



# Red Hat Satellite 6.4

## クイックスタートガイド

Red Hat Satellite Server からの物理および仮想ホストのインストール、設定、プロ  
ビジョニング



# Red Hat Satellite 6.4 クイックスタートガイド

---

Red Hat Satellite Server からの物理および仮想ホストのインストール、設定、プロビジョニング

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

## 法律上の通知

Copyright © 2018 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

## 概要

Red Hat Satellite のクイックスタートガイドは、Red Hat Satellite Server の迅速なインストールや設定、さらに物理および仮想ホストのプロビジョニングに役立つように作成されています。本書は、ネットワークに関する知識とスキルを十分に持つ Satellite 管理者を主な対象としています。

## 目次

<b>第1章 本書の対象者</b> .....	<b>3</b>
<b>第2章 SATELLITE SERVER のインストール</b> .....	<b>4</b>
2.1. SATELLITE SERVER のインストールの準備	4
2.2. RED HAT SUBSCRIPTION MANAGEMENT への登録	4
2.3. SATELLITE サブスクリプションの確認、およびホストへのサブスクリプションの割り当て	4
2.4. リポジトリの設定	6
2.5. SATELLITE SERVER のインストール	6
2.6. カスタマーポータルでサブスクリプションの割り当ての作成	6
2.6.1. 割り当てへのサブスクリプションの追加	7
2.7. カスタマーポータルからのサブスクリプションマニフェストのエクスポート	7
2.8. マニフェストの SATELLITE SERVER へのアップロード	7
<b>第3章 オブジェクトのデフォルト組織およびロケーションへの関連付け</b> .....	<b>9</b>
3.1. 新規ドメインの作成	9
3.2. ドメインとデフォルトの組織との関連付け	9
3.3. サブネットの設定	10
<b>第4章 サブスクリプションのインポートおよびコンテンツの同期</b> .....	<b>13</b>
4.1. RED HAT リポジトリの有効化	13
4.2. カスタム製品の作成	14
4.3. コンテンツの同期	15
<b>第5章 コンテンツの管理およびプロモーション</b> .....	<b>17</b>
5.1. アプリケーションライフサイクル環境の作成	17
5.2. シンプルなコンテンツビューの作成	17
5.3. RED HAT ENTERPRISE LINUX リポジトリの追加	18
5.4. コンテンツビューへの PUPPET モジュールの追加	18
5.5. コンテンツビューの公開	19
5.6. アクティベーションキーの作成と編集	20
<b>第6章 システムのパッチ修正</b> .....	<b>22</b>
6.1. 既存のホストの登録	22
6.1.1. 既存の Red Hat Enterprise Linux ホストの登録	22
6.1.2. Atomic Host の登録	22
6.2. KATELLO エージェントを使用したシステムのパッチ修正	23
6.2.1. Katello エージェントのインストール	23
6.2.2. エラータのコンテンツホストへの適用	23
6.3. リモートの実行を使用したシステムのパッチ修正	24
6.3.1. ホストでのリモート実行の有効化	24
6.3.2. ホストへのエラータのインストール	24
<b>第7章 プロビジョニング設定の最終設定</b> .....	<b>25</b>
7.1. カスタムのプロビジョニングテンプレートの作成	25
7.2. ホストグループの作成	26
<b>第8章 PXE を使わない検出イメージを使用したホストのプロビジョニング</b> .....	<b>28</b>



## 第1章 本書の対象者

クイックスタートガイドでは、お使いの環境の前提条件をもとに Satellite Server をインストールして設定する方法を紹介します。ここでは、必要なトポロジを設定し、ネットワークでホストをプロビジョニングするのに必要な設定情報をすべて提供します。

お使いの環境が以下の前提条件に当てはまる場合は、本書をご利用ください。

- Red Hat Satellite を新規インストールする場合
- Satellite Server のデプロイメントが1つの場合
- Red Hat コンテンツ配信ネットワークに接続して、パッケージや更新を取得する場合
- Satellite サーバーに接続する外部の DNS サーバーがすでにある場合
- お使いの環境では DHCP を使用せず、外部の DHCP サービスを使用する場合
- パッチ修正用の Satellite に登録するホストが存在する場合
- PXE を使用しない検出を使用して、ホストをプロビジョニングする場合
- キックスタートベースのデプロイメントを使用する場合
- デプロイメントで、シンプルなコンテンツビューを使用する場合
- IPv4 を使用している場合

本書のタスクではサンプルの情報を使用する場合があります。その場合は、お使いの環境に合わせて値を変更してください。

お使いの環境が上記の前提条件に該当しない場合には、以下のドキュメントを参照して、Satellite Server をインストールしてください。

- [オンラインネットワークからの Satellite Server のインストール](#)
- [オフラインネットワークからの Satellite Server のインストール](#)
- [「10 Steps to Build an SOE: How Red Hat Satellite 6 Supports Setting Up a Standard Operating Environment」\(英語版\)](#)
- [Red Hat Satellite 6.4 プロビジョニングガイド](#)

## 第2章 SATELLITE SERVER のインストール

### 2.1. SATELLITE SERVER のインストールの準備

Satellite Server をインストールする前に、前提条件をすべて満たしていることを確認してください。

#### 前提条件

- Satellite Server をインストールおよび設定する前に Red Hat Enterprise Linux 7 ホストを作成する必要があります。Red Hat Enterprise Linux バージョン 7.5 以降がサポートされます。Red Hat Enterprise Linux 7 のインストール方法は『[Red Hat Enterprise Linux 7 インストールガイド](#)』を参照してください。
- インストール要件 (たとえばストレージ要件) を満たしており、ネットワークポートとファイアウォールが開いていて設定されている必要があります。詳しい情報は『[オンラインネットワークからの Satellite Server のインストール](#)』の「[インストールのための環境準備](#)」を参照してください。

