



Red Hat Network Satellite 5.5

チャンネル管理ガイド

Red Hat Network Satellite

エディション 7

Red Hat Network Satellite 5.5 チャンネル管理ガイド

Red Hat Network Satellite

エディション 7

Red Hat Documentation Team

法律上の通知

Copyright © 2010 Red Hat, Inc.

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 Network Satellite チャンネル管理ガイドへようこそ

目次

第1章 はじめに	3
第2章 RHN チャンネルとは	4
2.1. ベースチャンネルと子チャンネル	4
2.2. チャンネルをサブスクリプションする	4
2.3. 利用可能なチャンネル	5
2.4. ツールとリポジトリ、および実践例について	5
第3章 カスタムのパッケージを構築する	7
3.1. RED HAT NETWORK のパッケージを構築する	7
3.1.1. RPM の利点	7
3.1.2. RHN RPM ガイドライン	8
3.2. RHN パッケージ用のデジタル署名	9
3.2.1. GnuPG キーペアを生成する	9
3.2.2. パッケージに署名する	11
第4章 カスタムチャンネルとパッケージの管理	12
4.1. チャンネル管理の特権	12
4.2. ソフトウェアチャンネルの管理	12
4.3. 管理しているソフトウェアチャンネルの詳細	13
4.4. ソフトウェアパッケージの管理	15
4.5. ソフトウェアチャンネルを作成する	16
4.6. パッケージをソフトウェアチャンネルに割り当てる	17
4.7. ソフトウェアチャンネルのクローンを作成する	17
4.8. ソフトウェアチャンネルを削除する	18
第5章 カスタムのエラータ管理	20
5.1. エラータの管理	20
5.1.1. 発行済みエラータ	20
5.1.2. 未発行のエラータ	20
5.2. 管理しているエラータの詳細	20
5.3. エラータの作成と編集	21
5.4. パッケージをエラータに割り当てる	22
5.5. エラータのクローンを作成する	22
第6章 カスタムパッケージをアップロードして保守作業を行う	24
6.1. パッケージを RHN PROXY SERVER にアップロードする	24
6.1.1. RHN Package Manager を設定して使用する	24
6.2. パッケージを RHN SATELLITE SERVER にアップロードする	28
6.2.1. RHN Push アプリケーションを設定する	28
6.2.2. RHN Push アプリケーションを使用する	30
付録A 改訂履歴	32
索引	33

第1章 はじめに

本ガイドでは RHN Proxy Server および RHN Satellite Server 用にカスタマイズしたソフトウェアチャンネルの配備や保守に関連する事項について解説しています。RHN Satellite Server または RHN Proxy Server のインストールと設定が完了してからご使用ください。

本文中、Red Hat Network Web サーバー上で実行される動作に関する記載があります。RHN Proxy Server を使用している場合、Web サーバーとは <https://rhn.redhat.com> にある Red Hat Network の中央サーバーを指しています。Satellite を使用している場合は、お客様のサイトにある RHN Satellite Server を指しています。

第2章 RHN チャンネルとは

Red Hat Network (RHN) チャンネルとは、複数のソフトウェアパッケージをひとつにまとめた集合です。目的に応じたルールでパッケージ群を分別する場合に役立ちます。例えば、特定の **Red Hat** ディストリビューションのパッケージ群をすべて一つのチャンネルに含ませることができます。または、特定のアプリケーション用の全パッケージを含ませたり、特定ファミリーの傘下となるアプリケーション群用の全パッケージを含ませたりすることもできます。利用する側の特定のニーズに合わせてチャンネルを定義することも可能です。例えば、企業で支給するノートブックパソコンにインストールさせる全パッケージを含ませたチャンネル、なども作成することができます。

2.1. ベースチャンネルと子チャンネル

チャンネルには **2** 種類のタイプがあります。ベースチャンネルと子チャンネルです。ベースチャンネルは、アーキテクチャと **Red Hat Enterprise Linux** のリリースに応じたパッケージ群で構成されます。子チャンネルは、ベースチャンネルに付随するチャンネルで、その他のパッケージを含んでいます。

システムがサブスクライブできるベースチャンネルはひとつのみです。そのベースチャンネルに付随する子チャンネルは複数サブスクライブさせることができます。**Red Hat Network** チャンネルにサブスクライブさせたシステムは、そのチャンネルから利用できるパッケージしかインストールや更新を行うことはできません。

システムを **Red Hat Network** に登録すると、そのシステムの **Red Hat Enterprise Linux** バージョンに応じたベースチャンネルが割り当てられます。このデフォルトのベースチャンネルをプライベートのベースチャンネルに変更する場合は、システムの登録が完了してからシステム単位で行なうことができます。または、カスタムのベースチャンネルに関連付けを行ったアクティベーションキーを使用することもできます。このようなアクティベーションキーで登録を行うと、システムに自動的にそのカスタムのベースチャンネルが割り当てられます。

Red Hat Network Web サイト上の **チャンネル** ページ (上部ナビゲーションバーにある **チャンネル** タブの下) には、全ベースチャンネルとその子チャンネルの一覧が表示されます。チャンネル名をクリックすると、**チャンネルの詳細** ページが表示されます。このページには、そのチャンネル内の全パッケージ、エラータ、関連付けられているシステムなどの一覧が表示されています。

2.2. チャンネルをサブスクライブする

システムを次の手順でチャンネルにサブスクライブさせます。

- アクティベーションキーを使って登録する – アクティベーションキーを使用する方法が簡潔で早いため、**RHN Proxy Server** や **RHN Satellite Server** いずれかのクライアントとしてシステムを登録する場合には適した方法になります。アクティベーションキーを使用して登録されたシステムはそのアクティベーションキーに関連付けられたすべてのチャンネルにサブスクライブされます。アクティベーションキーに関しては『**Red Hat Network** クライアント設定ガイド』または『**Red Hat Network** リファレンスガイド』をご覧ください。
- インストール時に登録する – **Red Hat Update Agent** または **Red Hat Network Registration Client** のいずれかで初めてシステムを登録すると、そのシステムの **Red Hat Enterprise Linux** バージョンに応じたベースチャンネルが自動的に割り当てられます。そのデフォルトのベースチャンネルをプライベートのベースチャンネルに変更する場合は、一旦システムの登録を完了させた後、**RHN Web** サイトからシステム単位で行なうことができます。または、カスタムのベースチャンネルに関連付けを行ったアクティベーションキーを使用することもできます。このようなアクティベーションキーで登録を行うと、システムに自動的にそのカスタムのベースチャンネルが割り当てられます。このアプリケーションの使い方については、エンタイトルメントレベル (**Management** または **Provisioning**) に応じて『**RHN** リファレンスガイド』の該当章を参照してください。

- **Web** サイトでサブスクライブするシステムのベースチャンネルに応じて、さまざまな子チャンネルのサブスクライブが可能です。子チャンネルへのサブスクライブは **RHN Web** サイトから行なうことができます。カスタムのベースチャンネルを作成している場合、そのカスタムのベースチャンネルへの再割り当ても **Web** サイトから行なうことができます。オンラインでのチャンネルへのサブスクライブについては、『**RHN** リファレンスガイド』の **Red Hat Network Web** サイトの章を参照してください。
- **spacewalk-channel** コマンドラインツール (CLI) を使用する **-spacewalk-channel** を使用すると、**Red Hat Network Web** サイトにログインしなくてもコマンドラインから特定のチャンネルへのサブスクライブを行なうことができます。

いくつか例を示します。2種類のチャンネルにサブスクライブさせる場合:

```
spacewalk-channel --add -c rhn-tools-rhel-i386-server-5 -c
rhel-i386-server-vt-5 --user username --password password
```

チャンネルのサブスクライブを中止する場合:

```
spacewalk-channel --remove -c rhn-tools-rhel-i386-server-5 -c
rhel-i386-server-vt-5 --user username --password password
```

サブスクライブしているチャンネルを表示させる場合:

```
spacewalk-channel --list
```

2.3. 利用可能なチャンネル

Red Hat Network には多くのチャンネルがあります。全てのユーザーが利用できるチャンネル、特定の組織に所属するユーザーが利用できるチャンネル、またアクセスを購入した場合にのみ利用できるチャンネルなどがあります。チャンネルは次のようなカテゴリに別れます。

- 有料サービスチャンネル – 有料サービスチャンネルへのアクセスを直接購入した場合、または特定の **Red Hat** ソリューションと合わせてアクセスを購入した場合に使用できるチャンネルです。**Red Hat Enterprise Linux** などが有料サービスチャンネルの一例です。
- カスタムのチャンネル – カスタムのパッケージを管理する目的で管理者により作成されるチャンネルです。プライベートのチャンネルとも呼ばれるこのチャンネルは、デフォルトでは作成側の企業でしか表示されません。このため、企業外からのアクセスは一切ありません。ただし、プライベートのチャンネルは、組織的な信頼を設定しチャンネルを共有することで複数の組織間での共有が可能になります。組織の信頼については『リファレンスガイド』をご覧ください。

本ガイドでは **RHN Proxy Server** または **RHN Satellite Server** でカスタムチャンネルを作成して保守を行っていく過程について焦点を置き説明していきます。

2.4. ツールとリポジトリ、および実践例について

チャンネルを作成、管理する前に、ユーザーが使用できる各種ツールやリポジトリの違いについて注意が必要です。特に **RHN Satellite Server** と **RHN Proxy Server** の両方を導入する場合は、使用できるユーティリティや収納場所も増えるため重要となります。また、**Proxy** と **Satellite** を組み合わせることにより、最適なパフォーマンスを得るためのベストプラクティスを実現します。

まずパッケージ管理ツールについてよく知っておきましょう。

- **RHN Package Manager** - カスタムのパッケージを RHN Proxy Server 上のカスタムチャンネルにプッシュする際に使用します。
- **RHN Push** - カスタムのパッケージを RHN Satellite Server 上のカスタムチャンネルにプッシュする際に使用します。
- **RHN Satellite Synchronization Tool** - Red Hat Network Classic から RHN Satellite Server に標準パッケージをインポートしたり同期したりする際に使用します。インターネット経由または CD や DVD の ISO イメージを利用して行います。

