



Red Hat Satellite 6.1

プロビジョニングガイド

Red Hat Satellite Server から物理ホストと仮想ホストのプロビジョニングを実行するためのガイド。

エディション 1

Red Hat Satellite 6.1 プロビジョニングガイド

Red Hat Satellite Server から物理ホストと仮想ホストのプロビジョニングを実行するためのガイド。

エディション 1

Red Hat Satellite Documentation Team

法律上の通知

Copyright © 2015 Red Hat.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

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 プロビジョニングガイドは、物理ホストと仮想ホストのプロビジョニングに備えて Red Hat Satellite をインストールし、設定する際に役立つタスクベースの文書です。本書には、必要なネットワークポロジのセットアップや、必要なサービスの設定、およびネットワーク上でのホストのプロビジョニングに必要な他の設定に関する情報すべてが記載されています。本書は、ネットワークの十分な知識とスキルを持つ Satellite 管理者を主な対象としています。

目次

第1章 RED HAT SATELLITE を使用したプロビジョニングの概要	3
1.1. RED HAT ENTERPRISE LINUX ホストの作成	3
1.2. ネットワークポロジのセットアップ	3
1.3. ホストの登録および設定	3
第2章 RED HAT SATELLITE サービスの設定	6
2.1. DNS、DHCP、および TFTP の設定	6
2.2. 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	8
第3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	11
3.1. マニフェストの作成	11
3.2. マニフェストの SATELLITE SERVER へのアップロード	11
3.3. RED HAT リポジトリの有効化	11
3.4. カスタム製品およびリポジトリの作成	12
3.5. コンテンツの同期	13
第4章 コンテンツの管理とプロモーション	14
4.1. アプリケーションライフサイクル環境の作成	14
4.2. コンテンツビューの作成	14
4.3. RED HAT ENTERPRISE LINUX リポジトリの追加	15
4.4. PUPPET モジュールの追加	15
4.5. コンテンツビューの公開	15
4.6. アクティベーションキーの作成と編集	16
第5章 プロビジョニング設定の完了	17
5.1. プロビジョニングテンプレートの作成	17
5.2. ホストグループの作成	17
第6章 ホストのプロビジョニング	20
6.1. PXE を使用したホストのプロビジョニング	20
6.2. ブートディスクを使用したホストのプロビジョニング	20
6.3. 静的 IP アドレスを使用したホストのプロビジョニング	24
付録A 用語集	27
付録B 改訂履歴	31

第1章 RED HAT SATELLITE を使用したプロビジョニングの概要

本書は、ホストのプロビジョニングに備えて Red Hat Satellite Server を設定する際に役立つガイドです。本書では、Red Hat Enterprise Linux のインストールも含め、標準的なネットワークポロジータと利用可能になると予想されるサービス、およびホストの Red Hat Subscription Management への登録方法について説明します。

1.1. RED HAT ENTERPRISE LINUX ホストの作成

@Core または @Base パッケージセットのいずれかを使用して、Red Hat Enterprise Linux サーババージョン 6.6 以降を x86_64 にインストールします。Red Hat Enterprise Linux のインストールについての詳細は、[Red Hat Enterprise Linux 6 インストールガイド](#)を参照してください。

1.2. ネットワークトポロジータのセットアップ

本書では、Satellite 6 を実行するホストが、DHCP、DNS、および TFTP サービスを有効にできる専用サブネットにデプロイされていることを前提としています。使用されるサンプルでは 172.17.13.0/24 サブネットを使用します。さらに、Satellite で管理される example.org DNS ドメインを使用します。

また、これらのサンプルでは Satellite ホストに以下のネットワークの詳細が適用されていることを前提とします。以下のパラメーターをご使用のデプロイメントに合わせて調整してください。

- ホスト名: satellite.example.org
- IP アドレス = 172.17.13.2
- ネットマスク: 255.255.255.0

1.3. ホストの登録および設定

以下のセクションでは、ホストを登録する方法、サブスクリプションを特定する方法、さらにサブスクリプションを割り当て、ホストでコンテンツを使用可能にする方法について説明します。

1.3.1. Red Hat Subscription Management への登録

このプロセスの最初の手順では、ホストを Red Hat Subscription Management に登録します。この登録により、ユーザーの利用可能なサブスクリプションに基づいて、ホストがコンテンツをサブスクライブし、使用することが可能になります。このコンテンツには、Red Hat Enterprise Linux、Red Hat Software Collections (RHSC)、および Red Hat Satellite などが含まれます。Satellite を登録するには `subscription_manager register` コマンドを使用します。

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

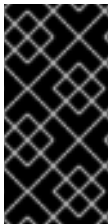
1.3.2. Satellite サブスクリプションの特定

ホストを登録した後は、Satellite サブスクリプションプール ID を特定する必要があります。この ID は、必要なサブスクリプションをホストに割り当てるために必要です。Satellite サブスクリプション

は、Red Hat Enterprise Linux、Red Hat Software Collections (RHSC)、および Red Hat Satellite と同様に Satellite コンテンツへのアクセスを提供します。Satellite サブスクリプションのみが必須のサブスクリプションになります。

以下のコマンドを実行して、Satellite サブスクリプションを特定します。

```
# subscription-manager list --all --available
Subscription Name: Red Hat Satellite
Provides:          Oracle Java (for RHEL Server)
                  Red Hat Satellite 6 Beta
                  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
```



重要

プール ID を書き留めておくようにしてください。この値は、サブスクリプションを Satellite ホストに割り当てるときに必要になります。この例では、プール ID は 8a85f9874152663c0541943739717d11 です。お客様のサブスクリプションでは、これとは異なる プール ID になります。

以下のコマンドを実行してサブスクリプションを Satellite に割り当てます。実際のプール ID に置き換えてください。

```
# subscription-manager attach --pool=8a85f9874152663c0541943739717d11
Successfully attached a subscription for: Red Hat Satellite
```

以下のコマンドを実行して、サブスクリプションが正常に割り当てられていることを確認します。

```
# subscription-manager list --consumed
+-----+
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 Beta
                  Red Hat Satellite 6 Beta
                  Red Hat Software Collections Beta (for RHEL Server)
                  Red Hat Satellite Capsule Beta
                  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
```

1.3.3. リポジトリの設定

すべての既存リポジトリを無効にし、Red Hat Enterprise Linux 6、Red Hat Software Collections、および Satellite 6 リポジトリのみを有効にします。以下のコマンドを実行します。

```
# subscription-manager repos --disable=*
# subscription-manager repos --enable rhel-6-server-rpms \
--enable rhel-server-rhsc1-6-rpms --enable rhel-6-server-satellite-6.0-
rpms
```

1.3.4. Satellite 6 のインストール

Satellite 6 をインストールするには、以下のコマンドを実行します。

```
# yum install katello
```

第2章 RED HAT SATELLITE サービスの設定

以下のサンプル設定では、Satellite はホストを 172.17.13.0/24 サブネットにプロビジョニングします。このセクションでは、サブネットにプロビジョニングされるクライアントに対してサービスを提供できるように DNS、DHCP、および TFTP を設定する方法について説明します。

2.1. DNS、DHCP、および TFTP の設定

このセクションでは、**example.org** ドメインおよび 172.17.13.x サブネットの権威 DNS サービスを提供するために BIND (**named**) を実行できるように Satellite を設定する方法について説明します。これには、**example.org** ゾーンファイルに含まれることになる、前方参照の DNS ゾーンのセットアップが必要です。さらに、逆引き参照の DNS ゾーンが 172.17.13.x サブネットに作成され、**13.17.172.in-addr.arpa** 逆引きゾーンファイルに含まれます。これにより、Satellite からプロビジョニングされるホストは正しい名前解決パラメーターを使用できます。このセクションでは、ホストが PXE を使って起動できるように TFTP プロキシを設定する方法についても説明します。