### 2.2. RED HAT SUBSCRIPTION MANAGEMENT への登録

Red Hat Subscription Management にホストを登録すると、ユーザーが利用可能なサブスクリプションにホストを登録して、サブスクリプションのコンテンツを使用できるようになります。これには、Red Hat Enterprise Linux、Red Hat Software Collection (RHSC)、Red Hat Satellite などのコンテンツが含まれます。

1. ユーザー名およびパスワードを使用して Satellite Server を登録します。

```
# subscription-manager register
Username:
Password:
The system has been registered with ID: 541084ff2-44cab-4eb1-9fa1-
7683431bcf9a
```

### 2.3. SATELLITE サブスクリプションの確認、およびホストへのサブスクリプションの割り当て

ホストを登録したら、Satellite サブスクリプションプール ID を確認する必要があります。このプール ID を使用して、必要なサブスクリプションをホストに割り当てることができます。Satellite サブスクリプションは、Red Hat Enterprise Linux、Red Hat Software Collections (RHSC)、Red Hat Satellite、および Satellite コンテンツへのアクセスを提供します。必要なサブスクリプションは Satellite サブスクリプションのみとなります。

1. Satellite サブスクリプションを確認するには、以下のコマンドを入力します。

```
# subscription-manager list --all --available
```

この出力では、以下のような内容が表示されます。

```
Subscription Name: Red Hat Satellite
Provides:           Red Hat Satellite 6
                   Red Hat Enterprise Linux Server
                   Red Hat Satellite
                   Red Hat Enterprise Linux Load Balancer (for RHEL
```



```

Server)
SKU:                MCT0370
Pool ID:            8a85f9874152663c0541943739717d11
Available:         3
Suggested:         1
Service Level:     Premium
Service Type:      L1-L3
Multi-Entitlement: No
Ends:              10/07/2014
System Type:       Physical

```

- あとで Satellite ホストに割り当てられるように、プール ID をメモします。実際に使用するプール ID は、この例で使用されているものとは異なります。
- Satellite Server にサブスクリプションを割り当てるには、お使いの環境のプール ID を使用して、以下のコマンドを実行します。

```
# subscription-manager attach --
pool=8a85f9874152663c0541943739717d11
```

この出力では、以下のような内容が表示されます。

```
Successfully attached a subscription for: Red Hat Satellite
```

- サブスクリプションが正しく割り当てられたことを確認するには、以下のコマンドを実行します。

```
# subscription-manager list --consumed
```

この出力では、以下のような内容が表示されます。

```

+-----+
  Consumed Subscriptions
+-----+
Subscription Name: Red Hat Satellite
Provides:          Red Hat Satellite
                  Red Hat Enterprise Linux Server
                  Red Hat Software Collections (for RHEL Server)
                  Red Hat Satellite
                  Red Hat Satellite 6
                  Red Hat Software Collections (for RHEL Server)
                  Red Hat Satellite Capsule
                  Red Hat Enterprise Linux Load Balancer (for RHEL
Server)
                  Red Hat Satellite with Embedded Oracle
                  Red Hat Satellite Capsule
                  Red Hat Enterprise Linux High Availability (for
RHEL Server)
SKU:                MCT0370
Contract:          10293569
Account:           5361051
Serial:            1653856191250699363
Pool ID:           8a85f9874152663c0541943739717d11
Active:            True
Quantity Used:     1

```

```
Service Level:      Premium
Service Type:       L1-L3
Status Details:
Starts:             10/08/2013
Ends:               10/07/2014
System Type:        Physical
```

## 2.4. リポジトリの設定

1. 既存のリポジトリをすべて無効にします。

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

2. Red Hat Satellite、Red Hat Enterprise Linux、Red Hat Software Collections の各リポジトリを有効にします。

```
# subscription-manager repos --enable=rhel-7-server-rpms \
--enable=rhel-server-rhsc1-7-rpms \
--enable=rhel-7-server-satellite-6.4-rpms \
--enable=rhel-7-server-satellite-maintenance-6-rpms \
--enable=rhel-7-server-ansible-2.6-rpms
```

3. メタデータを消去します。

```
# yum clean all
```

4. リポジトリが有効になっていることを確認します。

```
# yum repolist enabled
```

## 2.5. SATELLITE SERVER のインストール

1. Satellite Server パッケージをインストールします。

```
# yum install satellite
```

2. Satellite Server をインストールして、初期設定を実行します。

```
# satellite-installer --scenario satellite \
--foreman-initial-organization "$ORG" \
--foreman-initial-location "$LOCATION" \
--foreman-admin-password redhat \
--foreman-proxy-puppetca true \
--foreman-proxy-tftp true \
--enable-foreman-plugin-discovery
```

## 2.6. カスタマーポータルでサブスクリプションの割り当ての作成

マニフェストを作成するには、最初に割り当てを作成し、その割り当てからマニフェストをエクスポートする必要があります。

サブスクリプション情報は、Red Hat カスタマーポータルでアクセスできます。そこでサブスクリプション割り当てを使用して、Red Hat Satellite サーバーなどのオンプレミス管理アプリケーションで使用するサブスクリプションを割り当てることができます。

1. ブラウザーで **カスタマーポータル** (<https://access.redhat.com/>) を開き、Red Hat アカウントでログインします。
2. カスタマーポータルの左上にある **サブスクリプション** に移動します。
3. **サブスクリプション割り当て** に移動します。
4. **新規サブスクリプションの割り当て** をクリックします。
5. **名前** フィールドに名前を入力します。
6. **タイプ** の一覧から、お使いの Satellite Server に一致するタイプとバージョンを選択します。
7. **作成** をクリックします。

### 2.6.1. 割り当てへのサブスクリプションの追加

以下の手順では、サブスクリプションを割り当てに追加する方法について説明します。

1. **サブスクリプション割り当て** に移動します。
2. 変更するサブスクリプションの名前を選択します。
3. **サブスクリプション** タブをクリックします。
4. **サブスクリプションの追加** をクリックします。
5. Red Hat 製品サブスクリプションの一覧が表示されます。各製品に対する**エンタイトルメントの数量**を入力します。
6. **送信** をクリックして割り当てを完了します。