各ツールにはそれぞれ対応するパッケージリポジトリがあります。 **RHN Package Manager**、 **RHN Push** いずれの場合も、 **Proxy** または **Satellite** にアップロードするカスタムパッケージの置き場所として一時的なステージングディレクトリを用意する必要があります。使用后、ステージングディレクトリは削除してください。



注記

カスタムのパッケージをアーカイブする際は、 **Red Hat Network** とは別の場所にアーカイブすることを推奨しています。

RHN Proxy Server と RHN Satellite Server の両方を使用している場合は、 **RHN Push** および **RHN Satellite Synchronization Tool** 以外のツールは使用しないでください。 **Proxy** と **Satellite** を組み合わせる場合、カスタムのパッケージやチャンネルのアップロード先は **Satellite のみ** にしてください。 **Proxy** の役目は、この **Satellite** からパッケージを取得してクライアントのシステム群に配信することです。

第3章 カスタムのパッケージを構築する

ソフトウェアパッケージを構築する際、失敗の原因となるものは数多くあります。特に、Red Hat Network を介してパッケージを配信しインストールする必要がある場合に顕著になります。本章では Red Hat Network を介して正しく配信できるパッケージの構築方法の概要を記載しています。RPM を使用する理由、RHN 用のパッケージの構築方法、パッケージの正しい署名法などについて説明しています。

3.1. RED HAT NETWORK のパッケージを構築する

Red Hat Network は RPM パッケージマネージャ (RPM) の技術を使用して、各クライアントシステムに適用できる追加ソフトウェアや更新などを確認します。Red Hat Network から取り込まれるパッケージは通常 RPM 形式になりますが、全 ISO イメージは Red Hat Network Web サイトのソフトウェア タブで入手可能です。ただし、Satellite Server の場合には利用できません。Satellite で Solaris のサポートを有効にしている場合は、RHN Push を使用して Solaris のクライアント群で使用されるカスタムチャンネルに Solaris のパッケージ群をアップロードしてください。

RPM はソフトウェアパッケージのインストール、アンインストール、アップグレード、検証などをユーザーが簡単に行うことができるツールです。また、ソフトウェア開発者なら、このツールを使用してプログラムのコンパイル版とそのソースコードをエンドユーザーおよび開発者向けにパッケージ化することもできます。

3.1.1. RPM の利点

RPM には次のような利点があります。

簡単なアップグレード

RPM を使用すると、新たに再インストールを行わなくてもシステムの個別コンポーネントをアップグレードすることができます。Red Hat から Red Hat Enterprise Linux の新しいバージョンがリリースされたとき、ユーザーによるアップグレードのための再インストールは必要ありません。RPM によって完全に自動制御されているシステムアップグレードが用意されています。パッケージ内の設定ファイルはアップグレード後も保持されているためカスタムの設定が失われることはありません。パッケージのインストールおよびアップグレードには同じ RPM ファイルが使用されるため、パッケージの更新に特別なアップグレードファイルは必要ありません。

パッケージのクエリ

RPM はクエリのオプションを用意していますので、全パッケージまたは特定のファイルの検索を RPM データベース全体に対して行うことができます。また、任意のファイルの所属先またはパッケージの所属先を簡単に見つけることもできます。パッケージに含まれているファイルは圧縮されたアーカイブに入っています。そのカスタムバイナリヘッダーには役に立つパッケージの情報やその内容が含まれています。RPM によりヘッダーのクエリを素早くそして簡単に行うことができます。

システムの検証

パッケージを検証する機能も備わっています。パッケージに関連したファイルが削除されている可能性を懸念している場合、パッケージの検証を行いパッケージが提供するファイルの状態を確認することができます。この検証ではすべての異常が報告されます。エラーが存在する場合はそのファイルを簡単に再インストールすることができます。修正した設定ファイルは再インストールしても維持されます。

純粋なソース

RPM の重要な設計目標のひとつは、オリジナルのソフトウェア著者により配布された通りの純粋なソフトウェアソースを使用できるようにすることです。RPM を使用すると、純粋なソースに使

用されたパッチや構築方法に関する詳細な解説を付けてパッケージ化することができます。これは複数の理由で重要な利点となります。例えば、プログラムの新しいバージョンがリリースされた場合、コンパイルを完全に最初から始める必要はなく、パッチを見て何が**必要そうなのか**を判定することができます。ソフトウェアを正常に構築できるようにコンパイルされているデフォルト設定や変更が全てこの技術を使用すると簡単に確認することができます。

ソースを純粋に保持することは開発者以外の人にとっては重要には見えないかもしれませんが、純粋なソースの維持は高品質なソフトウェアにつながるためエンドユーザーにとっても重要なものとなります。

3.1.2. RHN RPM ガイドライン

RPM の長所は依存関係を定義し正確に競合を識別する能力にあります。Red Hat Network は RPM のこうした側面に依存しています。Red Hat Network は自動化した環境を提供することでパッケージインストール中に手動による介入が必要ないようにしています。そのため、Red Hat Network による配信用 RPM を構築する場合には次のルールに従うことが重要となります。

1. RPM についてよく理解しておいてください。パッケージを正しく構築するには、RPM の重要な機能について基本的に理解しておくことが大切になります。RPM に関する詳細については次のリソースをまずご覧ください。
 - http://docs.fedoraproject.org/en-US/Fedora_Draft_Documentation/0.1/html/RPM_Guide/index.html
 - http://docs.fedoraproject.org/en-US/Fedora_Draft_Documentation/0.1/html/Packagers_Guide/index.html
 - <http://www.gurulabs.com/GURULABS-RPM-LAB/GURULABS-RPM-GUIDE-v1.0.PDF>
2. 子チャンネル用に RPM を構築する場合は、子チャンネルのベースチャンネルと同じバージョンの Red Hat Enterprise Linux の新規インストール上でそのパッケージを構築します。最初に Red Hat Network からのすべての更新を適用することを忘れないようにしてください。
3. RPM パッケージは `--force` または `--nodeps` のオプションを使用しないでインストールしてください。ビルドシステムで RPM を正常にインストールできない場合、Red Hat Network でもその RPM を自動的にインストールすることはできません。
4. RPM パッケージのファイル名は **NVR** (名前、バージョン、リリース) 形式にしてください。また、パッケージのアーキテクチャも含ませる必要があります。**name-version-release.arch.rpm** が正しい形式となります。例えば、有効な RPM パッケージのファイル名が **pkgname-0.84-1.i386.rpm** とすると、パッケージ名は **pkgname**、バージョンは **0.84**、リリースは **1**、アーキテクチャは **i386** になります。
5. RPM パッケージはパッケージのメンテナにより署名が行われているはずですが、署名のないパッケージも Red Hat Network で配信することができますが、そのパッケージを受信するには **yum** に強制的に受理させる必要があります。このため、パッケージへの署名を強く推奨します。パッケージに署名する方法については「**RHN パッケージ用のデジタル署名**」に記載されています。
6. 署名が変更されたり再コンパイルが行われたなど、パッケージが何らかの形で変更された場合にはそのバージョンまたはリリースを増分させる必要があります。つまり、RHN から配信する各 RPM の **NVRA** (アーキテクチャを含む) は、混乱を避けるため固有のビルドにそれぞれ対応していなければなりません。
7. RPM パッケージがそれ自体を廃止予定にすることはできません。

8. 一つのパッケージを複数のパッケージに分割させる場合は、依存関係に十分に注意してください。どうしても必要な理由がない限り、既存のパッケージを分割するのは避けるようにしてください。
9. インタラクティブなプレインストール、ポストインストール、プレアンインストール、ポストアンインストールなどのスクリプトにパッケージを依存させることはできません。インストール中にユーザーによる直接介入を必要とする場合は **Red Hat Network** で動作させることはできません。
10. プレインストール、ポストインストール、プレアンインストール、ポストアンインストールなどのいずれのスクリプトにも `stderr` や `stdout` に書き込みを絶対行わせないようにしてください。メッセージが必要な場合は `/dev/null` にリダイレクトさせてください。これ以外はファイルに書き込みを行なわせてください。
11. `spec` ファイルを作成する場合は `/usr/share/doc/rpm-<version>/GROUPS` のグループ定義を使用します。完全に一致するものがない場合は二番目に適合するものを選択します。
12. RPM の依存性機能を利用して、プログラムの実行が必ずインストールの後に行われるようにします。



重要

ファイルをアーカイブしてからポストインストールのスクリプトでアンアーカイブする手順で RPM は作成しないでください。RPM の利点が失われます。

アーカイブ内のファイルがファイル一覧に含まれていないと、競合に関する検証や確認を行うことができません。大半の場合、RPM 自体が効率的にアーカイブを圧縮したり展開したりすることができます。例えば、`%postun` セクションでクリーンアップを行わない `%post` の中にはファイルを作成しないでください。

3.2. RHN パッケージ用のデジタル署名

RHN から配信されるパッケージにはすべて デジタル署名があるはずです。デジタル署名は固有となるプライベートキーで作成され、対応するパブリックキーを使って検証することができます。パッケージを作成したら、SRPM (ソース RPM) と RPM は GnuPG キーでデジタルに署名することができます。パッケージのインストールが行われる前に、信頼できる機関によってパッケージが署名されているか、また署名後にパッケージに変更が行われていないかがパブリックキーを使用して検証されます。

3.2.1. GnuPG キーペアを生成する

GnuPG キーペアはプライベートキーとパブリックキーで構成されます。以下のようにしてキーペアを生成します。

1. シェルプロンプトで `root` ユーザーになり次のコマンドを入力します。

```
gpg --gen-key
```

`root` 以外のユーザーでこのコマンドを実行すると次のようなメッセージが表示されます。

```
gpg: Warning: using insecure memory!
```

