Red Hat Enterprise Linux 5

デプロイメントガイド

Red Hat Enterprise Linux 5 の導入、設定、および管理

Last Updated: 2023-03-01

Red Hat Enterprise Linux 5 デプロイメントガイド

Red Hat Enterprise Linux 5 の導入、設定、および管理

Enter your first name here. Enter your surname here.
Enter your organisation's name here. Enter your organisational division here.
Enter your email address here.

法律上の通知

概要

導入ガイドでは、Red Hat Enterprise Linux 5 の導入、設定、管理の関連情報について説明しています。

目次

はじめに

『Red Hat Enterprise Linux デプロイメントガイド』 へようこそ。

Red Hat Enterprise Linux デプロイメントガイド には、ニーズに合わせて Red Hat Enterprise Linux システムをカスタマイズする方法に関する情報が記載されています。システムの設定とカスタマイズに関する包括的なタスク指向ガイドが必要な場合は、このマニュアルを参照してください。

本書では、以下のような多くの中間トピックについて説明します。

- ネットワークインターフェイスカード(NIC)の設定
- 仮想プライベートネットワーク(VPN)の設定
- Samba 共有の設定
- RPM によるソフトウェアの管理
- システムに関する情報の決定
- カーネルのアップグレード

このマニュアルは、以下の主要カテゴリーに分類されます。

- ファイルシステム
- パッケージ管理
- ネットワーク関連の設定
- システムの設定
- システム監視
- カーネルおよびドライバーの設定
- セキュリティーおよび認証
- Red Hat のお客様および認定

本ガイドでは、Red Hat Enterprise Linux システムの基本を理解していることを前提としています。Red Hat Enterprise Linux のインストールサポートが必要な場合は、『Red Hat Enterprise Linux インストールガイド』 を参照してください。

1. 本書の表記慣例

本書では、特定の単語は異なるフォント、typefaces、size、および weights で表されます。この強調表示は体系的です。異なる単語は同じスタイルで表され、特定のカテゴリーに含まれることを示します。この方法を表す単語のタイプには、以下が含まれます。

command

Linux コマンド（およびその他のオペレーティングシステムコマンドを使用する場合）はこの方法になります。このスタイルは、コマンドラインで単語またはフレーズを **入力** し、Enter を押してコマンドを呼び出すことができることを示します。コマンドには、独自の（ファイル名など）の異なるスタイルで表示される単語が含まれる場合があります。このような場合は、コマンドの一部とみなされるため、フレーズ全体がコマンドとして表示されます。以下に例を示します。

cat testfile コマンドを使用して、現在の作業ディレクトリーにある **testfile** という名前のファイルの内容を表示します。

ファイル名

ファイル名、ディレクトリー名、パス、および RPM パッケージ名はこのように表されます。このスタイルは、特定のファイルまたはディレクトリーに、システム上にその名前が存在することを示します。例 :

ホームディレクトリーの **.bashrc** ファイルには、bash シェル定義と独自の使用のエイリアスが含まれます。

/etc/fstab ファイルには、さまざまなシステムデバイスとファイルシステムに関する情報が含まれます。

Web サーバーのログファイル分析プログラムを使用する場合は、**webalizer** RPM をインストールします。

application

このスタイルは、プログラムが（システムソフトウェアではなく）エンドユーザーアプリケーションであることを示します。以下に例を示します。

Mozilla を使用して Web を参照します。

key

このスタイルにはキーボードのキーが表示されます。以下に例を示します。

Tab 補完を使用してディレクトリー内の特定のファイルを一覧表示するには、**ls** と入力してから 文字を入力し、最後に **Tab** キーを押します。ターミナルは、その文字で始まる作業ディレクトリー内のファイルの一覧を表示します。

キー+組み合わせ

このようにキーストロークの組み合わせを表現します。以下に例を示します。

Ctrl+**Alt**+**Backspace** キーの組み合わせはグラフィカルセッションを終了し、グラフィカルログイン画面またはコンソールに戻ります。

GUI インターフェイスにあるテキスト

GUI インターフェイス画面またはウィンドウのタイトル、単語、またはフレーズがこのスタイルに表示されます。このスタイルに表示されるテキストは、特定の GUI 画面または GUI 画面の要素（チェックボックスまたはフィールドに関連付けられたテキストなど）を示します。以下に例を示します。

スクリーンセーバーにパスワードを必要とする場合は、**Require Password** のチェックボックスを選択します。

GUI 画面またはウィンドウのメニューのトップレベル

このスタイルの単語は、単語がプルダウンメニューのトップレベルであることを示します。GUI 画面で単語をクリックすると、メニューの残りの部分が表示されます。以下に例を示します。

GNOME 端末の **File** で、**New Tab** オプションでは、同じウィンドウで複数のシェルプロンプトを開くことができます。

GUI メニューから一連のコマンドを入力する手順は、以下の例のようになります。

Applications （パネルのメインメニュー）> Programming > Emacs Text Editor に移動して、Emacs テキストエディターを起動します。

button on a GUI screen or window

このスタイルは、GUI 画面のクリック可能なボタンでテキストが見つかることを示しています。以下に例を示します。

Back ボタンをクリックして、最後に表示した Web ページに戻ります。

コンピューターの出力

このスタイルのテキストは、エラーメッセージやコマンドへの応答などのシェルプロンプトに表示されるテキストを示します。以下に例を示します。

ls コマンドはディレクトリーの内容を表示します。以下に例を示します。

```
Desktop   about.html   logs   paulwesterberg.png
Mail   backupfiles   mail   reports
```

このスタイルでは、コマンド（この場合は ディレクトリーの内容）に応じて返される出力が表示されます。

prompt

このスタイルには、コンピューターに入力できることを示すプロンプトが表示されます。例 :

$

#

[stephen@maturin stephen]$

leopard login:

user input

このスタイルには、コマンドラインまたは GUI 画面のテキストボックスにユーザーが入力するテキストが表示されます。以下の例では、**text** がこのスタイルに表示されます。

テキストベースのインストールプログラムでシステムを起動するには、**boot:** プロンプトで **text** コマンドを入力する必要があります。

<replaceable>

このスタイルには、ユーザーが提供するデータに置き換えることが意図されている例で使用されるテキストが表示されます。以下の例では、&lt ;*version-number*> がこのスタイルに表示されます。

カーネルソースのディレクトリーは /**usr**/**src**/**kernels**/ ***<version-number>*** / です。< *version-number* > は、このシステムにインストールされているカーネルのバージョンとタイプです。

さらに、いくつかの異なるストラテジーを使用して、特定の情報に注意を促します。緊急の順序では、これらの項目には注記、ヒント、重要、注意、または警告のマークが付けられます。以下に例を示します。

注記

Linux では大文字と小文字が区別されることに注意してください。つまり、rose は ROSE ではなく rOsE ではありません。

ヒント

/usr/share/doc/ ディレクトリーには、システムにインストールされているパッケージの追加ドキュメントが含まれています。

重要な影響

DHCP 設定ファイルを変更しても、DHCP デーモンを再起動するまで変更は反映されません。

注意

root としてルーチンタスクを実行しないでください。システム管理タスクに root アカウントを使用する必要がない限り、通常のユーザーアカウントを使用します。

WARNING

必要なパーティションのみを削除するように注意してください。他のパーティションを削除すると、データ損失やシステム環境が破損する可能性があります。

2. フィードバックをお寄せください

『Red Hat Enterprise Linux デプロイメントガイド』 でエラーが見つかった場合や、このマニュアルを改善する方法がない場合は、ご連絡ください。コンポーネントの **Deployment_Guide** に対してBugzilla（）でレポートを送信します。

ドキュメントの改善に関するご意見がある場合は、できるだけ具体的にお試しください。エラーが見つかった場合は、簡単に確認できるように、セクション番号と周りのテキストを含めます。

パート I. ファイルシステム

ファイルシステム は、コンピューターに保存されているファイルおよびディレクトリーを参照します。ファイルシステムは、*ファイルシステムタイプ* と呼ばれる異なる形式を持つことができます。これらの形式により、情報がファイルおよびディレクトリーとして保存される方法が決まります。ファイルシステムの種類によっては、データの冗長コピーを保存するものもありますが、ファイルシステムの種類によっては、ハードドライブへのアクセスが速くなります。ここでは、ext3、swap、RAID、および LVM のファイルシステムタイプを説明します。また、ファイルパーミッションをカスタマイズするためのパーティションおよびアクセス制御リスト(ACL)を管理する **parted** ユーティリティーについても説明します。

第1章 ファイルシステム構造

1.1. 共通の構造を共有する理由

ファイルシステムの構造は、オペレーティングシステムの組織の最も基本的なレベルです。オペレーティングシステムがユーザー、アプリケーション、およびセキュリティーモデルと対話する方法は、ストレージデバイスでファイルを整理する方法によって異なります。一般的なファイルシステム構造を提供すると、ユーザーとプログラムがファイルにアクセスして書き込むことができます。

ファイルシステムは、ファイルを 2 つの論理カテゴリーに分類します。

- 共有可能ファイルと非共有ファイル
- 変数ファイルと静的ファイル

共有可能 ファイルは、ローカルおよびリモートホストからアクセスできるファイルです。共有不 *可能* なファイルは、ローカルでのみ利用できます。ドキュメントなどの *変数* ファイルはいつでも変更できます。バイナリーなどの *静的* ファイルは、システム管理者のアクションなしでは変更されません。

この方法でファイルを検索する理由は、ファイルの機能と、それらを保持するディレクトリーに割り当てられたパーミッションを関連付けるのに役立つためです。オペレーティングシステムとそのユーザーが特定のファイルと対話する方法により、そのディレクトリーが読み取り専用パーミッションまたは読み取り/書き込みパーミッションでマウントされているかどうか、および各ユーザーがそのファイルにアクセスできるレベルが決まります。この組織のトップレベルは重要です。基礎となるディレクトリーへのアクセスは制限されるか、またはセキュリティーの問題がトップレベルからもマニフェストされてしまう可能性があります。これは、厳密な構造に準拠しません。

1.2. ファイルシステム階層標準(FHS)の概要

Red Hat Enterprise Linux は、*Filesystem Hierarchy Standard* (*FHS*) のファイルシステム構造を使用します。これは、多くのファイルタイプやディレクトリーの名前、場所、およびパーミッションを定義します。

FHS ドキュメントは、すべての FHS 準拠のファイルシステムにとって信頼できるリファレンスですが、この標準では、多くの領域が未定義または拡張可能です。このセクションでは、標準の概要と、標準でカバーされていないファイルシステムの部分について説明します。

標準への準拠は多くのことを意味しますが、最も重要な点は、他の準拠システムと互換性であり、**/usr/** パーティションを読み取り専用としてマウントする機能です。ディレクトリーには共通の実行ファイルが含まれており、ユーザーが変更すべきではないため、この 2 番目の点は重要です。また、**/usr/** ディレクトリーは読み取り専用としてマウントされているため、CD-ROM または読み取り専用 NFS マウントを介して別のマシンからマウントできます。

1.2.1. FHS 組織

ここで記載されているディレクトリーおよびファイルは、FHS ドキュメントで指定された小さなサブセットです。最も詳しい情報については、最新の FHS ドキュメントを参照してください。

完全な標準は http://www.pathname.com/fhs/ でオンラインで利用できます。

1.2.1.1. /boot/ ディレクトリー

/boot/ ディレクトリーには、Linux カーネルなどのシステムの起動に必要な静的ファイルが含まれます。これらのファイルは、システムが正しく起動するためには不可欠です。

警告

/boot/ ディレクトリーを削除しないでください。削除すると、システムが起動できなくなります。

1.2.1.2. /dev/ ディレクトリー

/dev/ ディレクトリーには、システムに接続されているデバイス、またはカーネルが提供する仮想デバイスを表すデバイスノードが含まれます。これらのデバイスノードは、システムが適切に機能するために不可欠です。**udev** デーモンは、**/dev/** にあるこれらのデバイスノードをすべて作成し、削除します。

/dev ディレクトリーおよびサブディレクトリー内のデバイスは、文字（入出力のシリアルストリームのみを提供）または block（ランダムにアクセス可能）のいずれかです。キャラクターデバイスにはマウス、キーボード、モデムが含まれ、ブロックデバイスにはハードディスク、フロッピードライブなどが含まれます。システムに GNOME または KDE がインストールされている場合は、外部ドライブや cd などのデバイスが接続時（例：usb など）または挿入(CD または DVD ドライブなど)時に自動的に検出され、コンテンツを表示するポップアップウィンドウが自動的に表示されます。**/dev** ディレクトリー内のファイルは、システムが適切に機能するために不可欠です。

表1.1 /dev内の一般的なファイルの例

File	説明
/dev/hda	プライマリー IDE チャネル上のマスターデバイス。
/dev/hdb	プライマリー IDE チャネルのスレーブデバイス。
/dev/tty0	最初の仮想コンソール。
/dev/tty1	2 番目の仮想コンソール
/dev/sda	プライマリー SCSI または SATA チャネルの最初のデバイス。
/dev/lp0	最初の並列ポート。

1.2.1.3. /etc/ ディレクトリー

/etc/ ディレクトリーは、マシンのローカルとなる設定ファイル用に予約されています。**/etc/** に配置されるバイナリーはありません。**/etc/** に置かれたバイナリーは、**/sbin/** または **/bin/** に配置する必要があります。

/etc 内のディレクトリーの例は、**X11/** および **skel/** です。

```
/etc
  |- X11/
  |- skel/
```

/etc/X11/ ディレクトリーは、**xorg.conf** などの X Window System 設定ファイル用です。 **/etc/skel/** ディレクトリーは、ユーザーの初回作成時にホームディレクトリーを設定するために使用されるスケルトンユーザーファイル用です。また、アプリケーションはこのディレクトリーに設定ファイルを保存し、実行時にそれらを参照する可能性があります。

1.2.1.4. /lib/ ディレクトリー

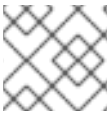
/lib/ ディレクトリーには、**/bin/** および **/sbin/** でバイナリーを実行するために必要なライブラリーのみが含まれている必要があります。これらの共有ライブラリーイメージは、システムを起動し、root ファイルシステム内でコマンドを実行する場合に特に重要です。

1.2.1.5. /media/ ディレクトリー

/media/ ディレクトリーには、usb ストレージメディア、DVD、CD-ROM、Zip ディスクなどのリムーバブルメディアのマウントポイントとして使用されるサブディレクトリーが含まれます。

1.2.1.6. /mnt/ ディレクトリー

/mnt/ ディレクトリーは、NFS ファイルシステムのマウントなどの、一時的にマウントされたファイルシステム用に予約されています。すべてのリムーバブルメディアには、**/media/** ディレクトリーを使用してください。自動的に検出されたリムーバブルメディアは、**/media** ディレクトリーにマウントされます。



注記

/mnt ディレクトリーは、インストールプログラムでは使用しないでください。

1.2.1.7. /opt/ ディレクトリー

/opt/ ディレクトリーは、ほとんどのアプリケーションソフトウェアパッケージのストレージを提供します。

/opt/ ディレクトリーにファイルを配置するパッケージにより、パッケージと同じ名前を持つディレクトリーが作成されます。次に、このディレクトリーはファイルシステム全体に分散されるファイルを保持し、システム管理者は特定のパッケージ内の各ファイルのロールを簡単に判別できるようにします。

たとえば、**sample** が **/opt/** ディレクトリーにある特定のソフトウェアパッケージの名前である場合、そのファイルはすべて **/opt/sample/** bin/（バイナリーの場合は **/opt/sample/bin/**、**man ページの場合は /opt/sample/bin/** など）に配置されます。

多くの異なるサブパッケージ、データファイル、追加フォント、clipart などを含むパッケージも **/opt/** ディレクトリーに置かれ、その大きなパッケージ自体を整理する方法を提供します。このようにして、サンプルパッケージには、それぞれが /opt/ **sample** /tool **1/ や /opt/sample/tool 2/ などの独自のサブディレクトリーに配置されるツール** があり、各パッケージは独自の **bin/**、**man/**、およびその他の同様のディレクトリーを持つことができます。

1.2.1.8. /proc/ ディレクトリー

/proc/ ディレクトリーには、カーネルから情報を抽出するか、カーネルへの情報送信を行う特別なファイルが含まれます。たとえば、システムメモリー、CPU 情報、ハードウェア設定などが挙げられます。

/proc/ で利用できる大量のデータや、このディレクトリーを使用してカーネルと通信する方法が多数あるため、章全体がサブジェクトに展開されています。詳細は、5章*proc ファイルシステム* を参照してください。

1.2.1.9. /sbin/ ディレクトリー

/sbin/ ディレクトリーには、root ユーザーが使用する実行ファイルが保存されます。**/sbin/** の実行可能ファイルは、システム管理とシステム復旧操作を実行するために、システムの起動時に使用されます。このディレクトリーでは、FHS は以下を示しています。

> **/sbin** には、**/bin** のバイナリーに加えて、システムの起動、復元、リカバリー、および修復に不可欠なバイナリーが含まれています。**/usr/** の後に実行されるプログラムは（問題がない場合に）マウントすることがわかっています。通常、**/usr/sbin** に配置されます。ローカルにインストールされたシステム管理プログラムは、**/usr/local/sbin** に配置する必要があります。

少なくとも、以下のプログラムは **/sbin/** になければなりません。

```
arp, clock,
halt, init,
fsck.*, grub,
ifconfig, mingetty,
mkfs.*, mkswap,
reboot, route,
shutdown, swapoff,
swapon
```

1.2.1.10. /srv/ ディレクトリー

/srv/ ディレクトリーには、Red Hat Enterprise Linux を実行しているシステムが提供するサイト固有のデータが含まれます。このディレクトリーは、FTP、WWW、または CVS などの特定サービスのデータファイルの場所をユーザーに提供します。特定のユーザーにのみ関連するデータは、**/home/** ディレクトリー内になければなりません。

1.2.1.11. /sys/ ディレクトリー

/sys/ ディレクトリーは、2.6 カーネルに固有の新しい **sysfs** 仮想ファイルシステムを使用します。2.6 カーネルのホットプラグハードウェアデバイスのサポートが増えると、**/sys/** ディレクトリーには同様の情報が **/proc/** に保持されますが、ホットプラグデバイスに関する特定のデバイス情報の階層ビューが表示されます。

1.2.1.12. /usr/ ディレクトリー

/usr/ ディレクトリーは、複数のマシンにまたがって共有できるファイル用です。多くの場合、**/usr/** ディレクトリーは独自のパーティションにあり、読み取り専用でマウントされます。少なくとも、以下のディレクトリーは **/usr/** のサブディレクトリーである必要があります。

```
/usr
   |- bin/
   |- etc/
   |- games/
   |- include/
   |- kerberos/
   |- lib/
```

```
|- libexec/
|- local/
|- sbin/
|- share/
|- src/
|- tmp -> ../var/tmp/
```

/usr/ ディレクトリーの **bin/** サブディレクトリーには実行ファイルが含まれ、**etc/** にはシステム全体の設定ファイル、**games** is for games、**include/** には C ヘッダーファイルが含まれ、**kerberos/** にはバイナリーやその他の Kerberos 関連のファイルが含まれます。**lib/** には、ユーザーまたはシェルスクリプトが直接使用されないように設計されていないオブジェクトファイルとライブラリーが含まれます。**libexec/** ディレクトリーには、他のプログラムによって呼び出される小さなヘルパープログラムが含まれています。**sbin/** はシステム管理バイナリー(**/sbin/** ディレクトリーに属さないもの)用です。**share/** には、アーキテクチャー固有ではないファイル、**src/** はソースコード用です。

1.2.1.13. /usr/local/ ディレクトリー

FHS は以下を示しています。

> **/usr/local** 階層は、ソフトウェアをローカルでインストールする場合にシステム管理者が使用します。システムソフトウェアの更新時に上書きされないようにする必要があります。これは、ホストのグループ間で共有できるが、**/usr** にないプログラムやデータに使用できます。

/usr/ local/ ディレクトリーは、/usr/ ディレクトリーの構造に似ています。これには、**/usr/** ディレクトリー内のサブディレクトリーと似た以下のサブディレクトリーがあります。

```
/usr/local
 |- bin/
 |- etc/
 |- games/
 |- include/
 |- lib/
 |- libexec/
 |- sbin/
 |- share/
 |- src/
```

Red Hat Enterprise Linux では、**/usr/local/** ディレクトリーに使用することが、FHS で指定されたものと若干異なります。FHS は、**/usr/local/** が、システムソフトウェアのアップグレードで安全を確保できるソフトウェアが保存されている場所である必要があることを示しています。ソフトウェアアップグレードは *RPM Package Manager* (*RPM*)を使用して安全に実行できるため、ファイルを **/usr/local/** に配置してファイルを保護する必要はありません。代わりに、**/usr/local/** ディレクトリーは、マシンのローカルにあるソフトウェアに使用されます。

たとえば、**/usr/** ディレクトリーがリモートホストから読み取り専用の NFS 共有としてマウントされている場合でも、**/usr/local/** ディレクトリーの下にパッケージまたはプログラムをインストールすることができます。

1.2.1.14. /var/ ディレクトリー

FHS では、Linux が **/usr/** を読み取り専用としてマウントする必要があるため、ログファイルを書き込むプログラムや、**spool/** または **lock/** ディレクトリーが必要なプログラムは、それらを **/var/** ディレクトリーに書き込む必要があります。FHS の状態 **/var/** は以下を対象にしています。

> ...変数データファイル。これには、スプールディレクトリーおよびファイル、管理およ

びロギングデータ、および一時および一時ファイルが含まれます。

以下は、**/var/** ディレクトリーにあるディレクトリーの一部です。

```
/var
   |- account/
   |- arpwatch/
   |- cache/
   |- crash/
   |- db/
   |- empty/
   |- ftp/
   |- gdm/
   |- kerberos/
   |- lib/
   |- local/
   |- lock/
   |- log/
   |- mail -> spool/mail/
   |- mailman/
   |- named/
   |- nis/
   |- opt/
   |- preserve/
   |- run/
   +- spool/
       |- at/
       |- clientmqueue/
       |- cron/
       |- cups/
       |- exim/
       |- lpd/
       |- mail/
       |- mailman/
       |- mqueue/
       |- news/
       |- postfix/
       |- repackage/
       |- rwho/
       |- samba/
       |- squid/
       |- squirrelmail/
       |- up2date/
       |- uucp
       |- uucppublic/
       |- vbox/
|- tmp/
|- tux/
|- www/
|- yp/
```

messages および **lastlog** などのシステムログファイルは、**/var/log/** ディレクトリーに移動します。**/var/lib/rpm/** ディレクトリーには、RPM システムデータベースが含まれます。ロックファイルは、通常はファイルを使用するプログラムのディレクトリーにある **/var/lock/** ディレクトリーに移動します。**/var/spool/** ディレクトリーには、データファイルが保存されるプログラムのサブディレクトリーがあります。

1.3. RED HAT ENTERPRISE LINUX の特別なファイルの場所

Red Hat Enterprise Linux は、特別なファイルに対応するために FHS 構造を若干拡張しています。

RPM に関連するほとんどのファイルは、**/var/lib/rpm/** ディレクトリーに保持されます。RPM の詳細は、12章*RPM でのパッケージ管理* を参照してください。

/var/cache/yum/ ディレクトリーには、システムの RPM ヘッダー情報を含む **Package Updater** が使用するファイルが含まれます。この場所は、システムの更新中にダウンロードされた RPM を一時的に保存するためにも使用できます。**Red Hat Network** の詳細は、15章 *システムの登録およびサブスクリプション管理* を参照してください。

Red Hat Enterprise Linux に固有の別の場所は **/etc/sysconfig/** ディレクトリーです。このディレクトリーには、さまざまな設定情報が格納されています。システムの起動時に実行されるスクリプトの多くは、このディレクトリー内のファイルを使用します。このディレクトリー内の内容や、起動プロセスでこれらのファイルがプレイするロールの詳細は、32章***sysconfig* ディレクトリー** を参照してください。

第2章 MOUNT コマンドの使い方

Linux、UNIX、および同様のオペレーティングシステムでは、CD、DVD、USB フラッシュドライブなどの異なるパーティションやリムーバブルデバイスのファイルシステムは、ディレクトリーツリーの特定ポイント（つまり *マウントポイント*）に接続して、再度切り離すことができます。ファイルシステムの接続または割り当て解除は、それぞれ **mount** コマンドまたは **umount** コマンドを使用できます。本章では、これらのコマンドの基本的な使用方法について説明し、マウントポイントの移動や共有サブツリーの作成などの高度なトピックについて説明します。

2.1. 現在マウントされているファイルシステムの一覧表示

現在接続している全ファイルシステムを表示させるには、**mount** コマンドを実行します。いずれの引数も付けません。

```
mount
```

上記のコマンドで既知のマウントポイントの一覧が表示されます。行ごとにデバイス名、ファイルシステムのタイプ、マウントしているディレクトリー、およびマウントオプションなどの情報が以下のような形で表示されます。

device on *directory* type *type* (*options*)

デフォルトでは、出力には **sysfs** や**tmpfs** などのさまざまな仮想ファイルシステムが含まれます。特定のファイルシステムタイプのデバイスのみを表示するには、コマンドラインで **-t** オプションを指定します。

```
mount -t type
```

一般的なファイルシステムタイプのリストについては、表2.1「一般的なファイルシステムのタイプ」を参照してください。mount コマンドを使用して、**マウント** されたファイルシステムを一覧表示する方法は、例2.1「現在マウントされている **ext3** ファイルシステムの一覧表示」 を参照してください。

例2.1 現在マウントされている ext3 ファイルシステムの一覧表示

通常、/ パーティションと /**boot** パーティションはいずれも **ext3** を使用するようにフォーマットされます。このファイルシステムを使用するマウントポイントのみを表示するには、シェルプロンプトで次のように入力します。

```
~]$ mount -t ext3
/dev/mapper/VolGroup00-LogVol00 on / type ext3 (rw)
/dev/vda1 on /boot type ext3 (rw)
```

2.2. ファイルシステムのマウント

特定のファイルシステムを接続するには、以下のような形式で **mount** コマンドを使用します。

```
mount [option...] device directory
```

mount コマンドを実行すると、**/etc/fstab** 設定ファイルの内容を読み取り、指定したファイルシステムが一覧表示されているかどうかを確認します。このファイルには、デバイス名の一覧と、選択したファイルシステムがマウントされるディレクトリー、ファイルシステムタイプおよびマウントオプションが

含まれます。このため、このファイルで指定されているファイルシステムをマウントする場合は、以下のいずれかのコマンドのバリアントを使用できます。

```
mount [option...] directory
mount [option...] device
```

root としてログインしていない限り、ファイルシステムをマウントするパーミッションが必要です(「マウントオプションの指定」を参照してください)。

2.2.1. ファイルシステムタイプの指定

ほとんどの場合は、**mount** によって自動的にファイルシステムが検出されます。ただし、**NFS** (Network File System) や **CIFS** (Common Internet File System) などの認識できないファイルシステムがあるため、こうしたファイルシステムの場合は手動で指定しなければなりません。ファイルシステムのタイプを指定するには、以下の形式で **mount** コマンドを使用します。

```
mount -t type device directory
```

表2.1「一般的なファイルシステムのタイプ」 は、**mount** コマンドで使用できる一般的なファイルシステムのタイプの一覧を提供します。利用可能なすべてのファイルシステムタイプの完全なリストについては、「インストールされているドキュメント」 に記載の関連する man ページを参照してください。

表2.1 一般的なファイルシステムのタイプ

型	説明
ext2	**ext2** ファイルシステム。
ext3	**ext3** ファイルシステム。
ext4	**ext4** ファイルシステム。
iso9660	**ISO 9660** ファイルシステム。通常、これは光学メディア (通常は CD) で使用されます。
jfs	IBM が作成した **JFS** ファイルシステム。
nfs	**NFS** ファイルシステム。通常、これはネットワーク経由でファイルにアクセスするために使用されます。
nfs4	**NFSv4** ファイルシステム。通常、これはネットワーク経由でファイルにアクセスするために使用されます。
ntfs	**NTFS** ファイルシステム。これは通常、Windows オペレーティングシステムを実行しているマシンで使用されます。
udf	**UDF** ファイルシステム。通常、これは光学メディア (通常は DVD) で使用されます。
vfat	**FAT** ファイルシステム。通常、これは Windows オペレーティングシステムを実行しているマシンや、USB フラッシュドライブやフロッピーディスクなどの特定のデジタルメディアで使用されます。

使用例は、例2.2「USB フラッシドライブのマウント」 を参照してください。

例2.2 USB フラッシドライブのマウント

多くの場合、古い USB フラッシュドライブは FAT ファイルシステムを使用します。このようなドライブが **/dev/sdc1** デバイスを使用し、**/media/flashdisk/** ディレクトリーが存在すると仮定すると、**root** で次のコマンドを実行します。

```
~]# mount -t vfat /dev/sdc1 /media/flashdisk
```

2.2.2. マウントオプションの指定

追加のマウントオプションを指定するには、以下の形式でコマンドを実行します。

```
mount -o options
```

複数のオプションを指定する場合は、コンマの後にスペースを挿入しないでください。挿入すると、**mount** はスペースに続く値を追加のパラメーターとして誤って解釈します。

表2.2「一般的なマウントオプション」 一般的なマウントオプションの一覧を提供します。利用可能なオプションの一覧は、「インストールされているドキュメント」 に記載の関連する man ページを参照してください。

表2.2 一般的なマウントオプション

オプション	説明
async	ファイルシステム上での非同期の入/出力を許可します。
auto	**mount -a** コマンドを使用したファイルシステムの自動マウントを許可します。
defaults	**async,auto,dev,exec,nouser,rw,suid** のエイリアスを指定します。
exec	特定のファイルシステムでのバイナリーファイルの実行を許可します。
loop	イメージをループデバイスとしてマウントします。
noauto	**mount -a** コマンドを使用したファイルシステムの自動マウントを無効にします。
noexec	特定のファイルシステムでのバイナリーファイルの実行は許可しません。
nouser	普通のユーザー (つまり **root** 以外のユーザー) によるファイルシステムのマウントおよびアンマウントは許可しません。
remount	ファイルシステムがすでにマウントされている場合は再度マウントを行います。
ro	読み取り専用でファイルシステムをマウントします。
rw	ファイルシステムを読み取りと書き込み両方でマウントします。

オプション	説明
user	普通のユーザー (つまり **root** 以外のユーザー) によるファイルシステムのマウントおよびアンマウントを許可します。

使用例は、例2.3「ISO イメージのマウント」 を参照してください。

例2.3 ISO イメージのマウント

ISO イメージ (または一般的にはディスクイメージ) はループデバイスを使用することでマウントすることができます。Fedora 14 インストールディスクの ISO イメージが現在の作業ディレクトリーに存在し、**/media/cdrom/** ディレクトリーが存在すると仮定すると、**root** で以下のコマンドを実行してイメージをこのディレクトリーにマウントできます。

```
~]# mount -o ro,loop Fedora-14-x86_64-Live-Desktop.iso /media/cdrom
```

ISO9660 は設計上、読み取り専用のファイルシステムになっていることに注意してください。

2.2.3. マウントの共有

システム管理作業の中には、同じファイルシステムにディレクトリーツリー内の複数の場所からのアクセスしないといけない場合があります (chroot 環境を準備する場合など)。このような要件に対処するために、**mount** コマンドは、特定のマウントを複製する手段を提供する **--bind** オプションを実装します。以下のような使用法になります。

```
mount --bind old_directory new_directory
```

上記のコマンドは、両方の場所からファイルシステムにアクセスできますが、元のディレクトリー内にマウントされているファイルシステムには適用されません。これらのマウントも含めるには、次のように入力します。

```
mount --rbind old_directory new_directory
```

さらに、Red Hat Enterprise Linux 5.10 は可能な限り柔軟性を提供するために、*共有サブツリー と呼ばれる機能を実装します*。この機能により、以下の 4 つのマウントタイプを使用できます。

共有マウント

共有マウントを使用すると、特定のマウントポイントの正確なレプリカを作成できます。共有マウントが作成されると、元のマウントポイント内のすべてのマウントがそれに反映され、その逆も同様です。共有マウントを作成するには、シェルプロンプトで以下を入力します。

```
mount --make-shared mount_point
```

または、選択したマウントポイントとその下のすべてのマウントポイントのマウントタイプを変更できます。

```
mount --make-rshared mount_point
```

使用例は、例2.4「共有マウントポイントの作成」 を参照してください。

例2.4 共有マウントポイントの作成

他のファイルシステムが一般的にマウントされる場所は 2 つあります。リムーバブルメディア用の **/media** ディレクトリーと、一時的にマウントされるファイルシステム用の **/mnt** ディレクトリーです。共有マウントを使用すると、この 2 つのディレクトリーで同じコンテンツを共有できます。そのためには、**root** で、**/media** ディレクトリーを 「shared」 としてマークします。

```
~]# mount --bind /media /media
~]# mount --make-shared /media
```

以下のコマンドを使用して、複製を **/mnt** ディレクトリーに作成します。

```
~]# mount --bind /media /mnt
```

これで、**/media** 内のマウントが **/mnt** にも表示されることを確認できます。たとえば、CD-ROM ドライブに空でないメディアがあり、**/media/cdrom/** ディレクトリーが存在する場合は、以下のコマンドを実行します。

```
~]# mount /dev/cdrom /media/cdrom
~]# ls /media/cdrom
EFI  GPL  isolinux  LiveOS
~]# ls /mnt/cdrom
EFI  GPL  isolinux  LiveOS
```

同様に、**/mnt** ディレクトリーにマウントされているファイルシステムが **/media** に反映されていることを確認できます。たとえば、**/dev/sdc1** デバイスを使用する空でない USB フラッシュドライブがあり、**/mnt/flashdisk/** ディレクトリーが存在する場合は、以下を入力します。

```
~]# mount /dev/sdc1 /mnt/flashdisk
~]# ls /media/flashdisk
en-US  publican.cfg
~]# ls /mnt/flashdisk
en-US  publican.cfg
```

スレーブマウント

スレーブマウントを使用すると、指定したマウントポイントの限定的な複製を作成できます。スレーブマウントが作成されると、元のマウントポイント内のすべてのマウントがそれに反映されますが、スレーブマウント内のマウントは元のマウントに反映されません。スレーブマウントを作成するには、シェルプロンプトで以下を入力します。

```
mount --make-slave mount_point
```

または、選択したマウントポイントとその下のすべてのマウントポイントのマウントタイプを変更できます。

```
mount --make-rslave mount_point
```

使用例は、例2.5「スレーブマウントポイントの作成」 を参照してください。

例2.5 スレーブマウントポイントの作成

/media ディレクトリーのコンテンツが **/mnt** にも表示されるようにし、**/mnt** ディレクトリーのマウントを **/media** に反映させないようにするとします。これを実行するには、**root** で、最初に **/media** ディレクトリーを 「共有」 としてマークします。

```
~]# mount --bind /media /media
~]# mount --make-shared /media
```

次に、その複製を **/mnt** で作成します。ただし、「slave」 としてマークします。

```
~]# mount --bind /media /mnt
~]# mount --make-slave /mnt
```

これで、**/media** 内のマウントが **/mnt** にも表示されることを確認できます。たとえば、CD-ROM ドライブに空でないメディアがあり、**/media/cdrom/** ディレクトリーが存在する場合は、以下のコマンドを実行します。

```
~]# mount /dev/cdrom /media/cdrom
~]# ls /media/cdrom
EFI  GPL  isolinux  LiveOS
~]# ls /mnt/cdrom
EFI  GPL  isolinux  LiveOS
```

また、**/mnt** ディレクトリーにマウントされているファイルシステムが **/media** に反映されて **いない** ことを確認することもできます。たとえば、**/dev/sdc1** デバイスを使用する空でない USB フラッシュドライブがあり、**/mnt/flashdisk/** ディレクトリーが存在する場合は、以下を入力します。

```
~]# mount /dev/sdc1 /mnt/flashdisk
~]# ls /media/flashdisk
~]# ls /mnt/flashdisk
en-US  publican.cfg
```

プライベートマウント

プライベートマウントを使用すると、通常のマウントを作成できます。プライベートマウントが作成されると、元のマウントポイント内の後続のマウントがそれに反映されず、プライベートマウント内のマウントは元のマウントに反映されません。プライベートマウントを作成するには、シェルプロンプトで以下を入力します。

```
mount --make-private mount_point
```

または、選択したマウントポイントとその下のすべてのマウントポイントのマウントタイプを変更できます。

```
mount --make-rprivate mount_point
```

使用例は、例2.6「プライベートマウントポイントの作成」 を参照してください。

例2.6 プライベートマウントポイントの作成

例2.4「共有マウントポイントの作成」 でシナリオを考慮に入れて、**root** で以下のコマンドを使用して共有マウントポイントを作成していることを前提としています。

```
~]# mount --bind /media /media
~]# mount --make-shared /media
~]# mount --bind /media /mnt
```

/mnt ディレクトリーに 「private」 のマークを付けるには、次のように入力します。

```
~]# mount --make-private /mnt
```

これで、**/media** 内のマウントが **/mnt** に表示されないことを確認できます。たとえば、CD-ROM ドライブに空でないメディアがあり、**/media/cdrom/** ディレクトリーが存在する場合は、以下のコマンドを実行します。

```
~]# mount /dev/cdrom /media/cdrom
~]# ls /media/cdrom
EFI  GPL  isolinux  LiveOS
~]# ls /mnt/cdrom
~]#
```

また、**/mnt** ディレクトリーにマウントされているファイルシステムが **/media** に反映されていないことを確認することもできます。たとえば、**/dev/sdc1** デバイスを使用する空でない USB フラッシュドライブがあり、**/mnt/flashdisk/** ディレクトリーが存在する場合は、以下を入力します。

```
~]# mount /dev/sdc1 /mnt/flashdisk
~]# ls /media/flashdisk
~]# ls /mnt/flashdisk
en-US  publican.cfg
```

バインド不可能なマウント

バインド不可能なマウントを使用すると、特定のマウントポイントが重複しないようにすることができます。バインド不可能なマウントを作成するには、シェルプロンプトで以下を入力します。

```
mount --make-unbindable mount_point
```

または、選択したマウントポイントとその下のすべてのマウントポイントのマウントタイプを変更できます。

```
mount --make-runbindable mount_point
```

使用例は、例2.7「バインド不可能なマウントポイントの作成」 を参照してください。

例2.7 バインド不可能なマウントポイントの作成

/media ディレクトリーが共有されないようにするには、**root** として、シェルプロンプトで次のように入力します。

```
~]# mount --bind /media /media
~]# mount --make-unbindable /media
```

こうすることで、これ以降、このマウントの複製を作成しようとすると、以下のエラーが出て失敗します。

```
~]# mount --bind /media /mnt
mount: wrong fs type, bad option, bad superblock on /media/,
       missing code page or other error
       In some cases useful info is found in syslog - try
       dmesg | tail  or so
```

2.2.4. マウントポイントの移動

ファイルシステムがマウントされているディレクトリーを変更するには、次のコマンドを使用します。

```
mount --move old_directory new_directory
```

使用例は、例2.8「既存の NFS マウントポイントの移動」 を参照してください。

例2.8 既存の NFS マウントポイントの移動

ユーザーディレクトリーを含む NFS ストレージがあるとします。このストレージがすでに **/mnt/userdirs/** にマウントされている場合、**root** として次のコマンドを使用して、このマウントポイントを **/home** に移動できます。

```
~]# mount --move /mnt/userdirs /home
```

マウントポイントが正しく移動したことを確認するため、両方のディレクトリーのコンテンツを表示させます。

```
~]# ls /mnt/userdirs
~]# ls /home
jill  joe
```

2.3. ファイルシステムのアンマウント

以前にマウントしていたファイルシステムを切り離す場合は、以下のいずれかの **umount** コマンドを使用します。

```
umount directory
umount device
```

root としてログインしていない限り、ファイルシステムのマウントを解除する権限が必要であることに注意してください(「マウントオプションの指定」を参照してください)。使用例は、例2.9「CD のアンマウント」 を参照してください。

重要：ファイルシステムが使用されていないことを確認する

ファイルシステムが使用されている場合（たとえば、このファイルシステムでプロセスがファイルを読み取る場合）、**umount** コマンドを実行するとエラーを出して失敗します。次のように **fuser** コマンドを使用してファイルシステムにアクセスしているプロセスを判別します。

```
fuser -m directory
```

たとえば、**/media/cdrom/** ディレクトリーにマウントされているファイルシステムにアクセスしているプロセスの一覧を表示するには、次のコマンドを入力します。

```
~]$ fuser -m /media/cdrom
/media/cdrom:         1793  2013  2022  2435 10532c 10672c
```

例2.9 CD のアンマウント

以前に **/media/cdrom/** ディレクトリーにマウントされた CD をアンマウントするには、シェルプロンプトで次のように入力します。

```
~]$ umount /media/cdrom
```

2.4. 関連情報

コマンドなどの詳細については、以下のドキュメントをご覧ください。

2.4.1. インストールされているドキュメント

- **man 8 mount**: **mount** コマンドの man ページです。使い方などに関する詳細が記載されています。
- **man 8 umount**: **umount** コマンドの man ページです。使い方などに関する詳細が記載されています。
- **man 5 fstab**: **/etc/fstab** ファイル形式に関する詳細が記載されている man ページです。

2.4.2. 便利な Web サイト

- 『Shared subtrees』 – 共有サブツリーの概念について解説されている LWN の記事です。
- 『sharedsubtree.txt』: 共有サブツリーパッチに同梱されている拡張ドキュメント。

第3章 EXT3 ファイルシステム。

デフォルトのファイルシステムはジャーナリング *ext3* ファイルシステムです。

3.1. EXT3 の機能

ext3 ファイルシステムは、基本的に、ext2 ファイルシステムが拡張されたバージョンです。さまざまな改善点により、以下のような利点が提供されます。

可用性

予期しない停電やシステムクラッシュ (*クリーンでないシステムシャットダウン*とも言われる) が発生すると、マシンにマウントしている各 ext2 ファイルシステムは、**e2fsck** プログラムで整合性をチェックする必要があります。これは時間を浪費するプロセスであり、大量のファイルを含む大型ボリュームでは、システムの起動時間を著しく遅らせます。このプロセスの間、そのボリュームにあるデータは使用できません。

ext3 ファイルシステムで提供されるジャーナリングは、クリーンでないシステムシャットダウンが発生してもこの種のファイルシステムのチェックが不要であることを意味します。ext3 の使用していても整合性チェックが必要になる唯一の場面は、ハードドライブの障害が発生した場合など、ごく稀なハードウェア障害のケースのみです。クリーンでないシャットダウンの発生後に ext3 ファイルシステムを復元する時間は、ファイルシステムのサイズやファイルの数量ではなく、一貫性を維持するために使用される *ジャーナル* のサイズに依存します。デフォルトのジャーナルサイズは、ハードウェアの速度に応じて、復旧するのに約1秒かかります

データの整合性

ext3 ファイルシステムは、クリーンでないシステムシャットダウンが発生した際にデータの整合性が失われることを防止します。ext3 ファイルシステムにより、データが受けることのできる保護のタイプとレベルを選択できるようになります。デフォルトでは、ext3 ボリュームは、ファイルシステムの状態に関して高いレベルのデータの整合性を維持するように設定されています。

速度

一部のデータを複数回書き込みますが、ext3 のジャーナリングにより、ハードドライブのヘッドモーションが最適化されるため、ほとんどの場合、ext3 のスループットは ext2 よりも高くなります。速度を最適化するために 3 つのジャーナリングモードから選択できますが、システムに障害が発生する可能性のある状況では、モードの選択はデータの整合性がトレードオフの関係になることがあります。

簡単なトランジション

ext2 から ext3 に簡単に移行でき、再フォーマットをせずに、堅牢なジャーナリングファイルシステムの恩恵を受けることができます。このタスクの実行方法は、「ext3 ファイルシステムへの変換」を参照してください。

以下のセクションでは、ext3 パーティションを作成および調整する手順を説明します。ext2 パーティションについては、以下のパーティション分割とフォーマットのセクションをスキップして、直接「ext3 ファイルシステムへの変換」 に進んでください。

3.2. EXT3 ファイルシステムの作成

インストール後、ext3 ファイルシステムを新たに作成しないといけない場合があります。たとえば、システムに新しいディスクドライブを追加する場合は、ドライブのパーティション設定を行い、ext3 ファイルシステムを使用することができます。

ext3 ファイルシステムを作成する手順は次のとおりです。

1. **mkfs** を使用して、ext3 ファイルシステムでパーティションをフォーマットします。
2. **e2label** を使用してパーティションにラベルを付けます。

3.3. EXT3 ファイルシステムへの変換

tune2fs を使用すると、**ext2** ファイルシステムを **ext3** に変換できます。

注記

tune2fs を使用する前に、常に **e2fsck** ユーティリティーを使用してファイルシステムを確認してください。Red Hat Enterprise Linux のデフォルトのインストールでは、すべてのファイルシステムに ext3 を使用します。

ext2 ファイルシステムを **ext3** に変換するには、root としてログインし、ターミナルで以下のコマンドを入力します。

```
tune2fs -j <block_device>
```

&lt ;*block_device*> には、変換する ext2 ファイルシステムが含まれます。

有効なブロックデバイスは、以下の 2 つのタイプのエントリーのいずれかになります。

- マップされたデバイス - ボリュームグループの論理ボリューム（例：**/dev/mapper/VolGroup00-LogVol02** ）。
- 静的デバイス - 従来のストレージボリューム（例： **/dev/*hdbX***）。*hdb* はストレージデバイス名で、*X* はパーティション番号です。

df コマンドを実行して、マウントされたファイルシステムを表示します。

このセクションの残りの部分では、サンプルコマンドはブロックデバイスに以下の値を使用します。

```
/dev/mapper/VolGroup00-LogVol02
```

initrd イメージを再作成して、ext3 カーネルモジュールが含まれるようにする必要があります。これを作成するには、**mkinitrd** プログラムを実行します。**mkinitrd** コマンドの使用方法は、**man mkinitrd** と入力します。また、GRUB 設定が **initrd** を読み込んでいることを確認してください。

この変更を加えないと、システムは引き続き起動しますが、ファイルシステムは ext3 ではなく ext2 としてマウントされます。

3.4. EXT2 ファイルシステムへの復元

何らかの理由でパーティションを ext3 から ext2 に戻す場合は、最初に root としてログインして、パーティションをアンマウントして、以下を入力する必要があります。

```
umount /dev/mapper/VolGroup00-LogVol02
```

次に、root で以下のコマンドを入力して、ファイルシステムタイプを ext2 に変更します。

```
tune2fs -O ^has_journal /dev/mapper/VolGroup00-LogVol02
```

root で以下のコマンドを入力して、パーティションでエラーの有無を確認します。

```
e2fsck -y /dev/mapper/VolGroup00-LogVol02
```

次に、以下を入力して ext2 ファイルシステムとしてパーティションを再度マウントします。

```
mount -t ext2 /dev/mapper/VolGroup00-LogVol02 /mount/point
```

上記のコマンドで、*/mount/point* をパーティションのマウントポイントに置き換えます。

次に、マウント先のディレクトリーに変更し、パーティションのルートレベルで **.journal** ファイルを削除します。

```
rm -f .journal
```

これで ext2 パーティションができました。

パーティションを永続的に ext2 に変更する場合は、必ず **/etc/fstab** ファイルを更新するようにしてください。

第4章 EXT4 ファイルシステム

4.1. EXT4 の機能

ext4 ファイルシステムは、Red Hat Enterprise Linux 5 のデフォルトのファイルシステムである ext3 ファイルシステムのスケーラブルな拡張です。ext4 ファイルシステムは、最大 16 テラバイトのファイルおよびファイルシステムに対応します。また、サブディレクトリーの数を無制限にサポートします (ext3 ファイルシステムは最大 32,000 までしかサポートしません)。ただし、リンク数が 65,000 を超えると 1 にリセットされ、増加しなくなります。以下は、ext4 の最も重要な機能です。

主な特長

ext4 ファイルシステムはエクステントを使用します (ext2 および ext3 で使用される従来のブロックマッピングスキームとは異なります)。これにより、大きなファイルを使用する際のパフォーマンスが向上し、大きなファイルのメタデータオーバーヘッドが低減します。また、ext4 では、未使用のブロックグループと inode テーブルのセクションにそれぞれラベル付けが行なわれます。これにより、ファイルシステムのチェック時にこれらを省略することができます。また、ファイルシステムチェックの速度が上がるため、ファイルシステムが大きくなるほどその便宜性は顕著になります。

割り当て機能

Ext4 ファイルシステムには、以下のような割り当てスキームが備わっています。

- 永続的な事前割り当て
- 遅延割り当て
- マルチブロック割り当て
- ストライプ認識割り当て

遅延割り当てや他のパフォーマンスが最適化されるため、ext4 のディスクへのファイル書き込み動作は ext3 の場合とは異なります。ext4 では、プログラムが後で **fsync （）** 呼び出しを発行しない限り、ファイルシステムへの書き込みがオンディスクになる保証はありません。

ext3 では、**fsync()** の呼び出しがなくても、ファイルが新たに作成されると、そのほぼ直後にデフォルトでディスクへの書き込みが強制されます。この動作により、書き込まれたデータがオンディスクにあることを、**fsync()** を使用して確認しないというプログラムのバグが表面化しませんでした。一方、ext4 ファイルシステムは、ディスクへの変更書き込みの前に数秒間待機することが多く、書き込みを結合して再度順序付けを行うことにより、ext3 を上回るディスクパフォーマンスを実現しています。

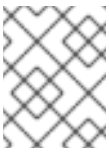
警告

ext3 とは異なり、ext4 ファイルシステムでは、トランザクションコミット時にディスクへのデータの書き込みを強制しません。このため、バッファーされた書き込みがディスクにフラッシュされるまでに時間がかかります。他のファイルシステムと同様、永続的なストレージにデータが書き込まれたことを確認するには、**fsync()** などのデータ整合性チェックの呼び出しを使用してください。

Ext4 のその他の機能

ext4 ファイルシステムでは次の機能にも対応しています。

- *拡張属性* (**xattr**)。これにより、システムはファイルごとに追加の名前と値のペアを関連付けることができます。
- *クォータジャーナリング*により、クラッシュ後の時間がかかるクォータの整合性チェックが不要になります。



注記

ext4 で対応しているジャーナリングモードは **data=ordered** (デフォルト) のみです。

- *サブセカンドのタイムスタンプ*: inode タイムスタンプフィールドをナノ秒で指定できるようにします。

4.2. EXT4 ファイルシステムの管理

Red Hat Eterprise Linux 5 で ext4 ファイルシステムを管理するには、e4fsprogs パッケージをインストールする必要があります。Yum ユーティリティーを使用して、パッケージをインストールできます。

```
~]# yum install e4fsprogs
```

e4fsprogs パッケージには、同等のアップストリーム e2fsprogs リリースから、静的バイナリーの名前が変更になりました。これは、ext4 の全変更を含む e2fsprogs コアユーティリティーの安定性を確保するために行われました。これらのユーティリティーで最も重要なのは、以下のとおりです。

- **mke4fs** - ext4 ファイルシステムの作成に使用するユーティリティー。
- **mkfs.ext4** - ext4 ファイルシステムの作成に使用するもう 1 つのコマンドです。
- **e4fsck** - ext4 ファイルシステムの不整合を修復するために使用されるユーティリティー。
- **tune4fs** - ext4 ファイルシステムの属性を変更するために使用されるユーティリティー。
- **resize4fs** - ext4 ファイルシステムのサイズを変更するのに使用するユーティリティー。
- **e4label** - ext4 ファイルシステムのラベルを表示または変更するために使用されるユーティリティー。
- **dumpe4fs** - ext4 ファイルシステムのスーパーブロックおよびブロックグループ情報を表示するのに使用するユーティリティー。
- **debuge4fs** - インタラクティブなファイルシステムデバッガーで、ext4 ファイルシステムを調べ、破損したファイルシステムを手動で修復し、**e4fsck** のテストケースを作成するために使用されます。

次のセクションでは、ext4 パーティションを作成および調整する手順を説明します。

4.3. EXT4 ファイルシステムの作成

インストール後、新しい ext4 ファイルシステムを作成する必要がある場合があります。たとえば、システムに新しいディスクドライブを追加する場合は、ドライブのパーティション設定を行い、ext4 ファイルシステムを使用することができます。

デフォルトのオプションは、ほとんどの使用シナリオに最適ですが、特定の方法で ext4 ファイルシステムを設定する必要がある場合は、利用可能なオプションについて **mke4fs** および **mkfs.ext4** コマンドの man ページを参照してください。ext4 ファイルシステムをより頻繁に作成する予定がある場合は、**mke4fs**、**/etc/mke4fs.conf** の設定ファイルを調べて変更することもできます。

ext4 ファイルシステムを作成する手順は次のとおりです。

1. **mkfs.ext4** または **mke4fs** コマンドを使用して、ext4 ファイルシステムでパーティションをフォーマットします。

   ```
   ~]# mkfs.ext4 block_device
   ```

   ```
   ~]# mke4fs -t ext4 block_device
   ```

   block_device は、作成する ext4 ファイルシステムを含むパーティションです。

2. **e4label** コマンドを使用して、パーティションにラベルを付けます。

   ```
   ~]# e4label <block_device> new-label
   ```

3. マウントポイントを作成し、新しいファイルシステムをそのマウントポイントにマウントします。

   ```
   ~]# mkdir /mount/point
   ~]# mount block_device /mount/point
   ```

有効なブロックデバイスは、以下の 2 つのタイプのエントリーのいずれかになります。

- *マップされたデバイス* - ボリュームグループの論理ボリューム（例：**/dev/mapper/VolGroup00-LogVol02** ）。
- *静的デバイス* - 従来のストレージボリューム（例： **/dev/*hdbX***）。*hdb* はストレージデバイス名で、*X* はパーティション番号です。

ストライプ化ブロックデバイス(RAID5 アレイなど)の場合は、ファイルシステムの作成時にストライプジオメトリーを指定できます。適切なストライプジオメトリーを使用すると、ext4 ファイルシステムのパフォーマンスが向上します。

lvm ボリュームまたは md ボリュームにファイルシステムを作成する場合、**mkfs.ext4** は最適なジオメトリーを選択します。オペレーティングシステムに配列情報をエクスポートするハードウェア RAID の中にも、こうした最適な配列を選択するものがあります。

ストライプジオメトリーを指定するには、以下のサブオプションとともに **mkfs.ext4** の **-E** オプション (拡張ファイルシステムオプション) を使用します。

stride=*value*

RAID チャンクサイズを指定します。

stripe-width=*value*

RAID デバイス内のデータディスク数、または 1 ストライプ内のストライプユニット数を指定します。

両方のサブオプションの場合、***value*** は、ファイルシステムのブロック単位で指定する必要があります。たとえば、4k ブロックのファイルシステムで、64k ストライド (16 x 4096) のファイルシステムを作成する場合は、次のコマンドを使用します。

```
~]# mkfs.ext4 -E stride=16,stripe-width=64 block_device
```

ファイルシステムの作成に関する詳細は、**man mkfs.ext4** を参照してください。

4.4. EXT4 ファイルシステムのマウント

ext4 ファイルシステムは、他のファイルシステムと同様に、追加オプションなしでマウントできます。

```
~]# mount block_device /mount/point
```

デフォルトのマウントオプションは、ほとんどのユーザーに最適です。acl、**no acl** 、**data**、quota、**noquota** 、user_xattr、**nouser_xattr** などのオプション、ext2 および ext3 ファイルシステムですでに使用されているものの多くは後方互換性があり、同じ使用方法や機能を持ちます。また、ext4 ファイルシステムでは、ext4 固有のマウントオプションが複数追加されました。以下に例を示します。

barrier / nobarrier

書き込みキャッシュが有効になっているデバイスへの電力供給が停止した場合でも、ファイルシステムの整合性を確保できるようにするため、ext4 ではデフォルトで書き込みバリアを使用します。書き込みキャッシュのないデバイス、またはバッテリーでバックアップされた書き込みキャッシュがあるデバイスの場合、**nobarrier** オプションを使用してバリアを無効にします。

```
~]# mount -o nobarrier block_device /mount/point
```

stripe=*value*

このオプションを使用すると、1 つのファイル操作に割り当てられるファイルシステムブロックの数を指定できます。RAID5 の場合、この数はディスク数で乗算した RAID チャンクサイズと同じでなければなりません。

journal_ioprio=*value*

このオプションを使用すると、コミット操作中に送信された I/O 操作の優先度を設定できます。オプションには、7 から 0 までの値を指定でき(0 が最も高い優先度)、はデフォルトでは 3 に設定されます。これは、デフォルトの I/O 優先度よりも若干高いです。

デフォルトのマウントオプションは、**tune4fs** ユーティリティーを使用してファイルシステムのスーパーブロックに設定することもできます。たとえば、次のコマンドは、デバッグを無効にし、ユーザー指定の拡張属性と Posix アクセス制御リストを有効にして、デフォルトで **/dev/mapper/VolGroup00-LogVol02** デバイスのファイルシステムをマウントするように設定します。

```
~]# tune4fs -o ^debug,user_xattr,acl /dev/mapper/VolGroup00-LogVol02
```

このトピックの詳細は、**tune4fs**(8) man ページを参照してください。

ext3 ファイルシステムは、形式を変更せずに ext4 としてマウントすることもできます。これにより、今後 ext3 として再度マウントすることができます。これを行うには、ext3 ファイルシステムを含むブロックデバイスで以下のコマンドを実行します。

```
~]# mount -t ext4 block_device /mount/point
```

そうすることで、ext3 ファイルシステムのみが、ファイル形式の変換を必要としない ext4 固有の機能を使用できます。これらの機能には、**遅延割り当て** と **マルチブロックの割り当て**、**エクステントマッピング** などの除外機能が含まれます。

警告

ext4 ドライバーを使用した ext3 ファイルシステムのマウントは、Red Hat Enterprise Linux 5 で完全にテストされていません。したがって、Red Hat はこの方法で ext3 ファイルシステムに対する一貫したパフォーマンスと予測可能な動作を保証することができ **ないため、このアクションはサポートされ** ません。

ext4 ファイルシステムのマウントオプションの詳細は、「マウントオプションの指定」 および **mount**(8)の man ページを参照してください。

注記

ファイルシステムの永続的なマウントを有効にする場合は、それに応じて **/etc/fstab** ファイルを更新するようにしてください。以下に例を示します。

```
/dev/mapper/VolGroup00-LogVol02    /test    ext4    defaults    0 0
```

4.5. EXT4 ファイルシステムのサイズ変更

ext4 ファイルシステムのサイズを大きくする前に、基礎となるブロックデバイスが将来的にファイルシステムを保持するのに十分なサイズであることを確認してください。該当するブロックデバイスのサイズを変更する場合は、ブロックデバイスに適した方法を選択してください。

拡張時には、ext4 ファイルシステムをマウントできます。縮小する際、ext4 ファイルシステムを **アンマウント** している必要があります。**resize4fs** コマンドを使用すると、ext4 ファイルシステムのサイズを変更できます。

```
~]# resize4fs block_devicenew_size
```

ext4 ファイルシステムのサイズを変更する際に、特定の単位を示す接尾辞が使用されていない限り、**resize2fs** ユーティリティーは、ファイルシステムのブロックサイズ単位でサイズを読み込みます。以下の接尾辞は、特定の単位を示しています。