割り当てにサブスクリプションを追加したら、マニフェストファイルをエクスポートします。

## 2.7. カスタマーポータルからのサブスクリプションマニフェストのエクスポート

1 つ以上のサブスクリプションがあるサブスクリプション割り当てを表示し、以下のいずれかからマニフェストをエクスポートできます。

- **サブスクリプション** セクションの **詳細** タブから **マニフェストのエクスポート** ボタンをクリックします。
- **サブスクリプション** タブから **マニフェストのエクスポート** ボタンをクリックします。

マニフェストをエクスポートすると、カスタマーポータルにより、選択したサブスクリプション証明書がエンコードされ、.zip アーカイブが作成されます。作成した .zip アーカイブはサブスクリプションのマニフェストで、Satellite サーバーにアップロードできます。

## 2.8. マニフェストの SATELLITE SERVER へのアップロード

Web UI を使用したマニフェストのアップロード

#### Web UI を使用したマニフェストのアップロード

1. 組織が正しいことを確認します。
2. コンテンツ > **Red Hat サブスクリプション** をクリックします。
3. マニフェストの**管理** をクリックして、サブスクリプションページを開きます。
4. **ファイルの選択** をクリックして、作成したマニフェストファイルを選択します。
5. **アップロード** をクリックして、マニフェストを Satellite Server にアップロードします。

#### Hammer CLI を使用したマニフェストのアップロード

1. Satellite Server にマニフェストをアップロードします。

```
# hammer subscription upload \  
--file ~/manifest_file.zip \  
--organization "organization_name"
```

## 第3章 オブジェクトのデフォルト組織およびロケーションへの関連付け

### 3.1. 新規ドメインの作成

この手順を使用して新規ドメインを作成します。

#### Web UI をご利用の場合

1. インフラストラクチャー > ドメイン に移動して、**ドメインの作成** をクリックします。
2. **DNS ドメイン** フィールドに、完全な DNS ドメイン名を入力します。
3. **フルネーム** フィールドで、プレーンテキストのドメイン名を入力します。
4. **パラメーター** タブをクリックします。
5. **パラメーターの追加** をクリックし、**名前** および **値** フィールドに入力します。
6. **ロケーション** タブをクリックして、ドメインがある場所を追加します。
7. **組織** タブをクリックして、ドメインが属する組織を追加します。
8. **送信** をクリックして変更を保存します。

#### CLI をご利用の場合

`hammer domain create` コマンドを使用して、ドメインを作成します。

```
# hammer domain create --name "domain_name.com" \  
--description "My example domain" \  
--dns-id 1 --locations "New York" \  
--organizations "My_Organization"
```



#### 注記

この例では `--dns-id` オプションに 1 を指定しています。1 は、Satellite Server の Integrated Capsule ID です。

### 3.2. ドメインとデフォルトの組織との関連付け

#### Web UI を使用したドメインとデフォルトの組織との関連付け

1. メインメニューで、**インフラストラクチャー** → **ドメイン** をクリックします。
2. **説明** コラムでドメインを選択します。
3. **ロケーション** タブで **Default\_Location** をクリックして選択項目一覧に追加します。
4. **組織** タブで、**Default\_Organization** をクリックして選択項目一覧に追加します。
5. **送信** をクリックします。

## Hammer CLI を使用したドメインとデフォルトの組織との関連付け

1. ドメインを組織に関連付けます。

```
$ hammer organization add-domain --name $ORG --domain domain_name
```

2. ドメインをロケーションに関連付けます。

```
$ hammer location add-domain --name $LOCATION --domain domain-name
```

## 3.3. サブネットの設定

### Web UI を使用したサブネットの設定

1. メインメニューで、**インフラストラクチャー** → **サブネット** をクリックします。
2. **新規サブネット** をクリックして、お使いの環境の情報を入力します。

- a. DHCP を使用する場合には、以下の情報を入力します。

- 名前: Provisioning\_Net
- ネットワークアドレス: 172.17.13.0
- ネットワークマスク: 255.255.255.0
- ゲートウェイアドレス: 172.17.13.1
- プライマリー DNS サーバー: 172.17.13.2
- セカンダリー DNS サーバー: 空白にする
- IPAM: なし
- 開始 IP アドレス: 172.17.13.100
- 終了 IP アドレス: 172.17.13.150
- VLAN ID: 空白にする
- ブートモード: DHCP

- b. 静的 IP アドレスを使用する場合には、以下の情報を入力します。

- 名前: Provisioning\_Net
- ネットワークアドレス: 172.17.13.0
- ネットワークマスク: 255.255.255.0
- ゲートウェイアドレス: 172.17.13.1
- プライマリー DNS サーバー: 172.17.13.2
- セカンダリー DNS サーバー: 空白にする

- IPAM: なし
  - 開始 IP アドレス: 172.17.13.100
  - 終了 IP アドレス: 172.17.13.150
  - VLAN ID: 空白にする
  - ブートモード: 静的
3. **送信** をクリックします。
  4. **Provisioning\_Net** をクリックして、サブネットを編集します。
  5. **ドメイン** タブで、example.org を選択します。
  6. **Capsule** タブで、Satellite Server のホスト名を示す TFTP および検出プロキシーカプセルを変更します。
  7. **ロケーション** タブの **すべての項目** で Default\_Location を選択して、デフォルトの場所とドメインを関連付けます。
  8. **組織** タブの **すべての項目** で Default\_Organization を選択して、ドメインとデフォルトの組織を関連付けます。

### Hammer CLI を使用したサブネットの設定

1. サブネットを作成します。
  - a. DHCP を使用してサブネットを作成します。

```
# hammer subnet create --name Provisioning_Net \  
--organizations 'RedHat' \  
--locations 'RDU' \  
--domain-ids 1 \  
--boot-mode DHCP \  
--network 172.17.13.0 \  
--mask 255.255.255.0 \  
--ipam None \  
--dns-primary 172.17.13.1 \  
--gateway 172.17.13.1 \  
--from 172.17.13.100 \  
--to 172.17.13.150 \  
--tftp-id 1
```

