Red Hat Enterprise Linux 6

移行計画ガイド

Red Hat Enterprise Linux 6 への移行

[FAMILY Given]

Red Hat Enterprise Linux 6 移行計画ガイド
Red Hat Enterprise Linux 6 への移行
エディッション 6.1

著者 [FAMILY Given]

Copyright © 2011 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at http://creativecommons.org/licenses/by-sa/3.0/. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

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

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, MetaMatrix, 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.

All other trademarks are the property of their respective owners.

1801 Varsity Drive
Raleigh, NC 27606-2072 USA
Phone: +1 919 754 3700
Phone: 888 733 4281
Fax: +1 919 754 3701

本ガイドでは、Red Hat Enterprise Linux 5 を稼働しているシステムの Red Hat Enterprise Linux 6 への移行を説明しています。

前書き v
1. 表記方法 v
1.1. 印刷における表記方法 v
1.2. 引用における表記方法 vi
1.3. 注記および警告 vii
2. フィードバックをお願いします。 vii

1. はじめに 1
1.1. Red Hat Enterprise Linux 6 1
1.2. アプリケーションの互換性 2

2. インストール 5
2.1. カーネルとブートのオプション 5
2.2. グラフィカルなインストーラ 5
2.2.1. デバイスとディスク 5
2.2.2. キックスタート 5
2.2.3. ネットワーク設定 9
2.2.4. 製品のサブスクリプションとコンテンツの更新 10
2.3. テキストベースのインストーラ 11

3. ストレージとファイルシステム 13
3.1. RAID 13
3.2. ext4 14
3.3. fusecompress 14
3.4. blockdev 14

4. ネットワーク設定とサービス 15
4.1. インターフェースと設定 15
4.2. サービスの初期化 15
4.3. IP テーブル/ファイアウォール 16
4.4. Apache HTTP Server 16
4.5. PHP 16
4.6. BIND 17
4.7. NTP 18
4.8. Kerberos 18
4.9. Mail 19
4.9.1. Sendmail 19
4.9.2. Exim 19
4.9.3. Dovecot 19
4.10. MySQL® 19
4.11. PostgreSQL 20
4.12. Squid 20
4.13. Bluetooth 20
4.14. Cron 20
4.15. ログ記録 21

5. コマンドラインツール 23
5.1. Grep 23
5.2. Sed 23
5.3. Pcre 23
5.4. シェル 23
5.5. Nautilus 23

6. デスクトップ 25

7. セキュリティと認証 27
7.1. SELinux 27

7.2. SSSD 27
7.3. LDAP 27
7.3.1. slapd 設定を変換する 27
7.4. Checksum 28
7.5. PAM (プラグ可能な認証モジュール) 28
7.6. システムのユーザー 28

8. カーネル 29
8.1. カーネル 29

9. パッケージとドライバーに関する変更点 31
9.1. システム設定ツールの変更点 31
9.2. Bash (Bourne-Again Shell) 33
9.3. その他のパッケージ変更点 34
9.4. ドライバー関連の変更点 35
9.5. ライブラリ関連の変更点 37

A. 改訂履歴 39

前書き

1. 表記方法

本ガイドは特定の単語や語句を強調したり、記載内容の特定部分に注意を引かせる目的で次のような表記方法を使用しています。

PDF版 および印刷版では、Liberation Fonts[1] セットから採用した書体を使用しています。ご使用のシステムに Liberation Fonts セットがインストールされている場合、HTML 版でもこのセットが使用されます。インストールされていない場合は代替として同等の書体が表示されます。注記: Red Hat Enterprise Linux 5 およびそれ以降のバージョンにはデフォルトで Liberation Fonts セットが収納されます。

1.1. 印刷における表記方法

特定の単語や語句に注意を引く目的で 4 種類の表記方法を使用しています。その表記方法および適用される状況は以下の通りです。

等幅の太字

シェルコマンド、ファイル名、パスなどシステムへの入力を強調するために使用しています。またキー配列やキーの組み合わせを強調するのにも使用しています。例えば、

> 現在作業中のディレクトリ内のファイル `my_next_bestselling_novel` の内容を表示させるには、シェルプロンプトで `cat my_next_bestselling_novel` コマンドを入力してから `Enter` を押してそのコマンドを実行します。

上記にはファイル名、シェルコマンド、キーが含まれています。すべて等幅の太字で表されているため文中内で見分けやすくなっています。

キーが 1 つの場合と複数のキーの組み合わせになる場合を区別するため、その組み合わせを構成するキー同士をハイフンでつないでいます。例えば、

> `Enter` を押してコマンドを実行します。
>
> 1 番目の仮想ターミナルに切り替えるは、`Ctrl+Alt+F2` を押します。X-Windows セッションに戻るには、`Ctrl+Alt+F1` を押します。

最初の段落では押すべき 1 つのキーを特定して強調しています。次の段落では同時に押すべき 3 つのキーの組み合わせが 2 種類ありそれぞれ強調されています。

ソースコードの説明では 1 段落内で提示されるクラス名、メソッド、関数、変数名、戻り値を上記のように 等幅の太字 で表示します。例えば、

> ファイル関連のクラス群はファイルシステムに対しては `filesystem`、ファイルには `file`、ディレクトリには `dir` をそれぞれ含みます。各クラスは個別に関連する権限セットを持っています。

プロポーショナルの太字

アプリケーション名、ダイアログボックスのテキスト、ラベル付きボタン、チェックボックスとラジオボタンのラベル、メニュータイトルとサブメニュータイトルなどシステム上で見られる単語や語句を表します。例えば、

[1] https://fedorahosted.org/liberation-fonts/

メインメニューバーから システム > 個人設定 > マウス の順で選択し マウスの個人設定 を起動します。ボタン タブ内で 左ききのマウス チェックボックスをクリックしてから 閉じる をクリックしマウスの主要ボタンを左から右に切り替えます (マウスを左ききの人が使用するのに適した設定にする)。

gedit ファイルに特殊な文字を挿入する場合は、メインメニューバーから アプリケーション > アクセサリ > 文字マップ の順で選択します。次に 文字マップ メニューバーから 検索 > 検索… と選択して 検索 フィールド内にその文字名を入力し 次 をクリックします。探している文字が 文字表 内で強調表示されます。この強調表示された文字をダブルクリックすると コピーするテキスト フィールド内に置かれるので次に コピー ボタンをクリックします。ここでドキュメントに戻り gedit メニューバーから 編集 > 貼り付け を選択します。

上記には、アプリケーション名、システム全体のメニュー名と項目、アプリケーション固有のメニュー名、GUI インタフェースで見られるボタンやテキストがあります。すべてプロポーショナルの太字で表示されているため文中内で見分けやすくなっています。

等幅の太字で且つ斜体 または プロポーショナルの太字で且つ斜体

等幅の太字やプロポーショナルの太字はいずれであっても斜体の場合は置換可能なテキストか変化するテキストを示します。斜体は記載されている通りには入力しないテキスト、あるいは状況に応じて変化する出力テキストを表します。例えば、

ssh を使用してリモートマシンに接続するには、シェルプロンプトで ssh *username@domain.name* と入力します。リモートマシンが example.com であり、そのマシンで使用しているユーザー名が john なら ssh *john@example.com* と入力します。

mount -o remount *file-system* コマンドは指定したファイルシステムを再マウントします。例えば、/home ファイルシステムを再マウントするコマンドは mount -o remount /home になります。

現在インストールされているパッケージのバージョンを表示するには、rpm -q *package* コマンドを使用します。結果として次を返してきます、*package-version-release*。

上記の太字斜体の単語 — *username*、*domain.name*、*file-system*、*package*、*version*、*release* に注目してください。いずれもコマンドを発行するときに入力するテキスト用のプレースホルダーかシステムにより出力されるテキスト用のプレースホルダーになっています。

タイトル表示のような標準的な使用の他、斜体は新しい重要な用語が初めて出現する場合にも使用されます。例えば、

Publican は DocBook の発行システムです。

1.2. 引用における表記方法

端末の出力とソースコード一覧は、視覚的に周囲の文から区別されています。

端末に送信される出力は mono-spaced roman (等幅の Roman) にセットされるので以下のように表示されます。

```
books        Desktop   documentation  drafts  mss    photos   stuff  svn
books_tests  Desktop1  downloads      images  notes  scripts  svgs
```

ソースコードの一覧も mono-spaced roman (等幅の Roman) でセットされますが、以下のように強調表示されます。

```
package org.jboss.book.jca.ex1;
```

```
import javax.naming.InitialContext;

public class ExClient
{
   public static void main(String args[])
       throws Exception
   {
      InitialContext iniCtx = new InitialContext();
      Object         ref    = iniCtx.lookup("EchoBean");
      EchoHome       home   = (EchoHome) ref;
      Echo           echo   = home.create();

      System.out.println("Created Echo");

      System.out.println("Echo.echo('Hello') = " + echo.echo("Hello"));
   }
}
```

1.3. 注記および警告

情報が見過ごされないよう 3 種類の視覚的なスタイルを使用して注意を引いています。

注記

注記は説明している部分に対するヒントや近道あるいは代替となる手段などになります。注記を無視しても悪影響はありませんが知っておくと便利なコツを見逃すことになるかもしれません。

重要

重要ボックスは見逃しやすい事項を詳細に説明しています。現在のセッションにのみ適用される設定上の変更点、更新を適用する前に再起動が必要なサービスなどがあります。重要ボックスを無視してもデータを喪失するような結果にはなりませんがイライラ感やフラストレーションが生じる可能性があります。

警告

警告は無視しないでください。警告を無視するとデータを喪失する可能性が非常に高くなります。

2. フィードバックをお願いします。

本ガイド内に誤字や脱字を発見された場合、またガイドに関して改善案をお持ちの場合はぜひ弊社にご連絡ください。Bugzilla (http://bugzilla.redhat.com/) にて Red Hat Enterprise Linux 製品に対しバグ報告を提出してください。

バグ報告を提出される際はガイドの識別子となる「doc-Migration_Guide」とバージョン番号「6」を忘れずにご記入ください。

本ガイドに対する改善案をお持ちの場合は、できるだけ詳細にご報告頂けるようお願いします。エラーを発見された場合には、セクション番号および前後の文章も報告頂けると迅速な対応が可能になります。

はじめに

移行計画ガイドでは Red Hat Enterprise Linux 5 インストールのマイナーバージョンを Red Hat Enterprise Linux 6 に移行する際に注目すべき主な動作変更を中心とした移行に関する説明を記載しています。

本ガイドでは Red Hat Enterprise Linux 5 と Red Hat Enterprise Linux 6 の製品上の変更点に関するガイドラインを説明することで Red Hat Enterprise Linux 6 の使い勝手を向上させることを目的としています。ただし、本ガイドは新機能をすべて説明するためのガイドではありません。Red Hat Enterprise Linux 5 の一部であり Red Hat Enterprise Linux 6 では変更されているアプリケーションやコンポーネントの 動作、またはその機能が別のパッケージによって置換されているパッケージに焦点を絞って説明しています。