`root` ユーザー以外のユーザーはメモリページをロックできないためこのメッセージが表示されます。他の人には自分のプライベート GnuPG キーやパスフレーズを知られたくないわけですから、`root` になってからキーペアを生成します。`root` ユーザーはメモリページのロックがで

きるため、この情報がディスクに書き込まれることはありません。

- キーペア生成のコマンドを実行すると、次のようなキーペア生成に関してオプション選択を求める表示が出力されます。

```
gpg (GnuPG) 1.2.6; Copyright (C) 2004 Free Software
Foundation, Inc. This program comes with ABSOLUTELY NO
WARRANTY. This is free software, and you are welcome to
redistribute it under certain conditions. See the file COPYING
for details. Please select what kind of key you want: (1) DSA
and ElGamal (default) (2) DSA (sign only) (4) RSA (sign only)
Your selection?
```

- デフォルトのオプション **(1) DSA and ElGamal** を受領します。このオプションにより、2種類の技術を使ってデジタル署名の作成と暗号化/解読を行うことができるようになります。**1** と入力して **Enter** を押します。
- 次にキーの長さを指定するキーサイズを選択します。キーが長いほどユーザーのメッセージに対する攻撃への対抗力が増します。少なくとも **1024** ビットの長さのキーを作成することを推奨します。
- 次のオプションではキーの有効期間を指定するよう求めてきます。有効期限の日付を選択する場合には、そのパブリックキーを使用するユーザーにも有効期限の日付を知らせ、新しいパブリックキーを渡さなければならないので注意してください。期限は設定しないことを推奨します。期限を選択しない場合、その確認が求められます。

```
Key does not expire at all Is this correct (y/n)?
```

- y** を押して決定を確認します。
- 次にユーザー名、Email アドレス、オプションのコメントなどを含むユーザー ID を入力します。ひとつずつ別々に入力が求められます。終了すると入力した情報の要約が表示されます。
- 選択を承認したらパスフレーズを入力します。



注記

アカウントのパスワードと同様、GnuPG でも最善のセキュリティ対策には適切なパスフレーズが欠かせません。パスフレーズは大文字、小文字、数字を混ぜて句読点も入れると良いでしょう。

- パスフレーズの入力と確認を行うとキーが生成されます。次のようなメッセージが表示されま

```
We need to generate a lot of random bytes. It is a good idea to
perform some
other action (type on the keyboard, move the mouse, utilize the
disks)
during the prime generation; this gives the random number generator
a
better chance to gain enough entropy.
```

```
+++++.++++.+++++.....+++++.++++.+++++.+++++.++++.
+++++.....+++++
```

画面上の生成動作が完了すると、`root` のホームディレクトリ内の `.gnupg` ディレクトリ配下に新しいキーが配置されます。`root` ユーザーでキーペアを生成した場合、キーペアが配置されるデフォルトの場所になります。

`root` のキーを表示させるには、次のコマンドを使用します。

```
gpg --list-keys
```

次のような出力になります。

```
/root/.gnupg/pubring.gpg ----- pub 1024D/B7085C8A 2002-02-18
Your Name<you@example.com>
sub 1024g/E12AF9C4 2002-02-18
```

パブリックキーを検索する場合は次のコマンドを使用します。

```
gpg --export -a 'Your Name' > public_key.txt
```

パブリックキーが `public_key.txt` ファイル内に書き込まれます。

このパブリックキーは非常に重要となります。`up2date` でカスタムのソフトウェアを受信する全クライアントシステムに配備しなければならないキーになります。このキーを組織全体に配備するテクニックについては『Red Hat Network クライアント設定ガイド』で説明します。

3.2.2. パッケージに署名する

パッケージに署名を行なう前に、まず `~/rpmmacros` ファイルに次の行を含ませます。

```
%_signature gpg
%_gpg_name B7085C8A
```

`_gpg_name` キー ID 値の `B7085C8A` は、パッケージの署名に使用する GPG キーリングのキー ID に置き換えます。この値で RPM にどの署名を使用するのか指示します。

パッケージ `package-name-1.0-1.noarch.rpm` に署名を行ないます。次のコマンドを使用します。

```
rpm --resign package-name-1.0-1.noarch.rpm
```

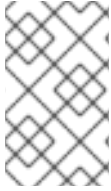
パスフレーズを入力します。パッケージが署名されていることを確認するには次のコマンドを使用します。

```
rpm --checksig -v package-name-1.0-1.noarch.rpm
```

Web 出力に **Good signature from "Your Name"** のフレーズが表示されるはずですが、`Your Name` は署名キーに関連付けした名前になります。

第4章 カスタムチャンネルとパッケージの管理

特定の組織や企業用に構築され保守が行われるパッケージの導入を管理者が Red Hat Network のインフラストラクチャを利用して行えるようにするのがカスタムのチャンネルです。チャンネルやパッケージの管理作業はすべて、RHN Web サイトの **チャンネル** タブで行います。使用方法については『RHN リファレンスガイド』にある RHN Web サイトの章をあわせてご覧ください。



注記

実稼働環境向けのテストを行っていないパッケージを導入すると問題が発生する可能性があるため、ステージング用に選択したシステム群を対象とするベータチャンネルをまず作成することを推奨しています。

例えば、複数の Web サーバー群で構成されるシステムグループがひとつあり、カスタムパッケージのセットを受信させたいとします。まず一時的なチャンネルを作って、このサーバーシステム群の代理となる重要性の低いシステムを集めた小グループにそのカスタムパッケージセットをインストールします。このシステム群は **実際に稼働中のライブシステムではなく、開発用またはステージング用のサーバー**になります。一時的なチャンネルは「**ソフトウェアチャンネルを削除する**」の手順に従い、後で削除してください。

4.1. チャンネル管理の特権

チャンネル管理の作業を行うには **チャンネル管理者**として正しいパーミッションを取得している必要があります。パーミッションは Red Hat Network Web サイトで変更することができます。パーミッションは最上位レベルの管理者となる **組織の管理者**によってユーザーに割り当てられます。チャンネル管理者の特権は次のように割り当てます。

1. Red Hat Network Web サイトに組織の管理者としてログインします。
2. 上部ナビゲーションバーで **ユーザー** タブをクリックしてチャンネル管理の機能を使用するユーザー名をクリックします。
3. **ユーザーの詳細** ページで **ロール** セクションまでスクロールして **チャンネル管理者** のラベルが付いたチェックボックスを選択します。ページ下部にある **サブミット (Submit)** をクリックします。組織の管理者には自動的にチャンネル管理者の特権が与えられています。
4. このユーザーで Red Hat Network Web サイトにログインし、上部ナビゲーションバーの **チャンネル** タブをクリックして **ソフトウェアチャンネルの管理** ボタンが左側のナビゲーションバーに表示されることを確認します。

4.2. ソフトウェアチャンネルの管理

標準の RHN Management レベルのユーザーが使用できるボタンやページの他、RHN Satellite Server と RHN Proxy Server をご利用のお客様は左側のナビゲーションバーにある **ソフトウェアチャンネルの管理** にもアクセスすることができます。このボタンをクリックすると **ソフトウェアチャンネル管理** のインターフェースが開きます。カスタムのソフトウェアチャンネル管理の作業はすべてここでを行います。



警告

RHN Proxy Server と RHN Satellite Server の両方を使用している場合、Proxy サーバー群は Satellite から直接更新を受信するためカスタムのチャンネルとパッケージの管理は Satellite 上でのみ行うようにしてください。任意の Proxy でパッケージやチャンネルの管理を手作業で行うとサーバー群が同期されない恐れがあります。

ソフトウェアチャンネル管理の一覧にあるリンクをクリックすると**管理しているソフトウェアチャンネルの詳細**ページの別のタブに移動します。チャンネル名をクリックすると**詳細**タブが開きます。パッケージ数をクリックすると**パッケージ**タブの**一覧表示/削除**サブタブが開きます。詳細については「[管理しているソフトウェアチャンネルの詳細](#)」を参照してください。

4.3. 管理しているソフトウェアチャンネルの詳細

実質的にはカスタムのチャンネル管理の作業はすべて**管理しているソフトウェアチャンネルの詳細**ページ内で行います。左側のナビゲーションバーにある**ソフトウェアチャンネルの管理**をクリックし、変更するチャンネル名を選択するとこのページにアクセスすることができます。このページは次のようなタブで構成されています。

- **詳細** – 親チャンネル、チャンネル名、要約、説明などのチャンネルに関する基本的な情報が表示されます。この情報の一部は変更が可能です。また、組織の管理者とチャンネル管理者からは**ユーザー毎のサブスクリプション制限**のコンボボックスが見えるようになっています。デフォルトの動作は、すべてのユーザーがシステムをこのチャンネルにサブスクライブさせることができるようになっています。このボックスのチェックを外して**チャンネルの更新**をクリックすると**サブスクライバー**タブが表示されます。このタブを使用して特定のユーザーにこのチャンネルへのサブスクリプション権限を与えます。
- **サブスクライバー** – カスタムチャンネルへのサブスクリプション権限を有しているユーザーの一覧を表示します。このタブは次の二つの条件が満たされると表示されます。一つ目の条件は、ログインしたユーザーが組織の管理者またはチャンネル管理者であることです。二つ目の条件は、**詳細**タブの**ユーザー毎のサブスクリプション制限**のコンボボックスが「**ご自分の組織の指定されたユーザーのみがこのチャンネルにサブスクライブできます。**」に設定されていることです。サブスクライバータブでは、システムをチャンネルにサブスクライブさせることができるユーザーを指定します。ユーザーのチェックボックスを選択して**更新 (Update)**をクリックすると、そのユーザーはシステムをチャンネルにサブスクライブさせることができますようになります。組織の管理者とチャンネル管理者は全チャンネルへのサブスクリプションアクセスが自動的に与えられています。
- **マネージャー** – カスタムチャンネルを管理する権限を有しているユーザーを一覧表示します。このタブが表示されるのは、組織の管理者およびチャンネル管理者です。ユーザーのチェックボックスを選択して**更新 (Update)**をクリックすると、そのユーザーにはこのチャンネルを管理するための全権限が許可されます。この権限では、新規のチャンネル作成はできません。組織の管理者とチャンネル管理者は全チャンネルへの管理アクセスが自動的に与えられています。
- **エラー** – 各カスタムチャンネルに関連するエラーを表示します。Red Hat Enterprise Linux ソフトウェアに対するエラー更新の生成および配信が Red Hat Network によって行なわれるのと同様、最新コードによるサーバー更新の一部として、企業独自のカスタムチャンネルに対するエラー更新の配信などはここで行ないます。このタブには、エラーの表示や追

