



Red Hat Enterprise Linux 8

新世代エンタープライズOSの概要をつかむ

Red Hat Enterprise Linuxは、米Red Hat社が開発・販売する有償Linuxディストリビューションです。2019年5月には、5年ぶりのメジャーバージョンである「Red Hat Enterprise Linux 8」の出荷が始まりました。Red Hat Enterprise 8の特徴や、前バージョンとの違いなどをRed Hat社の国内法人であるレッドハットの技術者が解説します。



Part1 概要編	p. 3
Part2 特徴編	p. 5
Part3 RHEL7ユーザーへの注意編	p.11
もっと詳しく知りたい人は	p.11

執筆

レッドハットソリューションアーキテクト 森若 和雄

Part1 概要編



Red Hat Enterprise Linuxとは？

「Red Hat Enterprise Linux」(以下、RHEL。RHELは「レル」などと読む)は、各種のオープンソースソフトウェア(OSS)を統合して作成したLinuxディストリビューションの一種です。2002年2月に最初のバージョンに相当する「Red Hat Linux Advanced Server 2.1」が出て以来17年ほど開発・販売が続いている製品です。2019年5月7日には最新メジャーバージョンである「Red Hat Enterprise Linux 8」(以下、RHEL8)の出荷が始まりました。

RHELは、主に企業向けのサーバー用途でシェアが高いOS製品です。RHELを開発・販売する米Red Hat社は、サーバー用OS市場において米Microsoft社に次いで2位のシェアを占めています。

Red Hat社は、RHELの品質保証および各メジャーバージョンについての10年以上にわたる長期間のメンテナンスや、24時間365日のサポート、公式トレーニングと認定資格制度、コンサルティングサービスなどを提供しています。

例えば、Red Hat社には専門のセキュリティチームが存在し、製品に見つかったセキュリティ上の脆弱性の修正やアナウンスの体制を整備しています。機械処理できる情報の提供も行っており、RHELに含まれる範囲であれば、あるシステムにどのような既知の脆弱性が存在するかをコマンド一つで簡単に調べられます。

RHELに付随するサービスとしては、「初めてLinuxを触ります」という人から「パフォーマンスチューニングに使える技術的な詳細を知りたい」という人までを幅広くカバーした公式のトレーニングコースを提供しています。年間定額で、実機演習付きのトレーニングコースを(事実上)受け放題の「Red Hat Learning Subscription」というサービスも始めています。さらに専門技術を持つコンサルタントによる支援も用意しています。

Red Hat社は、単にソフトウェアを集めて提供しているの

ではなく、このように企業での利用に必要な特性やサービスを持ち合わせた製品としてRHELを販売しています。

Red Hat社のミッション

Red Hat社は、顧客、パートナー、OSSコミュニティの架け橋となることをミッションとしています。例えば、顧客からの機能リクエストを受け付けてソフトウェアに機能を追加する際は、事前にOSSコミュニティと相談して修正を取り込んでから製品にバックポートする「アップストリームファースト」というポリシーを採っています。もちろんコミュニティが受け入れない修正もありますのでではありませんが、このようにして顧客とコミュニティをつなぐことで、顧客には必要な機能が手に入り、コミュニティには隠れたニーズの通知と、それに対応する機能拡張を提供できます。

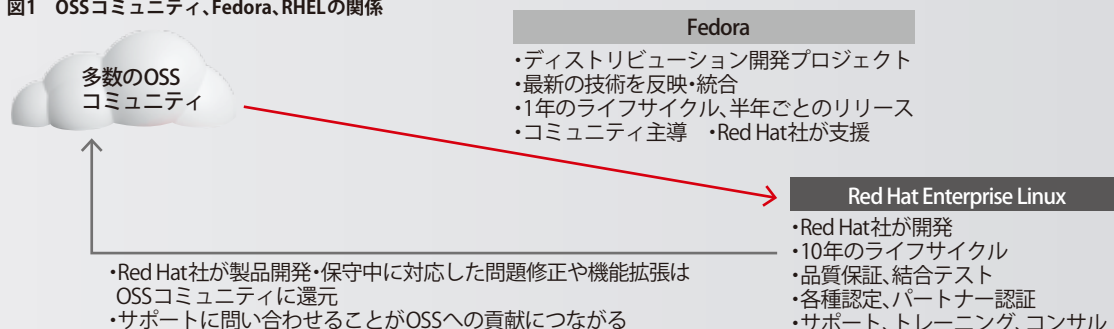
ハードウェアパートナーに対しては、問題が発生した場合に共同でサポートできる体制を構築した上で、認定ハードウェア、認定ハイパーバイザ、認定クラウドプロバイダとしてサポートが可能な動作環境をカタログ化しています。ソフトウェアパートナーに対しては、どれだけの期間どのライブラリについて互換性を維持するかという「Application Compatibility Guide^{*1}」を公開したり、カーネルモジュールを利用するソフトウェア向けにLinuxカーネルのABI(Application Binary Interface)互換性を維持したりするなどの支援をしています。さらにパートナー各社には、開発に利用できる製品の提供やオンライントレーニングの提供などを行っています。

FedoraとRHELの関係

Red Hat社が経済的・人的に支援するFedoraプロジェクトが開発する「Fedora」(<https://getfedora.org/ja/>)というLinuxディストリビューションがあります。Fedoraプロジェクトでは、最新のOSS技術を統合して半年おきにFedoraの新版をリリースする開発体制を採っています。