- **s** – 512 バイトのセクター
- **K** – キロバイト
- **M** – メガバイト
- **G** – ギガバイト

拡張時には、**size** パラメーターは任意（多くの場合冗長）です。**resize4fs** は、コンテナーで利用可能な領域（通常は論理ボリュームまたはパーティション）をすべて埋めるように自動的に拡張します。ext4 ファイルシステムのサイズ変更に関する詳細は、**resize4fs**(8) man ページを参照してください。

第5章 PROC ファイルシステム

Linux カーネルには、コンピューター上の物理デバイスへのアクセスを制御し、それらのデバイスとプロセスがいつ、どのような方法で情報のやりとりを行うかをスケジュールするという、2 つの主要な機能があります。**/proc/** ディレクトリー(**proc** ファイルシステムとも呼ばれます)には、カーネルの現在の状態を表す特別なファイルの階層が含まれており、アプリケーションとユーザーがシステムのカーネルビューにピア接続できるようにします。

/proc/ ディレクトリー内で、システムのハードウェアと現在実行しているプロセスの詳細情報を見つけることができます。さらに、**/proc/** ディレクトリーツリー内のファイルの一部をユーザーおよびアプリケーションが操作して、設定の変更をカーネルに通信できます。

5.1. 仮想ファイルシステム

Linux では、すべてのデータはファイルとして保存されます。大半のユーザーは、主要な 2 つのファイルタイプ (テキストとバイナリー) について精通していますただし、**/proc/** ディレクトリーには、*仮想ファイル* と呼ばれる別のタイプのファイルが含まれます。このため、**/proc/** は *仮想ファイルシステム* と呼ばれることがよくあります。

これらの仮想ファイルには固有の特性があります。これらのほとんどはゼロバイトとして一覧表示され、表示されても大量の情報が含まれます。さらに、仮想ファイルの日時設定の大半は、現在の時刻と日付を反映しており、これらは常に更新されることを示しています。

/proc/interrupts、**/proc/meminfo**、**/proc/mounts**、および **/proc/partitions** などの仮想ファイルは、システムのハードウェアを最大から移動できます。**/proc/filesystems** ファイルや **/proc/sys/** ディレクトリーなどの他の場合は、システム設定情報とインターフェイスを提供します。

情報を体系化するために、同様のトピックに関する内容が記載されたファイルは、仮想ディレクトリー/サブディレクトリーにグループ化されます。たとえば、**/proc/ide/** には、すべての物理 IDE デバイスの情報が含まれます。同様に、プロセスディレクトリーには、システムで実行している各プロセスに関する情報が含まれます。

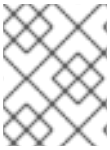
5.1.1. 仮想ファイルの表示

/proc/ ディレクトリー内のファイルで **cat** コマンド、**より多く** の、または **less** コマンドを使用すると、ユーザーはシステムに関する膨大な量の情報にすぐにアクセスできます。たとえば、コンピューターが持つ CPU のタイプを表示するには、**cat /proc/cpuinfo** と入力して、以下のような出力を受信します。

```
processor : 0
vendor_id : AuthenticAMD
cpu family : 5
model  : 9
model name : AMD-K6(tm) 3D+
Processor stepping : 1 cpu
MHz  : 400.919
cache size : 256 KB
fdiv_bug : no
hlt_bug  : no
f00f_bug : no
coma_bug : no
fpu  : yes
fpu_exception : yes
cpuid level : 1
```

```
wp  : yes
flags  : fpu vme de pse tsc msr mce cx8 pge mmx syscall 3dnow k6_mtrr
bogomips : 799.53
```

/proc/ ファイルシステムでさまざまな仮想ファイルを表示すると、人間が判読できないものの、一部の情報を簡単に理解できます。これは、仮想ファイルからデータを取得し、それを便利な方法で表示するユーティリティーが存在する理由の一部です。このユーティリティーの例には、**lspci**、**apm**、**free**、および **top** が含まれます。



注記

/proc/ ディレクトリー内の仮想ファイルの一部は、root ユーザーのみが読み取り可能です。

5.1.2. 仮想ファイルの変更

一般的なルールとして、**/proc/** ディレクトリー内のほとんどの仮想ファイルは読み取り専用です。ただし、一部の を使用してカーネルの設定を調整することができます。これは、**/proc/sys/** サブディレクトリー内のファイルに対して特に当てはまります。

仮想ファイルの値を変更するには、**echo** コマンドと、> より大きい記号(>)を使用して、新しい値をファイルにリダイレクトします。**たとえば、オンザフライでホスト名を変更するには、以下を入力します。**

```
echo www.example.com > /proc/sys/kernel/hostname
```

その他のファイルは、バイナリーまたはブール値のスイッチとして機能します。cat /proc/sys/net/ipv4/ip_forward を実行すると、0 または 1 のいずれかが返されます。0 は、カーネルがネットワークパケットを転送していないことを示します。echo コマンドを使用して ip_forward ファイルの値を 1 に変更すると、すぐにパケット転送が有効になります。

ヒント

/proc/sys/ サブディレクトリーで設定を変更するために使用されるもう 1つのコマンドは、/sbin/sysctl です。このコマンドの詳細については、を参照してください。 **「sysctl コマンドの使用」**

/proc/sys/ サブディレクトリーで利用可能なカーネル設定ファイルの一部の一覧は、 **「 /proc/sys/ 」** **を参照してください。**

5.1.3. プロセスディレクトリーへのアクセス制限

マルチユーザーシステムでは、/proc/ に保存されているプロセスディレクトリーを保護し、root ユーザーのみが表示できるようにすることが便利なことがよくあります。hidepid オプションを使用すると、これらのディレクトリーへのアクセスを制限できます。

ファイルシステムのパラメーターを変更するには、mount コマンドに -o remount オプションを指定して実行します。root で以下のコマンドを実行します。

```
mount -o remount,hidepid=value /proc
```

hidepid に渡される *値* は以下のいずれかになります。

- 0 (デフォルト): すべてのユーザーが、プロセスディレクトリーに保存されている全ユーザーが読み取り可能なファイルを読み取ることができます。
- 1 - ユーザーは自分のプロセスディレクトリーにのみアクセスできます。これにより、cmdline、sched、または status などの機密ファイルが root 以外のユーザーによるアクセスから保護されます。この設定は、実際のファイルパーミッションには影響しません。
- 2 - プロセスファイルは、root 以外のユーザーには表示されません。プロセスの存在は他の手段で学習できますが、有効な UID と GID は非表示になっています。これらの ID を非表示にすると、侵入者が実行中のプロセスに関する情報を収集するタスクが複雑になります。

例5.1 プロセスディレクトリーへのアクセス制限

root ユーザーのみがプロセスファイルにアクセスできるようにするには、以下を入力します。

```
~]# mount -o remount,hidepid=1 /proc
```

hidepid=1 を使用すると、root 以外のユーザーはプロセスディレクトリーのコンテンツにアクセスできません。これを試みると、以下のメッセージで失敗します。

```
~]$ ls /proc/1/
ls: /proc/1/: Operation not permitted
```

hidepid=2 を有効にすると、root 以外のユーザーにプロセスディレクトリーが非表示になります。

```
~]$ ls /proc/1/
ls: /proc/1/: No such file or directory
```

また、hidepid が 1 または 2 に設定されている場合でも、ファイルを処理するユーザーグループを指定できます。これを行うには、gid オプションを使用します。root で以下のコマンドを実行します。

```
mount -o remount,hidepid=value,gid=gid /proc
```

gid を、特定のグループ ID に置き換えます。選択したグループのメンバーの場合、プロセスファイルは hidepid が 0 に設定されているかのように動作します。ただし、システム全体のタスクを監視しないユーザーは、グループに追加しないでください。ユーザーおよびグループの管理に関する詳細は、37 章*ユーザーとグループ*を参照してください。

5.2. PROC ファイルシステム内のトップレベルファイル

以下は、/proc/ ディレクトリーの最上位にある、より有用な仮想ファイルの一覧です。

注記

ほとんどの場合、本セクションにリストされているファイルの内容は、マシンにインストールされた内容と同じではありません。これは、このドキュメント作業のために Red Hat Enterprise Linux を実行しているハードウェアに固有の情報の多くであるためです。

5.2.1. /proc/apm

このファイルは、*Advanced Power Management (APM)*システムの状態に関する情報を提供します。これは、apm コマンドによって使用されます。バッテリーのないシステムが AC 電源ソースに接続されている場合、この仮想ファイルは以下のようになります。

```
1.16 1.2 0x07 0x01 0xff 0x80 -1% -1 ?
```

このようなシステムで apm -v コマンドを実行すると、以下のような出力になります。

```
APM BIOS 1.2 (kernel driver 1.16ac) AC on-line, no system battery
```

電源ソースとしてバッテリーを使用しないシステムの場合、apm はマシンをスタンバイモードにするよりもわずかに実行できます。apm コマンドは、ラップトップで非常に便利です。たとえば、以下の出力は、電源アウトレットに接続しながら、ラップトップの cat /proc/apm コマンドにあります。

```
1.16 1.2 0x03 0x01 0x03 0x09 100% -1 ?
```

同じラップトップが電源ソースから数分間接続解除されると、apm ファイルの内容は以下のように変更されます。

```
1.16 1.2 0x03 0x00 0x00 0x01 99% 1792 min
```

apm -v コマンドは、以下のようなより有用なデータを提供するようになりました。

```
APM BIOS 1.2 (kernel driver 1.16) AC off-line, battery status high: 99% (1 day, 5:52)
```

5.2.2. /proc/buddyinfo

このファイルは、主にメモリーの断片化問題を診断するために使用されます。バディアルゴリズムを使用すると、各列は、任意の時点で利用可能な特定の注文（特定のサイズ）のページ数を表します。たとえば、ゾーン DMA（ダイレクトメモリーアクセス）の場合、90 of 2^(0*PAGE_SIZE)のメモリーのチャンクがあります。同様に、利用可能なメモリーの 6^(1*PAGE_SIZE)チャンクと 2 つの ^(2*PAGE_SIZE)チャンクがあります。

DMA 行は、システムの最初の 16 MB を参照し、HighMem 行はシステム上の 4 GB を超えるすべてのメモリーを参照し、Normal 行は の間のすべてのメモリーを参照します。

以下は、一般的な /proc/buddyinfo の出力例です。

```
Node 0, zone      DMA     90      6      2      1      1      ...
Node 0, zone   Normal   1650    310      5      0      0      ...
Node 0, zone  HighMem      2      0      0      1      1      ...
```

5.2.3. /proc/cmdline

このファイルは、起動時にカーネルに渡されるパラメーターを示します。/proc/cmdline ファイルの例を以下に示します。

```
ro root=/dev/VolGroup00/LogVol00 rhgb quiet 3
```

この出力は、以下を示しています。

ro

ルートデバイスは、システムの起動時に読み取り専用でマウントされます。カーネルブート行に **ro** が存在すると、**rw** のインスタンスが上書きされます。

root=/dev/VolGroup00/LogVol00

これは、どのディスクデバイスを使用するか、この場合は論理ボリューム(root ファイルシステムイメージがある)を指示します。**/proc/cmdline** の出力例では、root ファイルシステムイメージは、最初の LVM ボリュームグループ(**VolGroup00**)の最初の論理ボリューム(**LogVol00**)にあります。論理ボリューム管理を使用しないシステムでは、root ファイルシステムが **/dev/sda1** または **/dev/sda2** に置かれている可能性があります。つまり、そのドライブ上で別の（事前の）起動パーティションまたは swap パーティションがあるかどうかによって、最初の SCSI ディスクドライブまたは SATA ディスクドライブの最初のパーティションまたは 2 番目のパーティションのいずれかになります。

Red Hat Enterprise Linux で使用される LVM の詳細は、http://www.tldp.org/HOWTO/LVM-HOWTO/index.html を参照してください。

rhgb

Red Hat グラフィカル ブートを表す短い小文字の acronym で、カーネルコマンドラインで "rhgb" を指定すると、**/etc/inittab** でデフォルトのランレベルが 5 に設定されていることが前提となります。

```
id:5:initdefault:
```

quiet

起動時に非常に深刻なカーネルメッセージ以外のすべての詳細なカーネルメッセージを表示しないことを示します。

5.2.4. /proc/cpuinfo

この仮想ファイルは、システムが使用するプロセッサーのタイプを識別します。以下は、**/proc/cpuinfo** の一般的な出力の例です。

```
processor : 0
vendor_id : GenuineIntel
cpu family : 15
model  : 2
model name : Intel(R) Xeon(TM) CPU 2.40GHz
stepping : 7 cpu
MHz  : 2392.371
cache size : 512 KB
physical id : 0
siblings : 2
runqueue : 0
fdiv_bug : no
hlt_bug  : no
f00f_bug : no
coma_bug : no
fpu  : yes
fpu_exception : yes
cpuid level : 2
wp  : yes
```

```
flags  : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca  cmov pat pse36 clflush
dts acpi mmx fxsr sse sse2 ss ht tm
bogomips : 4771.02
```

- プロセッサー: 各プロセッサーに識別番号を提供します。プロセッサーが 1 つあるシステムでは、0 のみが存在します。
- CPU ファミリー: 作成者はシステム内のプロセッサーのタイプを識別します。Intel ベースのシステムの場合は、値を決定するために、86 の前に番号を付けます。これは、586、486、386 などの古いシステムのアーキテクチャーを特定しようとする場合に特に役立ちます。特定のアーキテクチャーごとに RPM パッケージがコンパイルされているので、この値はユーザーがインストールするパッケージを決定するのに役立ちます。
- モデル名: プロジェクト名を含むプロセッサーの共通名を表示します。
- CPU MHz - プロセッサーの megahertz の正確な速度を数十小数点に表示します。
- キャッシュサイズ: プロセッサーで利用可能なレベル 2 のメモリーキャッシュの量を表示します。
- siblings: ハイパースレッディングを使用するアーキテクチャーの同じ物理 CPU 上のシブリング CPU の数を表示します。
- フラグ: 浮動小数点ユニット(FPU)の存在や MMX 命令を処理する機能など、プロセッサーに関するさまざまな特性を定義します。

5.2.5. /proc/crypto

このファイルは、Linux カーネルによって使用されるインストール済み暗号の一覧を表示します。これには、それぞれの追加情報が含まれます。サンプルの /proc/crypto ファイルは以下のようになります。

```
name         : sha1
module       : kernel
type         : digest
blocksize    : 64
digestsize   : 20
name         : md5
module       : md5
type         : digest
blocksize    : 64
digestsize   : 16
```

5.2.6. /proc/devices

このファイルは、現在設定されているさまざまな文字およびブロックデバイスを表示します（モジュールが読み込まれていないデバイスは含まれません）。以下は、このファイルからの出力例です。

```
Character devices:
  1 mem
  4 /dev/vc/0
  4 tty
  4 ttyS
  5 /dev/tty
  5 /dev/console
  5 /dev/ptmx
```

```
  7 vcs
  10 misc
  13 input
  29 fb
  36 netlink
  128 ptm
  136 pts
  180 usb

Block devices:
  1 ramdisk
  3 ide0
  9 md
  22 ide1
  253 device-mapper
  254 mdp
```

/proc/devices の出力には、デバイスのメジャー番号と名前が含まれており、Character devices と Block devices の 2 つの主要なセクションに分かれています。

*キャラクターデバイス*は *ブロックデバイス* と似ていますが、以下の 2 つの基本的な相違点があります。

1. キャラクターデバイスにはバッファーは必要ありません。ブロックデバイスには利用可能なバッファーがあり、要求に対応する前にそれらを順序付けできます。これは、ハードドライブなどの情報を保存するために設計されたデバイスにとって重要です。デバイスに書き込む前に情報を順序付ける機能により、より効率的な順序で情報を配置することができるためです。

2. キャラクターデバイスは、サイズが事前設定されていないデータを送信します。ブロックデバイスは、デバイスごとに設定されたサイズのブロック内の情報を送受信できます。

デバイスの詳細は、以下のインストール済みドキュメントを参照してください。

```
/usr/share/doc/kernel-doc-<version>/Documentation/devices.txt
```

5.2.7. /proc/dma

このファイルには、使用中の登録済み ISA DMA チャンネルのリストが含まれています。サンプルの /proc/dma ファイルは以下のようになります。

```
4: cascade
```

5.2.8. /proc/execdomains

このファイルは、Linux カーネルが *現在サポートしている実行ドメイン* と、それらがサポートしている個人範囲を一覧表示します。

```
0-0   Linux           [kernel]
```

実行ドメインをオペレーティングシステムの personality とみなします。Solaris、UnixWare、FreeBSD などの他のバイナリー形式は Linux で使用できるため、プログラマーは、タスクのパーソナリティーを変更することで、オペレーティングシステムがバイナリーからのシステムコールを処理する方法を変更できます。PER_LINUX 実行ドメインを除き、さまざまなパーソナリティーを動的にロード可能なモジュールとして実装できます。

5.2.9. /proc/fb

このファイルには、フレームバッファーデバイス番号と、それを制御するドライバーが含まれるフレームバッファーデバイスの一覧が含まれます。フレームバッファーデバイスを含むシステムの /proc/fb の一般的な出力は以下のようになります。

```
0 VESA VGA
```

5.2.10. /proc/filesystems

このファイルは、カーネルで現在対応しているファイルシステムタイプの一覧を表示します。一般的な /proc/filesystems ファイルからの出力例を以下に示します。

```
nodev   sysfs
nodev   rootfs
nodev   bdev
nodev   proc
nodev   sockfs
nodev   binfmt_misc
nodev   usbfs
nodev   usbdevfs
nodev   futexfs
nodev   tmpfs
nodev   pipefs
nodev   eventpollfs
nodev   devpts
 ext2
nodev   ramfs
nodev   hugetlbfs
 iso9660
nodev   mqueue
 ext3
nodev   rpc_pipefs
nodev   autofs
```

最初の列は、ファイルシステムがブロックデバイスにマウントされているかどうかを示します。nodev で始まるものは、デバイスにマウントされません。2 番目のコラムには、サポートされているファイルシステムの名前が記載されています。

mount コマンドは、引数として指定されていない場合に、ここにリストされているファイルシステムを循環します。

5.2.11. /proc/interrupts

このファイルは、x86 アーキテクチャーの IRQ ごとの割り込み数を記録します。標準の /proc/interrupts は以下のようになります。

```
  CPU0
  0:   80448940        XT-PIC  timer
  1:     174412        XT-PIC  keyboard
  2:          0        XT-PIC  cascade
  8:          1        XT-PIC  rtc
 10:     410964        XT-PIC  eth0
 12:      60330        XT-PIC  PS/2 Mouse
```

```
 14:    1314121          XT-PIC  ide0
 15:    5195422          XT-PIC  ide1
NMI:          0
ERR:          0
```

マルチプロセッサーマシンの場合、このファイルは若干異なる場合があります。

```
    CPU0       CPU1
 0: 1366814704          0          XT-PIC  timer
 1:        128        340    IO-APIC-edge  keyboard
 2:          0          0          XT-PIC  cascade
 8:          0          1    IO-APIC-edge  rtc
12:       5323       5793    IO-APIC-edge  PS/2 Mouse
13:          1          0          XT-PIC  fpu
16:   11184294   15940594   IO-APIC-level  Intel EtherExpress Pro 10/100 Ethernet
20:    8450043   11120093   IO-APIC-level  megaraid
30:      10432      10722   IO-APIC-level  aic7xxx
31:         23         22   IO-APIC-level  aic7xxx
NMI:          0
ERR:          0
```

最初の列は IRQ 番号を参照します。システムの各 CPU には、独自の列と IRQ ごとに独自の割り込み数があります。次の列は割り込みのタイプを報告し、最後の列にはその IRQ にあるデバイスの名前が含まれます。

このファイルに表示される割り込みのタイプ（アーキテクチャー固有のもの）は、それぞれ異なることを意味します。x86 マシンでは、以下の値が一般的です。

- **XT-PIC**: これは古い AT コンピューター割り込みです。
- **IO-APIC-edge** - この割り込みの電圧シグナルは低から高に移行され、割り込みが発生して 1回だけシグナルを受ける エッジ が作成されます。この種の割り込みと **IO-APIC** レベル の割り込みは、586 ファミリー以降のプロセッサーを持つシステムでのみ表示されます。
- **IO-APIC-level** - シグナルが再び低いまで電圧シグナルが高い場合に割り込みを生成します。

5.2.12. /proc/iomem

このファイルは、各物理デバイスのシステムのメモリーの現在のマップを表示します。

```
00000000-0009fbff : System RAM
0009fc00-0009ffff : reserved
000a0000-000bffff : Video RAM area
000c0000-000c7fff : Video ROM
000f0000-000fffff : System ROM
00100000-07ffffff : System RAM
00100000-00291ba8 : Kernel code
00291ba9-002e09cb : Kernel data
e0000000-e3ffffff : VIA Technologies, Inc. VT82C597 [Apollo VP3] e4000000-e7ffffff : PCI Bus
#01
e4000000-e4003fff : Matrox Graphics, Inc. MGA G200 AGP
e5000000-e57fffff : Matrox Graphics, Inc. MGA G200 AGP
e8000000-e8ffffff : PCI Bus #01
```

```
e8000000-e8ffffff : Matrox Graphics, Inc. MGA G200 AGP
ea000000-ea00007f : Digital Equipment Corporation DECchip 21140 [FasterNet]
ea000000-ea00007f : tulip ffff0000-ffffffff : reserved
```

最初の列には、さまざまなタイプのメモリーで使用されるメモリーレジスターが表示されます。2 番目のコラムは、それらのレジスター内にあるメモリーの種類を一覧表示し、システム RAM 内のカーネルが使用するメモリーレジスターを表示します。または、ネットワークインターフェイスカードに複数のイーサネットポートがある場合は、ポートごとに割り当てられるメモリーレジスターを表示します。

5.2.13. /proc/ioports

/proc/ioports の出力は、デバイスとの入出力通信に使用される現在登録されているポートリージョンの一覧を提供します。このファイルは非常に長くなる可能性があります。以下は部分的なリストです。

```
0000-001f : dma1
0020-003f : pic1
0040-005f : timer
0060-006f : keyboard
0070-007f : rtc
0080-008f : dma page reg
00a0-00bf : pic2
00c0-00df : dma2
00f0-00ff : fpu
0170-0177 : ide1
01f0-01f7 : ide0
02f8-02ff : serial(auto)
0376-0376 : ide1
03c0-03df : vga+
03f6-03f6 : ide0
03f8-03ff : serial(auto)
0cf8-0cff : PCI conf1
d000-dfff : PCI Bus #01
e000-e00f : VIA Technologies, Inc. Bus Master IDE
e000-e007 : ide0
e008-e00f : ide1
e800-e87f : Digital Equipment Corporation DECchip 21140 [FasterNet]
e800-e87f : tulip
```

最初の列は、2 番目のコラムに一覧表示されるデバイス用に予約された I/O ポートアドレス範囲を示します。

5.2.14. /proc/kcore

このファイルは、システムの物理メモリーを表し、コアファイル形式で保存されます。ほとんどの /proc/ ファイルとは異なり、kcore はサイズを表示します。この値はバイト単位で指定され、使用される物理メモリー(RAM)のサイズに 4 KB を加えたサイズと同じです。

このファイルのコンテンツは、gdb などのデバッガーによって検査されるように設計されており、人間は読み取りできません。

注意

/proc/kcore 仮想ファイルは表示しないでください。端末上のファイルの Sramble テキスト出力の内容。このファイルが誤って表示されたら、Ctrl+C を押してプロセスを停止し、reset と入力してコマンドラインプロンプトを元に戻します。

5.2.15. /proc/kmsg

このファイルは、カーネルによって生成されたメッセージを保持するために使用されます。これらのメッセージは、/sbin/klogd や /bin/dmesg などの他のプログラムによって取得されます。

5.2.16. /proc/loadavg

このファイルは、時間の経過とともに CPU と IO の両方、および アップタイム およびその他のコマンドで使用される追加のデータに関する平均負荷を提供します。/proc/loadavg ファイルの例を以下に示します。

```
0.20 0.18 0.12 1/80 11206
```

最初の 3 列は、過去 1 分、5 分、15 分間の CPU および IO 使用率を測定します。4 列目には、現在実行中のプロセスの数と、プロセスの合計数が表示されます。最後の列には、最後に使用したプロセス ID が表示されます。

さらに、負荷平均とは、実行可能なプロセスの数（つまり、実行キューで CPU 共有を待機しているプロセスの数）を指します。

5.2.17. /proc/locks

このファイルは、カーネルによって現在ロックされているファイルを表示します。このファイルのコンテンツには、内部のカーネルデバッグデータが含まれており、システムの使用により大きく異なる可能性があります。負荷の少ないシステムの /proc/locks ファイルの例を以下に示します。

```
1: POSIX  ADVISORY  WRITE 3568 fd:00:2531452 0 EOF
2: FLOCK  ADVISORY  WRITE 3517 fd:00:2531448 0 EOF
3: POSIX  ADVISORY  WRITE 3452 fd:00:2531442 0 EOF
4: POSIX  ADVISORY  WRITE 3443 fd:00:2531440 0 EOF
5: POSIX  ADVISORY  WRITE 3326 fd:00:2531430 0 EOF
6: POSIX  ADVISORY  WRITE 3175 fd:00:2531425 0 EOF
7: POSIX  ADVISORY  WRITE 3056 fd:00:2548663 0 EOF
```

各ロックには、一意の番号で始まる独自の行があります。2 番目の列は、使用されるロッククラスを指し、FLOCK は flock システムコールからの古いスタイルの UNIX ファイルロックを示し、lockf システムコールから新しい POSIX ロックを表します。

3 列目には、ADVISORY または MANDATORY の 2 つの値を指定できます。ADVISORY は、ロックが他のユーザーがデータにアクセスできないことを意味します。他のユーザーがロックするのを防ぐだけです。MANDATORY は、ロックが保持される間、データへの他のアクセスが許可されないことを意味します。4 番目のコラムは、ロックが所有者の READ または WRITE アクセスを許可するかどうかを示します。5 番目のコラムには、ロックを保持するプロセスの ID が表示されます。6 番目のコラムには、

ロックされているファイルの ID が *MAJOR-DEVICE*:*MINOR-DEVICE*:*INODE-NUMBER* の形式で表示されます。7 番目と 8 番目のコラムは、ファイルのロックされたリージョンの開始と終了を示しています。

5.2.18. /proc/mdstat

このファイルには、複数ディスク、RAID 設定に関する現在の情報が含まれています。システムにこのような設定が含まれていない場合、/proc/mdstat は以下のようになります。

```
Personalities :  read_ahead not set unused devices: <none>
```

このファイルは、ソフトウェア RAID または md デバイスが存在しない限り、上記と同じ状態のままになります。この場合は、/proc/mdstat を表示して、md*X* RAID デバイスの現在の状態を見つけます。

以下の /proc/mdstat ファイルは、md0 が RAID 1 デバイスとして設定されているシステムを示していますが、現在ディスクを再同期しています。

```
Personalities : [linear] [raid1] read_ahead 1024 sectors
md0: active raid1 sda2[1] sdb2[0] 9940 blocks [2/2] [UU] resync=1% finish=12.3min algorithm 2
[3/3] [UUU]
unused devices: <none>
```

5.2.19. /proc/meminfo

これは、システムの RAM 使用率に関する多くの重要な情報を報告するため、/proc/ ディレクトリーで一般的に使用されるファイルの 1 つです。

以下の /proc/meminfo 仮想ファイルの例は、256 MB の RAM と 512 MB のスワップ領域があるシステムのものです。

```
MemTotal:      255908 kB
MemFree:        69936 kB
Buffers:        15812 kB
Cached:        115124 kB
SwapCached:         0 kB
Active:         92700 kB
Inactive:       63792 kB
HighTotal:          0 kB
HighFree:           0 kB
LowTotal:      255908 kB
LowFree:        69936 kB
SwapTotal:     524280 kB
SwapFree:      524280 kB
Dirty:              4 kB
Writeback:          0 kB
Mapped:         42236 kB
Slab:           25912 kB
Committed_AS:   118680 kB
PageTables:      1236 kB
VmallocTotal:  3874808 kB
VmallocUsed:     1416 kB
VmallocChunk:  3872908 kB
```

```
HugePages_Total:     0
HugePages_Free:      0
Hugepagesize:     4096 kB
```

ここでの情報の多くは、無料 の、top コマンド、および ps コマンドで使用されます。実際、free コマンドの出力は、/proc/meminfo の内容と構造と同様のものです。ただし、/proc/meminfo で直接確認すると、詳細が表示されます。

- MemTotal: 物理 RAM の合計容量（キロバイト単位）。
- MemFree - システムが使用していない物理メモリーの量（キロバイト単位）。
- バッファー - ファイルバッファーに使用される物理 RAM の容量（キロバイト単位）。
- cached - キャッシュメモリーとして使用される物理メモリーの量（キロバイト単位）。
- SwapCached: キャッシュメモリーとして使用されるスワップの量（キロバイト単位）。
- Active - アクティブな使用時のバッファーまたはページキャッシュメモリーの合計量（キロバイト単位）。これは最近使用されたメモリーであり、通常は他の目的で回収されません。
- inactive - 空きかつ利用可能なバッファーまたはページキャッシュメモリーの合計量（キロバイト単位）。これは最近使用されていないメモリーであり、他の目的で回収できます。
- HighTotal and HighFree: カーネル領域に直接マッピングされないメモリーの合計および空き容量（キロバイト単位）。HighTotal 値は、使用されるカーネルのタイプによって異なります。
- LowTotal and LowFree: カーネル領域に直接マップされるメモリーの合計および空き容量（キロバイト単位）。LowTotal 値は、使用されるカーネルのタイプによって異なります。
- SwapTotal - 利用可能なスワップの合計量（キロバイト単位）。
- swapfree - 空きスワップの合計量（キロバイト単位）。
- dirty: ディスクに書き戻されるのを待つメモリーの合計量（キロバイト単位）。
- writeback - ディスクにアクティブに書き込むメモリーの合計量（キロバイト単位）。
- mapped - mmap コマンドを使用してデバイス、ファイル、またはライブラリーをマッピングするために使用されたメモリーの合計量（キロバイト単位）。
- slab - カーネルが独自の使用のためにデータ構造をキャッシュするために使用するメモリーの合計量（キロバイト単位）。
- Committed_AS: ワークロードの完了に推定されるメモリーの合計量（キロバイト単位）。この値は最も悪いケースのシナリオ値を表し、スワップメモリーも含まれます。
- pagetables - 最小のページテーブルレベル専用のメモリーの合計量（キロバイト単位）。
- VMallocTotal - 割り当てられた仮想アドレス空間の合計量（キロバイト単位）。
- VMallocUsed - 使用されている仮想アドレス空間の合計量（キロバイト単位）。
- VMallocChunk: 利用可能な仮想アドレス空間の最大連続するメモリーブロック（キロバイト単位）。
- HugePages_Total: システムのヒュージページの合計数この数は、/proc/sys/vm/hugetlb_pool で指定されたヒュージページ用に確保されるメガバイト単位で dividing----------|----- size に

よって派生します。この統計は、x86、Itanium、および AMD64 アーキテクチャーにのみ表示されます。

- **HugePages_Free**: システムで利用可能なヒュージページの合計数この統計は、x86、Itanium、および AMD64 アーキテクチャーにのみ表示されます。
- **ubuntusize** - 各ヒュージページユニットのサイズ（キロバイト単位）。デフォルトでは、32 ビットアーキテクチャーの uniprocessor カーネルでは、値は 4096 KB です。SMP、hugemem カーネル、および AMD64 の場合、デフォルトは 2048 KB です。Itanium アーキテクチャーの場合、デフォルトは 262144 KB です。この統計は、x86、Itanium、および AMD64 アーキテクチャーにのみ表示されます。

5.2.20. /proc/misc

このファイルは、その他のメジャーデバイス（デバイス番号 10）に登録されているその他のドライバーを一覧表示します。

```
63 device-mapper 175 agpgart 135 rtc 134 apm_bios
```

最初の列は各デバイスのマイナー番号で、2 番目のコラムには使用中のドライバーが表示されます。

5.2.21. /proc/modules

このファイルは、カーネルにロードされているすべてのモジュールの一覧を表示します。そのコンテンツはシステムの設定や使用方法によって異なりますが、このサンプル **/proc/modules** ファイル出力と同様の方法で整理する必要があります。

注記

この例は、読み取り可能な形式に再フォーマットされました。この情報のほとんどは、**/sbin/lsmod** コマンドで表示することもできます。

```
nfs     170109  0 -         Live 0x129b0000
lockd    51593   1 nfs,      Live 0x128b0000
nls_utf8 1729    0 -         Live 0x12830000
vfat     12097   0 -         Live 0x12823000
fat      38881   1 vfat,     Live 0x1287b000
autofs4  20293   2 -         Live 0x1284f000
sunrpc   140453  3 nfs,lockd, Live 0x12954000
3c59x    33257   0 -         Live 0x12871000
uhci_hcd 28377   0 -         Live 0x12869000
md5      3777    1 -         Live 0x1282c000
ipv6     211845 16 -         Live 0x128de000
ext3     92585   2 -         Live 0x12886000
jbd      65625   1 ext3,     Live 0x12857000
dm_mod   46677   3 -         Live 0x12833000
```

最初の列には、モジュール名が含まれます。

2 番目の列は、モジュールのメモリーサイズ（バイト単位）を参照します。

3 列目には、現在読み込まれているモジュールのインスタンス数が記載されています。ゼロの値は、アンロードされたモジュールを表します。

4 番目のコラムは、モジュールが機能するために別のモジュールに依存し、他のモジュールを一覧表示するかどうかを示します。

5 番目のコラムには、モジュールがどの負荷状態であるかが記載されています。Live、Loading、または Unloading は唯一の可能な値です。

6 番目のコラムには、ロードされたモジュールの現在のカーネルメモリーオフセットが一覧表示されます。この情報は、デバッグや oprofile などのプロファイリングツールに役立ちます。

5.2.22. /proc/mounts

このファイルは、システムで使用されているすべてのマウントの一覧を表示します。

```
rootfs / rootfs rw 0 0
/proc /proc proc rw,nodiratime 0 0 none
/dev ramfs rw 0 0
/dev/mapper/VolGroup00-LogVol00 / ext3 rw 0 0
none /dev ramfs rw 0 0
/proc /proc proc rw,nodiratime 0 0
/sys /sys sysfs rw 0 0
none /dev/pts devpts rw 0 0
usbdevfs /proc/bus/usb usbdevfs rw 0 0
/dev/hda1 /boot ext3 rw 0 0
none /dev/shm tmpfs rw 0 0
none /proc/sys/fs/binfmt_misc binfmt_misc rw 0 0
sunrpc /var/lib/nfs/rpc_pipefs rpc_pipefs rw 0 0
```

ここで見つかった出力は /etc/mtab の内容と似ていますが、/proc/mount が最新です。

最初の列は、マウントされるデバイスを指定し、2 番目の列はマウントポイントを示し、3 番目のコラムはファイルシステムタイプを示し、3 番目のコラムは読み取り専用(ro)または読み取り/書き込み(rw)がマウントされているかどうかを示します。5 番目と 6 番目の列は、/etc/mtab で使用される形式に一致するように設計されたダミー値です。

5.2.23. /proc/mtrr

このファイルは、システムで使用されている現在の Memory Type Range Registers (MTRR)を参照します。システムアーキテクチャーが MTRR に対応している場合は、/proc/mtrr ファイルは以下のようになります。

```
reg00: base=0x00000000 (   0MB), size= 256MB: write-back, count=1
reg01: base=0xe8000000 (3712MB), size=  32MB: write-combining, count=1
```

MTRR は、Intel P6 ファミリーのプロセッサー(Pentium II 以降)とともに使用され、メモリー範囲へのプロセッサーアクセスを制御します。PCI または AGP バスでビデオカードを使用する場合は、適切に設定された /proc/mtrr ファイルにより、150% を超えるパフォーマンスが向上します。

多くの場合、この値はデフォルトで適切に設定されます。このファイルを手動で設定する方法は、以下の場所を参照してください。

```
/usr/share/doc/kernel-doc-<version>/Documentation/mtrr.txt
```

5.2.24. /proc/partitions

このファイルには、パーティションブロック割り当て情報が含まれます。基本的なシステムからのこのファイルのサンプリングは以下のようになります。

```
major minor  #blocks  name
  3     0   19531250 hda
  3     1     104391 hda1
  3     2   19422585 hda2
253     0   22708224 dm-0
253     1     524288 dm-1
```

ここでの情報のほとんどは、以下の列を除き、ユーザーにとって重要ではありません。

- major: このパーティションを持つデバイスのメジャー番号。/proc/partitions のメジャー番号 (3)は、 /proc/devices のブロックデバイス ide0 に対応します。
- minor: このパーティションを持つデバイスのマイナー番号。これは、パーティションを異なる物理デバイスに分割し、パーティション名の末尾の番号に関連するものです。
- #blocks - 特定のパーティションに含まれる物理ディスクブロックの数を一覧表示します。
- 名前 - パーティションの名前。

5.2.25. /proc/pci

このファイルには、システム上のすべての PCI デバイスの完全一覧が含まれます。PCI デバイスの数によっては、/proc/pci が長くなる可能性があります。基本的なシステムからのこのファイルのサンプリングは以下のようになります。

```
Bus  0, device 0, function 0: Host bridge: Intel Corporation 440BX/ZX - 82443BX/ZX Host bridge (rev 3). Master Capable. Latency=64. Prefetchable 32 bit memory at 0xe4000000 [0xe7ffffff].
Bus  0, device 1, function 0: PCI bridge: Intel Corporation 440BX/ZX - 82443BX/ZX AGP bridge (rev 3).   Master Capable. Latency=64. Min Gnt=128.
Bus  0, device 4, function 0: ISA bridge: Intel Corporation 82371AB PIIX4 ISA (rev 2).
Bus  0, device 4, function 1: IDE interface: Intel Corporation 82371AB PIIX4 IDE (rev 1). Master Capable. Latency=32. I/O at 0xd800 [0xd80f].
Bus  0, device 4, function 2: USB Controller: Intel Corporation 82371AB PIIX4 USB (rev 1). IRQ 5. Master Capable. Latency=32. I/O at 0xd400 [0xd41f].
Bus  0, device 4, function 3: Bridge: Intel Corporation 82371AB PIIX4 ACPI (rev 2). IRQ 9.
Bus  0, device 9, function 0: Ethernet controller: Lite-On Communications Inc LNE100TX (rev 33). IRQ 5. Master Capable. Latency=32. I/O at 0xd000 [0xd0ff].
Bus  0, device 12, function  0: VGA compatible controller: S3 Inc. ViRGE/DX or /GX (rev 1). IRQ 11. Master Capable. Latency=32. Min Gnt=4.Max Lat=255.
```

この出力には、バス、デバイス、および機能順にソートされたすべての PCI デバイスの一覧が表示されます。デバイスの名前とバージョンを指定する以外に、この一覧は詳細な IRQ 情報も提供します。これにより、管理者は競合をすばやく検索できます。

ヒント

この情報のより読みやすいバージョンを取得するには、以下を入力します。

```
lspci -vb
```

5.2.26. /proc/slabinfo

このファイルは、スラブレベルでのメモリー使用量に関する詳細情報を提供します。バージョン 2.2 より大きい Linux カーネルは、*slab プール*を使用してページレベルの上のメモリーを管理します。一般的に使用されるオブジェクトには、独自のスラブプールがあります。

非常に詳細な /proc/slabinfo ファイルを手動で解析する代わりに、/usr/bin/slabtop プログラムはカーネルスラブキャッシュ情報をリアルタイムで表示します。このプログラムは、列のソートや画面のリフレッシュなど、カスタム設定を行うことができます。

通常、/usr/bin/slabtop のスクリーンショットは以下のようになります。

```
Active / Total Objects (% used)    : 133629 / 147300 (90.7%)
Active / Total Slabs (% used)      : 11492 / 11493 (100.0%)
Active / Total Caches (% used)     : 77 / 121 (63.6%)
Active / Total Size (% used)       : 41739.83K / 44081.89K (94.7%)
Minimum / Average / Maximum Object : 0.01K / 0.30K / 128.00K
OBJS   ACTIVE USE      OBJ   SIZE     SLABS OBJ/SLAB CACHE SIZE NAME
44814  43159  96%    0.62K   7469      6     29876K ext3_inode_cache
36900  34614  93%    0.05K    492     75      1968K buffer_head
35213  33124  94%    0.16K   1531     23      6124K dentry_cache
7364   6463  87%    0.27K    526     14      2104K radix_tree_node
2585   1781  68%    0.08K     55     47       220K vm_area_struct
2263   2116  93%    0.12K     73     31       292K size-128
1904   1125  59%    0.03K     16    119        64K size-32
1666    768  46%    0.03K     14    119        56K anon_vma
1512   1482  98%    0.44K    168      9       672K inode_cache
1464   1040  71%    0.06K     24     61        96K size-64
1320    820  62%    0.19K     66     20       264K filp
678    587  86%    0.02K      3    226        12K dm_io
678    587  86%    0.02K      3    226        12K dm_tio
576    574  99%    0.47K     72      8       288K proc_inode_cache
528    514  97%    0.50K     66      8       264K size-512
492    372  75%    0.09K     12     41        48K bio
465    314  67%    0.25K     31     15       124K size-256
452    331  73%    0.02K      2    226         8K biovec-1
420    420 100%    0.19K     21     20        84K skbuff_head_cache
305    256  83%    0.06K      5     61        20K biovec-4
290      4   1%    0.01K      1    290         4K revoke_table
264    264 100%    4.00K    264      1      1056K size-4096
260    256  98%    0.19K     13     20        52K biovec-16
260    256  98%    0.75K     52      5       208K biovec-64
```

/usr/bin/slabtop に含まれる /proc/slabinfo で、より一般的に使用される統計の一部は次のとおりです。

- OBJS - 使用中のオブジェクト（メモリーブロック）や使用されていないスペアを含むオブジェクトの合計数（メモリーブロック）。
- ACTIVE - 使用中のオブジェクト（メモリーブロック）の数（割り当て済み）
- USE - アクティブなオブジェクトの合計パーセンテージ((ACTIVE/OBJS) (100))
- OBJ SIZE - オブジェクトのサイズ
- SLABS: スラブの合計数。

- **OBJ/SLAB**: スラブに適合するオブジェクト数。
- **CACHE SIZE**: スラブのキャッシュサイズ。
- **NAME** - slab の名前。

/usr/bin/slabtop プログラムの詳細は、**slabtop** の man ページを参照してください。

5.2.27. /proc/stat

このファイルは、最後に再起動されてからシステムに関するさまざまな統計を追跡します。非常に長い /proc/stat の内容は通常、以下の例のように開始します。

```
cpu  259246 7001 60190 34250993 137517 772 0
cpu0 259246 7001 60190 34250993 137517 772 0
intr 354133732 347209999 2272 0 4 4 0 0 3 1 1249247 0 0 80143 0 422626 5169433
ctxt 12547729
btime 1093631447
processes 130523
procs_running 1
procs_blocked 0
preempt 5651840
cpu  209841 1554 21720 118519346 72939 154 27168
cpu0 42536 798 4841 14790880 14778 124 3117
cpu1 24184 569 3875 14794524 30209 29 3130
cpu2 28616 11 2182 14818198 4020 1 3493
cpu3 35350 6 2942 14811519 3045 0 3659
cpu4 18209 135 2263 14820076 12465 0 3373
cpu5 20795 35 1866 14825701 4508 0 3615
cpu6 21607 0 2201 14827053 2325 0 3334
cpu7 18544 0 1550 14831395 1589 0 3447
intr 15239682 14857833 6 0 6 6 0 5 0 1 0 0 0 29 0 2 0 0 0 0 0 0 0 94982 0 286812
ctxt 4209609
btime 1078711415
processes 21905
procs_running 1
procs_blocked 0
```

より一般的に使用される統計には以下が含まれます。

- **cpu** - システムがユーザーモード、低優先度(nice)、システムモード、アイドルタスク、I/O 待機、IRQ (hardirq)、softirq になっている *jiffies* の数(1/100)を表します。IRQ (hardirq)は、ハードウェアイベントへの直接応答です。IRQ は、softirq の実行で大きさの作業をキューに入れるには最小限の作業を行います。softirq は IRQ よりも優先度が低いため、頻繁に中断される可能性があります。すべての CPU の合計は上部に表示され、各 CPU は以下の独自の統計と共に一覧表示されます。以下の例は、マルチスレッドが有効な 4 方向の Intel Pentium Xeon 設定であるため、4 つの物理プロセッサーと、合計 8 つのプロセッサー 4 つの仮想プロセッサーを表示します。
- **ページ** - システムがディスクに書き込んだメモリーページ数。
- **swap**: システムが送受信したスワップページの数。
- **intr** - システムが経験した割り込みの数。

- btime - 起動時間(1970 年 1 月 1 日からの秒数で測定)で、それ以外はエポックと呼ばれています。

5.2.28. /proc/swaps

このファイルは、スワップ領域とその使用率を測定します。スワップパーティションが 1 つしかない場合は、/proc/swaps の出力は以下のようになります。

```
Filename                        Type        Size    Used    Priority
/dev/mapper/VolGroup00-LogVol01 partition   524280  0       -1
```

この情報の一部は /proc/ ディレクトリー内の他のファイルにありますが、/proc/swaps は各スワップファイル名のスナップショット、スワップ領域のタイプ、合計サイズ、および使用中の領域のサイズ（キロバイト単位）を提供します。優先度の列は、複数のスワップファイルが使用されている場合に役立ちます。優先度が低いほど、スワップファイルが使用される可能性が高くなります。

5.2.29. /proc/sysrq-trigger

echo コマンドを使用してこのファイルに書き込むと、リモートの root ユーザーは、ローカル端末のようにほとんどの System Request Key コマンドをリモートで実行できます。このファイルに値を echo するには、/proc/sys/kernel/sysrq を 0 以外の値に設定する必要があります。システム要求キーの詳細は、「/proc/sys/kernel/」を参照してください。

このファイルに書き込むことはできますが、root ユーザーであっても読み取りを行うことはできません。

5.2.30. /proc/uptime

このファイルには、最後の再起動以降にシステムがどのくらいの時間であるかを詳細に説明します。/proc/uptime の出力は非常に最小限です。

```
350735.47 234388.90
```

最初の数は、システムが起動している合計秒数です。2 番目の数は、マシンがアイドル状態になった時間（秒単位）です。

5.2.31. /proc/version

このファイルは、使用中の Linux カーネルと gcc のバージョンと、システムにインストールされている Red Hat Enterprise Linux のバージョンを指定します。

```
Linux version 2.6.8-1.523 (user@foo.redhat.com) (gcc version 3.4.1 20040714 \ (Red Hat
Enterprise Linux 3.4.1-7)) #1 Mon Aug 16 13:27:03 EDT 2004
```

この情報は、ユーザーのログイン時に表示されるバージョンデータなど、さまざまな目的で使用されます。

5.3. /PROC/内のディレクトリー

カーネルに関する一般的な情報は、/proc/ ディレクトリー内のディレクトリーおよびサブディレクトリーにグループ化されます。

5.3.1. プロセスディレクトリー

すべての /proc/ ディレクトリーには、数値名を持つディレクトリーが多数含まれます。これらのリストは、以下のようになります。

```
dr-xr-xr-x  3 root     root           0 Feb 13 01:28 1
dr-xr-xr-x  3 root     root           0 Feb 13 01:28 1010
dr-xr-xr-x  3 xfs      xfs            0 Feb 13 01:28 1087
dr-xr-xr-x  3 daemon   daemon         0 Feb 13 01:28 1123
dr-xr-xr-x  3 root     root           0 Feb 13 01:28 11307
dr-xr-xr-x  3 apache   apache         0 Feb 13 01:28 13660
dr-xr-xr-x  3 rpc      rpc            0 Feb 13 01:28 637
dr-xr-xr-x  3 rpcuser  rpcuser        0 Feb 13 01:28 666
```

これらのディレクトリーは、プログラム *のプロセス* ID の後に名前が付けられ、そのプロセスに固有の情報が含まれるため、プロセスディレクトリー と呼ばれます。各プロセスディレクトリーの所有者およびグループは、プロセスを実行しているユーザーに設定されます。プロセスが終了すると、/proc/ プロセスディレクトリーが消失します。

各プロセスディレクトリーには以下のファイルが含まれます。

- cmdline - プロセスの起動時に発行されたコマンドが含まれます。
- CWD: プロセスの現在の作業ディレクトリーへのシンボリックリンク。
- environ: プロセスの環境変数の一覧。環境変数はすべての大文字で指定され、値は小文字です。
- exe: このプロセスの実行ファイルへのシンボリックリンクです。
- fd: 特定のプロセスのファイル記述子をすべて含むディレクトリー。番号付きのリンクには、以下のものがあります。

  ```
  total 0
  lrwx------  1 root  root  64 May  8 11:31 0 -> /dev/null
  lrwx------  1 root  root  64 May  8 11:31 1 -> /dev/null
  lrwx------  1 root  root  64 May  8 11:31 2 -> /dev/null
  lrwx------  1 root  root  64 May  8 11:31 3 -> /dev/ptmx
  lrwx------  1 root  root  64 May  8 11:31 4 -> socket:[7774817]
  lrwx------  1 root  root  64 May  8 11:31 5 -> /dev/ptmx
  lrwx------  1 root  root  64 May  8 11:31 6 -> socket:[7774829]
  lrwx------  1 root  root  64 May  8 11:31 7 -> /dev/ptmx
  ```

- maps: このプロセスに関連するさまざまな実行ファイルおよびライブラリーファイルにマップするメモリーの一覧。プロセスの複雑さによっては、このファイルが長くなる可能性がありますが、sshd プロセスからの出力例は以下のように始まります。

  ```
  08048000-08086000 r-xp 00000000 03:03 391479     /usr/sbin/sshd
  08086000-08088000 rw-p 0003e000 03:03 391479 /usr/sbin/sshd
  08088000-08095000 rwxp 00000000 00:00 0
  40000000-40013000 r-xp 0000000 03:03 293205 /lib/ld-2.2.5.so
  40013000-40014000 rw-p 00013000 03:03 293205 /lib/ld-2.2.5.so
  40031000-40038000 r-xp 00000000 03:03 293282 /lib/libpam.so.0.75
  40038000-40039000 rw-p 00006000 03:03 293282 /lib/libpam.so.0.75
  ```

```
40039000-4003a000 rw-p 00000000 00:00 0
4003a000-4003c000 r-xp 00000000 03:03 293218 /lib/libdl-2.2.5.so
4003c000-4003d000 rw-p 00001000 03:03 293218 /lib/libdl-2.2.5.so
```

- mem: プロセスによって保持されるメモリー。このファイルはユーザーが読み取ることはできません。
- root: プロセスのルートディレクトリーへのリンク。
- stat - プロセスのステータス
- statm - プロセスによって使用されているメモリーのステータス以下は、/proc/statm ファイルの例です。

```
263 210 210 5 0 205 0
```

7 列は、プロセスの異なるメモリー統計に関連します。左から右に、使用されているメモリーの次の側面を報告します。

1. プログラムの合計サイズ（キロバイト単位）。
2. メモリー部分のサイズ（キロバイト単位）。
3. 共有されるページ数。
4. コードであるページ数。
5. データ/スタックページ数。
6. ライブラリーページ数。
7. ダーティーページの数。

- status - stat または statm よりも読み取り可能な形式のプロセスのステータス。sshd の出力例を以下に示します。

```
Name: sshd
State: S (sleeping)
Tgid: 797
Pid: 797
PPid: 1
TracerPid: 0
Uid: 0 0 0 0
Gid: 0 0 0 0
FDSize: 32
Groups:
VmSize:     3072 kB
VmLck:         0 kB
VmRSS:      840 kB
VmData:     104 kB
VmStk:       12 kB
VmExe:      300 kB
VmLib:     2528 kB
SigPnd: 0000000000000000
SigBlk: 0000000000000000
SigIgn: 8000000000001000
```

```
SigCgt: 0000000000014005
CapInh: 0000000000000000
CapPrm: 00000000ffffffeff
CapEff: 00000000ffffffeff
```

この出力の情報には、プロセス名と ID、状態(S (sleeping)、R （実行中)、プロセスを実行しているユーザー/グループ ID、メモリー使用量に関する詳細なデータ）が含まれます。

5.3.1.1. /proc/self/

/proc/self/ ディレクトリーは、現在実行中のプロセスへのリンクです。これにより、プロセスはプロセス ID を把握せずに自身を確認することができます。

シェル環境内で、/proc/self/ ディレクトリーを一覧表示すると、そのプロセスのプロセスディレクトリーを一覧表示するのと同じコンテンツが生成されます。

5.3.2. /proc/bus/

このディレクトリーには、システムで利用可能なさまざまなバスに固有の情報が含まれています。たとえば、PCI バスと USB バスを含む標準システムでは、これらの各バスの現在のデータは、/proc/bus/pci/ などの同じ名前の /proc/bus/ サブディレクトリー内で利用できます。

/proc/bus/ 内で使用できるサブディレクトリーとファイルは、システムに接続されているデバイスによって異なります。ただし、各バスタイプには少なくとも 1つのディレクトリーがあります。これらのバスディレクトリー内には通常、バイナリーファイルを含む 001 などの数値名の少なくとも 1つのサブディレクトリーがあります。

たとえば、/proc/bus/usb/ サブディレクトリーには、USB バス上のさまざまなデバイスと、そのデバイスに必要なドライバーを追跡するファイルが含まれます。以下は、/proc/bus/usb/ ディレクトリーの一覧表示例です。

```
total 0 dr-xr-xr-x    1 root     root           0 May  3 16:25 001
-r--r--r--    1 root     root           0 May  3 16:25 devices
-r--r--r--    1 root     root           0 May  3 16:25 drivers
```

/proc/bus/usb/001/ ディレクトリーには、最初の USB バス上のすべてのデバイスが含まれ、デバイスファイルはマザーボード上の USB ルートハブを識別します。

以下は、/proc/bus/usb/devices ファイルの例です。

```
T:  Bus=01 Lev=00 Prnt=00 Port=00 Cnt=00 Dev#=  1 Spd=12  MxCh= 2
B:  Alloc=  0/900 us ( 0%), #Int=  0, #Iso=  0
D:  Ver= 1.00 Cls=09(hub  ) Sub=00 Prot=00 MxPS= 8 #Cfgs=  1
P:  Vendor=0000 ProdID=0000 Rev= 0.00
S:  Product=USB UHCI Root Hub
S:  SerialNumber=d400
C:* #Ifs= 1 Cfg#= 1 Atr=40 MxPwr=  0mA
I:  If#= 0 Alt= 0 #EPs= 1 Cls=09(hub  ) Sub=00 Prot=00 Driver=hub
E:  Ad=81(I) Atr=03(Int.) MxPS=   8 Ivl=255ms
```

5.3.3. /proc/driver/

このディレクトリーには、カーネルが使用する特定のドライバーに関する情報が含まれます。

ここで検出される一般的なファイルは rtc で、システムの *Real Time Clock (RTC)* のドライバーからの出力を提供します。これは、システムがオフになった間に時間を維持するデバイスです。/proc/driver/rtc からの出力例を以下に示します。

```
rtc_time        : 16:21:00
rtc_date        : 2004-08-31
rtc_epoch       : 1900
alarm           : 21:16:27
DST_enable      : no
BCD             : yes
24hr            : yes
square_wave     : no
alarm_IRQ       : no
update_IRQ      : no
periodic_IRQ    : no
periodic_freq   : 1024
batt_status     : okay
```

RTC の詳細は、以下のインストール済みドキュメントを参照してください。

/usr/share/doc/kernel-doc-*<version>*/Documentation/rtc.txt.