1.1. Red Hat Enterprise Linux 6

Red Hat Enterprise Linux はオープンソースコンピューティング業界で先頭を行くプラットフォームです。サブスクリプション方式で販売され、その真価を継続的に配信することでトップクラス企業のハードウェアベンダーやソフトウェアベンダーにより認定されています。デスクトップからデータセンターまで、Enterprise Linux はオープンソース技術による革新と真のエンタープライズクラスプラットフォームとしての安定性との統合を実現しています。

Red Hat Enterprise Linux 6 は、Red Hat に於ける次世代の総合的オペレーティングシステムのセットです。ミッションクリティカルな企業用コンピューティング向けに設計され、トップクラスのソフトウェアベンダーおよびハードウェアベンダーにより認定されています。本リリースは以下のアーキテクチャで単独キットとして使用できます。

- i386
- AMD64/Intel64
- System z
- IBM Power (64-bit)

本リリースではサーバー、デスクトップ、Red Hat の総合的なオープンソース経験などより得た改善の集大成を提供しています。以下に、数多くの改善の一部と本リリースに収納されている新しい機能を示します。

電力管理

PowerTOP、Power Management (ASPM, ALPM)、及び Tuned による適応型システムチューニングにより、アプリケーションスタックを介したティックレスカーネルおよび各種の改善で電力消費の目安となる wakeup を低減します。

次世代のネットワーキング

IPv6 に統合的に対応 (NFS 4、CIFS、モバイルサポート [RFC 3775]、ISATAP サポート)、FCoE や iSCSI、新たに改善を加えた mac80211 ワイヤレススタックなどを搭載しています。

信頼性、可用性、サービスの有用性

産業連携によるシステムレベルでの強化により、ほとんどのハードウェアに対して RAS 機能および NUMA アーキテクチャを実現しています。

より詳細なレベルでの管理

CFS (Completely Fair Scheduler) と CG (Control Groups) によりカーネル内のリソース管理やスケジューラを改善しています。

スケーラブルなファイルシステム

ext4 がデフォルトのシステムになります。xfs は堅固でスケーラブルでありながら優れたパフォーマンスを実現します。

仮想化

KVM にパフォーマンスの改善と新しい機能が加えられています。sVirt はホストの仮想マシンとデータをゲストによる違反から守ります。SRIOV および NPIV により物理デバイスの仮想使用におけるパフォーマンスが向上され、また libvirt によりカーネルの CG コントローラ機能が活用されています。

企業レベルのセキュリティ強化

SELinux では、使い易さを向上させアプリケーションのサンドボックス化やシステムサービスの対象範囲を大幅に拡張させている一方、SSSD では ID 認証サービスに対する統合アクセスやオフライン使用向けのキャッシング機能なども提供しています。

開発およびランタイムのサポート

SystemTap (再コンパイルすることなく実行中のカーネルを操作)、ABRT (バグ情報のシンプルな収集) を搭載し、GCC (バージョン 4.4.3)、glibc (バージョン 2.11.1)、GDB (バージョン 7.0.1) には改良を加えています。

1.2. アプリケーションの互換性

Red Hat Enterprise Linux の本リリースでは、旧式のオペレーティングシステムで実行するよう設計されたアプリケーションに対するダメージを最小限に抑え、継続してそのアプリケーションを実行させることができるよう依存性を提供しています。このため、本リリースと旧バージョン間で変更されている可能性のあるレガシーなインターフェースを維持できるよう主要なライブラリの旧バージョンが収納されています。これらのライブラリは C/C++ で記述されたアプリケーションの依存関係を解決することが主な目的です。

Red Hat Enterprise Linux のマイナーリリース間ではアプリケーションの再テストや再認定は必要ありません。Red Hat Enterprise Linux の互換性ポリシーでは、そのリリースの任意のバージョンで実行しているアプリケーションはそのリリースのライフ期間中を通じて継続的に実行することを保証しています。例えば、Red Hat Enterprise Linux 6.0 で認定されたアプリケーションは Red Hat Enterprise Linux 6.1 などの全マイナーリリースで完全な互換性を持つことになります。

互換性パッケージについての詳細には以下の表を参照してください。

表1.1 互換性ライブラリ

パッケージ	説明
compat-db	Berkeley DB データベース互換性ライブラリです。Berkeley Database (Berkeley DB) はプログラム型のツールキットで従来のアプリケーションおよびクライアント/サーバー型アプリケーションのいずれに対しても組込み型のデータベースサポートを提供します。このパッケージには、旧リリースに収納されていた Berkeley DB の各種バージョンが含まれています。
compat-expat1	Expat はストリーム指向の XML 構文解析ツールです。このパッケージは旧バージョンとのライブラリ互換性を提供します。
compat-glibc	glibc はシステムコールやその他の基本的な機能に使用される C ライブラリです。このパッケージは旧式

パッケージ	説明
	の glibc バージョンを必要とするバイナリのコンパイルに要される互換性 (及びランタイムライブラリ) を提供し、Red Hat Enterprise Linux の本リリースで実行できるようにします。
compat-libf2c-34	このパッケージは、Fortran 77 共有ライブラリの旧バージョン群を提供します。動的にリンクされた Fortran 77 プログラムの実行に必要となります。
compat-libgcc-296	2.96 libgcc.a ライブラリとサポートオブジェクトファイルを含み、GCC の旧バージョンとの互換性を維持します。
compat-libgfortran-41	このパッケージには、GCC 4.1.x コンパイルの Fortran アプリケーションとの互換性に要される Fortran 95 ランタイムライブラリが含まれています。
compat-libstdc++-295	GNU 標準 C++ ライブラリのバージョン 2.95 との互換性を提供します。
compat-libstdc++-296	GNU 標準 C++ ライブラリのバージョン 2.96 との互換性を提供します。
compat-libstdc++-33	GNU 標準 C++ ライブラリのバージョン 3.3 との互換性を提供します。
compat-libtermcap	このパッケージは termcap ベースの旧式プログラム用の互換性を提供します。
compat-openldap	OpenLDAP は LDAP (Lightweight Directory Access Protocol) アプリケーション及び開発ツールセットのオープンソース版です。compat-openldap パッケージには、一部のアプリケーションで必要になる可能性のある OpenLDAP 共有ライブラリの旧バージョンが含まれています。
openssl098e	このパッケージは、一部の SSL アプリケーションで必要になる可能性のある OpenSSL 0.98e を提供します。

インストール

本セクションでは、Red Hat Enterprise Linux 6 と Red Hat Enterprise Linux 5 とのインストール手順の違いについて簡単に説明します。Red Hat Enterprise Linux 5 のいずれのリリースから移行するかにより、その環境にここで説明しているオプションや技術がすでに存在している場合があるため、すべてが適しているとは限りません。

2.1. カーネルとブートのオプション

- Red Hat Enterprise Linux をインストールする前に、`boot:` プロンプトで `memtest86` と入力するとメモリーのテストを行うことができます。このオプションにより Anaconda システムインストーラではなく、独立型のメモリーテストソフトウェア Memtest86 が実行されます。このソフトウェアを一度実行すると、Esc キーを押すまで Memtest86 メモリーテストは繰り返して実行が継続されます。

- モジュールの読み込み順序の定義に、旧式のオプションとなる `scsi_hostadapter` ではなく `rdloaddriver` カーネルパラメータが必要になります。

2.2. グラフィカルなインストーラ

本セクションでは、グラフィカルなインストーラで変更になっている動作を説明します。

2.2.1. デバイスとディスク

- /dev/hdX というようなデバイス名の使い方は IDE ドライブ用の i386 と x86_64 のアーキテクチャでは非推奨になり、/dev/sdX という使い方に変更になります。この変更は PPC アーキテクチャには適用されません。

- インストールで Smart Array カードが検出されない場合はインストーラのプロンプトで `linux isa` と入力します。これで必要なカードを手作業で選択することができるようになります。

- 旧式の IDE ドライバーはデバイス毎に最大 63 パーティションまで対応していたのに対し、SCSI デバイスはデバイス毎に 15 パーティションに制限されています。Anaconda は Red Hat Enterprise Linux の他の部分と同じように新しい libata ドライバを使用しますので、インストール時やアップグレード中に 15 を越えるパーティションは検出できません。15 を越えるパーティションでシステムをアップグレードしている場合は、そのディスクの LVM (論理ボリュームマネージャ) への移行が必要な場合があります。

- カーネルによるストレージデバイスの処理方法が変更されたことで /dev/hdX や /dev/sdX などのデバイス名が旧リリースで使用されていた値と一致しないことがあります。Anaconda はパーティションラベルに依存することでこの問題を解決します。このラベルがないと、Anaconda によりパーティションにラベル付けが必要なことが警告されてきます。LVM (論理ボリューム管理) とそのデバイスマッパーを使用するシステムの場合、通常は再ラベルの必要はありません。

- root ファイルシステムを含む、暗号化したブロックデバイスへのインストール用のサポートが含まれます。

- 全ての IDE RAID コントローラに対応しているとは限りません。`dmraid` がご使用の RAID コントローラにまだ対応していない場合は、Linux ソフトウェア RAID を設定することで複数のドライブ群を RAID アレイに結合することができます。対応済みのコントローラの場合は、コンピュータの BIOS 内で RAID 機能を設定します。

- Red Hat Enterprise Linux 6 に収納されている GRUB のバージョンは ext4 に対応するようになるため、Anaconda では `/boot` や root のパーティションも含めいずれのパーティションでも ext4 ファイルシステムを使用できるようになります。

2.2.2. キックスタート

本セクションでは、自動化したインストール (キックスタート) での動作変更について説明します。

2.2.2.1. 動作変更

- 以前は、キックスタートファイルに network の行を持たせないことで「ネットワークの設定には DHCP を使用する」ということになっていました。しかし、キックスタートの他の部分では「他のすべての行がない場合はインストールを中断して入力を求める」ということになっているためこれと矛盾していました。本リリースでは、network の行を持たせないとインストールを中断して入力を求めるようになります。また、--bootproto=query オプションも非推奨になります。入力を必要とすることなく DHCP を使用させたい場合は、キックスタートファイルに network --bootproto=dhcp を追加してください。

- ディスクは、従来キックスタートの全過程でデバイスノード名 (sda など) を使って参照されていました。しかし、Linux カーネルがより動的なメソッドに転換してきたことにより再起動後のデバイス名が一致する保証がなくなってしまっているため、キックスタートスクリプトでデバイス名を使用するのが難しくなっています。一貫したデバイス名を維持するには、デバイスノード名の代わりに /dev/disk のアイテムを使用します。例えば、以下のデバイスノード名を使用していたとします。

```
part / --fstype=ext4 --onpart=sda1
```

  代わりに、以下のいずれかのようにすることができます。

```
part / --fstype=ext4 --onpart=/dev/disk/by-path/pci-0000:00:05.0-scsi-0:0:0:0-part1
part / --fstype=ext4 --onpart=/dev/disk/by-id/ata-ST3160815AS_6RA0C882-part1
```