*1 URLは「<https://access.redhat.com/articles/rhel8-abi-compatibility>」。

図1 OSSコミュニティ、Fedora、RHELの関係



このFedoraは、用途を絞らない汎用のOSとして開発されています。Fedoraをベースに、品質保証や互換性の確認などのエンジニアリングを経てRHELが作られます(p.3の図1)。RHEL8は、2018年5月にリリースされたFedora 28をベースとして、およそ1年の開発期間を経てリリースされました。

ただし、RHELではサポートをはじめとした各種のサービスを提供することが前提となるため、パッケージ数はFedoraより非常に少なくなっています。単純な比較は難しいのですが、ソースパッケージ(source rpm)の個数で比較すると、おおよそ2万パッケージほどが提供されるFedora 28に対して、RHEL8では2500パッケージほどが提供されます。

RHEL8の概要

RHEL8は、2029年5月までの10年間にわたってサポートが提供されます。RHELのマイナーリリースやメジャーリリースの出荷タイミングは従来不定期だったのですが、RHEL8以降では、マイナーリリースは半年おき、メジャーリリースは3年おきを目標に出荷するポリシーにするとRed Hat社は表明しています。

RHEL8がサポートするハードウェアアーキテクチャは、AMD64/Intel64アーキテクチャ、64ビットARMアーキテクチャ、リトルエンディアンのIBM Power Systems、IBM Zアーキテクチャの4種類。主要なソフトウェアパッケージは表1の通りです。ほとんどのパッケージは、RHEL8のライフサイクルの間は互換性を維持しつつメンテナンスされますが、後述する「Application Stream」で提供される一部コンポーネントについては、2~5年の独自のサポート期間でメンテナンスされます。

またRHEL8のサブスクリプションには、定期的に情報収集

を行い、設定ファイルや統計情報、ログなどから問題になりうる兆候を発見し、具体的な対策方法をレポートする「Red Hat Insights」というサービスのサブスクリプションが含まれるようになりました。アドオン製品としては、多数のシステムを集中管理してパッケージ更新などの作業ができる「Smart Management」、Pacemakerを利用した高可用性クラスタを構成できる「High Availability」、GFS2を利用したクラスタファイルシステム「Resilient Storage」が別途販売されません。

RHELを入手するには

RHELはバージョンによらないサブスクリプション形式で販売されています。すでにRHELのサブスクリプションを持っている場合は、「Red Hatカスタマーポータル」(<https://access.redhat.com/>)からダウンロードできます*2。評価用サブスクリプションを申請して、30日間無償で試用することもできます。

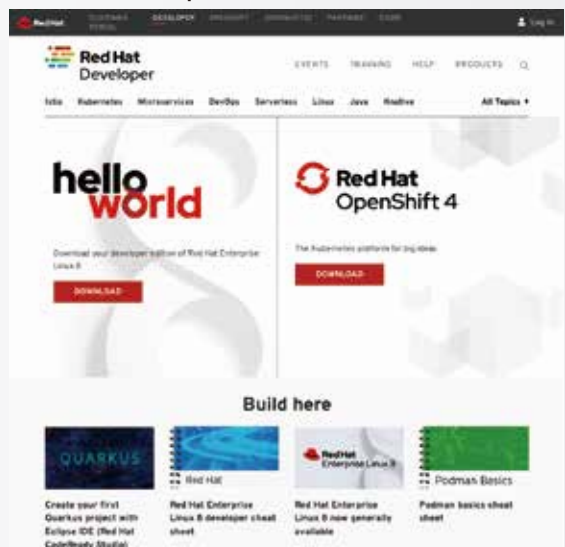
利用目的がソフトウェア開発であれば、RHELは無償で入手することもできます。利用者は1人のみでハードウェアも1台だけという制限はありますが、「Red Hat Developer Program」に参加して利用条件に同意すると、1年間開発用途に利用できるサブスクリプションが1本提供されます。期限切れ後も、再度利用条件に同意することで継続利用できます。ソフトウェア開発は、趣味のものであっても業務であっても問題ありません。

*2 RHEL8のダウンロードページのURLは「https://access.redhat.com/downloads/content/479/ver=/rhel--8/8.0/x86_64/product-software」。

表1 RHEL8の主なソフトウェアパッケージのバージョン

分類	主なパッケージのバージョン
カーネル	kernel 4.18
デスクトップ環境	GNOME 3.28
サーバー	httpd 2.4.37、squid 4.4.4、nginx 1.14.1、bind 9.11.4、postfix 3.3.1、dovecot 2.2.36、vsftpd 3.0.3、samba 4.9.1、cups 2.2.6、OpenSSH 7.8p1、rsyslog 8.37.0、MariaDB 10.3、MySQL 8.0.13、PostgreSQL 10.6、PostgreSQL 9.6、Redis 5.0.3、Pacemaker 2.0.1
言語処理系	Python 3.6/2.7、PHP 7.2、Perl 5.24/5.26、GCC 8.2.1、LLVM 7.0.1、Ruby 2.5.3、OpenJDK 1.8.0/11、golang 1.11.5、Rust 1.31.0