5.3.4. /proc/fs

このディレクトリーには、エクスポートされるファイルシステムが表示されます。NFS サーバーを実行している場合は、cat /proc/fs/nfsd/exports を実行すると、共有しているファイルシステムと、そのファイルシステムに付与されたパーミッションが表示されます。NFS を使用したファイルシステム共有の詳細は、21章*Network File System (NFS)*を参照してください。

5.3.5. /proc/ide/

このディレクトリーには、システム上の IDE デバイスに関する情報が含まれます。各 IDE チャネルは、/proc/ide/ide0 や /proc/ide/ide1 などの個別のディレクトリーとして表されます。さらに、ドライバー ファイルを利用できます。これにより、IDE チャネルで使用されるさまざまなドライバーのバージョン番号が提供されます。

```
ide-floppy version 0.99.
newide ide-cdrom version 4.61
ide-disk version 1.18
```

多くのチップセットは、このディレクトリーに ファイルも提供します。このファイルには、チャネルを介して接続されたドライブに関する追加データが含まれます。たとえば、一般的な Intel PIIX4 Ultra 33 チップセットは、/proc/ide/piix ファイルを生成し、IDE チャネルのデバイスに対して DMA または UDMA が有効になっているかどうかを示します。

```
Intel PIIX4 Ultra 33 Chipset.
------------ Primary Channel ---------------- Secondary Channel -------------
  enabled                     enabled

------------ drive0 --------- drive1 -------- drive0 ---------- drive1 ------
DMA enabled:    yes             no             yes              no
UDMA enabled:   yes              no             no               no
UDMA enabled:   2               X              X                X
UDMA DMA PIO
```

ide0 などの IDE チャネルのディレクトリーに移動すると、追加情報が提供されます。チャネルファイルはチャネル 番号を提供し、モデル はチャネルのバスタイプを特定します(pciなど)。

5.3.5.1. デバイスディレクトリー

各 IDE チャネルディレクトリー内にはデバイスディレクトリーがあります。デバイスディレクトリーの名前は、/dev/ ディレクトリーのドライブ文字に対応します。たとえば、ide0 の最初の IDE ドライブは hda になります。

注記

/proc/ide/ ディレクトリーには、これらの各デバイスディレクトリーへのシンボリックリンクがあります。

各デバイスディレクトリーには、情報および統計のコレクションが含まれます。これらのディレクトリーの内容は、接続されたデバイスの種類によって異なります。多くのデバイスに共通する便利なファイルには、以下のようなものがあります。

- cache - デバイスキャッシュ。
- 容量 - デバイスの容量(512 バイトブロック)。
- driver - デバイスの制御に使用するドライバーおよびバージョン
- ジオメトリー - デバイスの物理的および論理ジオメトリー。
- メディア -ディスク などのデバイスのタイプ。
- model - デバイスのモデル名または数。
- 設定 - 現在のデバイスパラメーターのコレクション。通常、このファイルには非常に便利な技術情報が含まれています。標準 IDE ハードディスクの 設定 ファイルのサンプルは以下のようになります。

```
name              value        min        max        mode
----              -----        ---        ---        ----
acoustic          0            0          254        rw
address           0            0          2          rw
bios_cyl          38752        0          65535      rw
bios_head         16           0          255        rw
bios_sect         63           0          63         rw
bswap             0            0          1          r
current_speed     68           0          70         rw
failures          0            0          65535      rw
init_speed        68           0          70         rw
io_32bit          0            0          3          rw
keepsettings      0            0          1          rw
lun               0            0          7          rw
max_failures      1            0          65535      rw
multcount         16           0          16         rw
nice1             1            0          1          rw
nowerr            0            0          1          rw
number            0            0          3          rw
pio_mode          write-only   0          255        w
```

```
unmaskirq       0       0       1       rw
using_dma       1       0       1       rw
wcache          1       0       1       rw
```

5.3.6. /proc/irq/

このディレクトリーは IRQ を CPU アフィニティーに設定するために使用されます。これにより、システムは特定の IRQ を 1つの CPU にのみ接続できます。または、CPU が IRQ の処理から除外できます。

各 IRQ には独自のディレクトリーがあり、各 IRQ の個別の設定を可能にします。/proc/irq/prof_cpu_mask ファイルは、IRQ ディレクトリー内のsmp_affinity ファイルのデフォルト値が含まれるビットマスクです。smp_affinity の値は、その特定の IRQ を処理する CPU を指定します。

/proc/irq/ ディレクトリーの詳細は、以下のインストール済みドキュメントを参照してください。

```
/usr/share/doc/kernel-doc-<version>/Documentation/filesystems/proc.txt
```

5.3.7. /proc/net/

このディレクトリーでは、さまざまなネットワークパラメーターと統計を包括的に確認します。このディレクトリー内の各ディレクトリーと仮想ファイルは、システムのネットワーク設定の要素を記述します。以下は、/proc/net/ ディレクトリーの部分的なリストです。

- arp: カーネルの ARP テーブルを一覧表示します。このファイルは、ハードウェアアドレスをシステムの IP アドレスに接続する際に特に便利です。
- ATM / ディレクトリー - このディレクトリー内のファイルには*Asynchronous Transfer Mode (ATM)* の設定と統計が含まれます。このディレクトリーは、主に ATM ネットワークおよび ADSL カードで使用されます。
- dev: システムに設定したさまざまなネットワークデバイスの一覧を表示し、統計の送受信を完了します。このファイルは、各インターフェイスが送受信したバイト数、インバウンドおよびアウトバウンドのパケット数、発生したエラーの数、ドロップされたパケット数などを表示します。
- dev_mcast - 各デバイスがリッスンしている Layer2 マルチキャストグループを一覧表示します。
- IGMP: このシステムが参加する IP マルチキャストアドレスを一覧表示します。
- ip_conntrack - IP 接続を転送するマシンの追跡されたネットワーク接続を一覧表示します。
- ip_tables_names - 使用中の iptables のタイプを一覧表示します。このファイルは、iptables がシステムでアクティブで、1つ以上の値(filter、mangle、または nat)が含まれる場合にのみ存在します。
- ip_mr_cache - マルチキャストルーティングキャッシュを一覧表示します。
- ip_mr_vif - マルチキャスト仮想インターフェイスを一覧表示します。
- netstat: TCP タイムアウト、SYN クッキーの送受信など、非常に詳細なネットワーク統計収集が含まれます。
- psched: グローバルパケットスケジューラーパラメーターを一覧表示します。

- **raw** - raw デバイスの統計を一覧表示します。
- **route** - カーネルのルーティングテーブルを一覧表示します。
- **rt_cache**: 現在のルーティングキャッシュが含まれます。
- **SNMP**: 使用中のさまざまなネットワークプロトコルの Simple Network Management Protocol (SNMP)データのリスト。
- **sockstat** - ソケット統計を提供します。
- TCP : 詳細な TCP ソケット情報が含まれます。
- **tr_rif** - トークンリング RIF ルーティングテーブルを一覧表示します。
- UDP : 詳細な UDP ソケット情報が含まれます。
- UNIX - 現在使用中の UNIX ドメインソケットを一覧表示します。
- ワイヤレス - ワイヤレス インターフェイスデータを一覧表示します。

5.3.8. /proc/scsi/

このディレクトリーは /proc/ide/ ディレクトリーに似ていますが、接続 SCSI デバイス用です。

このディレクトリーのプライマリーファイルは /proc/scsi/scsi で、認識されているすべての SCSI デバイスの一覧が含まれます。このリストから、デバイスのタイプ、モデル名、ベンダー、SCSI チャネル、および ID データが利用できます。

たとえば、システムに SCSI CD-ROM、テープドライブ、ハードドライブ、および RAID コントローラーが含まれている場合、このファイルは以下のようになります。

```
Attached devices:
Host: scsi1
Channel: 00
Id: 05
Lun: 00
Vendor: NEC
Model: CD-ROM DRIVE:466
Rev: 1.06
Type:   CD-ROM
ANSI SCSI revision: 02
Host: scsi1
Channel: 00
Id: 06
Lun: 00
Vendor: ARCHIVE
Model: Python 04106-XXX
Rev: 7350
Type:   Sequential-Access
ANSI SCSI revision: 02
Host: scsi2
Channel: 00
Id: 06
Lun: 00
Vendor: DELL
```

```
Model: 1x6 U2W SCSI BP
Rev: 5.35
Type:   Processor
ANSI SCSI revision: 02
Host: scsi2
Channel: 02
Id: 00
Lun: 00
Vendor: MegaRAID
Model: LD0 RAID5 34556R
Rev: 1.01
Type:   Direct-Access
ANSI SCSI revision: 02
```

システムが使用する各 SCSI ドライバーには、/proc/scsi/ 内に独自のディレクトリーがあります。これには、そのドライバーを使用する各 SCSI コントローラーに固有のファイルが含まれます。上記の例では、2 つのドライバーが使用されているため、aic7xxx/ ディレクトリーおよび megaraid/ ディレクトリーが存在します。各ディレクトリーのファイルには、通常、そのドライバーを使用する SCSI コントローラーの I/O アドレス範囲、IRQ 情報、および統計が含まれます。各コントローラーは、異なるタイプおよび量の情報を報告することができます。この例の Adaptec AIC-7880 Ultra SCSI ホストアダプターの ファイルにより、以下の出力が生成されます。

```
Adaptec AIC7xxx driver version: 5.1.20/3.2.4
Compile Options:
TCQ Enabled By Default : Disabled
AIC7XXX_PROC_STATS     : Enabled
AIC7XXX_RESET_DELAY    : 5
Adapter Configuration:
SCSI Adapter: Adaptec AIC-7880 Ultra SCSI host adapter
Ultra Narrow Controller     PCI MMAPed
I/O Base: 0xfcffe000
Adapter SEEPROM Config: SEEPROM found and used.
Adaptec SCSI BIOS: Enabled
IRQ: 30
SCBs: Active 0, Max Active 1, Allocated 15, HW 16, Page 255
Interrupts: 33726
BIOS Control Word: 0x18a6
Adapter Control Word: 0x1c5f
Extended Translation: Enabled
Disconnect Enable Flags: 0x00ff
Ultra Enable Flags: 0x0020
Tag Queue Enable Flags: 0x0000
Ordered Queue Tag Flags: 0x0000
Default Tag Queue Depth: 8
Tagged Queue By Device array for aic7xxx
host instance 1:       {255,255,255,255,255,255,255,255,255,255,255,255,255,255,255,255}
Actual queue depth per device for aic7xxx host instance 1:       {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1}
Statistics:

(scsi1:0:5:0) Device using Narrow/Sync transfers at 20.0 MByte/sec, offset 15
Transinfo settings: current(12/15/0/0), goal(12/15/0/0), user(12/15/0/0)
Total transfers 0 (0 reads and 0 writes)
  < 2K      2K+     4K+     8K+    16K+    32K+    64K+   128K+
Reads:       0       0       0       0       0       0       0       0
Writes:      0       0       0       0       0       0       0       0
```

```
(scsi1:0:6:0) Device using Narrow/Sync transfers at 10.0 MByte/sec, offset 15
Transinfo settings: current(25/15/0/0), goal(12/15/0/0), user(12/15/0/0)
Total transfers 132 (0 reads and 132 writes)
  < 2K     2K+     4K+     8K+    16K+    32K+    64K+   128K+
Reads:       0       0       0       0       0       0       0       0
Writes:      0       0       0       1     131       0       0       0
```

この出力では、チャネル ID に基づいてコントローラーに接続された SCSI デバイスへの転送速度と、そのデバイスによって読み取りまたは書き込まれたファイルの量とサイズに関する詳細な統計が表示されます。たとえば、このコントローラーは CD-ROM と 1秒あたり 20 メガバイトで通信していますが、テープドライブは 1秒あたり 10 メガバイトでのみ通信されます。

5.3.9. /proc/sys/

/proc/sys/ ディレクトリーは、/proc/ の他のディレクトリーとは異なります。これは、システムに関する情報を提供するだけでなく、システム管理者がカーネル機能をすぐに有効および無効にできるためです。

注意

/proc/sys/ ディレクトリーのさまざまなファイルを使用して実稼働システムの設定を変更する場合には注意が必要です。誤った設定を変更すると、カーネルが不安定になり、システムの再起動が必要になる場合があります。

このため、/proc/sys/ の値を変更する前に、そのファイルに対してオプションが有効であることを確認してください。

特定のファイルの設定が可能かどうか、または情報を提供するように設計されているかどうかを判断する適切な方法は、シェルプロンプトで -l オプションを使用して一覧表示することです。ファイルが書き込み可能である場合、これを使用してカーネルを設定できます。たとえば、/proc/sys/fs の部分的なリストは以下のようになります。

```
-r--r--r--  1 root  root  0 May 10 16:14 dentry-state
-rw-r--r--  1 root  root  0 May 10 16:14 dir-notify-enable
-r--r--r--  1 root  root  0 May 10 16:14 dquot-nr
-rw-r--r--  1 root  root  0 May 10 16:14 file-max
-r--r--r--  1 root  root  0 May 10 16:14 file-nr
```

このリストでは、dir-notify-enable ファイルおよび file-max ファイルを に書き込めるため、カーネルを設定するために使用できます。その他のファイルは、現在の設定に関するフィードバックのみを提供します。

/proc/sys/ ファイル内の値を変更するには、新しい値をファイルに echo します。たとえば、実行中のカーネルで System Request Key を有効にするには、以下のコマンドを入力します。

```
echo 1 > /proc/sys/kernel/sysrq
```

これにより、sysrq の値が 0 (off)から 1 (on)に変わります。

いくつかの /proc/sys/ 設定ファイルには、複数の値が含まれています。新しい値を正しく送信するには、echo コマンドで渡される各値の間に空白文字を配置します。以下に例を示します。

```
echo 4 2 45 > /proc/sys/kernel/acct
```

注記

システムを再起動すると、echo コマンドを使用して設定変更が消えます。システムの再起動後に設定変更を有効にするには、「sysctl コマンドの使用」 を参照してください。

/proc/sys/ ディレクトリーには、実行中のカーネルのさまざまな側面を制御するサブディレクトリーが複数含まれています。

5.3.9.1. /proc/sys/dev/

このディレクトリーは、システム上の特定のデバイスのパラメーターを提供します。ほとんどのシステムには、cdrom/ と raid/ の少なくとも 2 つのディレクトリーがあります。カスタマイズされたカーネルには、parport/ などの他のディレクトリーを設定できます。これにより、複数のデバイスドライバー間で 1 つの並列ポートを共有できます。

cdrom/ ディレクトリーには、いくつかの重要な CD-ROM パラメーターを示す info というファイルが含まれます。

```
CD-ROM information, Id: cdrom.c 3.20 2003/12/17
drive name:             hdc
drive speed:            48
drive # of slots:       1
Can close tray:         1
Can open tray:          1
Can lock tray:          1
Can change speed:       1
Can select disk:        0
Can read multisession:  1
Can read MCN:           1
Reports media changed:  1
Can play audio:         1
Can write CD-R:         0
Can write CD-RW:        0
Can read DVD:           0
Can write DVD-R:        0
Can write DVD-RAM:      0
Can read MRW:           0
Can write MRW:          0
Can write RAM:          0
```

このファイルを迅速にスキャンして、不明な CD-ROM の特性を検出できます。複数の CD-ROM がシステムで利用可能な場合、各デバイスには独自の情報列が指定されます。

/proc/sys/dev/cdrom 内のさまざまなファイル(autoclose や checkmedia など)を使用して、システムの CD-ROM を制御できます。これらの機能を有効または無効にするには、echo コマンドを使用します。

RAID サポートがカーネルにコンパイルされると、/proc/sys/dev/raid/ ディレクトリーが少なくとも 2 つのファイル(speed_limit_min および speed_limit_max)で利用できます。この設定により、ディスクの再同期など、I/O 集約タスク用の RAID デバイスのアクセラレーションを決定します。

5.3.9.2. /proc/sys/fs/

このディレクトリーには、クォータ、ファイルハンドル、inode、dentry 情報など、ファイルシステムのさまざまな側面に関するオプションの配列と情報が含まれています。

binfmt_misc/ ディレクトリーは、その他のバイナリー形式のカーネルサポートを提供するために使用されます。

/proc/sys/fs/ の重要なファイルには、以下が含まれます。

- dentry-state: ディレクトリーキャッシュのステータスを指定します。ファイルは以下のようになります。

  ```
  57411 52939 45 0 0 0
  ```

  最初の番号は、ディレクトリーキャッシュエントリーの合計数を示し、2 番目の番号には未使用のエントリーの数が表示されます。3 番目の番号は、ディレクトリーが解放された場合と回収可能な時点までの秒数を示し、システムによって現在要求されたページを 4 番目に測定します。最後の 2 つの数字は使用されず、ゼロのみを表示します。

- D quot-nr: キャッシュされたディスククォータエントリーの最大数を一覧表示します。

- file-max: カーネルが割り当てるファイルハンドルの最大数を一覧表示します。このファイルの値を増やすと、利用可能なファイルハンドルがないためにエラーを解決できます。

- file-nr: 割り当てファイルハンドルの数、使用済みファイルハンドル、およびファイルハンドルの最大数を一覧表示します。

- overflowgid および overflowuid: 16 ビットグループとユーザー ID のみをサポートするファイルシステムで使用するために、それぞれ固定グループ ID とユーザー ID を定義します。

- super-max: 利用可能なスーパーブロックの最大数を制御します。

- super-nr: 現在使用中のスーパーブロックの数を表示します。

5.3.9.3. /proc/sys/kernel/

このディレクトリーには、カーネルの操作に直接影響するさまざまな設定ファイルが含まれています。最も重要なファイルには以下が含まれます。

- acct - ログを含むファイルシステムで利用可能な空き領域の割合に基づいて、プロセスアカウンティングの一時停止を制御します。デフォルトでは、ファイルは以下のようになります。

  ```
  4 2 30
  ```

  最初の値は、ロギングの再開に必要な空き容量の割合を決定し、2 番目の値はロギングが一時停止されたときの空き領域のしきい値パーセンテージを設定します。3 番目の値は、カーネルがファイルシステムをポーリングして、ロギングを一時停止または再開する必要があるかどうかを確認する間隔を秒単位で設定します。

- cap-bound: システム上のプロセス *の機能* 一覧を提供する機能バウンディング設定を制御します。機能がここに一覧表示されていない場合は、特権の種類に関係なく、プロセスが実行でき

ません。これは、ブートプロセスの特定地点を超えた特定の事態を確実に行わないようにすることで、システムをよりセキュアにすることです。

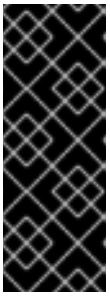
この仮想ファイルの有効な値の一覧については、以下のインストール済みドキュメントを参照してください。

/lib/modules/*<kernel-version>*/build/include/linux/capability.h.

- Ctrl-alt-del - Ctrl+Alt+Delete で init (0)を使用してコンピューターを正常に再起動するか、ダーティーバッファーをディスクと同期せずに直ちに再起動を行うか(1)を制御します。
- domainName: example.com などのシステムドメイン名を設定します。
- exec-shield - カーネルの Exec Shield 機能を設定します。exec Shield は、特定タイプのバッファーオーバーフロー攻撃に対する保護を提供します。

  この仮想ファイルには、以下の 2 つの値を使用できます。

  - 0 - Exec Shield を無効にします。
  - 1 - Exec Shield を有効にします。これはデフォルト値です。



重要な影響

Exec Shield が無効になっている間に起動したセキュリティーの影響を受けるアプリケーションを実行している場合は、Exec Shield を有効にするために Exec Shield が有効な場合にこれらのアプリケーションを再起動する必要があります。

- exec-shield-randomize - メモリー内のさまざまな項目の場所のランダム化を有効にします。これにより、潜在的な攻撃者がメモリー内のプログラムやデーモンを見つけるのを防ぐことができます。プログラムまたはデーモンが起動するたびに、静的メモリーまたは絶対メモリーアドレスで決して、毎回異なるメモリーの場所に配置されます。

  この仮想ファイルには、以下の 2 つの値を使用できます。

  - 0 - Exec Shield のランダム化を無効にします。これは、アプリケーションのデバッグに役立ちます。

- 1 - Exec Shield のランダム化を有効にします。これはデフォルト値です。注記：exec-shield - randomize を有効にするには、exec-shield ファイルも 1 に設定する必要があります。

- hostname - www.example.com などのシステムホスト名を設定します。

- ホットプラグ: システムによって設定の変更が検出されたときに使用されるユーティリティーを設定します。これは主に USB および Cardbus PCI で使用されます。このロールを実行するために新しいプログラムをテストしない限り、デフォルト値の /sbin/hotplug は変更しないでください。

- modprobe - カーネルモジュールの読み込みに使用するプログラムの場所を設定します。デフォルト値は /sbin/modprobe で、カーネルスレッドが kmod を呼び出すときにモジュールをロードするために kmod を呼び出すことを意味します。

- msgmax: あるプロセスから別のプロセスに送信されるメッセージの最大サイズを設定し、デフォルトでは 8192 バイトに設定されます。プロセス間のキューに置かれたメッセージはスワップ不可能なカーネルメモリーに保存されるため、この値を引き上げる際には注意してください。msgmax を増やすと、システムの RAM 要件が増えます。

- msgmnb - 1 つのメッセージキューに最大バイト数を設定します。デフォルトは 16384 です。

- MSGMNI: メッセージキュー識別子の最大数を設定します。デフォルトは 16 です。

- osrelease: Linux カーネルのリリース番号を一覧表示します。このファイルは、カーネルソースを変更して再コンパイルするだけで変更できます。

- OSType: オペレーティングシステムの種類を表示します。デフォルトでは、このファイルは Linux に設定されています。この値は、カーネルソースを変更して再コンパイルするだけで変更できます。

- overflowgid および overflowuid: 16 ビットグループとユーザー ID のみをサポートするアーキテクチャーのシステムコールで使用するために、それぞれ固定グループ ID とユーザー ID を定義します。

-

panic - システムがカーネルパニックが発生したときにカーネルが再起動を延期する秒数を定義します。デフォルトでは、値は 0 に設定されています。これにより、パニック後の自動再起動が無効になります。

- **printk: このファイルは、エラーメッセージの出力またはロギングに関連するさまざまな設定を制御します。カーネルによって報告される各エラーメッセージには、*メッセージ*の重要性を定義するログレベルが関連付けられています。ログレベルの値は、以下の順序で分類されます。**
  - **0 - カーネル緊急。システムが利用できません。**
  - **1 - カーネルアラート。すぐに対処する必要があります。**
  - **2 - 重大な問題があると見なされるカーネルの状態。**
  - **3 - 一般的なカーネルエラー状態。**
  - **4 - 一般的なカーネルの警告状態。**
  - **5 - 正常だが重大な状態のカーネル通知。**
  - **6: カーネル情報メッセージ。**
  - **7 - カーネルのデバッグレベルのメッセージ。**

printk ファイルには 4 つの値があります。

```
6   4   1   7
```

これらの値はそれぞれ、エラーメッセージを処理するための異なるルールを定義します。コンソールログレベル と呼ばれる最初の値は 、*コンソール* に出力されるメッセージの最も低い優先度を定義します。（優先順位が低いほど、ログレベル番号が高いことに注意してください。） 2 番目の値は、明示的なログレベルが付いていないメッセージのデフォルトログレベル

を設定します。3 番目の値は、コンソールログレベルに可能な限り低いログレベル設定を設定します。最後の値は、コンソールログレベルのデフォルト値を設定します。

- random/ ディレクトリー：カーネルの乱数の生成に関連する多数の値を一覧表示します。
- rtsig-max - いつでもシステムがキューに置かれた可能性のある POSIX リアルタイムシグナルの最大数を設定します。デフォルト値は 1024 です。
- rtsig-nr - カーネルによってキューに入れられた POSIX リアルタイムシグナルの現在の数を一覧表示します。
- sem - カーネル内でセ マフォ を設定します。セマフォとは、特定のプロセスの使用状況を制御するために使用される System V IPC オブジェクトです。
- shmall: システム全体で使用できる共有メモリーページの合計量を設定します。デフォルトでは、この値は 2097152 です。
- shmmax - カーネルで許可される最大共有メモリーセグメントサイズを設定します。デフォルトでは、この値は 33554432 です。ただし、カーネルは、これよりもはるかに大きな値をサポートします。
- SHMMN I: システム全体で共有メモリーセグメントの最大数を設定します。デフォルトでは、この値は 4096 です。
- ubuntu: この値がゼロ(0)以外の値に設定されている場合は、システム要求キーを有効にします。

System Request Key を使用すると、単純なキーの組み合わせでカーネルに即時入力できます。たとえば、System Request Key を使用すると、システムを直ちにシャットダウンまたは再起動したり、マウントされたすべてのファイルシステムを同期したり、コンソールに重要な情報をダンプしたりできます。System Request Key を開始するには、Alt+ *system request code> と入力します*。&lt *;system request code>* を、以下のシステム要求コードのいずれかに置き換えます。

- r - キーボードの raw モードを無効にし、これを XLATE に設定します（すべてのキーで Alt、Ctrl、または Shift など、Alt、Ctrl、Shift など）は認識しません。

- k - 仮想コンソールでアクティブなすべてのプロセスを強制終了します。*SAK*(*Secure Access Key*)とも呼ばれ、ユーザー名とパスワードを取得するように設計された Trojan コピーではなく、init からログインプロンプトが生成されることを確認するために使用されます。

- b - 最初にファイルシステムのマウントを解除したり、システムに接続されているディスクを同期したりせずにカーネルを再起動します。

- c - 最初にファイルシステムのマウントを解除したり、システムに接続されているディスクを同期したりせずにシステムがクラッシュします。

- o - システムをシャットオフします。

- s: システムに接続されているディスクの同期を試みます。

- u - すべてのファイルシステムのアンマウントと再マウントを読み取り専用として試行します。

- p: すべてのフラグを出力し、コンソールに登録します。

- T: コンソールにプロセスの一覧を出力します。

- m: コンソールにメモリー統計を出力します。

- 0 から 9: コンソールのログレベルを設定します。

- e: SIGTERM を使用して init 以外のすべてのプロセスを強制終了します。

- i - SIGKILL を使用して init 以外のすべてのプロセスを強制終了します。

- l - SIGKILL (initを含む)を使用してすべてのプロセスを強制終了します。この System Request Key コードを発行すると、システムは使用できなくなります。

- H: ヘルプテキストを表示します。

この機能は、開発カーネルを使用する場合や、システムのフリーズが発生した場合に最も有益です。

注意

無人コンソールは攻撃者にシステムにアクセスできるため、System Request Key 機能はセキュリティーリスクとみなされます。このため、デフォルトでは無効になっています。

システム要求キーの詳細は、/usr/share/doc/kernel-doc- <*version*> /Documentation/sysrq.txt を参照してください。

- ubuntu-key: System Request Key のキーコードを定義します(84 がデフォルトです)。
- ubuntu-sticky: システム要求キーが予約されたキーの組み合わせかどうかを定義します。許可される値は以下のとおりです。
  - 0: Alt+とシステム 要求コードを同時に押す必要があります。これはデフォルト値です。
  - 1 - Alt+0 -9] は同時に押す必要がありますが、/proc/sys/kernel/sysrq-timer elapses で指定する秒数の前に、システム要求コードはいつでも押します。
- ubuntu-timer - システムリクエストコードをスキップするまでに許可される秒数を指定します。デフォルト値は 10 です。
- tainted: GPL 以外のモジュールが読み込まれているかどうかを示します。

- 0 - GPL 以外のモジュールがロードされません。
- 1 - GPL ライセンスのない少なくとも 1 つのモジュール（ライセンスのないモジュールを含む）が読み込まれます。
- 2 - 少なくとも 1 つのモジュールが insmod -f コマンドで強制的に読み込まれました。

- threads-max: デフォルト値の 2048 で、カーネルが使用するスレッドの最大数を設定します。

- version - カーネルが最後にコンパイルされた日時を表示します。#3 など、このファイルの最初のフィールドは、カーネルがソースベースからビルドされた回数に関連しています。

5.3.9.4. /proc/sys/net/

このディレクトリーには、さまざまなネットワークトピックに関するサブディレクトリーが含まれます。カーネルのコンパイル時のさまざまな設定では、イーサネット/、ipv 4/、ipx/ 、ipv 6/ などの異なるディレクトリーがここで利用可能になります。これらのディレクトリー内のファイルを変更することで、システム管理者は実行中のシステムでネットワーク設定を調整できます。

Linux で利用可能なさまざまなネットワークオプションについては、最も一般的な /proc/sys/net/ ディレクトリーのみについて説明します。

/proc/sys/net/core/ ディレクトリーには、カーネルとネットワーク層間の相互作用を制御するさまざまな設定が含まれています。これらのファイルの最も重要なものは以下のとおりです。

- message_burst: 新しい警告メッセージを書き込むために必要な 10 秒の時間を設定します。この設定は、サービス *拒否(DoS*)攻撃を軽減するために使用されます。デフォルト設定は 50 です。

- message_cost - すべての警告メッセージにコストを設定します。このファイルの値が高い（デフォルトは 5）、警告メッセージが無視される可能性が高くなります。この設定は、DoS 攻撃を軽減するために使用されます。

DoS 攻撃の概念は、ターゲットシステムにエラーを生成し、ログファイルでディスクパーティションを埋めるか、またはエラーログを処理するためにシステムのリソースをすべて要求する要求で調整することです。message_burst および message_cost の設定は、システムの許容リスクと包括的なロギングの必要性に基づいて変更されるように設計されています。

- netdev_max_backlog - 特定のインターフェイスがパケットを処理できるよりも早く受信した場合にキューに入れることができるパケットの最大数を設定します。このファイルのデフォルト値は 300 です。
- optmem_max: ソケットごとに許可される最大補助バッファーサイズを設定します。
- rmem_default: 受信ソケットバッファーのデフォルトサイズをバイト単位で設定します。
- rmem_max: 受信ソケットバッファーの最大サイズをバイト単位で設定します。
- wmem_default: 送信ソケットバッファーのデフォルトサイズをバイト単位で設定します。
- wmem_max: 送信ソケットバッファーサイズをバイト単位で設定します。

/proc/sys/net/ipv4/ ディレクトリーには、追加のネットワーク設定が含まれます。この設定の多くは、システムに対する攻撃を防止したり、システムをルーターとして機能させるために使用する場合に便利です。

注意

これらのファイルが誤って変更すると、システムへのリモート接続に影響する可能性があります。

以下は、/proc/sys/net/ipv4/ ディレクトリー内の重要なファイルの一部の一覧です。

- icmp_destunreach_rate、icmp_echoreply_rate、icmp_paramprob_rate、および

icmp_timeexeed_rate - 特定の条件下でホストに最大 ICMP 送信パケットレートを設定します。設定 0 は遅延を取り除くため、適切ではありません。

- **icmp_echo_ignore_all および icmp_echo_ignore_broadcasts - カーネルは、すべてのホストからの ICMP ECHO パケットを無視することや、ブロードキャストアドレスとマルチキャストアドレスから発信されたパケットのみを無視することを許可します。値が 0 の場合はカーネルが応答し、1 の値はパケットを無視します。**

- **ip_default_ttl - デフォルトの *Time To Live (TTL)* を設定します。これにより、宛先に到達する前にパケットが行うホップ数を制限します。この値を増やすと、システムパフォーマンスが低下する可能性があります。**

- **ip_forward - システムのインターフェイスを許可して、パケットを相互に転送できるようにします。デフォルトでは、このファイルは 0 に設定されています。このファイルを 1 に設定すると、ネットワークパケットの転送が可能になります。**

- **ip_local_port_range - ローカルポートが必要な場合に TCP または UDP が使用するポートの範囲を指定します。最初の番号は使用する一番小さいポートで、2 番目の番号は最高のポートを指定します。デフォルトの 1024 から 4999 よりも多くのポートを必要とすることが予想されるシステムでは、32768 から 61000 の範囲を使用する必要があります。**

- **tcp_syn_retries - 接続の試行時にシステムが SYN パケットを再送信する回数に制限を指定します。**

- **tcp_retries1 - 受信接続への応答を試行する許可される再送信の数を設定します。デフォルトは 3 です。**

- **tcp_retries2 - TCP パケットの許可される再送信数を設定します。デフォルトは 15 です。**

ファイル（という名前）

/usr/share/doc/kernel-doc-*<version>*/Documentation/networking/ ip-sysctl.txt

/proc/sys/net/ipv4/ ディレクトリーで利用可能なファイルおよびオプションの完全な一覧が含まれます。

/proc/sys/net/ipv4/ ディレクトリー内に他の多数のディレクトリーが存在し、各ディレクトリーはネットワークスタックのさまざまな側面に対応します。/proc/sys/net/ipv4/conf/ ディレクトリーを使用すると、未設定のデバイス(/proc/sys/net/ipv4/conf/default / サブディレクトリー内)のデフォルト設定や、すべての特別な設定(/proc/sys/net/ipv4/conf / all/ サブディレクトリー)を上書きする設定など、各システムインターフェイスをさまざまな方法で設定できます。

/proc/sys/net/ipv4/neigh/ ディレクトリーには、システムに直接接続されたホスト（ネットワーク近接）と通信するための設定が含まれ、複数のホップが離れるシステム用の異なる設定も含まれます。

IPV4 でのルーティングには、独自のディレクトリー /proc/sys/net/ipv4/route/ もあります。conf/ と neigh/ とは異なり、/ proc/sys/net/ipv4/route/ ディレクトリーには、システム上の任意のインターフェイスとのルーティングに適用される仕様が含まれます。max_size、max_delay、min_delay などのこれらの設定の多くは、ルーティングキャッシュのサイズの制御に関連します。ルーティングキャッシュを削除するには、フラッシュ ファイルに値を書き込みます。

これらのディレクトリーと設定ファイルの使用可能な値に関する追加情報は、以下を参照してください。

> /usr/share/doc/kernel-doc-*<version>*/Documentation/filesystems/proc.txt

5.3.9.5. /proc/sys/vm/

このディレクトリーは、Linux カーネルの仮想メモリー(VM)サブシステムの設定を容易にします。カーネルは、一般的に swap 領域と呼ばれる仮想メモリーの多重でインテリジェントな使用を行います。

以下のファイルは、一般的に /proc/sys/vm/ ディレクトリーにあります。

- block_dump: ブロック I/O デバッグが有効な場合を設定します。ファイルに実行される読み取り/書き込みおよびブロックのダーティー操作はすべて、それに応じてログに記録されます。これは、ディスクスピンアップおよびラップトップのバッテリー消費のためにスピンダウンする場合に役立ちます。block_dump が有効な場合には、dmesg からすべての出力を取得できます。デフォルト値は 0 です。

ヒント

block_dump がカーネルのデバッグと同時に有効になっている場合は、block_dump によって生じる誤ったディスクアクティビティーが生成されるため、klogd デーモンを停止するのが prudent になります。

- dirty_background_ratio: pdflush デーモンを介して、合計メモリーのこのパーセンテージでダーティーデータのバックグラウンドライトバックを開始します。デフォルト値は 10 です。

- dirty_expire_centisecs: ダーティーインメモリーデータが書き込みの対象となるのに十分な古いかどうかを定義します。この間隔よりも長いダーティーインメモリーデータは、次に pdflush デーモンがウェイクアップしたときに書き込まれます。デフォルト値は 3000 で、1 秒の 100 番目の値で表されます。

- dirty_ratio: pdflush を使用して、ダーティーデータのジェネレーターの合計メモリーの割合で、ダーティーデータのライトバックを開始します。デフォルト値は 40 です。

- dirty_writeback_centisecs: pdflush デーモンのウェイクアップの間隔を定義します。これは、ダーティーインメモリーデータをディスクに定期的に書き込みます。デフォルト値は 500 で、1 秒の 100 番目の値で表されます。

- laptop_mode: ディスクを可能な限り停止したままにすることで、ハードディスクをスピンアップする回数を最小限に抑えるため、ノートパソコンのバッテリー電源を節約します。これにより、将来のすべての I/O プロセスを組み合わせてスピンアップの頻度が削減され、効率が向上します。デフォルト値は 0 ですが、ラップトップでバッテリーが使用される場合には自動的に有効になります。

  この値は、ユーザーにバッテリー電源が有効になると、acpid デーモンによって自動的に制御されます。ラップトップが ACPI (Advanced Configuration and Power Interface)仕様をサポートする場合は、ユーザーの変更や対話は必要ありません。

  詳細は、以下のインストール済みドキュメントを参照してください。

  /usr/share/doc/kernel-doc-*<version>*/Documentation/laptop-mode.txt

- lower_zone_protection: カーネルがメモリー割り当ての少ないゾーンを定める方法を決定します。これは、highmem メモリー領域が有効なマシンで使用される場合に有効です。デフォルト値は 0 で、保護はまったくありません。他のすべての整数値はメガバイト単位であるため、低メモリーは ユーザーによって割り当てられないように保護されます。

  詳細は、以下のインストール済みドキュメントを参照してください。

/usr/share/doc/kernel-doc-*<version>*/Documentation/filesystems/proc.txt

- max_map_count: プロセスが持つことができるメモリーマップ領域の最大数を設定します。ほとんどの場合、65536 のデフォルト値が適切です。

- min_free_kbytes - Linux 仮想マシン（仮想メモリーマネージャー）が最小キロバイト数を解放するように強制します。仮想マシンは、この数を使用して、システム内の 低mem ゾーンごとに pages_min の値を計算します。デフォルト値は、マシンの合計メモリーに関するものです。

- nr_hugepages: カーネルで現在設定されている hugetlb ページの数を示します。

  詳細は、以下のインストール済みドキュメントを参照してください。

  /usr/share/doc/kernel-doc-*<version>*/Documentation/vm/hugetlbpage.txt

- nr_pdflush_threads: 現在実行している pdflush デーモンの数を示します。このファイルは読み取り専用であるため、ユーザーが変更しないでください。I/O 負荷が大きい場合、カーネルによりデフォルト値の 2 が増加します。

- overcommit_memory: 大規模なメモリー要求が許可または拒否される条件を設定します。以下の 3 つのモードを使用できます。

  - 0: カーネルは、利用可能なメモリー量と無効な要求の失敗要求の量を見積もることで、コミット処理でヒューリスティックメモリーを実行します。ただし、メモリーは正確なアルゴリズムではなくヒューリスティックを使用して割り当てられるため、この設定は、システムで利用可能なメモリーをオーバーロードできることがあります。これはデフォルト設定です。

  - 1 - カーネルは、コミット処理でメモリーを実行しません。この設定では、メモリーのオーバーロードの可能性が高まりますが、メモリー集約型タスク（一部のサイエンティックソフトウェアによって実行されるタスクなど）のパフォーマンスになります。

  - 2 - すべての swap を追加するメモリーに対する要求に失敗し、/proc/sys/vm/overcommit_ratio で指定した物理 RAM のパーセント。この設定は、メ

モリーのオーバーコミットのリスクが低いユーザーにとって最適です。

注記

この設定は、物理メモリーよりも大きいスワップ領域があるシステムにのみ推奨されます。

- overcommit_ratio: /proc/sys/vm/overcommit_memory が 2 に設定されている場合に考慮される物理 RAM の割合を指定します。デフォルト値は 50 です。

- page-cluster: 1 回の試行で読み取られるページ数を設定します。実際には 16 ページに関連するデフォルト値の 3 は、ほとんどのシステムに適しています。

- swappiness: マシンをスワップする容量を決定します。値が高いほど、スワップがより多くなります。パーセンテージとしてデフォルト値が 60 に設定されています。

カーネルベースのドキュメントはすべて、ローカルにインストールされている場所にあります。

/usr/share/doc/kernel-doc- <*version*> /Documentation/ を追加情報が含まれています。

5.3.10. /proc/sysvipc/

このディレクトリーには、System V IPC リソースに関する情報が含まれます。このディレクトリーのファイルは、メッセージ(msg)、セマフォ(sem)、共有メモリー(shm)に対する System V IPC 呼び出しに関連します。

5.3.11. /proc/tty/

このディレクトリーには、システムで利用可能な、かつ現在使用されている *tty デバイス* に関する情報が含まれます。当初は *teletype devices* と呼ばれ、文字ベースのデータ端末は tty デバイスと呼ばれます。

Linux には、3 種類の tty デバイスがあります。*シリアルデバイス* は、モデム経由やシリアルケーブルなど、シリアル接続で使用されます。*仮想端末* は、システムコンソールで Alt+F-key > を押す際に利用可能な仮想コンソールなど、一般的なコンソール接続を作成します。*擬似端末* は、XFree86 など

の上位レベルのアプリケーションで使用される双方向通信を作成します。drivers ファイルは、以下の例のように、使用中の現在の tty デバイスの一覧です。

```
serial               /dev/cua       5  64-127 serial:callout
serial               /dev/ttyS      4  64-127 serial
pty_slave            /dev/pts     136   0-255 pty:slave
pty_master           /dev/ptm     128   0-255 pty:master
pty_slave            /dev/ttyp      3   0-255 pty:slave
pty_master           /dev/pty       2   0-255 pty:master
/dev/vc/0            /dev/vc/0      4       0 system:vtmaster
/dev/ptmx            /dev/ptmx      5       2 system
/dev/console         /dev/console   5       1 system:console
/dev/tty             /dev/tty       5       0 system:/dev/tty
unknown              /dev/vc/%d     4    1-63 console
```

/proc/tty/driver/serial ファイルは、各シリアル tty 行の使用状況の統計とステータスを一覧表示します。

tty デバイスをネットワークデバイスとして使用するため、Linux カーネルはデバイス上で *ライン規則* を強制します。これにより、ドライバーは特定のタイプのヘッダーにデバイスを介して送信されるすべてのデータブロックを持つ特定タイプのヘッダーを配置することができ、データブロックのストリームにあるデータブロックへの接続のリモートエンドが可能になります。SLIP および PPP は一般的な行規則であり、それぞれはシリアルリンクを介してシステムを相互に接続するために使用されます。

登録した行規則は ldiscs ファイルに保存され、より詳細な情報は ldisc/ ディレクトリー内にあります。

5.3.12. /proc/<PID>/

OOM (Out of Memory) は、スワップ領域を含む利用可能なメモリーがすべて割り当てられているコンピューティング状態を指します。この状況が発生すると、システムがパニックになり、期待どおりに機能しなくなります。/proc/sys/vm/panic_on_oom には OOM の動作を制御するスイッチがあります。1 に設定すると、カーネルは OOM でパニックになります。0 の設定は、OOM で oom_killer という名前の関数を呼び出すようカーネルに指示します。通常、oom_killer は不正なプロセスを強制終了でき、システムは存続します。

これを変更する最も簡単な方法は、新しい値を /proc/sys/vm/panic_on_oom にエコーすることです。

```
~]# cat /proc/sys/vm/panic_on_oom
1
~]# echo 0 > /proc/sys/vm/panic_on_oom
~]# cat /proc/sys/vm/panic_on_oom
0
```

また、oom_killer スコアを調整することで、プロセスが強制終了される優先順位を設定することもできます。/proc/<PID>/ に は、oom_adj と oom_score の 2 つのツールがラベル付けされています。oom_adj の有効なスコアは、-16 から +15 の範囲にあります。現在の oom_killer スコアを表示するには、プロセスの oom_score を表示します。oom_killer はスコアが最も高いプロセスを最初に強制終了します。

この例では、PID が 12465 のプロセスの oom_score を調整して、oom_killer がこれを強制終了する可能性が低くなります。

```
~]# cat /proc/12465/oom_score
79872
~]# echo -5 > /proc/12465/oom_adj
~]# cat /proc/12465/oom_score
78
```

また、-17 には特殊な値があり、そのプロセスの oom_killer を無効にします。以下の例では、oom_score は 0 の値を返します。これは、このプロセスが強制終了されないことを示しています。

```
~]# cat /proc/12465/oom_score
78
~]# echo -17 > /proc/12465/oom_adj
~]# cat /proc/12465/oom_score
0
```

badness （） と呼ばれる関数は、各プロセスの実際のスコアを決定するために使用されます。これには、確認した各プロセスに 'points' を追加します。プロセススコアは以下の方法で行われます。

1. 各プロセスのスコアは、メモリーサイズになります。

2. （カーネルスレッドを含まない）プロセスの子のメモリーサイズもスコアに追加されます。

3. プロセススコアは 'niced' プロセスのスコアを増やし、長時間実行されるプロセスの場合に減少します。

4. CAP_SYS_ADMIN および CAP_SYS_RAWIO 機能を持つプロセスのスコアは減少します。

5.
最後のスコアは、oom_adj ファイルに保存されている値でビットシフトされます。

したがって、oom_score 値が最も高いプロセスは、おそらく特権のない、最近起動したプロセスであり、その子とともに大量のメモリーを使用し、'niced' があり、生の I/O を処理しません。

5.4. SYSCTL コマンドの使用

/sbin/sysctl コマンドは、/proc/sys/ ディレクトリーでカーネル設定を表示、設定、および自動化するために使用されます。

/proc/sys/ ディレクトリーで設定可能なすべての設定の概要については、root で /sbin/sysctl -a コマンドを入力します。これにより、以下のような小さな部分で大きな包括的なリストが作成されます。

```
net.ipv4.route.min_delay = 2 kernel.sysrq = 0 kernel.sem = 250     32000     32     128
```

これは、各ファイルを個別に表示した場合に表示される情報と同じです。唯一の違いはファイルの場所です。たとえば、/proc/sys/net/ipv4/route/min_delay ファイルは net.ipv4.route.min_delay として一覧表示され、ディレクトリーのスラッシュはドットに置き換えられ、proc.sys 部分が想定されます。

sysctl コマンドを echo の代わりに使用して、/proc/sys/ ディレクトリー内の書き込み可能なファイルに値を割り当てることができます。たとえば、コマンドを使用する代わりに、以下を実行します。

```
echo 1 > /proc/sys/kernel/sysrq
```

以下のように同等の sysctl コマンドを使用します。

```
~]# sysctl -w kernel.sysrq="1"
kernel.sysrq = 1
```

/proc/sys/ でこのような単一の値をすばやく設定するとテスト時に役立ちますが、この方法は実稼働システムではマシンを再起動すると /proc/sys/ 内の特別な設定が失われるためです。カスタム設定を保持するには、/etc/sysctl.conf ファイルに追加します。

システムの起動時に、init プログラムが /etc/rc.d/rc.sysinit スクリプトを実行します。このスクリプトには、/etc/ sysctl.conf を使用して sysctl を実行し、カーネルに渡される値を判断するコマンドが含

まれています。したがって、/etc/sysctl.conf に追加した値は、システムが起動するたびに有効になります。

5.5. 関連情報

以下は、proc ファイルシステムに関する追加情報の追加ソースです。

5.5.1. インストールされているドキュメント

デフォルトでは、proc ファイルシステムに関する最適なドキュメントの一部がシステムにインストールされている。

- /usr/share/doc/kernel-doc- <*version*> /Documentation/filesystems/proc.txt: /proc/ ディレクトリーのすべての側面に関する情報が含まれていますが、これらに限定されています。

- /usr/share/doc/kernel-doc- &*lt;version*> /Documentation/sysrq.txt - システム要求キーオプションの概要。

- /usr/share/doc/kernel-doc- <*version*> /Documentation/sysctl/ - さまざまな sysctl のヒントを含むディレクトリー。これには、カーネルに関する値の変更(kernel.txt)、ファイルシステムへのアクセス(fs.txt)、および仮想メモリーの使用(vm.txt)が含まれます。

- /usr/share/doc/kernel-doc- &*lt;version*> /Documentation/networking/ip-sysctl.txt: IP ネットワークオプションの詳細な概要

5.5.2. 便利な Web サイト

- http://www.linuxhq.com/: この Web サイトでは、Linux カーネルのさまざまなバージョンに関するソース、パッチ、およびドキュメントの完全なデータベースを維持します。

第6章 RAID (REDUNDANT ARRAY OF INDEPENDENT DISKS)

RAID の登場した背景には、容量が小さく手頃なディスクドライブを複数集めてアレイに結合させ、容量が大きく高価なドライブに負けないパフォーマンスと冗長性を実現しようとする動きがありました。この複数のデバイスからなるアレイは、コンピューター上では単一の論理ストレージユニットまたはドライブとして表されます。

6.1. RAID とは

RAID により、情報は複数のディスクにアクセスできます。*ディスクのストライピング* (RAID レベル 0)、*ディスクのミラーリング* (RAID レベル 1)、*パリティーによるディスクのストライピング* (RAID レベル 5) などの技術を使用して冗長性を得ながら待ち時間を抑え、帯域幅を増幅させることでハードディスクがクラッシュした場合の復元力を最大限に引き出します。

RAID は、アレイ内の各ドライブにデータを一貫して分散します。RAID は、データを一貫してサイズのチャンクに分割します（通常は 32K または 64k ですが、他の値は受け入れ可能です）。各チャンクは、使用している RAID レベルに応じて、RAID アレイのハードドライブに書き込まれます。データが読み込まれるとこのプロセスが逆をたどります。その動作はアレイ内の複数のドライブがまるで一台の大容量ドライブであるかのように見えます。

6.1.1. RAID を使用する理由

システム管理者や大容量のデータを管理しているユーザーは、RAID 技術を使用することでメリットが得られます。RAID をデプロイする主な理由を以下に示します。

- 速度を高める
- 1 台の仮想ディスクを使用してストレージ容量を増加する
- ディスク障害を最小限に抑える

6.1.2. ハードウェア RAID とソフトウェア RAID

考えられる RAID アプローチには、ハードウェア RAID とソフトウェア RAID の 2 つがあります。

ハードウェア RAID

ハードウェアベースのアレイは、RAID サブシステムをホストとは別に管理します。ホストに

対して、1 RAID アレイごとに 1 つのディスクを表します。

ハードウェア RAID デバイスは SCSI コントローラーに接続し、RAID アレイを 1 つの SCSI ドライブとして表示します。外部 RAID システムは、すべての RAID 処理 「インテリジェンス」を、外部ディスクサブシステムにあるコントローラーに移動します。サブシステム全体が通常の SCSI コントローラーを介してホストに接続され、ホストには単一のディスクとして表示されます。

RAID コントローラーカードは、オペレーティングシステムへの SCSI コントローラーのように動作し、実際のドライブ通信をすべて処理します。ユーザーはドライブを RAID コントローラー（通常の SCSI コントローラーと同様）にプラグインし、RAID コントローラー設定に追加し、オペレーティングシステムで違いを認識しません。

ソフトウェア RAID

ソフトウェア RAID では、カーネルディスク (ブロックデバイス) コード内に各種の RAID レベルを実装しています。高価ディスクコントローラーカードやホットスワップシャーシなど、最優先的な解決策を提供します。[1] 必須ではありません。ソフトウェア RAID は、SCSI ディスクだけでなく安価な IDE ディスクでも機能します。現代の高速な CPU により、ソフトウェア RAID はハードウェア RAID を上回ります。

Linux カーネルには MD ドライバーが含まれており、RAID ソリューションは完全にハードウェアに依存しないようにすることができます。ソフトウェアベースのアレイのパフォーマンスは、サーバーの CPU パフォーマンスと負荷によって異なります。

ソフトウェア RAID の詳細は、以下の主要な機能を参照してください。

- スレッド再構築プロセス
- カーネルベースの設定
- 再構築なしで Linux マシン間でのアレイの移植性
- アイドルシステムリソースを使用したバックグラウンドのアレイ再構築
- ホットスワップ可能なドライブのサポート

- 特定の CPU 最適化を活用するための自動 CPU 検出

6.1.3. RAID レベルとリニアサポート

RAID は、レベル 0、1、4、5、リニアなどのさまざまな設定に対応します。これらの RAID タイプは以下のように定義されます。

レベル 0

RAID レベル 0 は、多くの場合 「ストライピング」 と呼ばれ、パフォーマンス指向のストライプ化データマッピング手法です。これは、アレイに書き込まれるデータがストライプに分割され、アレイのメンバーディスク全体に書き込まれることを意味します。これにより低い固有コストで高い I/O パフォーマンスを実現できますが、冗長性は提供されません。レベル 0 アレイのストレージ容量は、ハードウェア RAID のメンバーディスクの合計容量またはソフトウェア RAID のメンバーパーティションの合計容量と等しくなります。

レベル 1

RAID レベル 1、または 「ミラーリング」 は、他の RAID 形式よりも長く使用されています。レベル 1 は、アレイ内の各メンバーディスクに同一データを書き込むことで冗長性を提供し、各ディスクに 「ミラーリングされた」 コピーを残します。ミラーリングは、データの可用性の単純化と高レベルにより、いまでも人気があります。レベル 1 は、読み取り時にデータ転送レートに並列アクセスを使用する可能性のある 2 つ以上のディスクで動作しますが、より一般的に I/O トランザクションレートを提供するために独立して動作します。レベル 1 は、非常に優れたデータの信頼性を提供し、読み取り集約型アプリケーションのパフォーマンスを向上させますが、比較的コストが高くなります。レベル 1 アレイのストレージ容量は、ハードウェア RAID 内のミラーリングされたハードディスクの 1 つまたはソフトウェア RAID 内のミラーリングされたパーティションの 1 つの容量と同じです。

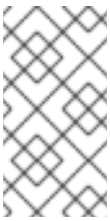
注記

RAID レベル 1 は、ドライブ領域を無駄にしたアレイ内のすべてのディスクに同じ情報を書き込むため、コストが高くなります。たとえば、ルート(/)パーティションが 2 つの 40G ドライブに存在するように RAID レベル 1 を設定している場合は、合計 80G ですが、その 80G の 40G のみにアクセスできます。もう 1 つの 40G は、最初の 40G のミラーのように動作します。

レベル 4

RAID レベル 4 でパリティー使用[2] データを保護するため、1 つのディスクドライブで連結します。大規模なファイル転送ではなく、トランザクション I/O に適しています。専用パリティー

ディスクは固有のボトルネックを表すため、レベル 4 は、ライトバックキャッシュなどの付随する技術なしではほとんど使用されません。RAID レベル 4 は、一部の RAID パーティションスキームのオプションですが、Red Hat Enterprise Linux RAID インストールで許可されるオプションではありません。ハードウェア RAID レベル 4 のストレージ容量は、メンバーディスクの容量と同じで、1 つのメンバーディスクの容量を引いたものになります。ソフトウェア RAID レベル 4 のストレージ容量は、メンバーパーティションの容量と同じで、パーティションのサイズが等しい場合は、パーティションの 1 つを引いたサイズになります。



注記

RAID レベル 4 は、RAID レベル 5 と同じ領域を使用しますが、レベル 5 にはより多くの利点があります。このため、レベル 4 はサポートされません。

レベル 5

RAID レベル 5 は RAID の最も一般的なタイプです。RAID レベル 5 は、アレイのメンバーディスクドライブの一部またはすべてにパリティーを分散することにより、レベル 4 に固有の書き込みボトルネックを排除します。パリティー計算プロセスは、パフォーマンスのボトルネックのみです。最新の CPU とソフトウェア RAID では、通常は非常に大きな問題ではありません。レベル 4 と同様に、結果は非対称パフォーマンスであり、読み取りは書き込みを大幅に上回ります。レベル 5 は多くの場合、非対称を減らすためにライトバックキャッシュで使用されます。ハードウェア RAID レベル 5 のストレージ容量は、メンバーディスクの容量と同じで、1 つのメンバーディスクの容量を引いたものになります。ソフトウェア RAID レベル 5 のストレージ容量は、メンバーパーティションの容量と同じで、パーティションのサイズが等しい場合は、パーティションの 1 つを引いたサイズになります。

リニア RAID

リニア RAID は、より大きな仮想ドライブを作成するドライブの簡易グループ化です。リニア RAID では、あるメンバードライブからチャンクが順次割り当てられます。最初のドライブが完全に満杯になったときにのみ次のドライブに移動します。これにより、メンバードライブ間の I/O 操作が分割される可能性はないため、パフォーマンスの向上は見られません。リニア RAID には冗長性がなく、実際に、信頼性は低下します。いずれかのメンバードライブに障害が発生した場合は、アレイ全体を使用することができません。容量はすべてのメンバーディスクの合計になります。

6.2. ソフトウェア RAID の設定

ユーザーは、グラフィカルインストールプロセス、テキストベースのインストールプロセス、またはキックスタートインストール中にソフトウェア RAID を設定できます。このセクションでは、Disk Druid アプリケーションを使用したインストールプロセス時のソフトウェア RAID 設定について説明し、以下の手順を説明します。

1. 物理ハードドライブでの *ソフトウェア RAID パーティション* の作成

2. ソフトウェア *RAID パーティションからの RAID デバイス* の作成

3. （オプション）RAID デバイスからの *LVM* の設定

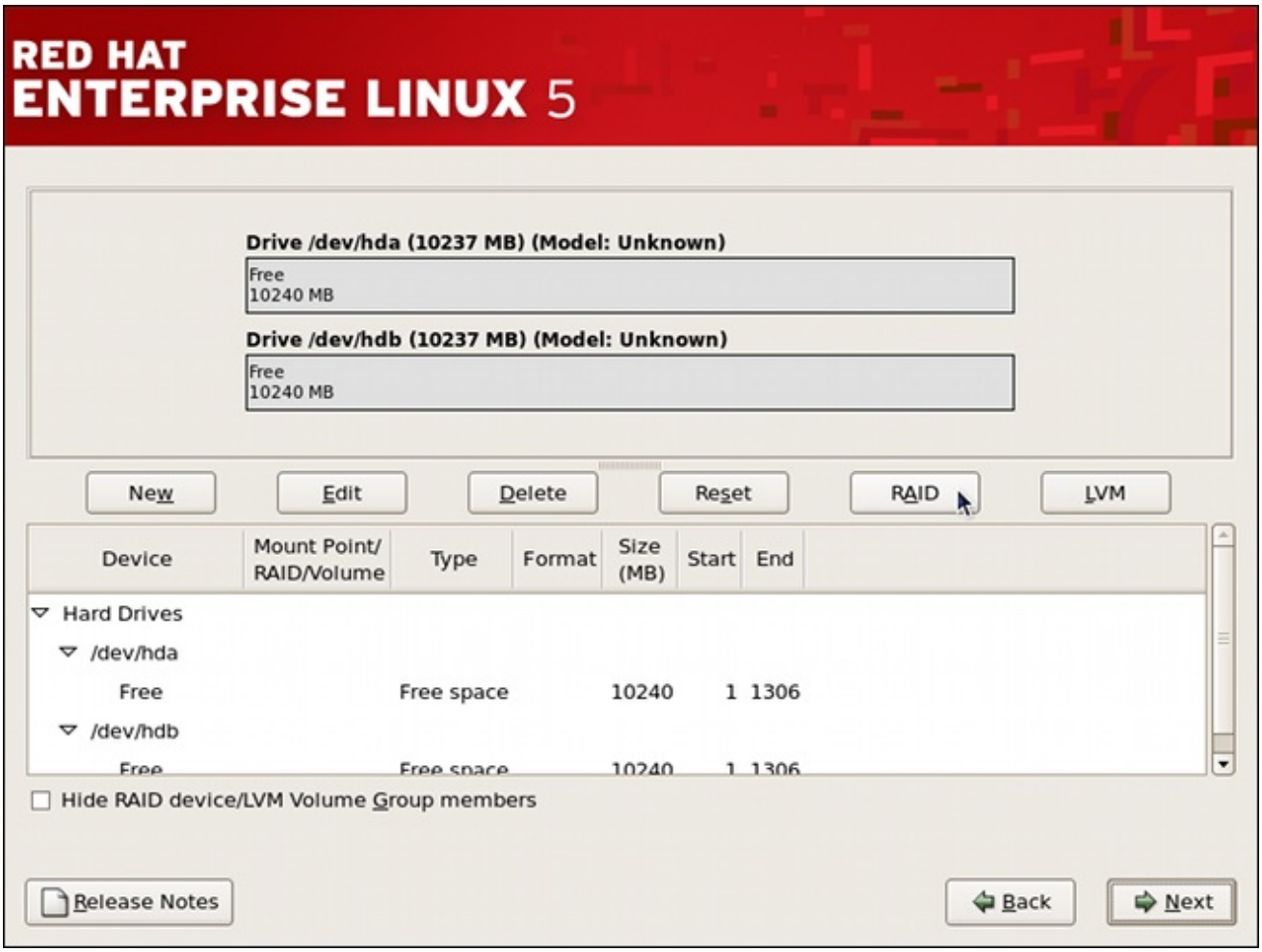
4. RAID デバイスから *ファイルシステム* を作成する。

ソフトウェア RAID を設定するには、Disk Partitioning Setup 画面のプルダウン一覧から Create custom layout を選択し、Next ボタンをクリックして、本セクションの他の手順に従います。本セクションのスクリーンショットの例では、2 つの 10 GB のディスクドライブ(/dev/hda および /dev/hdb)を使用して、単純な RAID 1 および RAID 0 設定の作成について説明し、複数の RAID デバイスを実装して簡単な RAID 設定を作成する方法について詳しく説明します。

6.2.1. RAID パーティションの作成

一般的な状況では、ディスクドライブが新しく、フォーマットされている、またはフォーマットされています。両方のドライブが、図6.1「2 つの空白ドライブ（準備完了）」 にパーティション設定のない raw デバイスとして表示されます。

図6.1 2 つの空白ドライブ（準備完了）

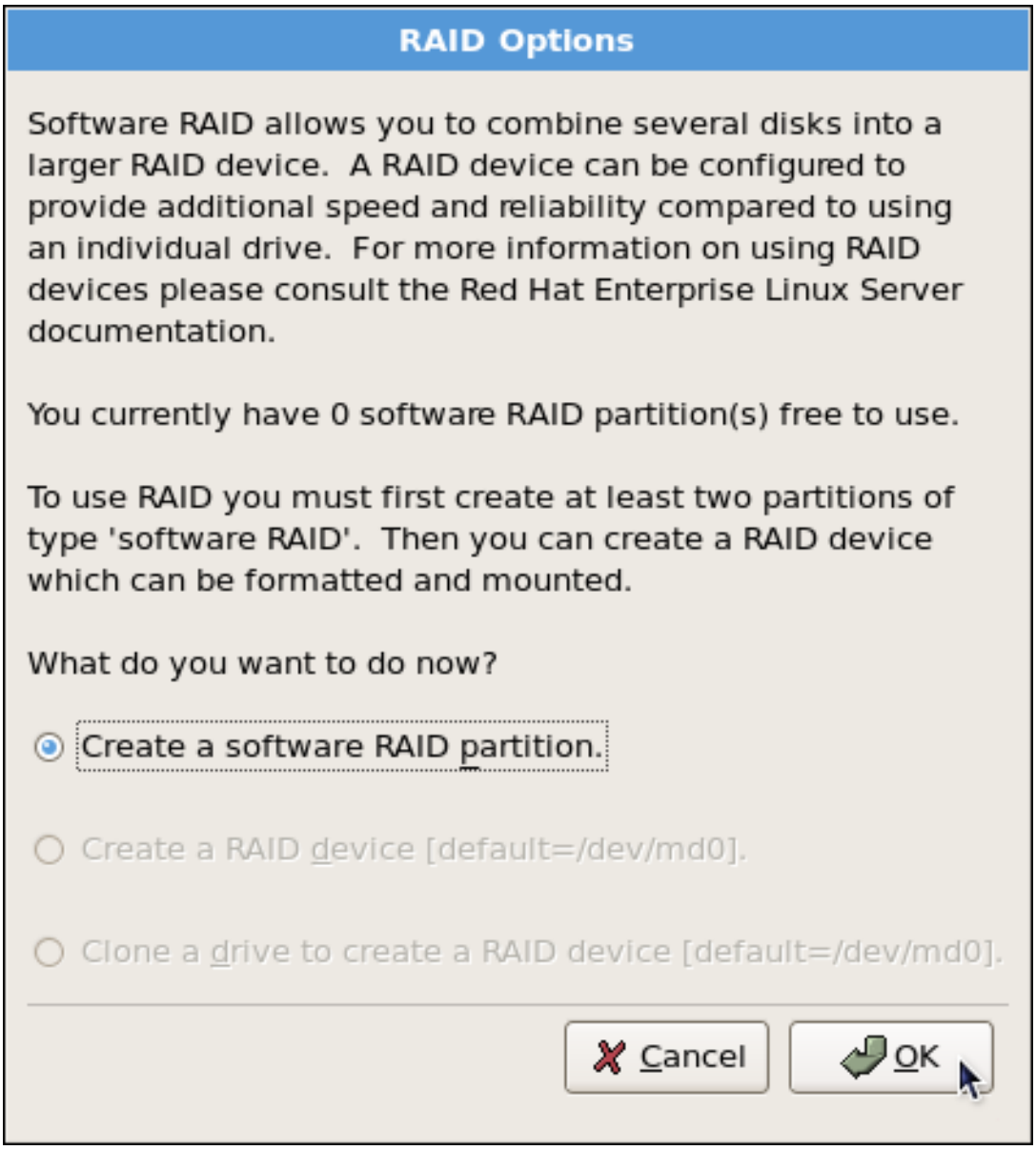


[D]