このネットワークのクライアントには、以下の特徴があります。

- DHCP の 172.17.13.100 から 172.17.13.150 までの範囲にある IP アドレスにアクセスできる。
- DNS 用に Satellite (172.17.13.2 の **satellite.example.org**) を使用する。
- PXE ブートを有効にするために Satellite (172.17.13.2 の **satellite.example.org**) から **pxelinux.0** ファイルを受信する。
- ホスト名は **hostname.example.org** であり、この **hostname** はホストのプロビジョニング時に設定される。



重要

この例では、Satellite Server で DHCP サービスを有効にします。タスクを続行する前に、ネットワーク管理者に連絡してください。

2.1.1. Satellite 設定オプション

以下の表では、Satellite Server を正しく設定するために必要な各種のオプションと値について説明しています。**katello-installer** コマンドは Puppet を使用します。そのため、このコマンドは追加のパッケージ (**bind**、**dhcp**、**xinetd** その他) をインストールし、必要な機能を追加できるようにそれらのパッケージを設定します。

選択可能なオプションの詳細の一覧については、**katello-installer --help** を実行します。

表2.1 Satellite 設定オプション

オプション	説明	値
<code>--foreman-admin-username</code>	最初の管理者のユーザー名です。	(ユーザー指定)
<code>--foreman-admin-password</code>	最初の管理者のパスワードです。	(ユーザー指定)

オプション	説明	値
--capsule-dns	DNS プロキシ機能を有効にします。	yes
--capsule-dns-interface	named がリッスンするインターフェースです。	eth0
--capsule-dns-zone	Satellite がホストする正引き DNS ゾーンです。	example.org
--capsule-dns-forwarders	不明なクエリーが転送される DNS サーバーです。	172.17.13.1
--capsule-dns-reverse	Satellite がホストする逆引き DNS ゾーンです。通常これは IP アドレスの最初の 3 つのオクテット (172.17.13) を逆にし、これに「.in-addr.arpa」を追加した値になります。	13.17.172.in-addr.arpa
--capsule-dhcp	DHCP プロキシ機能を有効にします。	yes
--capsule-dhcp-interface	DHCP がリッスンするインターフェースです。	eth0
--capsule-dhcp-range	クライアントに対して発行する IP アドレスの範囲です。	172.17.13.100 172.172.13.150
--capsule-dhcp-gateway	クライアントに対して発行するデフォルトのゲートウェイ IP です。	172.17.13.1
--capsule-dhcp-nameservers	クライアントが名前解決に使用する必要のあるホストです。このデプロイメントモデルでは、Satellite の IP を使って設定する必要があります。	172.17.13.2
--capsule-tftp	TFTP プロキシ機能を有効にします。これはクライアントを PXE ブートするために必要です。	yes
--capsule-tftp-servername	TFTP ホスト名を設定します。この TFTP ホスト名は、サーバーのホスト名 (satellite.example.org) と一致するように設定します。	\$(hostname)
--capsule-puppet	Puppet マスターを有効にします。	yes
--capsule-puppetca	Puppet CA を有効にします。	yes

2.1.2. Satellite サービスの設定

Satellite Server で必要なサービスを設定するために指定されたオプションを使い、root として以下の **katello-installer** コマンドを実行します。管理者ユーザー名とパスワードは、実際に必要な値に置き換えてください。

重要

- 『インストールガイド』の説明に従って Satellite をすでにインストールしている場合には、以下のコマンドに **--foreman-admin-username** と **--foreman-admin-password** オプションを含めないでください。
- 管理者ユーザー名とパスワードを指定しないと、デフォルトユーザー **admin** が作成され、パスワードが自動的に生成されます。資格情報はインストールプロセスの最後に表示されます。このパスワードを書き留めておいてください。さらに、パスワードは **/etc/katello-installer/answers.katello-installer.yaml** ファイルの **admin_password** パスワードから取得することもできます。

```
katello-installer \
--foreman-admin-username admin-username \
--foreman-admin-password admin-password \
--capsule-dns true \
--capsule-dns-interface eth0 \
--capsule-dns-zone example.org \
--capsule-dns-forwarders 172.17.13.1 \
--capsule-dns-reverse 13.17.172.in-addr.arpa \
--capsule-dhcp true \
--capsule-dhcp-interface eth0 \
--capsule-dhcp-range "172.17.13.100 172.17.13.150" \
--capsule-dhcp-gateway 172.17.13.1 \
--capsule-dhcp-nameservers 172.17.13.2 \
--capsule-tftp true \
--capsule-tftp-servername $(hostname) \
--capsule-puppet true \
--capsule-puppetca true
```

インストールプロセスの終わりに、**katello-installer** は以下のようなインストールの状態を表示します。

```
Success!
* Katello is running at https://satellite.example.org
  Default credentials are 'admin:*****'
* Capsule is running at https://satellite.example.org:9090
* To install additional capsule on separate machine continue by
running:"

    capsule-certs-generate --capsule-fqdn "$CAPSULE" --certs-tar
"~/ $CAPSULE-certs.tar"

The full log is at /var/log/katello-installer/katello-installer.log
```

web ブラウザーを使って <https://satellite.example.org> に移動し、Satellite ホームページを表示します。この例では、デフォルト組織 (Default_Organization) とデフォルトロケーションを使用しています。

2.2. 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

Satellite 6 は複数の組織 (論理的な管理区分) とロケーション (コンテンツ配信の物理的な区分) に対応するため、プロビジョニングに必要なテンプレート、サブネット、およびその他の項目を、デフォルト

組織 (Default_Organization) とデフォルトロケーション (Default_Location) に関連付ける必要があります。

以下の手順に従って、デフォルト組織 (Default_Organization) に対し、いずれかのコンテンツを追加する前のすべての「ベースコンテンツ (pre-seeded)」コンテンツを利用可能にします。

手順2.1 デフォルトロケーションの指定:

1. メインメニューで、**管理** → **ロケーション** をクリックしてから、**名前** 列にある **Default_Location** をクリックします。
2. **組織** をクリックして、組織の一覧を表示します。
3. **Default_Organization** をクリックして、これを **選択された項目** 一覧に追加し、**送信** をクリックします。

手順2.2 デフォルト組織の指定:

1. メインメニューで、**管理** → **組織** をクリックしてから、**名前** 列にある **Default_Organization** をクリックします。
2. **ロケーション** をクリックして利用可能なロケーションの一覧を表示します。
3. **Default_Location** をクリックして、これを **選択された項目** 一覧に追加し、**送信** をクリックします。

手順2.3 ドメインのデフォルト組織への関連付け:

1. メインメニューで、**インフラストラクチャー** → **ドメイン** をクリックしてドメイン 画面を開きます。
2. **フルネーム (Description)** 列にある **example.org** をクリックします。これにより、**ドメインの編集** 画面が開かれ、ドメインの詳細を更新できます。
3. **ドメイン** タブで、Satellite のホスト名を反映するように DNS ドメインを変更します。
4. **DNS Capsule** の値を Satellite Server に設定します。
5. **ロケーション** タブで、Default_Location をクリックし、これを **選択された項目** 一覧に追加します。これにより、ドメインがデフォルトロケーションに関連付けられます。
6. **組織** タブで、Default_Organization をクリックし、これを **選択された項目** 一覧に追加します。これにより、ドメインがデフォルト組織に関連付けられます。
7. **送信** をクリックして変更を適用します。