- b. 静的 IP アドレスを使用してサブネットを作成します。

```
# hammer subnet create --name Provisioning_Net \  
--organizations 'RedHat' \  
--locations 'RDU' \  
--domain-ids 1 \  
--boot-mode Static \  
--network 172.17.13.0 \  
--mask 255.255.255.0 \  
--ipam None \  
--dns-primary 172.17.13.1 \  
--
```

```
--gateway 172.17.13.1 \  
--from 172.17.13.100 \  
--to 172.17.13.150 \  
--tftp-id 1
```



#### 注記

検出プロキシは、Hammer CLI では設定できません。



## 第4章 サブスクリプションのインポートおよびコンテンツの同期

### 4.1. RED HAT リポジトリーの有効化

Red Hat Enterprise Linux ホストのプロビジョニングをサポートするには、必要な Red Hat リポジトリーを有効にする必要があります。



#### 注記

Red Hat Satellite 6.4 のインストールに必要な一覧は、『リリースノート』の「[コンテンツ配信ネットワーク \(CDN\) のリポジトリー](#)」を参照してください。

環境に合わせて適切なリリースを選択します。ここでは、以下のリポジトリーを有効する例を紹介します。

- **Red Hat Enterprise Linux Server (Kickstart)**  
このリポジトリーを有効にして、ホストのキックスタートを実行します。
- **Red Hat Enterprise Linux Server (RPMs)**  
継続中のコンテンツおよびエラータに対してこのリポジトリーを有効にします。
- **Red Hat Enterprise Linux Server - Satellite Tools (RPMs)**  
Puppet、katello-agent などソフトウェアをサポートするこのリポジトリーを有効にします。

#### Web UI を使用した Red Hat Enterprise Linux 7 Server リポジトリーの有効化

1. Satellite Web UI で、**コンテンツ > Red Hat リポジトリー** に移動します。
2. 検索フィールドを使用して **Red Hat Enterprise Linux 7 Server (RPMs)** のリポジトリー名を入力します。
3. 利用可能なリポジトリーペインで、**Red Hat Enterprise Linux 7 Server (RPMs)** をクリックして、リポジトリーセットを展開します。
4. **x86\_64 7.5** エントリーでは、**有効化** アイコンをクリックして、リポジトリーを有効にします。
5. 検索フィールドを使用して **Red Hat Satellite Tools 6.4 (for RHEL 7 Server) (RPMs)** のリポジトリー名を入力します。
6. 利用可能なリポジトリーペインで、**Red Hat Satellite Tools 6.4 (for RHEL 7 Server) (RPMs)** をクリックして、リポジトリーセットを展開します。
7. **x86\_64** エントリーでは、**有効化** アイコンをクリックして、リポジトリーを有効にします。
8. 検索フィールドを使用して **Red Hat Enterprise Linux Server 7 (Kickstart)** のリポジトリー名を入力します。
9. 利用可能なリポジトリーペインで、**Red Hat Enterprise Linux 7 Server (RPMs)** をクリックして、リポジトリーセットを展開します。
10. **x86\_64 7.5** エントリーでは、**有効化** アイコンをクリックして、リポジトリーを有効にします。

## Hammer CLI を使用した Red Hat Enterprise Linux 7 Server リポジトリの有効化

1. キックスタートリポジトリを有効にします。

```
# hammer repository-set enable --organization-label "$ORG" \  
--product 'Red Hat Enterprise Linux Server' \  
--basearch='x86_64' \  
--releasever='7Server' \  
--name 'Red Hat Enterprise Linux 7 Server (Kickstart)'
```

2. RPM リポジトリを有効にします。

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

3. Satellite Tools リポジトリを有効にします。

```
# hammer repository-set enable --organization "$ORG" \  
--product 'Red Hat Enterprise Linux Server' \  
--basearch='x86_64' \  
--name 'Red Hat Satellite Tools {ProductVersion} (for RHEL 7 Server)  
(RPMs)'
```

## 4.2. カスタム製品の作成

カスタムパッケージを含むリポジトリを有効にすることができますが、その前に、リポジトリ用に製品を手動で作成する必要があります。

### Web UI を使用したカスタム製品の作成

1. **コンテンツ** → **製品** をクリックしてから、**新製品** をクリックしてください。
2. カスタム製品の名前を入力します。
3. **保存** をクリックします。
4. 画面が更新されたら **リポジトリの作成** をクリックします。
5. **名前** フィールドに **Puppet Modules** と入力します。
6. **タイプ** フィールドに **Puppet** と入力し、**URL** フィールドは空にします。
7. **保存** をクリックします。

### Web UI を使用した Puppet モジュールのリポジトリへのアップロード

Puppet モジュールを Puppet モジュールリポジトリにアップロードします。Puppet Forge をローカルにミラーリングする場合は、URL に <https://forge.puppetlabs.com> を使用できます。このように設定すると、Puppet Forge のすべてのコンテンツが、ご使用の Satellite で利用可能になります。ただし、2700 を超えるモジュールのダウンロードが必要となり、利用可能な帯域幅によっては時間がかかりか

かる場合があります。この例では、単純で、他のモジュールとの依存関係がない `motd` モジュールを使用します。大容量のファイルを伴うパッケージをアップロードする場合は、Hammer CLI を使用してください。

1. `motd puppet` モジュールを <https://forge.puppetlabs.com/jeffmccune/motd> からダウンロードします。ダウンロードしたファイルには拡張子 `.tar.gz` が付いています。
2. コンテンツ → 製品 をクリックしてから、名前 フィールドの **カスタム製品** をクリックしてください。
3. リポジトリ タブで、**Puppet Modules** をクリックして、Puppet Modules リポジトリを変更します。
4. パッケージのアップロード セクションで、ファイルの選択 をクリックし、ダウンロードした `motd` モジュールに移動します。
5. アップロード をクリックします。

### Hammer CLI を使用したカスタム製品の作成およびリポジトリの有効化

1. カスタム製品を作成します。

```
$ hammer product create --name "product_name" --organization-label
org_label
```

