックします。
7. In the Products window, click the new product and then click **Create Repository**.
8. In the **Name** field, enter a name for the repository. This automatically populates the **Label** field.
9. From the **Type** list, select **file**.
10. In the **Upstream URL** field, enter the URL of the registry to use as a source. Add a corresponding user name and password in the **Upstream Username** and **Upstream Password** fields.
11. **保存**をクリックします。
12. Click the new repository.
13. Navigate to the **Upload File** and click **Browse**.
14. Select the **.iso** file and click **Upload**.

For CLI Users

1. Create the custom product:

```
# hammer product create \  
--name "My_ISOs" \  
--sync-plan "Example Plan" \  
--description "My_Product" \  
--organization "My_Organization"
```

2. Create the repository:

```
# hammer repository create \  
--name "My_ISOs" \  
--content-type "file" \  
--product "My_Product" \  
--organization "My_Organization"
```

3. Upload the ISO file to the repository:

```
# hammer repository upload-content \  
--path ~/bootdisk.iso \  
--name "My_ISOs" \  
--organization "My_Organization"
```

第14章 MANAGING CUSTOM FILE TYPE CONTENT

In Satellite, you might require methods of managing and distributing SSH keys and source code files or larger files such as virtual machine images and ISO files. To achieve this, custom products in Red Hat Satellite include repositories for custom file types. This provides a generic method to incorporate arbitrary files in a product.

リポジトリにファイルをアップロードし、アップストリームの Satellite Server からファイルを同期できます。ファイルをカスタムのファイルタイプリポジトリに追加すると、特定のバージョンをコンテンツビューに追加して、さまざまな Capsule Server でファイルのリポジトリを利用可能にするなど、通常の Satellite 管理機能を使用できます。クライアントは、**curl -O** で、HTTP または HTTPS からファイルをダウンロードする必要があります。

Satellite Server のファイルタイプリポジトリはカスタム製品に対してのみ作成できますが、ファイルタイプリポジトリは柔軟に作成できます。Satellite がインストールされているシステム、またはリモートの HTTP サーバーのディレクトリに個別のファイルタイプリポジトリを作成して、そのディレクトリのコンテンツを同期できます。この方法は、Satellite リポジトリに追加するファイルが複数ある場合に便利です。

14.1. CREATING A CUSTOM FILE TYPE REPOSITORY IN RED HAT SATELLITE

The procedure for creating a custom file type repository is the same as the procedure for creating any custom content, except that when you create the repository, you select the **file** type. You must create a product and then add a custom repository.

Procedure

To create a custom product, complete the following procedure:

1. Satellite Web UI で、**コンテンツ > 製品** に移動し、**製品の作成** をクリックして、以下の詳細情報を入力します。
2. **名前** フィールドで、製品の名前を入力します。Red Hat Satellite 6 では、**名前** に入力した内容に基づいて **ラベル** フィールドに自動的に入力されます。
3. オプション: **GPG キー** の一覧から、製品の GPG キーを選択します。
4. オプション: **同期プラン** リストから製品の同期プランを選択します。
5. **説明** フィールドに、製品の説明を入力し、**保存** をクリックします。

To create a repository for your custom product, complete the following procedure:

1. In the Products window, select the name of a product that you want to create a repository for.
2. Click the **Repositories** tab, and then click **New Repository**.
3. **名前** フィールドに、リポジトリの名前を入力します。Red Hat Satellite 6 では、名前に基づいて、**ラベル** フィールドに値が自動的に入力されます。
4. From the **Type** list, select **file**.
5. **アップストリーム URL** フィールドに、ソースとして使用するアップストリームリポジトリの URL を入力します。

6. アップストリームのリポジトリの SSL 証明書が信頼できる認証機関 (CA) によって署名されていることを確認する場合、**SSL の検証** チェックボックスを選択します。
7. **アップストリームのユーザー名** フィールドに、認証に必要な場合にアップストリームリポジトリのユーザー名を入力します。リポジトリに認証が必要ない場合はこのフィールドを空にします。
8. **アップストリームのパスワード** フィールドに、アップストリームリポジトリのパスワードを入力します。リポジトリに認証が必要ない場合はこのフィールドを空にします。
9. **保存** をクリックします。

For CLI Users

1. Create a Custom Product

```
# hammer product create \
--name "My File Product" \
--sync-plan "Example Plan" \
--description "My files" \
--organization "My_Organization"
```

表14.1 Optional Parameters for the `hammer product create` Command

Option	Description
<code>--gpg-key gpg_key_name</code>	Key name to search by
<code>--gpg-key-id gpg_key_id</code>	GPG key numeric identifier
<code>--sync-plan sync_plan_name</code>	Sync plan name to search by
<code>--sync-plan-id sync_plan_id</code>	Sync plan numeric identifier

2. Create a File Type Repository

```
# hammer repository create \
--name "My Files" \
--content-type "file" \
--product "My File Product" \
--organization "My_Organization"
```

表14.2 Optional Parameters for the `hammer repository create` Command

Option	Description
<code>--checksum-type sha_version</code>	Repository checksum, currently 'sha1' & 'sha256' are supported
<code>--download-policy policy_name</code>	Download policy for yum repos (either 'immediate', 'on_demand', or 'background').