  これにより、sda ではない、より一貫したディスクの参照を行わせることができるようになります。これは特に大規模なストレージ環境で役に立ちます。

- また、ディスクの参照にシェル系のエントリを使用することもできます。主に大規模なストレージ環境で clearpart と ignoredisk コマンドを使い易くするのが目的です。例えば、以下のデバイスノード名を使用していたとします。

```
ignoredisk --drives=sdaa,sdab,sdac
```

  代わりに、以下のようにすることができます。

```
ignoredisk --drives=/dev/disk/by-path/pci-0000:00:05.0-scsi-*
```

- 以前のバージョンに比べ、キックスタートはエラーで中断されることが多くなります。例えば、存在しないディスクを参照するとインストールが中断されエラーを通知します。深刻な問題を招く前にキックスタートファイル内のエラーを検出するよう設定されているためです。副作用として、各種のマシン構成に対して汎用となるよう設計されているファイルの場合、エラーの発生頻度が高くなることがあります。これらについてはその都度、対処してください。

- キックスタートのネットワーク情報に使用していた /tmp/netinfo ファイルが削除されています。インターフェースの設定には NetworkManager が Anaconda により使用され、設定は /etc/sysconfig/network-scripts/ にある ifcfg ファイル群に保存されます。この新しい場所を %pre スクリプトおよび %post スクリプトのネットワーク設定のソースとして使用することができます。

2.2.2.2. コマンドの変更

本セクションでは、コマンドとそのオプションに対する最も重要な変更について説明していきます。

- network --device オプションでは、デバイス名ではなく MAC アドレスでデバイスを参照できるようになります。ディスクと同様に、ネットワークデバイス名もデバイスが検出される順番に応じて再起動後に変化します。キックスタートで一貫した命名を行うためには、以下のようなエントリを使用することができます。

```
network --device=00:11:22:33:44:55 --bootproto=dhcp
```

- langsupport、key、mouse の各コマンドは削除されています。これらのコマンドを使用すると構文エラーになります。monitor コマンドも非推奨になっています。

  langsupport の代わりに、適切なグループをキックスタートファイルの %packages セクションに追加してください。例えば、フランス語のサポートを含める場合は @french-support を追加します。

  インストール時にインストールキーは要求されなくなるため、key オプションの代替はありません。このオプションはご使用ファイルから削除してください。

  X により自動的に設定を検出して設定を行うことができるため、mouse と monitor のコマンドは必要がなくなりました。これと同じ理由で xconfig --resolution= コマンドも必要性がなくなるため、すべてファイルから削除してもまったく問題ありません。

- part --start と part --end のコマンドは非推奨となっているため何の効果もありません。Anaconda は、特定のセクターの境界にはパーティションの作成を許可しなくなります。より厳密なレベルのパーティション設定を必要とする場合は、%pre 内の外部ツールを使用してから Anaconda に part --onpart コマンドを使って既存のパーティションを使用するよう指示します。これ以外の場合は、特定サイズのパーティションを作成するか、--grow を使用します。

- %post で手作業によりグループ群を作成する代わりに、group コマンドを使用してグループ群を作成することができるようになります。詳細についてはキックスタートのドキュメントを参照してください。

- デフォルトの autopart アルゴリズムが変更されています。すべてのマシンに対して autopart は /boot (またはアーキテクチャで要される他の特殊なブートローダパーティション) と swap を作成します。最低でも 50 GB の空きディスク領域を持つマシンには、autopart は適当なサイズの root パーティション (/) を作成してから残りを /home に割り当てます。空き領域の少ないマシンの場合、root (/) のみを作成します。

  /home ボリュームを作成したくない場合は、autopart は使用しないでください。代わりに、/boot、swap、及び / を指定し、必要に応じて root ボリュームが拡張できるようにします。

- Anaconda はインストール中に視認可能なデバイスを制御できる新しいストレージフィルタリングインターフェースを収納するようになります。このインターフェースは既存の ignoredisk、clearpart、zerombr のコマンドに相当します。ignoredisk はオプションのため、これをキックスタートファイルから除外しても、インストール中にフィルタのユーザーインターフェースは表示されません。このインターフェースを使用したい場合は以下を追加します。

```
ignoredisk --interactive
```

- /tmp/partition-include ファイルの --size=1 --grow オプションは使用できなくなります。適当なデフォルトサイズを指定しなければなりません。パーティションはそれに応じて拡張します。

2.2.2.3. パッケージの変更

これらの変更は %packages セクションに影響します。

- --ignoreDeps と --resolveDeps の引数は削除されています。Anaconda は自動的に依存関係を解決します。ただし、依存関係を満たさないパッケージのインストールは省略されます。

- すべてデフォルトの設定でインストールを行うデフォルトの GUI インストールと全く同じパッケージセットをキックスタートでインストールさせたい場合は以下を追加します。

```
%packages --default
%end
```

- また、オプションとして複数アーキテクチャのインストール用にインストールしたいパッケージのアーキテクチャを指定することもできます。例えば、

```
%packages
glibc.i686
%end
```

  これによりパッケージセットに x86 の glibc パッケージが追加されることになります。互換上 x86 のパッケージを必要とする x86-64 システムで役に立ちます。

- %packages セクションの全てのパッケージとグループを監査して移行することはできません。一部のパッケージやグループは削除されていたり、追加されていたり、また名前がへんこうされているものもあります。詳細についてはリリースノートを参照してください。

2.2.2.4. スクリプトの変更

これらの変更は %pre、%post、及び %traceback のスクリプトの使用に影響します。

- スクリプトの実行中に行われるエラーのログ記録が改善されています。スクリプトがその実行後に削除されなくなるため、スクリプトの確認が可能になります。何が実行されたのかを確認することができるため、スクリプトが動的に生成されるようなシステムでもっとも役に立ちます。また、全スクリプトの stderr と stdout の出力が常にログ記録されます。ただし重要な副作用がひとつあります。スクリプトが対話式のプログラムを使用する場合、そのスクリプトのヘッダに `--logfile=/dev/tty3` を渡す必要があり、これを行わないとプログラムと対話することができません。

2.2.2.5. 構文の変更

中核となるキックスタート構文への変更は非常に稀ですが、知っておくべき重要な構文の変更が 2 つあります。

- %include オプションはファイル名の他にも URL を引数として受け取ることができるようになります。

- %packages、%post、%pre、%traceback のセクションには末尾に %end オプションが必要になります。以前は、これらのセクションに明示的な終了トークンはなく、別のセクションの開始でそのセクションは終了していました。Red Hat Enterprise Linux 6 からは %end の使用は必須となります。%end トークンの無いファイルは失敗することになります。

2.2.2.6. 違いに関する要約

本セクションでは、Red Hat Enterprise Linux 6 でのコマンドとオプションの違いを一覧にします。

削除されたコマンド:

- key

- langsupport

- mouse

非推奨となったコマンド:

- monitor
- xconfig --resolution

追加されたコマンド:

- fcoe
- group
- rescue
- sshpw
- updates

2.2.2.7. pykickstart

pykickstart パッケージには、移行を容易にすることができるユーティリティが含まれています。最新のパッケージがインストールされていることを確認してください。ksverdiff コマンドは開始と終了の構文バージョンを受け取り、2 種類のバージョンで異なるコマンドとオプションを報告します。「新規」、「非推奨」、「削除済み」のコマンドとオプションを示します。例えば、

```
$ ksverdiff --from RHEL5 --to RHEL6

The following commands were removed in RHEL6:
langsupport mouse key

The following commands were deprecated in RHEL6:
monitor

The following commands were added in RHEL6:
sshpw group rescue updates fcoe
 ...
```

ksvalidator コマンドを使用すると、キックスタートファイルの有効性をチェックすることもできます。このコマンドはファイルの有効性を指定したキックスタート構文のバージョンに対してチェックします。ただし、part --ondisk=sdr を指定したがそのようなデバイスは存在しないなど、インストール時にしか発生しない問題については通知できません。使用例を示します。

```
$ ksvalidator --version RHEL6 my-rhel5-ks.cfg
```

2.2.3. ネットワーク設定

本セクションでは、ネットワーク設定に関連したグラフィカルインストーラの動作変更について説明します。

- Anaconda はインストール中のネットワークインターフェースの設定に NetworkManager を使用するようになります。Anaconda の主要なネットワークインターフェース設定の画面は削除されています。インストール中に詳細が必要な場合にのみ、ネットワーク設定の詳細入力がユーザーに求められます。インストール中に使用した設定は、後日、使用できるようシステムに書き込まれます。
- インストーラーの起動に boot.iso を使用する場合、デフォルトのインストールメソッドがすべて選択されていてもソースの選択画面が表示されます。

- PXE ブートの時にインストールメディア用に NFS 経由でマウントした .iso ファイルを使用する場合は、コマンドラインに repo=nfs:server:/path/ を追加します。install.img と product.img ファイルの抽出、nfs:server:/path/images/ ディレクトリへの配置も必要になります。product.img ファイルには他の定義や各種インストールクラスが含まれています。

- 複数のネットワークインターフェースを持つシステムの中には、システムの BIOS で認識されている最初のネットワークインターフェースに eth0 を割り当てないものがあることがあります。これにより、インストーラーが本来 PXE で使用されていたものと異なるネットワークインターフェースを使用しようとする場合があります。この動作を変更するには、pxelinux.cfg/* 設定ファイル内で以下を使用してください。

```
IPAPPEND 2 APPEND
      ksdevice=bootif
```

- この設定オプションで、システム BIOS と PXE が使用するのと同じネットワークインターフェースをインストーラに使用させるようにします。以下のオプションを使用することで、ネットワークスイッチにリンクされていることを検出する最初のネットワークデバイスをインストーラーに使用させるようにします。

```
ksdevice=link
```

2.2.4. 製品のサブスクリプションとコンテンツの更新

Red Hat Enterprise Linux 6 ではコンテンツの配信およびサブスクリプション管理により柔軟で最新のサービスを導入しています。本セクションではコンテンツサービスに対する変更について説明します。

- Red Hat Network ホストの環境はチャンネルベースのサブスクリプションから製品と品質ベースのサブスクリプション使用に更新されます。新しい証明書ベースの RHN では、サブスクリプションとシステムの管理用のクライアントツール群が再設計されているため、新いサブスクリプションとコンテンツ配信ネットワーク (CDN) で動作します。

  従来のチャンネルベースの RHN も RHN Classic で利用できます。

  これらの 2 種類のサブスクリプションサービスは同じプラットフォームで利用可能です。並列テクノロジーですべてのサブスクリプションをいずれの方法でも登録、管理することができます。

  Satellite または Proxy サーバーを使用している環境の場合、従来のチャンネルベースのサブスクリプションシステムが継続して使用されるため、システムの登録は RHN Classic で行われることになります。

- 新しいコンテンツサーバーのオプション、Red Hat Network Classic が firstboot ウィザードに追加されています。これは更新された RHN および CDN ではなく従来のチャンネルベースの RHN を使用します。デフォルトの Red Hat Network オプションは証明書ベースの新しい Red Hat Network 管理プラットフォームを使用します。