2. カスタム製品に新規リポジトリを作成します。

```
$ hammer repository create --name "repo_name" \
--organization-label org_label \
--product "product_name" \
--content-type cont_type \
--publish-via-http true \
--url "repo_url"
```

`--content-type` オプションでは、`yum`、`puppet` または `docker` のコンテンツのいずれかを指定します。このコマンドは、小文字しか対応していません。

3. カスタムリポジトリにパッケージをアップロードします。

```
$ hammer repository upload-content --product "product_name" \
--organization-label org_label \
--id "repo_id" \
--path path_to_dir
```

## 4.3. コンテンツの同期

Red Hat コンテンツ配信ネットワークのリポジトリは Satellite と同期させることができます。これは、リポジトリ URL を含むカスタムリポジトリ (Yum または Puppet など) を同期する場合にも適用されます。

### Web UI で Red Hat Enterprise Linux 7 サーバーを使用したコンテンツの同期

1. コンテンツ → 同期の状態 をクリックして、利用可能な製品一覧を表示します。

2. **Red Hat Enterprise Linux Server** → **7Server** → **x86\_64** に移動します。
3. 以下の製品を選択します。
  - Red Hat Enterprise Linux 7 Server RPMs x86\_64 6Server
  - Red Hat Enterprise Linux 7 Server - Satellite Tools 6.4 RPMs x86\_64
4. **Red Hat Enterprise Linux Server** → **7.5** → **x86\_64** に移動して、**Red Hat Enterprise Linux 7 Server Kickstart x86\_64 7.5** を選択します。
5. **今すぐ同期** をクリックします。  
利用可能な帯域幅によっては、同期に数時間かかる場合があります。

### Hammer CLI を使用したコンテンツの同期

1. キックスタートリポジトリを同期します。

```
$ hammer repository synchronize --organization "$ORG" \  
--product 'Red Hat Enterprise Linux Server' \  
--name 'Red Hat Enterprise Linux 7 Server Kickstart x86_64 7.5' \  
--async
```

2. RPM リポジトリを同期します。

```
$ hammer repository synchronize --organization "$ORG" \  
--product 'Red Hat Enterprise Linux Server' \  
--name 'Red Hat Enterprise Linux 7 Server RPMs x86_64 7Server' \  
--async
```

3. Satellite Tools リポジトリを同期します。

```
$ hammer repository synchronize --organization "$ORG" \  
--product 'Red Hat Enterprise Linux Server' \  
--name 'Red Hat Satellite Tools 6.4 for RHEL 7 Server RPMs x86_64' \  
--async
```

## 第5章 コンテンツの管理およびプロモーション

### 5.1. アプリケーションライフサイクル環境の作成

アプリケーションライフサイクル環境には、ソフトウェア開発ライフサイクル (SDLC) のプロモーションパスに関するステップまたはステージが表示されます。

1. **コンテンツ** → **ライフサイクル環境** をクリックして、**ライフサイクル環境パス** ページを開きます。
2. **新規環境の追加** をクリックして、**新規環境** ページを表示します。  
ライブラリーは、お使いの環境で使用可能な全コンテンツの提供元を示しています。
3. **名前** フィールドに名前を入力します。  
ラベルには同じ名前が自動的に設定されますが、自由に変更できます。また、ご使用の環境に関する説明を追加することができます。
4. **保存** をクリックします。

#### Hammer CLI を使用したライフサイクル環境の作成

1. ライフサイクル環境を作成します。

```
$ hammer lifecycle-environment create --name Development \  
--prior Library \  
--organization "$ORG"
```

### 5.2. シンプルなコンテンツビューの作成

コンテンツビューは、管理対象のコンテンツセレクションです。コンテンツビューには、オプションのフィルター機能と共にリポジトリが1つ以上 (yum または Puppet のいずれか) 含まれます。これらのフィルターは組み込むことも除外することもでき、これらのフィルターを使ってライフサイクル管理用にコンテンツのホストビューを調整することができます。また、コンテンツビューを使用して、クライアントホストで利用できるようコンテンツをカスタマイズすることもできます。

#### Web UI を使用したコンテンツビューの作成

1. **コンテンツ** → **コンテンツビュー** をクリックして、**新規ビューの作成** をクリックします。
2. **名前** フィールドに RHEL6 x86\_64 と入力します。  
ラベルが自動的に設定されます。
3. **複合ビュー?** のチェックボックスのチェックを外して、**保存** をクリックします。

#### Hammer CLI を使用したコンテンツビューの作成

1. コンテンツビューを作成します。

```
$ hammer content-view create --organization "$ORG" \  
--name 'RHEL7_Base' \  
--label rhel7_base \  
--description 'Core Build for RHEL 7'
```

## 5.3. RED HAT ENTERPRISE LINUX リポジトリの追加

Red Hat Enterprise Linux リポジトリをコンテンツビューに追加できます。オプションで、ここで説明している手順で、Red Hat リポジトリ、カスタムリポジトリ、またはその両方をコンテンツビューに追加できます。

以下の例では、すべてのコンテンツが公開されたコンテンツビューに含まれます。オプションで、フィルターを作成して、公開したコンテンツビューに組み込むか除外するかを制限することができます。

### Web UI を使用したリポジトリの追加

1. コンテンツ > コンテンツビュー に移動して、編集するコンテンツビューを選択します。
2. コンテンツビュー ペインで、**Yum コンテンツ** タブをクリックし、**リポジトリ** を選択します。
3. **追加** タブで、**リポジトリの選択** 一覧から以下のリポジトリを選択します。
  - Red Hat Enterprise Linux 7 Server Kickstart x86\_64 7.5
  - Red Hat Enterprise Linux 7 Server RPMs x86\_64 7Server
  - Red Hat Satellite Tools 6.4 for RHEL 7 Server RPMs x86\_64
4. **リポジトリの追加** をクリックします。  
コンテンツビューに追加したリポジトリの一覧を表示するには、**一覧/削除** タブをクリックします。