Option	Description
--gpg-key gpg_key_name	Key name to search by
--gpg-key-id gpg_key_id	GPG key numeric identifier
--mirror-on-sync boolean	Must this repo be mirrored from the source, and stale RPMs removed, when synced? Set to true or false , yes or no , 1 or 0 .
--publish-via-http boolean	Must this also be published using HTTP? Set to true or false , yes or no , 1 or 0 .
--upstream-username repository_username	Upstream repository user, if required for authentication
--upstream-password repository_password	Password for the upstream repository user
--url source_repo_url	URL of the Source repository
--verify-ssl-on-sync boolean	URL の SSL 証明書が信頼できる CA によって署名されているのを Katello が確認する必要がありますか? true または false , yes または no , もしくは 1 または 0 に設定します。

14.2. CREATING A CUSTOM FILE TYPE REPOSITORY IN A LOCAL DIRECTORY

You can create a custom file type repository, from a directory of files, on the base system where Satellite is installed using the **pulp-manifest** command. You can then synchronize the files into Satellite Server. When you add files to a file type repository, you can work with the files as with any other repository.

以下の手順を使用して、Satellite がインストールされているベースシステムのディレクトリーにリポジトリーを設定します。リモートサーバーのディレクトリーにファイルタイプリポジトリーを作成するには、[「Creating a Remote File Type Repository」](#) を参照してください。

Procedure

To create a file type repository in a local directory, complete the following procedure:

1. Ensure the Server and Satellite Tools repositories are enabled:

```
# subscription-manager repos --enable=rhel-7-server-rpms \
--enable=rhel-7-server-satellite-tools-6.5-rpms
```

2. Install the Pulp Manifest package:

```
# yum install python-pulp-manifest
```

3. Create a directory that you want to use as the file type repository in the HTTP server's public folder:

```
# mkdir my_file_repo
```

4. Add files to the directory or create a test file:

```
# touch my_file_repo/test.txt
```

5. Enter the Pulp Manifest command to create the manifest:

```
# pulp-manifest my_file_repo
```

6. Verify the manifest was created:

```
# ls my_file_repo
PULP_MANIFEST test.txt
```

Importing Files from a File Type Repository

To import files from a file type repository in a local directory, complete the following procedure:

1. Ensure a custom product exists in Satellite Server.
2. Satellite Web UI で、**コンテンツ** > **製品**に移動します。
3. Select the name of a product.
4. Click the **Repositories** tab and select **New Repository**.
5. **名前** フィールドに、リポジトリの名前を入力します。Red Hat Satellite 6 では、**名前** に入力した内容をもとに、このフィールドに値が自動的に入力されます。
6. **タイプ** リストから、リポジトリのコンテンツタイプを選択します。
7. **アップストリーム URL** フィールドに、ソースとして使用するリポジトリを使用したローカルディレクトリを入力します (**file:///my_file_repo** の形式)。
8. **SSL の検証** チェックボックスを選択してリポジトリの SSL 証明書をチェックするか、**SSL の検証** チェックボックスの選択を解除します。
9. オプション: **アップストリームのユーザー名** フィールドに、必要なアップストリームユーザー名を入力します。
10. オプション: **アップストリームのパスワード** フィールドに、アップストリームユーザー名のパスワードを入力します。
11. Select **Save** to save this repository entry.

Updating a File Type Repository

To update the file type repository, complete the following steps:

1. Satellite Web UI で、**コンテンツ** > **製品**に移動します。
2. Select the name of a product.

3. Select the name of the repository you want to update.
4. From the **Select Action** menu, select **Sync Now**.
5. Visit the URL where the repository is published to see the files.

14.3. CREATING A REMOTE FILE TYPE REPOSITORY

You can create a custom file type repository from a directory of files that is external to Satellite Server using the **pulp-manifest** command. You can then synchronize the files into Satellite Server over HTTP or HTTPS. When you add files to a file type repository, you can work with the files as with any other repository.

以下の手順を使用して、リモートサーバーのディレクトリーにリポジトリーを設定します。Satellite Server がインストールされているベースシステムのディレクトリーにファイルタイプリポジトリーを作成するには、[「Creating a Custom File Type Repository in a Local Directory」](#) を参照してください。

Prerequisites

Before you create a remote file type repository, ensure the following conditions exist:

- You have a Red Hat Enterprise Linux 7 server registered to your Satellite or the Red Hat CDN.
- Your server has an entitlement to the Red Hat Enterprise Linux Server and Satellite Tools repositories.
- HTTP サーバーがインストールされている。web サーバーの設定方法は『[システム管理者ガイド](#)』における Red Hat Enterprise Linux 7 での [「Apache HTTP Server」](#) を参照してください。

Procedure

To create a file type repository in a remote directory, complete the following procedure:

1. On your remote server, ensure that the Server and Satellite Tools repositories are enabled:

```
# subscription-manager repos --enable=rhel-7-server-rpms \  
--enable=rhel-7-server-satellite-tools-6.5-rpms
```