- 証明書ベースの RHN と RHN Classic は相互運用が可能です。システムを一方のサービスで登録する場合、他方のサービスはそれを認識できるため警告を発することはありません。システムの登録はいずれか一方のサービスでのみ行ってください。両方のサービスで登録を行うことはできません。

  現在、RHN Classic を使用しているシステムを新しい証明書ベースの Red Hat Network を使用するシステムに直接的に移行させるパスはありません。システムを一方のサービスから他方に移動する場合は 2 通りの方法があります。

  - yum ではなく起動 ISO を使ってシステムを Red Hat Enterprise Linux 6.1 またはそれ以降に更新します。

  - 手作業で RHN Classic からシステムを削除しホストの記録を消してから、そのシステムを証明書ベースの Red Hat Network に Red Hat サブスクリプションマネージャツールを使って登録します。

- クライアントツールの新しいセットとなる Red Hat サブクスリプションマネージャ GUI と CLI は証明書ベースの RHN からサブスクリプションを管理できるよう Red Hat Enterprise Linux 6.1 で提供されています。RHN Classic で管理しているシステムを操作できるよう既存の `rhn_*` ツールも使用できます。

2.3. テキストベースのインストーラ

Red Hat Enterprise Linux 6 のテキストモードのインストールオプションは、以前のバージョンに比べてかなり合理化されています。テキストモードでのインストールでは、以前プロセスの一部であった複雑な手順が省略され、簡単でわかりやすいインストールが行えるようになります。本セクションでは、テキストベースのインストーラを使用する際の動作変更について説明しています。

- Anaconda は、ベースとコアのグループからのみ自動的にパッケージを選択します。インストールプロセスが終了した時点でシステムが正しく動作し、更新や新しいパッケージのインストール準備が整った状態にするにはこれらのパッケージで十分です。

- システムのどこに Red Hat Enterprise Linux をインストールすべきかを指定できるよう anaconda によって以前のバージョンからの初期画面がまだ表示されます。ドライブ全体の使用、既存の Linux パーティションの削除、またはドライブ上の空き領域の使用のいずれかを選択できます。ただし、anaconda はパーティションのレイアウトを自動的にセットするため、この基本レイアウトからのパーティションやファイルシステムの追加または削除に関してユーザーに入力は求めなくなります。インストール時にカスタムなレイアウトを必要とする場合は、VNC 接続かキックスタートインストールでグラフィカルなインストールを実行する必要があります。LVM (論理ボリューム管理)、ファイルシステムの暗号化、サイズ変更が可能なファイルシステムなど高度なオプションについてはグラフィカルモードかキックスタートでしか利用できません。

- Anaconda はテキストベースのインストーラで自動的にブートローダの設定を行うようになります。

- キックスタートを使用したテキストモードのインストールは旧バージョンと同じ方法で実行されます。ただし、パッケージの選択、高度なパーティション設定、ブートローダ設定についてはテキストモードでは自動化されるようになるため、anaconda はこれらの手順で必要とされる情報についてユーザーによる入力を求めることができなくなります。したがって、キックスタートファイルにはパッケージ、パーティションの設定、及びブートローダの設定情報を必ず含ませるようにしてください。いずれかの情報が見つからないと anaconda はエラーメッセージを出力して終了することになります。

ストレージとファイルシステム

3.1. RAID

アップグレード

dmraid セットから mdraid セットへのアップグレード操作はサポートされていません。この種類のアップグレードが試行されると警告が表示されます。既存の mdraid セットからのアップグレード、および新規の mdraid セットの作成は可能です。

Raid セットをアップグレードすると、新しいデフォルトのスーパーブロックが問題を引き起こす恐れがあります。この新規のスーパーブロック形式 (RAID1 /boot パーティションの作成時を除き全デバイスで使用される) はアレイの先頭に位置するようになるため、ファイルシステムや LVM データはすべてそのパーティションの先頭より補正された位置になります。アレイを稼働していないと、LVM およびファイルシステムの mount コマンドは有効なボリュームやファイルシステムデータを持つデバイスを検出できない場合があります。これは意図した動作であり、RAID1 アレイ内に単独のディスクをマウントしたい場合には、その単独ディスクしか持っていないアレイを起動してからそのアレイをマウントする必要があると言うことであり、ベアディスクを直接マウントすることはできません。この変更が行われたのは、再同期が強制されていない場合に、ベアディスクを直接マウントすると警告無しにアレイを破損させる可能性があるためです。

次の再起動で、RAID システムはアレイ内に含まれていなかったディスクは互換性がないと判定する可能性があり、そのデバイスをアレイから切断します。これも正常な動作となります。アレイに他のディスクを再追加する準備ができたら、mdadm コマンドを使用してこのディスクをアレイにホット追加 (動作中の追加) します。この時点でディスクの変更部分 (書き込み予定のビットマップがある場合) またはディスク全体 (ビットマップがない場合) の再同期が行われます。これでアレイは再度同期化されることになります。この時点以降は、アレイは正常に構成されたと判定されるためデバイスがアレイから切断されることはなくなります。

新規のスーパーブロックは mdraid アレイと言う名前の概念をサポートします。アレイ同士を区別するための旧式のアレイ一覧表記 (例、/dev/md0、/dev/md1 など) への依存は無くなりました。アレイには任意の名前 (home、data、opt など) を選択できるようになります。--name=opt オプションを使用して選択した名前でアレイを作成します。どのような名前をアレイに付けてもその名前は /dev/md/ 内に作成されることになります。(ただし、名前として完全パスを与えた場合はそのパスが作成されます。また、「0」などの数字をひとつだけ与えると mdadm は旧式の /dev/mdx スキームを使ってアレイを開始します。) Anaconda インストーラでは現時点ではアレイ名の選択はできません。代わりに過去にアレイがどのように作成されたかをエミュレートする方法としてシンプルな番号形式を使用します。

この新しい mdraid アレイは書き込み予定ビットマップの使用に対応しています。これによりシステムはアレイから問題のある部分を識別することができるようになるため、不正なシャットダウンが発生した場合にはその問題となる部分のみが再同期を必要とし、ディスク全体を再同期する必要がなくなります。このため再同期に必要な時間が劇的に減少します。新たに作成したアレイには書き込み予定ビットマップが必要に応じて自動的に追加されます。例えば、swap に使用されるアレイおよび非常に小規模のアレイ (/boot アレイなど) の場合は書き込み予定ビットマップを持たせてもの利点はありません。デバイス上で mdadm --grow コマンドを使用してアップグレードを完了した後に以前から既存のアレイに書き込み予定ビットマップを追加することはできますが、書き込み予定ビットマップにより若干のパフォーマンス劣化が発生します (チャンクサイズ 65536 のビットマップで約 3% から 5% のダウンですが、8192 位の小さいサイズのビットマップチャンクの場合は 10% またはそれ以上になることがあります)。アレイに書き込み予定ビットマップを追加する場合には適度に大きめのチャンクサイズを維持するのが最適ということになります。推奨サイズは 65536 です。

3.2. ext4

ext3 からの移行

ext4 を使用する予定の方は、パーティションを新規にフォーマットされることをお勧めします。ただし、レガシーな ext3 パーティションから ext4 に変換したい場合には、`ext4migrate` ブートオプションを使って Red Hat Enterprise Linux 6 をインストールしても構いません。現在の ext3 のパーティションに存在するデータは ext4 の機能やその他の変更を使用しないため、ext3 から ext4 への変換を行っても ext4 の利点がすべて利用できるわけではないため注意してください。新規のデータについては ext4 の利点をすべて使用できます。このブートオプションを渡して ext4 に移行するのはお勧めしていません。移行を行う場合には、先にファイルシステムのバックアップをとっておくことを強くお勧めします。

動作の変更

Red Hat Enterprise Linux 6 では ext4 に完全対応するようになるため、新規のインストールに対しては ext4 がデフォルトのファイルシステムとなります。本セクションでは、この新しいファイルシステムにより導入される主な動作変更について説明しています。

- 収納されている GRUB ブートローダのバージョンは ext4 パーティションに対して完全対応となります。また、インストーラを使用する際も ext4 パーティション上にあらゆる `/boot` ファイルシステムを配置することができます。
- 収納されている e2fsprogs パッケージのバージョンは ext4 と完全互換になります。
- e4fsprogs パッケージを同梱した Red Hat Enterprise Linux 5.3 では、ext4 ファイルシステムを作成すると `ext4dev` ファイルシステムタイプになることがありました。これらのファイルシステムを開発バージョンとして識別する `test_fs` 機能フラグは、コマンド `tune2fs -E ^test_fs` を使用して削除することができます。これを行うことにより、ファイルシステムが通常の ext4 ファイルシステムとして認識されるようになります。

3.3. fusecompress

fusecompress

Fusecompress は特別な権限を持たないユーザーがマウントできる圧縮型ファイルシステムです。Red Hat Enterprise Linux 6 には、更新バージョンが含まれていてバグがいくつか修正されていますがオンディスク形式が変更されています。既存の Fusecompress ファイルシステムを使用しているユーザーはデータを新しい形式に移行する必要があります。アップグレードの前に解凍を行わない限り、fusecompress_offline1 パッケージが必要になります。

3.4. blockdev

blockdev

`blockdev --rmpart` コマンドオプションはもうサポートがありません。`partx(8)` と `delpart(8)` のコマンドが今回、この機能を提供します。

ネットワーク設定とサービス

4.1. インターフェースと設定

NetworkManager

Red Hat Enterprise Linux 6 はネットワークのインターフェースを設定する際にデフォルトで NetworkManager を使用します。

Infiniband

Infiniband サポート (特に `openib` 開始スクリプトと `openib.conf` ファイル) は Red Hat Enterprise Linux 5 では openib パッケージによって提供されていました。このパッケージ名はその機能をより正確に反映するために Red Hat Enterprise Linux 6 で変更されています。Infiniband 機能は rdma パッケージ内で配布されるようになります。そのサービスは `rdma` と言う名前になり、設定ファイルは `/etc/rdma/rdma.conf` に配置されます。

4.2. サービスの初期化

xinetd

Xinetd を使用してオンデマンドでネットワークサービスを開始します。xinetd 内の変更は open ファイルディスクリプタの許容限度に関連してきます。

- リスニングのメカニズムが `select()` から `poll()` に変更されています。この変更により、xinetd で使用される open ファイルディスクリプタの限度は変更可能になります。
- ファイルディスクリプタの限度がサービス単位でも変更できるようになります。`rlimit_files` ディレクティブを使ってサービスの設定ファイル内で行います。値は正の整数か UNLIMITED になります。

ランレベル

Red Hat Enterprise Linux 6 では、カスタムのランレベル 7、8、および 9 に対応しなくなるため使用できなくなります。

Upstart

Red Hat Enterprise Linux 6 では、sysvinit パッケージの init はイベントベースの init システムである Upstart に入れ替わっています。このシステムはブート時のサービスやタスクの開始、シャットダウン時の停止、システムの稼働中にはサービスやタスクの監視を行います。Upstart 自体に関する詳細は `init(8)` man ページを参照してください。