1. Disk Druid で RAID ボタンをクリックして、ソフトウェア RAID 作成画面を入力します。

2. 図6.2「RAID パーティションオプション」 に示されるよう に、ソフトウェア RAID パーティションの作成 を 選択して RAID パーティションを作成します。RAID パーティションや RAID デバイスが作成されるまで、他の RAID オプション（マウントポイントの入力など）は利用できないことに注意してください。OK をクリックして選択を確定します。

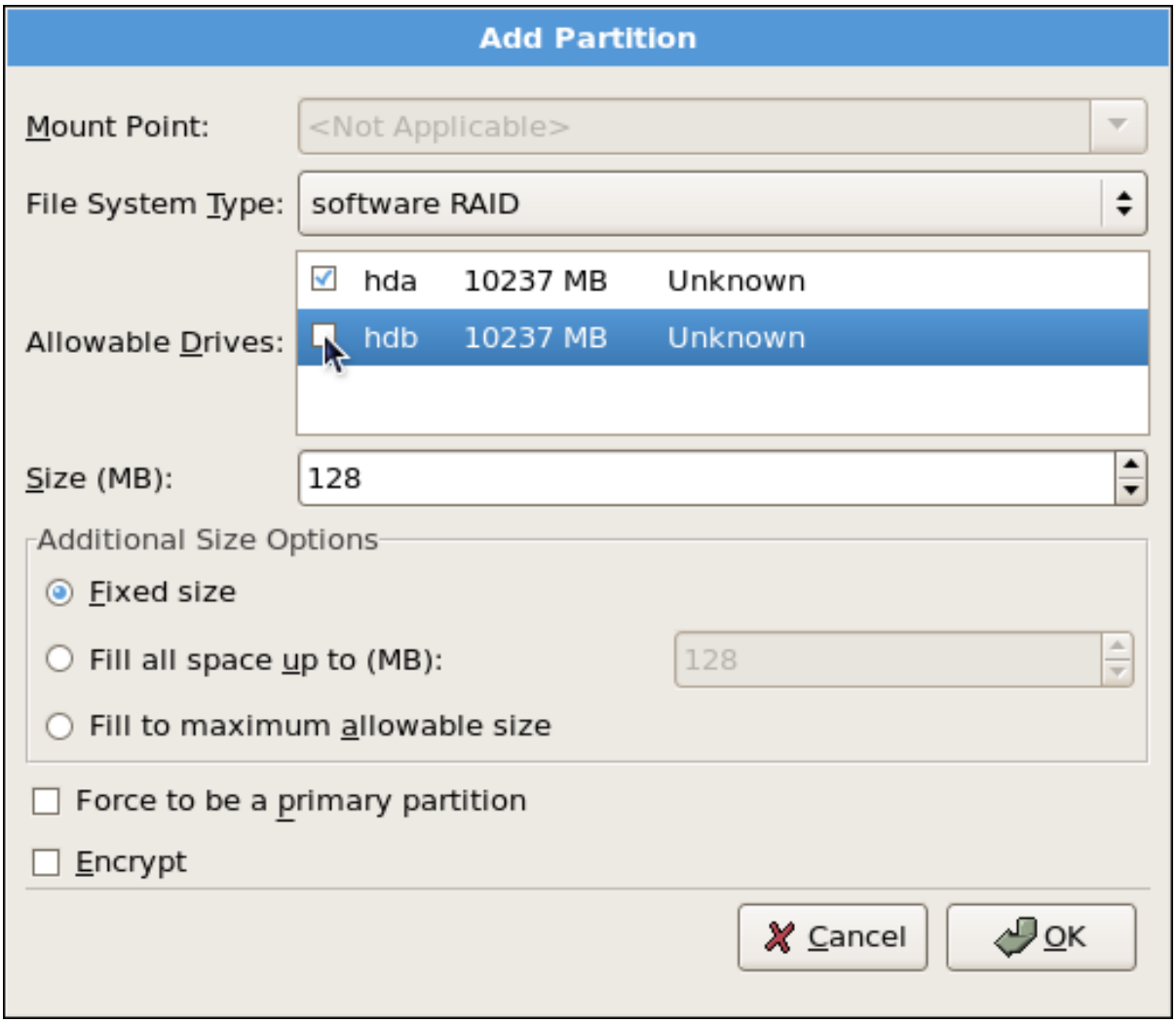
図6.2 RAID パーティションオプション



[D]

3. ソフトウェア RAID パーティションは、1 つのドライブに制限する必要があります。Allowable Drives には、RAID に使用するドライブを選択します。複数のドライブがある場合は、デフォルトですべてのドライブが選択されるため、不要なドライブの選択を解除する必要があります。

図6.3 RAID パーティションの追加



[D]

4. サイズ (MB) フィールドを編集し、パーティションのサイズ(MB 単位)を入力します。

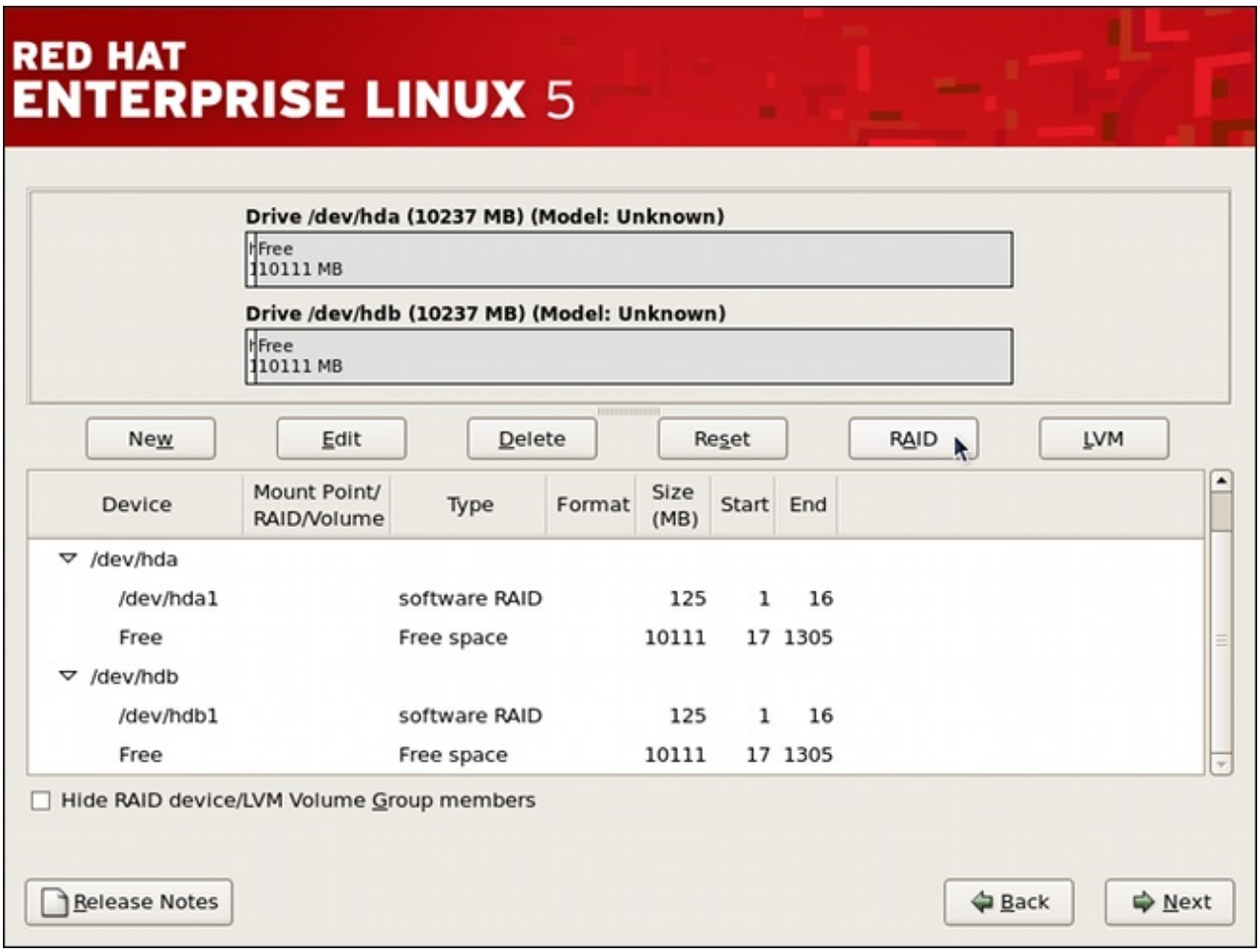
5. Fixed Size を選択して、パーティションサイズを指定します。Fill all space up (MB)を選択し、値(MB 単位)を入力してパーティションサイズの範囲を指定します。ハードディスクの最大使用可能容量を確保するために、最大許容サイズ を選択します。複数の領域を増やすと、ディスクで利用可能な空き領域が共有されることに注意してください。

6. パーティションをプライマリーパーティションにする場合は、Force to be a primary partition を選択します。プライマリーパーティションは、ハードドライブの最初の 4 つのパーティションの 1 つです。選択されていない場合は、パーティションが論理パーティションとして作成されます。他のオペレーティングシステムがすでにシステム上にある場合は、このオプションの選択を解除する必要があります。プライマリーパーティションと論理/拡張パーティションの詳細は、『Red Hat Enterprise Linux インストールガイド』 の付録セクションを参照してください。

これらの手順を繰り返して、RAID 設定に必要な数だけパーティションを作成します。すべての

パーティションは RAID パーティションである必要はないことに注意してください。たとえば、/boot パーティションのみをソフトウェア RAID デバイスとして設定し、root パーティション(/)、/home、および swap を通常のファイルシステムとして残すことができます。図6.4「RAID 1 パーティションの準備が Ready、Pre-Device、およびマウントポイントの作成」 RAID 1 設定(/boot用)に正常に割り当てられた領域を表示します。これは、RAID デバイスおよびマウントポイントの作成の準備ができています。

図6.4 RAID 1 パーティションの準備が Ready、Pre-Device、およびマウントポイントの作成



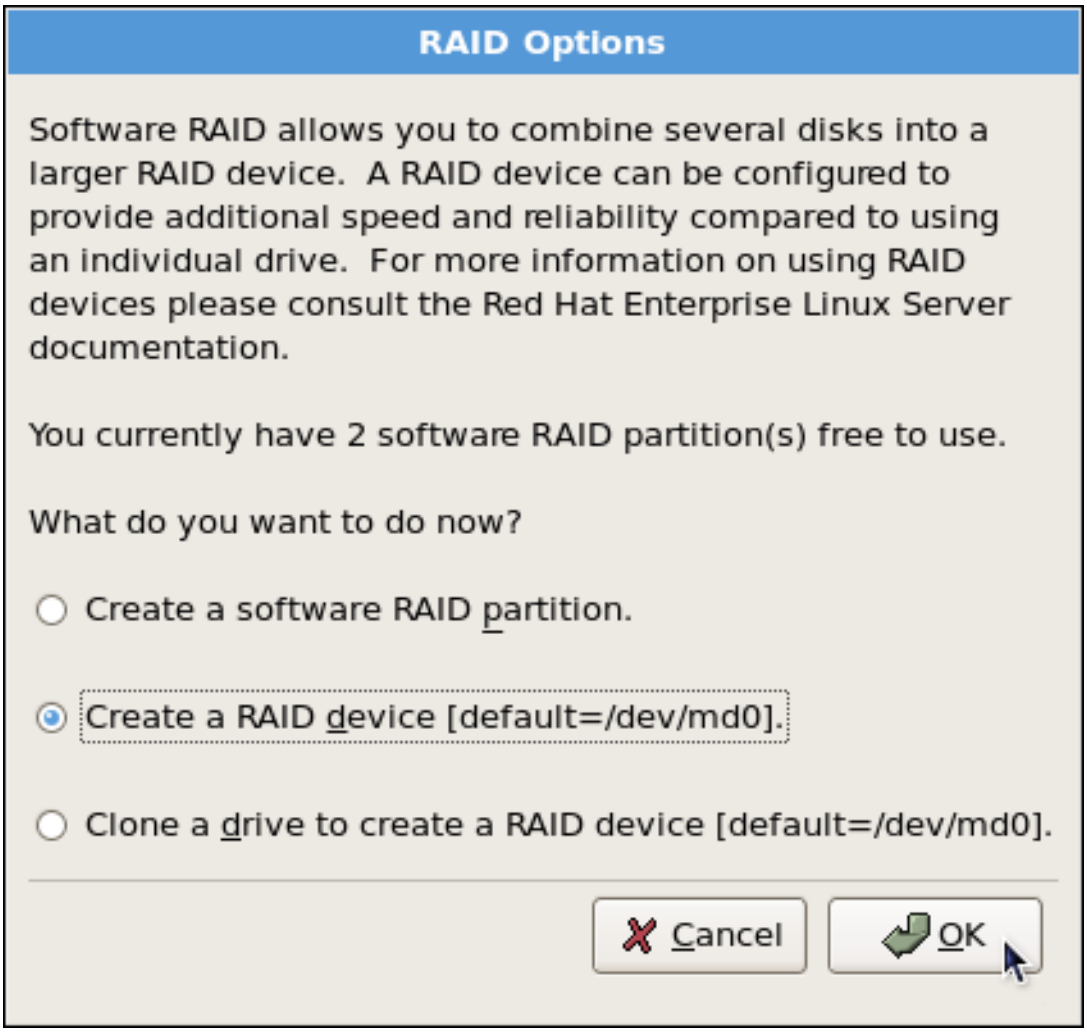
[D]

6.2.2. RAID デバイスとマウントポイントの作成

ソフトウェア RAID パーティションとしてすべてのパーティションを作成したら、RAID デバイスとマウントポイントを作成する必要があります。

1. メインのパーティション設定画面で、RAID ボタンをクリックします。RAID Options ダイアログが 図6.5「RAID オプション」 に示すように表示されます。

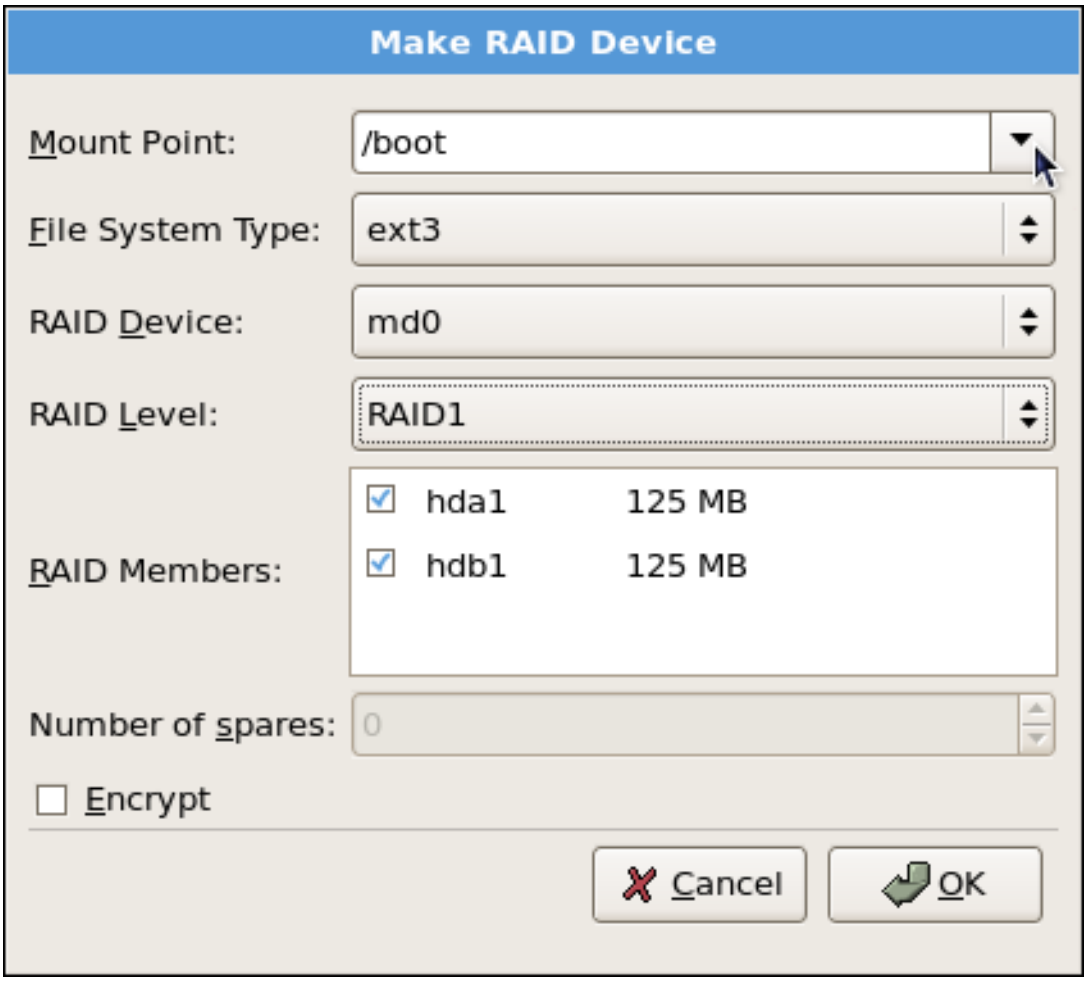
図6.5 RAID オプション



[D]

2. **RAID デバイスの作成 オプションを選択し、OK をクリックします。図6.6「RAID デバイスの作成およびマウントポイントの割り当て」 に示すように、Make RAID Device ダイアログが表示されます。これにより、RAID デバイスを作成してマウントポイントを割り当てることができます。**

図6.6 RAID デバイスの作成およびマウントポイントの割り当て



[D]

3. マウントポイント プルダウン リストから マウント ポイントを選択します。

4. File System Type プルダウンリストからパーティションのファイルシステムタイプを選択します。この時点で、動的 LVM ファイルシステムまたは従来の静的 ext2/ext3 ファイルシステムのいずれかを設定できます。インストールプロセス中に LVM とその設定の詳細は、11 章*LVM (論理ボリュームマネージャー)* を参照してください。LVM が必要ない場合は、次の手順に進みます。

5. RAID Device pulldown リストから、md0 などのデバイス名を選択します。

6. RAID レベル から、必要 な RAID レベルを選択します。

注記

/boot の RAID パーティションを作成する場合は、RAID レベル 1 を選択し、最初の 2 つのドライブ(IDE first、SCSI 秒)のいずれかを使用する必要があります。/boot の個別の RAID パーティションを作成しておらず、ルートファイルシステム（つまり /）に RAID パーティションを作成する場合は、RAID レベル 1 で、最初の 2 つのドライブ(IDE first、SCSI second)のいずれかを使用する必要があります。

7. 作成した RAID パーティションが RAID Members 一覧に表示されます。RAID デバイスの作成に使用するパーティションを選択します。

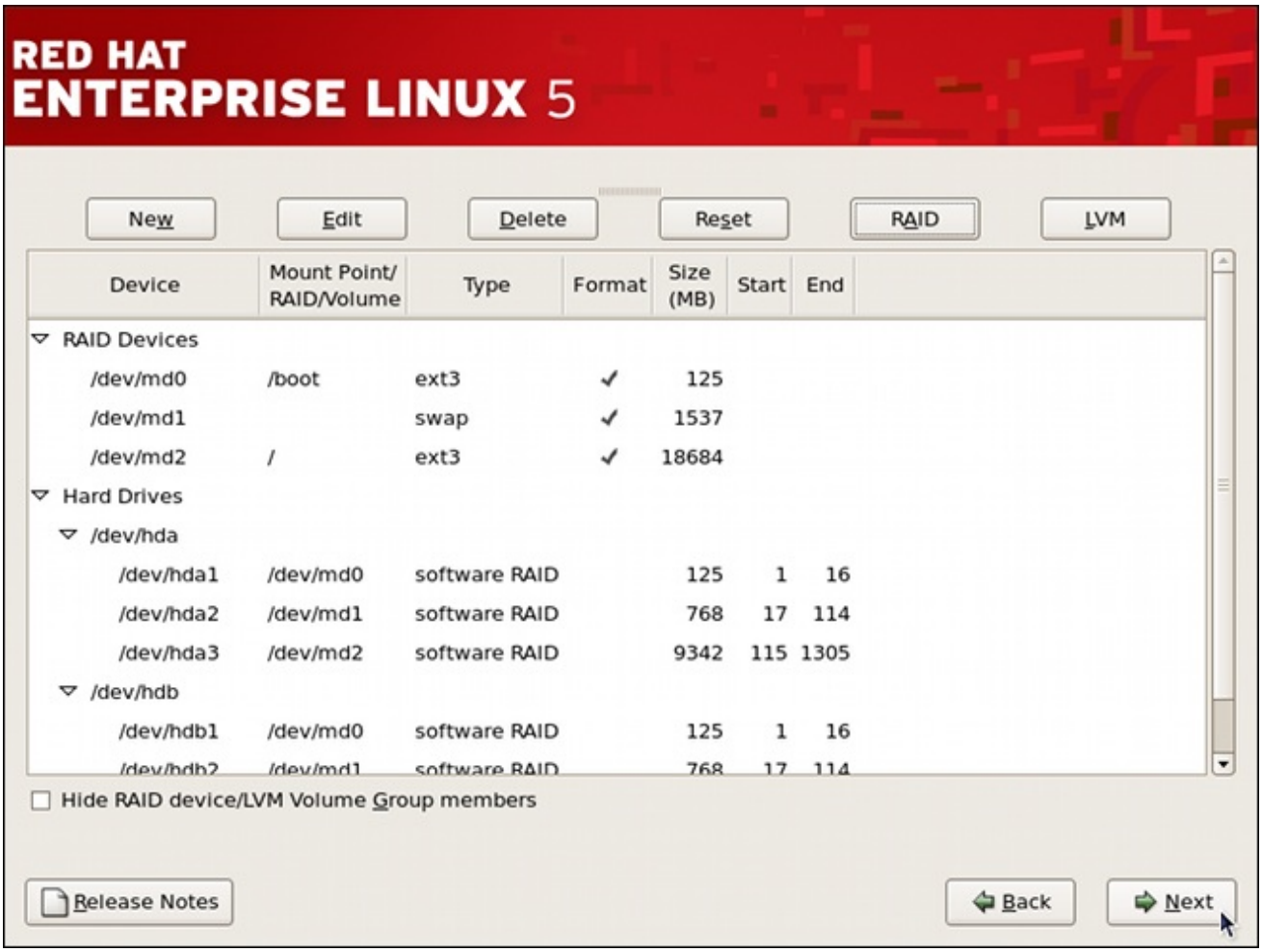
8. RAID 1 または RAID 5 を設定する場合は、スペアの数 フィールドに 予備のパーティションの数 を指定します。ソフトウェア RAID パーティションに障害が発生した場合、スペアは自動的に代替として使用されます。指定するスペアごとに、(RAID デバイスのパーティションに加えて)追加のソフトウェア RAID パーティションを作成する必要があります。RAID デバイスのパーティションと、スペアのパーティションを選択します。

9. OK をクリックして設定を確定します。RAID デバイスがドライブの 概要一覧に表示 されます。

10. root パーティション(/)、ホームディレクトリー(/)、swap などの追加のパーティション、デバイス、マウントポイントを設定するには、この章全体を繰り返します。

設定全体を完了すると、図6.7「RAID 設定の例」 に示す図は、RAID の使用を除き、デフォルト設定に似ています。

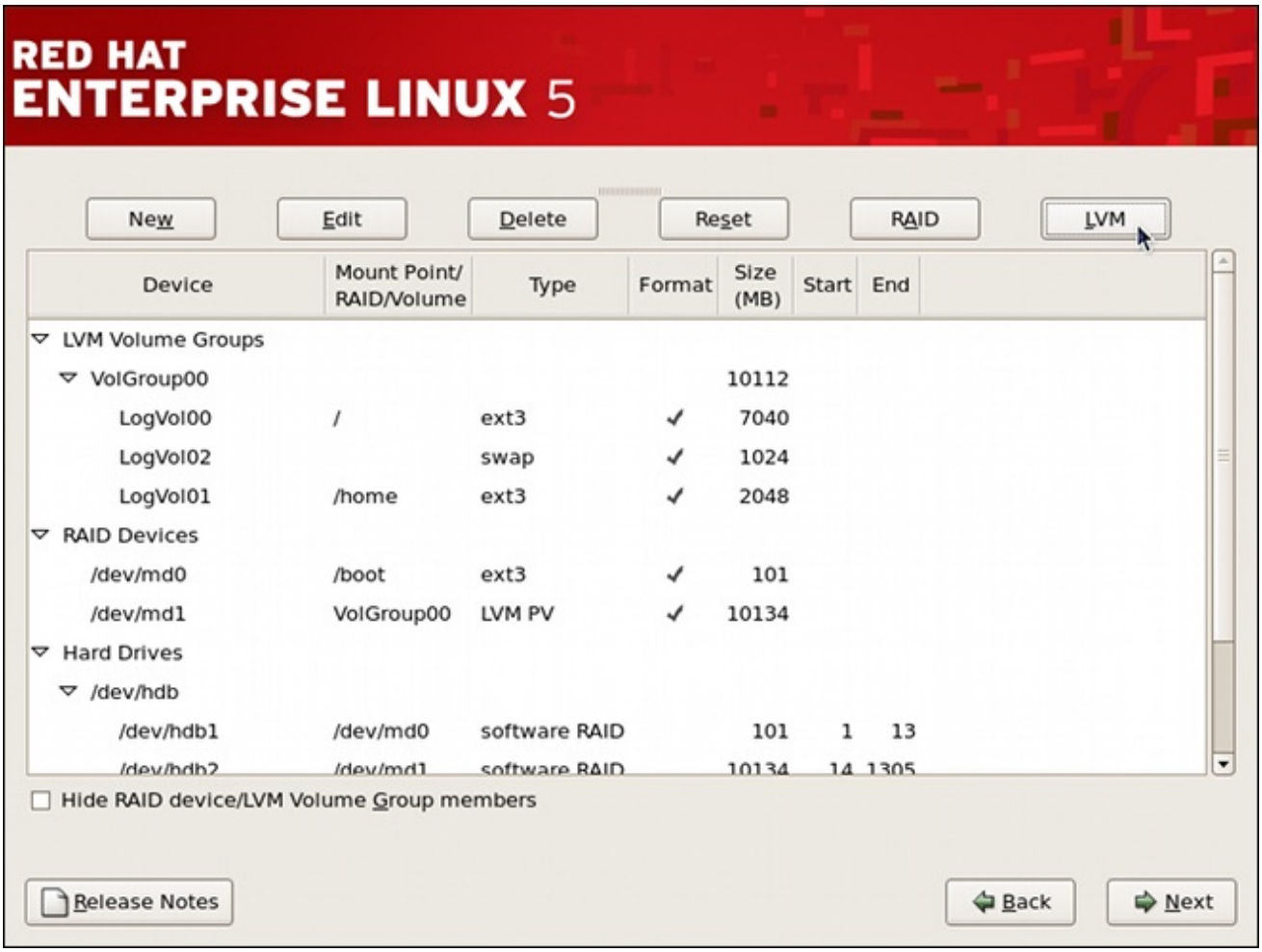
図6.7 RAID 設定の例



[D]

図6.8「LVM 設定を使用した RAID の例」 で示した図は、RAID および LVM の設定例です。

図6.8 LVM 設定を使用した RAID の例



[D]

Next をクリックして、インストールプロセスを続行できます。詳細は、『Red Hat Enterprise Linux インストールガイド』 を参照してください。

6.3. ソフトウェア RAID の管理

このセクションでは、インストール後のソフトウェア RAID の設定および管理について説明し、以下のトピックを説明します。

- **既存のソフトウェア RAID 設定の確認**
- **新しい RAID デバイスの作成**
- **アレイ内の障害のあるデバイスの置き換え**

- 既存のアレイに新しいデバイスを追加する。
- 既存の RAID デバイスの非アクティブ化および削除
- 設定を保存します。

本セクションのすべての例では、前のセクションでのソフトウェア RAID 設定を使用します。

6.3.1. RAID 設定の確認

ソフトウェア RAID を使用している場合は、現在アクティブなすべての RAID デバイスの基本情報が /proc/mdstat 特殊ファイルに保存されます。これらのデバイスを一覧表示するには、シェルプロンプトで以下を入力して、このファイルの内容を表示します。

```
cat /proc/mdstat
```

特定のデバイスが RAID デバイスかコンポーネントデバイスであるかを確認するには、root で以下の形式で コマンドを実行します。

```
mdadm --query device...
```

RAID デバイスの詳細を調べるには、次のコマンドを使用します。

```
mdadm --detail raid_device...
```

同様に、コンポーネントデバイスを調べるには、以下を入力します。

```
mdadm --examine component_device...
```

mdadm --detail コマンドは RAID デバイスに関する情報を表示しますが、mdadm --examine は、特定のコンポーネントデバイスに関連する RAID デバイスに関する情報のみをリレーします。この違いは、別の RAID デバイスのコンポーネントである RAID デバイスを使用する際に特に重要です。

mdadm --query コマンド、mdadm --detail コマンドおよび mdadm --examine コマンドの両方を使用すると、一度に複数のデバイスを指定できます。

例6.1 RAID 設定の確認

システムが 図6.7「RAID 設定の例」 の設定を使用すると仮定します。シェルプロンプトで以下を入力すると、/dev/md0 が RAID デバイスであることを確認できます。

```
~]# mdadm --query /dev/md0
/dev/md0: 125.38MiB raid1 2 devices, 0 spares. Use mdadm --detail for more detail.
/dev/md0: No md super block found, not an md component.
```

ご覧のとおり、上記のコマンドは RAID デバイスとその設定の概要のみを生成します。より詳細な情報を表示するには、代わりに以下のコマンドを使用します。

```
~]# mdadm --detail /dev/md0
/dev/md0:
        Version : 0.90
  Creation Time : Tue Jun 28 16:05:49 2011
     Raid Level : raid1
     Array Size : 128384 (125.40 MiB 131.47 MB)
  Used Dev Size : 128384 (125.40 MiB 131.47 MB)
   Raid Devices : 2
  Total Devices : 2
Preferred Minor : 0
    Persistence : Superblock is persistent

    Update Time : Thu Jun 30 17:06:34 2011
          State : clean
 Active Devices : 2
Working Devices : 2
 Failed Devices : 0
  Spare Devices : 0

           UUID : 49c5ac74:c2b79501:5c28cb9c:16a6dd9f
         Events : 0.6

    Number   Major   Minor   RaidDevice State
       0       3       1        0      active sync   /dev/hda1
       1       3      65        1      active sync   /dev/hdb1
```

最後に、現在アクティブな RAID デバイスの一覧を表示するには、次のコマンドを実行します。

```
~]$ cat /proc/mdstat
Personalities : [raid0] [raid1]
md0 : active raid1 hdb1[1] hda1[0]
      128384 blocks [2/2] [UU]

md1 : active raid0 hdb2[1] hda2[0]
      1573888 blocks 256k chunks
```

```
md2 : active raid0 hdb3[1] hda3[0]
      19132928 blocks 256k chunks

unused devices: <none>
```

6.3.2. 新しい RAID デバイスの作成

新しい RAID デバイスを作成するには、root で以下の形式の コマンドを使用します。

```
mdadm --create raid_device --level=level --raid-devices=number component_device...
```

これは、RAID アレイを作成する最も簡単な方法です。予備のデバイス数、ストライプアレイのブロックサイズ、アレイに write-intent ビットマップがある場合は、さらに多くのオプションを指定できます。これらのオプションはすべてパフォーマンスに大きな影響を与える可能性がありますが、本書では扱いません。詳細は、man ページの mdadm(8)の 『CREATE MODE』 セクションを参照してください。

例6.2 新しい RAID デバイスの作成

システムに未使用の SCSI ディスクドライブが 2 つあり、これらの各デバイスには同じサイズのパーティションが 1 つだけ存在することを前提とします。

```
~]# ls /dev/sd*
/dev/sda  /dev/sda1  /dev/sdb  /dev/sdb1
```

/dev/sda1 および /dev/sdb1 から新しい RAID レベル 1 アレイとして /dev/md3 を作成するには、次のコマンドを実行します。

```
~]# mdadm --create /dev/md3 --level=1 --raid-devices=2 /dev/sda1 /dev/sdb1
mdadm: array /dev/md3 started.
```

6.3.3. 障害のあるデバイスの置き換え

ソフトウェア RAID 内の特定のデバイスを置き換えるには、root で以下のコマンドを実行して、最初に faulty とマークされていることを確認します。

```
mdadm raid_device --fail component_device
```

次に、以下の形式で コマンドを使用して、アレイから障害のあるデバイスを削除します。

```
mdadm raid_device --remove component_device
```

デバイスが再び動作したら、アレイに再追加できます。

```
mdadm raid_device --add component_device
```

例6.3 障害のあるデバイスの置き換え

システムにアクティブな RAID デバイス /dev/md3 があり、以下のレイアウト（つまり、例6.2「新しい RAID デバイスの作成」で作成した RAID デバイス）があるとします。

```
~]# mdadm --detail /dev/md3 | tail -n 3
   Number   Major   Minor   RaidDevice State
      0       8       1        0      active sync   /dev/sda1
      1       8      17        1      active sync   /dev/sdb1
```

最初のディスクドライブに障害が発生し、置き換える必要があるとします。これを行うには、まず /dev/sdb1 デバイスを faulty とマークします。

```
~]# mdadm /dev/md3 --fail /dev/sdb1
mdadm: set /dev/sdb1 faulty in /dev/md3
```

次に、RAID デバイスから削除します。

```
~]# mdadm /dev/md3 --remove /dev/sdb1
mdadm: hot removed /dev/sdb1
```

ハードウェアの置き換え後すぐに、次のコマンドを使用して、デバイスをアレイに戻すことができます。

```
~]# mdadm /dev/md3 --add /dev/sdb1
mdadm: added /dev/sdb1
```

6.3.4. RAID デバイスの拡張

新しいデバイスを既存のアレイに追加するには、root で以下の形式のコマンドを使用します。

```
mdadm raid_device --add component_device
```

これにより、デバイスが予備デバイスとして追加されます。このデバイスをアクティブに使用するようにアレイを拡張するには、シェルプロンプトで以下を入力します。

```
mdadm --grow raid_device --raid-devices=number
```

例6.4 RAID デバイスの拡張

システムにアクティブな RAID デバイス /dev/md3 があり、以下のレイアウト（つまり、例6.2「新しい RAID デバイスの作成」で作成した RAID デバイス）があるとします。

```
~]# mdadm --detail /dev/md3 | tail -n 3
   Number   Major   Minor   RaidDevice State
      0       8       1        0      active sync   /dev/sda1
      1       8      17        1      active sync   /dev/sdb1
```

また、新しい SCSI ディスクドライブ /dev/sdc が追加され、パーティションが 1 つだけあることを前提としています。/dev/md3 アレイに追加するには、シェルプロンプトで以下を入力します。

```
~]# mdadm /dev/md3 --add /dev/sdc1
mdadm: added /dev/sdc1
```

これにより、/dev/sdc1 が予備デバイスとして追加されます。アレイのサイズを実際に使用するように変更するには、以下を入力します。

```
~]# mdadm --grow /dev/md3 --raid-devices=3
```

6.3.5. RAID デバイスの削除

既存の RAID デバイスを削除するには、最初に root で以下のコマンドを実行して無効にします。

```
mdadm --stop raid_device
```

無効にしたら、RAID デバイス自体を削除します。

```
mdadm --remove raid_device
```

最後に、特定のアレイに関連付けられたすべてのデバイスでゼロのスーパーブロックを実行します。

mdadm --zero-superblock *component_device*…

例6.5 RAID デバイスの削除

システムにアクティブな RAID デバイス /dev/md3 があり、以下のレイアウト（つまり、例6.4「RAID デバイスの拡張」で作成した RAID デバイス）があるとします。

```
~]# mdadm --detail /dev/md3 | tail -n 4
   Number   Major   Minor   RaidDevice State
      0       8       1        0      active sync   /dev/sda1
      1       8      17        1      active sync   /dev/sdb1
      2       8      33        2      active sync   /dev/sdc1
```

このデバイスを削除するには、シェルプロンプトで以下を入力して停止します。

```
~]# mdadm --stop /dev/md3
mdadm: stopped /dev/md3
```

停止したら、以下のコマンドを実行して /dev/md3 デバイスを削除できます。

```
~]# mdadm --remove /dev/md3
```

最後に、関連するすべてのデバイスからスーパーブロックを削除するには、次のコマンドを実行します。

```
~]# mdadm --zero-superblock /dev/sda1 /dev/sdb1 /dev/sdc1
```

6.3.6. 設定の保持

デフォルトでは、mdadm コマンドによる変更は現行セッションにのみ適用され、システムの再起動は維持されません。システムの起動時に、mdmonitor サービスは /etc/mdadm.conf 設定ファイルの内容を読み取り、起動する RAID デバイスを確認します。ソフトウェア RAID がグラフィカルインストールプロセス中に設定されている場合、このファイルには、デフォルトで 表6.1「一般的な mdadm.conf ディレクティブ」 に記載されているディレクティブが含まれています。

表6.1 一般的な mdadm.conf ディレクティブ

オプション	説明

オプション	説明
ARRAY	特定のアレイを特定できます。
デバイス	RAID コンポーネントをスキャンするデバイスの一覧を指定できます（例：「/dev/hda1」）。キーワード partitions を使用して、/proc/ partitions にリストされているすべてのパーティションを使用するか、またはコンテナーを使用してアレイコンテナーを指定することもできます。
MAILADDR	アラートの場合に、使用するメールアドレスを指定できます。

設定に関係なく、現在使用されている ARRAY 行を一覧表示するには、root で以下のコマンドを実行します。

```
mdadm --detail --scan
```

このコマンドの出力を使用して、/etc/mdadm.conf ファイルに追加する行を確認します。特定のデバイスの ARRAY 行を表示することもできます。

```
mdadm --detail --brief raid_device
```

このコマンドの出力をリダイレクトすることで、このような行を 1 つのコマンドで設定ファイルに追加できます。

```
mdadm --detail --brief raid_device >> /etc/mdadm.conf
```

例6.6 設定の保持

デフォルトでは、/etc/mdadm.conf には、システムのインストール時に作成されたソフトウェア RAID 設定が含まれます。

```
# mdadm.conf written out by anaconda
DEVICE partitions
MAILADDR root
ARRAY /dev/md0 level=raid1 num-devices=2 UUID=49c5ac74:c2b79501:5c28cb9c:16a6dd9f
ARRAY /dev/md1 level=raid0 num-devices=2 UUID=76914c11:5bfa2c00:dc6097d1:a1f4506d
ARRAY /dev/md2 level=raid0 num-devices=2 UUID=2b5d38d0:aea898bf:92be20e2:f9d893c5
```

例6.2「新しい RAID デバイスの作成」 で示したように /dev/md3 デバイスを作成した場合は、以下のコマンドを実行してこれを永続化できます。

```
~]# mdadm --detail --brief /dev/md3 >> /etc/mdadm.conf
```

6.4. 関連情報

RAID の詳細は、以下のリソースを参照してください。

6.4.1. インストールされているドキュメント

- mdadm man ページ： mdadm ユーティリティーの man ページです。
- mdadm.conf の man ページ：利用可能な /etc/mdadm.conf 設定オプションの包括的なリストを提供する man ページです。

---

[1] ホットスワップシャーシを使用すると、システムの電源を切らずにハードドライブを削除できます。

[2] パリティー情報は、アレイ内の残りのメンバーディスクのコンテンツに基づいて計算されます。この情報は、アレイ内のいずれかのディスクに障害が発生した場合にデータの再構築に使用できます。その後、再構築されたデータを使用して、交換前に失敗したディスクに I/O 要求に対応でき、交換後に失敗したディスクを接続します。

第7章 SWAP 領域

7.1. スワップ領域とは

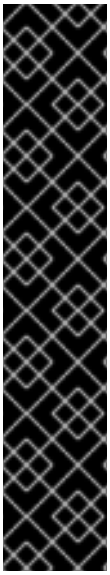
Linux の *スワップ領域* は、物理メモリー (RAM) が不足すると使用されます。システムに多くのメモリーリソースが必要で、RAM が不足すると、メモリーの非アクティブなページがスワップ領域に移動します。スワップ領域は、RAM が少ないマシンで役に立ちますが、RAM の代わりに使用しないようにしてください。スワップ領域はハードドライブにあり、そのアクセス速度は物理メモリーに比べると遅くなります。

スワップ領域の設定は、専用のスワップパーティション (推奨)、スワップファイル、またはスワップパーティションとスワップファイルの組み合せが考えられます。

過去数年、推奨されるスワップ領域のサイズは、システムの RAM サイズに比例して増加していました。ただし、最新のシステムのメモリー量が数百ギガバイトに増加するため、システムが必要とするスワップ領域が、そのシステムで実行しているメモリーワークロードの機能であることが認識されるようになりました。ただし、通常スワップ領域がインストール時に指定されており、システムのメモリーワークロードを事前に決定することが困難な場合は、以下の表を使用してシステムスワップを決定することが推奨されます。

表7.1 システムの推奨 swap 領域

システム内の RAM の容量	推奨されるスワップ領域
4GB 以下の RAM	最小 2GB のスワップ領域
4GB から 16GB の RAM	最小 4GB のスワップ領域
16GB から 64GB の RAM	最小 8GB のスワップ領域
64GB から 256GB の RAM	最小 16GB のスワップ領域
256GB から 512GB の RAM	最小 32GB のスワップ領域



重要な影響

スワップ領域として割り当てられたファイルシステムおよび LVM2 ボリュームは、変更時に使用 できません。たとえば、スワップ領域を割り当てることができるシステムプロセスや、カーネルによる swap 容量の割り当てと使用はできません。free コマンドおよび cat /proc/swaps コマンドを使用して、スワップの使用量と、使用中の場所を確認します。

スワップ領域の変更を実現する最善の方法は、システムをレスキューモードで起動し、本章の残りの部分で手順（各シナリオ）に従うことです。レスキューモードで起動する方法については、Red Hat Enterprise Linux インストールガイド を参照してください。ファイルシステムをマウントするように指示されたら、スキップ を選択します。

7.2. スワップ領域の追加

場合によっては、インストールした後にさらに swap 領域の追加が必要になることもあります。たとえば、システムの RAM 容量を 128 MB から 256 MB にアップグレードできますが、スワップ領域は 256 MB しかありません。メモリー意図しない操作を実行する場合や、大量のメモリーを必要とするアプリケーションを実行する場合は、スワップ領域を 512 MB に増やすことが有益です。

選択肢が 3 つあります: 新規の swap パーティションの作成、新規の swap ファイルの作成、あるいは既存の LVM2 論理ボリューム上で swap の拡張。この中では、既存論理ボリュームを拡張することが推奨されます。

7.2.1. LVM2 論理ボリュームでのスワップ領域の拡張

LVM2 スワップ論理ボリュームを拡張するには、以下を行います(/dev/VolGroup00/LogVol01 が拡張するボリュームであるとします)。

1. 関連付けられている論理ボリュームのスワップ機能を無効にします。

```
swapoff -v /dev/VolGroup00/LogVol01
```

2. LVM2 論理ボリュームのサイズを 256 MB 増やします。

```
lvm lvresize /dev/VolGroup00/LogVol01 -L +256M
```

3. 新しいスワップ領域をフォーマットします。

```
mkswap /dev/VolGroup00/LogVol01
```

4. **拡張論理ボリュームを有効にします。**

```
swapon -va
```

5. **論理ボリュームが適切に拡張されていることを確認します。**

```
cat /proc/swaps
free
```

7.2.2. スワップの LVM2 論理ボリュームの作成

スワップボリュームグループを追加するには (/dev/VolGroup00/LogVol02 が追加するスワップボリュームであると想定)、以下を実行します。

1. **サイズが 256 MB の LVM2 論理ボリュームを作成します。**

```
lvm lvcreate VolGroup00 -n LogVol02 -L 256M
```

2. **新しいスワップ領域をフォーマットします。**

```
mkswap /dev/VolGroup00/LogVol02
```

3. **次のエントリーを /etc/fstab ファイルに追加します。**

```
/dev/VolGroup00/LogVol02   swap    swap    defaults    0 0
```

4. **拡張論理ボリュームを有効にします。**

```
swapon -va
```

5. **論理ボリュームが適切に拡張されていることを確認します。**

```
cat /proc/swaps
free
```

7.2.3. スワップファイルの作成

swap ファイルを追加します。

1. 新しいスワップファイルのサイズをメガバイト単位で指定してから、そのサイズに 1024 をかけてブロック数を指定します。たとえばスワップファイルのサイズが 64 MB の場合は、ブロック数が 65536 になります。

2. root でシェルプロンプトで、count を目的のブロックサイズと同じにして以下のコマンドを入力します。

```
dd if=/dev/zero of=/swapfile bs=1024 count=65536
```

3. 新規に作成されたファイルの送信を変更します。

```
chmod 0600 /swapfile
```

4. 次のコマンドでスワップファイルをセットアップします。

```
mkswap /swapfile
```

5. スワップファイルを起動時に自動的にではなく、すぐに有効にする場合は、以下を実行します。

```
swapon /swapfile
```

6. 起動時に有効にするには、/etc/fstab を編集して以下のエントリーを含めます。

```
/swapfile       swap          swap    defaults       0 0
```

次にシステムが起動すると、新しいスワップファイルが有効になります。

7. 新しいスワップファイルを追加して有効にしたら、cat /proc/swaps コマンドまたは free コマンドの出力を表示して、そのファイルが有効であることを確認します。

7.3. スワップ領域の削除

インストールの後に swap 領域を減らすことが賢明な場合もあります。たとえば、システムの RAM 容量を 1 GB から 512 MB にダウングレードするとします。しかし、依然として 2 GB のスワップスペースが割り当てられています。ディスク領域が大きくなる (2 GB など) と無駄になる可能性があるため、スワップ領域を 1 GB に減らすことでメリットを得られることがあります。

ここでも選択肢が 3 つあります: swap 用に使用していた LVM2 論理ボリューム全体を削除、swap ファイルの削除、あるいは既存の LVM2 論理ボリューム上の swap 領域の低減。

7.3.1. LVM2 論理ボリュームでのスワップ領域の縮小

以下のようにして LVM2 の swap 論理ボリュームを縮小します (/dev/VolGroup00/LogVol01 が縮小するボリュームであるとします)。

1. 関連付けられている論理ボリュームのスワップ機能を無効にします。

```
swapoff -v /dev/VolGroup00/LogVol01
```

2. LVM2 論理ボリュームのサイズを変更して 512 MB 削減します。

```
lvm lvreduce /dev/VolGroup00/LogVol01 -L -512M
```

3. 新しいスワップ領域をフォーマットします。

```
mkswap /dev/VolGroup00/LogVol01
```

4. 拡張論理ボリュームを有効にします。

```
swapon -va
```

5. 論理ボリュームが正しく縮小されたことをテストします。

```
cat /proc/swaps
free
```

7.3.2. スワップの LVM2 論理ボリュームの削除

スワップ論理ボリュームは使用できません（システムのロックやボリューム上のプロセスはありません）。これを実現する最も簡単な方法は、システムをレスキューモードで起動することです。レスキューモードで起動する方法については、『Red Hat Enterprise Linux インストールガイド』 を参照してください。ファイルシステムをマウントするように指示されたら、スキップ を選択します。

swap ボリュームグループを削除します (/dev/VolGroup00/LogVol02 が削除するボリュームであるとします)。

1. 関連付けられている論理ボリュームのスワップ機能を無効にします。

```
swapoff -v /dev/VolGroup00/LogVol02
```

2. サイズ 512MB の LVM2 論理ボリュームを削除します。

```
lvm lvremove /dev/VolGroup00/LogVol02
```

3. /etc/fstab ファイルから次のエントリーを削除します。

```
/dev/VolGroup00/LogVol02   swap    swap    defaults    0 0
```

4. 論理ボリュームが削除されていることをテストします。

```
cat /proc/swaps
free
```

7.3.3. スワップファイルの削除

swap ファイルを削除します。

1. root としてシェルプロンプトで、以下のコマンドを実行してスワップファイルを無効にします(/swapfile は swap ファイル)。

```
swapoff -v /swapfile
```

2. /etc/fstab ファイルから該当するエントリーを削除します。

3.
実際のファイルを削除します。

```
rm /swapfile
```

7.4. SWAP 領域の移動

スワップスペースをある場所から別の場所に移動するには、スワップスペースを削除する手順を実行してから、スワップスペースを追加する手順を実行します。

第8章 ディスクストレージの管理

8.1. PARTEDを使用した標準パーティション

ユーティリティーの parted により、ユーザーは次のことができます。

- 既存パーティションテーブルの表示
- 既存パーティションのサイズ変更
- 空き領域または他のハードドライブからの、パーティションの追加

システムのディスク領域の使用量を表示するか、ディスク領域の使用量を監視する場合は、「ファイルシステム」 を参照してください。

デフォルトでは、Red Hat Enterprise Linux のインストール時に parted パッケージが含まれています。parted を開始するには、root としてログインし、シェルプロンプトで parted */dev/sda* コマンドを入力します (*/dev/sda* は、設定するドライブのデバイス名です)。

パーティションの削除またはサイズ変更を行う場合は、パーティションが存在するデバイスが使用中でない必要があります。使用中のデバイスに新しいパーティションを作成することは可能ですが、推奨されません。

デバイスを使用しない場合は、そのデバイスのパーティションはどれもマウントできず、そのデバイスのスワップ領域も有効にできません。

また、カーネルが変更を正しく認識しない可能性があるため、使用中はパーティションテーブルを変更しないでください。パーティションテーブルがマウントされたパーティションの実際の状態と一致しない場合、情報が間違ったパーティションに書き込まれ、データが失われたり上書きされたりする可能性があります。

これを実現する最も簡単な方法は、システムをレスキューモードで起動することです。ファイルシ

ステムをマウントするように指示されたら、スキップ を選択します。

または、ドライブに使用中のパーティション (ファイルシステムがアンマウントされないように使用またはロックしているシステムプロセス) がない場合、umount コマンドでパーティションをアンマウントし、swapoff コマンドで、ハードドライブのすべてのスワップ領域を無効にできます。

表8.1「parted コマンド」 には、一般的に使用される parted コマンドのリストが含まれています。以下のセクションでは、これらのコマンドと引数の一部について詳しく説明します。

表8.1 parted コマンド

コマンド	説明
check *minor-num*	ファイルシステムの簡単なチェックを実行します。
cp *from to*	ファイルシステムをあるパーティションから別のパーティションにコピーします。*from* と *to* はパーティションのマイナー番号です。
help	利用可能なコマンドの一覧を表示します。
mklabel *label*	パーティションテーブル用のディスクラベルを作成します。
mkfs *minor-num file-system-type*	タイプ *file-system-type* のファイルシステムを作成します。
mkpart *part-type fs-type start-mb end-mb*	新しいファイルシステムを作成せずに、パーティションを作成します。
mkpartfs *part-type fs-type start-mb end-mb*	パーティションを作成し、指定されたファイルシステムを作成します。
move *minor-num start-mb end-mb*	パーティションを移動します。
name *minor-num name*	Mac と PC98 のディスクラベル用のみのパーティションに名前を付けます。
print	パーティションテーブルを表示します。
quit	parted を終了します。

コマンド	説明
rescue *start-mb end-mb*	*start-mb* から *end-mb* へ、消失したパーティションを復旧します。
resize *minor-num start-mb end-mb*	パーティションのサイズを *start-mb* から *end-mb* に変更します。
rm *minor-num*	パーティションを削除します。
select *device*	設定する別のデバイスを選択します。
set *minor-num flag state*	パーティションにフラグを設定します。*state* はオンまたはオフのいずれかになります。
toggle [*NUMBER* [*FLAG*]	パーティション *NUMBER* 上の *FLAG* の状態を切り替えます。
unit *UNIT*	デフォルトのユニットを *UNIT* に設定します。

8.1.1. パーティションテーブルの表示

parted を開始した後、コマンド print を使用してパーティションテーブルを表示します。以下のようなテーブルが表示されます。

```
Model: ATA ST3160812AS (scsi)
Disk /dev/sda: 160GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos

Number  Start   End    Size    Type      File system  Flags
 1      32.3kB  107MB  107MB   primary   ext3         boot
 2      107MB   105GB  105GB   primary   ext3
 3      105GB   107GB  2147MB  primary   linux-swap
 4      107GB   160GB  52.9GB  extended  root
 5      107GB   133GB  26.2GB  logical   ext3
 6      133GB   133GB  107MB   logical   ext3
 7      133GB   160GB  26.6GB  logical               lvm
```

1 行目にはディスクのタイプ、製造元、モデル番号、インターフェイスが含まれ、2 行目にはディスクラベルのタイプが表示されます。4 行目より下の残りの出力は、パーティションテーブルを示しています。

パーティションテーブルでは、マイナー番号はパーティション番号です。たとえば、マイナー番

号 1 のパーティションは、/dev/sda1 に対応します。Start および End の値はメガバイト単位です。有効な タイプ は、metadata、free、primary、extended、または logical です。Filesystem はファイルシステムタイプで、次のいずれかになります。

- ext2
- ext3
- fat16
- fat32
- hfs
- jfs
- linux-swap
- ntfs
- reiserfs
- hp-ufs
- sun-ufs
- xfs

デバイスの Filesystem の値が表示されない場合は、ファイルシステムのタイプが不明であることを意味します。

Flags は、パーティションに設定したフラグを一覧表示しています。利用可能なフラグは、boot、root、swap、hidden、raid、lvm、または lba です。

ヒント

parted を再起動せずに別のデバイスを選択するには、select コマンドに続けてデバイス名 (たとえば、/dev/sda) を使用します。そうすることで、デバイスのパーティションテーブルの表示と設定を行うことができます。

8.1.2. パーティションの作成

警告

使用中のデバイスに、パーティションを作成しないようにしてください。

パーティションを作成する前に、レスキューモードで起動します (または、デバイス上のパーティションをアンマウントして、デバイス上の swap 領域をすべてオフにします)。

parted を起動します。ここで、/dev/*sda* は、パーティションを作成するデバイスです。

```
parted /dev/sda
```

現在のパーティションテーブルを表示し、十分な空き領域があるかどうかを確認します。

```
print
```

十分な空き容量がない場合は、既存のパーティションのサイズを変更できます。詳細は、「パーティションのサイズ変更」 を参照してください。

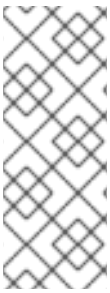
8.1.2.1. パーティションの作成

パーティションテーブルから、新しいパーティションの開始点と終了点、およびパーティションのタイプを決定します。プライマリーパーティションは、1 つのデバイス上に 4 つまで保有できます (この場合は拡張パーティションは含みません)。パーティションが 5 つ以上必要な場合は、プライマリー

パーティションを 3 つ、拡張パーティションを 1 つにし、その拡張パーティションの中に複数の論理パーティションを追加します。ディスクパーティションの概要について『は、『Red Hat Enterprise Linux インストールガイド』 の付録 ディスクパーティションの概要』 を参照してください。

たとえば、ハードドライブの 1024 メガバイトから 2048 メガバイトに ext3 ファイルシステムのプライマリーパーティションを作成するには、以下のコマンドを入力します。

```
mkpart primary ext3 1024 2048
```



ヒント

代わりに mkpartfs コマンドを使用すると、パーティションが作成されてからファイルシステムが作成されます。ただし、parted では、ext3 ファイルシステムの作成に対応していません。そのため、ext3 ファイルシステムを作成する場合は、mkpart を使用して、後述のように mkfs コマンドを実行してファイルシステムを作成します。

Enter を押すと変更が反映されるため、押す前に再度確認してください。

パーティションを作成したら、print コマンドを使用して、パーティションが正しいパーティションタイプ、ファイルシステムタイプ、およびサイズでパーティションテーブルにあることを確認します。また、ラベルを付けられるように、新しいパーティションのマイナー番号も覚えておいてください。また、の出力も表示されるはずです。

```
cat /proc/partitions
```

カーネルが新しいパーティションを認識するようにします。

8.1.2.2. パーティションのフォーマット

パーティションにはまだファイルシステムがありません。ファイルシステムを作成します。

```
mkfs -t ext3 /dev/sda6
```

WARNING

パーティションをフォーマットすると、そのパーティションに現存するすべてのデータが永久に抹消されます。

8.1.2.3. パーティションのラベル付け

次に、パーティションに ラベルを付けます。たとえば、新しいパーティションが /dev/sda6 で、/work のラベルを付けたいとします。

```
e2label /dev/sda6 /work
```

デフォルトでは、インストールプログラムはパーティションのマウントポイントをラベルとして使用して、ラベルが固有なものとなるようにします。ユーザーは使用するラベルを選択できます。

8.1.2.4. マウントポイントの作成

root として、マウントポイントを作成します。

```
mkdir /work
```

8.1.2.5. /etc/fstab への追加

root で、/etc/fstab ファイルを編集して新しいパーティションを含めます。新しい行は、以下のようになります。

```
LABEL=/work          /work                ext3    defaults        1 2
```

最初の列には LABEL= の後にパーティションを付けたラベルが含まれている必要があります。2 番目の列には、新しいパーティションのマウントポイントが含まれている必要があり、その次の列はファイルシステムタイプ (たとえば、ext3 または swap) である必要があります。フォーマットの詳細が必要な場合は、コマンド man fstab を使用して man ページを参照してください。

4 列目が defaults という単語の場合、パーティションは起動時にマウントされます。再起動せずにパーティションをマウントするには、root で次のコマンドを入力します。

-

```
mount /work
```

8.1.3. パーティションの削除

警告

パーティションが設定されているデバイスが使用中の場合は、削除しないでください。

パーティションを削除する前に、レスキューモードで起動します (または、デバイス上のパーティションをアンマウントして、デバイス上の swap 領域をすべてオフにします)。

parted を起動します。ここで、/dev/*sda* は、パーティションを削除するデバイスです。

```
parted /dev/sda
```

現在のパーティションテーブルを表示して、削除するパーティションのマイナー番号を確認します。

```
print
```

rm コマンドでパーティションを削除します。例えば、マイナー番号 3 のパーティションを削除するのは以下のコマンドです。

```
rm 3
```

変更は Enter を押すと変更が反映されるため、押す前にコマンドを再度確認してください。

パーティションを削除したら、print コマンドを使用して、パーティションテーブルから削除されていることを確認します。また、の出力も表示されるはずです。

```
cat /proc/partitions
```