2. Install the Pulp Manifest package:

```
# yum install python-pulp-manifest
```

3. Create a directory that you want to use as the file type repository in the HTTP server's public folder:

```
# mkdir /var/www/html/pub/my_file_repo
```

4. Add files to the directory or create a test file:

```
# touch /var/www/html/pub/my_file_repo/test.txt
```

5. Enter the Pulp Manifest command to create the manifest:

```
# pulp-manifest /var/www/html/pub/my_file_repo
```

6. Verify the manifest was created:

```
# ls /var/www/html/pub/my_file_repo
PULP_MANIFEST test.txt
```

Importing Files from a Remote a File Type Repository

To import files from a remote file type repository, complete the following procedure:

1. Ensure a custom product exists in Satellite Server, or create a custom product. For more information see [「Creating a Custom File Type Repository in Red Hat Satellite」](#)
2. Satellite Web UI で、**コンテンツ** > **製品**に移動します。
3. Select the name of a product.
4. Click the **Repositories** tab and select **New Repository**.
5. **名前** フィールドに、リポジトリの名前を入力します。Red Hat Satellite 6 では、**名前** に入力した内容をもとに、このフィールドに値が自動的に入力されます。
6. From the **Type** list, select **file**.
7. **アップストリーム URL** フィールドに、ソースとして使用するアップストリームリポジトリの URL を入力します。
8. アップストリームのリポジトリの SSL 証明書が信頼できる認証機関 (CA) によって署名されていることを確認する場合、**SSL の検証** チェックボックスを選択します。
9. **アップストリームのユーザー名** フィールドに、認証に必要な場合にアップストリームリポジトリのユーザー名を入力します。リポジトリに認証が必要ない場合はこのフィールドを空にします。
10. **アップストリームのパスワード** フィールドに、アップストリームリポジトリのパスワードを入力します。リポジトリに認証が必要ない場合はこのフィールドを空にします。
11. **保存** をクリックします。
12. **コンテンツ** > **製品** に移動します。更新するリポジトリが含まれる製品の名前を選択します。
13. In the product's window, select the name of the repository you want to update.
14. From the **Select Action** menu, select **Sync Now**.

Visit the URL where the repository is published to view the files.

14.4. UPLOADING FILES TO A CUSTOM FILE TYPE REPOSITORY IN RED HAT SATELLITE

Procedure

To upload files to a custom file type repository, complete the following steps:

1. Satellite Web UI で、**コンテンツ** > **製品**に移動します。
2. Select a custom product by name.

3. Select a file type repository by name.
4. Click **Browse** to search and select the file you want to upload.
5. Click **Upload** to upload the selected file to Satellite Server.
6. Visit the URL where the repository is published to see the file.

For CLI Users

```
# hammer repository upload-content \
--name "My Files" \
--organization "My_Organization" \
--path example_file
```

The **--path** option can indicate a file, a directory of files, or a glob expression of files. Globs must be escaped by single or double quotes.

14.5. DOWNLOADING FILES TO A HOST FROM A CUSTOM FILE TYPE REPOSITORY IN RED HAT SATELLITE

You can download files to a client over HTTPS using **curl -O**, and optionally over HTTP if the **Publish via HTTP** repository option is selected.

Prerequisites

- You have a custom file type repository. See [「Creating a Custom File Type Repository in Red Hat Satellite」](#) for more information.
- You know the name of the file you want to download to clients from the file type repository.
- To use HTTPS you require the following certificates on the client:
 1. The **katello-server-ca.crt**. For more information, see [Installing the Katello Root CA Certificate](#) in the **Administering Red Hat Satellite** guide.
 2. 組織のデバッグ証明書。詳細については、[「Creating an Organization Debug Certificate」](#)を参照してください。

Procedure

To download files to a host from a custom file type repository, complete the following procedure:

1. Satellite Web UI で、**コンテンツ** > **製品**に移動します。
2. Select a custom product by name.
3. Select a file type repository by name.
4. **HTTP 経由での公開** が有効になっているかどうかを確認します。有効になっていない場合は、**HTTPS** を使用するための証明書が必要です。
5. Copy the URL where the repository is published.

For CLI Users

1. List the file type repositories.

```
# hammer repository list --content-type file
---|-----|-----|-----|----
ID | NAME   | PRODUCT   | CONTENT TYPE | URL
---|-----|-----|-----|----
7 | My Files | My File Product | file        |
---|-----|-----|-----|----
```

2. Display the repository information.

```
# hammer repository info --name "My Files" --product "My File Product" --organization-id 1
```

HTTP が有効になっている場合には、出力は次のようになります。

```
Publish Via HTTP: yes
Published At:      http://satellite.example.com/pulp/isos/uuid/
```

HTTP が有効になっていない場合には、出力は次のようになります。

```
Publish Via HTTP: no
Published At:      https://satellite.example.com/pulp/isos/uuid/
```

3. On the client, enter a command in the appropriate format for HTTP or HTTPS:
For HTTP:

```
# curl -O satellite.example.com/pulp/isos/uuid/my_file
```

For HTTPS:

```
# curl -O --cert ./Default\ Organization-key-cert.pem --cacert katello-server-ca.crt
satellite.example.com/pulp/isos/uuid/my_file
```

第15章 MANAGING CUSTOM PUPPET CONTENT

Satellite で、Puppet モジュールを使用してホストの状態設定を組み込む場合は、Puppet モジュールが状態設定を組み込むのに使用するリポジトリで、カスタム製品を作成します。

15.1. CREATING A CUSTOM PUPPET REPOSITORY

The procedure for creating a custom Puppet module repository is the same as the procedure for creating any custom content, except that when you create the repository, you select the **puppet** type. You must create a product and then add a custom repository.