手順2.4 デフォルトサブネットの選択:

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** をクリックしてから、以下の情報を入力します。実際のデプロイメントに合わせて詳細情報を更新してください。
 - 名前: Provisioning_Net
 - ネットワーク (Network address): 172.17.13.0

- ネットワーク (Network mask): 255.255.255.0
 - ゲートウェイ (Gateway Address): 172.17.13.1
 - プライマリー DNS (Primary DNS Server): 172.17.13.2
 - セカンダリー DNS (Secondary DNS Server): 空白にする
 - 開始アドレス (Start of IP Range): 172.17.13.100
 - 終了アドレス (End of IP Range): 172.17.13.150
 - VLAN ID: 空白にする
3. **送信** をクリックします。
 4. サブネットを編集するために Provisioning_Net をクリックします。
 5. **ドメイン** タブで、example.org を選択します。
 6. **Capsule** タブで、Satellite のホスト名を反映するように DNS、DHCP、および TFTP Capsule を変更します。
 7. **ロケーション** タブで、**すべての項目** の下にある Default_Location を選択し、ドメインをデフォルトロケーションに関連付けます。
 8. **組織** タブで、**すべての項目** の下にある Default_Organization を選択し、ドメインをデフォルト組織に関連付けます。

手順2.5 インストールメディアの組織とロケーションへの関連付け:

1. メインメニューで、**ホスト** → **インストールメディア** をクリックします。
2. **名前** 列にある、使用するメディアの名前をクリックします。
3. **ロケーション** タブで、必要なロケーションを「選択された項目」の一覧に追加します。
4. **組織** タブで、必要な組織を「選択された項目」の一覧に追加し、**送信** をクリックします。

第3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

このセクションでは、コンテンツをダウンロードし、管理できるように Satellite をセットアップする方法について説明します。これには、マニフェストファイルの Satellite Server へのアップロード、Red Hat リポジトリの有効化、カスタム製品の作成、およびコンテンツの同期に関する情報が含まれます。

3.1. マニフェストの作成

このセクションでは、ご使用の Red Hat Satellite に適したマニフェストを作成する方法を説明します。

手順3.1 Satellite 6 のマニフェストの作成:

1. access.redhat.com に移動し、メインメニューの **サブスクリプション** をクリックします。
2. **Red Hat Subscription Management** セクションの **サブスクリプション管理 (Subscription Management Applications)** の下で、**Satellite** をクリックします。
3. マニフェストを作成する必要があるシステムを見つけます。必ず正しいバージョンを選択するようにしてください。
4. 割り当てる必要のあるそれぞれのサブスクリプションについて、そのサブスクリプションのチェックボックスを選択し、割り当てるサブスクリプションの数量を指定します。
5. **選択項目を追加** を選択します。



注記

すべてのサブスクリプションを割り当てるには、数分の時間がかかる場合があります。サブスクリプションが割り当てられたことを確認できるまで、数分ごとに画面を更新してください。

6. サブスクリプションが割り当てられた後に、**マニフェストのダウンロード** をクリックし、マニフェストファイルを既知のロケーションに保存します。

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

手順3.2 マニフェストの Satellite Server へのアップロード:

1. 適切な組織をまだ選択していない場合、**すべてのコンテキスト** → **任意の組織** → **Default_Organization** をクリックします。
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. **参照** をクリックして適切なマニフェストを選択してから **開く** をクリックします。
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。

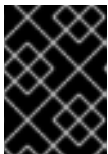
3.3. RED HAT リポジトリの有効化

このセクションでは、Red Hat Enterprise Linux 6 ホストのプロビジョニングをサポートするために必要な Red Hat リポジトリを有効にする方法について説明します。ご使用のデプロイメントに適した必要なリリースを選択してください。この手順では、以下のリポジトリを有効にします。

- Red Hat Enterprise Linux 6 Server (Kickstart): これを使って、Satellite はホストのキックスタートを実行できます。このリポジトリは Satellite 5 の「kickstart tree」と類似のものです。
- Red Hat Enterprise Linux 6 Server (RPM): このリポジトリは、継続的にコンテンツおよびエラータを提供します。これは、Satellite 5 の「rhel-x86_64-server-6」チャンネルと類似のものです。
- Red Hat Enterprise Linux 6 Server - Satellite Tools (RPMs): このリポジトリは、Puppet および katello-agent などのサポートソフトウェアを提供します。これは、Satellite 5 の「rhn-tools-rhel-x86_64-server-6」チャンネルと類似のものです。

手順3.3 Red Hat リポジトリの有効化:

1. コンテンツ → **Red Hat** リポジトリ を選択します。
2. キックスタート タブから **Red Hat Enterprise Linux Server** → **Red Hat Enterprise Linux 6 Server (Kickstart)** に移動し、Red Hat Enterprise Linux 6 Server Kickstart x86_64 6.5 (以降) のリポジトリを選択します。



重要

6Server Kickstart リポジトリは選択しないでください。Satellite の今後のバージョンでは、これらの不要な Kickstart リポジトリは非表示になります。

3. RPM タブから **Red Hat Enterprise Linux Server** → **Red Hat Enterprise Linux 6 Server (RPMs)** に移動し、Red Hat Enterprise Linux 6 Server RPMs x86_64 6Server リポジトリを選択します。
4. RPM タブから **Red Hat Enterprise Linux Server** → **Red Hat Enterprise Linux 6 Server - Satellite Tools (RPMs)** に移動し、Red Hat Enterprise Linux 6 Server - Satellite Tools RPMs x86_64 リポジトリを選択します。

3.4. カスタム製品およびリポジトリの作成

このセクションでは、デプロイする Puppet モジュールを反映させたカスタム製品を作成する方法について説明します。また、この手順を使って Puppet と Yum の両方のカスタムリポジトリを作成することもできます。

手順3.4 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 **新規製品** をクリックします。
2. **名前** フィールドに **Custom Products** と入力します。「ラベル」にはデータが自動的に生成されます。GPG キー、同期プランまたは説明を入力する必要はありません。
3. **保存** をクリックします。
4. 画面の更新後に、**リポジトリの作成** をクリックします。
5. 名前に **Puppet Modules** と入力します。「ラベル」にはデータが自動的に生成されます。

6. **タイプ** フィールドで、**puppet** を選択します。**URL** フィールドを空白のままにします。

7. **保存** をクリックします。

次の手順で、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。また、Puppet Forge をローカルにミラーリングするには、<https://forge.puppetlabs.com> を URL として使用することもできます。この設定にすると、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かなりの時間がかかる場合があります。この例では、motd モジュールを使用しています。このモジュールは単純で、他のモジュールとの依存関係がないためです。

手順3.5 Puppet モジュールのリポジトリへのアップロード:

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 **.tar.gz** 拡張子が付いています。
2. **コンテンツ** → **製品** をクリックしてから **名前** フィールドにある **Custom Products** をクリックします。
3. **リポジトリ** タブで、**Puppet Modules** をクリックし、Puppet Modules リポジトリを変更します。
4. **Puppet モジュールのアップロード** セクションで、**参照** 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。

3.5. コンテンツの同期

このセクションでは、Red Hat コンテンツ配信ネットワークのリポジトリを Satellite に同期する方法について説明します。この手順は、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。

手順3.6 リポジトリの Satellite への同期:

1. **コンテンツ** → **同期の状態** をクリックして、利用可能な製品の一覧を表示します。
2. **Red Hat Enterprise Linux Server** → **6Server** → **x86_64** に移動します。
3. 以下の製品を選択します:
 - Red Hat Enterprise Linux 6 Server RPMs x86_64 6Server
 - Red Hat Enterprise Linux 6 Server - Satellite Tools RPMs x86_64
4. **Red Hat Enterprise Linux Server** → **6.6** → **x86_64** に移動し、Red Hat Enterprise Linux 6 Server Kickstart x86_64 6.6 を選択します。
5. **今すぐ同期** をクリックします。

リポジトリが同期されるのを待ちます。利用可能な帯域幅によって数時間かかる場合があります。

第4章 コンテンツの管理とプロモーション

この章では、アプリケーションライフサイクル環境とコンテンツビューのセットアップ方法、および Red Hat Enterprise Linux リポジトリと Puppet モジュールの追加方法について説明します。また、コンテンツビューの公開方法と、アクティベーションキーの作成および編集方法についても説明します。

この章にある手順では、[3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](#) の手順が正常に完了していることを前提とします。

4.1. 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おけるステップまたはステージを表します。この例の前半では、「Dev」と「QA」の2つのライフサイクル環境を設定します。この例の後半では、それらの環境で使用するコンテンツビューを作成します。

手順4.1 アプリケーションライフサイクル環境の作成:

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** 画面を開きます。
2. **新規環境** ページを表示するために **新規環境の追加** をクリックします。**ライブラリー** はご使用の環境で使用できるすべてのコンテンツの出所になります。
3. **名前** フィールドに **Dev** と入力します。「ラベル」には同じ名前が自動的に設定されますが、これはご使用のニーズに合わせて変更できます。必要な場合はご使用の環境についての説明を追加することができます。
4. **保存** をクリックして、新しい組織を保存し、直前のページに戻ります。
5. **新規環境の追加** を再びクリックした後、今度は **QA** という環境を作成します。
6. **保存** をクリックします。

4.2. 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1つ以上のリポジトリ (yum または Puppet のいずれか) が含まれます。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、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ためにこれらを使用することもできます。

手順4.2 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** をクリックしてから、**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」にデータが自動的に設定されます。
3. **複合ビュー** チェックボックスがクリアされていることを確認してから、**保存** をクリックします。

コンテンツビューが正常に作成されると、**リポジトリの選択** 画面が自動的に表示されます。この画面を使用して選択されたリポジトリと Puppet モジュールをコンテンツビューに追加します。

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

以下の手順では、直前の手順で作成されたコンテンツビューに Enterprise Linux リポジトリを追加する方法について説明します。同じ手順を使用してすべての Red Hat またはカスタムリポジトリを追加することができます。

この例は、すべてのコンテンツが公開される単純なユースケースを示しています。フィルターを作成して、公開されたコンテンツビューにコンテンツを組み込んだり、コンテンツを除外したりするなどコンテンツを制御することもできます。

手順4.3 Red Hat Enterprise Linux RPM リポジトリの追加:

1. **リポジトリの選択** 画面の **追加** タブで、以下のそれぞれのリポジトリの横にあるチェックボックスを選択します。
 - Red Hat Enterprise Linux 6 Server Kickstart x86_64 6Server
 - Red Hat Enterprise Linux 6 Server RPMs x86_64 6Server
 - Red Hat Enterprise Linux 6 Server - Satellite Tools RPMs x86_64
2. **リポジトリの追加** をクリックします。ページが更新されると、**一覧/削除** タブでコンテンツビューのリポジトリの一覧を確認することができます。

4.4. PUPPET モジュールの追加

以下の手順では、[手順3.5「Puppet モジュールのリポジトリへのアップロード:」](#) でアップロードされた motd Puppet モジュールを追加する方法について説明します。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。

手順4.4 Puppet モジュールの追加:

1. **コンテンツビュー** ページを開いていることを確認してから、メインメニューで **コンテンツ** → **コンテンツビュー** をクリックします。
2. **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、利用可能な Puppet モジュールの一覧を表示します。必要なモジュールを見つけるために **フィルター** フィールドを使用することができます。
3. **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。
4. 追加するモジュールのバージョンの横にある **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 (Use Latest)」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、そのモジュールの最新バージョンが公開されるビューに組み込まれることとなります。

4.5. コンテンツビューの公開

これまでの手順をすべて正常に完了すると、Red Hat Satellite には、3 つの Red Hat Enterprise Linux リポジトリと 1 つの Puppet モジュールが含まれる 1 つのコンテンツビューがで

きます。次の手順では、これをライブラリーに公開します。

手順4.5 コンテンツビューのライブラリーへの公開:

1. **コンテンツ** → **コンテンツビュー** をクリックして、**コンテンツビュー** ページを表示します。
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ためにコメントを入力することができます。
4. **保存** をクリックしてコンテンツビューをライブラリーに公開します。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、**プロモート** をクリックすると、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。

4.6. 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後は、アクティベーションキーを作成する必要があります。後述の手順では、アクティベーションキーはホストグループに関連付けられます。これにより、プロビジョニングされたホストは Satellite に登録され、選択され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られます。

手順4.6 アクティベーションキーの作成:

1. メインメニューで、**コンテンツ** → **アクティベーションキー** をクリックしてから **新規アクティベーションキー** をクリックします。
2. 「名前」フィールドに、**ak-Reg_to_Dev** と入力します。
3. この例の便宜上、**コンテンツホストの制限** チェックボックスをクリアします。

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、キーを数量に制限のあるサブスクリプションに関連付ける場合、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. Dev 環境のチェックボックスを選択します。
5. **コンテンツビュー** ドロップダウンリストで、RHEL 6 x86_64 コンテンツビューを選択してから **保存** をクリックします。

アクティベーションキーを作成した後に、キーの各種パラメーターを編集することができます。

手順4.7 アクティベーションキーパラメーターの編集:

1. **アクティベーションキー** ページで、**サブスクリプション** → **追加** をクリックして利用可能なサブスクリプションの一覧を表示します。
2. このアクティベーションキーを使用する各ホストに割り当てるサブスクリプションを特定し、それぞれのサブスクリプションに隣接するチェックボックスを選択します。
3. **選択を追加** をクリックします。

第5章 プロビジョニング設定の完了

コンテンツビューおよびアクティベーションキーを正常に作成した後は、ホストのプロビジョニングに必要な残りの項目をセットアップする必要があります。これには、プロビジョニングテンプレートの設定とホストグループの作成が含まれます。

5.1. プロビジョニングテンプレートの作成

このセクションでは、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セットアップする方法について説明します。

手順5.1 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** 列のプロビジョニングテンプレート一覧にある **Satellite Kickstart Default** をクリックします。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、適用可能なオペレーティングシステムの一覧から **RHEL Server 6.5** を選択し、**送信** をクリックします。
4. **名前** 列のプロビジョニングテンプレート一覧にある **Kickstart default PXELinux** をクリックします。
5. **関連付け** タブで、適用可能なオペレーティングシステムの一覧から **RHEL Server 6.5** を選択し、**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** をクリックしてから **RHEL Server 6.5** をクリックします。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は、**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、**PXELinux** ドロップダウンリストから **Kickstart default PXELinux** を選択します。
10. **プロビジョニング (Provision)** ドロップダウンリストから **Satellite Kickstart Default** を選択し、**送信** をクリックします。