加、削除、クローン作成などができる **一覧表示/削除**、**追加**、**クローン作成** などのサブタブが含まれています。エラータのクローン作成は RHN Satellite Server からしか行えませんので注意してください。

- **一覧表示/削除** – カスタムチャンネルに現在関連付けされている全てのエラータを表示し、その関連付けを取り消すことができます。チャンネルからエラータの関連付けを取り消すには、エラータのチェックボックスを選択してページ右下の **エラータの削除** をクリックします。取り消すエラータが一覧表示されている確認のページが表示されます。**確認** をクリックして取消しの作業を完了します。
- **追加** – チャンネルにエラータを追加することができます。チャンネルに適用できる可能性があるエラータがすべて表示されます。チャンネルにエラータを追加するには、該当のチェックボックスを選択して **エラータの追加** をクリックします。エラータの管理については [5章 カスタムのエラータ管理](#) を参照してください。
- **クローン作成** – Satellite を利用している場合、これを使用するとクローン作成したチャンネル用にエラータと関連のパッケージを複製することができます。このサブタブは、「オリジナルの状態」または「エラータ選択」いずれかのオプションでクローン作成したチャンネルに入力が行われると表示されます。ターゲットチャンネル(元となったチャンネル)に対してエラータが発行されると常に **クローン** タブもまたエラータを取得します。現在の状態のオプションでチャンネルをクローン作成した場合に便利です。クローン作成オプションの詳細については「[ソフトウェアチャンネルのクローンを作成する](#)」を参照してください。

クローン作成したチャンネルにターゲットチャンネルからのエラータを含ませるには、各アドバイザリのドロップダウンメニューから **マージ** または **クローン作成** のいずれかを選択します。マージオプションは、そのエラータのクローンが以前に作成されている場合にのみ表示されます。エラータのクローンの重複作成は避けながら、複数チャンネル全体に渡りそのエラータを関連付けたい場合にはマージを使用します。以前のクローンからエラータに修正を加える場合など、新規でエラータのクローンを作成する場合は、**クローン作成** オプションを使用します。

デフォルトでは、クローンのエラータはオリジナルの Red Hat アドバイザリのラベルを継承し、「RH」のプレフィックス部分が「CL」になります。例えば、RHSA-2003:324 なら CLSA-2003:324 になります。以降、同じアドバイザリのクローンは「CM」、「CN」という具合に二番目の文字でその作成順序が表されます。ラベルは **管理しているエラータの詳細** ページで変更することができます。詳細は「[管理しているエラータの詳細](#)」を参照してください。

エラータのクローンが以前に作成されている場合、そのエラータは **マージ** オプションの他、**所有するエラータ (Owned Errata)** というコラム内に値を含んでいます。エラータラベルは詳細ページにリンクしています。そのクローンのエラータは発行済みなのか、またはオリジナルのアドバイザリから変更が加えられたのかは、括弧で囲まれた **pub** と **mod** の各フラグで識別します。フラグの前にプラスの印 + があれば肯定を示し、そのクローンのエラータは発行されています。フラグの前のマイナスの印 - は否定を示します。例えば、**(-mod)** はパッケージが削除されたという意味になる場合があります。カスタムのエラータの発行および編集については「[エラータの管理](#)」を参照してください。

クローンのチャンネルからエラータを取り除く場合は、ドロップダウンメニューで **何も行わない** を選択します。変更を確認したら **エラータのクローン作成** をクリックします。確認のページで発生する変更を確認して **エラータの更新** をクリックします。

- **同期** – 最初にチャンネルのクローンを作成した時には含まれていず、その後の更新で追加されたエラータパッケージを表示します。このページでは、目的のエラータのチェックボックスに印を付け **エラータの同期** をクリックすることで、クローンのチャンネルを現

在のエラータで同期させることができます。

- **パッケージ** – 各カスタムチャンネルに関連付けされたパッケージを表示します。このタブにはパッケージの表示、追加、削除を行うことができる **一覧表示/削除**、**追加**、**比較**などのサブタブが含まれています。
 - **一覧表示/削除** – 現在、カスタムチャンネルに関連付けられている全パッケージを表示します。また、その関連性を取り消すこともできます。チャンネルからパッケージを削除する場合は、そのパッケージのチェックボックスを選択してページ右下の**パッケージの削除**をクリックします。削除対象のパッケージを一覧表示した確認のページが表示されず。**確認**をクリックして削除作業を完了します。



重要

この一覧は標準のソフトウェアチャンネルの**詳細**ページで使用できるパッケージ一覧とは異り、パッケージの最新バージョンだけでなくデータベースに残っている全バージョンを表示します。最新バージョンを削除すると旧バージョンに戻ることができます。

- **追加** – チャンネルにパッケージを追加することができます。利用可能なパッケージを表示するには **表示** ドロップダウンメニューから1つオプションを選択して**表示**をクリックします。編集集中のチャンネルにパッケージを追加する場合は、該当するチェックボックスを選択して**パッケージの追加**をクリックします。この手順に関する詳細は「[パッケージをソフトウェアチャンネルに割り当てる](#)」を参照してください。
- **比較** – 異なるチャンネル間でパッケージ一覧の比較ができます。違いを見るには、「**比較対照 (Compare to):**」ドロップダウンメニューからもう1つのチャンネルを選択して**比較 (Compare)**をクリックします。両方のチャンネルには含まれていないパッケージが全て表示され、それぞれの既存チャンネルの場所を示します。
- **リポジトリ** – **リポジトリの管理**を選択して、チャンネルに **yum** リポジトリを割り当てリポジトリの内容を同期します。
 - **追加 / 削除** – 設定したリポジトリの一覧を表示します。リポジトリ名の横にあるチェックボックスを選択してから **リポジトリの更新** をクリックするとリポジトリの追加や削除を行うことができます。
 - **同期** – 設定したリポジトリを一覧表示します。同期のスケジュールはドロップダウンボックスを使って設定できます。また、**今すぐ同期 (Sync Now)** をクリックすると直ちに同期を開始することもできます。

4.4. ソフトウェアパッケージの管理

チャンネル内でパッケージを追加したり、削除したりする他、データベースおよびファイルシステムの両方からパッケージを完全に削除するオプションがあります。ファイルシステムからの削除には一時間ほどの遅れがあります。**ソフトウェアパッケージ管理**ページで行うことができます。このページには左側のナビゲーションバーにある**ソフトウェアパッケージの管理**をクリックしてアクセスします。



警告

データベースからパッケージを削除した場合、再びパッケージのアップロードを行えば元に戻すことはできますが、エラーとの関連付けはすべて失われます。再ロードする際に手作業でエラーに関連付けし直す必要があります。詳細は [5章カスタムのエラータ管理](#) を参照してください。

データベースからパッケージを削除する

1. **ソフトウェアパッケージの管理** のページに行き、**表示** のドロップダウンメニューから削除するパッケージを含むオプションを選択し **表示** をクリックします。
2. 削除するパッケージのチェックボックスを選択して **パッケージの削除** をクリックします。選択したパッケージの一覧が記載された確認ページが表示されます。

実際のパッケージは **RHN Proxy Server** に保管してあるため、そのカスタムパッケージは表示されますが **RHN Web** サイトからのダウンロードはできません。カスタムパッケージを取得する場合は、**up2date** を使用してクライアントシステムから行う必要があります。一方、**RHN Satellite Server** はそれ自体の Web サイトを提供するため、カスタムパッケージには **HTTP** または **Red Hat Update Agent** 経由でアクセスすることができます。カスタムパッケージを取得するには、そのパッケージを含むチャンネルにクライアントシステムをサブスクライブさせる必要があります。

4.5. ソフトウェアチャンネルを作成する

パッケージをサーバーにアップロードする前に、パッケージを格納するカスタムチャンネルを作成します。詳細は [6章カスタムパッケージをアップロードして保守作業を行う](#) を参照してください。パッケージをアップロードしたら、「[パッケージをソフトウェアチャンネルに割り当てる](#)」に説明してあるように Web サイトから再割り当てをすることができます。

チャンネルは次のように **Red Hat Network Web** サイト上で作成します。

1. チャンネル管理者として **Red Hat Network Web** サイトにログインします。
2. 上部ナビゲーションバーで **チャンネル** タブをクリックしてから左側のナビゲーションバーにある **ソフトウェアチャンネルの管理** ボタンをクリックします。
3. **ソフトウェアチャンネル管理** ページで右上にある **新しいソフトウェアチャンネルの作成** をクリックします。 **RHN Satellite Server** の管理者には **チャンネルのクローン作成** のオプションが表示されます。詳細は「[ソフトウェアチャンネルのクローンを作成する](#)」を参照してください。
4. **新規チャンネル** ページ上でページの指示に従ってチャンネルの詳細を定義します。ほとんどのチャンネル管理作業で、チャンネルを識別する際は **チャンネルラベル** が使用されるため、ラベルにはわかりやすいラベル名を付けてください。既存のチャンネルの詳細を確認するとよいでしょう。

GPG キーの URL はサーバー上のキーの配置場所になります。この配置場所は、クライアント設定のプロセスで定義します。『**Red Hat Network クライアント設定ガイド**』を参照してください。**GPG キー ID** は「**DB42A60E**」など固有の識別子となります。**GPG キー** のフィンガーブ