プロセスは Upstart でジョブとして認識されており、`/etc/init` ディレクトリ内のファイル群によって定義されます。Upstart については man ページ内詳しくが記載があります。コマンドの概要については `init(8)`、ジョブの構文については `init(5)` に説明があります。

Red Hat Enterprise Linux 6 の Upstart では、以下のような動作変更があります。

- `/etc/inittab` ファイルは非推奨となるため、initdefault 行で デフォルトのランレベルを設定する場合にしか使用されなくなります。その他の設定は `/etc/init` ディレクトリ内の upstart のジョブで行われます。
- アクティブな tty コンソール数は `/etc/sysconfig/init` 内の ACTIVE_CONSOLES 変数でセットされるようになります。この変数は `/etc/init/start-ttys.conf` ジョブによって読み込まれます。デフォルト値は ACTIVE_CONSOLES=/dev/tty[1-6] となり、tty1 から tty6 上で getty を起動します。

- シリアルコンソールがプライマリのシステムコンソールになる場合、シリアル getty も自動的に設定されます。旧リリースでは、/etc/inittab を編集する kudzu によって行われていました。Red Hat Enterprise Linux 6 では、プライマリのシリアルコンソールの設定は /etc/init/serial.conf で行われます。
- デフォルト以外のシリアルコンソールで実行している getty を設定するには、/etc/inittab を編集するのではなく、Upstart ジョブを記述する必要があります。例えば、ttyS1 上の getty を設定したい場合は、以下のようなジョブファイル (/etc/init/serial-ttyS1.conf) になるでしょう。

```
# This service maintains a getty on /dev/ttyS1.

start on stopped rc RUNLEVEL=[2345]
stop on starting runlevel [016]

respawn
exec /sbin/agetty /dev/ttyS1 115200 vt100-nav
```

旧リリースと同じく、この getty で root ログインを許可したい場合は /etc/securetty 内に必ず ttyS1 があることを確認してください。

Upstart への移動のため、/etc/shutdown.allow を使用してマシンのシャットダウンを行うことができるユーザーを定義する方法はサポートされなくなります。

4.3. IP テーブル/ファイアウォール

IPTable には SECMARK ターゲットモジュールが含まれています。SELinux などのセキュリティサブシステムで使用されるパケットに関連付けられるセキュリティマーク値の設定に使用されます。mangle 表内でのみ有効となります。使用例については以下を参照してください。

```
iptables -t mangle -A INPUT -p tcp --dport 80 -j SECMARK --selctx ¥ system_u:object_r:httpd_packet_t:s0
```

4.4. Apache HTTP Server

Red Hat Enterprise Linux 6 への移行の際、注意すべき Apache HTTP Server の違いについて以下に示します。

- mod_file_cache、mod_mem_cache、mod_imagemap のモジュールには対応しなくなります。
- Charset=UTF-8 オプションがデフォルトの IndexOptions ディレクティブに追加されています。UTF-8 以外の文字セットによるディレクトリ表示を必要とする場合には (mod_autoindex で生成されるものなど)このオプションを変更してください。
- distcache で配信されるセッションキャッシュは mod_ssl では対応しなくなります。
- プロセス ID (pid) のデフォルトの位置が /var/run から /var/run/httpd に移動しています。
- mod_python パッケージはアップストリームでの開発が中止されたため同梱されなくなります。Red Hat Enterprise Linux 6 では代替として mod_wsgi を同梱させ、WSGI インターフェースによる Python スクリプト記述に対応します。

4.5. PHP

PHP の変更点を以下に示します。

- PHP はバージョン 5.3 にアップグレードされています。互換性に関する問題によりスクリプトの更新を必要とする場合があります。詳細については以下の URL を参照してください。

  - http://php.net/manual/migration52.php
  - http://php.net/manual/migration53.php
- デフォルト設定に次の変更が行われています (/etc/php.ini)。
  - error_reporting が E_ALL & ~E_DEPRECATED に設定されるようになります (以前は (previously E_ALL)。
  - short_open_tag が Off に設定されるようになります (以前は On)。
  - variables_order が GPCS に設定されるようになります (以前は EGPCS)。
  - enable_dl が Off に設定されるようになります (以前は On)。
- mime_magic、dbase、ncurses の拡張は配信されなくなります。

4.6. BIND

BIND 設定には主要な変更点がいくつかあります。

- デフォルトの ACL 設定 – Red Hat Enterprise Linux 5 では、デフォルトの ACL 設定では全ホストに対して問い合わせを許可し、また再帰的な問い合わせを提供していました。Red Hat Enterprise Linux 6 のデフォルトでは、信頼すべきデータに対する問い合わせは全ホストが行えますが、再帰的な問い合わせについてはローカルネットワークのホストしか行うことはできません。
- 新しい allow-query-cache オプション – allow-recursion オプションは非推奨となり、このオプションが使用されるようになっています。信頼できないデータ全て (再帰的な検索や root ネームサーバーのヒントなど) を含むサーバーのキャッシュへのアクセスを制御します。
- Chroot 環境の管理 – bind-chroot-admin スクリプトは chroot 以外の環境から chroot 環境への symlink の作成に使用されていましたが、今後は非推奨となるため存在しなくなります。代わりに設定は chroot 以外の環境で直接管理ができるようになります。このため、ファイルがまだ chroot に存在しない場合、named の起動中に init スクリプトにより必要なファイルが chroot 環境に自動的にマウントされます。
- /var/named ディレクトリのパーミッション – /var/named ディレクトリには書き込みができなくなります。書き込みを必要とするゾーンファイル (動的 DNS ゾーン、DDNS など) はすべて書き込み可能となる新しいディレクトリ /var/named/dynamic に配置するようにしてください。
- dnssec [yes|no] オプションはなくなります。グローバルなオプション dnssec [yes|no] は、新たに dnssec-enable と dnssec-validation の 2 種類のオプションに分割されています。dnssec-enable オプションは DNSSEC サポートを有効にします。dnssec-validation オプションは DNSSEC 検証を有効にします。dnssec-enable を再帰的サーバーで「no」に設定すると、DNSSEC 検証を行う別のサーバーではフォワーダーとしては使用できないということになります。いずれのオプションもデフォルトでは「yes」にセットされています。
- rndc 管理ユーティリティを使用している場合は、/etc/named.conf 内で controls ステートメントを指定する必要がなくなります。named サービスは自動的にループバックデバイスを介して制御の接続を許可し、named と rndc の両方がインストール時に生成された同じシークレットキーを使用します (/etc/rndc.key 内にある)。

デフォルトのインストールでは、BIND は DNSSEC 検証を有効するとインストールされるため、ISC DLV レジスタを使用します。つまり、キーを ISC DLV レジスタ内に持つ署名付きドメイン (gov.、se.、cz. など) はすべて再帰的サーバー上で暗号化検証が行われます。キャッシュポイゾニングの試行により検証が失敗した場合、エンドユーザーにはこの偽のなりすましデータは与えられません。DNSSEC の導入は広範囲に実装されるようになっている機能であり、エンドユーザーにとってインターネットをより安全に利用できるようにするための重要

なステップとなります。このため Red Hat Enterprise Linux 6 では DNSSEC に完全対応しています。前述のように DNSSEC 検証は /etc/named.conf 内の dnssec-validation オプションを使って制御します。

4.7. NTP

NTP (Network Time Protocol) はコンピュータシステムのクロックをネットワーク経由で同期するのに使用されます。Red Hat Enterprise Linux 6 では、デフォルトの設定ファイル /etc/ntp.conf に以下のコメントアウトされている行が含まれるようになります。

```
#server 127.127.1.0 # local clock
#fudge 127.127.1.0 stratum 10
```

この設定は、NTP サーバーか参照クロックに明確に同期される場合にのみ、ntpd がネットワーククライアントに時刻情報を配信するということになります。同期されない場合でも ntpd が時間情報を提供するようにするには、上記 2 行のコメントを解除し有効にする必要があります。

また、ntpd に -x オプションを付けて (/etc/sysconfig/ntpd ファイル内の OPTIONS 内) 開始した場合または /etc/ntp/step-tickers 内にサーバーが指定されている場合、サービスは起動前に ntpdate コマンドを実行しなくなります。ntpd サービスとは別に単独で有効にできる ntpdate サービスが使用できるようになります。他のサービスが起動前に正確な時刻を必要とする場合にのみ使用すべきサービスのため、この ntpdate サービスはデフォルトでは無効になっています。これ以外の場合、後で時刻調節が ntpd によって実施された時に正常に機能しなくなります。

デフォルトの NetworkManager 構成でこのサービスを実行すると問題に遭遇する場合があります。Red Hat Enterprise Linux 導入ガイドに説明してあるように、NETWORKWAIT=1 を /etc/sysconfig/network に追加する必要があるかも知れません。

4.8. Kerberos

Red Hat Enterprise Linux 6 では、Kerberos クライアントとサーバー (KDC を含む) は、デフォルトでは des-cbc-crc、des-cbc-md4、des-cbc-md5、des-cbc-raw、des-cbc-raw、des3-cbc-raw、des-hmac-sha1、及び arcfour-hmac-exp 暗号のキーを使用しません。デフォルトでは、クライアントはこれらのタイプのキーを持つサービスへの認証はできません。

ほとんどのサービスのキータブに新しいキーセット (より強力な暗号で使用するキーも含む) を追加することができるため、ダウンタイムを発生することがありません。また、サービスのキーを与えるチケットも同様に kadmin の cpw -keepold コマンドを使用してより強力な暗号で使用するキーを含んだセットに更新することができます。

弱い暗号の使用を継続する必要のあるシステムは、一時的な迂回策として /etc/krb5.conf ファイル内の libdefaults セクションに allow_weak_crypto オプションを必要とします。この変数は、デフォルトでは false にセットしてあるため、このオプションを有効にしないと認証は失敗します。

```
[libdefaults]
allow_weak_crypto = yes
```

また、使用可能な共有ライブラリとしても、アプリケーション内でのサポートされている認証メカニズムとしても、Kerberos IV のサポートは削除されています。ロックアウトポリシー用に新たに追加されたサポートはデータベースダンプ形式に変更が必要となります。旧式の KDC が使用できる形式でデータベースをダンプする必要があるマスターの KDC には、-r13 オプションを付けた kdb5_util の dump コマンドを実行させる必要があります。

4.9. Mail

4.9.1. Sendmail

Red Hat Enterprise Linux 5 の一部のリリースでは、sendmail MTA (Mail Transport Agent) がデフォルトで外部ホストからのネットワーク接続を受け取っていました。Red Hat Enterprise Linux 6 では、sendmail はデフォルトではローカルのシステム (localhost) からの接続しか受け付けません。リモートホストに対してサーバーとして動作できる機能を sendmail に与える場合は、以下の手順のいずれかを行います。