5.2. ホストグループの作成

このセクションでは、ホストグループを作成し、設定する方法について説明します。ホストグループは事実上、各ホストに同じプロパティを設定せずに複数ホストをプロビジョニングする場合の、再利用可能なホストテンプレートになります。

手順5.2 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** をクリックしてから **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



注記

以下のフィールドは、**ライフサイクル環境** フィールドに値を入力した後のみ表示されます。

- **コンテンツソース:** Capsule (Satellite Server の場合あり) の FQDN。
 - **Puppet CA:** Satellite の FQDN。
 - **Puppet マスター:** Satellite の FQDN。
3. **Puppet クラス** タブで、利用可能なクラスの一覧から `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、このフィールドにデータが自動的に設定されます。



注記

このフィールドにデータが自動的に設定されない場合は、組織とロケーションの関連付けが正しく行なわれていないことを意味します。[手順2.5「インストールメディアの組織とロケーションへの関連付け:」](#)を参照してください。

- **パーティションテーブル:** Kickstart default
 - **ルートパス:** *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、アクティベーションキー の一覧から ak-Reg_To_Dev を選択します。
9. 送信 をクリックします。

第6章 ホストのプロビジョニング

この章では、Red Hat Satellite Server を使って新規ホストをプロビジョニングする方法について説明します。これまでの章では、プロビジョニングに必要なものすべてをインストールし、設定する方法を説明してきました。ホストのプロビジョニングを試行する前に、これらの章のすべてのタスクを正常に完了していることを確認してください。

Satellite はホストのプロビジョニングにおける 2 つの主なアプローチを提供します。1 つ目は、DHCP および TFTP サービスを必要とする PXE ブートです。2 つ目は、PXE サービスが利用できない場合にホストのプロビジョニングを可能にするブートディスクを使用したプロビジョニングです。

6.1. PXE を使用したホストのプロビジョニング

以下の手順では、Satellite 6 Server からホストをプロビジョニングする方法を説明します。

手順6.1 ホストのプロビジョニング:

1. メインメニューで、**ホスト** → **新規ホスト** をクリックし、**新規ホスト** ページを開きます。
2. **ホスト** タブで、以下の値を入力します。
 - **名前:** ホストに適した名前を選択します。例: **host1.example.org**
 - **ホストグループ:** RHEL6Server-x86_64 を選択します。



注記

新規ホストはホストグループに設定されるデフォルト値を継承します。つまり、これらの値を再入力することなくホストをすぐにビルドできます。

- **コンテンツソース:** Satellite の \$FQDN です。このフィールドには、ホストグループに基づいてデータが自動的に設定されます。
3. **ネットワーク** タブで、以下の値を入力します。
 - **MAC アドレス:** 新規ホストの MAC アドレスです。Satellite Server はこの値を使って DHCP アドレスを予約します。必ず正しい値を入力してください。
 - **サブネット:** **Provisioning_Net** この値は自動的に設定されます。
 - **IP アドレス:** この値は自動的に設定されます。
- Puppet** クラス、オペレーティングシステム、パラメーター、または **追加情報** タブには一切変更を加えないでください。
4. **送信** をクリックします。
 5. ホスト (物理ホストまたは仮想マシンのいずれか) のパワーをオンにしてください。PXE ブートが実行され、インストールプロセスが開始されます。

詳細は、<https://access.redhat.com/articles/1169613> を参照してください。

6.2. ブートディスクを使用したホストのプロビジョニング

Satellite のネットワークプロビジョニングモデルは、通常 DHCP および TFTP サービスを必要とする

PXE に基づいています。しかし、すべての Satellite デプロイメントでこれらのサービスが利用可能であるとは限らないため、ブートディスクのプロビジョニング機能では、そのようなデプロイメントでのプロビジョニングを有効にするために、ホスト固有、完全ホスト、および汎用タイプのブートディスクイメージを提供しています。

ブートディスクイメージの各タイプにはそれぞれの利点がありますが、いずれもネットワークインフラストラクチャーの制御のない環境用に設計されています。したがって DHCP 予約または TFTP 設定は不要になります。

ブートイメージはハイブリッド ISO イメージ (ISO ファイルまたは USB ディスクとして利用可能) として作成され、物理メディアまたは仮想ディスクないしは CD のいずれかから起動することができます。

表6.1 ブートイメージタイプの特徴比較

タイプ	汎用	DHCP 必須	DHCP 予約	事前登録ホスト	オペレーティングシステム固有
ホスト固有イメージ	No	No	No	Yes	No
完全ホストイメージ	No	Yes	No	Yes	Yes
汎用イメージ	Yes	Yes	No	Yes	No

6.2.1. 要件

Satellite ブートディスク機能の前提条件となるすべてのパッケージは、通常デフォルトでインストールされています。以下の条件に対応していることを確認してから、次に進んでください。

- 使用するイメージのタイプにかかわらず、ホストはイメージから起動する前に Satellite に登録されている必要があります。ホストがビルドモードの場合、正しいプロビジョニングテンプレートを提供するためにホストの識別がそれらの MAC または IP アドレスを使って行われます。
- ホスト固有のイメージの場合、ホスト IP アドレスおよびサブネットが事前に設定されており、かつサブネットのゲートウェイ、サブネットマスク、および DNS リゾルバーが正しく設定されていることを確認します。これらの値を設定するには、**インフラストラクチャー → サブネット** に移動します。
- 管理者以外のユーザーのイメージへのアクセスを許可するには、ユーザーに「Boot disk access」ロールを追加するか、または既存ロールに「download_bootdisk」パーミッションを追加します。
- 適切な iPXE プロビジョニングテンプレートが利用可能であることを確認します。Red Hat Enterprise Linux は「Kickstart default iPXE」テンプレートを使用します。



注記

gPXE イメージは現在サポートされていません。

ホストおよび汎用イメージタイプは、PXELinux の異なるセットのハードウェアドライバーをサポートする iPXE テクノロジーに基づいています。サポートされているハードウェアの一覧については、http://ipxe.org/appnote/hardware_drivers を参照してください。

iPXE に関連する問題に直面する場合でも、完全ホストイメージの場合は、組み込みカーネルおよび RAM ディスクが含まれており、PXE サポートのないカードを含むあらゆる種類のネットワークカードを読み込むことができます。

- 適切な標準テンプレートが利用可能であることを確認します。Red Hat Enterprise Linux は、プロビジョニングテンプレートとして「Kickstart RHEL default」を使用します。



重要

デフォルトの Satellite キックスタートプロビジョニングテンプレートを使用していない場合、使用するテンプレートがオペレーティングシステムの設定に必要な静的 IP の詳細情報を提供するものであることを確認します。キックスタートファイルの場合、以下の設定を使用することができます。

```
network --bootproto <%= @static ? "static" : "dhcp" %> --
hostname <%= @host %> <%= " --ip=#{@host.ip} --netmask=#
{@host.subnet.mask} --gateway=#{@host.subnet.gateway} --
nameserver=#{@host.subnet.dns_primary},#
{@host.subnet.dns_secondary}" if @static %>
```

6.2.2. ブートディスクイメージの作成

このセクションでは、ホスト固有、完全ホスト、および汎用ブートディスクイメージを作成する方法について説明します。イメージの作成には Web UI またはコマンドラインのいずれかを使用することができます。これら 2 つの方法について説明します。



注記

コマンドラインを使用してイメージを作成するには、`ruby193-rubygem-foreman_bootdisk` パッケージがインストールされていることを確認します。通常、これはデフォルトでインストールされています。