リントは「CA20 8686 2BD6 9DFC 65F6 ECC4 2191 80CD DB42 A60E」などの文字列になります。キーIDはキーのフィンガープリント内の最後の8文字と同じである点に注意してください。

5. 終了したらページ下部の **チャンネルの作成** をクリックします。

4.6. パッケージをソフトウェアチャンネルに割り当てる

初めてパッケージをアップロードする場合、任意のカスタムチャンネルまたは複数のカスタムチャンネルに割り当てることができます。チャンネルへの割り当てを行わなくても構いません。詳細は [6章 カスタムパッケージをアップロードして保守作業を行う](#) を参照してください。アップロードが終了したら、カスタムチャンネル間で再割り当てを行ったり、割り当てチャンネル無しのリポジトリに再度割り当てたりすることもできます。

以下のようにして行ないます。

1. 上部ナビゲーションバーにある **チャンネル** タブ、次に左側のナビゲーションバーの **ソフトウェアチャンネルの管理** をクリックします。
2. **ソフトウェアチャンネル管理** ページで、パッケージを割り当てるチャンネルのチャンネル名をクリックします。
3. **管理しているソフトウェアチャンネルの詳細** ページで、**パッケージ** のタブ、次に **追加** のサブタブをクリックします。編集しているチャンネルにパッケージを割り当てるには、**表示** のドロップダウンメニューからそのパッケージを含むオプションを選択し **表示** をクリックします。



注記

編集しているチャンネルにすでに割り当てられているパッケージは表示されません。チャンネルに割り当てをしていないパッケージは **チャンネルに割り当てられていないパッケージ (Packages in no channels)** のメニューアイテムで確認できます。 **管理している全パッケージ (All managed packages)** を選択すると使用できる全パッケージが表示されます。

4. 編集しているチャンネルに割り当てるパッケージのチェックボックスを選択して、ページ右下にある **パッケージの追加** をクリックします。選択したパッケージが記載された確認ページが表示されます。
5. **確認** をクリックするとパッケージがチャンネルに割り当てられます。 **管理しているソフトウェアチャンネルの詳細** ページの **一覧表示/削除** のサブタブに新しいパッケージが表示されます。

パッケージがチャンネルに関連付けられると、エラータのキャッシュが更新されて変更が反映されます。全ての変更が使用可能になる前にユーザーがチャンネルの編集を終了できるよう更新は若干遅れます。キャッシュへの変更を手動で開始する場合は **一覧表示/削除** サブタブの上部にあるテキスト内の **直ちに変更をコミット** をクリックします。

4.7. ソフトウェアチャンネルのクローンを作成する

RHN Satellite Server のチャンネル管理者の場合、パッケージの割り当てを用意するためソフトウェアチャンネルのクローンを作成することもできます。クローンを作成することにより、チャンネルの完全な複製が可能になるため、パッケージとエラータのカスタムソフトウェアチャンネルへの割り当てが迅速に行なえるようになります。

次のようにして行ないます。

1. 上部ナビゲーションバーの **チャンネル** タブ、次に左側のナビゲーションバーの **ソフトウェアチャンネル管理** をクリックすると、**ソフトウェアチャンネル管理** のページに移動します。
2. 右上の **チャンネルのクローン作成 (clone channel)** をクリックします。

クローン作成のオプションが3つ表示されます。チャンネルの現在の状態、チャンネルのオリジナル状態、エラータ選択の3つです。詳細については **Web** ページに記載されていますが以下に要約を記します。

- **チャンネルの現在の状態 (Current state of the channel)** – 現在、ターゲットチャンネル(元となるチャンネル)にある最新の全パッケージと全エラータです。
 - **チャンネルのオリジナルの状態 (Original state of the channel)** – ターゲットチャンネルからの全オリジナルパッケージを含みます。ただし、エラータや関連の更新パッケージは含まれません。
 - **エラータ選択 (Select Errata)** – ターゲットチャンネルからの全オリジナルパッケージが含まれます。このオプションでは特定のエラータや関連の更新パッケージを除外することができます。
3. **クローン作成 (Clone)** フィールド内のラジオボタンを使用して該当オプションを選択します。**クローン作成元 (Clone From)** ドロップダウンメニューを使ってターゲットチャンネルを確定し、**チャンネルの作成** をクリックします。
 4. 「**ソフトウェアチャンネルを作成する**」で説明されている通り、**新規のソフトウェアチャンネル (New Software Channel)** ページのフィールドを入力していきます。デフォルトの値のままで充分でしょう。
 5. **チャンネルの作成** をクリックします。「オリジナルの状態」または「現在の状態」のいずれかを選択すると、**管理しているソフトウェアチャンネルの詳細** ページの **詳細** タブが表示されます。新しいチャンネルの設定を変更します。詳しくは、「**管理しているソフトウェアチャンネルの詳細**」を参照してください。

チャンネルのクローン作成に「エラータ選択」のオプションを選んだ場合は、**管理しているソフトウェアチャンネルの詳細** ページの **クローン (Clone)** サブタブに移動します。ここでクローン作成をして新しいチャンネルに含ませるエラータや関連パッケージを個別に選択することができます。詳細については「**管理しているソフトウェアチャンネルの詳細**」を参照してください。



注記

整合性を確保し再生可能なパッケージを維持するため、日付に基づいて全チャンネルのクローンを作成するコマンドがあります。**spacewalk-clone-by-date** というコマンドです。

4.8. ソフトウェアチャンネルを削除する

RHN Satellite Server や RHN Proxy Server の管理者の場合、未使用のチャンネルの削除を行なうことができます。チャンネル → **ソフトウェアチャンネルの管理** ページで行ないます。このページの右上にある **ソフトウェアチャンネルの削除** をクリックしてチャンネルを削除します。次のページで **チャンネルの削除** をクリックして動作を完了します。



注記

チャンネル → ソフトウェアチャンネルの管理 ページについては、「[管理しているソフトウェアチャンネルの詳細](#)」で詳しく説明しています。



重要

チャンネルを削除しても一緒にパッケージが削除されるわけではありません。RHN Satellite からパッケージを削除する場合は、「[ソフトウェアパッケージの管理](#)」を参照してください。

Web サイトでチャンネルの削除を行なう場合は、まずは次の点をよく検討してから行なうようにしてください。

- チャンネルを削除してもそのチャンネルのパッケージはサーバー上に残ります。チャンネル削除後にパッケージも削除する方法があります。
- チャンネルを削除すると、そのチャンネルに関連するエラータの行き所がなくなり孤立する可能性があります。
- Satellite サーバーでは、子チャンネルがある場合には親チャンネルの削除は行なわれません。先に子チャンネルを削除してから親チャンネルを削除するようにしてください。
- チャンネルを削除する前に、キックスタートディストリビューションの関連付けの解除、またはキックスタートディストリビューションの削除を先に行なってください。
- Proxy で作成したチャンネルを Satellite に接続する場合は、RHN Proxy Server 上のチャンネルを削除してください。

第5章 カスタムのエラータ管理

カスタムエラータを使用するとカスタムチャンネル内のパッケージにエラータアラートを発行することができるようになります。エラータの管理作業はすべて RHN Web サイトの **エラータ** タブで行います。管理作業については『Red Hat Network リファレンスガイド』の RHN Web サイトの章もあわせてご覧ください

5.1. エラータの管理

標準の RHN Management レベルのユーザーの方が使用できるボタンやページに加え、RHN Satellite Server および RHN Proxy Server をご利用のお客様は左側のナビゲーションバーにある **エラータの管理** もご利用頂けます。このボタンから **エラータ管理** のインターフェースを開き、カスタムエラータの管理作業を行います。



警告

RHN Proxy Server と RHN Satellite Server の両方を使用する場合、Proxy の役目は Satellite から更新を受信することですので、エラータの管理作業は Satellite 上でのみ行なうようにしてください。両方を使用する構成でエラータの管理作業を Proxy で行うと、サーバーが同期しなくなる恐れがあります。

エラータ管理 一覧内のアドバイザリをクリックすると **管理しているエラータの詳細** ページの **詳細** タブに移動します。この部分については「**管理しているエラータの詳細**」を参照してください。

5.1.1. 発行済みエラータ

発行済みエラータ ページは左側のナビゲーションバーの **エラータの管理** をクリックするとデフォルトで表示されます。独自に作成、発信しているエラータアラートを表示します。

既存の発行済みエラータを編集するには、「**エラータの作成と編集**」に記載されている手順に従います。このエラータを配信する場合は **エラータの詳細** ページの右上にある **通知の送信** をクリックします。エラータアラートが影響を受ける全システムの管理者に送信されます。

5.1.2. 未発行のエラータ

未発行のエラータ ページは左側のナビゲーションバー内の **エラータの管理** の下にある **未発行** をクリックすると表示されます。独自に作成したエラータアラートでまだ配信していないものが表示されます。

既存の未発行エラータを編集する場合は「**エラータの作成と編集**」に記載されている手順に従います。エラータを発行するには **エラータの詳細** ページの右上にある **エラータの発行** をクリックします。次にエラータに関連付けられているチャンネルを確認してから今度は右下にある **エラータの発行** ボタンをクリックします。エラータアラートが **発行済み** ページに移動して配信待ちになります。

5.2. 管理しているエラータの詳細

発行済み または **未発行** のページ内の管理エラータアラートのアドバイザーをクリックすると**管理しているエラータの詳細** ページが表示されます。このページはさらに**詳細**、**チャンネル**、**パッケージ** の3つのタブに分割されています。