図2 「Red Hat Developer」サイトのトップページ



複数人で共用するシステムには使えませんが、手元のノートPCにRed Hat Developer Programで入手したサブスクリプションを利用し、共用のリポジトリやCI (Continuous Integration) 用のサーバーには購入したServer用のサブスクリプションを利用するといった使い分けが可能です。

Red Hat Developer Programの参加者には、RHELや各種ミドルウェアのサブスクリプションが提供されるだけでなく、英語ですが開発者向けの技術解説のブログや、各種のチートシート、電子書籍、動画デモなどでの情報提供もなされます。興味がある人はぜひ「Red Hat Developer」([https://developers.](https://developers.redhat.com/)

redhat.com/)のWebサイトにアクセスしてみてください(図2)。Red Hat Developer Programへの参加は、同Webサイトで手続きできます。

それ以外の一般的な用途にRHELを使う場合は、サブスクリプションを購入する必要があります。継続的に利用するのであれば、販売会社で1年または3年のサブスクリプションを購入するとよいでしょう。短期間の利用であれば、各種パブリッククラウドサービスでオンデマンド販売されているRHELインスタンスを使う方法もあります。

Part2 特徴編



頻繁なアップデートと安定性の両立

今までのRHELでは基本的に「次のメジャーバージョン/マイナーリリースでやるべきことを決め、それができたらリリースする」という開発モデルを採用していました。前述の通りRHEL8からは「メジャーバージョンは3年おき、マイナーリリースは半年おきにリリースすることを目標として、そのスケジュールに間に合うように盛り込む変更の量を調整する」という開発モデルに変更されました(図3)。リリースタイミングを予測可能にすること、マイナーリリースの出荷頻度を上げて新しい技術を取り込むタイミングを増やすことが狙いです。

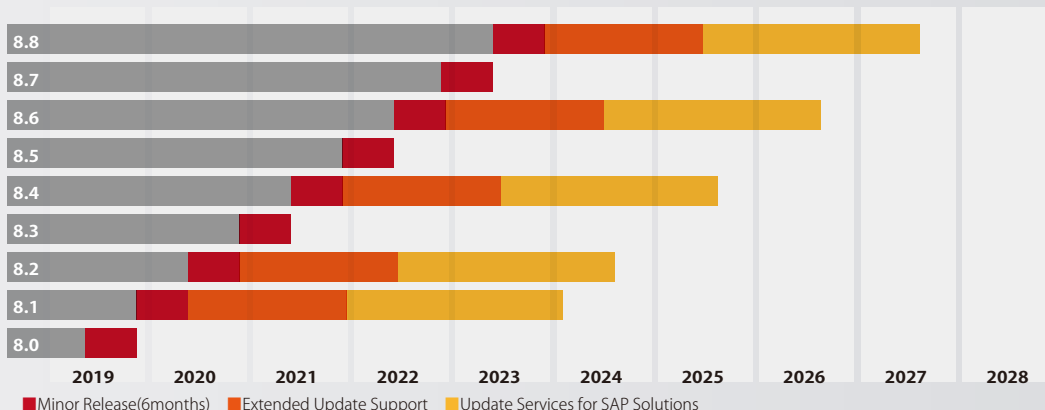
RHEL8の開発における大きなテーマの一つが「頻繁なアップデートと安定性の両立」です。活発に開発されているOSSの新機能を利用したいユーザーが多数いる状況にもかかわらず、従来のRHELでは、基本的にメジャーリリースのタイミングでのみ新しいソフトウェアを導入し、それを長期間維持するというポリシーを採用していました。RHELはメンテナ

ンス期間が長いメジャーバージョンのリリース間隔も長くなっています。例えば、RHEL7からRHEL8の間は4年11カ月と、5年近い間隔が空きました。開発期間を考慮すると、現在のRHEL7では、およそ6年前の技術を利用していることとなります。

これは特にソフトウェア開発者が利用する言語処理系やデータベース、Webサーバーなどで深刻な問題になります。例えば、RHEL7に標準で含まれるgitではバージョンが古すぎてGitHubの利用に問題が出たり、RubyやPHPなどの言語処理系が古すぎるためRuby on Railsなどの一般的なフレームワークやライブラリを利用できなかつたりする問題が表面化していました。

RHEL7では対策として、一部のソフトウェアについては、RHELの10年間というライフサイクルとは非同期に、3年または2年のサポート期間で新しいバージョンのソフトウェアを提供する「Red Hat Software Collections」という製品を付属しています。これは、RHEL標準のパッケージと競合しないように工夫して新しいバージョンのパッケージを提供す

図3 RHEL8のリリーススケジュールとサポート期間



る製品です。しかしRed Hat Software Collectionsの仕組みでは、「利用前に専用コマンドの実行が必要で、スクリプトなどに専用の修正が必要になる」「パッケージ名やインストール先のパスなどが通常のパッケージと異なるためメンテナンス上の負荷が大きい」などの不満点がありました。

リポジトリを二つに整理

RHEL8ではこの問題に対して、(1)RHELのリポジトリを、基盤となるOS機能のコアとして位置付けたパッケージが含まれる「BaseOS」と、BaseOS上で動作する「AppStream」(Application Stream)の二つに分ける、(2)AppStreamについては同時に複数バージョンを提供できる基盤を整え、頻繁に新しいバージョンを提供する、というアプローチで解決を図っています。