カーネルがパーティションが削除されていることを知っていることを確認します。

最後の手順は、/etc/fstab ファイルからそれを削除することです。削除したパーティションを宣言している行を見つけ、ファイルから削除します。

8.1.4. パーティションのサイズ変更

警告

パーティションが設定されているデバイスが使用中の場合は、サイズを変更しないでください。

パーティションのサイズを変更する前に、レスキューモードで起動します (または、デバイス上のパーティションをアンマウントして、デバイス上の swap 領域をすべてオフにします)。

parted を起動します。ここで、/dev/*sda* は、パーティションのサイズを変更するデバイスです。

```
parted /dev/sda
```

現在のパーティションテーブルを表示して、サイズを変更するパーティションのマイナー番号と、パーティションの開始点と終了点を決定します。

```
print
```

パーティションのサイズを変更するには、resize コマンドの後に、パーティションのマイナー番号、、開始位置 (メガバイト単位)、終了位置 (メガバイト単位) を使用します。以下に例を示します。

```
resize 3 1024 2048
```

WARNING

デバイスの空き容量より大きなパーティションは作れません。

パーティションのサイズを変更した後、print コマンドを使用して、パーティションのサイズが正しく変更され、正しいパーティションタイプであり、正しいファイルシステムタイプであることを確認します。

システムを通常モードに再起動した後、df コマンドを使用して、パーティションがマウントされ、新しいサイズで認識されていることを確認します。

8.2. LVM パーティションの管理

以下のコマンドは、コマンドプロンプトで lvm help を発行すると確認できます。

表8.2 lvm コマンド

コマンド	説明
dumpconfig	アクティブな設定をダンプします。
形式	利用可能なメタデータ形式の一覧表示
help	ヘルプコマンドの表示
lvchange	論理ボリュームの属性を変更します。
lvcreate	論理ボリュームを作成します。
lvdisplay	論理ボリュームに関する情報を表示します。
lvextend	論理ボリュームに領域を追加
lvmchange	デバイスマッパーの使用により、このコマンドは非推奨になりました。

コマンド	説明
lvmdiskscan	物理ボリュームとして使用するデバイスを一覧表示します。
lvmsadc	アクティビティーデータの収集
lvmsar	アクティビティーレポートの作成
lvreduce	論理ボリュームのサイズを縮小する
lvremove	システムから論理ボリュームを削除する
lvrename	論理ボリュームの名前変更
lvresize	論理ボリュームのサイズ変更
lvs	論理ボリュームに関する情報を表示します。
lvscan	すべてのボリュームグループにある論理ボリュームの一覧を表示します。
pvchange	物理ボリュームの属性の変更
pvcreate	LVM で使用する物理ボリュームの初期化
pvdata	物理ボリュームのディスク上のメタデータを表示します。
pvdisplay	物理ボリュームのさまざまな属性を表示します。
pvmove	エクステントをある物理ボリュームから別の物理ボリュームに移動する
pvremove	物理ボリュームから LVM ラベルを削除します。
pvresize	ボリュームグループが使用する物理ボリュームのサイズ
pvs	物理ボリュームに関する情報を表示します。

コマンド	説明
pvscan	物理ボリュームの一覧を表示します。
segtypes	利用可能なセグメントタイプを一覧表示します。
vgcfgbackup	バックアップボリュームグループの設定
vgcfgrestore	ボリュームグループ設定の復元
vgchange	ボリュームグループ属性の変更
vgck	ボリュームグループの整合性の確認
vgconvert	ボリュームグループのメタデータ形式の変更
vgcreate	ボリュームグループの作成
vgdisplay	ボリュームグループ情報の表示
vgexport	システムからボリュームグループの登録を解除します。
vgextend	ボリュームグループへの物理ボリュームの追加
vgimport	エクスポートしたボリュームグループをシステムに登録します。
vgmerge	ボリュームグループのマージ
vgmknodes	/dev/ にボリュームグループデバイスの特別なファイルを作成します。
vgreduce	ボリュームグループからの物理ボリュームの削除
vgremove	ボリュームグループの削除
vgrename	ボリュームグループの名前変更
vgs	ボリュームグループに関する情報を表示します。

コマンド	説明
vgscan	**すべてのボリュームグループの検索**
vgsplit	**新しいボリュームグループへの物理ボリュームの移動**
version	**ソフトウェアおよびドライバーのバージョン情報を表示します。**

第9章 ディスククォータの実装

ディスク領域はディスククォータによって制限できます。ディスククォータは、ユーザーが過度のディスク領域を消費するか、パーティションが満杯になる前にシステム管理者に警告をします。

ディスククォータは、ユーザーグループにも個別のユーザーにも設定できます。これにより、ユーザーが参加しているプロジェクトに割り振られた領域 (プロジェクトごとに所有グループが存在すると想定) とは別に、ユーザー固有のファイル (電子メールなど) に割り振った領域を管理することが可能になります。

さらにクォータは、消費されるディスクブロックの数の制御だけでなく、inode (UNIX ファイルシステム内のファイルに関する情報を含むデータ構造) の数も制御するように設定できます。inode はファイル関連の情報を組み込むように使用されるため、これが作成されるファイルの数を制御することも可能にします。

ディスククォータを実装するには、quota RPM をインストールしておく必要があります。

注記

RPM パッケージのインストールに関する詳細は、パートII「パッケージ管理」 を参照してください。

9.1. ディスククォータの設定

ディスククォータを実装するには、以下の手順を行います。

1. /etc/fstab を修正することで、ファイルシステムごとのクォータを有効にします。
2. ファイルシステムを再マウントします。
3. クォータデータベースファイルを作成して、ディスク使用状況テーブルを生成します。

4. クォータポリシーを割り当てます。

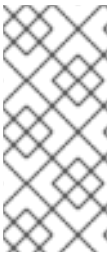
この各手順は、以下のセクションで詳しく説明します。

9.1.1. クォータの有効化

root として、テキストエディターを使用して、/etc/fstab ファイルを編集します。クォータを必要とするファイルシステムに usrquota オプションや grpquota オプションを追加します。

```
/dev/VolGroup00/LogVol00 /        ext3    defaults        1 1
LABEL=/boot             /boot     ext3    defaults        1 2
none                    /dev/pts  devpts  gid=5,mode=620  0 0
none                    /dev/shm  tmpfs   defaults        0 0
none                    /proc     proc    defaults        0 0
none                    /sys      sysfs   defaults        0 0
/dev/VolGroup00/LogVol02 /home    ext3    defaults,usrquota,grpquota  1 2
/dev/VolGroup00/LogVol01 swap     swap    defaults        0 0 . . .
```

この例では、/home ファイルシステムがユーザーとグループの両方のクォータを有効にしています。



備考

以下の例では、Red Hat Enterprise Linux のインストール時に別の/home パーティションが作成されたことを前提としています。root (/) パーティションは /etc/fstab ファイル内でクォータポリシーを設定するために使用できます。

9.1.2. ファイルシステムの再マウント

usrquota オプションや grpquota オプションを追加した後、fstab エントリーが変更された各ファイルシステムを再マウントします。ファイルシステムがどのプロセスでも使用されていない場合は、以下のいずれかの方法を使用します。

- umount コマンドを発行してから mount コマンドを実行してファイルシステムを再マウントします（さまざまなファイルシステムタイプをマウントおよびアンマウントするための特定の構文については、umount および mount の両方の man ページを参照してください）。

- mount -o remount < *file-system* > コマンド(< *file-system* > はファイルシステムの名前)を発行して、ファイルシステムを再マウントします。たとえば、/home ファイルシステムを

再マウントする場合、発行するコマンドは mount -o remount /home です。

ファイルシステムが現在使用中の場合、そのファイルシステムを再マウントする最も簡単な方法は、システムを再起動することです。

9.1.3. クォータデータベースファイルの作成

クォータが有効な各ファイルシステムを再マウントすると、システムはディスククォータを操作できるようになります。ただし、ファイルシステム自体がクォータに対応するようにするには、追加の設定が必要です。次のステップとして、quotacheck コマンドを実行します。

quotacheck コマンドは、クォータが有効なファイルシステムを検証し、現在のディスク使用状況のテーブルをファイルシステムごとに構築します。このテーブルは、ディスク使用状況のオペレーティングシステム用コピーを更新するのに使用されます。また、ファイルシステムのディスククォータが更新されます。

ファイルシステムにクォータファイル (aquota.user および aquota.group) を作成するには、quotacheck コマンドの -c オプションを使用します。たとえば、/home ファイルシステムでユーザーとグループのクォータが有効になっている場合は、/home ディレクトリーにファイルを作成します。

```
quotacheck -cug /home
```

-c オプションは、クォータが有効になっているファイルシステムごとにクォータファイルを作成することを指定し、-u オプションは、ユーザークォータをチェックすることを指定し、-g オプションは、グループクォータをチェックすることを指定します。

-u オプションまたは -g オプションのいずれも指定しない場合は、ユーザークォータファイルのみが作成されます。-g のみを指定すると、グループクォータファイルのみが作成されます。

ファイルの作成後、以下のコマンドを実行し、クォータを有効にしたファイルシステムごとの現在のディスク使用量のテーブルを生成します。

```
quotacheck -avug
```

使用されるオプションは次のとおりです。

- A: クォータが有効 な ローカルマウントのファイルシステムをすべてチェックします。
- v: クォータチェックの進行時に詳細なステータス情報を表示します。
- u - ユーザーディスククォータ情報の確認
- g - グループディスククォータ情報の確認

quotacheck の実行が終了すると、有効なクォータ (ユーザーやグループ) に対応するクォータファイルに、/home などの、クォータが有効になっているローカルにマウントされた各ファイルシステムのデータが取り込まれます。

9.1.4. ユーザーごとのクォータ割り当て

最後の手順は、edquota コマンドを使用したディスククォータ割り当てです。

ユーザーにクォータを設定するには、シェルプロンプトで、root で次のコマンドを実行します。

```
edquota username
```

クォータが必要な各ユーザーに対して、この手順を実行します。たとえば、クォータが /home パーティションの /etc/fstab (以下の例では /dev/VolGroup00/LogVol02) に対して有効であり、コマンド edquota testuser を実行すると、システムでデフォルトとして設定されたエディターで以下のような出力が表示されます。

```
Disk quotas for user testuser (uid 501):
Filesystem                   blocks     soft     hard    inodes   soft   hard
/dev/VolGroup00/LogVol02  440436        0        0     37418      0      0
```

備考

EDITOR 環境変数により定義されたテキストエディターは、edquota により使用されます。エディターを変更するには、~/.bash_profile ファイルの EDITOR 環境変数を、使用するエディターのフルパスに設定します。

最初の列は、クォータが有効になっているファイルシステムの名前です。2 列目には、ユーザーが

現在使用しているブロック数が示されます。その次の 2 列は、ファイルシステム上のユーザーのソフトブロック制限およびハードブロック制限を設定するのに使用されます。inodes 列は、ユーザーが現在使用している inode の数を示します。最後の 2 列は、ファイルシステムのユーザーに対するソフトおよびハードの inode 制限を設定するのに使用されます。

ハードブロック制限は、ユーザーまたはグループが使用できる最大ディスク容量 (絶対値) です。この制限に達すると、それ以上のディスク領域は使用できなくなります。

ソフトブロック制限は、使用可能な最大ディスク容量を定義します。ただし、ハード制限とは異なり、ソフト制限は一定時間超過する可能性があります。この時間は *猶予期間* として知られています。猶予期間の単位は、秒、分、時間、日、週、または月で表されます。

いずれかの値が 0 に設定されていると、その制限は設定されません。テキストエディターで必要な制限に変更します。以下に例を示します。

```
Disk quotas for user testuser (uid 501):
Filesystem              blocks    soft    hard   inodes   soft   hard
/dev/VolGroup00/LogVol02  440436   500000   550000    37418      0      0
```

ユーザーのクォータが設定されていることを確認するには、以下のコマンドを使用します。

```
quota testuser
```

9.1.5. グループごとのクォータ割り当て

クォータは、グループごとに割り当てることもできます。たとえば、devel グループのグループクォータを設定するには（グループはグループクォータを設定する前に存在している必要があります）、次のコマンドを使用します。

```
edquota -g devel
```

このコマンドにより、グループの既存クォータがテキストエディターに表示されます。

```
Disk quotas for group devel (gid 505):
Filesystem              blocks    soft    hard    inodes    soft    hard
/dev/VolGroup00/LogVol02  440400       0       0     37418       0       0
```

この制限を変更して、ファイルを保存します。

グループクォータが設定されたことを確認するには、次のコマンドを使用します。

```
quota -g devel
```

9.1.6. ソフト制限の猶予期間の設定

指定されたクォータ(inode またはブロックおよびユーザーまたはグループのいずれか)にソフト制限が設定されている場合、またはソフト制限を超過できる期間、またはソフト制限を超過できる時間を設定するには、コマンドで設定する必要があります。

```
edquota -t
```

他の edquota コマンドは特定のユーザーまたはグループのクォータで動作しますが、-t オプションはクォータが有効になっているすべてのファイルシステムで動作します。

9.2. ディスククォータの管理

クォータが実装されている場合には、若干の保守が必要となります — 大半は、クォータの超過監視および精度確認という形となります。

当然ながら、ユーザーが繰り返しクォータを超過したり、常にソフトリミットに達している場合には、ユーザーのタイプや、ユーザーの作業にディスク容量が及ぼす影響の度合に応じて、システム管理者には 2 つの選択肢があります。管理者は、ユーザーが使用するディスク領域を節約する方法をわかるようにするか、ユーザーのディスククォータを拡大するかのいずれかを行うことができます。

9.2.1. 有効化と無効化

クォータはゼロに設定することなく、無効にすることができます。すべてのユーザーとグループのクォータをオフにするには、以下のコマンドを使用します。

```
quotaoff -vaug
```

-u オプションまたは -g オプションのいずれも指定しない場合は、ユーザークォータのみが無効になります。-g のみを指定すると、グループクォータのみが無効になります。-v スイッチにより、コマンドの実行時に詳細なステータス情報が表示されます。

クォータを再度有効にするには、同じオプションを指定して quotaon コマンドを使用します。

たとえば、すべてのファイルシステムに対してユーザーとグループのクォータを有効にするには、次のコマンドを使用します。

```
quotaon -vaug
```

/home などの特定のファイルシステムにクォータを有効にするには、以下のコマンドを使用します。

```
quotaon -vug /home
```

-u オプションまたは -g オプションのいずれも指定しない場合は、ユーザークォータのみが有効になります。-g のみが指定されている場合は、グループのクォータのみが有効になります。

9.2.2. ディスククォータに関するレポート

ディスク使用状況のレポートを作成するには、repquota ユーティリティーの実行が必要になります。たとえば、コマンド repquota /home により、以下のような出力が表示されます。

```
*** Report for user quotas on device /dev/mapper/VolGroup00-LogVol02
Block grace time: 7days; Inode grace time: 7days
                        Block limits                File limits
User            used    soft    hard  grace    used  soft  hard  grace
----------------------------------------------------------------------
root      --      36       0       0              4     0     0
kristin   --     540       0       0            125     0     0
testuser  --  440400  500000  550000          37418     0     0
```

クォータが有効なすべてのファイルシステム (オプション -a) のディスク使用状況レポートを表示するには、次のコマンドを使用します。

```
repquota -a
```

レポートは読みやすいですが、いくつか説明しておくべき点があります。各ユーザーの後に表示される--を利用すると、ブロック制限または inode 制限を超えたかどうかをすばやく判断できます。いずれかのソフト制限を超えると、対応する - の代わりに + が表示されます。最初の - はブロックの制限を表し、2 つ目は inode の制限を表します。

grace 列は通常空白です。ソフト制限が超過した場合、その列には猶予期間に残り時間量に相当する時間指定が含まれます。猶予期間が過ぎると、代わりに none が表示されます。

9.2.3. クォータの精度維持

ファイルシステムが正常にアンマウントされない場合（システムクラッシュなどが原因で）、quotacheck を実行する必要があります。ただし、システムがクラッシュしていない場合でも、quotacheck は定期的に実行できます。quotacheck を定期的に実行する安全な方法には、以下が含まれます。

次回の再起動時に quotacheck を確実に実行する

ほとんどのシステムで最適な方法

この方法は、定期的に再起動する (ビジーな) 複数ユーザーシステムに最も適しています。

root で、シェルスクリプトを touch /forcequotacheck コマンドを含む /etc/cron.daily/ または /etc/cron.weekly/ ディレクトリーに置くか、crontab -e コマンドを使ってスケジュールを設定します。このスクリプトは root ディレクトリーに空の forcequotacheck ファイルを作成するため、起動時にシステムの init スクリプトがこれを検索します。このディレクトリーが検出されると、init スクリプトは quotacheck を実行します。その後、init スクリプトは /forcequotacheck ファイルを削除します。このように、cron でこのファイルが定期的に作成されるようにスケジュールすることにより、次回の再起動時に quotacheck を確実に実行することができます。

cron の設定に関する詳細は、39章*自動タスク* を参照してください。

シングルユーザーモードで quotacheck を実行

quotacheck を安全に実行する別の方法として、クォータファイルのデータ破損の可能性を防ぐために、システムをシングルユーザーモードで再起動（再）する方法があります。

```
~]# quotaoff -vaug /<file_system>
~]# quotacheck -vaug /<file_system>
~]# quotaon -vaug /<file_system>
```

実行中のシステムで quotacheck を実行

必要な場合には、いずれのユーザーもログインしておらず、チェックされているファイルシステムに開いているファイルがない状態のマシン上で quotacheck を実行することができます。quotacheck -vaug < *file_system* > コマンドを実行します。このコマンドは、quotacheck が指定された < *file_system* > を読み取り専用として再マウントできない場合に失敗します。チェックの後には、ファイルシステムは読み込み/書き込みとして再マウントされることに注意してください。

ライブファイルシステムで QUOTACHECK を実行しない

読み込み/書き込みでマウントされているライブのファイルシステム上での quotacheck の実行は、quota ファイルが破損する可能性があるため、推奨されません。

cron の設定に関する詳細は、39章*自動タスク* を参照してください。

9.3. 関連情報

ディスククォータの詳細は、以下のリソースを参照してください。

9.3.1. インストールされているドキュメント

- quotacheck、edquota、repquota、quota on、および quotaoff の man ページ

9.3.2. 関連書籍

- 『Red Hat Enterprise Linux Introduction to System Administration』; Red Hat, Inc. - http://www.redhat.com/docs/ および ドキュメント CD で利用可能なマニュアルには、新しい Red Hat Enterprise Linux システム管理者向けのストレージ管理（ディスククォータを含む）に関する背景情報が含まれています。

第10章 アクセス制御リスト

ファイルとディレクトリーには、ファイルの所有者、そのファイルに関連したグループ、およびシステムを使用する他のすべてのユーザーの権限セットが設定されます。しかし、これらの権限には制限があります。たとえば、ユーザーごとに異なる権限を設定することはできません。そのため *アクセス制御リスト* (ACL) が実装されています。

Red Hat Enterprise Linux 5 カーネルは、ext3 ファイルシステムおよび NFS でエクスポートされるファイルシステムの ACL サポートを提供します。ACL は、Samba 経由でアクセスする ext3 ファイルシステムでも認識されます。

ACL の実装には、カーネルでのサポートと acl パッケージが必要になります。このパッケージには、ACL 情報の追加、修正、削除および、取得のためのユーティリティーが同梱されています。

cp コマンドと mv コマンドは、ファイルとディレクトリーに関連するすべての ACL のコピーまたは移動を実行します。

10.1. ファイルシステムのマウント

ファイルやディレクトリー用に ACL を使用する前に、そのファイルまたはディレクトリーのパーティションを ACL サポートでマウントする必要があります。ローカルの ext3 ファイルシステムの場合は、以下のコマンドでマウントできます。

```
mount -t ext3 -o acl <device-name> <partition>
```

以下に例を示します。

```
mount -t ext3 -o acl /dev/VolGroup00/LogVol02 /work
```

または、パーティションが /etc/fstab ファイルにリストされている場合は、パーティションのエントリーに acl オプションを含めることができます。

```
LABEL=/work     /work      ext3    acl       1 2
```

Samba 経由で ext3 ファイルシステムにアクセスし、そのアクセスに対して ACL が有効になっている場合は、ACL が認識されます。これは、--with-acl-support オプションでコンパイルされているためです。Samba 共有のアクセス時またはマウント時に特別なフラグは必要ありません。

10.1.1. NFS

デフォルトでは、NFS サーバーでエクスポートされているファイルシステムが ACL をサポートし、NFS クライアントが ACL を読み込める場合は、クライアントシステムで ACL が使用されています。

サーバーの設定時に NFS 共有上の ACL を無効にするには、/etc/exports ファイルに no_acl オプションを追加します。クライアントに NFS 共有をマウントする際に ACL を無効にするには、コマンドラインまたは /etc/fstab ファイルで no_acl オプションでマウントします。

10.2. アクセス ACL の設定

ACL には、*アクセス ACL* と *デフォルト ACL* と 2 つのタイプがあります。アクセス ACL は、特定のファイルまたはディレクトリーに対するアクセス制御リストです。デフォルト ACL は、ディレクトリーにのみ適用されます。ディレクトリー内のファイルにアクセス ACL が設定されていない場合は、そのディレクトリーにデフォルト ACL のルールが適用されます。デフォルト ACL は任意です。

ACL は以下のように設定できます。

1. 各ユーザー
2. 各グループ
3. 実効権マスクを使用して
4. ファイルのユーザーグループに属さないユーザーに対して

setfacl ユーティリティーは、ファイルとディレクトリー用の ACL を設定します。-m オプションを使用して、ファイルまたはディレクトリーの ACL を追加または変更します。

```
setfacl -m <rules> <files>
```

*ルール(<rules>)*は以下の形式で指定する必要があります。複数のルールをコンマで区切って同じコマンドに指定することもできます。

u:*<uid>*:*<perms>*

ユーザーにアクセス ACL を設定します。ユーザー名または UID を指定できます。システムで有効な任意のユーザーを指定できます。

g:*<gid>*:*<perms>*

グループにアクセス ACL を設定します。グループ名または GID を指定できます。システムで有効な任意のグループを指定できます。

m:*<perms>*

実効権マスクを設定します。このマスクは、所有グループの全権限と、ユーザーおよびグループの全エントリーを結合したものです。

o:*<perms>*

ファイルのグループに属さないユーザーにアクセス ACL を設定します。

空白は無視されます。パーミッション(&*lt;perms*>)は、読み取り、書き込み、および実行の場合は r、w、x の文字の組み合わせでなければなりません。

ファイルまたはディレクトリーにすでに ACL が設定されている状態で、setfacl コマンドを使用した場合は、設定するルールが既存の ACL に追加されるか、既存のルールが修正されます。

たとえば、ユーザー andrius に読み取りと書き込みの権限を付与するには以下を実行します。

```
setfacl -m u:andrius:rw /project/somefile
```

ユーザー、グループ、またはその他のユーザーに対するパーミッションをすべて削除するには、-x オプションを使用して権限を指定しません。

```
setfacl -x <rules> <files>
```

たとえば、UID 500 のユーザーからすべての権限を削除するには以下を実行します。

```
setfacl -x u:500 /project/somefile
```

10.3. デフォルト ACL の設定

デフォルトの ACL を設定するには、ルールの前に d: を追加して、ファイル名ではなくディレクトリーを指定します。

たとえば、/share/ ディレクトリーにデフォルト ACL を設定し、ユーザーグループに属さないユーザーの読み取りと実行を設定するには、以下のコマンドを実行します (これにより、個別ファイルのアクセス ACL が上書きされます)。

```
setfacl -m d:o:rx /share
```

10.4. ACL の取り込み

ファイルまたはディレクトリーに設定されている既存の ACL を確認するには、getfacl コマンドを使用します。以下の例では、getfacl でファイルの既存の ACL を確認します。

```
getfacl home/john/picture.png
```

上記のコマンドは、次のような出力を返します。

```
# file: home/john/picture.png
# owner: john
# group: john
user::rw-
group::r--
other::r--
```

ディレクトリーにデフォルト ACL が指定されている場合は、以下のようにデフォルト ACL も表示されます。

```
[john@main /]$ getfacl home/sales/
# file: home/sales/
# owner: john
# group: john
user::rw-
user:barryg:r--
group::r--
mask::r--
```

```
other::r--
default:user::rwx
default:user:john:rwx
default:group::r-x
default:mask::rwx
default:other::r-x
```

10.5. ACL が設定されているファイルシステムのアーカイブ作成

WARNING

tar コマンドおよび dump コマンドは ACL をバックアップし ませ ん。

star ユーティリティーは、ファイルのアーカイブ生成に使用される点で tar ユーティリティーと似ています。しかし、一部のオプションは異なります。最も一般的に使用されるオプションの一覧は 表10.1「star のコマンドラインオプション」 を参照してください。すべての利用可能なオプションは、star の man ページを参照してください。このユーティリティーを使用するには star パッケージが必要になります。

表10.1 star のコマンドラインオプション

オプション	説明
-c	アーカイブファイルを作成します。
-n	ファイルを抽出しません。-x と併用すると、ファイルが行う抽出を表示します。
-r	アーカイブ内のファイルを入れ替えます。パスとファイル名が同じファイルが置き換えられ、アーカイブファイルの末尾に書き込まれます。
-t	アーカイブファイルのコンテンツを表示します。
-u	アーカイブファイルを更新します。アーカイブにファイルが存在しない場合や、アーカイブ内の同じ名前のファイルよりも新しい場合は、ファイルはアーカイブの最後に書き込まれます。このオプションは、アーカイブがファイルであるか、またはバックスペースがあるブロックされていないテープの場合にのみ機能します。

オプション	説明
-x	アーカイブからファイルを抽出します。**-U** と併用すると、アーカイブ内のファイルがファイルシステムにあるファイルよりも古い場合、そのファイルは抽出されません。
-help	最も重要なオプションを表示します。
-xhelp	最も重要ではないオプションを表示します。
-/	アーカイブからファイルを抽出する際に、ファイル名から先頭のスラッシュを削除します。デフォルトでは、ファイルの抽出時にストライプ化されます。
-acl	作成時または抽出時に、ファイルおよびディレクトリーに関連する **ACL** をアーカイブまたは復元します。

10.6. 旧システムとの互換性

指定したファイルシステムのいずかのファイルに **ACL** が設定されている場合、そのファイルシステムには **ext_attr** 属性があります。この属性は、以下のコマンドを使用すると確認できます。

```
tune2fs -l <filesystem-device>
```

ext_attr 属性を持つファイルシステムは古いカーネルでマウントできますが、それらのカーネルは設定されている **ACL** を強制しません。

バージョン **1.22** 以降の **e2fsprogs** パッケージ **(Red Hat Enterprise Linux 2.1** および **4** のバージョンも含む**)** に含まれている **e2fsck** ユーティリティーのバージョンは、**ext_attr** 属性を使用してファイルシステムを確認できます。古いバージョンではこの確認が拒否されます。

10.7. 関連情報

詳細は、以下のリソースを参照してください。

10.7.1. インストールされているドキュメント

- **ACL** の **man** ページ **- ACL** の説明

- gedit の man ページ：ファイルアクセス制御リストの取得方法
- setfacl の man ページ：ファイルアクセス制御リストの設定方法
- star の man ページ： star ユーティリティーとそのオプションの詳細を説明します。

10.7.2. 便利な Web サイト

- http://acl.bestbits.at/ - ACL の Web サイト

第11章 LVM (論理ボリュームマネージャー)

11.1. LVM とは

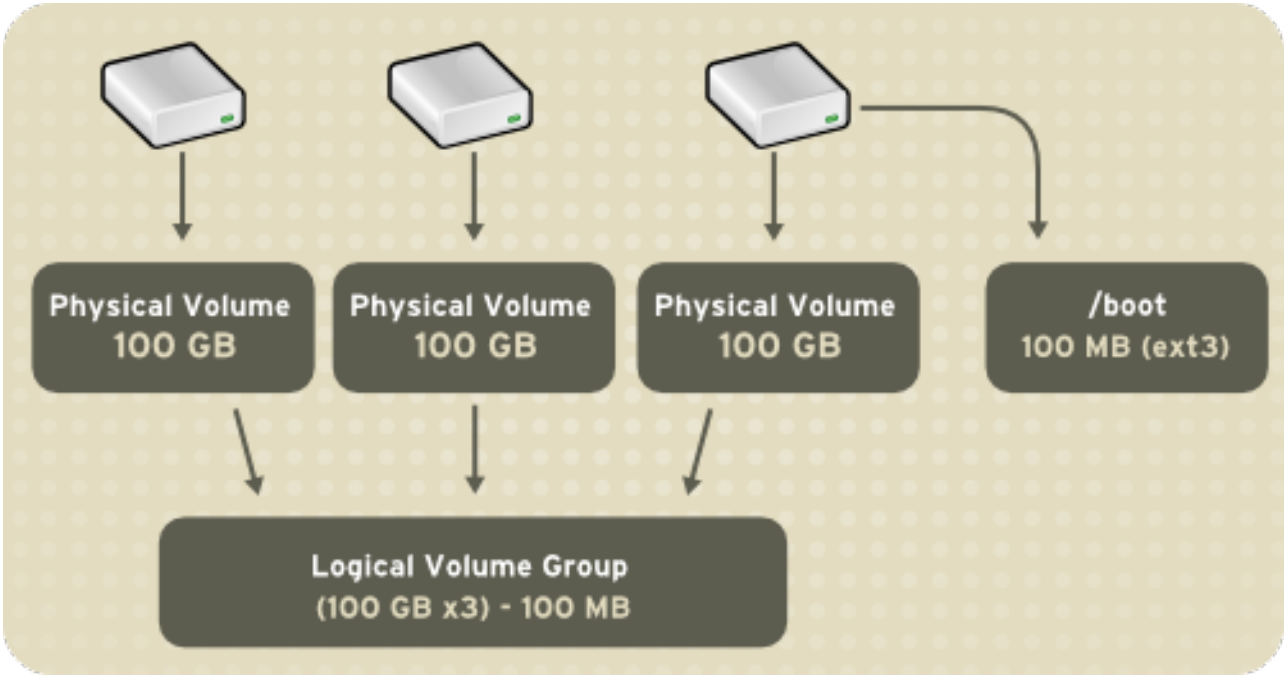
LVM は、ディスクの割り当て、論理ボリュームのストライピング、ミラーリング、サイズ変更などを含む、論理ボリュームを管理するためのツールです。

LVM では、ハードドライブまたはハードドライブのセットが 1 つ以上の *物理ボリューム* に割り当てられます。LVM 物理ボリュームは、2 つ以上のディスクにまたがる可能性のある他のブロックデバイスに配置できます。

/boot パーティションを除き、物理 *ボリュームは論理ボリューム* に結合されます。/boot パーティションは、ブートローダーが読み取ることができないため、論理ボリュームグループでは配置できません。root (/)パーティションが論理ボリュームにある場合は、ボリュームグループの一部ではない別の /boot パーティションを作成します。

物理ボリュームは複数のドライブにまたがることができないため、複数のドライブにまたがるには、ドライブごとに 1 つ以上の物理ボリュームを作成します。

図11.1 論理ボリューム

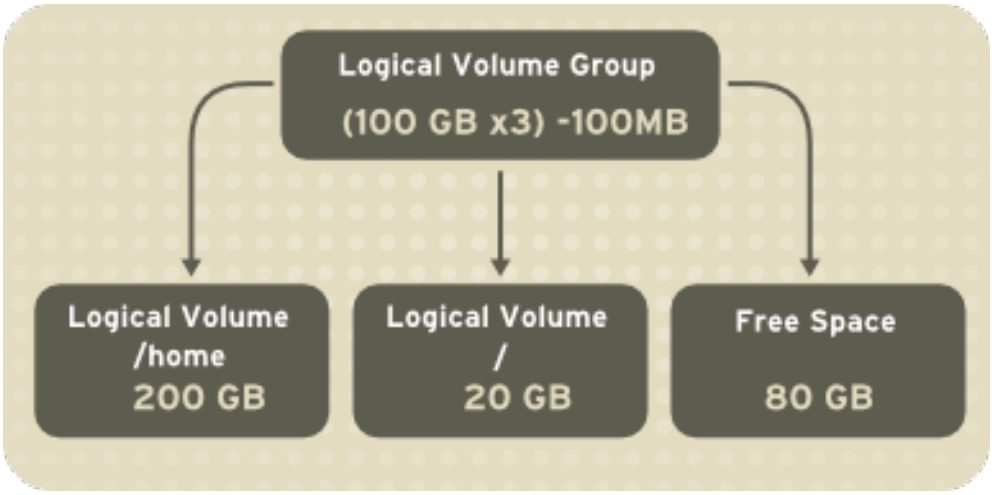


[D]

ボリュームグループは *論理ボリューム* に分割され、/home や / などのマウントポイント、ext2 や ext3 などのファイルシステムタイプが割り当てられます。パーティションが最大容量に達すると、ボリュームグループの空き領域を論理ボリュームに追加して、パーティションのサイズを増やすことがで

きます。新しいハードドライブをシステムに追加すると、それをボリュームグループに追加したり、論理ボリュームであるパーティションのサイズを大きくしたりできます。

図11.2 論理ボリューム



[D]

一方、システムが ext3 ファイルシステムでパーティション設定されている場合、ハードドライブは定義されたサイズのパーティションに分割されます。パーティションがいっぱいになると、パーティションのサイズを拡張するのは簡単ではありません。パーティションを別のハードドライブに移動した場合でも、元のハードドライブのスペースを別のパーティションとして再割り当てするか、使用しないようにする必要があります。

インストールプロセス中に LVM を設定する方法は、「LVM 設定」 を参照してください。

11.1.1. LVM2 とは

LVM バージョン 2 (LVM2)は、2.6 カーネルに含まれるデバイスマッパードライバーを使用する Red Hat Enterprise Linux 5 のデフォルトです。LVM2 は、2.4 カーネルを実行しているバージョンの Red Hat Enterprise Linux からアップグレードできます。

11.2. LVM 設定

LVM は、グラフィカルインストールプロセス、テキストベースのインストールプロセス、またはキックスタートインストール中に設定できます。system-config-lvm ユーティリティーを使用して、インストール後に独自の LVM 設定を作成できます。次の 2 つのセクションでは、インストール時に ディスク Druid を使用してこのタスクを完了することに重点を置いています。3 番目のセクションでは、LVM ユーティリティー(system-config-lvm)を紹介します。これにより、X ウィンドウまたはグラフィカルに LVM ボリュームを管理できます。

LVM について理解するには、「LVM とは」 を最初にお読みください。LVM の設定に必要な手順の概要は次のとおりです。

- ハードドライブからの *物理ボリューム* の作成

- 物理 *ボリュームからのボリュームグループ*の作成

- ボリュームグループから *論理ボリューム* を作成し、論理ボリュームのマウントポイントを割り当てます。

以下の例では、2 つの 9.1 GB SCSI ドライブ(/dev/sda および /dev/sdb)が使用されます。インストール時に、関連付けられた論理ボリュームと共に単一の LVM ボリュームグループを使用して簡単な設定を作成する方法が詳述されています。

11.3. 自動パーティション設定

ディスクパーティションの設定 画面で、選択し たドライブで linux パーティションの削除 を選択し、プルダウン一覧からデフォルトのレイアウトを作成 します。

Red Hat Enterprise Linux では、LVM がディスクのパーティション設定におけるデフォルトの方法です。LVM を実装していない場合、または RAID のパーティション設定が必要な場合は、Disk Druid を使用した手動ディスクパーティション設定が必要です。

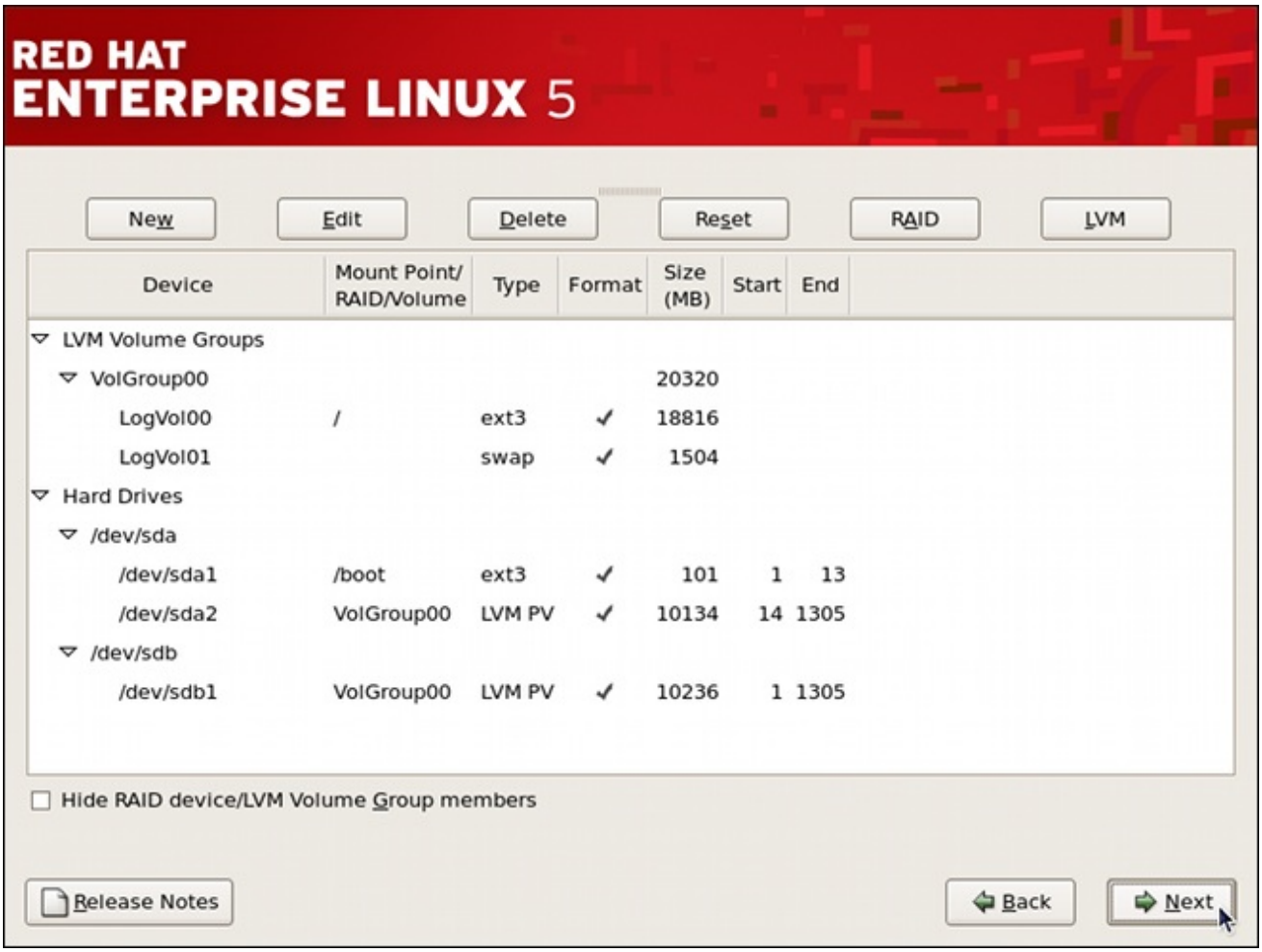
以下のプロパティーは、自動的に作成された設定を設定します。

- /boot パーティションは、LVM 以外のパーティションにあります。以下の例では、最初のドライブ(/dev/sda1)の最初のパーティションになります。ブート可能なパーティションは、LVM 論理ボリュームには存在 できません。

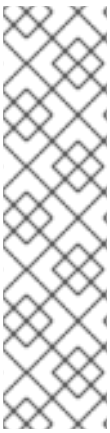
- 1 つの LVM ボリュームグループ(VolGroup00)が作成されます。これは、選択したすべてのドライブと、残りのすべての領域にまたがるものです。以下の例では、最初のドライブの残り(/dev/sda2)と、2 番目のドライブ全体(/dev/sdb1)がボリュームグループに割り当てられます。

- 2 つの LVM 論理ボリューム(LogVol00 および LogVol01)が、新たに作成したスパンのボリュームグループから作成されます。以下の例では、推奨されるスワップ領域は自動的に計算され LogVol01 に割り当てられ、残りはルートファイルシステム LogVol00 に割り当てられます。

図11.3 2 つの SCSI ドライブを使用した自動 LVM 設定



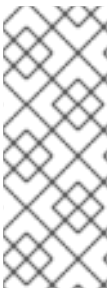
[D]



注記

クォータを有効にすると、/home や /var などの他のマウントポイントを含めるように自動設定を変更することが推奨されます。これにより、各ファイルシステムに独自の独立したクォータ設定制限を持たせることができます。

ほとんどの場合、デフォルトの自動 LVM パーティションで十分ですが、高度な実装ではパーティションテーブルの変更または手動設定が必要になる場合があります。



注記

今後のメモリーアップグレードが予想される場合は、ボリュームグループにいくつかの空き領域を残しておくと、システム上のスワップ領域の論理ボリュームを簡単に拡張できるようになります。この場合、LVM 自動設定を変更して、将来的な増加のために利用可能な領域を確保する必要があります。

11.4. 手動 LVM パーティション設定

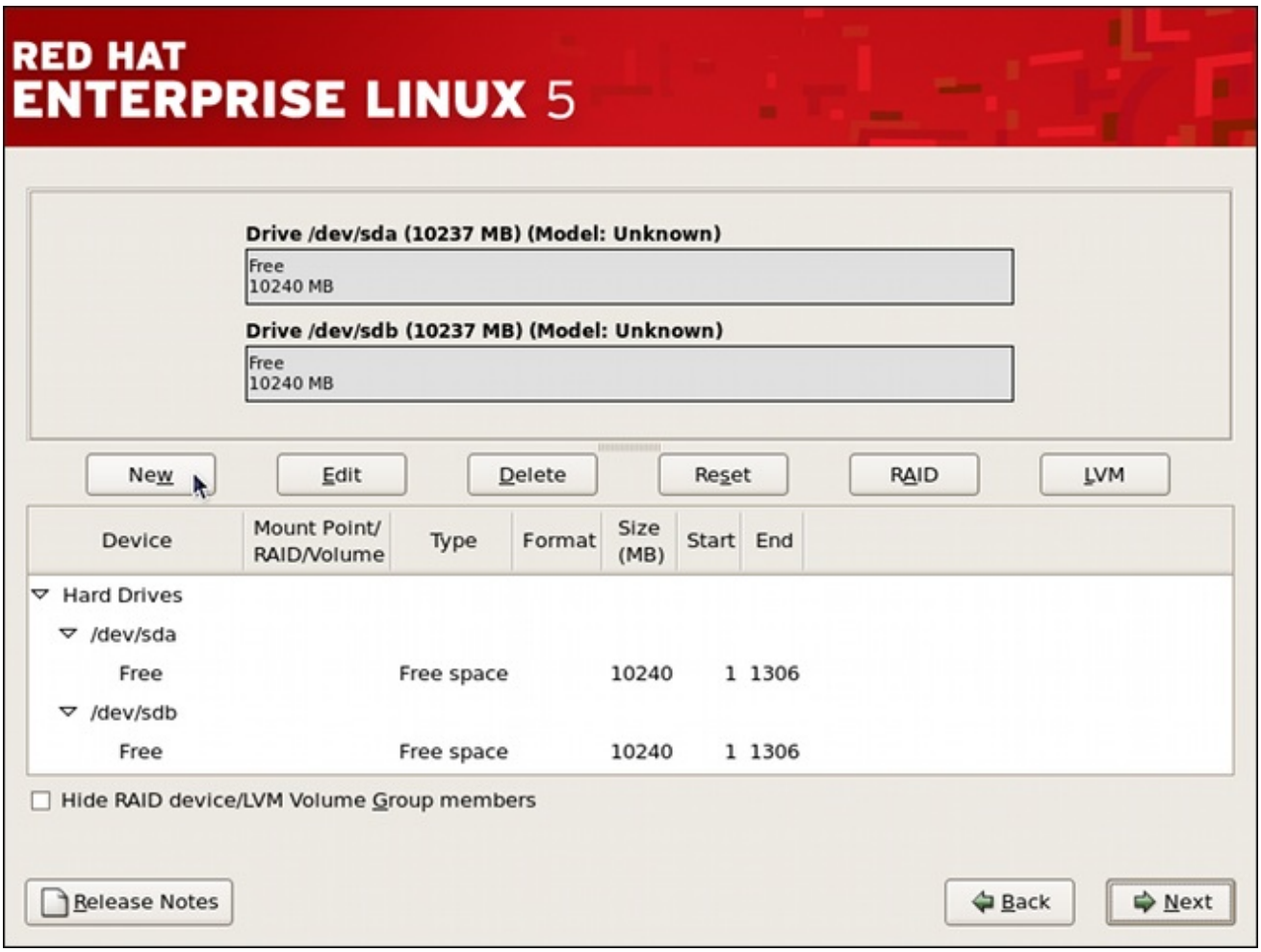
次のセクションでは、Red Hat Enterprise Linux 用に LVM を手動で設定する方法を説明します。LVM でシステムを手動で設定する方法は多数あるため、「自動パーティション設定」 で行うデフォルト設定は以下のようになります。

ディスクパーティション設定 画面で、プルダウン一覧から カスタムレイアウトの作成 を選択し、画面の右下にある Next ボタンをクリックします。

11.4.1. /boot パーティションの作成

一般的な状況では、ディスクドライブが新規またはフォーマット済みである。以下の図は、図11.4「空白ドライブを 2 つ（準備完了）」 で、パーティションが設定されていない raw デバイスとして両方のドライブを示しています。

図11.4 空白ドライブを 2 つ（準備完了）



[D]

WARNING

GRUB ブートローダーは読み取れないため、/boot パーティションは LVM ボリュームには存在できません。

1. New を選択します。

2. Mount Point プルダウンメニューから /boot を選択します。

3. File System Type プルダウンメニューから ext3 を選択します。

4. Allowable Drives エリアから sda チェックボックスを選択します。

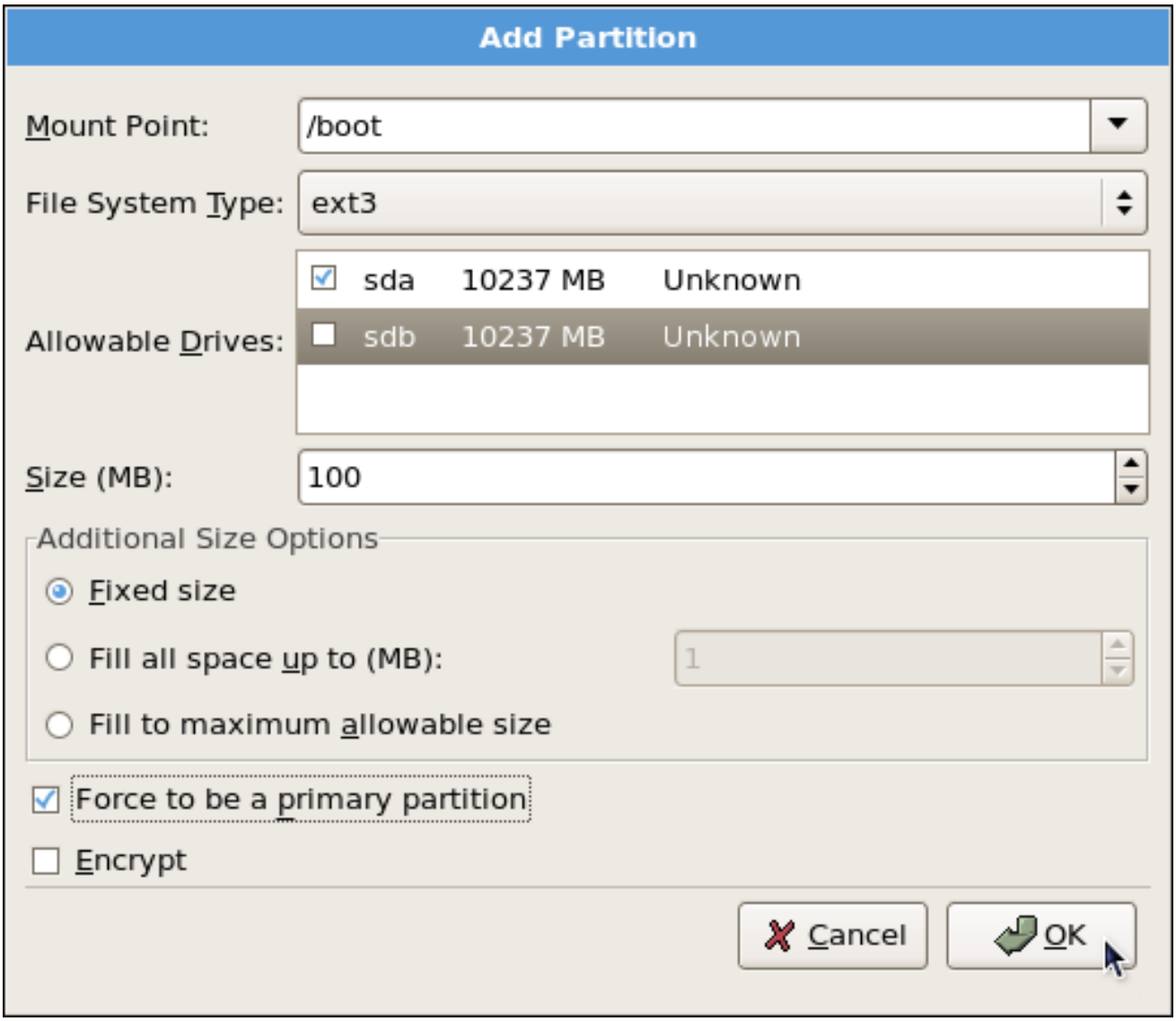
5. Size (MB) メニューの 100 （デフォルト）のままにします。

6. Additional Size Options エリアで Fixed size （デフォルトの）ラジオボタンを選択したままにします。

7. パーティションをプライマリーパーティションにするには、Force to be a primary partition を選択します。プライマリーパーティションは、ハードドライブの最初の 4 つのパーティションの 1 つです。選択されていない場合は、パーティションが論理パーティションとして作成されます。他のオペレーティングシステムがすでにシステム上にある場合は、このオプションの選択を解除する必要があります。プライマリーパーティションと論理/拡張パーティションの詳細は、『Red Hat Enterprise Linux インストールガイド』 の付録セクションを参照してください。

図11.5「ブートパーティションの作成」 を参照して、入力した値を確認します。

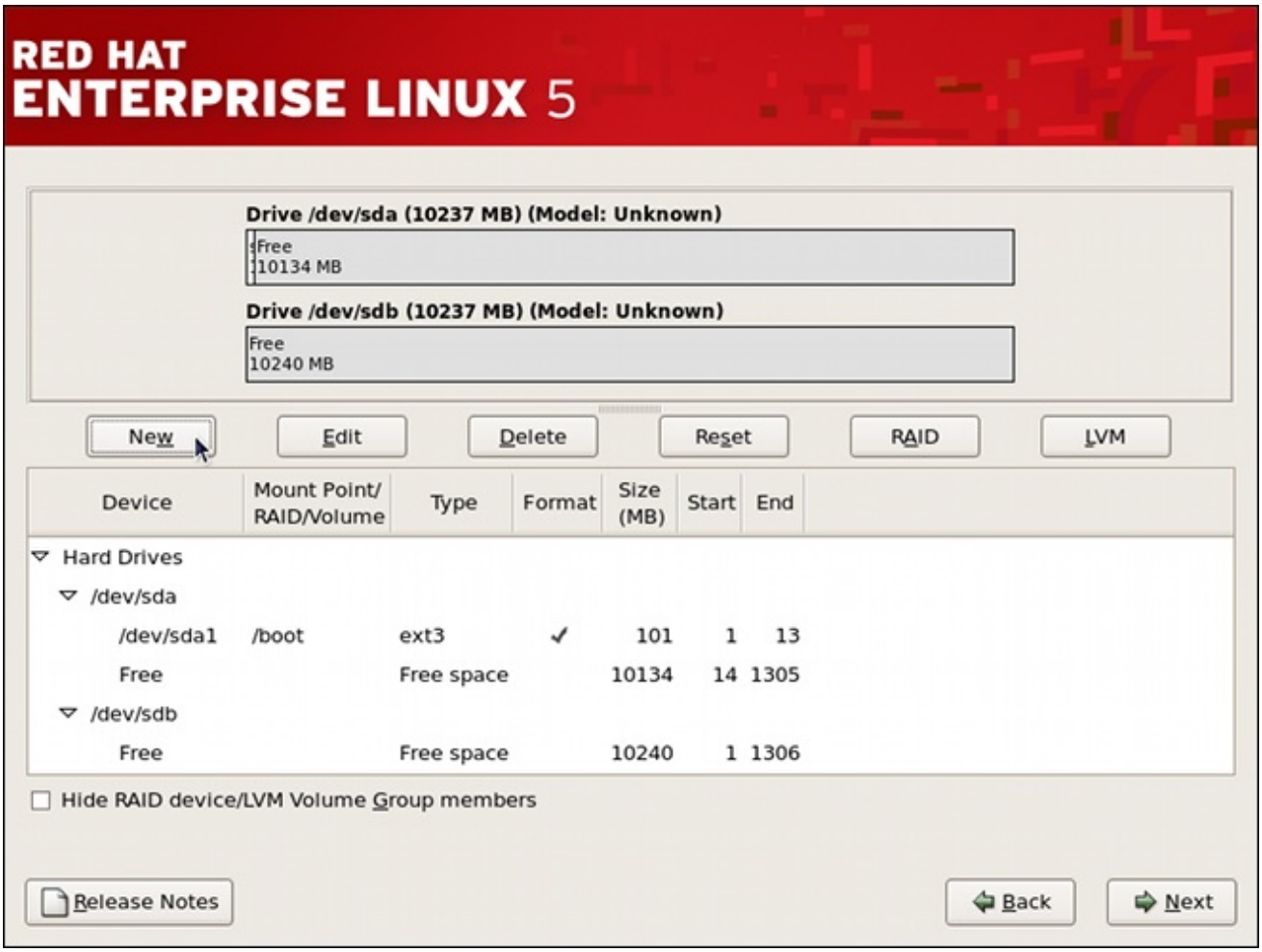
図11.5 ブートパーティションの作成



[D]

OK をクリックしてメイン画面に戻ります。以下の図は、正しく設定されたブートパーティションを示しています。

図11.6 表示される /boot パーティション



[D]

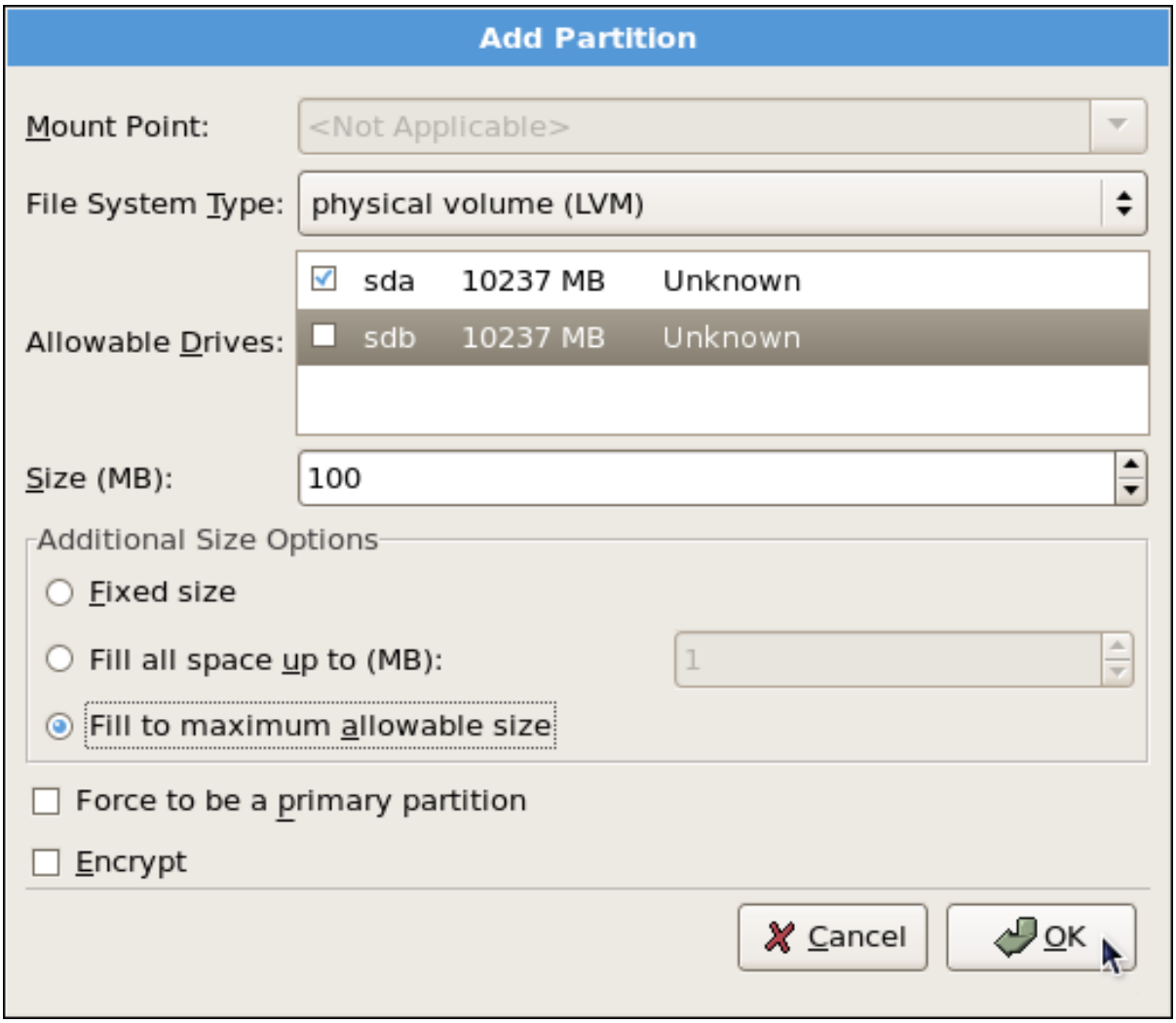
11.4.2. LVM 物理ボリュームの作成

ブートパーティションが作成されると、残りのディスク領域をすべて LVM パーティションに割り当てることができます。成功した LVM 実装を作成する最初の手順は、物理ボリュームの作成です。

1. New を選択します。

2. 図11.7「物理ボリュームの作成」 に示されるように、File System Type プルダウンメニューから 物理ボリューム(LVM) を選択します。

図11.7 物理ボリュームの作成



[D]

3. マウントポイントは入力できません（すべての物理ボリュームを作成してから、すべてのボリュームグループを作成したら）。

4. 物理ボリュームを 1 つのドライブに制限する必要があります。Allowable Drives には、物理ボリュームを作成するドライブを選択します。複数のドライブがある場合は、すべてのドライブが選択されるため、1 つのドライブ以外のすべてのドライブの選択を解除する必要があります。