- **詳細** – カスタムのエラータアラート作成中に入力した主要な情報を表示します。概要 (シノプシス)、アドバイザー名と種類、関連製品、バグ、詳細、ソリューション、キーワード、リファレンス、メモなどが含まれます。いずれかの情報を変更する場合は該当フィールドで変更を行ってから **エラータの更新** をクリックします。
- **チャンネル** – 選択したエラータに関連しているチャンネルを表示します。この関連付けを変更するには、適切なチェックボックスを選択または選択解除し、**チャンネルの更新** ボタンをクリックします。
- **パッケージ** – 選択したエラータに関連付けられたパッケージを管理します。このタブにはパッケージの表示、追加、削除ができる **一覧表示/削除** と **追加** の二つのサブタブが含まれています。
 - **一覧表示/削除** – カスタムのエラータに現在関連付けられている全パッケージを表示します。また、その関連性を取り消すこともできます。エラータからパッケージを削除する場合は、パッケージのチェックボックスを選択してページ右下の **パッケージの削除** をクリックします。削除するパッケージの一覧が記載された確認のページが表示されます。**確認** をクリックして削除の作業を完了します。
 - **追加** – エラータにパッケージを追加することができます。利用できるパッケージを表示するには **表示** のドロップダウンメニューからオプションを1つ選び **表示** をクリックします。パッケージを編集中的エラータに追加する場合は、該当するチェックボックスを選択してから **パッケージの追加** をクリックします。この手順の詳細については「**パッケージをエラータに割り当てる**」を参照してください。

5.3. エラータの作成と編集

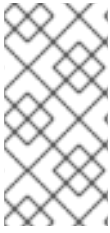
手順に従ってカスタムのエラータアラートを作成します。

1. 上部のナビゲーションバーで **エラータ** をクリックしてから左側のナビゲーションバーで **エラータの管理** をクリックします。 **エラータの管理** ページから **新規エラータの作成** をクリックします。
2. **アドバイザー** フィールドでエラータ用のわかりやすいラベルを入力します。その企業や組織内で採用している命名規則などに従うのがよいでしょう。カスタムのエラータと Red Hat から発行されるエラータとを混同しないようラベルの先頭に「RH」を付けないようにしてください (大文字小文字を問わず)。
3. 次に、残りの必須フィールドをすべて入力してから **エラータの作成** ボタンをクリックします。Red Hat の標準的なエラータを参考にして適切にフィールド入力されているか確認してください。

RHN Satellite Server 管理者はエラータを作成する場合に既存エラータのクローン作成を使用することもできます。エラータをクローン作成するとパッケージの関連付けが維持されるためエラータの発行が容易になります。詳細については「**エラータのクローンを作成する**」を参照してください。

既存のエラータアラートの詳細を編集するには、**エラータ管理** ページのアドバイザーをクリックして **詳細** タブの該当フィールド内で変更を加え **エラータ更新** ボタンをクリックします。エラータのチャンネル関連付けを変更する場合は **チャンネル** タブをクリックします。そのパッケージを表示させ変更を行う場合は **パッケージ** タブをクリックしてします。

エラータを削除する場合は、**エラータ管理** ページ内のチェックボックスを選択し **エラータの削除** ボタンをクリックしてから削除の確認を行います。発行済みのエラータの削除には数分かかることがあります。



注記

システムにエラータアラートが発行されたときに **Email** を受信したい場合は、**RHN Web** サイトで **ユーザーの RHN => 個人設定** と進み **電子メールの通知を受け取る** を選択します。システムをサブスクライブさせている各システムの管理者に便利な設定となります。

5.4. パッケージをエラータに割り当てる

次の手順に従ってパッケージをエラータに割り当てます。

1. 編集するエラータを選択したら **パッケージ** タブをクリックしてから **追加** のサブタブをクリックします。
2. パッケージを編集中のエラータに関連付けるため、関連付けを行いたいパッケージを持っているチャンネルを **表示** ドロップダウンメニューから選択して **表示** をクリックします。編集中的エラータに既に関連付けされているパッケージは表示されません。 **管理している全パッケージ (All managed packages)** を選択すると利用できるパッケージが全て表示されます。
3. **表示** をクリックすると選択したオプションのパッケージ一覧が表示されます。ページヘッダーはまだ編集中的エラータを表示しているので注意してください。
4. この一覧で編集中的エラータに割り当てるパッケージのチェックボックスを選択し、ページの右下にある **パッケージの追加** をクリックします。
5. パッケージ一覧が記載された確認ページが表示されます。 **確認** をクリックしてパッケージをエラータに関連付けます。 **管理しているエラータの詳細** ページの **一覧表示/削除** サブタブに新しいパッケージ一覧が表示されます。

エラータへのパッケージの割り当てが完了すると、エラータキャッシュが更新され変更が反映されます。全ての変更が使用可能になる前にユーザーがチャンネルの編集を終了できるよう更新は若干遅れます。手動でキャッシュの変更を開始するには、ページ上部の **直ちに変更をコミット** の指示に従います。

5.5. エラータのクローンを作成する

RHN Satellite Server の一部として、エラータの複製をクローン作成し配信することができます。クローン作成できるのは、いずれかのチャンネルに適用できる可能性があるエラータのみです。任意のエラータを適用しているチャンネルからクローンチャンネルを作成した場合、そのエラータは作成したクローンのチャンネルにも適用可能となります。エラータのクローン作成機能を利用するには、上部ナビゲーションバーの **エラータ** をクリックし左側のナビゲーションバーの **エラータのクローン作成** をクリックします。このボタンは RHN Satellite Server ご利用のユーザーにしか表示されません。

エラータのクローン作成 ページに行き、**表示** のドロップダウンメニューからエラータを含むチャンネルを選択して **表示** をクリックします。エラータ一覧が表示されたらクローン作成するエラータのチェックボックスを選択して **エラータのクローン作成** をクリックします。エラータの一覧が記載された確認ページが表示されます。 **確認** をクリックしてクローンの作成を終了します。

クローンのエラータは未発行のエラータ一覧に表示されます。ここでエラータのテキストや関連付けられているパッケージを確認することができます。準備が整ったらそのエラータを発行して該当するユーザーが利用できるようにします。

第6章 カスタムパッケージをアップロードして保守作業を行う

プライベートチャンネルへのパッケージのアップロードには2通りの仕組みがあり、使用している Red Hat Network サービスによって異なります。

RHN Proxy Server をご利用のお客様は **RHN Package Manager** アプリケーションを使用して頂くこととなります。パッケージヘッダー情報を中央の Red Hat Network サーバーに送り、**RHN Package Manager** を起動した Proxy のローカルリポジトリにそのパッケージ自体を格納します。

RHN Satellite Server をご利用のお客様は **RHN Push** アプリケーションを使用して頂くこととなります。パッケージのヘッダー情報をローカルの RHN Satellite Server に送り、**RHN Push** を起動した Satellite のローカルレポジトリにそのパッケージを格納します。

本章ではこれら両方のツールについて詳しく見ていくことにします。



警告

RHN Proxy Server と RHN Satellite Server の両方を使用している場合は、**RHN Push** のみを使用するようにしてください。Proxy と Satellite を組み合わせて使用している場合には、カスタムパッケージおよびチャンネルのアップロードは Satellite に対してのみ行います。Proxy サーバーはここからパッケージを取得してクライアントシステム群に配信することになります。

6.1. パッケージを RHN PROXY SERVER にアップロードする

RHN Package Manager を使用すると、RHN Proxy Server 経由でプライベートの RHN チャンネルに割り当てられたカスタムパッケージを提供することができます。RHN Proxy Server で提供するパッケージを Red Hat Enterprise Linux の公式パッケージのみにする場合は、**RHN Package Manager** をインストールしないようにしてください。

RHN Package Manager を使用する場合は、**rhns-proxy-package-manager** RPM パッケージとその依存パッケージをインストールします。このパッケージは登録している RHN Proxy Server のシステム群で使用することができます。**up2date rhns-proxy-package-manager** を実行してインストールを行います。



注記

RHN サーバーにアップロードされるのはパッケージのヘッダー情報のみです。RHN にクライアントシステム群のパッケージ依存関係を解決させるためにこのヘッダー情報が必要となります。実際のパッケージファイルは (***.rpm**) RHN Proxy Server に収納されます。このため、カスタムパッケージは RHN ウェブサイトに表示されていてもダウンロードすることはできません。クライアントシステムにカスタムパッケージを取得させる場合は **up2date** を使用してください。

6.1.1. RHN Package Manager を設定して使用する

RHN Package Manager を使ってパッケージを RHN にアップロードする前に、まずそのパッケージを手作業で Proxy サーバー自体にコピーする必要があります。例えば、開発ホストから **scp** を使用します。

```
scp foo.rpm root@rhnpool.example.com:/tmp
```

RHN Package Manager を使用して Red Hat Network にパッケージをアップロードする場合、前のステップでサーバーにコピーしたファイルをポイントします。



注記

クライアントシステムにパッケージを取得させるにはチャンネルが必要になります。このため、Red Hat Network にカスタムパッケージをアップロードする前に、まず、そのカスタムパッケージを割り当てるプライベートチャンネルを少なくとも1つ作成しておきます。

次のコマンドにより、パッケージヘッダーを RHN サーバーにアップロードされ、パッケージ自体は RHN Proxy Server のリポジトリにコピーされます。

```
rhn_package_manager -c label_of_private_channel pkg-list
```

`label_of_private_channel` は、パッケージの割り当て用に作成したカスタムのチャンネルです。作成時に指定した正確なチャンネルラベルを必ず使用してください。チャンネルをひとつまたは複数指定すると (`-c` または `--channel` を使用)、アップロードするパッケージのヘッダーは指定した全チャンネルにリンクされます。チャンネルを指定しないと、そのパッケージは [パッケージの管理](#) ページの [チャンネルがありません](#) のセクションに置かれます。パッケージを再割り当てする方法については「[パッケージをソフトウェアチャンネルに割り当てる](#)」を参照してください。

`pkg-list` はアップロードするパッケージの一覧を表示します。これらのパッケージは既に Proxy ホストに物理的にコピーされていなければなりません。別の方法として、`-d` オプションを使用してチャンネルに追加するパッケージを含んだローカルのディレクトリを指定することもできます。RHN Package Manager は標準入力からパッケージの一覧を読み取ることもできます (`-stdin` を使用)。