BaseOSリポジトリのパッケージについては、機能のバックポートなどがありますが基本的に10年間同じバージョンを維持し、各種の環境でRHELが動作する安定した基盤を維持します。

一方、AppStreamリポジトリには、互換性維持についてのポリシー^{*3}が異なる次のようなパッケージが混在します。

- RHELのメジャーバージョンが同一の間はバージョンを維持するもの
- RHELのマイナーリリースに合わせて新しいバージョンに置き換えるもの
- 並行して複数のバージョンが提供され、ユーザーが選択して切り替えていくもの

並行して複数のバージョンが提供されるパッケージは、RHEL8で導入された「モジュール」という仕組みで提供されます。ここでいうモジュールは、複数のパッケージをグループ化したもので、従来存在したパッケージグループに近いものです。モジュールは、バージョンに関する情報によって「ストリーム」という単位に分割できます。例えば、nodejsを提供するモジュールがあった場合、そのモジュールをバージョン8系列のストリームと、バージョン9系列のストリームなどに分割できます。ユーザーは、各モジュールから一つのストリームを選択してRHELで利用できます。

なお、同一モジュールの複数のストリームを同時に利用することはできません。これは競合が発生する恐れを回避するためです^{*4}。

図4 RHELとストリームのライフサイクルは独立
PostgreSQLモジュールのストリームを例に概念図を作成した。

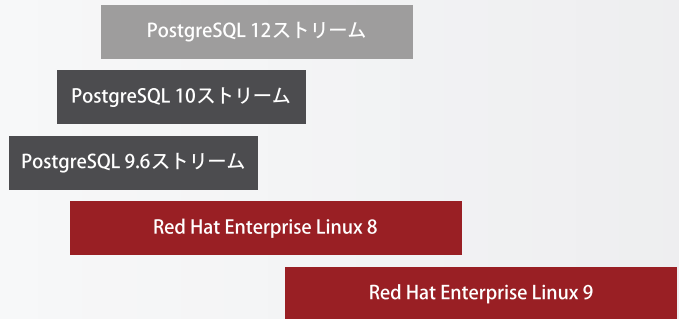


表2 RHEL8の出荷開始時点で提供されているストリームとライフサイクル

ストリーム	開始日	終了日
authd 1.4.4	May 2019	May 2021
container-tools 1	May 2019	May 2021
dotnet 2.1	May 2019	Aug 2021
git 2.18	May 2019	May 2021
httpd 2.4	May 2019	May 2024
Identity Management DL1	May 2019	May 2024
mariadb 10.3	May 2019	May 2023
maven 3.5	May 2019	May 2022
mercurial 4.8	May 2019	May 2022
mysql 8	May 2019	Apr 2023
nginx 1.14	May 2019	May 2021
nodejs 10	May 2019	Apr 2021
openjdk 1.8.0	May 2019	Jun 2023
openjdk 11	May 2019	Oct 2024
perl 5.24	May 2019	May 2021
php 7.2	May 2019	May 2021
postgresql 10	May 2019	Nov 2022
postgresql 9.6	May 2019	Nov 2021
python 2.7	May 2019	Jun 2024
redis 5	May 2019	May 2022
ruby 2.5	May 2019	Feb 2021
scala 2.1	May 2019	May 2022
swig 3	May 2019	May 2022
varnish 6	May 2019	May 2022

*3 ポリシーの詳細は*1のドキュメントに記載されています。

*4 同一ソフトウェアの複数バージョンを同時に利用したい場合は、コンテナ技術の利用を推奨しています。

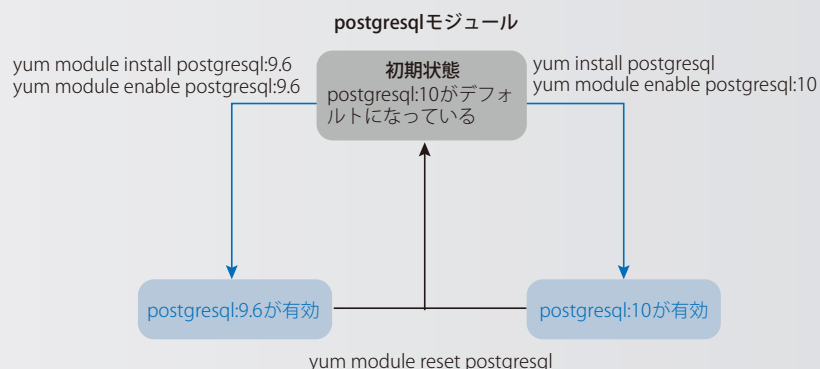
図5 モジュールの一覧情報を表示させた様子

```
# yum module list
(略)
Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMs)
Name                Stream      Profiles
389-ds              1.4
ant                 1.10 [d]    common [d]
container-tools    1.0         common [d]
container-tools    rhel8 [d][e] common [d]
(略)
postgresql         10 [d]      client, server [d]
postgresql         9.6         client, server [d]
python27           2.7 [d]     common [d]
python36           3.6 [d][e] common [d], build
(略)
Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

Name	Stream	Profiles	Summary
389-ds	1.4		389 Directory Server (base)
ant	1.10 [d]	common [d]	Java build tool
container-tools	1.0	common [d]	Common tools and dependencies for container runtimes
container-tools	rhel8 [d][e]	common [d]	Common tools and dependencies for container runtimes
postgresql	10 [d]	client, server [d]	PostgreSQL server and client module
postgresql	9.6	client, server [d]	PostgreSQL server and client module
python27	2.7 [d]	common [d]	Python programming language, version 2.7
python36	3.6 [d][e]	common [d], build	Python programming language, version 3.6

図6 モジュールの状態遷移

postgresqlモジュールを例に説明している。



モジュールを使ってみる

実際にRHEL8でモジュールを使ってみましょう。モジュールの一覧は次のコマンドで表示できます。yumコマンドの代わりにdnfコマンドを使用しても構いません。