Procedure

1. Satellite Web UI で、**コンテンツ** > **製品** に移動し、使用する製品をクリックします。
2. Click **Create Repository**.
3. **名前** フィールドで、リポジトリの名前を入力します。Red Hat Satellite 6 では、**名前** に入力した内容に基づいて **ラベル** フィールドに値が自動的に入力されます。
4. **タイプ** の一覧から **puppet** を選択します。
5. In the **URL** field, enter the URL of the external repository to use as a source. You can use a repository source to synchronize your own Puppet modules.
6. **保存** をクリックします。

For CLI Users

1. Enter the following command to create a Puppet module repository:

```
# hammer repository create \  
--name "PostgreSQL Puppet Modules" \  
--content-type "puppet" \  
--product "PostgreSQL" \  
--organization "My_Organization"
```

15.2. MANAGING INDIVIDUAL PUPPET MODULES

RPM コンテンツと Puppet モジュールの両方が含まれるカスタム製品を作成して、カスタム RPM コンテンツを使用するサーバーをインストールして設定する場合には、[「Creating a Custom Puppet Repository」](#)の手順を使用してから、以下の手順に従い、Puppet モジュールをアップロードします。

カスタム RPM のサポート

Red Hat does not support the modules from Puppet Forge. For any issues with these modules, contact the module developer.

Prerequisites

1. From the Puppet Forge website, download the module that you want to use, for example, <https://forge.puppetlabs.com/puppetlabs/postgresql>.
2. In your web browser, click **download latest tar.gz** to save to your local file system.

Procedure

1. Satellite Web UI で、**コンテンツ** > **製品** に移動し、管理する Puppet リポジトリが含まれる製品を選択します。
2. In the repository window, click the new Puppet repository, which displays the details page for that repository.
3. **Puppet モジュールのアップロード** エリアに移動し、**参照** をクリックします。新たにダウンロードして抽出された Puppet モジュールを選択し、**アップロード** をクリックします。

To manage and remove Puppet modules from a product, complete the following steps:

1. In the window for your Puppet Modules repository, navigate to the upper right of the window to the **Content Counts** area. In the **Puppet Modules** row, click the numerical value that is displayed for the Puppet Modules.
2. Puppet モジュールリポジトリウィンドウの **Puppet モジュールの管理** で、管理するモジュールを選択して、**アクションの選択** をクリックし、アクションを実行するか、**Puppet モジュールの削除** を選択します。

For CLI Users

1. Copy the Puppet module to your Satellite Server's file system:

```
$ scp ~/puppet_module.tar.gz root@satellite.example.com:~/.
```

2. Import the Puppet module to the Puppet Modules repository:

```
# hammer repository upload-content \
--path ~/puppet_module.tar.gz \
--name "My Puppet Modules" \
--organization "My_Organization"
```

15.3. SYNCHRONIZING PUPPET REPOSITORIES

In addition to creating a repository of uploaded Puppet modules, Satellite Server can synchronize a complete Puppet module repository. In this example, Satellite Server synchronizes the entire Puppet Forge repository.

カスタム RPM のサポート

Red Hat does not support the modules from Puppet Forge. The modules are used to demonstrate the synchronization process. For any issues with these modules, contact the module developer.

Procedure

1. In the Satellite web UI, navigate to **Content** > **Products** and click **New Product**.
2. **名前** フィールドで、製品の名前を入力します。Red Hat Satellite 6 では、**名前** に入力した内容に基づいて **ラベル** フィールドに自動的に入力されます。
3. オプション: **GPG キー** の一覧から、製品の GPG キーを選択します。
4. オプション: **同期プラン** リストから製品の同期プランを選択します。

5. **説明** フィールドには、製品の説明を入力します。
6. **保存** をクリックします。
7. Click **Create Repository**, which displays a form for a new repository.
8. **名前** フィールドに、リポジトリの名前を入力します。Red Hat Satellite 6 では、**名前** に入力した名前をもとに、このフィールドに値が自動的に入力されます。
9. **タイプ** の一覧から **puppet** を選択します。
10. In the **URL** field, enter <http://forge.puppetlabs.com/>.
11. Click **Save**
12. 新しい Puppet リポジトリを選択し、**今すぐ同期** をクリックして、Puppet Forge から Satellite Server に全モジュールをインポートします。この処理には時間がかかることがあります。

For CLI Users

1. Create the product:

```
# hammer product create \
--name "Puppet Forge" \
--sync-plan "Example Plan" \
--description "All modules from Puppet Forge" \
--organization "My_Organization"
```

2. Create the Puppet Forge repository:

```
# hammer repository create \
--name "Puppet Forge Modules" \
--content-type "puppet" \
--product "Puppet Forge" \
--organization "My_Organization" \
--url http://forge.puppetlabs.com/
```

3. Synchronize the repository:

```
# hammer repository synchronize \
--name "Puppet Forge Modules" \
--product "Puppet Forge" \
--organization "My_Organization"
```

The Puppet Forge repository contains several thousand modules and can take a long time to synchronize.