Red Hat Network サーバーの URL、HTTP プロキシのユーザー名とパスワード (HTTP プロキシに認証を必要とする場合)、パッケージが存在する上部ディレクトリなど、その他のオプションは設定ファイルに指定されています。この特殊な設定は [編集しないでください](#)。またこの設定は `/etc/rhn/default/rhn_proxy_package_manager.conf` に格納されています。このデフォルト設定ファイル内に指定されているオプションの値は、メインの設定ファイル `/etc/rhn/rhn.conf` の設定値や RHN Package Manager に渡すコマンドラインオプションなどで上書きすることができます。

このファイルに設定されていないパラメータについては、現在ログインしているユーザーのホームディレクトリにある `.rhn_package_manager` から読み込まれます。また、ここにもない場合は最終的には `/etc/rhn/rhn_package_manager.conf` から読み込まれます。これらのファイルが他の人から読み取られないよう必ず適切なパーミッションを持たせるようにしてください。

パッケージをアップロードしたらローカルディレクトリが RHN サーバーのチャンネルイメージと同期しているかどうか確認します。

```
rhn_package_manager -s -c name_of_private_channel
```

この `-s` オプションを使用することで、不足している全パッケージが一覧表示されます。このパッケージは、RHN サーバーにはアップロードされているけれどローカルのディレクトリにはないパッケージになります。このオプションを使用する場合は組織の管理者になる必要があります。アプリケーションにより RHN のユーザー名とパスワードの入力が求められます。

--copyonly オプションは引数に記載されているファイルを **Satellite** にはアップロードせずに指定のチャンネルにコピーします。RHN Proxy Server 上のチャンネルにパッケージが1つ不足しているけれど、このチャンネル内の全パッケージ群の再インポートは行ないたくない場合に便利です。

```
rhn_package_manager -c channel-name --copyonly /path/to/missing/file
```

パッケージ一覧は RHN サーバーに保存してあるため **RHN Package Manager** を使用してもチャンネル内のパッケージ一覧を検索することができます。

```
rhn_package_manager -l -c name_of_private_channel
```

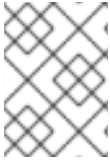
-l オプションを使用すると指定したチャンネル内にある各パッケージのパッケージ名、バージョン番号、リリース番号、アーキテクチャ、チャンネル名を表示します。他のオプションについては表 6.1 「**rhn_package_manager** のオプション」を参照してください。

RHN Package Manager (rhn_package_manager) の全コマンドラインオプションの要約については、表 6.1 「**rhn_package_manager** のオプション」をご覧ください。

表 6.1 **rhn_package_manager** のオプション

オプション	説明
-v, --verbose	標準の出力メッセージの詳細レベルが冗長になります。
-d, --dir DIRECTORY_NAME	このディレクトリからのパッケージを処理します。
-c, --channel CHANNEL_NAME	パッケージを検索するチャンネルを指定します。-c を複数回使用すると複数のチャンネルを指定することができます (例、 -c channel_one -c channel_two)。
-n, --count NUMBER	呼び出しごとに処理するヘッダー数です。デフォルトは 32 です。
-l, --list	指定したチャンネルのパッケージを一覧表示します。
-s, --sync	ローカルのディレクトリがサーバーと同期しているかどうか確認します。
-p, --printconf	現在の設定を表示して終了します。

オプション	説明
--newest	サーバーにあるパッケージより新しいパッケージのみをプッシュします。ソースパッケージはバージョン同士の比較が行われないという点で特殊となります。新しいかどうかの定義は関連するバイナリパッケージに依存します。このオプションを RHN Package Manager とソースパッケージだけで使用すると、パッケージのアップロードは行なわれますが、関連バイナリパッケージがアップロードされるまでソースパッケージは RHN Web インターフェースに表示されません。 --source と比較して見てください。 --source --newest を一緒に使用すると、単独ソースパッケージが新しいパッケージで 更新される ので、関連バイナリパッケージを先にアップロードしておく必要はありません。
--source	指示されたソースパッケージをアップロードします。この場合、ソースパッケージはプレーンで単独のパッケージとして扱われ、別途に既存するバイナリパッケージと関連する特殊なソースパッケージとしては 扱われません 。例えば、通常のソース制御管理とは別に、開発者やテスターに向けてアプリケーションソースを配布したい場合などに使用できます。
--stdin	標準出力からパッケージ名を読み込みます。
--nosig	パッケージに署名がない場合も失敗しません。
--no-ssl	SSL をオフにします (推奨できません)。
--stdin	標準出力からパッケージ名を読み込みます。
--username USERNAME	RHN のユーザー名を指定します。指定しないと有効なチャンネル管理者のユーザー名の入力が必要です。
--password PASSWORD	RHN パスワードを指定します。指定しないと有効なチャンネル管理者のパスワードの入力が必要です。
--dontcopy	アップロード後の手順でパッケージをパッケージツリー内の最終的な格納場所にはコピーしません。
--copyonly	パッケージのコピーのみを行い再インポートは行いません。
--test	プッシュするパッケージの一覧を出力するだけです。
-, --help	オプションの一覧とヘルプの画面を表示します。
--usage	使用可能なオプションの簡単な説明を表示します。
--copyonly	パッケージのコピーのみを行います。



注記

これらのコマンドラインオプションは `rhn_package_manager` の man ページ (`man rhn_package_manager`) でも説明されています。

6.2. パッケージを RHN SATELLITE SERVER にアップロードする

RHN Push アプリケーションを使用すると、RHN Satellite Server 経由でプライベートの RHN チャンネルに割り当てられたカスタムのパッケージを提供することができます。RHN Satellite Server で提供するパッケージを Red Hat Enterprise Linux の公式パッケージのみにする場合は、**RHN Push** をインストールしないようにしてください。

RHN Push を使用する場合は、`rhnpush` パッケージとその依存パッケージをインストールします。このパッケージは登録している RHN Satellite Server のシステム群で使用することができます。

`up2date rhnpush` を実行してインストールを行ないます。

RHN Push により RPM ヘッダー情報が RHN Satellite Server のデータベースにアップロードされ、その RPM は RHN Satellite Server のパッケージリポジトリに格納されます。RHN Proxy Server の **RHN Package Manager** とは異なり、**RHN Push** はパッケージ情報だけでなくヘッダーであっても RHN Satellite Server のデータベースから外部へは配布することはありません。



注記

Satellite のインストールで Solaris OS のシステムのサポートを有効にしている場合は、Solaris クライアントから RHN Push を使用して Solaris パッケージのコンテンツを Solaris のカスタムチャンネルにアップロードすることができます。

6.2.1. RHN Push アプリケーションを設定する

RHN Push のインストール時に中央設定ファイルが `/etc/sysconfig/rhn/rhnpushrc` にインストールされます。このファイルには表6.2「`rhnpush` オプション」に記載されている全てのオプションの値が含まれています。

`rhnpush` コマンドを発行するディレクトリに応じて設定を変更する場合、複数の異なる設定ファイルがあると便利です。現在のディレクトリ内の設定値 (`./rhnpushrc`) は、ユーザーのホームディレクトリ内の設定値 (`~/rhnpushrc`) より優先され、中央設定ファイル (`/etc/sysconfig/rhn/rhnpushrc`) 内の設定値より先に使用されます。

たとえば、現在のディレクトリの設定ファイルは、以下を指定する場合に使用します。

- 移植するソフトウェアチャンネル
- 呼び出すユーザー名を含ませるためのホームディレクトリの設定ファイル
- パッケージを受け取るサーバーを識別するための中央設定ファイル

`rhnpush` コマンドの全コマンドラインオプションを表6.2「`rhnpush` オプション」に示します。

表6.2 `rhnpush` オプション

オプション	説明
-------	----

オプション	説明
-v --verbose	詳細レベルが冗長になります。 -vv 、 -vvv といった具合にオプションは複数回使用することができます。
-d, --dir DIRECTORY	このディレクトリからのパッケージを処理します。
-c, --channel=CHANNEL_LABEL	パッケージを受け取るチャンネルを指定します。チャンネルラベルの指定は必須となります。チャンネルラベルはチャンネル名とは異なります。 -c を複数回使用することで複数のチャンネルを指定することができます (例、 -c=CHANNEL_ONE -c=CHANNEL_TWO)。
-n, --count N_HEADERS_PER_CALL	呼び出しごとに処理するヘッダー数です。整数にしてください。デフォルトは 25 です。
-l, --list	指定したチャンネルのみを表示します。
-r, --reldirRELATIVE_DIRECTORY	各ファイルにこの相対ディレクトリを関連付けます。
-o, --orgidORGANIZATION_ID	組織や企業の ID 番号を含めます。整数にしてください。
-u, --username USERNAME	指定したチャンネルに管理アクセス権を持つユーザーの RHN ユーザー名を含ませます。ユーザー名を与えないと rhnpush により有効なチャンネル管理者のユーザー名の入力が求められます。ユーザー名とパスワードは一定期間 ~/ .rhnpushcache にキャッシュされます。デフォルトは 5 分です。新しいユーザー名とパスワードを強制する場合は --new-cache を使用します。
-p, --password PASSWORD	指定したチャンネルに管理アクセス権を持つユーザーの RHN パスワードを含ませます。パスワードを与えないと rhnpush により有効なチャンネル管理者のパスワードの入力が求められます。ユーザー名とパスワードは一定期間 ~/ .rhnpushcache にキャッシュされます。デフォルト値は 5 分です。新しいユーザー名とパスワードを強制する場合は --new-cache を使用します。
-s, --stdin	標準入力からパッケージ一覧を読み込みます。例えば、パイプされた ls コマンドなど。
-X, --exclude GLOB	この glob 式と一致するパッケージを除きます。
--force	チャンネル内に現在、その名前とバージョンのパッケージが存在する場合でもパッケージのアップロードを強制します。このオプションを付けないと既存しているパッケージのアップロードはエラーを返すことになります。
--nosig	パッケージに署名がない場合も失敗しません。