```
# yum module list
```

実行結果は、例えば図5のようになります。モジュール名、ストリーム名、プロファイル名、概要の四つの情報が表示されます。なお、モジュールの仕組みは単純に複数のパッケージをまとめる用途にも利用

ストリームのライフサイクル

AppStreamリポジトリで提供されるモジュールのストリームには、RHEL8の10年間のライフサイクルとは独立した、2~5年のライフサイクルが定義されています(図4)。この期間はストリームごとに異なります。

RHEL8の出荷開始時点で提供されているストリームの一覧と、そのライフサイクルは表2の通りです*5。主要なWebサーバーソフトウェア、言語処理系、開発者用ツール、データベースが含まれていることが分かります。これらについては、ユーザーの選択次第で、新しいバージョンの処理系を利用することも、ある程度の期間同一バージョンを利用し続けることも可能になります。

なお、モジュールの仕組みを使っていないパッケージの一部にも、RHELとは独立したライフサイクルが定義されているものがあります。そうしたものの例としては、gitやdotnetが挙げられます。

されています。そのため、すべてのモジュールが複数のストリームを提供しているわけではありません。

モジュールには一般にデフォルトのストリームが設定されており、初期状態ではそのストリームがyumコマンドの操作対象となります。初期状態ではストリームは有効でも無効でもありません。

ストリームは自動的に有効化されることがあります。例えば、ストリームに含まれるパッケージをインストールすると、そのストリームが自動的に有効化されますし、ストリーム同士の依存関係によって自動的に有効化されることもあります。

ストリームを明示的に有効化することもできます。その場合、有効にしたストリームが、yumコマンドの操作対象となります。例えば、「yum module enable postgresql:9.6」というコマンドを実行して「postgresql:9.6」というストリームを有効にすると、「yum info postgresql」といったコマンドを実行

*5 ストリームのライフサイクルについての最新情報は「Red Hat Enterprise Linux 8 Application Streams Life Cycle」(<https://access.redhat.com/node/4079021>)のページで確認できます。

図7 同一モジュールの複数のストリームを同時に有効化することはできない

```
■postgresql:9.6ストリームを有効化
# yum module enable postgresql:9.6

■yum info postgresqlを実行した結果
# yum info postgresql
(略)
Available Packages
Name       : postgresql
Version    : 9.6.10
(略)

■この状態でpostgresql:10ストリームを有効化するとエラーになる
# yum module enable postgresql:10
(略)
The operation would result in switching of module 'postgresql' stream '9.6' to stream '10'
Error: It is not possible to switch enabled streams of a module.
(略)

■postgresqlモジュールの状態をリセットするとpostgresql:10ストリームを有効化できる
# yum module reset postgresql
# yum module enable postgresql:10
(略)

=====
Package      Arch          Version      Repository    Size
=====
Enabling module streams:
postgresql           10
(略)
```