15.4. SYNCHRONIZING PUPPET MODULES FROM A GIT REPOSITORY

Red Hat Satellite 6 includes a utility called **pulp-puppet-module-builder**, which you can install on other systems from the **pulp-puppet-tools** RPM. This tool checks out a Git repository, builds all the modules, and publishes them in a structure that Satellite 6 can synchronize. One common method is to run the utility on Satellite Server itself, publish to a local directory, and synchronize against that directory. For example:

```
# mkdir /modules
# chmod 755 /modules
# pulp-puppet-module-builder \
--output-dir=/modules \
--url=git@mygitserver.com:mymodules.git \
--branch=develop
```

この例では、Git リポジトリの **develop** ブランチを **git@mygitserver.com:mymodules.git** からチェックアウトし、**/modules** に公開します。このディレクトリーを Satellite Server の新規リポジトリの URL (**file:///modules**) として追加します。

Publishing Puppet Modules on a Remote HTTP Server

リモートの HTTP サーバー上にモジュールを公開する場合でも同じプロセスを実行します。たとえば、Puppet モジュールを公開する標準 Web ホストとして **webserver.example.com** を使用します。

```
# mkdir /var/www/html/modules/
# chmod 755 /var/www/html/modules/
# pulp-puppet-module-builder \
--output-dir=/var/www/html/modules/ \
--url=git@mygitserver.com:mymodules.git \
--branch=develop
```

On Satellite Server, set the repository's URL to <http://webserver.example.com/modules/>.

Synchronizing Puppet Modules from a Git repository using the web UI

Use the following procedure to synchronize Puppet modules from a Git repository.

Procedure

1. カスタム製品を作成し、**リポジトリの作成** をクリックします。
2. **タイプ** の一覧から **puppet** を選択します。
3. In the **URL** field, enter the URL of the external Git repository to use as a source in the following format: **file:///modules**.

For CLI Users

1. Create the Puppet Forge repository:

```
# hammer repository create \
--name "Modules from Git" \
--content-type "puppet" \
--product "MyProduct" \
--organization "My_Organization" \
--url file:///modules
```

付録A USING AN NFS SHARE FOR CONTENT STORAGE

Your environment requires adequate hard disk space to fulfill content storage. In some situations, it is useful to use an NFS share to store this content. This appendix shows how to mount the NFS share on your Satellite Server's content management component.



重要

NFS 共有に、**/var/lib/pulp** すべてをマウントしないでください。**/var/lib/pulp** のファイルシステムは、帯域幅が高く、レイテンシーの低いストレージを使用するので、レイテンシーが高く、帯域幅の低いストレージでは、パフォーマンスが低下する問題が発生する可能性があります。NFS 共有は、**/var/lib/pulp/content** ディレクトリーにのみ使用してください。

1. Create the NFS share. This example uses a share at **nfs.example.com:/satellite/content**. Ensure this share provides the appropriate permissions to Satellite Server and its **apache** user.

2. Shutdown the Satellite services on the Satellite host:

```
# foreman-maintain service stop
```

3. Ensure Satellite Server has the **nfs-utils** package installed:

```
# yum install nfs-utils
```

4. You need to copy the existing contents of **/var/lib/pulp/content** to the NFS share. First, mount the NFS share to a temporary location:

```
# mkdir /mnt/temp
# mount -o rw nfs.example.com:/satellite/content /mnt/temp
```

Copy the existing contents of **/var/lib/pulp/content** to the temporary location:

```
# cp -r /var/lib/pulp/content/* /mnt/temp/.
```

5. Set the permissions for all files on the share to use the **apache** user. This ID of this user is usually 48.
6. Unmount the temporary storage location:

```
# umount /mnt/temp
```

7. Remove the existing contents of **/var/lib/pulp/content**:

```
# rm -rf /var/lib/pulp/content/*
```

8. Edit the **/etc/fstab** file and add the following line:

```
nfs.example.com:/satellite/content /var/lib/pulp/content nfs
rw,hard,intr,context="system_u:object_r:httpd_sys_rw_content_t:s0"
```

This makes the mount persistent across system reboots. Ensure to include the SELinux context.