### Hammer CLI を使用したリポジトリの追加

1. Kickstart リポジトリを追加します。

```
$ hammer content-view add-repository --organization "$ORG" \
--name 'RHEL7_Base' \
--product 'Red Hat Enterprise Linux Server' \
--repository 'Red Hat Enterprise Linux 7 Server Kickstart x86_64
7.5'
```

2. RPM リポジトリを追加します。

```
$ hammer content-view add-repository --organization "$ORG" \
--name 'RHEL7_Base' \
--product 'Red Hat Enterprise Linux Server' \
--repository 'Red Hat Enterprise Linux 7 Server RPMs x86_64 7Server'
```

3. サテライトツールリポジトリを追加します。

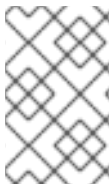
```
$ hammer content-view add-repository --organization "$ORG" \
--name 'RHEL7_Base' \
--product 'Red Hat Enterprise Linux Server' \
--repository 'Red Hat Satellite Tools 6.4 for RHEL 7 Server RPMs
x86_64'
```

## 5.4. コンテンツビューへの PUPPET モジュールの追加

Puppet モジュールは、ユーザー、ファイル、およびサービスなどのリソースを管理するために使用できるコードとデータの自己完結型のバンドルです。

### Web UI を使用した Puppet モジュールの追加

1. コンテンツビュー ページに移動します。
2. コンテンツ → コンテンツビュー をクリックします。
3. **Puppet モジュール** タブで **新規モジュールの追加** をクリックして、利用可能な Puppet モジュール一覧を表示します。  
フィルターフィールドを使用すると、必要なモジュールを特定しやすくなります。
4. **バージョンの選択** をクリックして、`motd` モジュールを選択します。
5. 追加するモジュールのバージョンの横の **バージョンの選択** をクリックします。



#### 注記

使用する Puppet モジュールのバージョンを選択する際に「最新を使用」を選択すると、コンテンツビューの新規バージョンが公開されるたびに、公開されるビューに、そのモジュールの最新バージョンが組み込まれることとなります。

### Hammer CLI を使用した Puppet モジュールの追加

1. Puppet モジュールを追加します。

```
$ hammer content-view puppet-module add \
  --content-view cv_name \
  --name module_name
```

## 5.5. コンテンツビューの公開

コンテンツビューおよび Puppet モジュールを作成したら、ライブラリーに公開できます。

### Web UI を使用したコンテンツビューの公開

1. コンテンツ → コンテンツビュー をクリックします。
2. 公開するコンテンツビューの名前をクリックします。
3. **新規バージョンの公開** をクリックして **新規バージョンの公開** ページを表示します。  
ここではバージョンが指定されており、コンテンツビューへの変更を反映する際のコメントを入力することができます。
4. **保存** をクリックして、ライブラリーのコンテンツビューを公開します。  
表示される画面で公開の進捗をモニターできます。
5. 公開プロセスが完了したら **プロモート** をクリックします。  
利用可能なプロモーションパスの一覧 (Library → Dev → QA) が表示されます。
6. Dev 環境のチェックボックスを選択してから **バージョンのプロモート** をクリックします。

## Hammer CLI を使用したコンテンツビューの公開

1. コンテンツビューを公開します。

```
$ hammer content-view publish --organization "$ORG" \  
--name RHEL7_Base \  
--description 'Initial Publishing'
```

2. コンテンツビューをプロモートします。

```
$ hammer content-view version promote \  
--organization "$ORG" \  
--content-view RHEL7_Base \  
--to-lifecycle-environment Development
```

## 5.6. アクティベーションキーの作成と編集

コンテンツビューを正常に公開したら、アクティベーションキーを作成してホストグループと関連付ける必要があります。

### Web UI を使用したアクティベーションキーの作成および編集

1. メインメニューから **コンテンツ** → **アクティベーションキー** → **新規アクティベーションキー** をクリックします。
2. 名前フィールドで、名前を入力します。
3. 必要に応じて、**コンテンツのホスト制限** のチェックボックスのチェックを外します。  
このフィールドを使用して、指定されたアクティベーションキーが使用される回数を制御できます。たとえば、数量に制限のあるサブスクリプションにキーを関連付ける場合に、その数量を超えないようにアクティベーションキーに制限を設定することができます。
4. お使いの環境について、チェックボックスを選択します。
5. **コンテンツビュー** のドロップダウンリストで、RHEL 7 x86\_64 コンテンツビューを選択して **保存** をクリックします。
6. **アクティベーションキー** ページで **サブスクリプション** タブをクリックし、**追加** タブをクリックして、利用可能なサブスクリプション一覧を表示します。
7. 利用可能なサブスクリプションの一覧から、追加するサブスクリプションを選択します。
8. **選択済み項目** をクリックします。

### Hammer CLI を使用したアクティベーションキーの作成および編集

1. アクティベーションキーを作成します。

```
$ hammer activation-key create --name ak-Reg_To_Dev_EL7 \  
--organization-label org_label \  
--content-view RHEL7_Base \  
--lifecycle-environment Development
```

2. サブスクリプションを追加します。



```
$ hammer subscription list --organization "$ORG"  
$ hammer activation-key add-subscription \  
--name ak-Reg_To_Dev_EL7 \  
--subscription-id 8 \  
--organization "$ORG"
```

3. リポジトリを有効にする製品コンテンツを設定します。

```
$ hammer activation-key content-override --name ak-Reg_To_Dev_EL7 \  
--organization "$ORG" \  
--content-label rhel-7-server-satellite-tools-6.4-rpms \  
--value 1
```

システムのパッチ修正に Satellite Server を使用する予定がない場合は「[プロビジョニングテンプレートの作成](#)」に進んでください。

## 第6章 システムのパッチ修正

Satellite サーバーからシステムにパッチを当てるには、システムを最初に登録して、Katello エージェントをインストールするか、リモート実行を使用します。デフォルトでは、リモート実行が有効になっています。この章では、両方の方法を説明します。お使いの環境に適したオプションを選択してください。