5. 物理ボリュームのサイズを入力します。

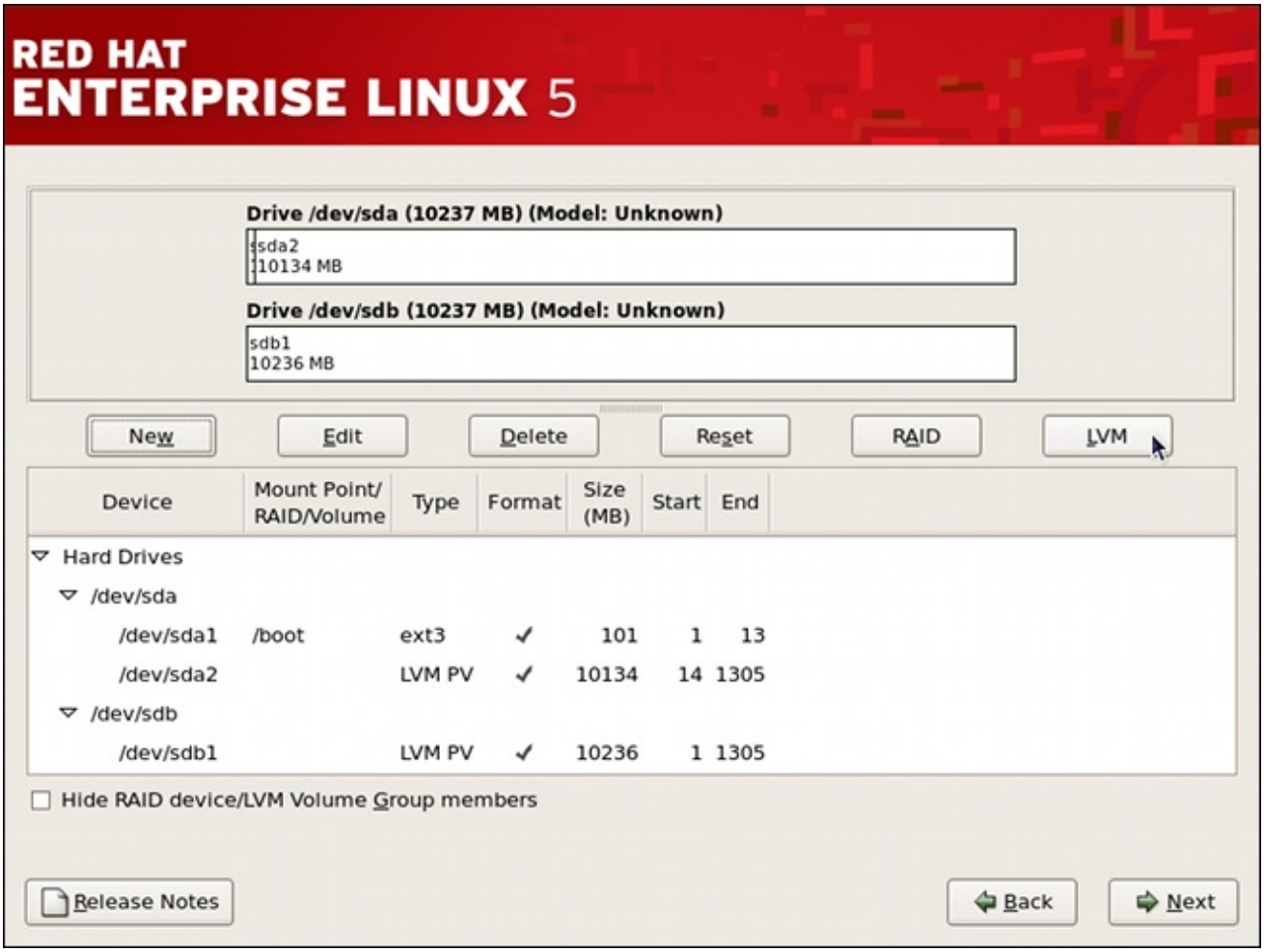
6. Fixed size を選択して、指定したサイズに物理ボリュームを設定し、Fill all space up (MB) を選択し、物理ボリュームサイズの範囲を割り当てるサイズを MB 単位で入力するか、Fill to maximum allowable size を選択して拡張し、ハードディスクで利用可能な領域をすべて埋めます。複数の拡張可能がある場合には、ディスクで利用可能な空き領域を共有します。

7.
パーティションをプライマリーパーティションにする場合は、Force to be a primary partition を選択します。

8.
OK をクリックしてメイン画面に戻ります。

これらの手順を繰り返して、LVM 設定に必要な数の物理ボリュームを作成します。たとえば、ボリュームグループに複数のドライブにまたがる場合は、ドライブごとに物理ボリュームを作成します。以下の図は、繰り返しプロセスが完了した両方のドライブを示しています。

図11.8 2 つの物理ボリュームが作成されている



[D]

11.4.3. LVM ボリュームグループの作成

すべての物理ボリュームを作成したら、ボリュームグループを作成できます。

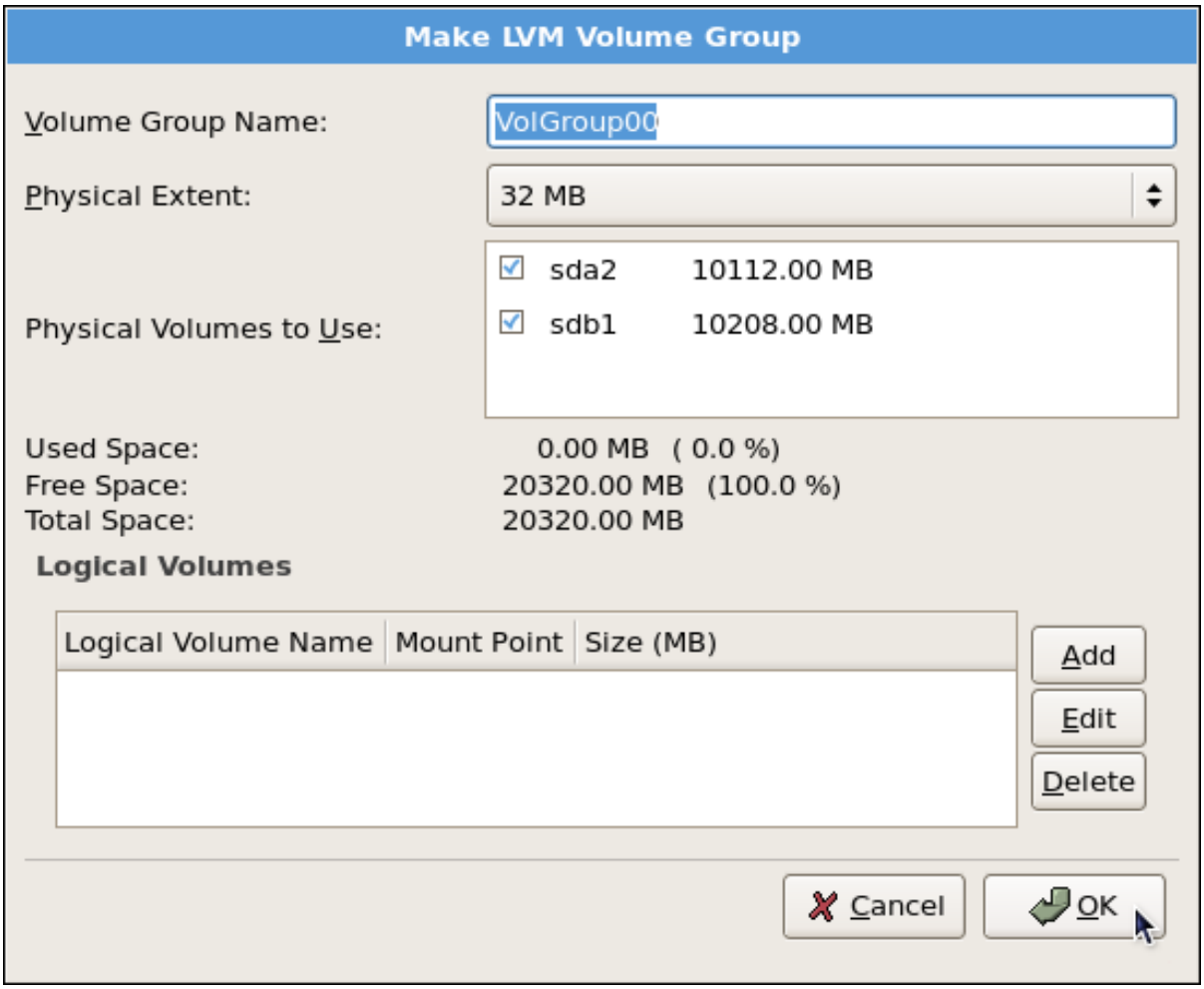
1.
LVM ボタンをクリックして、物理ボリュームをボリュームグループに収集します。ボリュームグループは基本的に物理ボリュームの集合です。論理ボリュームは複数指定できます

が、物理ボリュームは 1 つのボリュームグループにしか存在できません。

注記

ボリュームグループには、オーバーヘッドのディスク領域があります。ボリュームグループのサイズは、物理ボリュームサイズの合計よりもわずかに小さくなります。

図11.9 LVM ボリュームグループの作成



[D]

2. 必要に応じて 、ボリュームグループ名 を変更します。

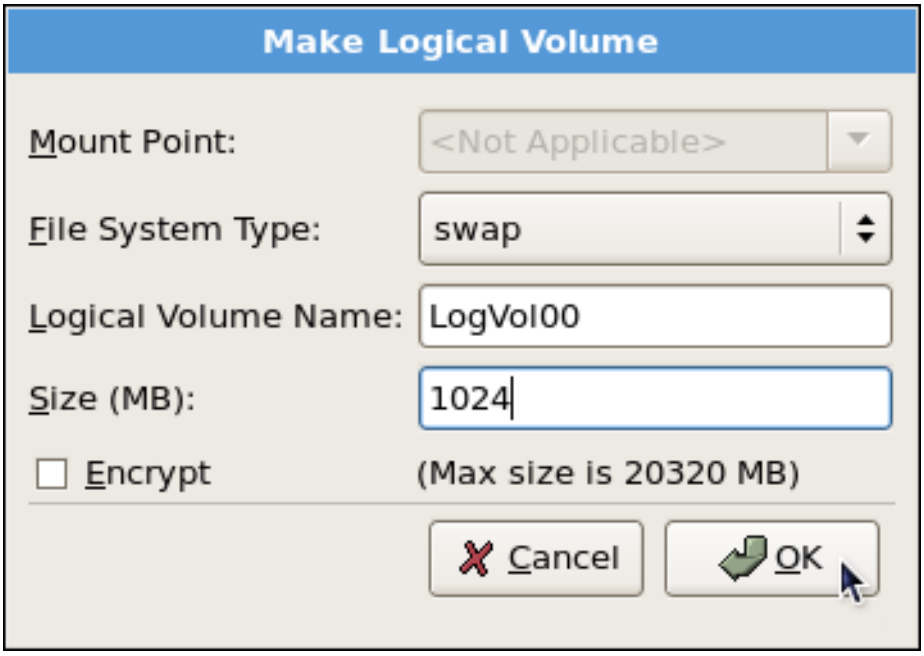
3. ボリュームグループ内のすべての論理ボリュームは、*物理エクステント(PE)* ユニットに割り当てる必要があります。物理エクステントは、データの割り当て単位です。

4. ボリュームグループに使用する物理ボリュームを選択します。

11.4.4. LVM 論理ボリュームの作成

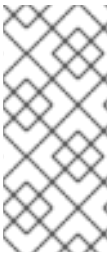
/、/home、swap 領域などのマウントポイントを使用して論理ボリュームを作成します。/boot は論理ボリュームにすることはできません。論理ボリュームを追加するには、Logical Volumes セクションの Add ボタンをクリックします。図11.10「論理ボリュームの作成」 に示されるようにダイアログウィンドウが表示されます。

図11.10 論理ボリュームの作成



[D]

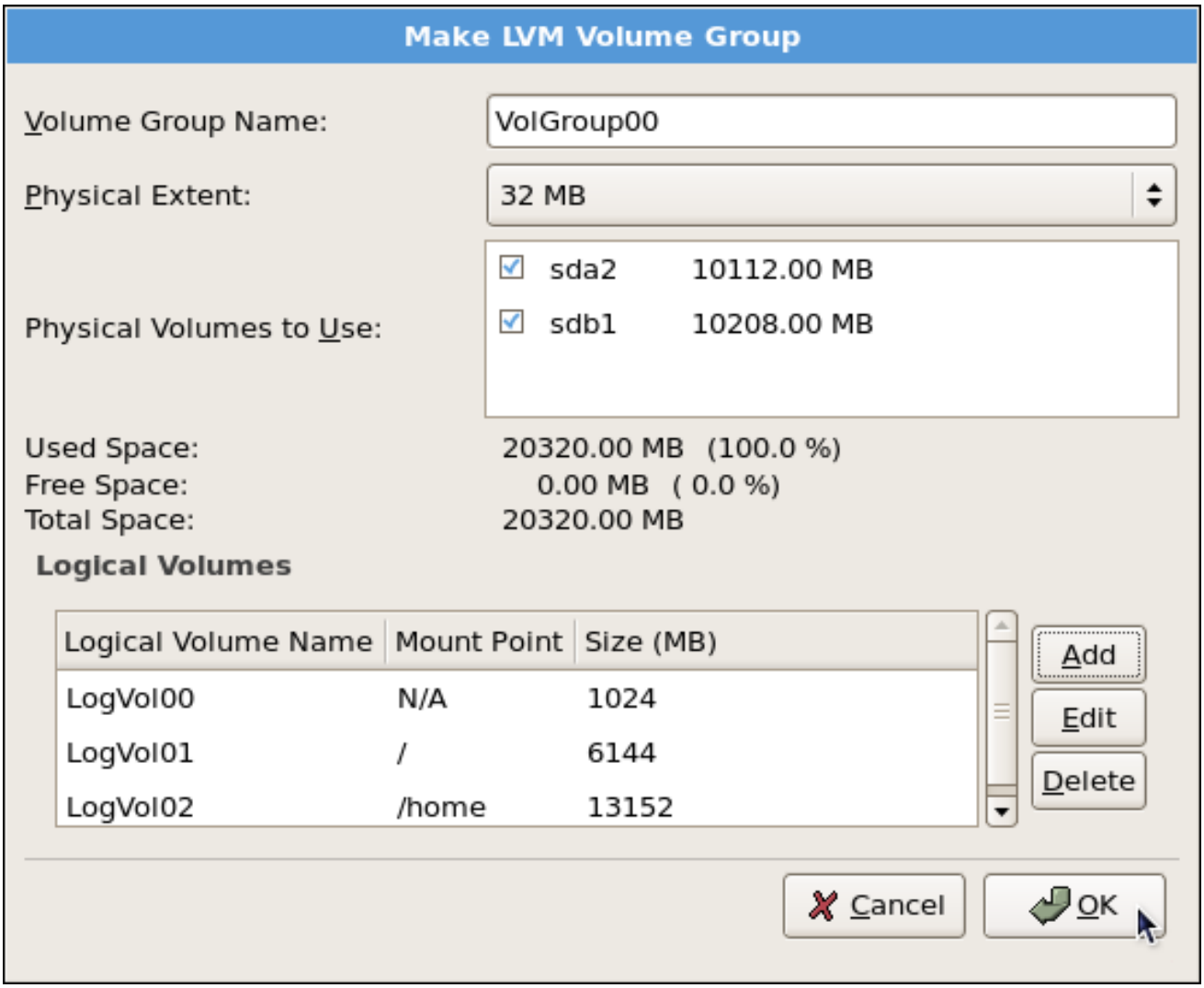
作成するボリュームグループごとに、これらの手順を繰り返します。



ヒント

後で論理ボリュームを拡張できるように、ボリュームグループに空き領域を残しておく必要がある場合があります。デフォルトの自動設定はこれを行いませんが、この手動設定例は、今後の拡張用に、約 1 GB の空き領域として残されています。

図11.11 保留中の論理ボリューム

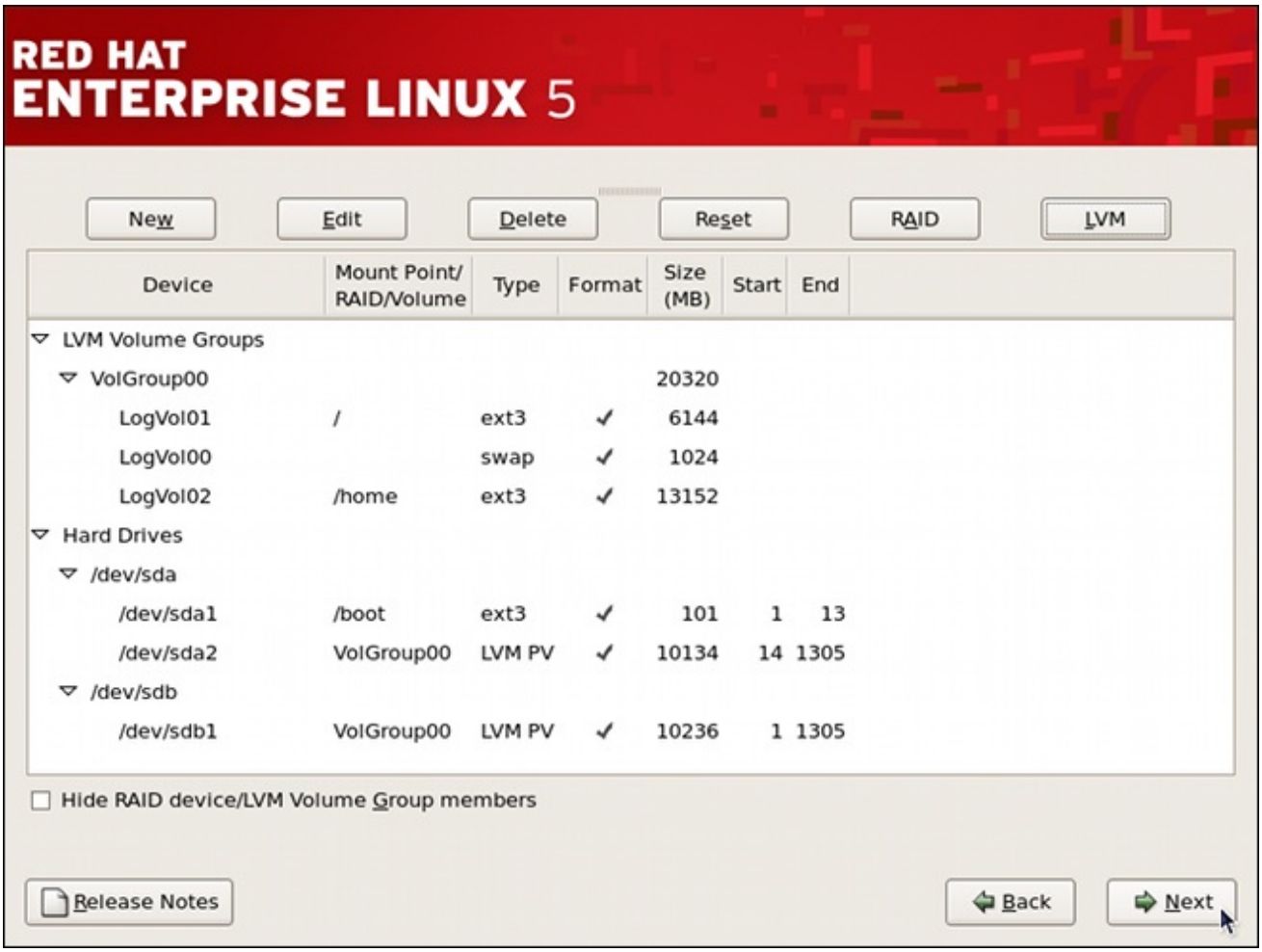


[D]

OK をクリックして、ボリュームグループと関連するすべての論理ボリュームを適用します。

以下の図は、最終的な手動設定を示しています。

図11.12 最終的な手動設定



[D]

11.5. LVM ユーティリティー SYSTEM-CONFIG-LVMの使用

LVM ユーティリティーを使用すると、X ウィンドウ内またはグラフィカルに論理ボリュームを管理できます。メニューパネルから System > Administration > Logical Volume Management から を選択すると、アプリケーションにアクセスできます。または、ターミナルから system-config-lvm と入力して、論理ボリューム管理ユーティリティーを起動することもできます。

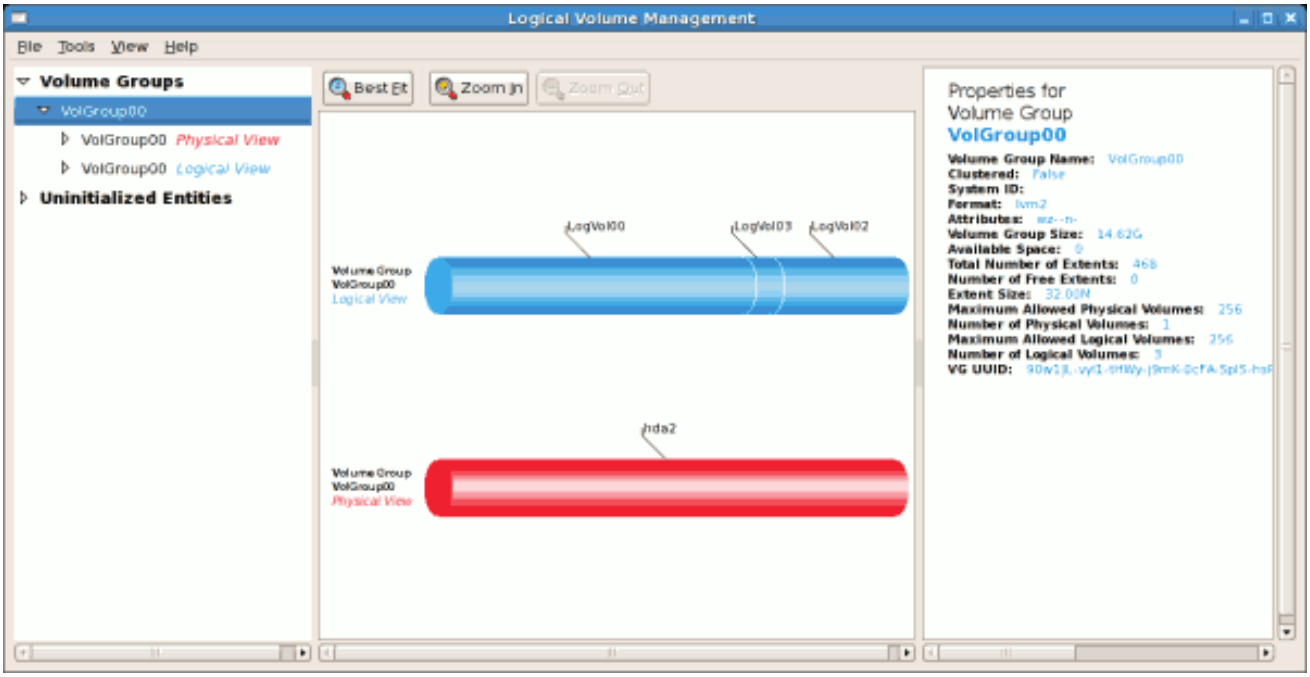
このセクションで使用されている例では、インストール中に作成されたボリュームグループの詳細を以下に示します。

> /boot - (Ext3) file system. Displayed under 'Uninitialized Entities'. (DO NOT initialize this partition).
> LogVol00 - (LVM) contains the (/) directory (312 extents).
> LogVol02 - (LVM) contains the (/home) directory (128 extents).
> LogVol03 - (LVM) swap (28 extents).

上記の論理ボリュームはディスクエンティティー /dev/hda2 に作成され、/boot は /dev/hda1 に作成されました。このシステムは、初期化されていないエンティティーも設定されており、図11.17「初期化されていないエンティティー」 で説明されています。次の図は、LVM ユーティリティーのメイン

ウィンドウを示しています。上記の設定の論理ビューと物理ビューを以下に示します。3 つの論理ボリュームは、同じ物理ボリューム (hda2) に存在します。

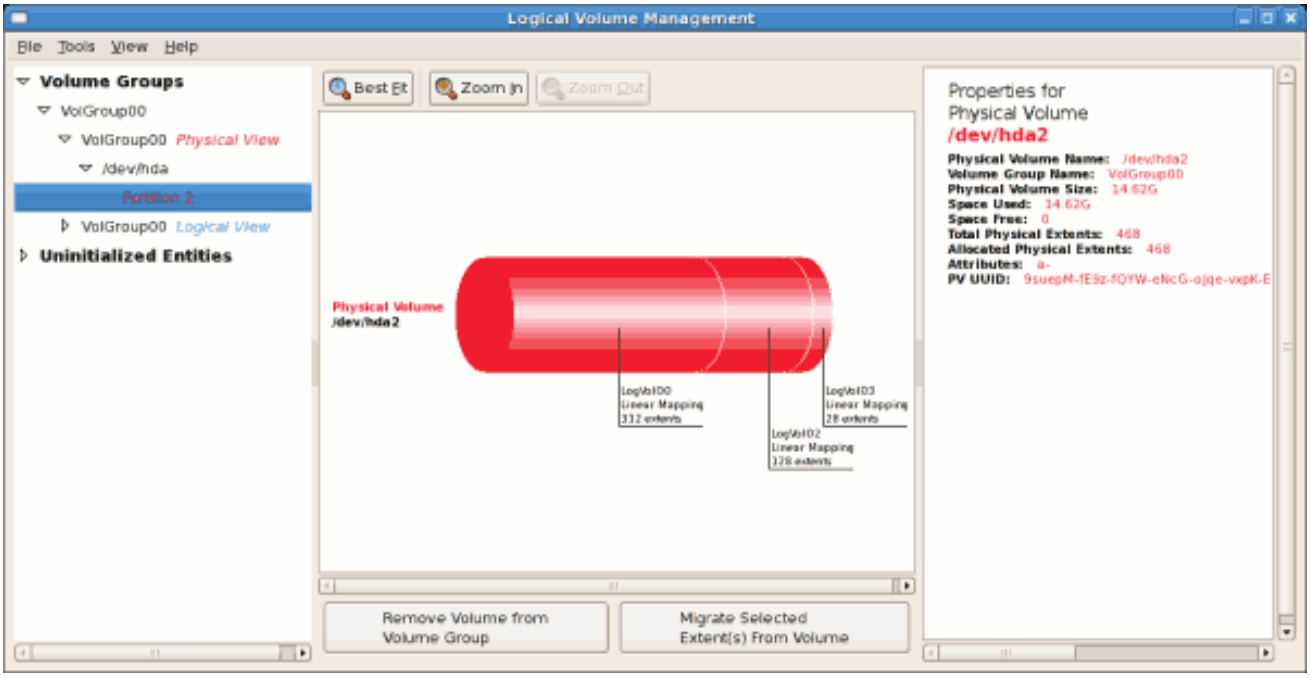
図11.13 メイン LVM ウィンドウ



[D]

次の図は、ボリュームの物理ビューを示しています。このウィンドウでは、ボリュームを選択してボリュームグループから削除したり、エクステントをボリュームから別のボリュームグループに移行したりできます。エクステントを移行する手順については 図11.22「エクステントの移行」 で説明しています。

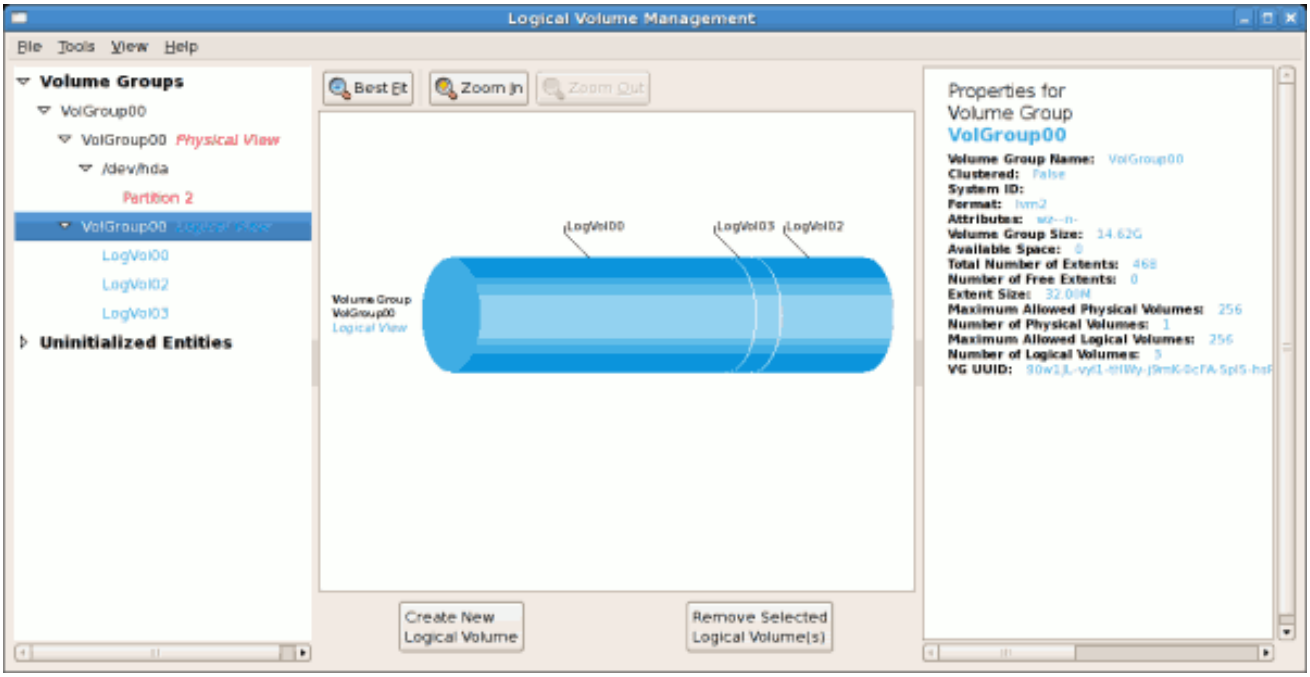
図11.14 物理ビューウィンドウ



[D]

次の図は、選択したボリュームグループの論理ビューを示しています。論理ボリュームのサイズは、表示される個々の論理ボリュームサイズでも示されます。

図11.15 論理ビューウィンドウ

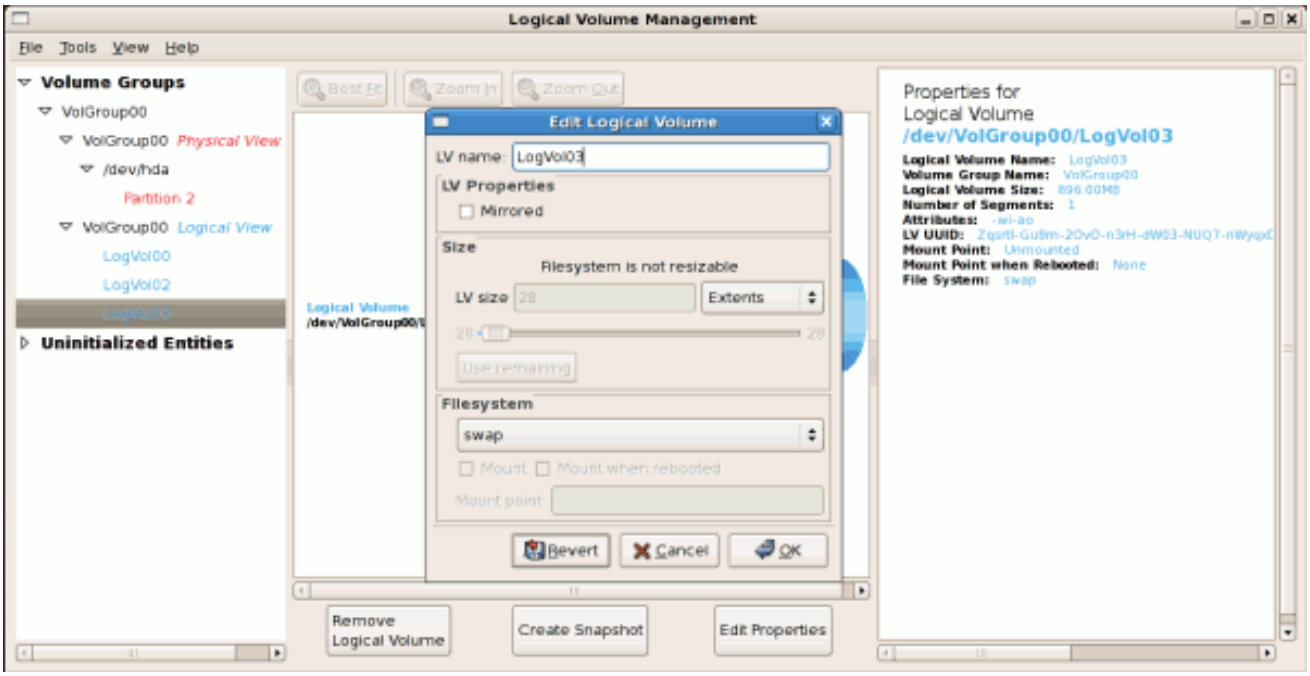


[D]

左側の列で、ボリュームグループ内の個々の論理ボリュームを選択して、それぞれの詳細を表示できます。この例では、'LogVol03' の論理ボリューム名を 'Swap' に変更することが目的です。この操作を実行するには、それぞれの論理ボリュームを選択し、Edit Properties ボタンをクリックします。これにより、論理ボリューム名、サイズ（エクステント）を変更したり、論理ボリュームグループで使用可能な残りの領域を使用できる論理ボリュームの編集ウィンドウが表示されます。次の図はこれを示しています。

現在、ボリュームグループに空き領域がないため、この論理ボリュームのサイズを変更することはできません。残りのスペースがある場合、このオプションは有効になります (図11.31「論理ボリュームの編集」 を参照)。OK ボタンをクリックして、変更を保存します (これにより、ボリュームが再マウントされます)。変更をキャンセルするには、Cancel ボタンをクリックします。最後のスナップショット設定に戻すには、元に Revert ボタンをクリックします。スナップショットは、LVM ユーティリティーウィンドウの Create Snapshot ボタンをクリックして作成できます。選択した論理ボリュームが / (root)ディレクトリーで使用されている場合、ボリュームのアンマウントはできないため、このタスクは成功しません。

図11.16 論理ボリュームの編集

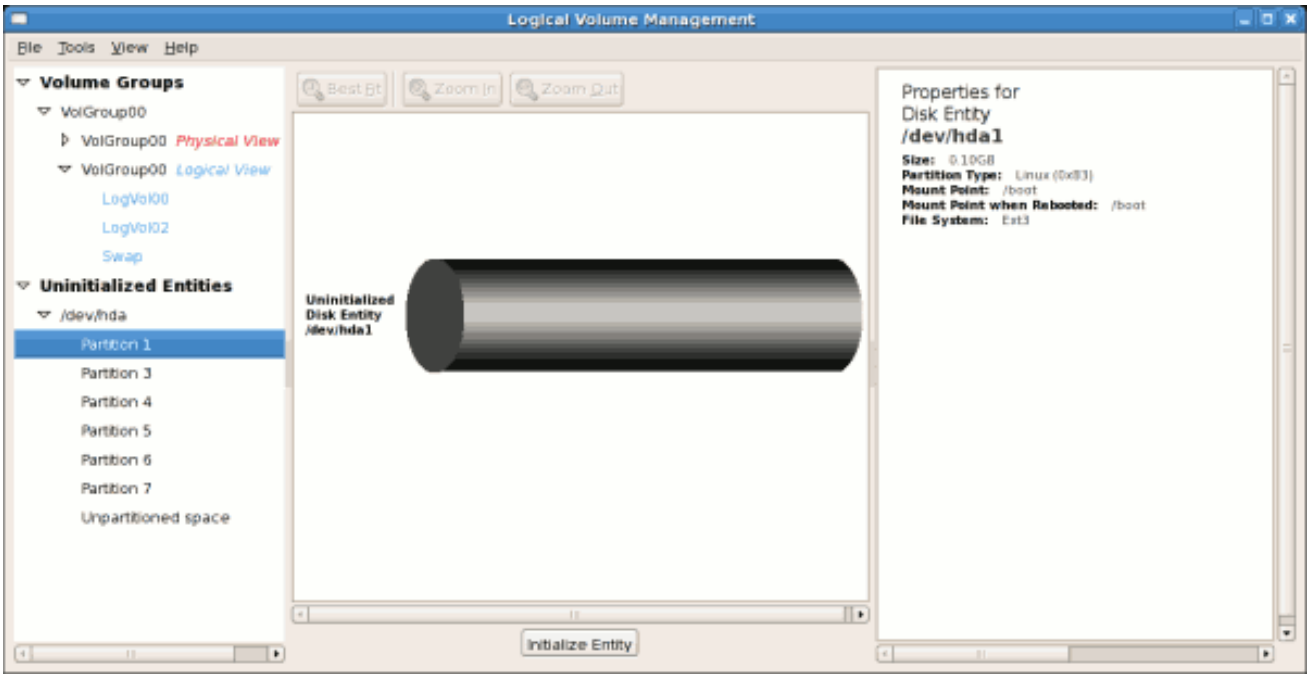


[D]

11.5.1. 初期化されていないエンティティーの使用

初期化されていないエンティティーは、パーティション化されていないスペースと非 LVM ファイルシステムで設定されています。この例では、インストール中にパーティション 3、4、5、6、および 7 が作成され、ハードディスクに一部のパーティション化されていないスペースが残っています。各パーティションを表示し、ウィンドウの右側の列にあるディスクエンティティーのプロパティーを読んで、重要なデータを削除していないことを確認します。この例では、パーティション 1 は /boot であるため初期化できません。初期化されていないエンティティーを以下に示します。

図11.17 初期化されていないエンティティー



[D]

この例では、パーティション 3 が初期化され、既存のボリュームグループに追加されます。パーティションまたはパーティション化されていないスペースを初期化するには、パーティションを選択し、Initialize Entity ボタンをクリックします。初期化されると、ボリュームは未割り当てボリュームリストに表示されます。

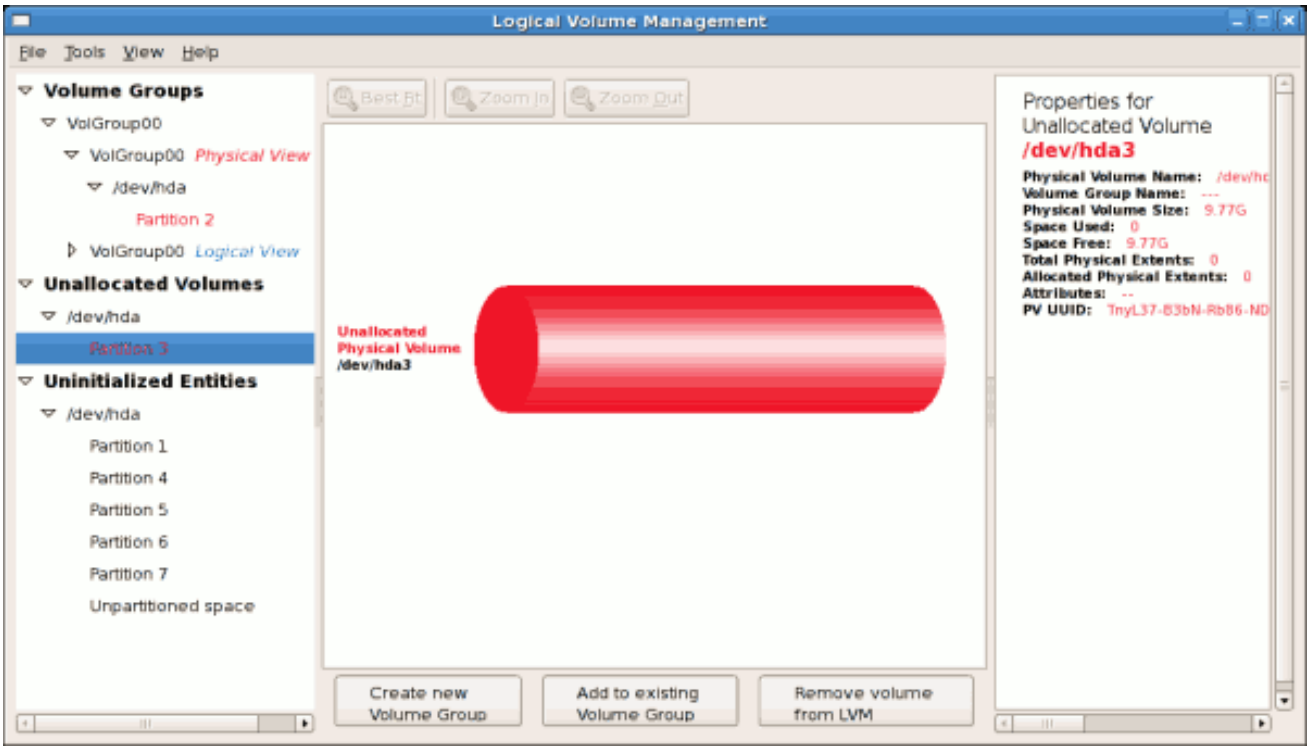
11.5.2. 未割り当てボリュームのボリュームグループへの追加

初期化されると、ボリュームは未割り当てボリュームリストに表示されます。以下の図は、未割り当てのパーティション (パーティション 3) を示しています。ウィンドウの下部にあるそれぞれのボタンを使用すると、以下のことができます。

- 新しいボリュームグループの作成
- 未割り当てのボリュームの既存のボリュームグループへの追加
- LVM からのボリュームの削除

ボリュームを既存のボリュームグループに追加するには、Add to Existing Volume Group ボタンをクリックします。

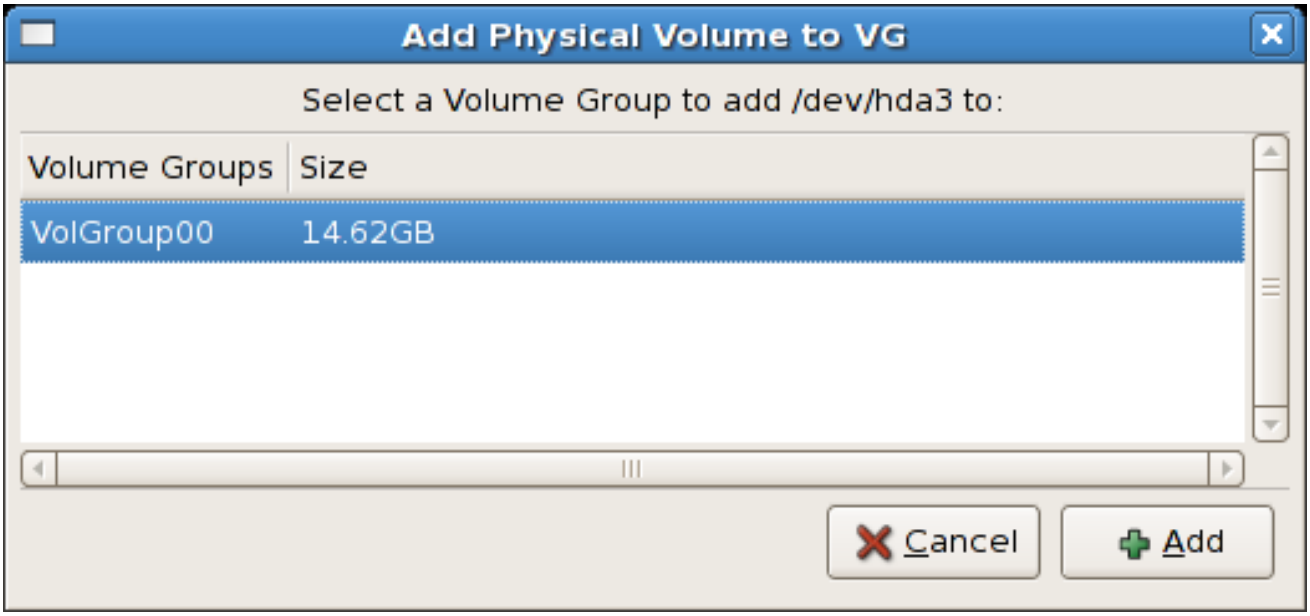
図11.18 未割り当てのボリューム



[D]

Add to Existing Volume Group ボタンをクリックすると、初期化する物理ボリュームを追加できる既存のボリュームグループを一覧表示するポップアップウィンドウが表示されます。ボリュームグループは、1 つ以上のハードディスクにまたがることができます。この例では、以下に示すように、ボリュームグループは 1 つだけ存在します。

図11.19 ボリュームグループへの物理ボリュームの追加



[D]

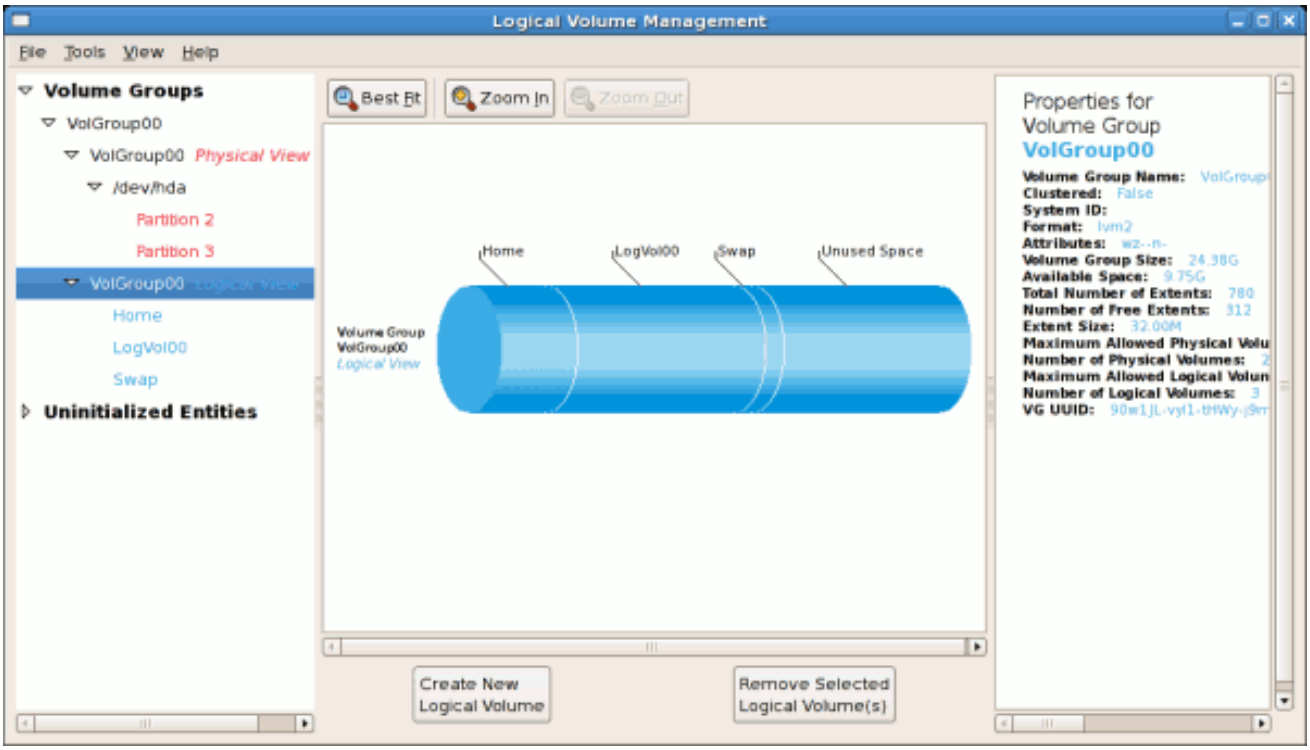
既存のボリュームグループに追加されると、新しい論理ボリュームは、選択したボリュームグルー

プの未使用スペースに自動的に追加されます。未使用のスペースを使用して、以下のことができます。

- 新しい論理ボリュームを作成します（ Create New Logical Volume (s) ボタンをクリックします。
- 既存の論理ボリュームの 1 つを選択し、エクステントを増やす (「ボリュームグループの拡張」 を参照)
- 既存の論理ボリュームを選択し、Remove Selected Logical Volume(s) ボタンをクリックして、ボリュームグループから削除する。この操作を実行するために未使用のスペースを選択できないことに注意してください。

以下の図は、新しいボリュームグループを追加した後の 'VolGroup00' の論理ビューを示しています。

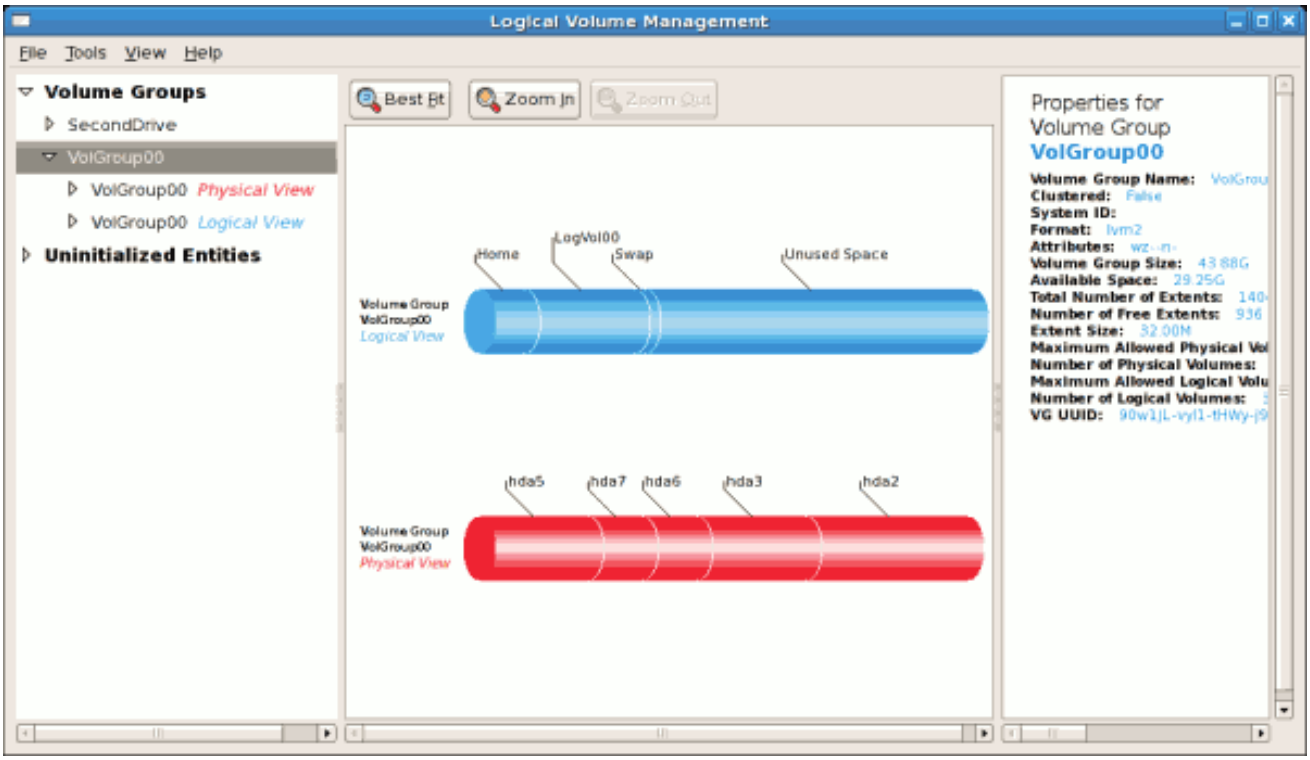
図11.20 ボリュームグループの論理ビュー



[D]

以下の図では、初期化されていないエンティティー (パーティション 3、5、6、および 7) が 'VolGroup00' に追加されています。

図11.21 ボリュームグループの論理ビュー

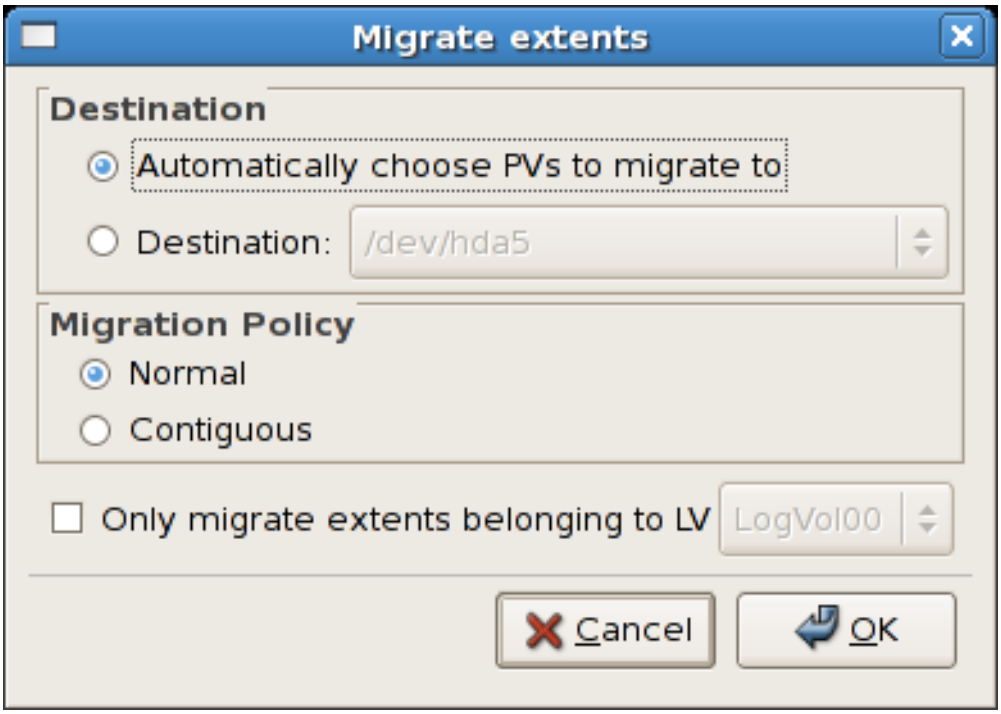


[D]

11.5.3. エクステントの移行

物理ボリュームからエクステントを移行するには、ボリュームを選択し、選択し たボリュームから移行 ボタンを クリックします。ボリュームグループ内でエクステントを移行するには、十分な数の空きエクステントが必要であることに注意してください。十分な数の空きエクステントがない場合は、エラーメッセージが表示されます。この問題を解決するには、ボリュームグループを拡張してください(「ボリュームグループの拡張」を参照してください)。ボリュームグループで十分な数の空きエクステントが検出されると、ポップアップウィンドウが表示され、エクステントの宛先を選択するか、LVM で移行先となる物理ボリューム(PV)を自動的に選択できます。以下で説明します。

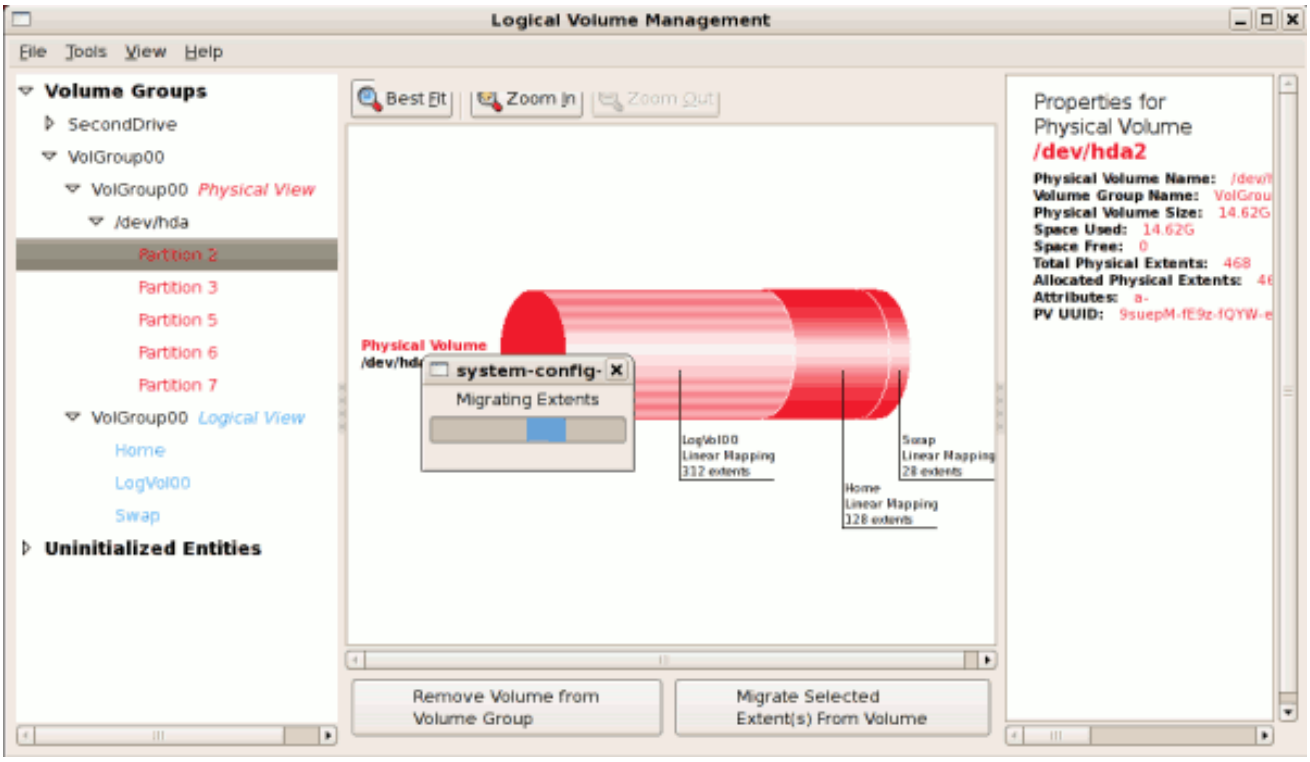
図11.22 エクステントの移行



[D]

以下の図は、進行中のエクステントの移行を示しています。この例では、エクステントはパーティション 3 に移行されました。

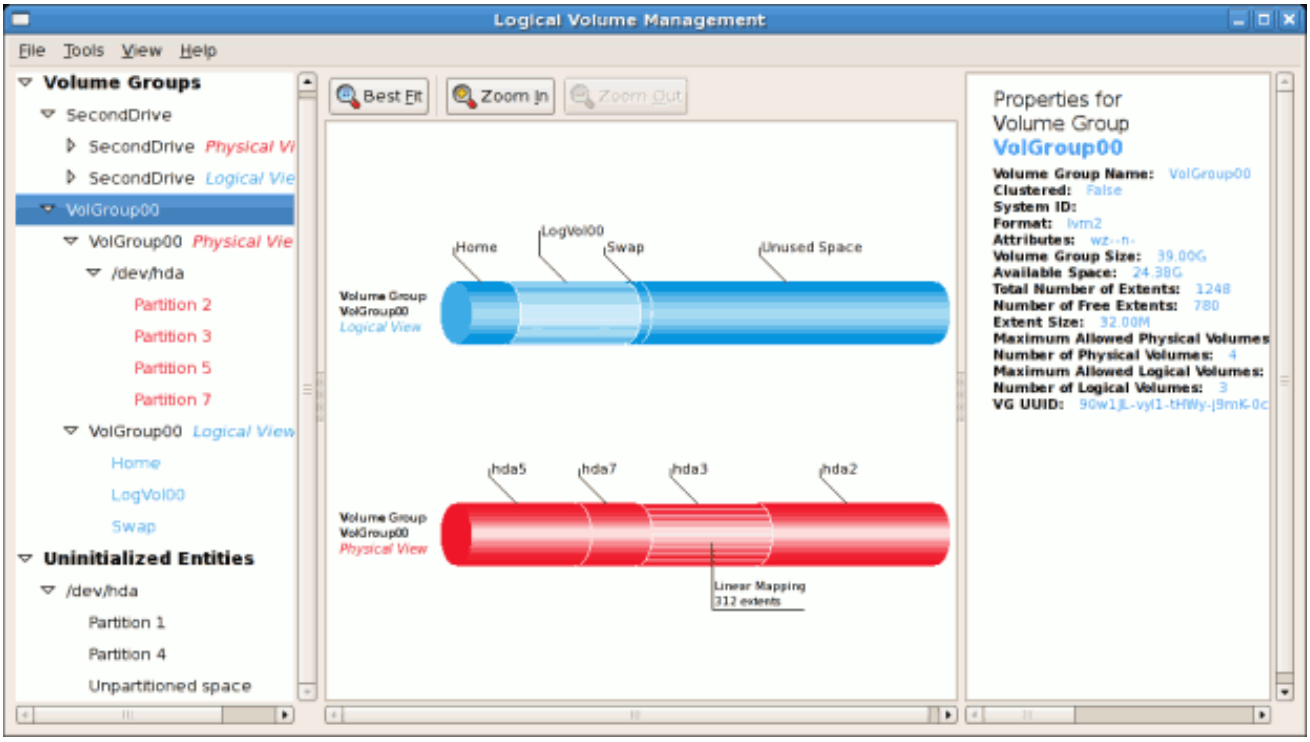
図11.23 進行中のエクステントの移行



[D]