9. Enable the mount:

```
# mount -a
```

10. Confirm the NFS share mounts to **var/lib/pulp/content**:

```
# df
Filesystem              1K-blocks  Used Available Use% Mounted on
...
nfs.example.com:/satellite/content 309506048 58632800 235128224 20%
/var/lib/pulp/content
...
```

Also confirm that the existing content exists at the mount on **var/lib/pulp/content**:

```
# ls /var/lib/pulp/content
```

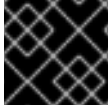
11. Start the Satellite services on the Satellite host:

```
# foreman-maintain service start
```

Satellite Server now uses the NFS share to store content. Run a content synchronization (see [\[Content Synchronization Overview\]](#)) to ensure the NFS share works as expected.

付録B IMPORTING CONTENT ISOS INTO A DISCONNECTED SATELLITE

In high security environments where hosts are required to function in a closed network disconnected from the Internet, Satellite Server can provision systems with the latest security updates, errata, and packages. To accomplish this, download the Content ISOs for Red Hat Satellite from the Red Hat Customer Portal and import them into Satellite Server.



重要

This section is not required if your Satellite Server is connected to the Internet.

Download the product ISO from the Red Hat Customer Portal, as follows:

1. Navigate to Downloads (at the very top of the window) and select Red Hat Satellite.
2. Open the Content ISOs tab. All products in your subscription are listed here.
3. Click the link for the product name, such as Red Hat Enterprise Linux 6 Server (x86_64) to download the ISO.
4. Copy all of Satellite Content ISOs to a directory Satellite can access. This example uses **/root/isos**.
5. Create a local directory that are shared through httpd on the Satellite. This example uses **/var/www/html/pub/sat-import/**.

```
# mkdir -p /var/www/html/pub/sat-import/
```

6. Mount and recursively copy the contents of the first ISO to the local directory:

```
# mkdir /mnt/iso
# mount -o loop /root/isos/first_iso /mnt/iso
# cp -ruv /mnt/iso/* /var/www/html/pub/sat-import/
# umount /mnt/iso
# rmdir /mnt/iso
```

7. Repeat the above step for each ISO until you have copied all the data from the Content ISOs into **/var/www/html/pub/sat-import/**.
8. Ensure the SELinux contexts for the directory are correct:

```
# restorecon -rv /var/www/html/pub/sat-import/
```

9. Satellite Server now contains the content from the Content ISOs. However, Satellite Server needs to point to this location as the CDN URL. In the Satellite Web UI, navigate to **Content > Red Hat Subscriptions** .
10. Click **Manage Manifest**.
11. On the Subscription Manifest information screen, select the **Actions** tab.
12. Navigate to Red Hat Provider Details. Click the edit icon on the **Red Hat CDN URL** and change the URL to the Satellite host name with the newly created directory, for example:

<http://server.example.com/pub/sat-import/>

13. Click **Save** and then upload your manifest using [「Importing a Subscription Manifest into Satellite Server」](#) .

Satellite is now acting as its own CDN with the files located in <http://server.example.com/pub/sat-import/>. This is not a requirement. The CDN can be hosted on a different machine inside the same disconnected network as long as it is accessible to Satellite Server using HTTP.

If your environment changes from disconnected to connected, you can reconfigure a disconnected Satellite to pull content directly from Red Hat Customer Portal:

1. In the Satellite Web UI, navigate to **Content > Red Hat Subscriptions**
2. Click **Manage Manifest**.
3. On the Subscription Manifest information screen, select the **Actions** tab.
4. Navigate to Red Hat Provider Details. Click the edit icon on the **Red Hat CDN URL** and change the URL to the Red Hat CDN URL:
<https://cdn.redhat.com>
5. Click **Save**

Satellite Server pulls content directly from Red Hat Customer Portal on the next synchronization.

付録C IMPORTING CONTENT ISOS INTO A CONNECTED SATELLITE

Even if Satellite Server can connect directly to the Red Hat Customer Portal, you can perform the initial synchronization from locally mounted content ISOs. When the initial synchronization is completed from the content ISOs, you can switch back to downloading content through the network connection. To accomplish this, download the Content ISOs for Red Hat Satellite from the Red Hat Customer Portal and import them into Satellite Server. For locations with bandwidth limitations, using an **On Demand** or **Background** download policy might be more efficient than downloading and importing Content ISOs.

Note that if you synchronize a Red Hat Enterprise Linux ISO, all minor versions of Red Hat Enterprise Linux also synchronize. You require adequate storage on your Satellite to account for this.



重要

This section is not required if your Satellite Server is connected to the Internet.

この例では、Red Hat Enterprise Linux 7 リポジトリのコンテンツ ISO からの初回同期の実施方法を説明します。

Downloading the content ISOs from the Red Hat Customer Portal

1. In your browser, navigate to [Red Hat Customer Portal](#) and log in.
2. Click **DOWNLOADS**.
3. Select **Red Hat Satellite**.
4. **コンテンツ ISO タブ**を選択します。
5. Search for the section required, in this example **Red Hat Enterprise Linux 7**.
6. Click the link for the product name to reveal the ISO files.
7. Using your browser, download the required ISOs to a location accessible by your browser. For example, to your workstation's **Downloads** directory.