```
# yum distro-sync
```

なお同コマンドは、モジュールに関係しないパッケージについては最新版に更新する働きをします。そのため、モジュールに関係するパッケージだけを差し替えたい場合には、使用できません。そうした場合に適用できるスマートな方法は現状なく、ストリームに含まれているパッケージをいったん削除した後で、利用したいストリームを有効化してインストールする作業が必要になります。

RHELでのモジュールの管理や利用については、公式ドキュメントの「ユーザー領域コンポー

した際には、有効にした「postgresql:9.6」というストリームがその操作対象となります。前述の通り、同一モジュールの複数のストリームを同時に有効化することはできません。あるストリームを有効化した状態で他のストリームを有効化する場合は、いったん「yum module reset postgresql」のようにコマンドを実行してモジュールの状態をリセットしてから、目的のストリームを有効化します。

モジュールの状態遷移についてp.7の図6にまとめました。また図7には、yumコマンドを使ってストリームの有効化やモジュールのリセットを実行した様子を示しました。

なお、ストリームは、「モジュール名:ストリーム名/プロファイル名」の形式で指定します。ストリーム名やプロファイル名は省略可能で、省略した場合にはデフォルトの値(yum module listの出力で「[d]」が付いているもの)が指定されます。

導入済みパッケージの同期

ストリームを切り替えてもリポジトリの状態が変わるだけで、システムに現在導入されているパッケージの状態はそのままです。有効化したストリームに合わせて導入済みのパッケージを更新あるいはダウングレードするには、次のコマンドを実行します。

ーネットのインストール、管理、および削除」に詳しくまとまっていますので参照してください^{*6}。



「OS利用の難しさ」を軽減する

RHEL8のもう一つのテーマが、各種のシーンで「RHELの利用をより簡単にする」ということです。RHELの利用が特別難しいというわけではありません。OSの新しい機能や最新情報を十二分に活用した効果的な運用をするのは、もともと簡単なことではないのです。

Red Hat社では、前述の通り各種トレーニングなども実施していますが、それと同時にRHELの利用をより簡単にするためのツールの開発にも力を注いでいます。以下では、そうしたツールの主なものを紹介します。

簡単にOSを管理できる「Web Console」

RHEL8では、従来バージョンに付属していたいくつかの管理用GUIツールが提供終了または非推奨となり、その代わりにWeb UIベースの管理用インタフェース「Web Console」(図8)が提供されるようになりました。

Web Consoleは「Cockpit」(<https://cockpit-project.org/>)というソフトウェアを基に開発されたもので、各種統計情報やログの参照、ネットワークやストレージ、ファイアウォー

^{*6} 同ドキュメントのURLは「https://access.redhat.com/documentation/ja-jp/red_hat_enterprise_linux/8/html/installing_managing_and_removing_user_space_components/index」です。

図8 「Web Console」の画面

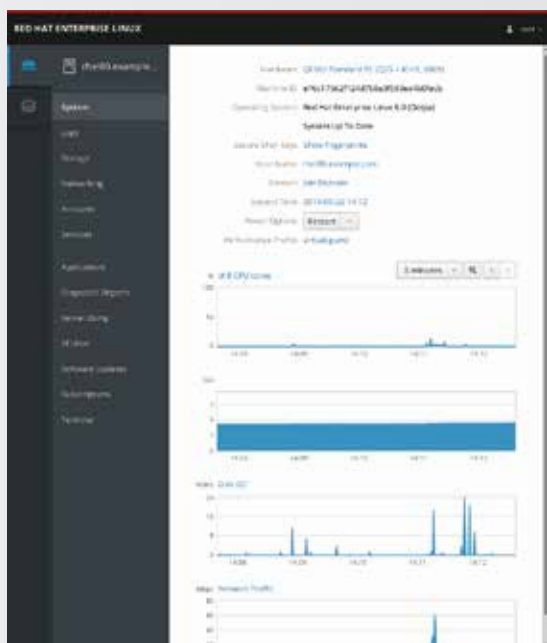
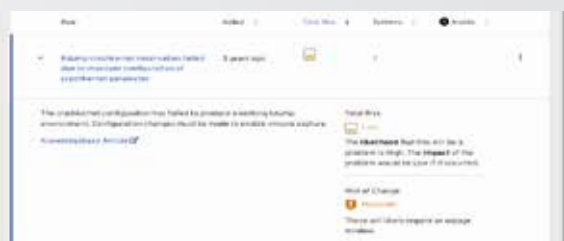


図9 「Red Hat Insights」のレポート例(一部を抜粋)
kdump設定の問題が指摘されている。



ルの設定、サービスの管理、KVM (Kernel-based Virtual Machine)による仮想マシンの管理などのさまざまな操作を実施できます。端末エミュレータとしても利用できます。

このようなツールではセキュリティが心配なところです。Web Consoleのログイン画面は「cockpit-ws」というユーザー権限で動作していますが、ログインすると(ログインした)ユーザーの権限でcockpit-sessionプロセスが起動し、それを使って各種操作を実施します。そのため、従来のGUIツールやコマンドラインツールを利用した管理に近いセキュリティを実現できます。さらにWeb Console自身では、ほとんど何も権限管理を行いませんので設定も非常にシンプルです。

Web Consoleの利用方法は、「Web コンソールを使用した

システムの管理」という公式ドキュメントに詳しく記載されています^{*7}。

障害を未然に防ぐ「Red Hat Insights」

前述した通り、RHEL8のサブスクリプションには「Red Hat Insights」というサービスのサブスクリプションが含まれるようになりました^{*8}。Red Hat Insightsは、RHELの6、7、8およびそれらをベースとした製品で利用可能な、SaaS (Software as a Service)形式で提供されるシステム診断サービスです。

Red Hat Insightsにシステムを登録すると、設定内容や導入ソフトウェア、統計情報、ログ情報の一部などの情報を定期的に収集してRed Hat Insightsに送信します。Red Hat Insightsは、約600種類のルールに基づいて送信された情報からリスクを検出し、問題の説明や対処方法、関連するナレッジベースがまとまったレポートを生成します(図9)。必要なアクションをまとめて、構成管理ツール「Ansible」用の設定ファイル(Playbook)を自動生成する機能も提供します。

従来は、障害が発生してからユーザーがサポート窓口にお問い合わせ、それを受けてシステム解析をして、問題把握や対処法を見つけるという流れになっていました。一方、Red Hat Insightsでは、定期的なシステム分析によって問題につながりそうな兆候を検出できます。ルールはナレッジベースとともに継続的に追加されており、多忙なシステム管理者であってもRed Hat社の最新の知見を素早く活用できます。

RHEL8には、Red Hat Insightsのクライアント「insights-client」がデフォルトでインストールされます。インターネットに直接接続できる環境であれば、次のコマンドを実行して、Red Hatアカウントとパスワードを入力すると登録が完了します^{*9}。