エクステントが移行されると、未使用のスペースが物理ボリュームに残ります。次の図は、ボリュームグループの物理ビューと論理ビューを示しています。最初は hda2 にあった LogVol00 のエクステントは、現在は hda3 にあります。エクステントを移行すると、ハードディスクをアップグレードした場合に論理ボリュームを移動したり、ディスクスペースをより適切に管理したりできます。

図11.24 ボリュームグループの論理的および物理的ビュー

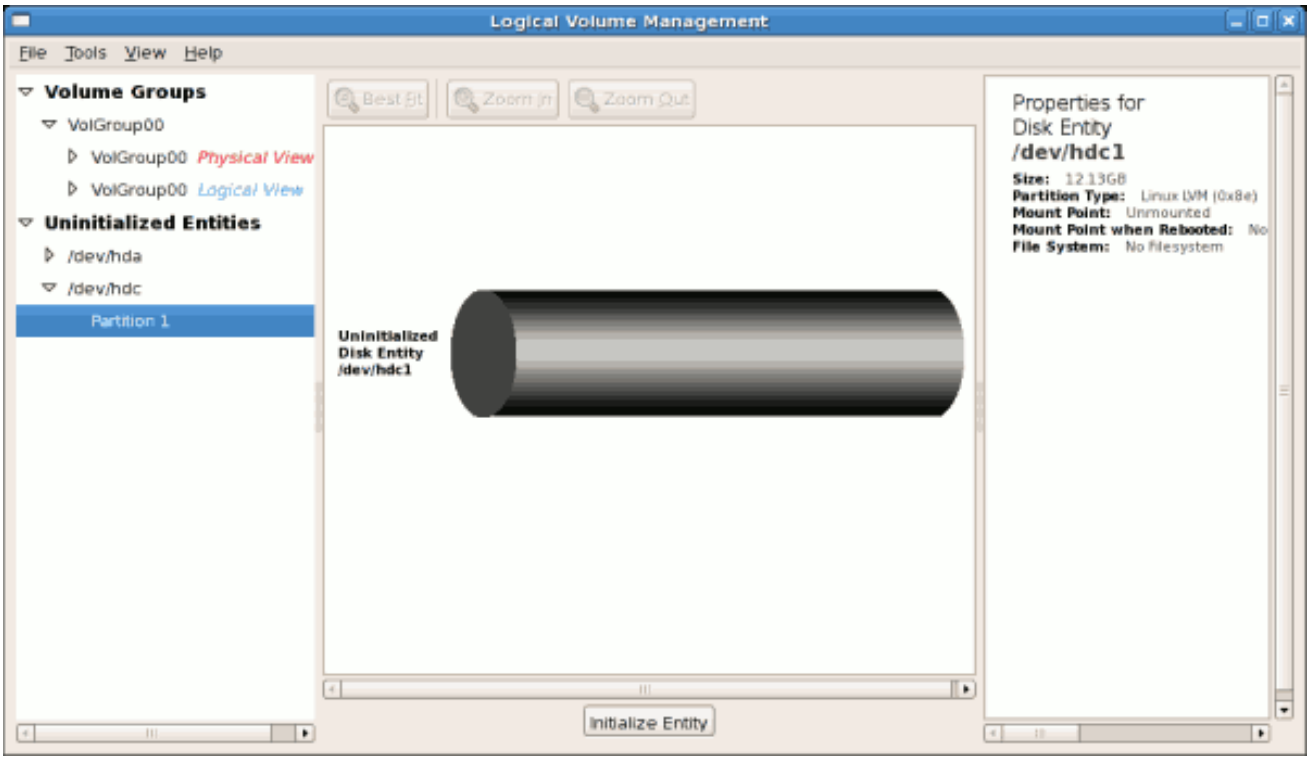


[D]

11.5.4. LVM を使用した新しいハードディスクの追加

この例では、新しい IDE ハードディスクが追加されました。以下の図は、新しいハードディスクの詳細を示しています。下の図から、ディスクは初期化されておらず、マウントされていません。パーティションを初期化するには、Initialize Entity ボタンをクリックします。詳細は、「初期化されていないエンティティーの使用」 を参照してください。図11.26「新しいボリュームグループの作成」 に示すように、初期化されると、LVM は、新しいボリュームを未割り当てボリュームのリストに追加します。

図11.25 初期化されていないハードディスク

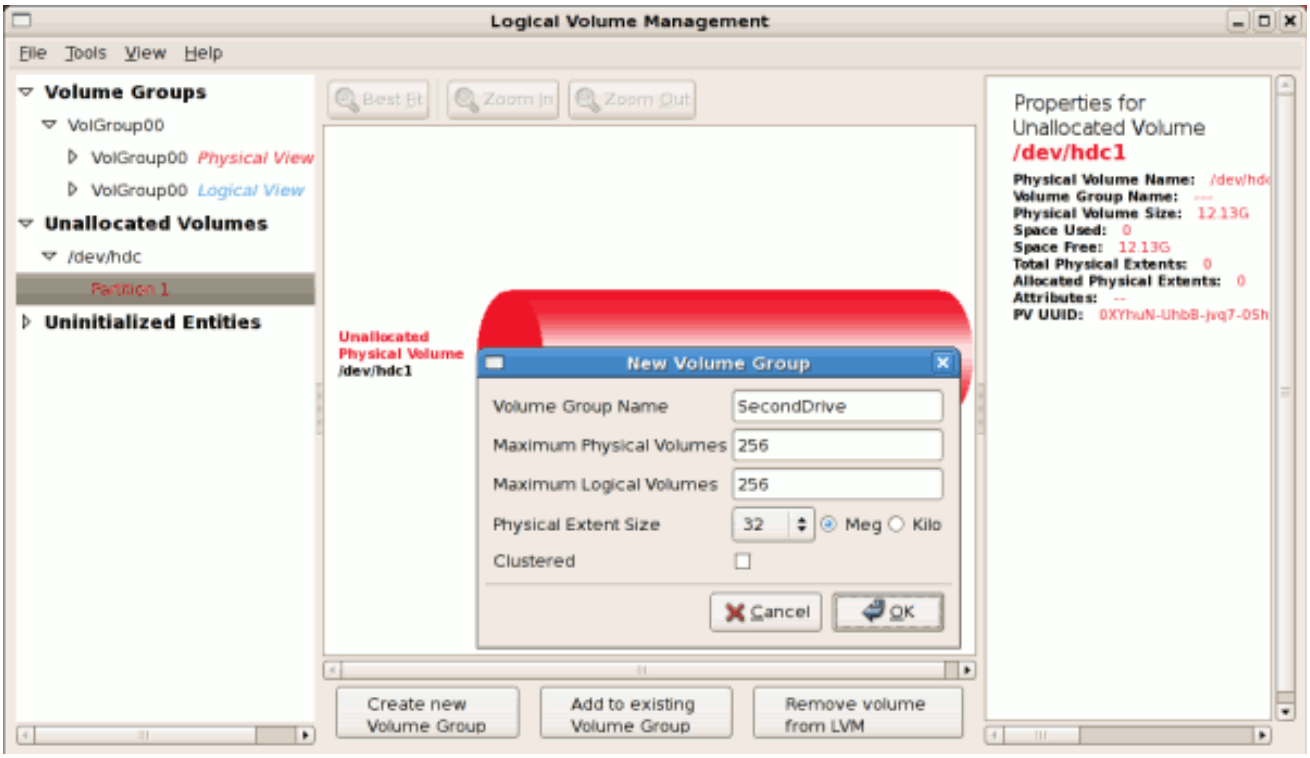


[D]

11.5.5. 新しいボリュームグループの追加

初期化されると、LVM は新しいボリュームを未割り当てボリュームのリストに追加し、既存のボリュームグループに追加したり、新しいボリュームグループを作成したりできます。LVM からボリュームを削除することもできます。LVM から削除されたボリュームが、図11.25「初期化されていないハードディスク」 に示されるように、初期化されていないエンティティーの一覧に一覧表示されます。この例では、以下に示すように新しいボリュームグループが作成されました。

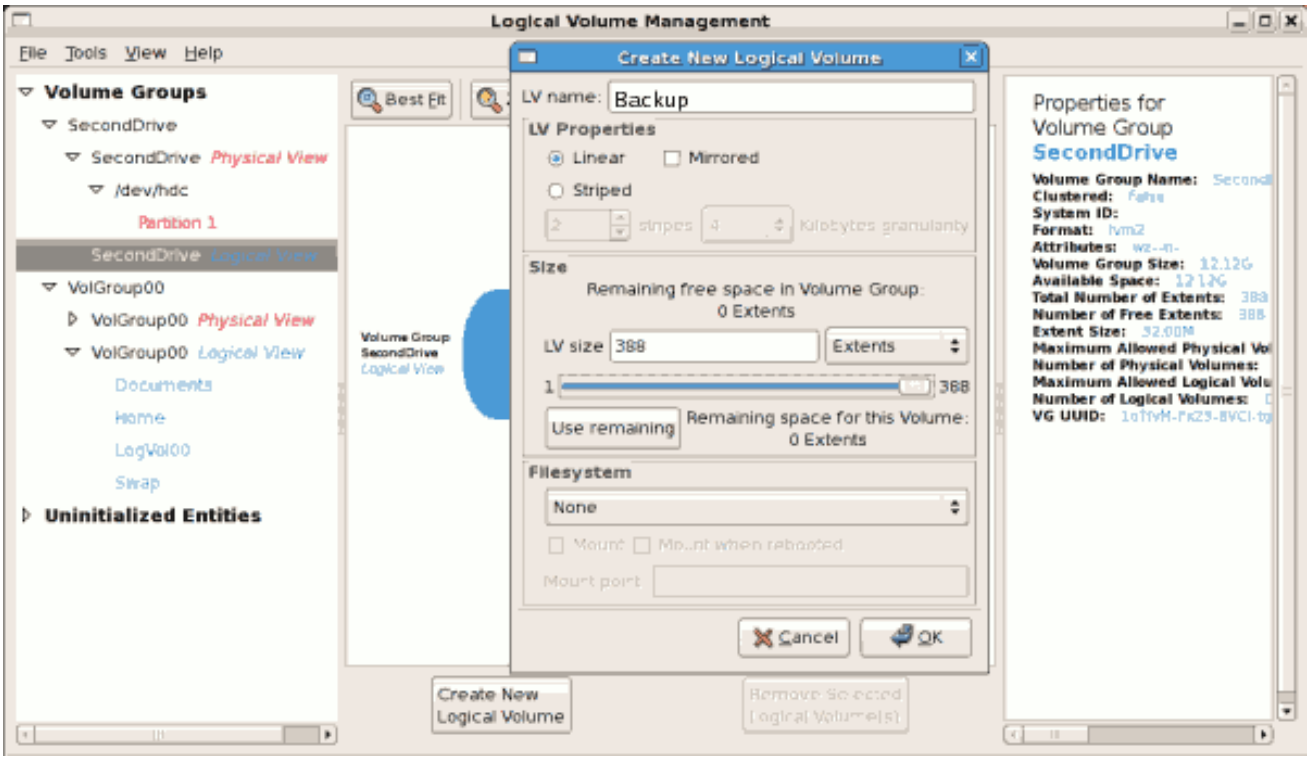
図11.26 新しいボリュームグループの作成



[D]

新しいボリュームグループを作成すると、以下に示すように、既存のボリュームグループのリストに表示されます。論理ボリュームが作成されていないため、論理ビューには未使用のスペースを含む新しいボリュームグループが表示されます。論理ボリュームを作成するには、以下に示すように、ボリュームグループを選択し、**Create New Logical Volume** ボタンをクリックします。ボリュームグループで使用するエクステントを選択してください。この例では、ボリュームグループ内のすべてのエクステントを使用して、新しい論理ボリュームを作成しました。

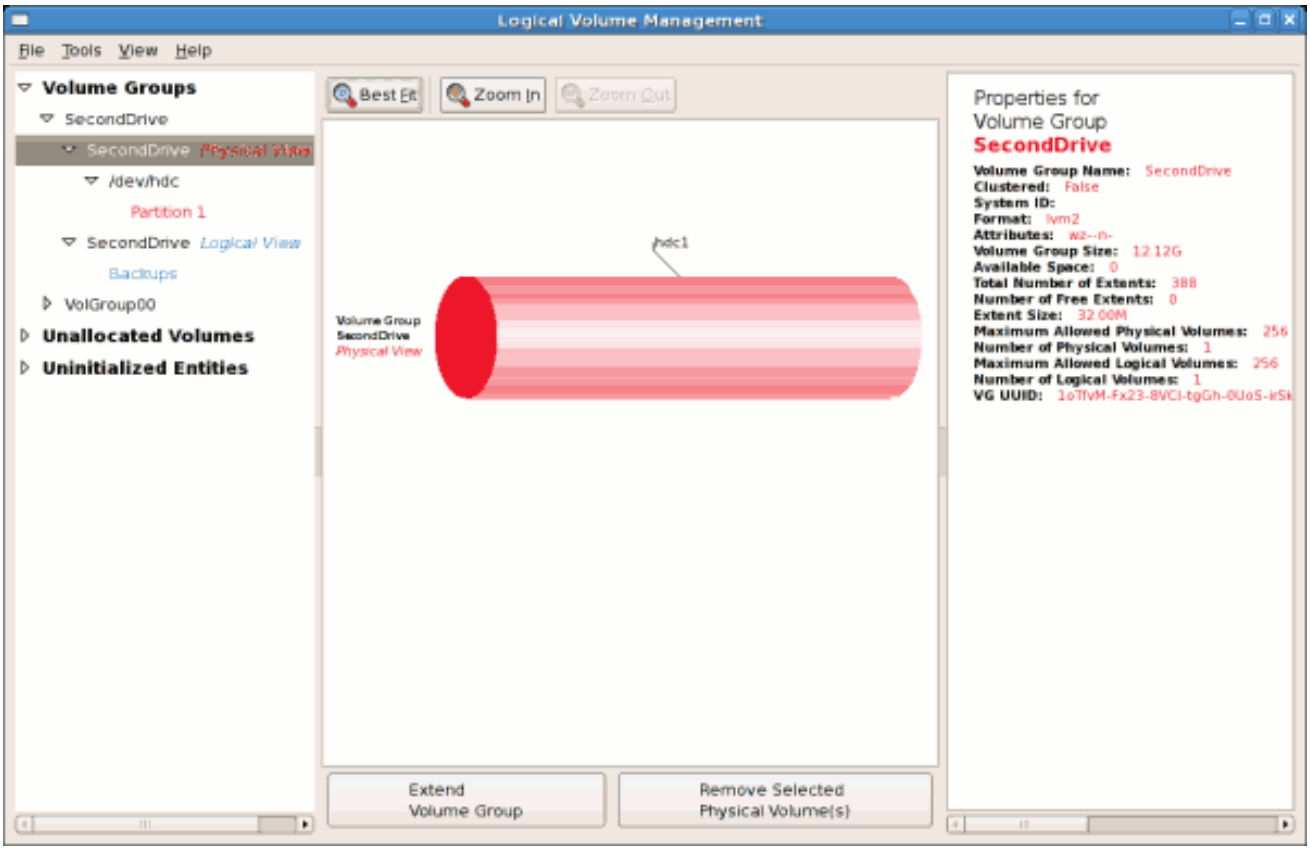
図11.27 新しい論理ボリュームを作成する



[D]

次の図は、新しいボリュームグループの物理ビューを示しています。このボリュームグループの 'Backups' という名前の新しい論理ボリュームも一覧表示されます。

図11.28 新しいボリュームグループの物理ビュー

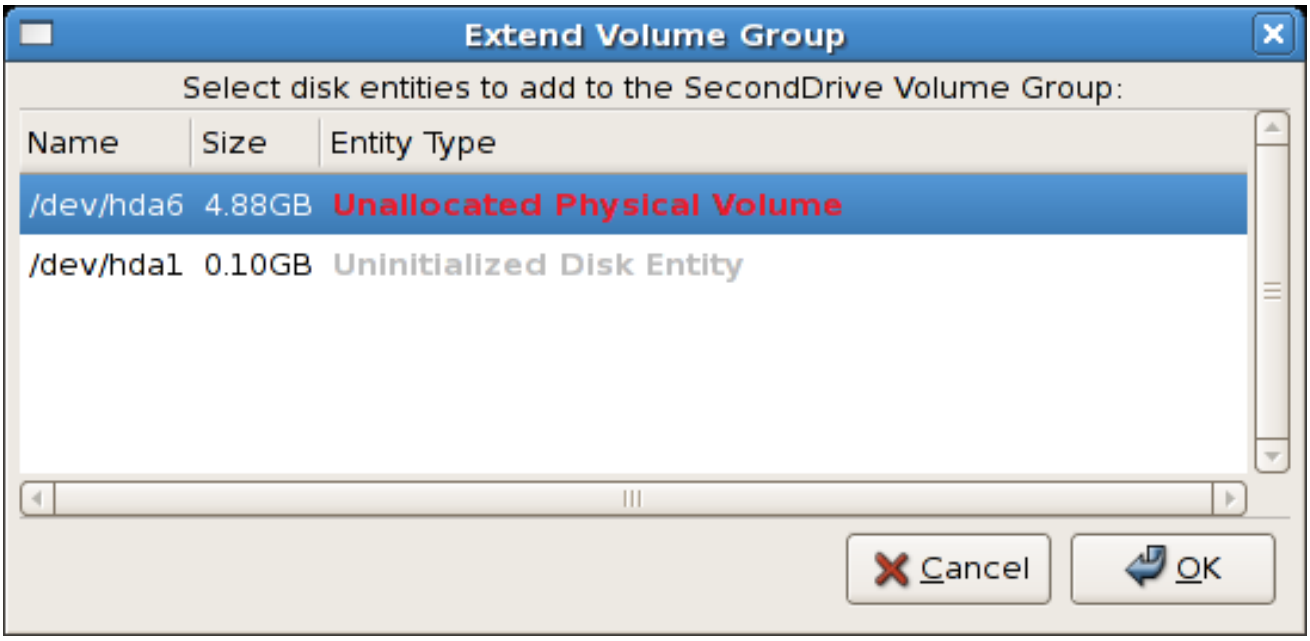


[D]

11.5.6. ボリュームグループの拡張

この例では、初期化されていないエンティティー (パーティション) を含むように新しいボリュームグループを拡張することが目的でした。これは、ボリュームグループのサイズまたはエクステントのサイズまたは数を増やすためです。ボリュームグループを拡張するには、Extend Volume Group ボタンをクリックします。これにより、以下に示すように 'Extend Volume Group' ウィンドウが表示されます。'Extend Volume Group' ウィンドウで、ボリュームグループに追加するディスクエンティティー (パーティション) を選択できます。重要なデータが削除されないように、初期化されていないディスクエンティティー（パーティション）の内容を確認してください(図11.25「初期化されていないハードディスク」を参照してください)。この例では、以下に示すように、ディスクエンティティー (パーティション) /dev/hda6 が選択されています。

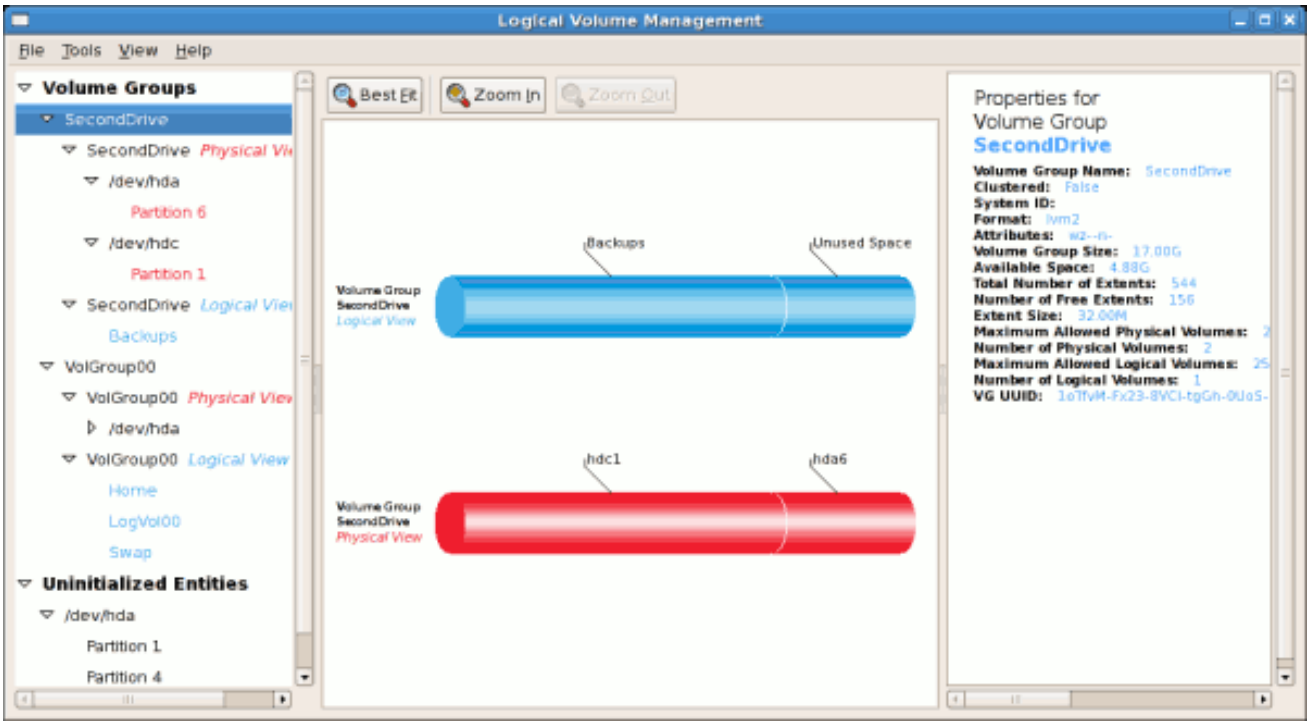
図11.29 ディスクエンティティーの選択



[D]

新しいボリュームが追加されると、ボリュームグループに 'Unused Space' として追加されます。以下の図は、ボリュームグループが拡張された後の論理ビューと物理ビューを示しています

図11.30 拡張ボリュームグループの論理的および物理的ビュー



[D]

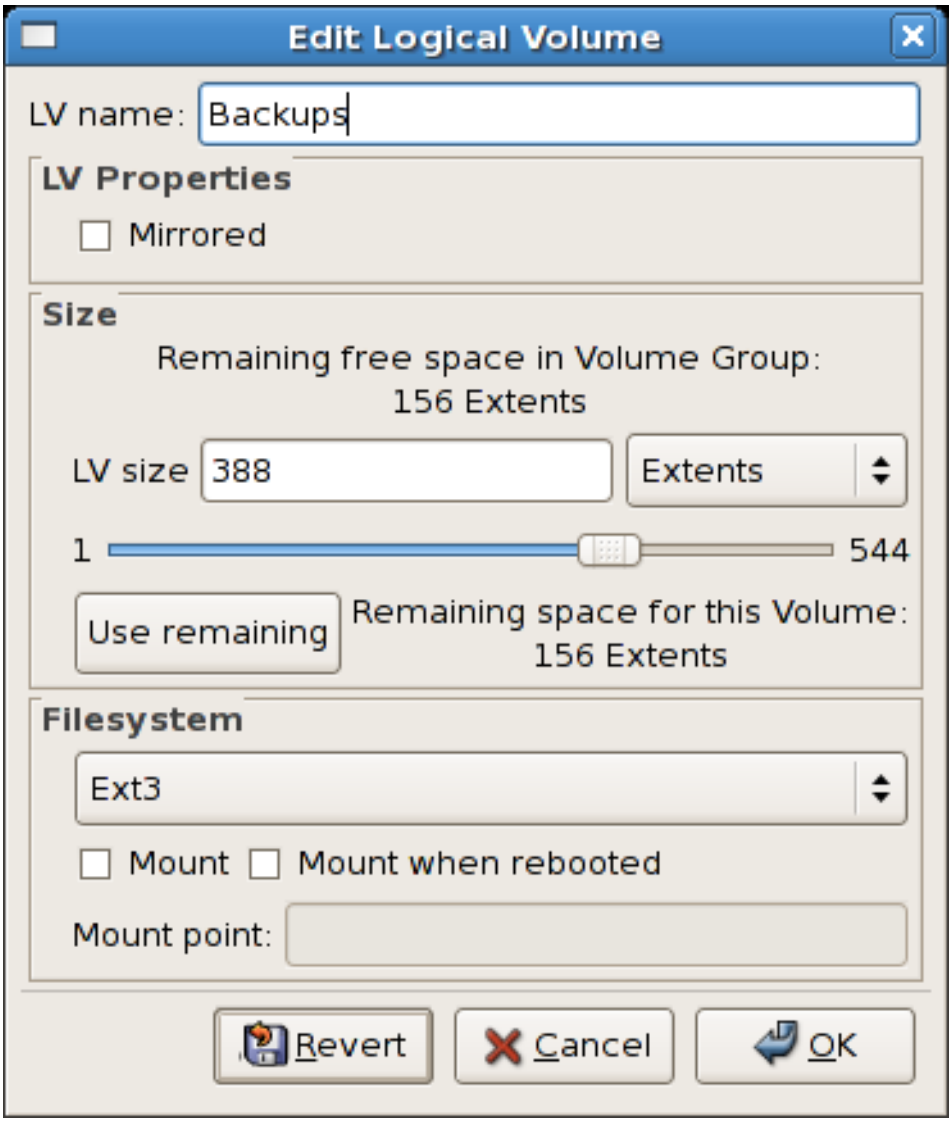
11.5.7. 論理ボリュームの編集

LVM ユーティリティーを使用すると、ボリュームグループ内の論理ボリュームを選択し、その名

前、サイズを変更して、ファイルシステムのオプションを指定できます。この例では、'Backups' という名前の論理ボリュームが、ボリュームグループの残りのスペースに拡張されました。

Edit Properties ボタンをクリックすると、'Edit Logical Volume' ポップアップウィンドウが表示され、そこから論理ボリュームのプロパティーを編集できます。このウィンドウでは、変更後のボリュームをマウントし、システム再起動時にマウントすることもできます。マウントポイントを明記する必要があることに注意してください。指定したマウントポイントが存在しない場合は、作成を求めるポップアップウィンドウが表示されます。'Edit Logical Volume' ウィンドウを以下に示します。

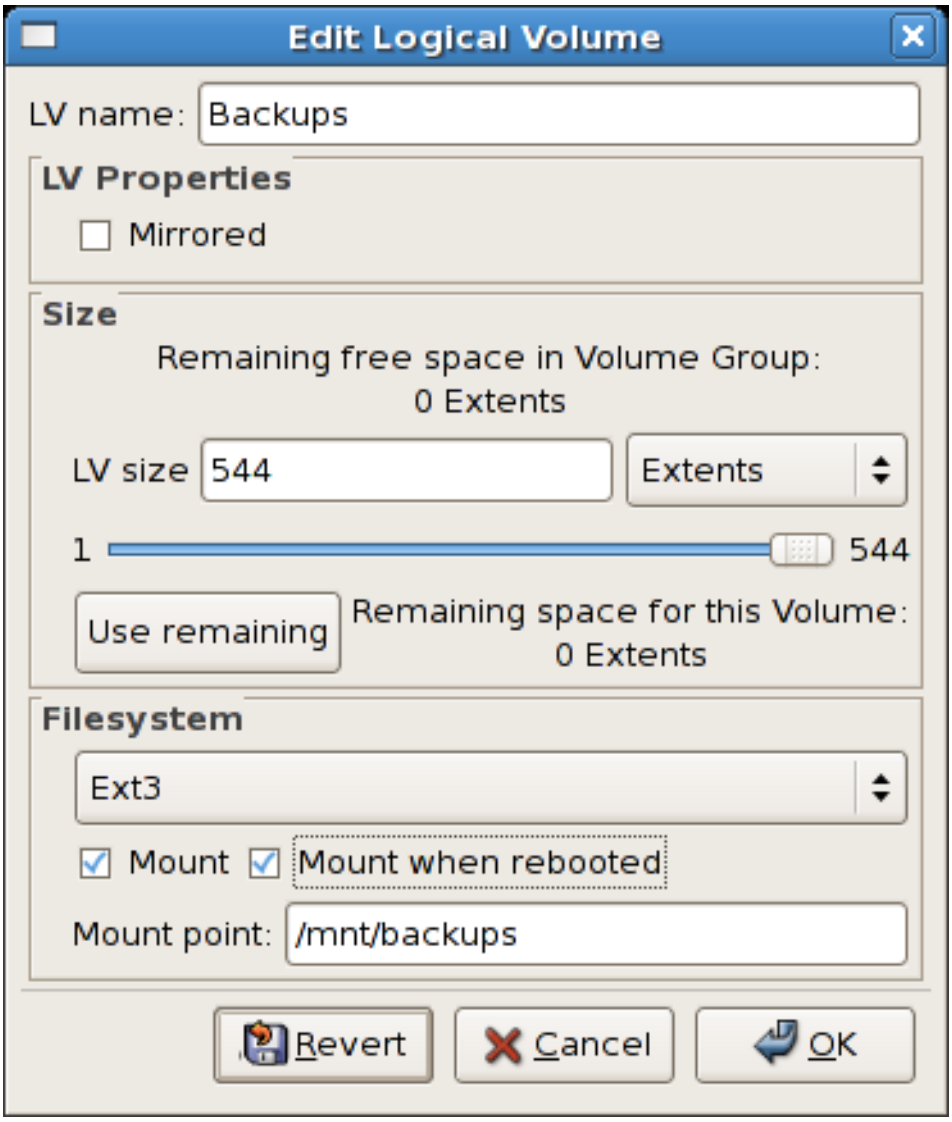
図11.31 論理ボリュームの編集



[D]

ボリュームをマウントする場合は、優先マウントポイントを示す 'Mount' チェックボックスを選択します。システムの再起動時にボリュームをマウントするには、'Mount when rebooted' のチェックボックスをオンにします。この例では、新しいボリュームは /mnt/backups にマウントされます。これを以下の図に示します。

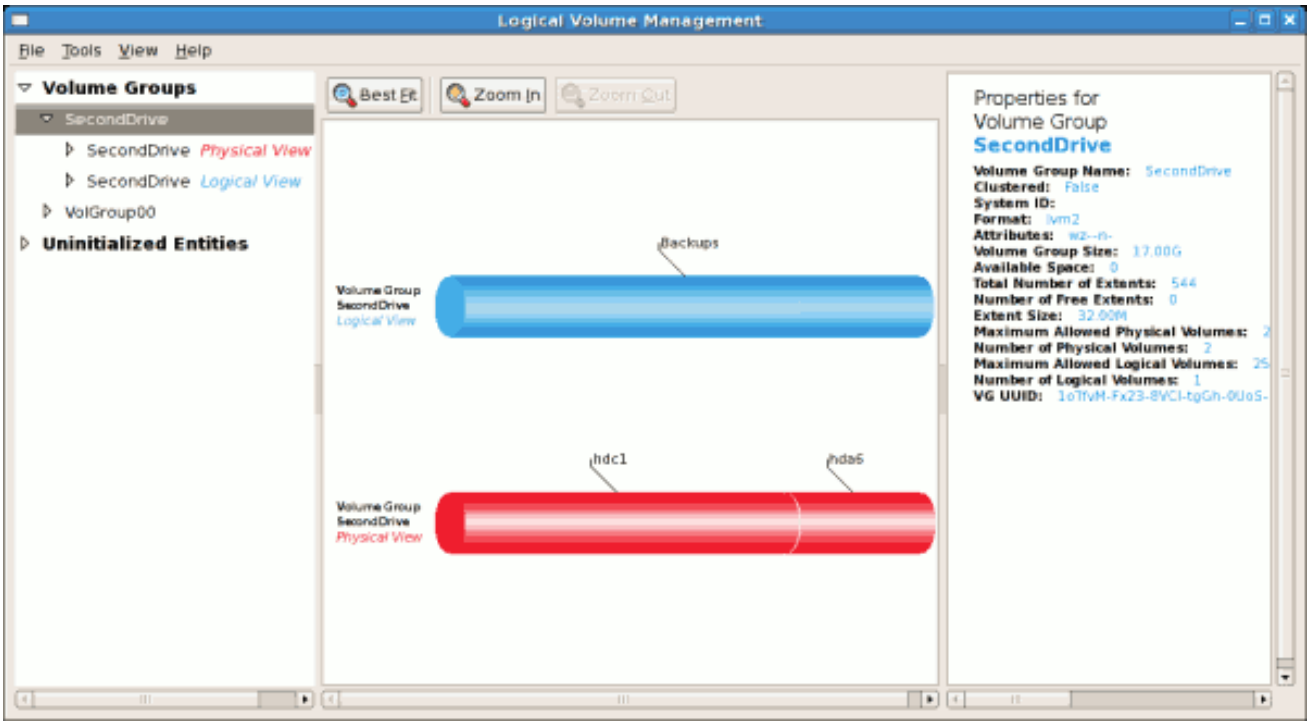
図11.32 論理ボリュームの編集 - マウントオプションの指定



[D]

以下の図は、論理ボリュームが未使用スペースに拡張された後のボリュームグループの論理ビューと物理ビューを示しています。この例では、Backups という名前の論理ボリュームが 2 つのハードディスクにまたがることに注意してください。ボリュームは、LVM を使用して 2 つ以上の物理デバイスにストライプ化できます。

図11.33 論理ボリュームの編集



[D]

11.6. 関連情報

これらのソースを使用して、LVM の詳細を確認してください。

11.6.1. インストールされているドキュメント

- rpm -qd lvm2 —このコマンドは、man ページを含む lvm パッケージから入手可能なすべてのドキュメントを表示します。
- lvm help —このコマンドは、使用可能なすべての LVM コマンドを表示します。

11.6.2. 便利な Web サイト

- http://sources.redhat.com/lvm2 — LVM2 Web ページ。概要、メーリングリストへのリンクなどが含まれています。
- http://tldp.org/HOWTO/LVM-HOWTO/ — Linux ドキュメントプロジェクトの LVM HOWTO ページ。

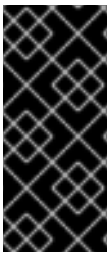
パート II. パッケージ管理

Red Hat Enterprise Linux システムのすべてのソフトウェアは、RPM パッケージに分割され、インストール、アップグレード、または削除が可能です。ここでは、グラフィカルツールおよびコマンドラインツールを使用して、Red Hat Enterprise Linux システムで RPM パッケージを管理する方法を説明します。

第12章 RPM でのパッケージ管理

RPM Package Manager (RPM)は、Red Hat Enterprise Linux やその他の Linux システムおよび UNIX システムで実行されるオープンパッケージシステムです。Red Hat, Inc. は、他のベンダーが独自の製品に RPM を使用することを推奨します。RPM は GPL の用語で配布されます。

この ユーティリティーは、rpm パッケージによって処理されるように構築されたパッケージでのみ機能します。エンドユーザーの場合、RPM によりシステムの更新が容易になります。RPM パッケージのインストール、アンインストール、およびアップグレードは、短いコマンドで実行できます。RPM は、インストールされたパッケージとそのファイルのデータベースを維持するため、システム上で強力なクエリーと検証を呼び出すことができます。グラフィカルインターフェイスを使用する場合は、Package Management Tool を使用して多くの RPM コマンドを実行できます。詳細は、13章*パッケージ管理ツール* を参照してください。



重要

パッケージをインストールする際には、お使いのオペレーティングシステムとアーキテクチャーと互換性があることを確認してください。これは通常、パッケージ名を確認して判断できます。

アップグレード時に、RPM は設定ファイルを慎重に処理するため、カスタマイズを失うことはありません（通常の .tar.gz ファイルで実行できないもの）。

開発者の場合、RPM を使用すると、ソフトウェアのソースコードを取得して、エンドユーザー向けにソースパッケージとバイナリーパッケージにパッケージ化できます。このプロセスは非常にシンプルで、1 つのファイルおよび任意のパッチから実行されます。これにより、*元* のソースとパッチ間の明確な説明と、新しいバージョンのソフトウェアがリリースされると、パッケージのメンテナーンスが容易になります。

注記

RPM はシステムに変更を加えるため、RPM パッケージのインストール、削除、またはアップグレードを行うには、root でログインする必要があります。

12.1. RPM 設計ゴール

RPM の使用方法を理解するには、RPM の設計目的を理解すると便利です。

アップグレード可能性

RPM を使用すると、完全に再インストールしなくても、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。RPM (Red Hat Enterprise Linux など)に基づくオペレーティングシステムの新しいリリースを取得する場合は、（他のパッケージシステムに基づくオペレーティングシステムと同様に）マシンに を再インストールする必要はありません。RPM を使用すると、システムのインテリジェントな完全自動化のインプレースアップグレードが可能になります。パッケージの設定ファイルはアップグレード後も保持されるため、カスタマイズを失うことはありません。パッケージをアップグレードするのに特別なアップグレードファイルは必要ありません。これは、同じ RPM ファイルを使用してシステムにパッケージをインストールおよびアップグレードするためです。

強力なクエリー

RPM は、強力なクエリーオプションを提供するように設計されています。データベース全体からパッケージの検索や、特定のファイルの検索が可能です。また、どのパッケージがパッケージに属しているかを簡単に確認することもできます。また、パッケージが存在する場所から簡単に確認することができます。RPM パッケージに含まれるファイルは圧縮アーカイブにあり、パッケージとそのコンテンツに関する有用な情報が含まれるカスタムのバイナリーヘッダーとともに、個々のパッケージにすばやく簡単にクエリーできます。

システムの検証

もう 1 つの強力な RPM 機能は、パッケージを検証する機能です。一部のパッケージで重要なファイルを削除した場合は、パッケージを確認できます。その後、異常が通知されます。その時点では、必要に応じてパッケージを再インストールできます。変更した設定ファイルは、再インストール時に保持されます。

純粋なソース

重要な設計目標は、ソフトウェアの元の作成者が配布する 初期 ソフトウェアソースを使用できるようにすることでした。RPM では、元のソースと、使用されたパッチ、完全なビルド命令があります。これは、いくつかの理由で重要な利点です。たとえば、新しいバージョンのプログラムがリリースされた場合は、コンパイルするためにゼロから開始する必要はありません。パッチを確認して、必要 な 操作を確認できます。コンパイル済みのデフォルトと、適切に構築するためにソフトウェアを正しく構築するために行われたすべての変更については、この手法を使用して簡単に確認できます。

ソースを保持する目的は、開発者にとってのみ重要であると思われるかもしれませんが、エンドユーザー向けの品質の高いソフトウェアも向上します。

12.2. RPM の使用

RPM には 5 つの基本的な動作モードがあります（パッケージビルドをカウントしない）: インス

トール、アンインストール、アップグレード、クエリー、および検証。本セクションでは、各モードの概要を説明します。詳細とオプションは、rpm --help または man rpm を試してください。RPM の詳細は、「関連情報」 を参照してください。

12.2.1. RPM パッケージの検索

RPM パッケージを使用する前に、そのパッケージの検索場所を知っている必要があります。インターネット検索は多くの RPM リポジトリーを返しますが、Red Hat が構築した RPM パッケージを検索する場合は、以下の場所にあります。

- Red Hat Enterprise Linux CD-ROM
- http://www.redhat.com/apps/support/errata/から入手できる Red Hat エラータページ
- Red Hat Network: Red Hat Network の詳細は、*15章システムの登録およびサブスクリプション管理* を参照してください。

12.2.2. インストール

RPM パッケージには、通常 foo-1.0-1.i386.rpm などのファイル名があります。ファイル名には、パッケージ名(foo)、バージョン(1.0)、リリース(1)、およびアーキテクチャー(i386)が含まれます。パッケージをインストールするには、root でログインして、シェルプロンプトで以下のコマンドを入力します。

```
rpm -ivh foo-1.0-1.i386.rpm
```

または、以下のコマンドを使用することもできます。

```
rpm -Uvh foo-1.0-1.i386.rpm
```

インストールに成功すると、以下の出力が表示されます。

```
Preparing...                ########################################### [100%]
   1:foo                    ########################################### [100%]
```

ご覧のとおり、RPM はパッケージ名を出力し、パッケージのインストール中にハッシュマークの連続を進捗メーターとして出力します。

パッケージのインストールまたはアップグレード時に、パッケージの署名が自動的にチェックされます。署名は、パッケージが承認された当事者によって署名されていることを確認します。たとえば、署名の検証に失敗すると、以下のようなエラーメッセージが表示されます。

```
error: V3 DSA signature: BAD, key ID 0352860f
```

新しいヘッダーのみの署名の場合には、以下のようなエラーメッセージが表示されます。

```
error: Header V3 DSA signature: BAD, key ID 0352860f
```

署名を検証するための適切なキーがインストールされていない場合、メッセージには以下のような NOKEY という単語が含まれます。

```
warning: V3 DSA signature: NOKEY, key ID 0352860f
```

パッケージの署名の確認に関する詳細は、「パッケージの署名の確認」 を参照してください。

WARNING

カーネルパッケージをインストールする場合は、代わりに rpm -ivh を使用する必要があります。詳細は、*44章 カーネルの手動アップグレード* を参照してください。

12.2.2.1. インストール済みパッケージの準備

同じ名前とバージョンのパッケージがすでにインストールされている場合は、以下の出力が表示されます。

```
Preparing...                ########################################### [100%]
package foo-1.0-1 is already installed
```

ただし、パッケージをインストールしたい場合は、--replacepkgs オプションを使用して、エラーを無視するように RPM に指示します。

```
rpm -ivh --replacepkgs foo-1.0-1.i386.rpm
```

このオプションは、RPM からインストールされたファイルを削除する場合や、RPM から元の設定ファイルをインストールする場合に役立ちます。

12.2.2.2. 競合するファイル

別のパッケージで既にインストールされている ファイルを含むパッケージをインストールしようとすると、以下が表示されます。

```
Preparing...                ########################################### [100%]
file /usr/bin/foo from install of foo-1.0-1 conflicts with file from package bar-2.0.20
```

RPM がこのエラーを無視させるには、--replacefiles オプションを使用します。

```
rpm -ivh --replacefiles foo-1.0-1.i386.rpm
```

12.2.2.3. 解決できない依存関係

RPM パッケージは、他のパッケージに依存する場合があります。つまり、正しく実行するために他のパッケージをインストールする必要があることを意味します。未解決の依存関係があるパッケージをインストールしようとすると、以下のような出力が表示されます。

```
error: Failed dependencies:
        bar.so.2 is needed by foo-1.0-1
Suggested resolutions:
 bar-2.0.20-3.i386.rpm
```

Red Hat Enterprise Linux CD-ROM セットからパッケージをインストールする場合は、通常、依存関係を解決するために必要なパッケージを提案します。Red Hat Enterprise Linux CD-ROM または Red Hat Network から推奨されるパッケージを見つけ、コマンドに追加します。

```
rpm -ivh foo-1.0-1.i386.rpm bar-2.0.20-3.i386.rpm
```

両方のパッケージのインストールに成功すると、以下のような出力が表示されます。

```
Preparing...                ########################################### [100%]
   1:foo                    ########################################### [ 50%]
   2:bar                    ########################################### [100%]
```

依存関係を解決するパッケージを提案しない場合は、-q --whatprovides オプションの組み合わせを試して、必要なファイルが含まれているパッケージを判断できます。

```
rpm -q --whatprovides bar.so.2
```

インストールを強制的に実行するには（パッケージが正しく実行されない可能性があるため推奨されません）、--nodeps オプションを使用します。

12.2.3. アンインストール

パッケージのインストールと同じように、パッケージをアンインストールします。シェルプロンプトで以下のコマンドを入力します。

```
rpm -e foo
```

注記

元のパッケージ ファイル foo -1.0-1.i386.rpm ではなく、パッケージ名 foo を使用していた点に注意してください。*パッケージをアンインストールするには、foo を元のパッケージの実際のパッケージ名に置き換えます。*

別のインストール済みパッケージが、削除しようとしているパッケージに依存している場合に、パッケージをアンインストールするときに依存関係エラーが発生する可能性があります。以下に例を示します。

```
error: Failed dependencies:
 foo is needed by (installed) bar-2.0.20-3.i386.rpm
```

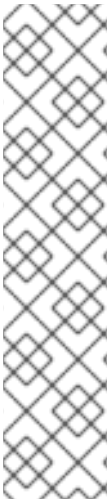
RPM はこのエラーを無視して、パッケージをアンインストール（パッケージに依存する可能性がある）をアンインストールするには、--nodeps オプションを使用します。

12.2.4. アップグレード

パッケージのインストールと類似しています。シェルプロンプトで以下のコマンドを入力します。

```
rpm -Uvh foo-2.0-1.i386.rpm
```

パッケージのアップグレードの一環として、RPM は foo パッケージの古いバージョンを自動的にアンインストールします。-U は、以前のバージョンの パッケージがインストールされていなくても、パッケージをインストールするため注意してください。



ヒント

RPM は以前のカーネルパッケージに置き換えられるため、カーネルパッケージのインストールに -U オプションを使用することは推奨されません。これは実行中のシステムに影響を与えませんが、次回の再起動時に新しいカーネルが起動できない場合は、他のカーネルが起動できなくなります。

-i オプションを使用すると、GRUB ブートメニュー(/etc/grub.conf)にカーネルが追加されます。同様に、古い不要なカーネルを削除すると、GRUB からカーネルが削除されます。

RPM は設定ファイルを使用してパッケージのインテリジェントなアップグレードを行うため、以下のようなメッセージが表示される場合があります。

```
saving /etc/foo.conf as /etc/foo.conf.rpmsave
```

このメッセージは、設定ファイルに加えられた変更は パッケージ内の新しい設定ファイルと 転送されない可能性 があるので、RPM は元のファイルを保存し、新しいファイルをインストールします。システムが適切に機能するように、2 つの設定ファイル間の違いを調査し、できるだけ早く解決する必要があります。

古い *バージョン番号（つまり、パッケージの更新バージョンがすでにインストールされている場合）を持つパッケージにアップグレードしようとすると、出力は以下のようになります。*

```
package foo-2.0-1 (which is newer than foo-1.0-1) is already installed
```

RPM を強制的にアップグレードするには、--oldpackage オプションを使用します。

```
rpm -Uvh --oldpackage foo-1.0-1.i386.rpm
```

12.2.5. Freshening

Freshening はアップグレードに似ていますが、既存のパッケージのみがアップグレードされる点が異なります。シェルプロンプトで以下のコマンドを入力します。

```
rpm -Fvh foo-1.2-1.i386.rpm
```

RPM の newen オプションは、コマンドラインで指定されたパッケージのバージョンを、システムに既にインストールされているパッケージのバージョンと照合します。すでにインストール済みのパッ

ケージの新しいバージョンが RPM の newen オプションで処理されると、新しいバージョンにアップグレードされます。ただし、RPM の newen オプションは、同じ名前のパッケージが存在しない場合には、パッケージをインストールしません。アップグレードでは、古いバージョンのパッケージがすでにインストールされているかどうかに関係なく、パッケージをインストールするため、RPM のアップグレードオプションとは異なります。

Freshening は、単一パッケージまたはパッケージグループで機能します。多数の異なるパッケージをダウンロードしたばかりで、システムにインストールされているパッケージのみをアップグレードする場合は、最新化によってジョブが実行されます。そのため、RPM を使用する前にダウンロードしたグループから不要なパッケージを削除する必要はありません。

この場合は、以下のコマンドを実行します。

```
rpm -Fvh *.rpm
```

RPM は、すでにインストールされているパッケージのみを自動的にアップグレードします。

12.2.6. クエリー

RPM データベースは、システムにインストールされているすべての RPM パッケージに関する情報を保存します。これは /var/lib/rpm/ ディレクトリーに保存され、インストールされているパッケージ、各パッケージのバージョン、およびその他からパッケージ内のファイルへの変更をクエリーするために使用されます。

このデータベースにクエリーを行うには、-q オプションを使用します。rpm -q package name コマンドは、インストールされているパッケージパッケージ名、バージョン、およびリリース番号を表示します。たとえば、rpm -q foo を使用してインストールされたパッケージ foo をクエリーすると、以下のような出力が生成される可能性があります。

```
foo-2.0-1
```

-q で以下の パッケージ選択オプション を使用して、クエリーをさらに絞り込むか、または認定することもできます。

- -a - 現在インストールされているすべてのパッケージをクエリーします。
- -f &lt ;filename&gt ; - パッケージが f <filename> を所有する RPM データベースを照会し ます。ファイルを指定する場合は、ファイルの絶対パスを指定します（例： rpm -qf /bin/ls

) 。

- *-p < packagefile> - アン インストールされたパッケージ < packagefile> にクエリーを実行し ます。*

クエリーされたパッケージについて表示する情報を指定する方法は複数あります。以下のオプションは、検索する情報の種類を選択するために使用されます。これらは パッケージクエリーオプション と呼ばれます。

- *-i は、名前、説明、リリース、サイズ、ビルド日、インストール日、ベンダーなどのパッケージ情報を表示します。*
- *-l は、パッケージに含まれるファイルの一覧を表示します。*
- *-s は、パッケージ内のすべてのファイルの状態を表示します。*
- *-d は、ドキュメントとしてマークされたファイル(man ページ、情報ページ、README など)を表示します。*
- *-c は、設定ファイルとしてマークされているファイルの一覧を表示します。インストール後に編集してシステムにパッケージ(sendmail.cf、passwd、inittab など)に合わせて編集するファイルです。*

ファイルの一覧を表示するオプションの場合は、コマンドに -v を追加して、一般的な ls -l 形式で一覧を表示します。

12.2.7. 検証中

パッケージを確認すると、パッケージからインストールされたファイルに関する情報を元のパッケージの同じ情報と比較します。特に、を確認すると、各ファイルのサイズ、MD5 合計、パーミッション、タイプ、所有者、およびグループを比較します。

コマンド rpm -V はパッケージを検証します。クエリー用にリストされた Verify Options を使用して、検証するパッケージを指定できます。検証の単純な使用は rpm -V foo です。これは、foo パッケージのすべてのファイルが、最初にインストールされたときと同じであることを確認します。以下に例を示します。

- 特定のファイルを含むパッケージを確認するには、次のコマンドを実行します。

```
rpm -Vf /usr/bin/foo
```

この例では、/usr/bin/foo は、パッケージのクエリーに使用されるファイルへの絶対パスです。

- システム全体にインストールされているパッケージをすべて確認するには、次のコマンドを実行します。

```
rpm -Va
```

- RPM パッケージファイルに対してインストールされたパッケージを確認するには、次のコマンドを実行します。

```
rpm -Vp foo-1.0-1.i386.rpm
```

このコマンドは、RPM データベースが破損していると思われる場合に役立ちます。

すべてが適切に検証された場合は、出力はありません。不一致がある場合は、表示されます。出力の形式は 8 文字(c は設定ファイルを表す)の文字列で、ファイル名になります。8 文字は、ファイルの 1 つの属性を、RPM データベースで記録した属性の値と比較した結果を示します。シングルピリオド(.)は、テストに合格することを意味します。以下の文字は、特定の不一致を示しています。

- 5 - MD5 チェックサム
- s - ファイルサイズ
- l - シンボリックリンク
- t - ファイルの変更時間
- d - device

- *u - user*
- *g - グループ*
- *m - モード（パーミッションとファイルタイプを含む）*
- *?: 読み取りできないファイル*

出力が表示された場合は、最適な判断でパッケージを削除し、再インストールするか、別の方法で問題を修正する必要があるかどうかを判断します。

12.3. パッケージの署名の確認

パッケージが破損していない、または改ざんされていないことを確認する場合は、シェルプロンプトで以下のコマンドを入力して md5sum のみを調べます。< rpm-file > は RPM パッケージのファイル名に置き換えます。

```
rpm -K --nosignature <rpm-file>
```

メッセージ &lt ;rpm-file>: md5 OK が表示されます。この簡単なメッセージは、ダウンロードによってファイルが破損していないことを意味します。詳細なメッセージを表示するには、コマンドで -K を -Kvv に置き換えます。

一方、信頼できるのは、このパッケージを作成した開発者がどのようなものですか ?パッケージが開発者の GnuPG キー で 署名 されている場合、開発者が実際にその人であることが分かります。

RPM パッケージは、ダウンロードしたパッケージを確実に信頼できるものにするために、Gnu Privacy Guard （または GnuPG）を使用して署名できます。

GnuPG はセキュアな通信を行うためのツールです。これは、電子プライバシープログラムである PGP の暗号化技術を完全かつ無料で置き換えています。GnuPG を使用すると、ドキュメントの有効性を認証し、他の受信者との間でデータを暗号化/復号化できます。GnuPG は PGP 5 のx ファイルも復号化および検証できます。

インストール時に、GnuPG はデフォルトでインストールされます。これにより、GnuPG をすぐに

使用し、Red Hat から受け取ったパッケージを検証することができます。この作業を行う前に、まず Red Hat の公開鍵をインポートする必要があります。

12.3.1. キーのインポート

Red Hat パッケージを確認するには、Red Hat GPG キーをインポートする必要があります。これを行うには、シェルプロンプトで以下のコマンドを実行します。

```
rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
```

RPM 検証用にインストールされた鍵の一覧を表示するには、以下のコマンドを実行します。

```
rpm -qa gpg-pubkey*
```

Red Hat キーの場合、出力には以下が含まれます。

```
gpg-pubkey-37017186-45761324
```

特定のキーの詳細を表示するには、rpm -qi の後に直前のコマンドの出力を使用します。

```
rpm -qi gpg-pubkey-37017186-45761324
```

12.3.2. パッケージの署名の確認

ビルダーの GnuPG キーをインポートした後に RPM ファイルの GnuPG 署名を確認するには、以下のコマンドを使用します(< rpm-file > は RPM パッケージのファイル名に置き換えてください)。

```
rpm -K <rpm-file>
```

すべてが正常に終了すると、md5 gpg OK というメッセージが表示されます。これは、パッケージの署名が検証され、破損していないことを意味します。

12.4. RPM 使用率の実用的な例および一般的な例

RPM は、システムの管理と問題の診断と修正の両方に役立つツールです。すべてのオプションを把握する最適な方法は、いくつかの例を確認することです。

-

誤って一部のファイルを削除したが、削除したものは確認されていない可能性があります。システム全体を確認し、何が欠落しているかを確認するには、次のコマンドを試してください。

```
rpm -Va
```

一部のファイルが見つからないか、破損しているように見える場合は、パッケージを再インストールするか、アンインストールしてから、パッケージを再インストールする必要があります。

- *場合によっては、認識していないファイルが表示される場合があります。所有しているパッケージを検索するには、次のコマンドを実行します。*

```
rpm -qf /usr/bin/ggv
```

出力は以下のようになります。

```
ggv-2.6.0-2
```

- *以下のシナリオで、上記の 2 つの例を組み合わせることができます。/usr/bin/paste に問題があるとします。そのプログラムを所有するパッケージを確認しますが、どのパッケージを所有するかを確認して ください。以下のコマンドを入力します。*

```
rpm -Vf /usr/bin/paste
```

さらに、適切なパッケージが検証されます。

- *特定のプログラムの詳細情報を調べるか ?以下のコマンドを試して、そのプログラムを所有するパッケージに含まれるドキュメントを見つけることができます。*

```
rpm -qdf /usr/bin/free
```

出力は以下のようになります。

```
/usr/share/doc/procps-3.2.3/BUGS
/usr/share/doc/procps-3.2.3/FAQ
/usr/share/doc/procps-3.2.3/NEWS
/usr/share/doc/procps-3.2.3/TODO
/usr/share/man/man1/free.1.gz
```

```
/usr/share/man/man1/pgrep.1.gz
/usr/share/man/man1/pkill.1.gz
/usr/share/man/man1/pmap.1.gz
/usr/share/man/man1/ps.1.gz
/usr/share/man/man1/skill.1.gz
/usr/share/man/man1/slabtop.1.gz
/usr/share/man/man1/snice.1.gz
/usr/share/man/man1/tload.1.gz
/usr/share/man/man1/top.1.gz
/usr/share/man/man1/uptime.1.gz
/usr/share/man/man1/w.1.gz
/usr/share/man/man1/watch.1.gz
/usr/share/man/man5/sysctl.conf.5.gz
/usr/share/man/man8/sysctl.8.gz
/usr/share/man/man8/vmstat.8.gz
```

- 新しい RPM が見つかる場合もありますが、何が起こっているかは分からません。その情報を見つけるには、次のコマンドを使用します。

```
rpm -qip crontabs-1.10-7.noarch.rpm
```

出力は以下のようになります。

```
Name        : crontabs                     Relocations: (not relocatable)
Version     : 1.10                           Vendor: Red Hat, Inc.
Release     : 7                         Build Date: Mon 20 Sep 2004 05:58:10 PM EDT
Install Date: (not installed)               Build Host: tweety.build.redhat.com
Group       : System Environment/Base       Source RPM: crontabs-1.10-7.src.rpm
Size        : 1004                             License: Public Domain
Signature   : DSA/SHA1, Wed 05 Jan 2005 06:05:25 PM EST, Key ID 219180cddb42a60e
Packager    : Red Hat, Inc. <http://bugzilla.redhat.com/bugzilla>
Summary     : Root crontab files used to schedule the execution of programs.
Description : The crontabs package contains root crontab files. Crontab is the
program used to install, uninstall, or list the tables used to drive the
cron daemon. The cron daemon checks the crontab files to see when
particular commands are scheduled to be executed. If commands are
scheduled, then it executes them.
```

- おそらく、crontabs RPM によってインストールされるファイルを確認します。以下を入力します。

```
rpm -qlp crontabs-1.10-5.noarch.rpm
```

出力は以下の例のようになります。

```
/etc/cron.daily
```

```
/etc/cron.hourly
/etc/cron.monthly
/etc/cron.weekly
/etc/crontab
/usr/bin/run-parts
```

これらはいくつかの例です。RPM を使用する場合、使用が多くなる場合があります。

12.5. 関連情報

RPM は、パッケージのクエリー、インストール、アップグレード、および削除を行う多くのオプションとメソッドを備えた、非常に複雑なユーティリティーです。RPM の詳細は、以下のリソースを参照してください。

12.5.1. インストールされているドキュメント

- rpm --help - このコマンドは、RPM パラメーターのクイックリファレンスを表示します。
- man rpm: RPM の man ページには、rpm --help コマンドよりも RPM パラメーターに関する詳細が記載されています。