Importing the Content ISOs

1. Satellite Server に接続している端末で、必要なすべての Satellite コンテンツ ISO を一時的に保存するディレクトリを作成します。この例では、**/tmp/isos/rhel7** を使用します。

```
# mkdir -p /tmp/isos/rhel7
```

2. On your workstation, copy the ISO files to Satellite Server:

```
$ scp ~/Downloads/iso_file root@satellite.example.com:/tmp/isos/rhel7
```

3. On Satellite Server, create a directory to serve as a mount point for the ISOs:

```
# mkdir /mnt/iso
```

4. Create a working directory to hold the contents of all the ISOs:

```
# mkdir /mnt/rhel7
```

5. Mount and recursively copy the contents of the first ISO to the working directory:

```
# mount -o loop /tmp/isos/iso_file /mnt/iso
# cp -ruv /mnt/iso/* /mnt/rhel7/
# umount /mnt/iso
```

6. Repeat the above step for each ISO until you have copied all the data from the Content ISOs into **/mnt/rhel7**.
7. If required, remove the empty directory used as the mount point:

```
# rmdir /mnt/iso
```

8. If required, remove the temporary working directory and its contents to regain the space:

```
# rm -rf /tmp/isos/
```

Performing the Initial Synchronization

1. Set the owner and the SELinux context for the directory and its contents to be the same as **/var/lib/pulp**:

```
# chcon -R --reference /var/lib/pulp /mnt/rhel7/
# chown -R apache:apache /mnt/rhel7/
```

2. Create or edit the **/etc/pulp/content/sources/conf.d/local.conf** file. Insert the following text into the file:

```
[rhel-7-server]
enabled: 1
priority: 0
expires: 3d
name: Red Hat Enterprise Linux 7 Server
type: yum
base_url: file:///mnt/rhel7/content/dist/rhel/server/7/7Server/x86_64/os/
```

The **base_url** path might differ in your content ISO. The directory specified in **base_url** must contain the **repodata** directory, otherwise the synchronization fails. To synchronize multiple repositories, create a separate entry for each of them in the configuration file **/etc/pulp/content/sources/conf.d/local.conf**.

3. Satellite Web UI で、コンテンツ > Red Hat リポジトリ に移動して、Red Hat Enterprise Linux 7 Server RPMs x86_64 Server のリポジトリを有効にします。
4. Under **Content > Sync Status** select the repository to be synchronized and click **Synchronize Now**.

Note that there is no indication in the Satellite web UI of which source is being used. In case of problems with a local source, Satellite pulls content through the network. To monitor the process, enter the following command in a terminal (limited to Red Hat Enterprise Linux 7 base systems):


```
# journalctl -f -l SYSLOG_IDENTIFIER=pulp | grep -v worker[^\,\.]heartbeat
```

The above command displays interactive logs. First, Satellite Server connects to the Red Hat Customer Portal to download and process repository metadata. Then, the local repository is loaded. In case of any errors, cancel the synchronization in the Satellite web UI and verify your configuration.

After successful synchronization you can detach the local source by removing its entry from **/etc/pulp/content/sources/conf.d/local.conf**.

付録D SYNCHRONIZING TEMPLATES WITH GIT

Red Hat Satellite 6 enables synchronization of Job Templates, Provisioning Templates, and Partition Table Templates between Satellite Server and a Git repository or a local directory.

This section details the workflow for:

- installing and configuring the TemplateSync plug-in
- performing exporting and importing tasks

D.1. ENABLING THE TEMPLATESYNC PLUG-IN

1. To enable the plug-in on your Satellite Server:

```
# satellite-installer --enable-foreman-plugin-templates
```

2. To verify that the plug-in is installed correctly, ensure **Administer** > **Settings** includes the **TemplateSync** menu.

D.2. CONFIGURING THE TEMPLATESYNC PLUG-IN

Navigate to **Administer** > **Settings** > **TemplateSync** to configure the plug-in. The following table explains the attributes behavior. Note that some attributes are only used on importing or exporting tasks.

表D.1 Synchronizing Templates Plug-in configuration

Parameter	API parameter name	Meaning on importing	Meaning on exporting
Associate	associate Accepted values: always, new, never	Associates templates with OS, Organization, and Location based on metadata.	N/A
Branch	branch	Specifies the default branch in Git repository to read from.	Specifies the default branch in Git repository to write to.
Dirname	dirname	Specifies the subdirectory under the repository to read from.	Specifies the subdirectory under the repository to write to.
Filter	filter	Imports only templates with names that match this regular expression.	Exports only templates with names that match this regular expression.
Force import	force	Imported templates overwrite locked templates with the same name.	N/A

Parameter	API parameter name	Meaning on importing	Meaning on exporting
Metadata export mode	metadata_export_mode Accepted values: refresh, keep, remove	N/A	Defines how metadata is handled when exporting: <ul style="list-style-type: none"> ● Refresh – remove existing metadata from the template content and generate new metadata based on current assignments and attributes. ● Keep – retain the existing metadata. ● Remove – export template without metadata. Useful if you want to add metadata manually.
Negate	negate Accepted values: true, false	Imports templates ignoring the filter attribute.	Exports templates ignoring the filter attribute.
Prefix	prefix	Adds specified string to the beginning of the template if the template name does not start with the prefix already.	N/A
Repo	repo	Defines the path to the repository to synchronize from.	Defines the path to a repository to export to.
Verbosity	verbose Accepted values: true, false	Enables writing verbose messages to the logs for this action.	N/A

D.3. IMPORTING AND EXPORTING TEMPLATES

Importing and exporting tasks are available through a series of API calls. API calls use the role-based access control system, which enables the tasks to be executed as any user. The TemplateSync plug-in allows synchronizing with a Git repository or a local directory.