### 6.1. 既存のホストの登録

このセクションでは、Red Hat Enterprise Linux ホストおよび Atomic Host の登録方法を説明します。

#### 6.1.1. 既存の Red Hat Enterprise Linux ホストの登録

1. ホスト上で、以前のデータを消去して更新したデータが正しくアップロードされていることを確認します。

```
# subscription-manager clean
```

2. katello-ca-consumer-latest RPM をインストールします。

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

これで、正しい証明書がインストールされ、クライアントと Satellite Server 間の通信が可能になります。

3. ホストを登録します。

```
# subscription-manager register --org=$ORG \  
--activationkey=ak-Reg_To_Dev_EL7
```

#### 6.1.2. Atomic Host の登録

以下の手順では、サブスクリプションマネージャーで Atomic Host を登録する方法を説明します。

1. Satellite Server から **katello-rhsm-consumer** を取得します。

```
[root@atomic_client ~]# wget  
http://satellite.example.com/pub/katello-rhsm-consumer
```

2. **katello-rhsm-consumer** のモードを実行できるように変更します。

```
[root@atomic_client ~]# chmod +x katello-rhsm-consumer
```

3. **katello-rhsm-consumer** を実行します。

```
[root@atomic_client ~]# ./katello-rhsm-consumer
```

4. Red Hat サブスクリプションマネージャーで登録します。

```
[root@atomic_client ~]# subscription-manager register
```



## 注記

Atomic はアプライアンスとして機能するため、これに **katello-agent** をインストールすることは推奨されません。

## 6.2. KATELLO エージェントを使用したシステムのパッチ修正

### 6.2.1. Katello エージェントのインストール

コンテンツホストで **katello-agent** RPM パッケージをインストールします。

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

Red Hat Satellite Server または Capsule Server が、コンテンツホストに適用可能なエラータに関する情報を提供できるように、**goferd** サーバーが実行している必要があります。

**goferd** が実行していることを確認します。

- Red Hat Enterprise Linux 6 の場合は、以下のコマンドを実行します。

```
# service goferd start
```

- Red Hat Enterprise Linux 7 の場合は、以下のコマンドを実行します。

```
# systemctl start goferd
```

### 6.2.2. エラータのコンテンツホストへの適用

#### Web UI を使用したコンテンツホストへのエラータの適用

1. ホスト → コンテンツホスト に移動して、**auth01.example.com** をクリックします。
2. エラータ タブを選択して、コンテンツホストに適用可能なエラータ一覧を表示します。
3. 一覧からエラータを選択します。
4. **選択項目の適用** をクリックします。
5. 確認メッセージが表示されるので、**適用** をクリックします。
6. エラータがクライアントに適用されたことを確認します。

```
[root@client ~]# yum list-sec
```

#### Hammer CLI を使用したコンテンツホストへのエラータの適用

1. **auth01.example.com** ホストに適用するエラータを表示します。

```
# hammer host errata list --host auth01.example.com \  
--organization "$ORG"
```

2. ホストで必要なエラータを適用します。

```
# hammer host errata apply --host auth01.example.com \  
--organization "$ORG" \  
--errata-ids errata_id.
```

3. エラータがクライアントに適用されたことを確認します。

```
[root@client ~]# yum list-sec
```

## 6.3. リモートの実行を使用したシステムのパッチ修正

### 6.3.1. ホストでのリモート実行の有効化

Satellite Server のインストール中に、公開 SSH キーと共に内部のカプセルサーバーが自動的にインストールされます。内部の Capsule Server は、`/usr/share/foreman-proxy/.ssh/id_rsa_foreman_proxy` から SSH キーを読み込みます。公開 SSH キーをホストに配信すれば、リモート実行を有効にできます。

Satellite Server 上でキーをホストに配信します。

```
# ssh-copy-id -i ~foreman-proxy/.ssh/id_rsa_foreman_proxy.pub  
root@auth01.example.com
```

### 6.3.2. ホストへのエラータのインストール

Satellite は、エラータのインストールなど、リモートジョブを実行するデフォルトのジョブテンプレートを提供します。

1. **ホスト** → **すべてのホスト** に移動して、ホスト名の横のチェックボックスをクリックします。
2. **アクションの選択** をクリックして、ドロップダウンメニューから **リモートジョブのスケジュール** を選択します。
3. **ジョブカテゴリー** ドロップダウンメニューで、**Katello** を選択します。
4. **ジョブテンプレート** ドロップダウンメニューで、**Errata-Katello SSH デフォルトのインストール** を選択します。
5. **エラータ** フィールドで、エラータ ID を入力します。
6. **今すぐ実行** を選択して、**送信** をクリックします。

## 第7章 プロビジョニング設定の最終設定

### 7.1. カスタムのプロビジョニングテンプレートの作成

同じ設定を持つ複数ホストのプロビジョニングに使用できる、プロビジョニングテンプレートを設定できます。

#### Web UI を使用したプロビジョニングテンプレートの作成

1. メインメニューで、**ホスト** → **プロビジョニングテンプレート** をクリックします。
2. **名前** のコラムのプロビジョニングテンプレート一覧で **Kickstart Default iPXE** または **PXELinux** を選択します。  
これにより、テンプレートのカスタマイズに使用する各種設定タブが表示されます。
3. **関連付け** タブで、適切なオペレーティングシステムを選択して、**送信** をクリックします。
4. **名前** コラムで、適切なプロビジョニングテンプレートを選択します。  
PXE 以外の検出を使用する場合には、**Red Hat kexec の検出** のプロビジョニングテンプレートを選択します。
5. **関連付け** タブで、適切なオペレーティングシステムを選択して、**送信** をクリックします。
6. メインメニューで、**ホスト** → **オペレーティングシステム** をクリックして、適切なオペレーティングシステムを選択します。  
これにより、オペレーティングシステムのカスタマイズに使用する各種設定タブが表示されます。
7. **パーティションテーブル** タブで **Kickstart Default** を選択します。
8. **インストールメディア** タブで、  
Default\_Organization/Library/Red\_Hat\_6\_Server\_Kickstart\_x86\_64\_6Server が表示され、選択されていることを確認します。
9. **テンプレート** タブで、プロビジョニングのドロップダウンリストから **Katello Kickstart Default** を選択します。
10. **kexec** ドロップダウンリストから **Red Hat kexec の検出** を選択して、**送信** をクリックします。