6.2.2.1. ホスト固有イメージの作成

静的ネットワークでホスト固有のイメージを作成するには、Satellite のホストおよびサブネットデータを使用できます。この動作は動的です。Satellite からイメージのチェーンロードが行われ、結果として現在のオペレーティングシステムとビルド状態は、イメージに保存されるのではなく Satellite によって提供されます。

手順6.2 Web UI を使用したホスト固有のイメージの作成:

1. **ホスト** → **すべてのホスト** に移動し、該当するホスト名をクリックします。
2. **ブートディスク** をクリックしてから、ホスト **hostname** イメージ をクリックします。

hammer CLI ツールを使用してホスト固有のイメージを作成するには、以下のコマンドを実行します。

```
# hammer bootdisk host --host client.example.com
```

Satellite Server 上のコマンドラインからホスト固有イメージを作成するには、以下のコマンドを実行します。

```
# foreman-rake bootdisk:generate:host NAME=client.example.com OUTPUT=/tmp
```

OUTPUT の値を、ディレクトリーまたはファイルのいずれかの適切な宛先パスに設定します。**foreman** ユーザーは指定した宛先への書き込みアクセスを持っている必要があります。

6.2.2.2. 完全ホストイメージの作成

完全ホストイメージはホスト固有のイメージに似ていますが、完全ホストイメージの場合は、Satellite からチェーンロードするのではなく、初期オペレーティングシステムブートローダーが含まれます。これは、チェーンロードに失敗したホストの場合に役立ちますが、ホストのオペレーティングシステム、ブートローダーまたはテンプレートが変更される場合や、ビルドトークンが必要な場合でそれらの有効期限が切れる場合にはイメージが古くなり、無効になるというマイナス面があります。

手順6.3 Web UI を使用した完全ホストイメージの作成:

1. **ホスト** → **すべてのホスト** に移動し、該当するホスト名をクリックします。
2. **ブートディスク** をクリックしてから **完全ホスト *hostname* イメージ** をクリックします。

完全ホストイメージの作成プロセスには、サイズの大きなオペレーティングシステムブートローダーのダウンロードが含まれるため、長い時間がかかります。

hammer CLI ツールを使用して完全ホストイメージを作成するには、以下のコマンドを実行します。

```
# hammer bootdisk host --host client.example.com --full true
```

Satellite Server 上のコマンドラインから完全ホストイメージを作成するには、以下のコマンドを実行します。

```
# foreman-rake bootdisk:generate:full_host NAME=client.example.com
```

6.2.2.3. 汎用イメージの作成

汎用イメージは、すべての登録済みホストで使用できる単一の ISO ファイルを提供します。IP アドレスの詳細はこれらのイメージに保存することができませんが、これはネットワークが DHCP プールを提供する必要があることを意味します。ホストを起動するために汎用イメージを使用すると、DHCP によってホストに割り当てられた MAC アドレスまたは IP に一致する登録済みホストのテンプレートについて Satellite への問い合わせが行われます。

オペレーティングシステムの iPXE テンプレートが設定される方法に応じて、DHCP によって割り当てられる IP アドレスか、または静的 IP アドレスのいずれかを使用したインストールを続けることができます。キックスタートファイルを使用して、追加のネットワーク設定オプションを指定することができます。

手順6.4 Web UI からの汎用イメージの作成:

1. **ホスト** → **すべてのホスト** に移動し、該当するホスト名をクリックします。

2. **ブートディスク** をクリックしてから **汎用イメージ** をクリックします。

hammer CLI ツールを使用して汎用イメージを作成するには、以下のコマンドを実行します。

```
# hammer bootdisk generic
```

Satellite Server 上のコマンドラインから汎用イメージを作成するには、以下のコマンドを実行します。

```
# foreman-rake bootdisk:generate:generic
```

6.2.2.4. USB イメージの作成

ISO ファイルを作成する場合は、常に **isohybrid** コマンドで渡されます。つまり、結果として作成されるファイルはディスクとしても起動可能であり、USB デバイスへのコピーにも適していることとなります。

ISO ファイルを USB デバイスにコピーするには、以下のコマンドを実行します。デバイス名と入力ファイルがご使用の環境に適した正しい値であることを確認してください。

```
# dd if=fqdn.iso of=/dev/sdX
```

6.3. 静的 IP アドレスを使用したホストのプロビジョニング

Red Hat Satellite 6 では、所定 MAC アドレスの DHCP レコードを予約するため、すべてのシステムが DHCP 用に設定されていることが想定されています。静的 IP アドレスを使ったホストのプロビジョニングは、カスタムプロビジョニングテンプレートやホストパラメーターを使用するか、またはサブネット情報に基づいて行うことができます。

6.3.1. 静的 IP アドレスを割り当てるためのカスタムテンプレートの使用

プロビジョニングされるすべてのホストの静的 IP サポートを提供するカスタムプロビジョニングテンプレートを作成することができます。PXE 設定テンプレートはコピーして、「RHEL 7.1 static」などのように別のオペレーティングシステムに関連付けることができます。つまり、このテンプレートを使用してシステムをキックスタートすると、静的 IP の受け取りが行われます。Satellite 6.0 および 6.1 では既存の「新規ホストの作成」ワークフローでの動的または静的 IP 設定の選択に対応していないため、現時点でこの方法は必須になっています。

この方法では、静的ネットワークを有効にするために PXE テンプレートを編集する必要があります。以下の例で説明されているように、PXE テンプレートを編集し、`&static=yes` を `foreman_url('provision')` の各インスタンスの末尾に追加します。

手順6.5 PXE テンプレートの編集:

1. **ホスト** → **すべてのホスト** に移動し、編集する必要のあるテンプレートを持つホストの名前をクリックします。
2. **テンプレート** タブをクリックし、選択可能なテンプレートタイプの一覧を表示します。
3. **PXE Linux** テンプレート タイプの **編集** をクリックします。テンプレートがテンプレートエディターに表示されます。

4. 必要な変更を加えてから **送信** をクリックします。PXE テンプレートの更新方法の例については、[例6.1「静的 IP アドレスを割り当てるための PXE テンプレートの編集」](#) を参照してください。

例6.1 静的 IP アドレスを割り当てるための PXE テンプレートの編集

```
default linux
label linux
kernel <%= @kernel %>
<% if @host.operatingsystem.name == 'Fedora' and
@host.operatingsystem.major.to_i > 16 -%>
append initrd=<%= @initrd %> ks=<%=
foreman_url('provision')%>&static=yes ks.device=bootif network
ks.sendmac
<% elsif @host.operatingsystem.name != 'Fedora' and
@host.operatingsystem.major.to_i >= 7 -%>
append initrd=<%= @initrd %> ks=<%=
foreman_url('provision')%>&static=yes network ks.sendmac
<% else -%>
append initrd=<%= @initrd %> ks=<%=
foreman_url('provision')%>&static=yes ksdevice=bootif network kssendmac
<% end -%>
IPAPPEND 2
```

6.3.2. 静的 IP アドレスを割り当てるためのホストパラメーターの使用

ホストパラメーターを使用して、ホストごとに静的 IP アドレスをセットアップすることができます。新規ホストを作成する場合、**ホスト** → **新規ホスト** に移動してから **パラメーター** タブに必要なパラメーターを入力します。

[例6.2「静的 IP アドレスを割り当てるためのホストパラメーターの使用」](#) は、静的 IP アドレスを設定するために、ホストパラメーターと共に **if** ステートメントを使用する方法を示しています。