オプション	説明
--new-cache	RHN Push にキャッシュされているユーザー名とパスワードを破棄させ、新しいユーザー名とパスワードを受け取るかその入力を求めるよう強制します。最初にユーザー名とパスワードの入力を間違えた場合に便利です。
--newest	サーバーにあるパッケージバージョンより新しいバージョンのパッケージのみをプッシュします。ソースパッケージはバージョン同士の比較が行われないという点で特殊となります。新しいかどうかの定義は関連するバイナリパッケージに依存します。このオプションを RHN Push とソースパッケージだけで使用すると、パッケージのアップロードは行なわれますが、関連バイナリパッケージがアップロードされるまでソースパッケージは RHN Web インターフェースに表示されません。 --source と比較してみてください。 --source --newest を一緒に使用すると、単独ソースパッケージが新しいパッケージで 更新される ので、関連バイナリパッケージを先にアップロードしておく必要はありません。
--header	ヘッダーのみをアップロードします。
--source	指示されたソースパッケージをアップロードします。この場合、ソースパッケージはプレーンで単独のパッケージとして扱われ、別途に既存するバイナリパッケージと関連する特殊なソースパッケージとしては 扱われません 。例えば、通常のソース制御管理とは別に、開発者やテスターに向けてアプリケーションソースを配布したい場合などに使用できます。
--server SERVER	パッケージのアップロード先となるサーバーを指定します。現在、 http://localhost/APP の値が必要です。このパラメータは必須です。
--test	プッシュするパッケージの一覧のみを表示し、実際のプッシュは行いません。
-h, --help	オプションの簡単な説明を表示します。
-, --usage	使用法の要約を表示します。



注記

これらのコマンドラインオプションについては **rhnpush** の **man** ページ (**man rhnpush**) でも説明されています。

6.2.2. RHN Push アプリケーションを使用する



注記

クライアントシステムにパッケージを取得させるにはチャンネルが必要になります。このため、カスタムパッケージをアップロードする前に、まず、そのカスタムパッケージを割り当てるプライベートチャンネルを少なくとも1つ作成しておきます。

次のコマンドでパッケージヘッダーを **RHN Satellite Server** にアップロードし、パッケージを **RHN Satellite Server** のパッケージリポジトリにコピーします。

```
rhnpush -c label_of_private_channel pkg-list
```

RHN Push 設定ファイルの値は、コマンドラインでオプションと値を指定すると上書きすることができます。

```
rhnpush -c label_of_private_channel --server=localhost pkg-list
```

label_of_private_channel は、パッケージの割り当て用に作成したカスタムのチャンネルです。作成時に指定した正確なチャンネルラベルを必ず使用してください。チャンネルをひとつまたは複数指定すると (**-c** または **--channel** を使用)、アップロードするパッケージのヘッダーは指定した全チャンネルにリンクされます。チャンネルを指定しないと、そのパッケージは **パッケージの管理** ページの **チャンネルがありません** のセクションに置かれます。パッケージを再割り当てする方法については「[パッケージをソフトウェアチャンネルに割り当てる](#)」を参照してください。

--server オプションはパッケージのインストール先となるサーバーを指定するため必須となります。**RHN Push** は外部のシステムにインストールしても構いませんが、**RHN Push** の実行は **RHN Satellite Server** 上でローカルに行なうことをお勧めします。

pkg-list はアップロードするパッケージの一覧です。別の方法として、**-d** オプションを使用してチャンネルに追加するパッケージを含んだローカルのディレクトリを指定することもできます。**RHN Push** は標準入力からパッケージ一覧を読み込むこともできます (**--stdin** を使用)。

付録A 改訂履歴

改訂 8-4.2.400 Rebuild with publican 4.0.0	2013-10-31	Rüdiger Landmann
改訂 8-4.2 翻訳および査読構成を完了	Thu Apr 18 2013	Noriko Mizumoto
改訂 8-4.1 翻訳ファイルをXMLソースファイル8-4版と同期	Fri Jan 4 2013	Terry Chuang
改訂 8-4.1 翻訳ファイルをXMLソースファイル8-4版と同期	Wed Nov 21 2012	Terry Chuang
改訂 8-4 5.5 向け最終パッケージ	Wed Sept 19 2012	Dan Macpherson
改訂 8-3 BZ#768267 セクション 4.8 に重要な記述を追加	Fri Aug 17 2012	Athene Chan
改訂 8-2 若干の編集	Fri Aug 17 2012	Athene Chan
改訂 8-1 BZ#768267 「ソフトウェアチャンネルを削除する」のセクションの記載内容を編集	Fri Aug 17 2012	Athene Chan
改訂 8-0 RHN Satellite 5.5 のリリース用に第1章から4章までを準備 RHN Satellite 5.5 のリリース用に第5章から6章までを準備 BZ#768267 チャンネルの削除と、チャンネル削除した場合のパッケージに対する影響に関する記載内容を修正 テクニカルレビューによる編集を適用	Tue June 26 2012	Athene Chan
改訂 7-0 BZ#798415 RPM のリンクを更新	Thu May 24 2012	Athene Chan
改訂 6-3 z-stream リリースの変更を y-stream に適用	Mon Aug 15 2011	Lana Brindley
改訂 6-2 リリース用の準備	Wed Jun 15 2011	Lana Brindley
改訂 6-1 翻訳者からの修正	Fri May 27 2011	Lana Brindley
改訂 6-0 翻訳の準備	Fri May 6 2011	Lana Brindley
改訂 5-8 BZ#701846 - QE レビュー	Thu May 5 2011	Lana Brindley
改訂 5-7 BZ#637722 - QE レビュー	Wed April 27 2011	Lana Brindley
改訂 5-6 BZ#637722 - チャンネルとパッケージの管理 BZ#679529 - パッケージのメンテナンス	Thu March 24 2011	Lana Brindley

索引

シンボル

とは

[RPM の利点](#), [RPM の利点](#)

エラータの管理

[詳細を表示する](#), [管理しているエラータの詳細](#)

エラータアラート

[クローンを作成する](#), [エラータのクローンを作成する](#)

[作成と編集](#), [エラータの作成と編集](#)

[未発行の管理](#), [未発行のエラータ](#)

[発行済みの管理](#), [発行済みエラータ](#)

[管理する](#), [カスタムのエラータ管理](#)

カスタムのパッケージ, [カスタムのパッケージを構築する](#)

[RHN Satellite Server にアップロード](#), [パッケージを RHN Proxy Server にアップロードする](#), [パッケージを RHN Satellite Server にアップロードする](#)

構築

[ガイドライン](#), [RHN RPM ガイドライン](#)

[構築する](#), [Red Hat Network のパッケージを構築する](#)

[署名する](#), [パッケージに署名する](#)

ソフトウェア

[チャンネル管理](#), [管理しているソフトウェアチャンネルの詳細](#)

チャンネル

とは, [RHN チャンネルとは](#)

[クローンを作成する](#), [ソフトウェアチャンネルのクローンを作成する](#)

[削除する](#), [ソフトウェアチャンネルを削除する](#)

パッケージ

[Solaris と UNIX](#), [パッケージを RHN Satellite Server にアップロードする](#)

[パッケージのアップロード](#), [カスタムパッケージをアップロードして保守作業を行う](#)

方法

[GnuPG キーの作成](#), [GnuPG キーペアを生成する](#)

[RHN Package Manager の設定](#), [RHN Package Manager を設定して使用する](#)

[RHN Push の設定](#), [RHN Push アプリケーションを設定する](#)

[RPM 以外のパッケージの配信](#), [パッケージを RHN Satellite Server にアップロードする](#)

[カスタムパッケージの構築](#), [Red Hat Network のパッケージを構築する](#)

[チャンネルのクローンを作成する](#), [ソフトウェアチャンネルのクローンを作成する](#)

[チャンネルパッケージ一覧を検索](#), [RHN Package Manager を設定して使用する](#)

パッケージを **RHN Proxy Server** にアップロード, **パッケージを RHN Proxy Server にアップロードする**

不足しているパッケージを **Satellite** にコピー, **RHN Package Manager** を設定して使用する

管理しているソフトウェアチャンネル

詳細, **管理しているソフトウェアチャンネルの詳細**

管理しているチャンネルの詳細 (**Managed Channel Details**), **管理しているソフトウェアチャンネルの詳細**

G

GnuPG キー

を使ってパッケージに署名する, **パッケージに署名する**
作成する, **GnuPG** キーペアを生成する

gpg キー, **GnuPG** キーペアを生成する

R

RHN Package Manager, **パッケージを RHN Proxy Server にアップロードする**

rhn_package_manager, **RHN Package Manager** を設定して使用する

インストールする, **パッケージを RHN Proxy Server にアップロードする**

チャンネル、指定する, **RHN Package Manager** を設定して使用する

チャンネルパッケージ一覧を検索, **RHN Package Manager** を設定して使用する

パッケージのヘッダーをアップロード, **RHN Package Manager** を設定して使用する

ローカルのパッケージ一覧の確認, **RHN Package Manager** を設定して使用する

不足しているパッケージを **Satellite** にコピー, **RHN Package Manager** を設定して使用する

設定する, **RHN Push** アプリケーションを設定する

設定ファイル, **RHN Package Manager** を設定して使用する

RHN Push

インストールする, **パッケージを RHN Satellite Server にアップロードする**

チャンネル、指定する, **RHN Push** アプリケーションを使用する

使用する, **RHN Push** アプリケーションを使用する

rhn_package_manager, **RHN Package Manager** を設定して使用する

(参照 **RHN Package Manager**)

コマンドラインオプション, **RHN Package Manager** を設定して使用する

rhn_package_manager.conf, **RHN Package Manager** を設定して使用する

RPM

の利点, **RPM** の利点

RPM パッケージマネージャ (参照 **RPM**)

W

Web サイト

ソフトウェアチャンネルの管理, [管理しているソフトウェアチャンネルの詳細](#)