#### Hammer CLI を使用したプロビジョニングテンプレートの作成

1. プロビジョニングテンプレートを作成します。

```
$ hammer template create --name template_name --file  
path_to_template_file
```

PXE を使用しない検出を使用する場合は、**kexec** テンプレートタイプを使用する必要があります。

2. オペレーティングシステムを追加します。

```
$ hammer template add-operatingsystem id
```

## 7.2. ホストグループの作成

ホストグループを作成して設定します。ホストグループを使えば複数のホストをプロビジョニングできるため、ホストごとに同じプロパティを指定する必要がなくなります。アクティベーションキーをホストグループに関連付ければ、プロビジョニングされたホストを Satellite Server に登録し、選択したライフサイクル環境、コンテンツビュー、サブスクリプションなどに関連付けることができます。

### Web UI を使用したホストグループの作成

1. メインメニューで、**設定** → **ホストグループ** をクリックして、**新規ホストグループ** をクリックします。
2. **ホストグループ** タブで、以下の値を入力します。
  - 名前: RHEL6Server-x86\_64
  - ライフサイクル環境: Default\_Organization/DEV
  - コンテンツビュー: RHEL\_6\_x86\_64
  - コンテンツソース: Capsule (今回は Satellite Server) の完全修飾ドメイン名
  - Puppet CA: Satellite の完全修飾ドメイン名
  - Puppet マスター: Satellite の完全修飾ドメイン名
3. **Puppet クラス** タブで、利用可能なクラス一覧から **motd puppet** モジュールを選択します。
4. **ネットワーク** タブで、以下の値を選択します。
  - ドメイン: example.org
  - サブネット: Provisioning\_Net
  - レルム: この例の便宜上、このフィールドは空白のままにします。たとえば、IPA などのレルム管理を設定している場合は、ここで該当するレルムを選択します。
5. **オペレーティングシステム** タブで以下の値を選択します。
  - アーキテクチャー: x86\_64
  - オペレーティングシステム: RHEL Server 6.5
  - メディア: Default\_Organization/Library Red Hat Server 6.5 x86\_64。この例では自動的に設定されます。組織とロケーションの関連付けが正しく設定されていないと、自動的にこのフィールドに値が入力されません。
  - パーティションテーブル: Kickstart default
  - Root パスワード: changeme
6. **ロケーション** タブで **Default\_location** を選択します。
7. **組織** タブで **Default\_Organization** を選択します。
8. **アクティベーションキー** タブで、適切なキーを選択します。
9. **送信** をクリックします。

## Hammer CLI を使用したホストグループの作成

1. ホストグループを作成します。

```
$ hammer hostgroup create --name "hostgroup_name" \  
--environment "environment_name" \  
--architecture "architecture_name" \  
--domain domain_name \  
--subnet subnet_name \  
--puppet-proxy proxy_name \  
--puppet-ca-proxy ca-proxy_name \  
--operatingsystem "os_name" \  
--partition-table "table_name" \  
--medium "medium_name" \  
--organization-ids org_ID1,org_ID2... \  
--location-ids loc_ID1,loc_ID2...
```

2. アクティベーションキーを追加します。

```
$ hammer hostgroup set-parameter --hostgroup "hostgroup_name" \  
--name "kt_activation_keys" \  
--value key_name
```

## 第8章 PXE を使わない検出イメージを使用したホストのプロビジョニング

Satellite では、主に 2 種類のアプローチでホストのプロビジョニング、PXE ブート、起動ディスクプロビジョニングを行います。起動ディスクプロビジョニング、または PXE を使用しない起動では、PXE サービスが利用できない場合にホストがプロビジョニングされます。本ガイドは、PXE を使用しないホストのプロビジョニングについても触れています。

検出イメージ ISO を使用して、ベアメタルのハードウェアまたは仮想システムのいずれかを起動し、Satellite Server にアタッチすることができます。DHCP および静的 IP ネットワークの両方がサポートされます。検出プラグインはデフォルトでインストールされており、サブスクリプションしているリポジトリに含まれます。

1. Satellite Server が検出プラグインのバージョンと互換性があることを確認します。
  - 検出プラグイン: tfm-rubygem-foreman\_discovery-5.0.0.4-1 以降
  - 検出イメージ: foreman-discovery-image-3.4.4-5 以降
  - Smart Proxy プラグイン: rubygem-smart\_proxy\_discovery-1.0.3-2 以降  
プラグインのアップグレードが必要な場合には、再起動をする必要があります。
2. 物理システムの場合は、ISO イメージを USB スティックまたは CDRROM/DVD に移動します。仮想システムの場合は、ISO イメージを仮想 CD-ROM にアタッチします。
3. システムで ISO イメージを起動します。  
テキストベースのユーザーインターフェースが表示されます。
4. 環境に合わせて、**手動のネットワーク設定** または **DHCP で検出** のいずれかを選択します。
5. ホストに複数の NIC がある場合には、主要なネットワークインターフェースを選択して **選択** をクリックします。
6. 該当する場合には、ネットワークの認証情報を入力して **次へ** をクリックします。
7. 完全な Satellite Server URL を入力して、**サーバー** の接続タイプを選択し、**次へ** をクリックします。
8. カスタムファクトを入力して自動プロビジョニングルールをトリガーし、**確認** をクリックします。  
ホストが検出され、「discovery\_kexec」ファクトが Satellite Server に送信されるので、プロビジョニング中に再起動されないようになります。
9. Web UI で **ホスト** → **検出済みホスト** に移動して、ホストを表示します。
10. **プロビジョニング** をクリックして、ホストの情報を入力して、**送信** をクリックします。