- /etc/mail/sendmail.mc を編集して、DAEMON_OPTIONS 行を変更しネットワークデバイス上でもリッスンするようにします
- /etc/mail/sendmail.mc の DAEMON_OPTIONS 行に # 印を入れてコメントアウトします。

上記いずれかの変更を反映させるには、sendmail-cf パッケージをインストールしてから /etc/mail/sendmail.cf を再生成します。次のコマンドを実行して行います。

```
su -c 'yum install sendmail-cf'
su -c 'make -C /etc/mail'
```

4.9.2. Exim

Exim は Red Hat Enterprise Linux 6 から削除されています。Postfix がデフォルトで推奨される MTA になります。

4.9.3. Dovecot

Dovecot の設定

Dovecot 2.x の構成が変更されています。マスター設定ファイル /etc/dovecot.conf が /etc/dovecot/dovecot.conf に移動されているため、Dovecot 構成の他の部分も /etc/dovecot/conf.d/*.conf に移動されています。構成の大部分は同じであり、この新しいバージョンとの互換性があります。以下のコマンドを使用するとご使用の構成をテストして、この新バージョンで名前変更されているオプション、削除されているオプション、変更されているオプションを一覧表示させることができます。

```
doveconf [-n] -c /old/dovecot.conf
```

4.10. MySQL®

DBD ドライバー

MySQL DBD ドライバーは二重ライセンスとなり、ライセンスに関連する問題は解決されました。結果、apr-util-mysql パッケージが Red Hat Enterprise Linux 6 のソフトウェアレポジトリに収納されるようになります。

4.11. PostgreSQL

データベースをアップグレードする

PostgreSQL 8.4 (postgresql84-* パッケージ) を使用していた既存の Red Hat Enterprise Linux 5 からアップグレードを行っている場合は、Red Hat Enterprise 6 PostgreSQL パッケージが当座の代替として動作するようになります。

ただし、PostgreSQL 8.1 (postgresql-* パッケージ) またはこれより旧式のバージョンを使用していた Red Hat Enterprise Linux 5 インストールからのアップブレードで、維持しておく必要があるデータベースコンテンツが既存している場合、データ形式が変更されているためダンプに従い記載されている手順を再ロードする必要があります (http://www.postgresql.org/docs/8.4/interactive/install-upgrading.html)。必ずダンプの手順を 先に 行ってから、Red Hat Enterprise Linux 6 のアップグレードを行ってください。

その他の変更

PostgreSQL 8.1 から 8.4 への移行に関連して考えられるアプリケーションの互換問題については、http://wiki.postgresql.org/wiki/WhatsNew84 を参照してください。

4.12. Squid

Squid は 3.1 に更新されているため、ネイティブの IPv6 サポートを提供するようになります。設定ファイル `/etc/squid/squid.conf` が大幅に短縮されています。Squid 3.1 の設定オプションが変更され、一部の旧バージョンとは完全な下方互換がありません。設定及びその他の変更についての詳細については、Squid 3.1 リリースノートを参照してください (http://www.squid-cache.org/Versions/v3/3.1/RELEASENOTES.html)。

Squid は ncsa_auth ヘルパーと pam_auth ヘルパーによるユーザー認証機能を提供します。これらのヘルパーのパーミッションが Red Hat Enterprise Linux 6 で変更されています。以前のリリースでは、認証に必要なシステムファイルへのアクセスに高位のパーミッションが必要であったため、ncsa_auth と pam_auth 用の setuid フラグが有効にされていました。Red Hat Enterprise Linux 6 では、Squid はこれらのヘルパーに setuid フラグの設定を必要としなくなります。setuid フラグの実行時にセキュリティに関する危険性が発生するためこの変更が加えられました。通常の機能はこれらのフラグを設定することなく維持されています。

4.13. Bluetooth

オンデマンドの Bluetooth サービス

Bluetooth デバイスに対応するため、Red Hat Enterprise Linux の旧バージョンではデフォルトで Bluetooth バックグランドサービスが起動していました。本リリースでは、Bluetooth サービスは必要に応じてオンデマンドで起動し、デバイスの使用が停止されてから 30 秒後に自動で停止します。これにより全体的な初期起動時間とリソース消費が低減されます。

4.14. Cron

Vixie cron と Cronie

Red Hat Enterprise Linux 6 には vixie-cron の代替として cronie パッケージが含まれています。これらのパッケージの主な違いは、定期的なジョブ (毎日、毎週、毎月) の実行方法にあります。Cronie は `/etc/anacrontab` ファイルを使用します。このファイルのデフォルト状態を以下に示します。

```
# the maximal random delay added to the base delay of the jobs
RANDOM_DELAY=45

# the jobs will be started during the following hours only
START_HOURS_RANGE=3-22

# period in days   delay in minutes   job-identifier      command

1   5  cron.daily nice run-parts /etc/cron.daily
7   25  cron.weekly nice run-parts /etc/cron.weekly
@monthly  45  cron.monthly nice run-parts /etc/cron.monthly
```

これらの定期的なジョブはランダム遅延を含め 03:00 から 22:00 の時間帯で毎日一回実行されます。例えば、cron.daily は 5 分の強制遅延とそれに加えて 0 分から 45 分のランダム遅延があります。4 時から 5 時の間で遅延なくジョブを実行することもできます。

```
RANDOM_DELAY=0 # or don't use this option at all

START_HOURS_RANGE=4-5

# period in days   delay in minutes   job-identifier      command
1   0  cron.daily nice run-parts /etc/cron.daily
7   0  cron.weekly nice run-parts /etc/cron.weekly
@monthly  0  cron.monthly nice run-parts /etc/cron.monthly
```

cronie の機能を以下に示します。

- /etc/anacrontab 内でのジョブ開始のランダムな遅延が可能です。
- 定期的なジョブの時間幅を /etc/anacrontab で定義することができます。
- 各 cron テーブルに CRON_TZ 変数を付けそれ自体に定義したタイムゾーンを持たせることができます。
- デフォルトでは cron デーモンは inotify でテーブル内の変更をチェックします。

cronie と cronie-anacron についての詳細については Red Hat Enterprise Linux 導入ガイドを参照してください。

4.15. ログ記録

dateext オプションが /etc/logrotate.conf 内でデフォルトで有効になります。このオプションは、日付 (YYYYMMDD 形式) を表す拡張を追加することによりログファイルの古いバージョンをアーカイブします。以前はファイルに番号が追記されていました。

コマンドラインツール

本セクションでは、Red Hat Enterprise Linux 6 の各コマンドラインツールの動作変更について説明しています。

5.1. Grep

grep コマンドの動作では、大文字と小文字の文字列検索について変更が加えられています。[a-z] 形式の範囲検索の使用は、LC_COLLATE 変数に依存します。

LC_COLLATE=C とセットすると旧来の動作を保存することができるため、この方法で範囲検索を実行した場合に正しい結果を得ることができます。しかし、Red Hat Enterprise Linux 6 では、範囲検索には [[:lower:]]、[[:upper:]] 形式の使用をお勧めします。

この変更は出力に多大な影響を及ぼします。そのため、正しい結果を引き続き得ることができるようスクリプトとプロセスの再確認を行ってください。

5.2. Sed

-i オプションを付けて sed コマンドを使用すると読み取り専用ファイルの内容、および他の保護ファイルの削除ができます。任意のファイルのパーミッションはそのファイルに対して行うことができる動作を定義し、任意のディレクトリのパーミッションはそのディレクトリ配下のファイル群の一覧に対して行うことができる動作を定義します。このことから sed では読み取り専用のディレクトリ内にある書き込み可能なファイル上では -i を使用できません。また、このようなファイル上で -i オプションを使用すると、シンボリックリンクやハードリンクを破棄することになります。

5.3. Pcre

pcre パッケージは 7.8 に更新されています。これには以下のような動作変更が含まれています。

- UTF-8 チェックは RFC 2279 ではなく RFC 3629 を参照するようになります。これにより、受け取る文字列での制約が厳しくなります。例えば、UTF-8 文字の序数値は 0x0010FFFF に制限されるようになります。

  ```
  $ echo -ne "¥x00¥x11¥xff¥xff" | recode UCS-4-BE..UTF8 | pcregrep --utf-8 '.'
  pcregrep: pcre_exec() error -10 while matching this line:
  ```

  RFC の詳細については http://tools.ietf.org/html/rfc3629#section-12 を参照してください。

- PCRE の旧バージョンでコンパイルした保存済みのパターンは再コンパイルが必要です。これは事前にコンパイルした PCRE 式を外部メモリ (1 ファイルなど) に直列化して後それを読み込むようなアプリケーションに影響を及ぼします。通常、大規模のスパムフィルタなどパフォーマンスを向上させる目的で行われます。

5.4. シェル

シェルのバイナリファイル群の場所が変更されています。例えば、bash バイナリと ksh のバイナリは /usr/bin からなくなります。いずれのバイナリも /bin に配置されるようになります。バイナリの新しい場所をポイントするようスクリプトを更新する必要があります。

5.5. Nautilus

nautilus-open-terminal パッケージでは、右クリックによる Open Terminal オプションで現在のディレクトリ内に新規のターミナルウィンドウを開きます。以前は、Desktop からこのオプションを選択すると、新規のターミ

ナルウィンドウの位置はユーザーのホームディレクトリに設定されていました。Red Hat Enterprise Linux 6 では、デフォルトの動作では Desktop ディレクトリで開きます (~/Desktop/)。以前の動作を有効にするには、次のコマンドを使って desktop_opens_home_dir GConf boolean を true にセットします。

```
gconftool-2 -s /aps/nautilus-open-terminal/desktop_opens_dir --type=bool true
```

デスクトップ

- Red Hat Enterprise Linux 6 では、GUI コンソールが tty7 から tty1 に移動しています。

GDM の設定

GDM 設定がいくつか GConf 内で管理されるようになります。

GDM のデフォルト Greeter はシンプルな Greeter と呼ばれ、GConf で設定します。デフォルト値は gdm-simple-greeter.schemas ファイル内の GConf に保存されています。この値を編集する場合は、`gconftool2` か `gconf-editor` を使用します。Greeter には以下のオプションがあります。

- /apps/gdm/simple-greeter/banner_message_enable

  ```
  false (boolean)
  ```

  バナーのテキストメッセージが表示されるかどうかを制御します。

- /apps/gdm/simple-greeter/banner_message_text

  ```
  NULL (string)
  ```

  greeter ウィンドウで表示するバナーのテキストメッセージを指定します。

- /apps/gdm/simple-greeter/logo_icon_name

  ```
  computer (string)
  ```

  greeter ロゴに使用するテーマアイコン名を設定します。

- /apps/gdm/simple-greeter/disable_restart_buttons

  ```
  false (boolean)
  ```

  ログインウィンドウに再起動ボタンを表示するかどうかを制御します。

- /apps/gdm/simple-greeter/wm_use_compiz

  ```
  false (booleans)
  ```

  metacity ではなく compiz をウィンドウマネージャとして使用するかどうかを制御します。

GConf を使用するとプラグインも無効にすることができます。例えば、サウンドのプラグインを無効にしたい場合、以下のキーを設定解除します: `/apps/gdm/simple-greeter/settings-manager-plugins/sound/active`