```
# insights-client --register
```

登録完了後、同サービスのポータルページ (<https://cloud.redhat.com/insights/>)でレポートを閲覧したり、発見された問題に対応するためのAnsible Playbookを生成したりするといった操作ができます。

仮想マシン配備を簡単にする「Image Builder」

仮想化基盤やIaaS (Infrastructure as a Service)を利用する場合、仮想マシンイメージを作成して、そのイメージを使ってOSやアプリケーションを配備したいケースがあります。そうしたときに便利なのが「Image Builder」というツールです。Image BuilderはRHEL7から提供されていましたが、RHEL8で

*7 同ドキュメントのURLは「https://access.redhat.com/documentation/ja-jp/red_hat_enterprise_linux/8/html-single/managing_systems_using_the_web_console/index」です。

*8 サブスクリプションの変更はRHEL8の出荷に合わせて発表されました。この変更はRHEL6やRHEL7にも適用されます。

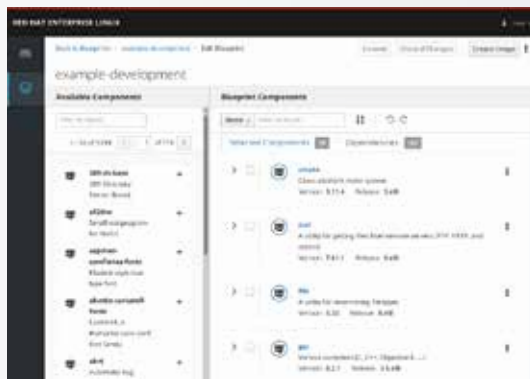
*9 パブリッククラウドサービス上のシステムや、プロキシ経由での接続が必要な場合などには登録前に設定作業が必要です。設定作業については「<https://access.redhat.com/ja/content/3385201#getstarted>」というURLのページを参照してください。

はWeb UIが追加されてAWS (Amazon Web Services)やAzure、GCP (Google Cloud Platform)用の仮想マシンイメージも作成できるようになりました。

Image Builderでは、ユーザーが作成した「blueprint」と呼ばれる必要パッケージやカスタマイズ内容が記載されたテキストファイルと、対象イメージの設定を入力として仮想マシンイメージを生成します。blueprintの編集やイメージの作成は、Web Consoleと統合されたGUIでも可能です(図10)。

なおImage Builderの詳しい利用方法は、「RHEL システムイメージのカスタマイズの作成」という公式ドキュメントを参照してください*10。

図10 Web Consoleに統合された「blueprintエディタ」



コンテナ配布を簡単にする「Universal Base Image」

自作のソフトウェアをコンテナイメージの形式で配布したいことがあります。しかし、RHELのコンテナイメージをベースイメージにすると、契約上、第三者に提供できません。そのため、作成したコンテナイメージをいったんRed Hat社に登録し、Red Hat社から配布する必要があります。

この問題を解決するべく、「Universal Base Image」(UBI)と呼ばれるコンテナイメージがRHEL8と同時にリリースされました。UBIには、標準的な「ubi」、サイズを小さくした「ubi-minimal」、systemdを含み複数サービスを起動できる「ubi-init」の3種類があります(図11)。また、言語ランタイムを含むイメージも提供されます。

UBIは、RHEL7またはRHEL8を基に作成されます。UBIはRHELのコンテナイメージと同じポリシー(6週おきまたはクリティカルな脆弱性対応ごとに再構築)で更新され、無償で入手、なおかつ自由に改変および再配布できます。Red Hatのコンテナ基盤(RHELまたはOpenShift)上で動作する場合には、UBI部分についてRed Hat社からのサポートを受けるこ

ともできます。

自作アプリケーションの配布に利用することが想定されており、RHELに含まれる言語処理系や主なライブラリのパッケージがUBI専用のリポジトリから提供されます。UBI専用リポジトリではなくRHELのリポジトリを利用すると契約上の制限を受け、再配布などができなくなるため注意が必要です。UBIのリポジトリに必要なライブラリやコマンドがない場合には、Red HatのBugzillaでパッケージの追加をリクエストできます。

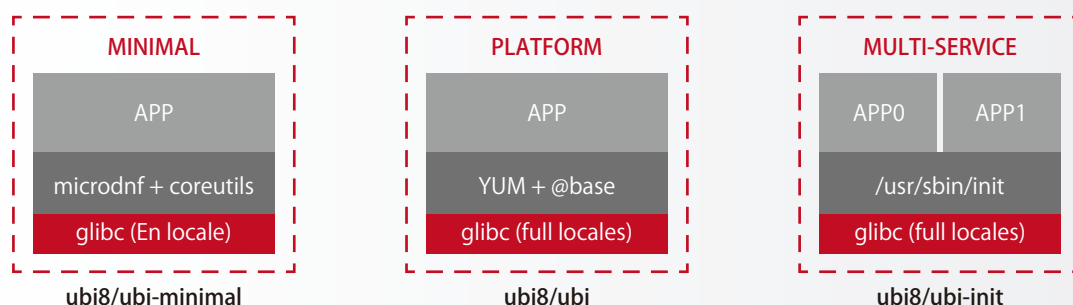
サブスクリプションが登録されたRHEL上でUBIを利用すると、UBI用のサブセットのリポジトリだけでなく、RHELのフルセットのリポジトリが利用できるよう自動的に構成されます。再配布のために利用する場合にはUBIのリポジトリだけが有効であることを確実にするため、yumコマンドに「--disableplugin=subscription-manager」オプションを付けて実行します。

UBIの詳細については、「Introducing the Red Hat Universal Base Image」という公式ブログ記事を参照してください*11。

*10 同ドキュメントのURLは「https://access.redhat.com/documentation/ja-jp/red_hat_enterprise_linux/8/html-single/composing_a_customized_rhel_system_image/index」です。

*11 同ブログ記事のURLは「<https://www.redhat.com/en/blog/introducing-red-hat-universal-base-image>」です。

図11 RHEL8をベースにする3種類のUBIの概要



Part3 RHEL7ユーザーへの注意編



RHEL7とRHEL8の違いを知る

RHEL7からRHEL8になって変更された点は、「RHEL 8の導入における検討事項」という公式ドキュメントにまとまっています^{*12}。RHEL7とRHEL8の差が気になる人は、まずこのドキュメントを読むことをお勧めします。

もちろん、パッケージについては、バージョンの変更や入れ替え、提供終了といった変更が多数あります。アプリケーションサーバーソフトウェア「Apache Tomcat」のパッケージ提供がなくなって「JBoss Web Server」が後継になったこと、ディレクトリサーバーソフトウェア「OpenLDAP」のパッケージ提供がなくなって「Red Hat Directory Server」が後継になったことの2点については、後継ソフトウェアがいずれも有償のため、費用の面からも注意が必要です。

RHEL種別の扱いとsyspurpose

RHELは、用途によって「Server」「Desktop」などの種別に分かれています。RHEL7までは、提供・サポートするソフトウェアのセットも種別により異なっていて、インストールメディアもそれぞれ別のものでした。一方RHEL8では、すべての種別がソフトウェアとしては同一になり、インストールメディアやコンポーネントのサポート状況も同じで、用途やハードウェアの制限だけが異なるように変更されました。

この変更に伴い、ダウンロードページで選択する製品名が「Red Hat Enterprise Linux Server」のような種別を軸にした

^{*12} 同ドキュメントのURLは「https://access.redhat.com/documentation/ja-jp/red_hat_enterprise_linux/8/html-single/considerations_in_adopting_rhel_8/」です。

ものから、「Red Hat Enterprise Linux for x86_64」のようにアーキテクチャを軸にしたものになっています。やや分かりにくいので、注意が必要です^{*13}。

システムの利用目的はインストーラで選択するか、インストール後にsyspurposeコマンドで設定します。例えば、一つの仮想マシンイメージをワークステーションとサーバーで共用して、仮想マシン作成後にsyspurposeコマンドで用途を設定するような利用方法が可能です。

Ansibleの対応

RHEL8では、インストール状況によってPythonプログラム実行用の通常のコマンド（pythonやpython2、python3）が存在しない場合があります。Pythonはyumコマンドの実装などディストリビューションの基盤に使われており、RHEL8ではそのための処理系を通常のユーザーが利用するための処理系と分離して「platform-python」として用意しています。

Ansible 2.8以降ではplatform-pythonを自動検出できません。しかしAnsible 2.7以前は自動検出できないため、platform-pythonのみしかシステムに存在しない場合に問題が生じます。platform-pythonを利用する場合には、Ansibleのインベントリで、python処理系を次のように指定する必要があります。