例6.2 静的 IP アドレスを割り当てるためのホストパラメーターの使用

```
<% if @host.params['my_param_deploy_static'] == "true" -%>
  network --bootproto <%= "static --ip=#{@host.ip} --netmask=#
{@host.subnet.mask} --gateway=#{@host.subnet.gateway} --nameserver=#
[[@host.subnet.dns_primary,@host.subnet.dns_secondary].reject{|n|
n.blank?}.join(',')}" %> --hostname <%= @host %>
<% else -%>
  network --bootproto 'dhcp' --hostname <%= @host %>
<% end -%>
```

詳細は、[『Red Hat Satellite 6 User Guide』](#) を参照してください。

6.3.3. 静的 IP アドレスを設定するためのサブネットの使用

ホストのサブネットに基づいて静的な IP アドレスを使ったホストのプロビジョニングを行えるよう Red Hat Satellite を設定することができます。サブネットをセットアップする場合、DHCP または静的ブートモードのいずれかを指定することができます。Red Hat Enterprise Linux インストールプロ

グラム (Anaconda) はこの値を使用して、静的 IP アドレスまたは DHCP プールからのアドレスを割り当てるかどうかを判別します。このサブネットにプロビジョニングされるすべてのホストが静的 IP アドレスを受け取れるようにするには「静的」を指定します。

手順6.6 デフォルトの静的ブートモードの指定:

1. インフラストラクチャー → サブネット に移動して、設定する必要があるサブネットの名前をクリックします。新規サブネットを作成するには、[手順2.4「デフォルトサブネットの選択:](#)」の手順に従います。
2. サブネット タブで、ブートモード ドロップダウンリストから **静的** を選択してから **送信** をクリックします。

新規ホストを作成し、それをこのサブネットに割り当てると、ホストは常にデフォルトで静的 IP アドレスを使用します。

付録A 用語集

以下は、本書全体で使用されている用語です。これらの用語を十分に理解しておくこと、Red Hat Satellite 6 を理解するのに役立ちます。

Activation Key (アクティベーションキー)

登録時のアクションを制御するためにキックスタートファイルで使用される登録トークンです。これは Red Hat Satellite 5 のアクティベーションキーと似ていますが、登録後の Puppet によるパッケージと設定管理のコントロールに必要な機能のサブセットを提供します。

Application Life Cycle Environment (アプリケーションライフサイクル環境)

アプリケーションライフサイクル環境は、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おけるステップまたはステージを表します。プロモーションパスは開発パスとしても知られています。パッケージや Puppet モジュールなどのコンテンツは、コンテンツビューの公開およびプロモートにより、ライフサイクル環境間を移動します。すべてのコンテンツビューにはバージョンがあります。つまり、特定のバージョンを標準的なプロモーションパス上でプロモートすることができます。たとえば、「development (開発)」から「test (テスト)」、さらには「production (本番)」に移行できます。Red Hat Satellite 5 では、チャンネルのクローン作成がこの概念を実行するものとなってきました。

Attach (割り当て)

RPM コンテンツへのアクセスを提供するサブスクリプションをホストに割り当てるプロセスです。

Capsule

Capsule は Red Hat Satellite 6 デプロイメントで使用できる追加サーバーであり、他のサービス (Puppet マスター、DHCP、DNS、TFTP など) をローカル化するほか、コンテンツの連合化 (フェデレーション) と配布 (ディストリビューション) を容易にします。

Catalog (カタログ)

カタログは、1 つの特定のコンピューターのシステム状態について説明するドキュメントです。これは、管理する必要のあるすべてのリソースと、それらリソース間の依存関係を一覧表示します。

Compute Profile (コンピュートプロファイル)

コンピュートプロファイルは、コンピュートリソース上にある新規仮想マシンのデフォルト属性を指定します。

Compute Resource (コンピュートリソース)

コンピュートリソースは、Red Hat Satellite 6 がホストやシステムのデプロイに使用する仮想またはクラウドのインフラストラクチャーです。例として、Red Hat Enterprise Virtualization Manager、OpenStack、EC2、および VMWare などがあります。

Content (コンテンツ)

コンテンツには、ソフトウェアパッケージ (RPM ファイル) および Puppet モジュールが含まれます。これらのコンテンツはライブラリーに同期された後に、コンテンツビューを使ってライフサイクル環境にプロモートされます。これにより、ホストがこれらのコンテンツを使用できるようになります。

Content Delivery Network (CDN) (コンテンツ配信ネットワーク (CDN))

コンテンツ配信ネットワーク (CDN) は、同じ地理的拠点にサーバーが設置されるコロケーション形態の場合の、Red Hat コンテンツの配信に使用されるメカニズムです。たとえば、ヨーロッパの Satellite がコンテンツを同期する場合、ヨーロッパにあるソースからコンテンツをプルします。

Content Host (コンテンツホスト)

コンテンツホストは、コンテンツとサブスクリプションに関連するタスクを管理するホストの一部です。

Content View (コンテンツビュー)

コンテンツビューは、製品やパッケージおよび Puppet モジュールをインテリジェントなフィルタリングやスナップショット作成などの各種機能に組み合わせたコンテンツの定義です。コンテンツビューは Red Hat Satellite 5 のチャンネルとクローン作成の組み合わせをさらに改良したものです。

External Node Classifier (外部ノードの分類子)

外部ノードの分類子は、ホストの設定時に Puppet マスターが使用する追加データを提供する Puppet コンストラクトです。Red Hat Satellite 6 は、Satellite デプロイメントで Puppet マスターに対する外部ノードの分類子として機能します。

Facter

Facter は一種のプログラムであり、それが実行されるシステムについての情報 (ファクト) を提供します。たとえば、Facter は合計メモリー、オペレーティングシステムのバージョン、アーキテクチャーなどをレポートすることができます。Puppet モジュールは、Facter によって収集されるホストデータに基づいて特定の設定を有効にします。

Hammer

Hammer は、Red Hat Satellite 6 のコマンドラインツールです。Hammer を標準 CLI として使用するか、またはスクリプトやインタラクティブシェルで使用して Red Hat Satellite 6 を管理します。

Hiera

Hiera は設定データのキー/値のルックアップツールであり、Puppet マニフェストからサイト固有のデータを取り分けることができます。

Host (ホスト)

ホストは、Red Hat Satellite 6 が管理する、物理および仮想を含むすべてのシステムを指します。

Host Collection (ホストコレクション)

ホストコレクションは、Satellite 5 のシステムグループに相当するもので、1 つ以上のホストのユーザー定義グループを指します。

Host Group (ホストグループ)

ホストグループは、ホストをビルドするためのテンプレートです。これには、(利用可能な RPM ファイルと Puppet モジュールを定義する) コンテンツビューや、(ソフトウェアと設定を最終的に決定する) 適用対象の Puppet クラスが含まれます。

Location (ロケーション)

ロケーションは、物理的な場所を表すデフォルト設定のコレクションです。これらはネスト化できるため、ロケーションの階層的なコレクションをセットアップできます。たとえば、「中東」から始まり、「テルアビブ」、「データセンター東」、さらには「ラック 22」と詳細化してデフォルトをセットアップできます。

Library (ライブラリー)

ライブラリーにはすべてのバージョンが含まれ、これにはユーザーがデプロイするソフトウェアの最近同期したバージョンも含まれます。IT インフラストラクチャーライブラリー (ITIL) ^[1] の組織 (organization) または部署 (department) のコンテキストでは、これは確定版メディアライブラリー^[2] に相当します (以前の名称は確定版ソフトウェアライブラリー)。