セキュリティと認証

本章では、SELinux、SSSD、LDAP、Checksum、PAM などのセキュリティおよび認証に関する動作変更について説明しています。

7.1. SELinux

sshd デーモンは限定サービスとなります。

7.2. SSSD

SSSD (System Security Services Daemon) は プロバイダ と呼ばれるリモートの ID 認証メカニズムへのアクセスを提供します。SSSD によりプロバイダを SSSD のバックエンドとしてプラグインできるようになるため、ローカルやネットワーク上の ID 認証ソースを抽出して、あらゆる ID データプロバイダをプラグインすることができます。ドメイン はユーザー情報を含んでいるデータベースであり、プロバイダの識別情報のソースとして機能することができます。複数の識別情報プロバイダに対応し、2 つ以上の識別サーバーを別々のユーザーネームスペースとして動作させることができます。収集された情報は標準の PAM および NSS インターフェースでフロントエンド上のアプリケーションに対して利用することができます。

SSSD は複数のサービスから成るセットとして稼働し、それを使用するアプリケーションには依存しません。このため、アプリケーションはリモートのドメインにそれ自体を接続する必要がなくなります。また、どれが使用されているを認識する必要もありません。ID およびグループメンバーシップ情報の強固なローカルキャッシング機能により、ID のソース (LDAP、NIS、IPA、DB、Samba など) に関係無く運用が可能になり、パフォーマンスが向上されるため、オフラインで稼働しているためオンライン認証が利用できない場合であっても認証を行うことができます。また、SSSD では同じタイプの複数のプロバイダ (例、複数の LDAP プロバイダ) を使用できるため、ドメイン修飾型の ID 要求を別のプロバイダで解決することができます。詳細については、「Red Hat Enterprise Linux 6 導入ガイド」でご覧ください。

7.3. LDAP

OpenLDAP

OpenLDAP サービスに必要となる設定が Red Hat Enterprise Linux 6 で変更になっています。旧バージョンでは、slapd は /etc/openldap/slapd.conf から設定していました。Red Hat Enterprise Linux 6 の slapd 設定は特殊な LDAP ディレクトリ (/etc/openldap/slapd.d/) 内事前定義済みのスキーマと DIT (ディレクトリ情報ツリー) と共に格納されるようになります。この設定スキーマの詳細については、openldap.org[1] でご覧になれます。以下のセクションでは、旧来の設定ファイルを新しいディレクトリで機能するように変換する方法について例をあげて説明しています。

7.3.1. slapd 設定を変換する

この例では、旧来の slapd 設定から変換するファイルが /etc/openldap/slapd.conf にあり、OpenLDAP 設定の新しいディレクトリは /etc/openldap/slapd.d/ にあると仮定しています。

- 新しいディレクトリの /etc/openldap/slapd.d/ の内容を削除します。

```
# rm -rf /etc/openldap/slapd.d/*
```

- slaptest を実行して設定ファイルの有効性をチェックし、新しい設定ディレクトリを指定します。

[1] http://www.openldap.org/doc/admin24/slapdconf2.html#Configuration%20Layout

```
slaptest -f /etc/openldap/slapd.conf -F /etc/openldap/slapd.d
```

- 新しいディレクトリ上でパーミッションを設定します。

```
chown -R ldap:ldap /etc/openldap/slapd.d
```

```
chmod -R 000 /etc/openldap/slapd.d
```

```
chmod -R u+rwX /etc/openldap/slapd.d
```

- サービスが新しい設定ディレクトリ内で機能していることを確認してから、古い設定ファイルを削除します。

```
rm -rf /etc/openldap/slapd.conf
```

7.4. Checksum

Red Hat Enterprise Linux は以前より多くの場所でデータ検証と認証用の SHA-256 ダイジェストアルゴリズムを使用するようになり、暗号的に貧弱な SHA-1 アルゴリズムと MD5 アルゴリズムからアップグレードしています。

7.5. PAM (プラグ可能な認証モジュール)

PAM サービス用の共通設定は /etc/pam.d/system-auth-ac ファイルに配置されています。

認証モジュールは追加の PAM 設定ファイルにも書き込まれるようになります。追加の PAM 設定ファイルは /etc/pam.d/password-auth-ac、etc/pam.d/smartcard-auth-ac、/etc/pam.d/fingerprint-auth-ac になります。

sshd の PAM モジュール及び ftpd などの他のリモートサービスが Red Hat Enterprise Linux6 では /etc/pam.d/system-auth ではなく /etc/pam.d/password-auth ファイルを収納するようになります。

7.6. システムのユーザー

Red Hat Enterprise Linux 6 では、静的に割り当てられた UID/GID 番号のしきい値 (/usr/share/doc/setup-*/uidgid ファイル内の setup パッケージで定義される) が 100 から 200 に増えています (Red Hat Enterprise Linux 3、4、5 では 100)。この変更は、動的または静的に割り当てられた 100 から 200 までの UID/GID を持つシステムに影響を与えるため、一部のアプリケーションのインストールおよび実行に障害が発生する原因となる場合があります。

UID/GID の動的な割り当ては Red Hat Enterprise Linux 6 では 499 以下の範囲になります。setup パッケージによって強制される予約番号を持たない静的なシステムユーザーを作成する場合には、300 またはそれ以上の UID/GID 番号の使用をお勧めします。

カーネル

8.1. カーネル

mkinitrd は dracut ツールに置き換えられました。また、/etc/modprobe.conf ファイルはカーネルモジュールの管理ではデフォルトでは使用されなくなります。ただし、手作業で作成した場合にはまだ使用することができます。dracut ツールの使用例については以下を参照してください。

```
# mv /boot/initramfs-$(uname -r).img /boot/initramfs-$(uname -r)-old.img
# dracut --force /boot/initramfs-$(uname -r).img $(uname -r)
```

パッケージとドライバーに関する変更点

収納されているパッケージとシステムドライバー群は Red Hat Enterprise Linux のリリースの度に定期的な変更を受けます。変更が行われる理由として、「新しい機能を提供するためオペレーティングシステムにパッケージやドライバーが追加または更新される、パッケージやドライバーが旧式で無効なハードウェア対応となるため削除される」、「パッケージやドライバーのアップストリームプロジェクトが維持されなくなる、またはハードウェアベンダーによるハードウェア固有のパッケージやドライバーの対応が行われなくなるため削除される」などの場合があるためです。

本章では、Red Hat Enterprise Linux 6 で更新されたパッケージとドライバー、新規となるパッケージとドライバ、また非推奨となっているものや廃止されたもの (削除されたもの) についても一覧で記載します。

9.1. システム設定ツールの変更点

system-config-bind

system-config-bind ツールは非推奨となっているため削除され、代替ツールはありません。Red Hat Enterprise Linux 6 では、ネームサーバーの設定を編集する場合は `named.conf` ファイルから手作業で行うことをお勧めします。BIND に関する総合的なガイドは bind パッケージの一部として `/usr/share/doc/bind-x.y.z` 内にインストールされます。また、設定例については `/usr/share/doc/bind-x.y.z/sample` ディレクトリをご覧ください。ただし、旧バージョンの system-config-bind ツールは標準の BIND 設定を生成するため、ご使用の環境によっては古い設定ファイルを正しい場所に移動して十分なテストを実行すれば Red Hat Enterprise Linux 6 にある BIND のバージョンに移行することも可能です。

system-config-boot

system-config-boot ツールでは GRUB ブートローダのグラフィカルな設定が可能でした。Red Hat Enterprise Linux 6 では、グラフィカルな設定は非推奨となっているため削除され、代替となるものはありません。多くのユーザーにとってはデフォルトの GRUB 設定で十分なはずですが、手作業による変更が必要な場合には `/boot/grub` ディレクトリ内の `grub.conf` ファイルにブートの設定がありますので、ここで変更することができます。GRUB の設定に関する詳細については GRUB のホームページ http://www.gnu.org/software/grub/ をご覧ください。

system-config-cluster

system-config-cluster ツールは非推奨となっているため削除され、代替のツールはありません。ricci と luci (Conga プロジェクトのパッケージ) の使用をお薦めします。

system-config-display

system-config-display ツールは、対応デスクトップとなる GNOME および KDE の両方で XRandr 設定ツールに入れ替わっています。ディスプレイ管理は以下のメニューオプションのいずれかで動的に行われるようになるため、デフォルトの X サーバーインストール内には明示的な設定ファイル (`xorg.conf`) はありません。

GNOME: システム → 設定 → ディスプレイ

KDE: システムの設定 → コンピュータ管理 → ディスプレイ

注記: ディスプレイ設定にコマンドラインユーティリティ (`xrandr`) を使用することもできます。詳細については、`xrandr --help` コマンドまたは `man xrandr` コマンドで man ページを参照してください。

system-config-httpd

system-config-httpd ツールは非推奨となっているため削除され、代替となるツールはありません。Web サーバーの設定は手作業で行ってください。設定は `/etc/httpd` ディレクトリで行うことができます。主要設定

ファイルは /etc/httpd/conf/httpd.conf にあります。このファイル内には、ほとんどのサーバー設定に関して詳細な説明が記載されています。ただし、必要であれば Apache Web サーバーに関して詳細に記載されているドキュメントが httpd-manual パッケージ内に収納されています。

system-config-lvm

system-config-lvm ツールは非推奨となっています。論理ボリュームの管理については gnome-disk-util または lvm のツールを使って行ってください。

system-config-netboot

system-config-netboot ツールは非推奨となっているため削除され、代替のツールはありません。Red Hat Satellite の使用をお薦めします。

system-config-network

system-config-network ツールは、最新でパワフルなネットワーク設定ツールとある NetworkManager に入れ替わっています。NetworkManager-applet (nm-applet) はデフォルトでいずれの対応デスクトップ環境にもインストールされ、システムトレイのパネルエリアに配置されます。詳細については、NetworkManager のホームページをご覧ください (http://projects.gnome.org/NetworkManager/)。

system-config-nfs

system-config-nfs ツールは非推奨となっているため削除され、代替のツールはありません。NFS サーバーの設定については手作業で行ってください。

system-config-rootpassword

system-config-rootpassword ツールは、ユーザーの管理と設定を行うことができるパワフルなツール system-config-users ツールに入れ替わっています。root パスワードは system-config-users ツール内で設定できます。個人設定ダイアログ内の 「システムのユーザーとグループを隠す」 オプションのチェックを外します。root ユーザーが主要一覧に表示されるようになり、他のユーザーと同じようにパスワードを変更することができるようになります。

system-config-samba

system-config-samba ツールは非推奨となるため削除され、代替のツールはありません。SMB サーバーの設定は手作業で行ってください。

system-config-securitylevel

system-config-securitylevel ツールは削除されています。ユーザーの方は system-config-firewall ツールを使用してください。

system-config-soundcard

system-config-soundcard ツールは削除されています。サウンドカードの検出と設定は自動的に行われます。

system-config-switchmail

system-config-switchmail ツールは非推奨となっているため削除され、代替のツールはありません。Red Hat Enterprise Linux 6 では Postfix がデフォルトの MTA (メール転送エージェント) で推奨となります。別の MTA を使用している場合は、その設定ファイル固有の技術に応じて手作業で設定を行ってください。