```
ansible_python_interpreter=/usr/libexec/platform-python
```

^{*13} 筆者もこの違いに引っかけ、「もう出荷されてるはずなのにダウンロードページにRHEL8が出てこない」と10分ほど悩んでしまいました。

もっと詳しく知りたい人は

ここまでRHEL8の概要や特徴、RHEL7ユーザーへの注意点を紹介してきました。最後にRHEL8についてもっと詳しく知りたい人向けの情報源を表3にまとめましたので参考にしてください。

「Explore Red Hat Enterprise Linux 8」では、学習用Webサービス「Katacoda」による演習環境で、RHEL8のいくつかの新機能を試すことができます。「Red Hat Enterprise Linux 8 New

Features for Experienced Linux Administrators」は、RHEL7までに習熟しているシステム管理者向けのRHEL8新機能についてのトレーニングコースです。

「赤帽エンジニアブログ」は、社内有志で書いているブログです。RHEL8関連記事のほか、AnsibleやOpenShift、ストレージやシステム統合フレームワークのApache Camelなどの記事もあります。

表3 RHEL8についての情報源

分類	タイトル	URL
公式ドキュメント	Product Documentation for Red Hat Enterprise Linux 8	https://access.redhat.com/documentation/ja-jp/red_hat_enterprise_linux/8/
	Product Documentation for Red Hat Insights 1.0	https://access.redhat.com/documentation/en-us/red_hat_insights/1.0/
新機能体験	Explore Red Hat Enterprise Linux 8	http://lab.redhat.com/
公式トレーニング	Red Hat Enterprise Linux 8 New Features for Experienced Linux Administrators	https://www.redhat.com/en/services/training/rh354-red-hat-enterprise-linux-8-new-features-experienced-linux-administrators
ブログ記事	赤帽エンジニアブログ	https://rheb.hatenablog.com/



Red Hat

レッドハット株式会社

〒150-0013

東京都渋谷区恵比寿4-1-18 恵比寿ネオナート 3階

jp.redhat.com

Red Hat, Red Hat Enterprise Linux, Red Hat logoは、米国およびその他の国におけるRed Hat, Inc.およびその子会社の商標または登録商標です。Linux®は、米国およびその他の国におけるLinus Torvaldsの登録商標です。