Prerequisites

For imported templates to appear in the Satellite web UI, each template must contain the location and organization that the template belongs to. This applies to all template types. Before you import a template, ensure that you add the following section to the template:

```
<%#
kind: provision
name: My Kickstart File
oses:
- RedHat 7
- RedHat 6
locations:
- First Location
- Second Location
organizations:
- Default Organization
- Extra Organization
%>
```

Hammer を使用したテンプレートのインポートやエクスポートも可能です。詳細は、『[Hammer CLI ガイド](#)』の「[プロビジョニングテンプレート](#)」を参照してください。

D.3.1. Synchronizing Templates with a Git repository

1. Configure a Git server that uses SSH authorization, for example gitosis, gitolite, or git daemon.
2. Configure the TemplateSync plug-in settings on a **TemplateSync** tab.
 - a. Change the **Branch** setting to match the target branch on a Git server.
 - b. Change the **Repo** setting to match the Git repository. For example, for the repository located in **git@git.example.com/templates.git** set the setting into **ssh://git@git.example.com/templates.git**.
3. Accept Git SSH host key as the foreman user:

```
# sudo -u foreman ssh git.example.com
```

You can see the **Permission denied, please try again.** message in the output, which is expected, because the SSH connection cannot succeed yet.

4. Create an SSH key pair if you do not already have it. Do not specify any passphrase.

```
# sudo -u foreman ssh-keygen
```

5. Configure your Git server with the public key from your Satellite, which resides in **/usr/share/foreman/.ssh/id_rsa.pub**.
6. Export templates from your Satellite Server to the Git repository specified in the **TemplateSync** menu:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" \
-u login:password \
-k https://satellite.example.com/api/v2/templates/export \
-X POST

{"message":"Success"}
```

7. Import templates to Satellite Server after their content was changed:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" \
-u login:password \
-k https://satellite.example.com/api/v2/templates/import \
-X POST

{"message":"Success"}
```

Note that templates provided by Satellite are locked and you cannot import them by default. To overwrite this behavior, change the **Force import** setting in the **TemplateSync** menu to **yes** or add the **force** parameter **-d '{ "force": "true" }'** to the import command.

D.3.2. Synchronizing templates with a local directory

Synchronizing templates with a local directory is useful if you have configured any revision control system repository in the local directory. That way, you can edit templates and track the history of edits in the directory. You can also synchronize changes to Satellite Server after editing the templates.

1. Create the directory where templates are stored and apply appropriate permissions and SELinux context:

```
# mkdir -p /usr/share/templates_dir/
# chown foreman /usr/share/templates_dir/
# chcon -t httpd_sys_rw_content_t /usr/share/templates_dir/ -R
```

2. Change the **Repo** setting on the **TemplateSync** tab to match the export directory **/usr/share/templates_dir/**.
3. Export templates from your Satellite Server to a local directory:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" \
-u login:password \
-k https://satellite.example.com/api/v2/templates/export \
-X POST \

{"message":"Success"}
```

4. Import templates to Satellite Server after their content was changed:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" \
-u login:password \
-k https://satellite.example.com/api/v2/templates/import \
```

```
-X POST
```

```
{"message":"Success"}
```

Note that templates provided by Satellite are locked and you cannot import them by default. To overwrite this behavior, change the **Force import** setting in the **TemplateSync** menu to **yes** or add the **force** parameter `-d '{"force": "true"}'` to the import command.



注記

You can override default API settings by specifying them in the request with the `-d` parameter. The following example exports templates to the `git.example.com/templates` repository:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" \
-u login:password \
-k https://satellite.example.com/api/v2/templates/export \
-X POST \
-d '{"repo":"git.example.com/templates"}'
```

D.4. ADVANCED GIT CONFIGURATION

You can perform additional Git configuration for the TemplateSync plug-in using the command line or editing the `.gitconfig` file.

Accepting a self-signed Git certificate

If you are using a self-signed certificate authentication on your Git server, validate the certificate with the `git config http.sslCAPath` command.

For example, the following command verifies a self-signed certificate stored in `/cert/cert.pem`:

```
# sudo -u foreman git config --global http.sslCAPath cert/cert.pem
```

For a complete list of advanced options, see the `git-config` manual page.

D.5. UNINSTALLING THE PLUG-IN

To avoid errors after uninstallation:

1. Disable the plug-in using the Satellite installer:

```
# satellite-installer --no-enable-foreman-plugin-templates
```

2. Clean custom data of the plug-in. The command does not affect any templates that you created.

```
# foreman-rake templates:cleanup
```

3. Uninstall the plug-in:

```
# yum remove tfm-rubygem-foreman_templates
```