9.2. Bash (Bourne-Again Shell)

Red Hat Enterprise Linux 6 には Bash のバージョン 4.1 がデフォルトシェルとして含まれています。本セクションでは旧バージョンに対してこのバージョンにより取り込まれた互換性に関する問題点を説明します。

- Bash-4.0 およびそれ以降のバージョンは、プロセス置換の構成をブレース展開を使って変更せずに渡すことができるようになるため、内容の展開はすべて別々に指定し、各プロセス置換を別々に入力しなければならなくなります。
- Bash-4.0 およびそれ以降のバージョンでは Posix が指定するのと同様に、SIGCHLD が wait の組込みに割り込みを許可するため、全ての子を待機するよう「wait」を使用する場合、子を終了したら SIGCHLD トラップは常に呼び出されることがなくなります。
- Bash-4.0 およびそれ以降のバージョンは、クローズ用の区切り文字 $() コマンド置換を検索する場合に Posix のルールに従うようになり、旧バージョンのような動作はしなくなりますが、より多くの構文および解析のエラーを先に捕らえてからコマンド置換を評価するためのサブシェルを生成します。
- プログラム可能な完了コードは、コマンドラインを単語に分解する際にシェルのメタキャラクタのセットではなく、読み込み行と同じ区切り文字のセットを使用します。そのため、プログラム可能な完了と読み込み行の一貫性が増すはずです。
- read のビルトインが時間切れになると、入力の read を指定変数に割り当てようとするため、十分な入力がない場合には変数が空の文字列にセットされる原因にもなります。旧バージョンは文字の read を破棄していました。
- Bash-4.0 およびそれ以降のバージョンでは、パイプラインのコマンドのひとつがコマンド一覧を実行している間に SIGINT によって終了させられた場合、シェルは割り込みを受けたかのような動作をします。
- Bash-4.0 およびそれ以降のバージョンでは、`set -e` オプションの処理法が変更されるため、パイプラインが失敗すると (失敗したパイプライン内の最後のコマンドが単純なコマンドでない場合も) シェルは終了します。これは Posix が指定するものとは異なります。この部分の基準を更新する作業が進展中です。Bash-4.0 の動作はリリースの時点での合意を得ようとしている動作です。
- Bash-4.0 およびそれ以降のバージョンでは、”.” がシステムの PATH に存在しない場合でも、`.` (`source`) ビルトインがファイル名の引数を現在のディレクトリで検索する原因になっていた Posix モードのバグを修正しています。Posix は、この場合シェルは PWD 変数内を調べるべきではないと述べています。
- Bash-4.1 は、[[コマンドに演算子を使用して文字列を比較する場合は現在のロケールを使用します。`compatNN` shopt オプションの 1 つをセットすることで以前の動作に戻すことができます。

正規表現

既に記載されている点に加えて、正規表現に対するパターン引数に引用符を付けて条件演算子 =~ に一致させると、正規表現の一致が動作しなくなる場合があります。これは全てのアーキテクチャで起こります。3.2 以前のバージョンとなる bash では、[[コマンドの =~ 演算子に対する正規表現の引数に引用符を付けることによる効果は規定されていませんでした。実用的な効果としては、パターン引数に二重引用符を付けると、バックスラッシュ「/」で特殊パターン文字に引用符を付けなければなりませんでした。これにより、二重引用符が付けられた単語展開で行われるバックスラッシュの処理を妨害し、== シェルパターン一致演算子が引用符が付けられた文字を処理する方法と矛盾していました。

bash バージョン 3.2 では、=~ 演算子に対して一重の引用符が付けられた文字列引数および二重の引用符が付けられた文字列引数内の文字には内部的に引用符が付けられるようシェルが変更されました。これにより、(「`.'」、「`['」、「`¥'」、「`(', `)」、「`*'」、「`+'」、「`?'」、「`{'」、「`|'」、「`^'」、「`$'」) を処理する正規表現に重要となる文字の特殊な意味を抑制し、そのまま強制的に一致が行われるようにします。== パターン一致演算子がそのパターン引数内の引用符が付けられた部分を処理する方法との矛盾がなくなります。

引用符付き文字列引数の処理が変更されてから問題がいくつか出現してきています。主要な問題として、パターン引数内の空白、bash 3.1 と bash 3.2 間で引用符付き文字列の処理が異なるなどの問題があげられます。いずれの問題もシェル変数を使ってパターンを維持することで解決できる可能性があります。[[コマンドの全オペランドでシェル変数を展開する場合には単語の分割は行われないため、変数を割り当てる時に好きなようにパターンに引用符を付けてから、空白を含む可能性のある単独の文字列にその値を展開することができるようになります。最初の問題は、バックスラッシュか他の引用メカニズムを使用してパターン内で空白をエスケープすることにより解決できる可能性があります。

Bash 4.0 は shopt ビルトインに対するいくつかのオプションで制御される 互換性レベル の概念を取り入れています。compat31 オプションを有効にすると、bash は =~ 演算子の右側の引用に関して 3.1 の動作を取り戻します。

9.3. その他のパッケージ変更点

更新されたパッケージ

以下の表では Red Hat Enterprise Linux 6 で更新が行われたパッケージと注意を必要とする変更点について示します。

表9.1 更新されたパッケージ

更新が行われたパッケージ	説明
OProfile	OProfile は 0.9.5 に更新されています。この新しいバージョンには、Intel Atom および i7 のプロセッサ、AMD Family 11h のプロセッサ、AMD Family 10h の IBS (Instruction Based Sampling) 機能が含まれています。
quota、edquota、setquota	ユーザー名またはユーザー ID を引数として受け取るようになります。引数が数字で表される場合はユーザー ID とみなされます。これ以外は ID に自動的に変換されます。ユーザー名が数字のみで構成されているような場合には問題となる可能性があるので注意してください。`quota` パッケージが更新されています。`quota`、`edquota`、`setquota` などのユーティリティでユーザー名を強制的に ID に変換していた `-x` 引数は削除されています。この機能は `--always-resolve` オプションで提供されるようになります。
module-init-tools	`/etc/modprobe.conf` はデフォルトでは存在しません。手作業で作成すればまだ使用可能です。

廃止されたパッケージ

以下の表では Red Hat Enterprise Linux 6 廃止 (削除) されたパッケージとその代替パッケージを示します。

表9.2 廃止されたパッケージ

廃止されたパッケージ	代替パッケージ
aspell	hunspell、aspell はビルド依存関係としてのみ提供されます。スペルチェックを使用するアプリケーションは hunspell を使用しなければなりません。
beecrypt	NSS/OpenSSL
crash-spu-commands	代替のパッケージなし。Cell 固有のパッケージは含まれなくなります。

廃止されたパッケージ	代替パッケージ
dhcpv6/dhcpv6-client	dhcp/dhclient のバイナリには IPv6 対応機能が組み込まれるようになります。
elfspe2	代替のパッケージなし。Cell 固有のパッケージは含まれなくなります。
exim	Postfix
gnbd	代用として iSCSI のご使用を推奨します。
gnome-vfs	gvfs
ipsec-tools	Openswan
kmod-gnbd	代用として iSCSI のご使用を推奨します。
lam	openmpi
libspe2	代替のパッケージなし。Cell 固有のパッケージは含まれなくなります。
libspe2-devel	代替のパッケージなし。Cell 固有のパッケージは含まれなくなります。
linuxwacom	xorg-x11-drv-wacom
mod_python	mod_wsgi、Python スクリプトの記述には代替として WSGI インターフェースを使用する mod_wsgi を使用することができます。
mkinitrd	dracut
nss_ldap	nss-pam-ldapd、pam_ldap
openmotif-2.2	openmotif-2.3
spu-tools	代替のパッケージなし。Cell 固有のパッケージは含まれなくなります。
switchdesk	セッションの管理は、サポートされている 2 種類のセッションマネージャ GDM と KDM で行われます。
syslog	rsyslog
SysVinit	upstart
vixie-cron	cronie

非推奨パッケージ

- qt3
- GFS1
- gcj、パフォーマンス確保のため Red Hat Enterprise Linux 6 には収納されますが、今後のリリースでは収納されない可能性があります。

9.4. ドライバー関連の変更点

本セクションでは、Red Hat Enterprise Linux 6 でのドライバー関連の変更点を説明します。ドライバーはすべてデフォルトで initramfs にロードされる点に注意してください。

廃止されたドライバー

- aic7xxx_old

- atp870u
- cpqarray
- DAC960
- dc395x
- gdth
- hfs
- hfsplus
- megaraid
- net/tokenring/
- paride
- qla1280
- sound/core/oss
- sound/drivers/opl3/*
- sound/pci/nm256

非推奨ドライバー

- aacraid
- aic7xxx
- i2o
- ips
- megaraid_mbox
- mptlan
- mptfc
- sym53c8xx

廃止されたカーネルコンポーネント

- NBD - Network Block Device は Red Hat Enterprise Linux 6 では iSCSI に入れ替わっています。
- HFS - Apple ファイルシステムのサポートは Red Hat Enterprise Linux 6 で廃止されました。
- Tux - Web Server アクセラレータは Red Hat Enterprise Linux 6 で廃止されました。
- 非 PAE の x86 カーネル – Red Hat Enterprise Linux の旧バージョンは i686 アーキテクチャ用に複数のカーネルを含んでいました (PAE 付きカーネルと PAE なしカーネル)。非 PAE ハードウェアが大量に販売されてから長い年月を経ています。したがって、Red Hat Enterprise Linux 6 には PAE を含んだカーネルがひとつのみのとなります。

- 先行の I/O スケジューラは非推奨となり Red Hat Enterprise Linux 6 からはなくなります。CFQ (Completely Fair Queueing) I/O スケジューラが代替となります。このスケジューラは 2006 年から Linux カーネル内のデフォルトの I/O スケジューラでした。先行の I/O スケジューラをご使用の方は CFQ で作業負荷のテストを実施してパフォーマンス関連の問題が見られる場合にはバグを提出されることをお勧めします。CFQ があらゆるテスト用作業負荷下で先行の I/O スケジューラと同等のパフォーマンスを行えることが目標となりますが、すべてのテストが期待通りでこの目標から外れるようなパフォーマンスはないとは保証できません。

9.5. ライブラリ関連の変更点

32-bit のライブラリはデフォルトでは Red Hat Enterprise Linux 6 にインストールされません。`/etc/yum.conf` で `multilib_policy=all` をセットするとこの動作を変更することができます。これにより、multilib ポリシーがシステム全域のポリシーとして有効になります。

付録A 改訂履歴

改訂 6.1-39　Wed May 18 2011　Radvan Scott [FAMILY Given]
sradvan@redhat.com

6.1 リリース用に点検