Manifest (マニフェスト)

マニフェストは、サブスクリプションをカスタマーポータルから Red Hat Satellite 6 に移行します。マニフェストの機能は、Red Hat Satellite 5 で使用されている証明書に似ています。

証明書およびサブスクリプションのタイプの詳細は、以下を参照してください。

- [RHN Classic, Red Hat Satellite, and Channel Entitlements](#)^[3]
- [The Structure of Satellite Certificates \(Classic Style of Certificates\)](#)^[4]

Organization (組織)

組織は、Satellite 6 デプロイメント内の複数のシステム、コンテンツおよびその他の機能からなる単独のコレクションです。

Product (製品)

コンテンツリポジトリのコレクションです。「製品」には Red Hat 製品、またはソフトウェアと設定コンテンツで構成される新規に作成される製品が含まれます。

Promote (プロモート)

ソフトウェアと設定コンテンツで構成されるコンテンツビューを 1 つのアプリケーションライフサイクル環境から別のアプリケーションライフサイクル環境に移行する動作を指します。たとえば、「Development (開発)」から「QA」、または「Production (本番)」への移行などが含まれます。

Provisioning Template (プロビジョニングテンプレート)

プロビジョニングテンプレートは、キックスタートファイル、スニペットその他プロビジョニング操作のためのユーザー定義テンプレートです。Satellite 6 のこのテンプレートは、Red Hat Satellite 5 のキックスタートプロファイルと Cobbler スニペットと同様の機能を提供します。

Pulp Node (Pulp ノード)

Pulp ノードは、コンテンツをミラーリングする Capsule Server のコンポーネントです。これは Red Hat Satellite 5 Proxy に似ています。主な違いは、Pulp ノードの場合には、コンテンツがホストによって使用される前にそのノード上でコンテンツをステージングできる点にあります。

Puppet Agent (Puppet エージェント)

Puppet エージェントは、ホスト上で実行されるエージェントであり、設定の変更をホストに適用します。

Puppet Master (Puppet マスター)

Puppet マスターは、Puppet エージェントが実行する Puppet マニフェストをホストに提供する Capsule Server のコンポーネントです。

Puppet Module (Puppet モジュール)

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。

Repository (リポジトリ)

リポジトリは、コンテンツのコレクション用にストレージを提供します。たとえば、YUM リポジトリまたは Puppet リポジトリなどがあります。

Role (ロール)

ロールは、ホストなどの一連のリソースに適用されるパーミッションのコレクションを指定します。

Smart Proxy (スマートプロキシー)

スマートプロキシーは、DNS または DHCP などの外部サービスと統合できる Capsule Server のコンポーネントです。

Smart Variable (スマート変数)

スマート変数は Puppet クラスの動作を制御する設定値です。この値は、ホスト、ホストグループ、組織、またはロケーションに設定できます。

Standard Operating Environment (SOE) (標準運用環境 (SOE))

標準運用環境 (SOE) は、アプリケーションがデプロイされるオペレーティングシステムの制御されたバージョンです。

Subscription (サブスクリプション)

サブスクリプションは、Red Hat からコンテンツとサービスを受け取る手段です。

Synchronizing (同期)

同期は、外部リソースのコンテンツを Red Hat Satellite 6 ライブラリーにミラーリングすることを指します。

Synchronization Plan (同期プラン)

同期プランは、コンテンツ同期のスケジュールに基づく実行を可能にします。

User Group (ユーザーグループ)

ユーザーグループは、ユーザーのコレクションに割り当てることができるロールのコレクションです。これは、Red Hat Satellite 5 のロールに似ています。

User (ユーザー)

ユーザーは、Red Hat Satellite を使用できるように登録されたすべてのユーザーを指します。認証および認可については、組み込みロジック、外部 LDAP リソースまたは Kerberos を使って実行できます。

[1] http://en.wikipedia.org/wiki/Information_Technology_Infrastructure_Library

[2] http://en.wikipedia.org/wiki/Definitive_Media_Library

[3] https://access.redhat.com/site/documentation/en-US/Red_Hat_Subscription_Management/1/html/MigratingRHN/sat-certs.html

[4] https://access.redhat.com/site/documentation/en-US/Red_Hat_Subscription_Management/1/html/Subscription_Concepts_and_Workflows/index.html#subscription_legacy

付録B 改訂履歴

改訂 1.2-8 Removing beta watermark.	Wed August 12 2015	Hayley Hudgeons
改訂 1.2-7 翻訳および校閲完了	Tue Jun 23 2015	Aiko Sasaki
改訂 1.2-6.1 翻訳ファイルを XML ソースバージョン 1.2-6 と同期	Tue Jun 23 2015	Aiko Sasaki
改訂 1.2-6 更新された製品バージョンを使用して再ビルド。	Fri May 15 2015	David O'Brien
改訂 1.2-5 テクニカルレビュー向けのビルド。	Mon May 11 2015	David O'Brien
改訂 1.2-4 Satellite 6.1 で静的 IP を使用方法を文書化。	Thu Apr 16 2015	David O'Brien
改訂 1.2-3 ブランドを更新。 前書きを削除。	Wed Apr 8 2015	Megan Lewis
改訂 1.2-2 新規ポータル設計に適したマニフェストの作成手順を更新。	Fri Apr 3 2015	David O'Brien
改訂 1.2-1 BZ 1207429: 「ホストグループの作成」手順のエラーを修正。コンテンツソースは Capsule を参照する必要あり。 「PTR zone file」への混乱を招く参照を明確化。 BZ 1203878: 「RH Common」リポジトリ名を「Satellite Tools」に更新。 BZ 1168240: 「ドメイン」対「ゾーン」についてのセクションを明確化。 BZ 1176466: katello-installer コマンドの必要なオプションを明確化。 ホストグループの作成方法についての手順を明確化。 BZ 1183657: 「puppet module (puppet モジュール)」と「catalog (カタログ)」を用語集に追加。 「バグを報告する」リンクのエンティティを修正。	Tue Mar 31 2015	David O'Brien
改訂 1.2-0 BZ 1184671: katello-installer オプションのエラーを修正。	Thu Feb 22 2015	David O'Brien
改訂 1.1-1 Satellite が正しいバージョンを参照するためのホストの作成方法についてのセクションを更新。 「カスタム製品およびリポジトリの作成」セクションの関連リンクを修正。	Mon Dec 22 2014	David O'Brien
改訂 1.1-0 詳細情報を参照するためにマニフェストを作成する方法についてのセクションを更新。	Wed Dec 10 2014	David O'Brien
改訂 1.0-0 QA レビューに基づく更新を実装。 初回の公開版リリース。	Wed Nov 26 2014	David O'Brien
改訂 0.1-1	Thu Nov 20 2014	David O'Brien

用語集の用語を追加。

コンテンツの管理とプロモーションでのコンテンツビューの作成の一部としてコンテンツビューの定義を追加。

参照される手順の xref を追加。

管理者ユーザー名とパスワードを含めるために初期設定についてのセクションを更新。

ホストのプロビジョニングについての章を更新。

前書きの追加。

プロビジョニングテンプレートの設定についての章のドラフト版を作成。

コンテンツ管理とプロモーションについてのセクションのドラフト版の作成。

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についてのセクションを改訂。

改訂 0.1-0

Fri Oct 17 2014

David O'Brien

「サービスの設定」の章の初版ドラフトを追加。

「プロビジョニングの概要」の章のドラフト版を追加。

基本セクションに関連してスケルトン TOC を追加。

Publican による初版作成。