12.5.2. 便利な Web サイト

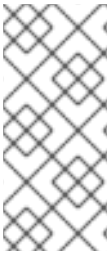
- http://www.rpm.org/ - RPM の Web サイトです。
- https://lists.rpm.org/mailman/listinfo/rpm-list - このリンクにアクセスして、アーカイブされた RPM メーリングリストにサブスクライブします。

12.5.3. 関連書籍

- Eric Foster-Johnson の 『Red Hat RPM ガイド』 は、RPM パッケージ形式と RPM パッケージ管理ユーティリティーのすべての詳細に関する優れたリソースです。でオンラインで http://docs.fedoraproject.org/drafts/rpm-guide-en/ 使用できます。

第13章 パッケージ管理ツール

グラフィカルインターフェイスを使用してシステムのパッケージを表示および管理する場合は、パッケージ 管理ツール (pirut と呼ばれる)を使用できます。このツールを使用すると、インストールしたパッケージを削除したり、システムに互換性のあるパッケージをダウンロード（およびインストール）したりする、使いやすいインターフェイスを使用して、システムの基本パッケージ管理を実行できます。また、システムにインストールされているパッケージと、Red Hat Network からダウンロードできるパッケージを確認できます。さらに、rpm コマンドと同じ方法でパッケージをインストールまたは削除すると、Package Management Tool は重要な依存関係を自動的に解決します。

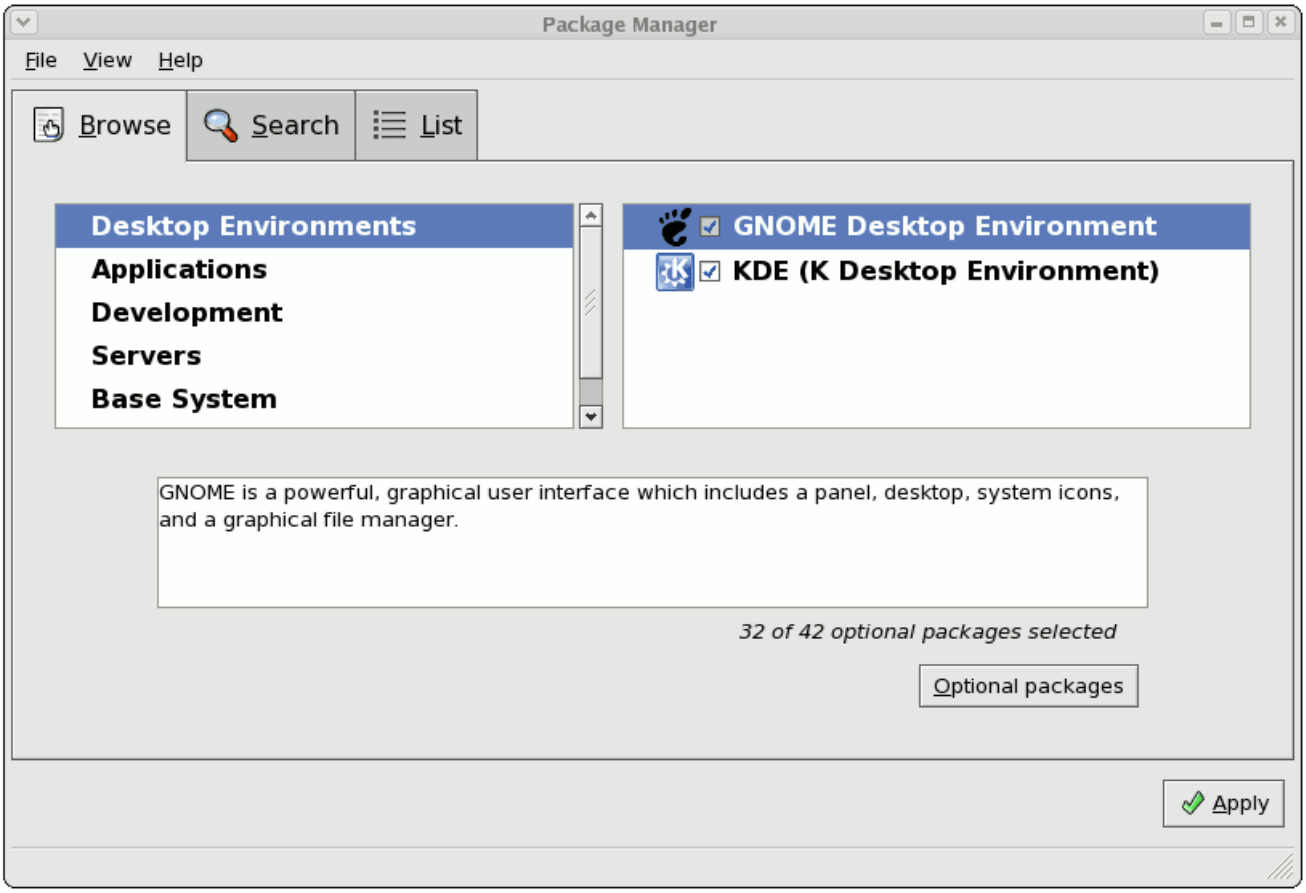


注記

Package Management Tool はパッケージのインストールおよび削除時に依存関係を自動的に解決できますが、rpm -e --nodeps または rpm -U --nodeps と同じ方法で強制的にインストール/削除することはできません。

Package Management Tool を実行するには、X Window System が必要です。アプリケーションを起動するには、Applications （パネルのメインメニュー）> Add/Remove Software に移動します。または、シェルプロンプトでコマンド system-config-packages または pirut を入力することもできます。

図13.1 パッケージ管理ツール



[D]

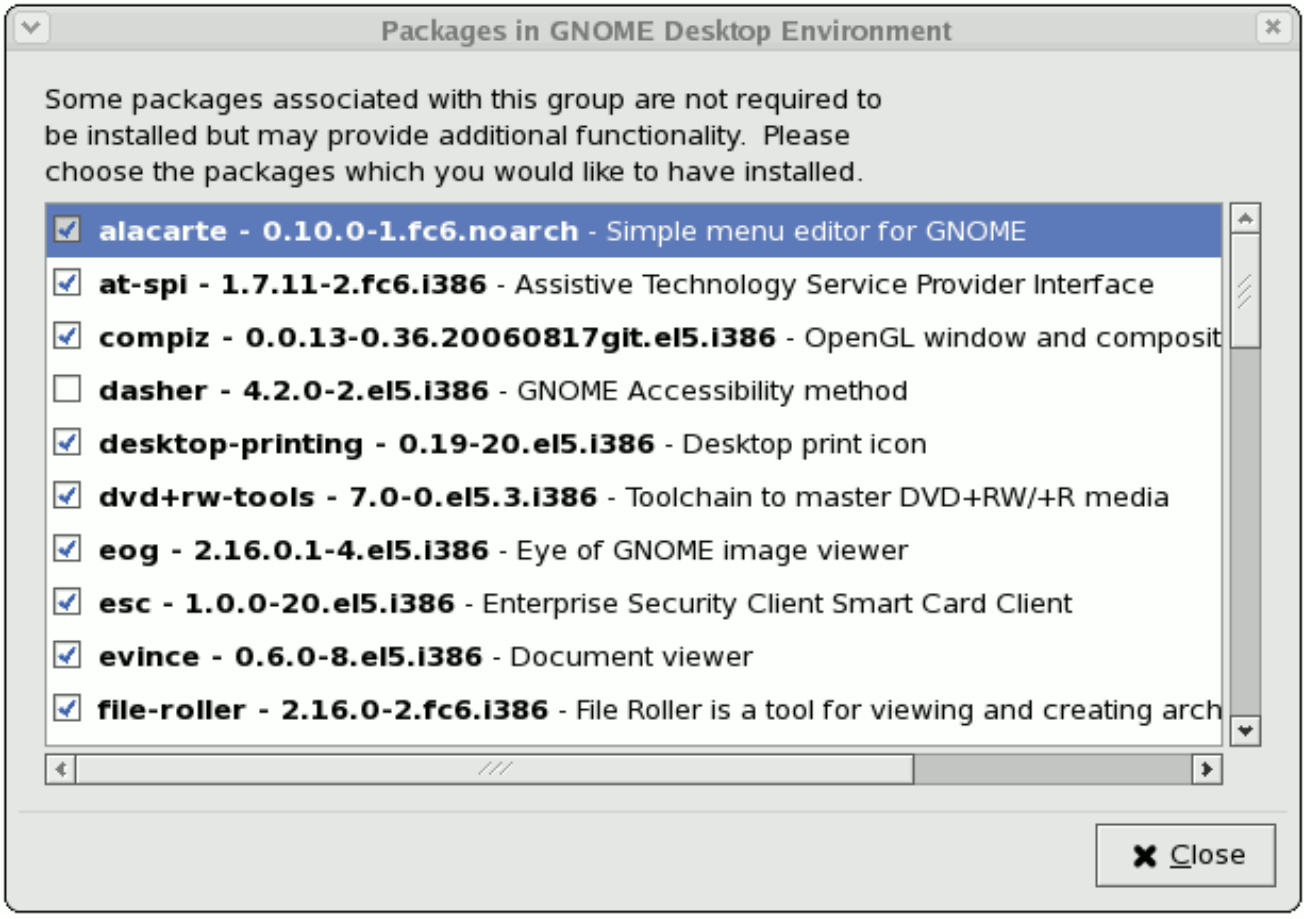
13.1. パッケージの一覧表示および分析

Package Management Tool を使用すると、システムにインストールされているすべてのパッケージ、およびダウンロード可能なパッケージを検索および一覧表示できます。Browse、Search、および List タブには、パッケージの表示、分析、インストール、または削除に関するさまざまなオプションがあります。

Browse タブでは、グループ別にパッケージを表示できます。図13.1「パッケージ管理ツール」 の左側のウィンドウには、選択できるさまざまなパッケージグループタイプが表示されます（例：デスクトップ環境、アプリケーション、開発など）。パッケージグループタイプを選択すると、右側のウィンドウにそのタイプの異なるパッケージグループが表示されます。

パッケージグループに含まれるパッケージを表示するには、Optional packages をクリックします。インストールされているパッケージがチェックされます。

図13.2 オプションパッケージ



[D]

List タブには、インストールされているパッケージ、またはダウンロード可能なパッケージの一覧が表示されます。システムにすでにインストールされているパッケージには、緑色のチェック(

)のマークが付けられます。

デフォルトでは、メインウィンドウの上の All パッケージ オプションが選択されます。これにより、すべてのパッケージが表示されることを指定します。Installed packages オプションを使用して、システムにインストールされているパッケージのみを表示し、Available packages オプションを使用して、ダウンロードおよびインストールできるパッケージを表示します。

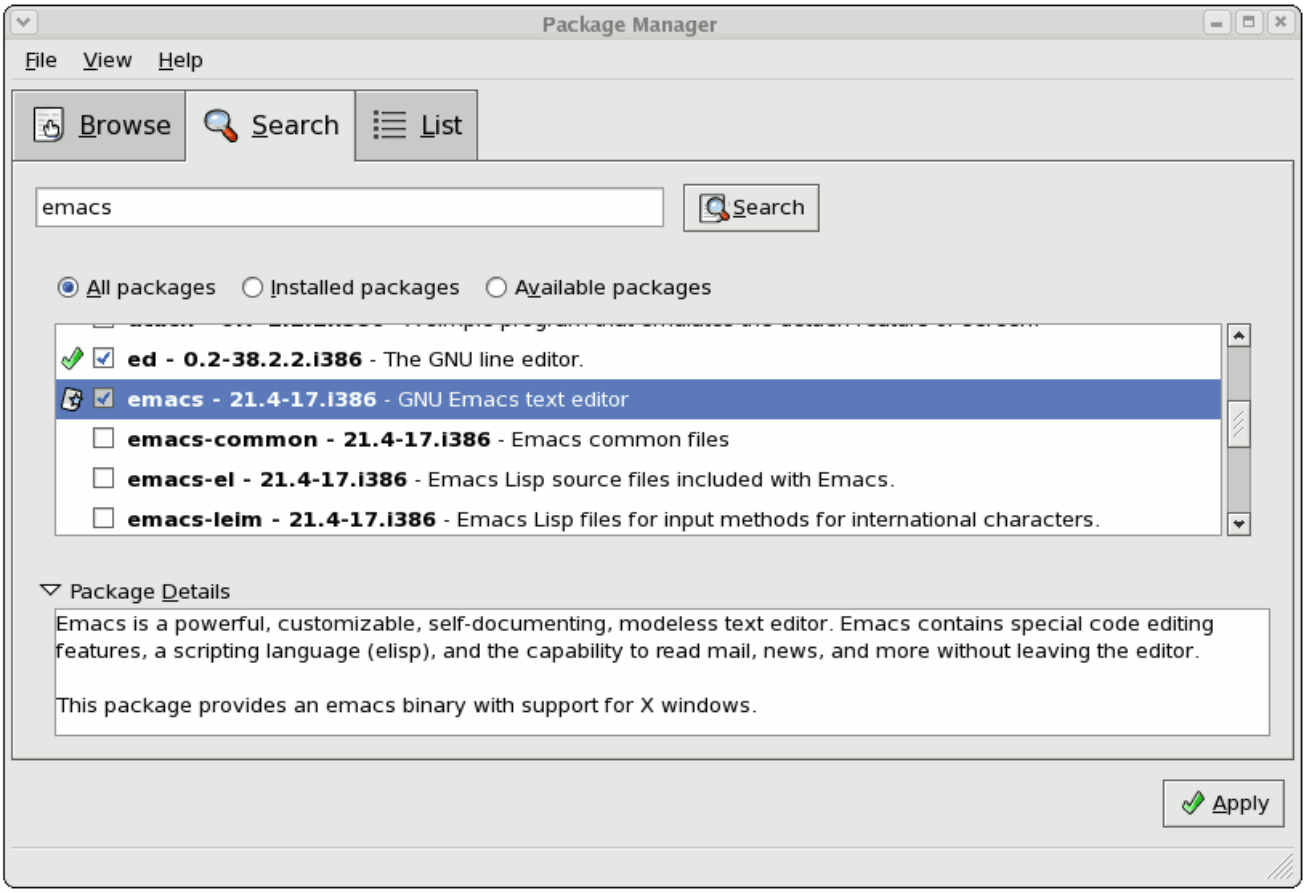
Search タブでは、キーワードを使用して特定のパッケージを検索できます。このタブでは、パッケージの簡単な説明を表示することもできます。これを行うには、パッケージを選択し、メインウィンドウの下にある パッケージ Details ボタンをクリックします。

13.2. パッケージのインストールと削除

ダウンロード可能なパッケージをインストールするには、パッケージ名の横にあるチェックボックスをクリックします。これを実行すると、そのチェックボックスの横にインストールアイコン(

)が表示されます。これは、パッケージがダウンロードおよびインストールのためにキューにあることを示しています。ダウンロードおよびインストールするパッケージを複数選択できます。選択したら、**Apply** ボタンをクリックします。

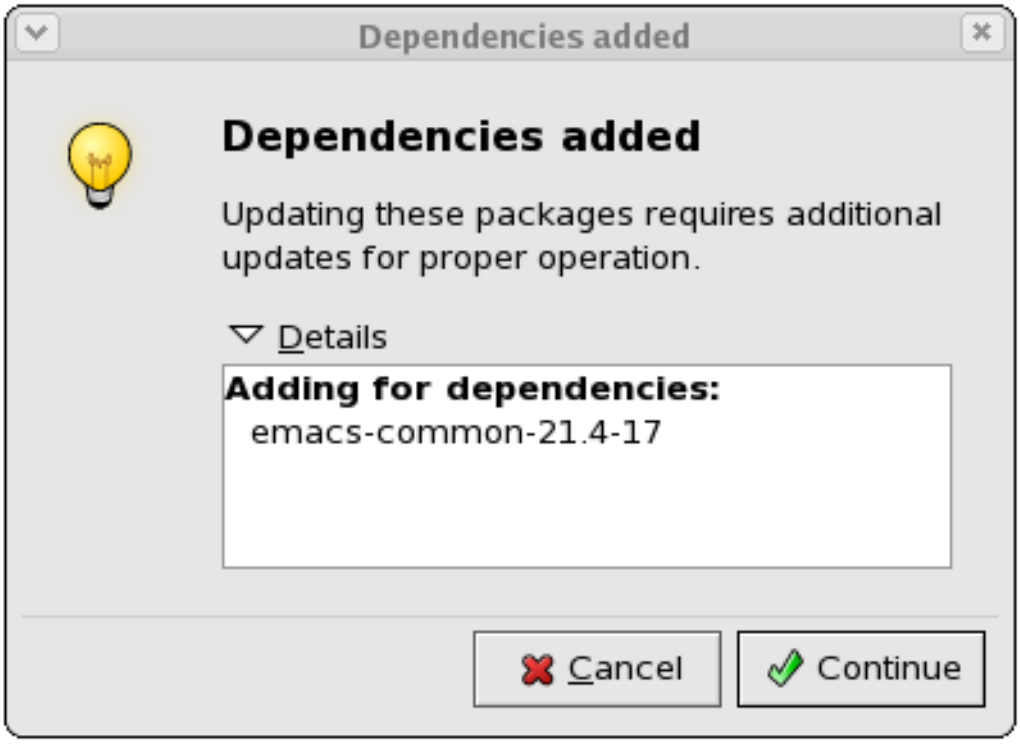
図13.3 パッケージのインストール



[D]

選択したダウンロードにパッケージの依存関係がある場合は、**Package Management Tool** がそれに応じて通知します。**Details** をクリックして、必要なパッケージを確認します。（他のすべての依存パッケージとともに）パッケージのダウンロードとインストールを続行するには、**C**をクリックします。

図13.4 パッケージの依存関係：インストール

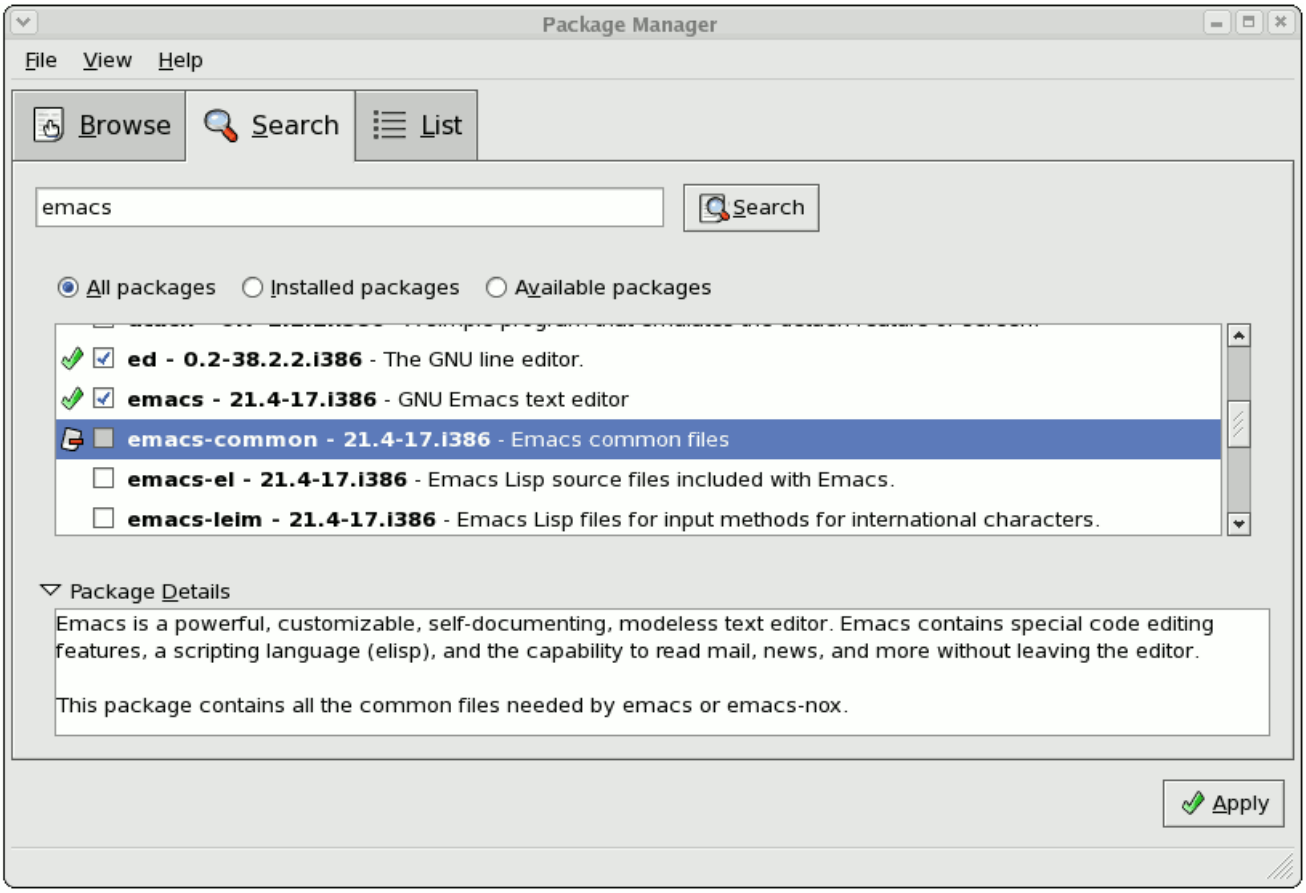


[D]

パッケージの削除は、同様の方法で実行できます。システムにインストールされているパッケージを削除するには、パッケージ名の横にあるチェックボックスをクリックします。パッケージ名の横に表示される緑色のチェックは、パッケージ削除アイコン(

*)に置き換えられます。これは、パッケージが削除のためにキューに入れられていることを示しています。同時に削除する複数のパッケージを選択することもできます。削除するパッケージを選択したら、**Apply** ボタンをクリックします。*

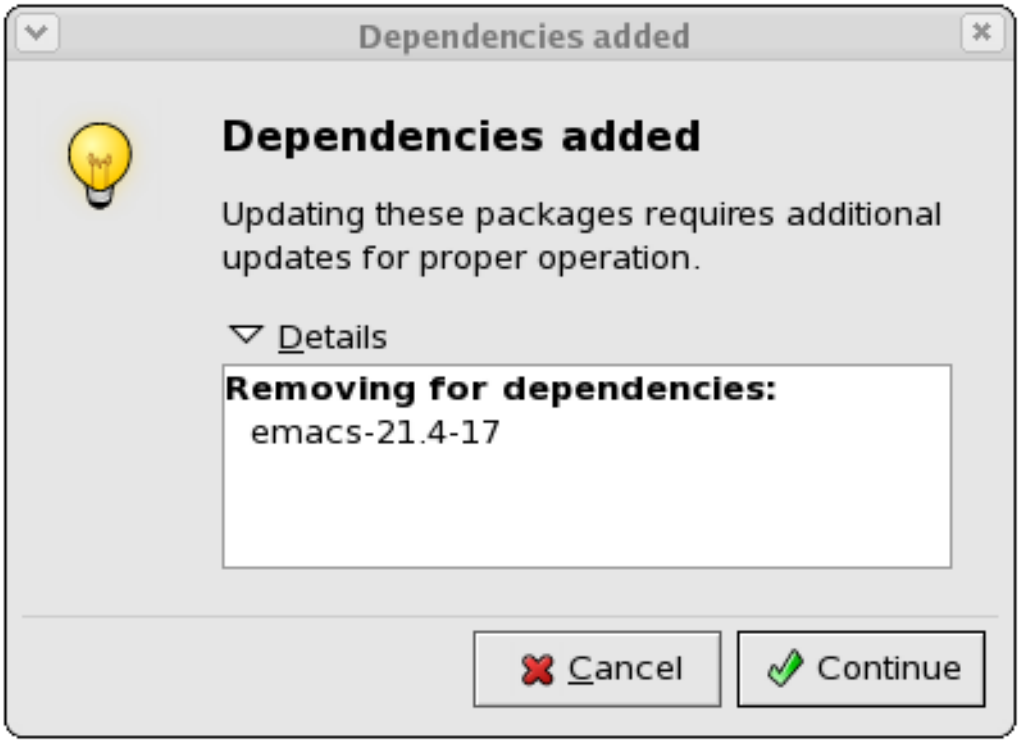
図13.5 パッケージの削除



[D]

インストールされているその他のパッケージが、削除するパッケージに依存している場合は、それらも削除されます。このような依存関係がある場合は、Package Management Tool が通知します。Details をクリックして、削除するパッケージに依存しているパッケージを表示します。選択したパッケージ/（およびその他の依存パッケージすべてとともに）の削除を続行するには、Cをクリックします。

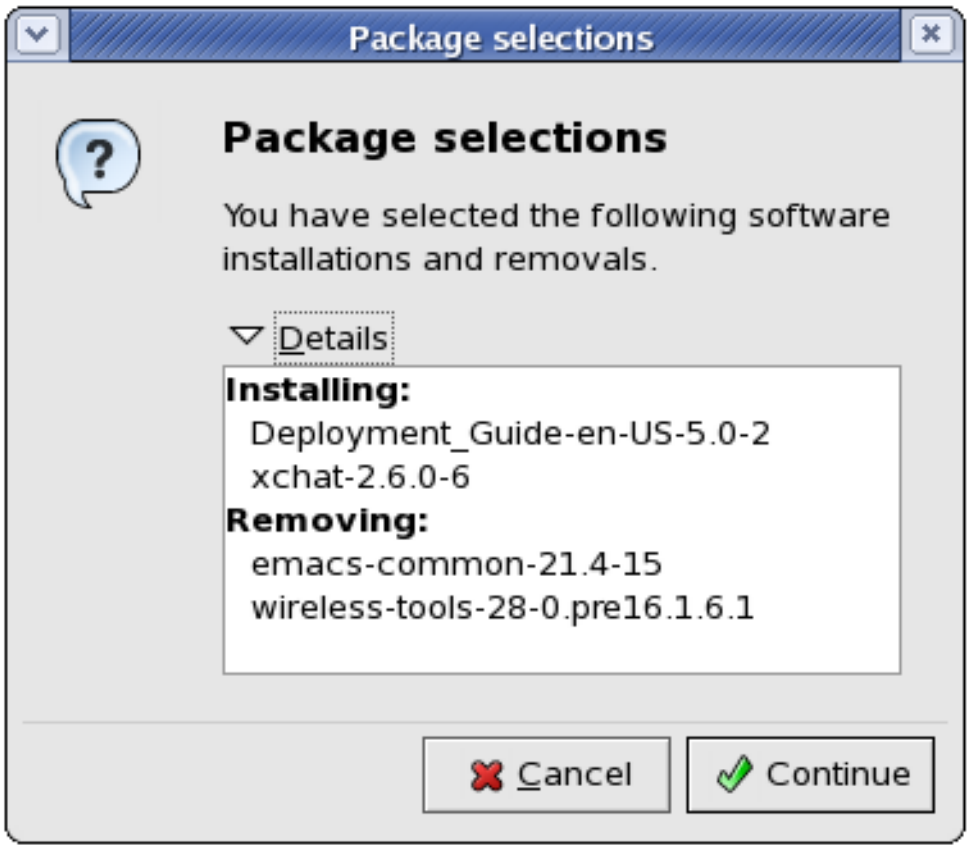
図13.6 パッケージの依存関係：削除



[D]

インストール/削除するパッケージを選択してから **Apply** *をクリックして、複数のパッケージをインストールおよび削除できます。パッケージの 選択 ウィンドウには、インストールおよび削除パッケージの数が表示されます。*

図13.7 パッケージの同時インストールと削除



[D]

第14章 YUM (YELLOWDOG UPDATER MODIFIED)

yum (YellowDog Update)は、 RPM のインストールを改善するために Dukevideo によって開発されたパッケージマネージャーです。yum は、多くのリポジトリーでパッケージとその依存関係を検索するため、依存関係の問題を軽減するために一緒にインストールできます。Red Hat Enterprise Linux 5.10 は yum を使用してパッケージを取得し、RPM をインストールします。

yum が推奨されるため、up2date が非推奨になりました(Yellowdog Updater Modified)。Red Hat Enterprise Linux 5.10 にソフトウェアをインストールし、更新するツールのスタック全体が yum をベースとするようになりました。これには、Anaconda を介した初期インストールから pirut などのソフトウェア管理ツールへのすべてが含まれます。

Yum を使用すると、システム管理者は、ローカルの（つまり、ローカルネットワーク上で利用可能）リポジトリーを設定して、Red Hat が提供するパッケージを補完することができます。これは、Red Hat が公式にサポートしていないアプリケーションおよびパッケージを使用するユーザーグループに便利です。

ローカルの yum リポジトリーを使用すると、ローカルユーザーが利用できるパッケージを補完できるだけでなく、ネットワーク全体の帯域幅も節約されます。さらに、ローカルの yum リポジトリーを使用するクライアントでは、Red Hat Network から最新のパッケージをインストールまたは更新するために個別に登録する必要はありません。

14.1. YUM リポジトリーの設定

Red Hat Enterprise Linux パッケージのリポジトリーを設定するには、以下の手順に従います。

1. createrepo パッケージをインストールします。

```
~]# yum install createrepo
```

2. リポジトリーに指定するすべてのパッケージを 1 つのディレクトリー（例：/mnt/local_repo ）にコピーします。

3. そのディレクトリー（例： createrepo /mnt/local_repo）で createrepo を実行します。これにより、Yum リポジトリーに必要なメタデータが作成されます。

14.2. YUM コマンド

yum コマンドは通常 、yum <command> <package name/s> として実行され ます。デフォルトでは、yum は、インストール/アップグレード時にすべてのパッケージ依存関係を解決するように設定されたすべてのリポジトリーを自動的に確認しようとします。

以下は、最も一般的に使用される yum コマンドの一覧です。利用可能な yum コマンドの完全なリストは、man yum を参照してください。

yum install & lt;package name/s>

パッケージまたはパッケージグループの最新バージョンをインストールするために使用されます。指定したパッケージ名に一致するパッケージがない場合は、シェルグロブであると想定され、一致するものがインストールされます。

yum update & lt;package name/s>

指定したパッケージを利用可能な最新バージョンに更新するために使用されます。パッケージ名や指定がない場合は、yum はインストール済みパッケージの更新を試みます。

--obsoletes オプションが使用されている場合（つまり、yum --obsoletes < package name/s >）、yum は廃止されたパッケージを処理します。そのため、更新間で廃止されるパッケージは削除され、それに応じて置き換えられます。

yum check-update

このコマンドを使用すると、インストール済みパッケージで利用可能な更新を確認できます。yum は、すべてのリポジトリーからパッケージの更新が利用可能な場合に一覧を返します。

yum remove & lt;package name/s>

指定したパッケージと、削除されるパッケージに依存するその他のパッケージを削除するために使用されます。

yum provides &lt ;file name>

特定のファイルまたは機能を提供するパッケージを決定するために使用されます。

yum search <keyword>

このコマンドは、すべてのリポジトリーの RPM の説明、要約、パッケージャー、およびパッケージ名フィールドで指定されたキーワードを含むパッケージを見つけるために使用されます。

yum localinstall & lt;absolute path to package name/s>

yum を使用して、マシン内にローカルにあるパッケージをインストールする場合に使用されます。

14.3. YUM オプション

yum オプションは通常、特定の yum コマンドの前に記載されます（例 : yum <options> <command> <package name/s&gt ;）。これらのオプションのほとんどは、設定ファイルを使用してデフォルトとして設定できます。

以下は、最も一般的に使用される yum オプションの一覧です。利用可能な yum オプションの完全なリストは、man yum を参照してください。

-y

トランザクションのすべての質問に yes と回答します。

-t

トランザクションで指定されたパッケージに関するエラーに対するエラーの許容性が yum に設定します。たとえば、yum update package1 package2 を実行し、package2 がすでにインストールされている場合は、yum は引き続き package1 をインストールします。

--exclude=<package name>

特定のトランザクションで、名前または glob で特定のパッケージを除外します。

14.4. YUMの設定

デフォルトでは、yum は /etc/yum.conf で設定されます。一般的な /etc/yum.conf ファイルの例を以下に示します。

```
[main]
cachedir=/var/cache/yum
keepcache=0
debuglevel=2
logfile=/var/log/yum.log
distroverpkg=redhat-release
tolerant=1
exactarch=1
```

```
obsoletes=1
gpgcheck=1
plugins=1
metadata_expire=1800
[myrepo]
name=RHEL 5 $releasever - $basearch
baseurl=http://local/path/to/yum/repository/
enabled=1
```

一般的な /etc/yum.conf ファイルは、[main] セクションとリポジトリーセクションという 2 種類のセクションで設定されています。[main] セクションは 1 つだけ指定できますが、1 つの /etc/yum.conf 内に複数のリポジトリーを指定できます。

14.4.1. [main] オプション

[main] セクションは必須であり、必要なセクションは 1 つのみです。[main] セクションで使用できるオプションの完全リストは、man yum.conf を参照してください。

以下は、[main] セクションで最も一般的に使用されるオプションの一覧です。

cachedir

このオプションは、yum がキャッシュとデータベースファイルを保存するディレクトリーを指定します。デフォルトでは、yum のキャッシュディレクトリーは /var/cache/yum です。

keepcache= <1 or 0>

keepcache=1 を設定すると、インストールの成功後もヘッダーとパッケージのキャッシュを維持するように yum に指示します。keepcache=1 がデフォルトです。

reposdir= <absolute path to directory of .repo files>

このオプションを使用すると、.repo ファイルが置かれているディレクトリーを指定できます。.repo ファイルには、リポジトリー情報が含まれます(/etc/yum.confの [repository] セクションと同様)。

yum は、.repo ファイルおよび /etc/yum.conf ファイルの [repository] セクションからすべてのリポジトリー情報を収集して、各トランザクションに使用するリポジトリーのマスターリストを作成します。[repository] セクションと .repo ファイルの両方に使用できるオプションの詳細は、「[repository] Options」 を参照してください。

reposdir が設定されていない場合、yum はデフォルトのディレクトリー /etc/yum.repos.d を

使用します。

gpgcheck=<1 or 0>

これにより、ローカルパッケージのインストールなど、全リポジトリーのパッケージで GPG 署名の確認を無効または有効にします。デフォルトは gpgcheck=0 で、GPG チェックを無効にします。

このオプションが /etc/yum.conf ファイルの [main] セクションで設定されている場合は、すべてのリポジトリーの GPG チェックルールが設定されます。ただし、これを個別のリポジトリーに設定することもできます。つまり、あるリポジトリーで GPG チェックを有効にしつつ、別のリポジトリーで無効にすることもできます。

assumeyes= <1 or 0>

これにより、yum が重要なアクションの確認を要求するかどうかを決定します。assumeyes=0 の場合のデフォルト。これは、yum により確認を求めるプロンプトが表示されることを意味します。

assumeyes=1 を設定すると、yum はコマンドラインオプション -y と同じように動作します。

tolerant= <1 または 0>

有効にすると(tolerant=1)、yum はパッケージに関してコマンドラインでエラーを許容します。これは、yum コマンドラインオプション -t と似ています。

このデフォルト値は tolerant=0 （トレラントではありません）です。

exclude=<package name/s>

このオプションを使用すると、インストール/更新時にキーワードでパッケージを除外できます。複数のパッケージを指定する場合は、スペースで区切られたリストになります。ワイルドカードを使用したシェル glob (* や ? など)を使用できます。

retries= <number of retries>

これにより、エラーを返す前に yum がファイルの取得を試行する回数が設定されます。これを 0 に設定すると、yum は永久に再試行されます。デフォルト値は 6 です。

14.4.2. [repository] Options

/etc/yum.conf ファイルの [repository] セクションには、yum がパッケージのインストール、更新、および依存関係の解決時にパッケージを検索するために使用できるリポジトリーに関する情報が含まれます。リポジトリーエントリーは以下の形式になります。

```
[repository ID]
name=repository name
baseurl=url, file or ftp://path to repository
```

別の .repo ファイルにリポジトリー情報を指定することもできます（例： rhel5.repo）。.repo ファイルに置かれたリポジトリー情報の形式は、/etc/yum.conf の [repository] と同じです。

.repo ファイルは、reposdir= を使用して /etc/yum.conf の [main] セクションで別のリポジトリーパスを指定しない限り、通常は /etc/yum.repos.d に配置されます。.repo ファイルと /etc/yum.conf ファイルには、複数のリポジトリーエントリーを含めることができます。

各リポジトリーエントリーは、以下の必須部分で設定されます。

[repository ID]

リポジトリー ID は、リポジトリー識別子として機能する一意の 1 単語の文字列です。

name=repository name

これは、人間が判読可能な、リポジトリーを記述する文字列です。

baseurl=http、file、または ftp://path

これは、リポジトリーの repodataディレクトリーが置かれているディレクトリーへの URL です。リポジトリーがマシンのローカルにある場合は、ローカルリポジトリー への baseurl=file:// パスを 使用します。HTTP を使用してリポジトリーがオンラインにある場合は、baseurl=http://リンク を使用します。リポジトリーがオンラインで FTP を使用する場合は、baseurl=ftp://リンク を使用します。

特定のオンラインリポジトリーで基本的な HTTP 認証が必要な場合は、username: password @link として追加して、baseurl 行でユーザー名とパスワードを指定できます。たとえば、のリポジトリーで http://www.example.com/repo/ user というユーザー名とパスワード os password が必要な場合、baseurl リンクは baseurl=http://user:/repo/ として指定できます。

以下は、リポジトリーエントリーで最も一般的に使用されるオプションの一覧です。リポジトリーエントリーの完全なリストは、man yum.conf を参照してください。

gpgcheck=<1 or 0>

これにより、特定のリポジトリーの確認を GPG 署名を無効化/有効にします。デフォルトは gpgcheck=0 で、GPG チェックを無効にします。

gpgkey=URL

このオプションを使用すると、リポジトリーの ASCII アラームされた GPG キーファイルの URL を参照できます。このオプションは、通常、yum がパッケージの検証に公開鍵を必要とし、必要な鍵が RPM データベースにインポートされていない場合に使用されます。

このオプションを設定すると、yum は指定された URL からキーを自動的にインポートします。assumeyes=1 (/etc/yum.confの [main] セクションに)または -y (yum トランザクション)を設定しない限り、キーをインストールする前にプロンプトが表示されます。

exclude=<package name/s>

このオプションは、/etc/yum.conf の [main] セクションの exclude オプションと似ています。ただし、指定したリポジトリーにのみ適用されます。

includepkgs=<package name/s>

このオプションは、除外の とは異なり ます。このオプションをリポジトリーに設定すると、yum はそのリポジトリーで指定されたパッケージのみを表示できます。デフォルトでは、リポジトリー内の全パッケージが yum に表示されます。

14.5. ISO と YUM を使用してシステムをオフラインでアップグレード

インターネットまたは Red Hat Network から切断されたシステムの場合は、yum update コマンドを Red Hat Enterprise Linux インストール ISO イメージで使用すると、システムを最新のマイナーバージョンに簡単かつ迅速にアップグレードできます。以下の手順はアップグレードプロセスを示しています。

1. ISO イメージをマウントするターゲットディレクトリーを作成します。このディレクトリーはマウント時に自動的に作成されないため、root で次の手順に進む前に作成します。

```
mkdir mount_dir
```

mount_dir をマウントディレクトリーへのパスに置き換えます。Typicaly は、/media/ ディレクトリーにサブディレクトリーとして作成します。

2.

Red Hat Enterprise Linux 5 インストール ISO イメージを、以前に作成したターゲットディレクトリーにマウントします。root で以下のコマンドを実行します。

```
mount -o loop iso_name mount_dir
```

iso_name を ISO イメージへのパスに置き換え、mount_dir をターゲットディレクトリーへのパスに置き換えます。ブロックデバイスとしてファイルをマウントするには、-o loop オプションが必要です。

3.

マウントディレクトリーから、.discinfo ファイルの最初の行にある数値を確認します。

```
head -n1 mount_dir/.discinfo
```

このコマンドの出力は ISO イメージの ID 番号で、以下の手順を実行するのに知っておく必要があります。

4.

/etc/yum.repos.d/ ディレクトリーに新しいファイル（例： **new.repo** *）を作成し、以下の形式でコンテンツを追加します。正常に機能するために、このディレクトリーの設定ファイルの拡張子は* **.repo** *である必要があります。*

```
[repository]
mediaid=media_id
name=repository_name
baseurl=repository_url
gpgkey=gpg_key
enabled=1
gpgcheck=1
```

media_id を、mount_dir/.discinfo にある数値に置き換えます。repository_name の代わりにリポジトリー名を設定します。repository_url はマウントポイント内のリポジトリーディレクトリーへのパスに、gpg_key は GPG キーへのパスに置き換えます。

たとえば、Red Hat Enterprise Linux 5 Server ISO のリポジトリー設定は以下のようになります。

```
[rhel5-Server]
```

```
mediaid=1354216429.587870
name=RHEL5-Server
baseurl=file:///media/rhel5/Server
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=1
```

5. 前の手順で作成した /etc/yum.repos.d/new.repo を含むすべての yum リポジトリーを更新します。root で以下のコマンドを実行します。

```
yum update
```

これにより、システムはマウントされた ISO イメージで提供されたバージョンにアップグレードされます。

6. アップグレードに成功したら、root 権限で ISO イメージをアンマウントできます。

```
umount mount_dir
```

ここで、mount_dir はマウントディレクトリーへのパスです。また、最初の手順で作成されたマウントディレクトリーを削除することもできます。root で以下のコマンドを実行します。

```
rmdir mount_dir
```

7. 以前に作成された設定ファイルを別のインストールまたは更新に使用しない場合は、その設定ファイルを削除できます。root で以下のコマンドを実行します。

```
rm /etc/yum.repos.d/new.repo
```

例14.1 Red Hat Enterprise Linux 5.8 から 5.9 へのアップグレード

インターネット接続にアクセスせずにシステムをアップグレードする必要があるとします。これを行うには、新しいバージョンのシステムで ISO イメージを使用します（例： RHEL5.9-Server-20121129.0-x86_64-DVD1.iso ）。ターゲットディレクトリー /media/rhel5/ を複製している。root で ISO イメージがあるディレクトリーに移動し、以下のコマンドを入力します。

```
~]# mount -o loop RHEL5.9-Server-20121129.0-x86_64-DVD1.iso /media/rhel5/
```

マウントされたイメージの ID 番号を確認するには、以下のコマンドを実行します。

```
~]# head -n1 /media/rhel5/.discinfo
1354216429.587870
```

マウントポイントを yum リポジトリーとして設定するには、この番号が必要です。/etc/yum.repos.d/rhel5.repo ファイルを作成し、以下のテキストを挿入します。

```
[rhel5-Server]
mediaid=1354216429.587870
name=RHEL5-Server
baseurl=file:///media/rhel5/Server
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=1
```

yum リポジトリーを更新します。これにより、システムが RHEL5.9-Server-20121129.0-x86_64-DVD1.iso が提供するバージョンに効果的にアップグレードされます。root で以下のコマンドを実行します。

```
~]# yum update
```

システムが正常にアップグレードされたら、イメージをアンマウントし、ターゲットディレクトリーと設定ファイルを削除します。

```
~]# umount /media/rhel5/
```

```
~]# rmdir /media/rhel5/
```

```
~]# rm /etc/yum.repos.d/rhel5.repo
```

14.6. 便利な YUM 変数

以下は、yum コマンドと yum 設定ファイル（例： /etc/ yum.conf ファイルおよび .repo ファイル）の両方に使用できる変数の一覧です。

$releasever

これは、distroverpkg に記載されているように、パッケージのバージョンに置き換えられます。デフォルトは redhat-release パッケージのバージョンです。

$arch

これは、Python の os.uname （） に記載されているように、システムのアーキテクチャーに置き換えられます。

$basearch

これは、ベースアーキテクチャーに置き換えられます。たとえば、$arch=i686 の場合は、$basearch=i386 となります。

$YUM0-9

これは、同じ名前のシェル環境変数の値に置き換えられます。シェル環境変数が存在しない場合は、設定ファイルの変数は置き換えられません。

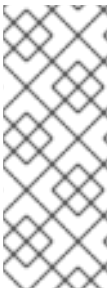
第15章 システムの登録およびサブスクリプション管理

効果的なアセット管理には、ソフトウェアインベントリーを処理するメカニズムが必要です (製品の種類と、ソフトウェアがインストールされているシステムの数の両方)。サブスクリプションサービスは、そのメカニズムを提供し、組織全体に対するサブスクリプションのグローバル割り当てと、1 つのシステムに割り当てられた特定のサブスクリプションの両方に対して透過性を提供します。

Red Hat Subscription Manager は yum と連携して、サブスクリプション管理でコンテンツ配信をユニットします。Subscription Manager は、サブスクリプションシステムの関連付けのみを処理します。yum またはその他のパッケージ管理ツールは、実際のコンテンツ配信を処理します。14章YUM (Yellowdog Updater Modified) では、yum の使用方法を説明します。

15.1. RED HAT SUBSCRIPTION MANAGER ツールの使用

登録とサブスクリプションはいずれも、Red Hat Subscription Manager と呼ばれる GUI および CLI ツールを使用してローカルシステムで管理されます。



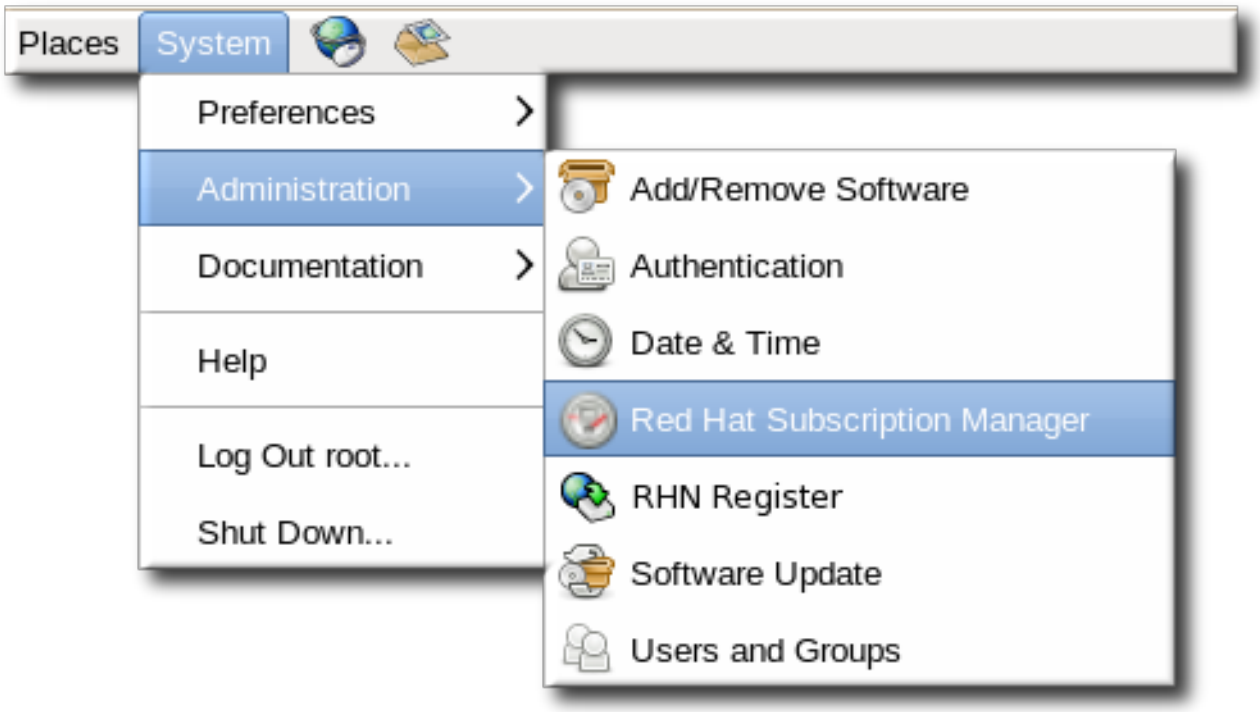
注記

Red Hat Subscription Manager ツールは、システムに変更を行うという性質上、常に root として実行されます。しかし、Red Hat Subscription Manager は、サブスクリプションサービスのユーザーアカウントとしてサブスクリプションサービスに接続します。

15.1.1. Red Hat Subscription Manager GUI の起動

Red Hat Subscription Manager は、トップ管理バーの System > Administration メニューに、管理ツールの 1 つとして表示されます。

図15.1 Red Hat Subscription Manager メニューオプション



または、1 つのコマンドを使用してコマンドラインから Red Hat Subscription Manager GUI を開くこともできます。

```
[root@server1 ~]# subscription-manager-gui
```

15.1.2. subscription-manager コマンドラインツールの実行

Red Hat Subscription Manager UI を介して実行できる操作はいずれも、subscription-manager ツールを実行しても実行できます。このツールの形式は以下のとおりです。

```
[root@server1 ~]# subscription-manager command [options]
```

各コマンドには、コマンドと共に使用される専用のオプション セットがあります。subscription-manager のヘルプおよび man ページに、より詳細な情報が記載されています。

表15.1 一般的な subscription-manager コマンド

コマンド	説明
register	サブスクリプションサービスに新しいシステムを登録するか、または識別します。

コマンド	説明
unregister	マシンの登録を解除します。これにより、そのサブスクリプションが削除され、サブスクリプションサービスからマシンが削除されます。
subscribe	特定のサブスクリプションをマシンにアタッチします。
redeem	ハードウェアおよび BIOS 情報に基づいて、マシンをベンダーから購入した事前に指定したサブスクリプションに自動的にアタッチします。
unsubscribe	特定のサブスクリプションまたはマシンからすべてのサブスクリプションを削除します。
list	マシンと互換性があるすべてのサブスクリプションを一覧表示します。マシンに実際にアタッチされているサブスクリプション、またはマシンで利用可能な未使用のサブスクリプションを一覧表示します。

15.2. システムの登録と登録解除

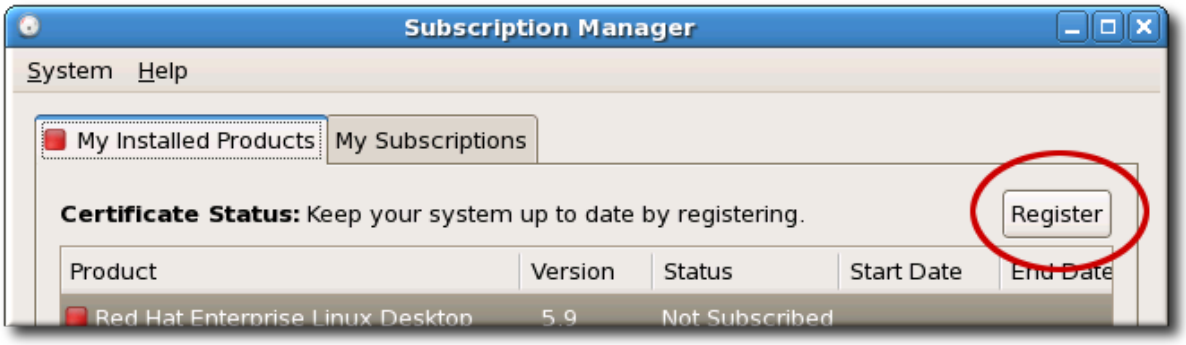
システムは、初回起動プロセス時またはキックスタートセットアップの一部として、サブスクリプションサービスに登録することができます (どちらも『Installation Guide』で説明されています)。システムを設定した後に登録することや、システムがそのサブスクリプションサービス内で管理されなくなった場合に、サブスクリプションサービスインベントリーから削除 (登録解除) することもできます。

15.2.1. GUI からの登録

1. Subscription Manager を起動します。以下に例を示します。

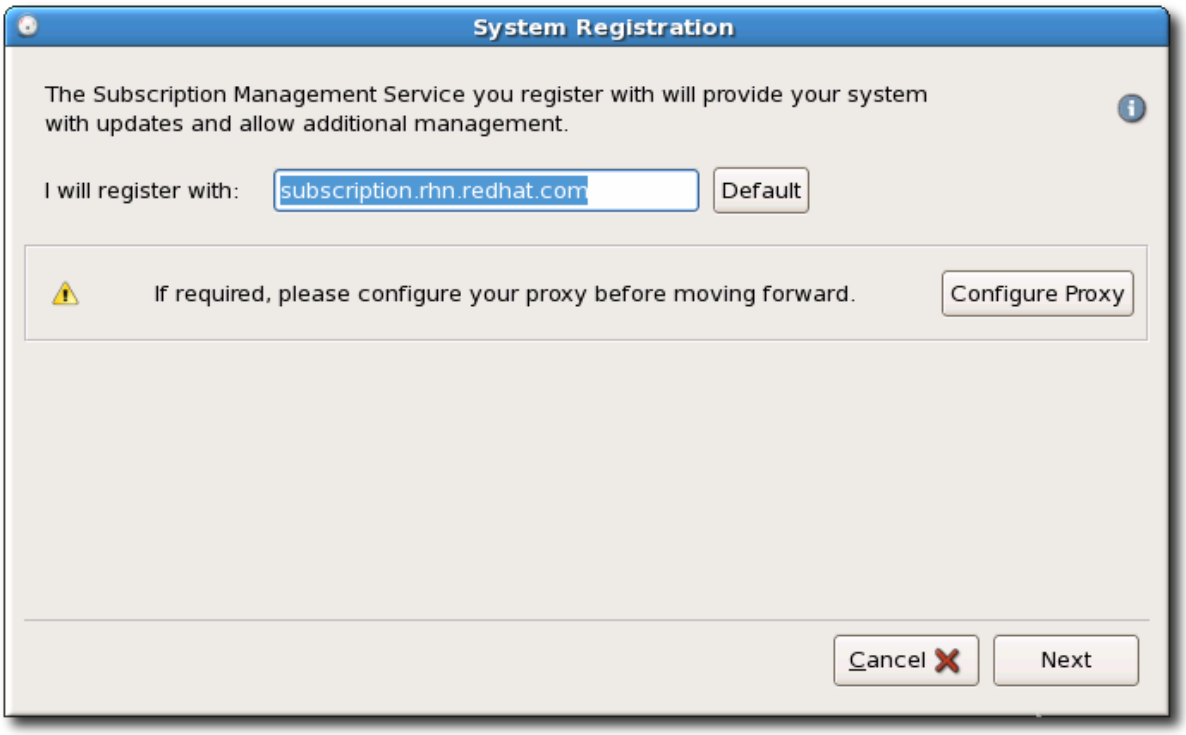
```
[root@server ~]# subscription-manager-gui
```

2. システムがまだ登録されていない場合は、My Installed Products タブの右上にあるウィンドウの上部に Register ボタンがあります。



3.

登録に使用するサブスクリプションサーバーを特定するには、サービスのホスト名を入力します。デフォルトのサービスは、カスタマーポータルの Subscription Management で、ホスト名が subscription.rhn.redhat.com になります。Subscription Asset Manager などの別のサブスクリプションサービスを使用するには、ローカルサーバーのホスト名を入力します。



証明書ベースのサブスクリプションを使用し、認識するサブスクリプションサービスがあり、システムは初回起動時にそれらのいずれかに登録できます。

- *カスタマーポータルの Subscription Management。Red Hat がホストしているサービスです (デフォルト)。*

- *Subscription Asset Manager。オンプレミスのサブスクリプションサーバーです。プロキシーとして動作し、コンテンツ配信をカスタマーポータルのサービスに送信します。*

- *CloudForms System Engine。オンプレミスのサービスです。サブスクリプションサービスとコンテンツ配信の両方を処理します。*

4. *ログインする サブスクリプションサービスの ユーザー認証情報を入力します。*

使用するユーザーの認証情報は、サブスクリプションサービスによって異なります。カスタマーポータルに登録する場合は、管理者または企業アカウントに Red Hat Network の認証情報を使用します。

ただし、Subscription Asset Manager または CloudForms System エンジンの場合、使用するユーザーアカウントはオンプレミスサービス内に作成され、おそらくカスタマーポータルのユーザーアカウントと同じではありません。

5. *必要に応じて、Manually assign subscriptions after registration チェックボックスを選択します。*

デフォルトでは、登録プロセスは自動的に最適なサブスクリプションをシステムに割り当

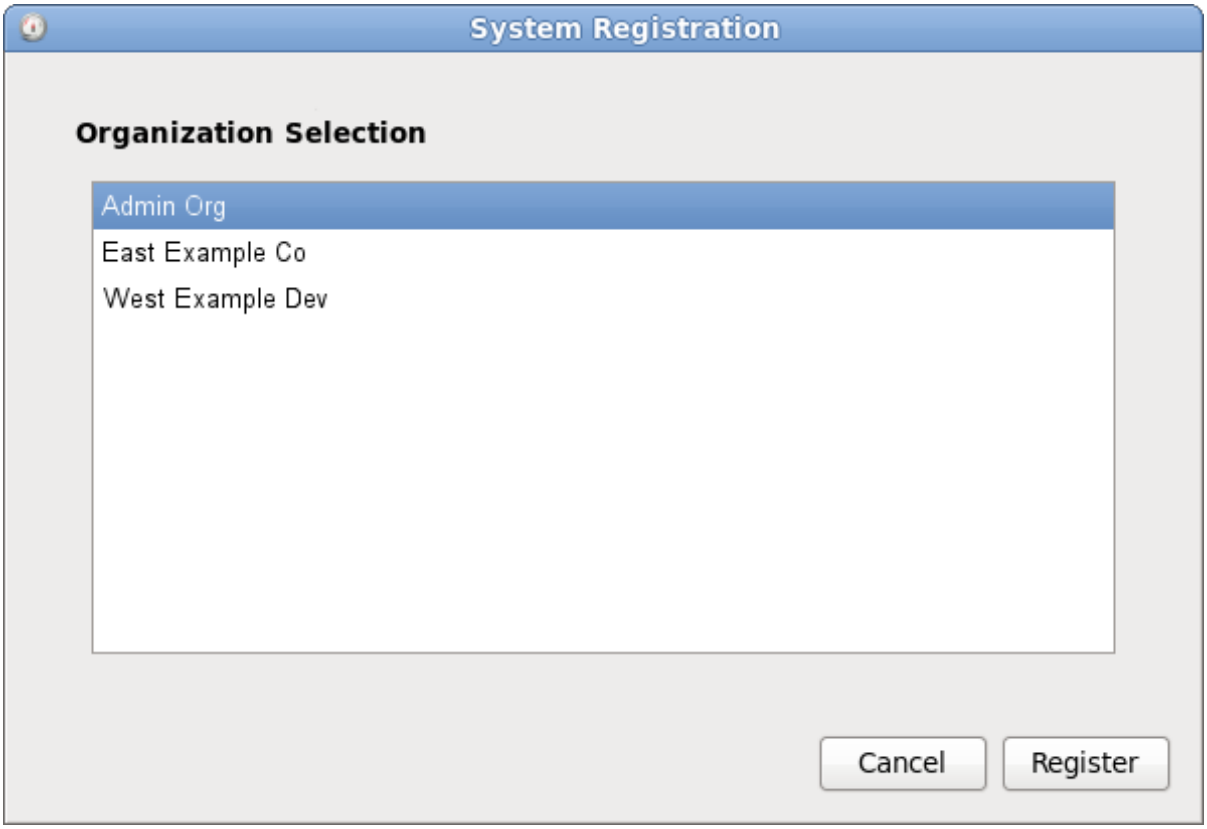
てます。「サブスクリプションのアタッチと削除」 にあるように、サブスクリプションを手動で選択できるようにオフにできます。

6.

登録が開始されると、Subscription Manager はシステムを登録する組織および環境 (組織内のサブドメイン) をスキャンします。

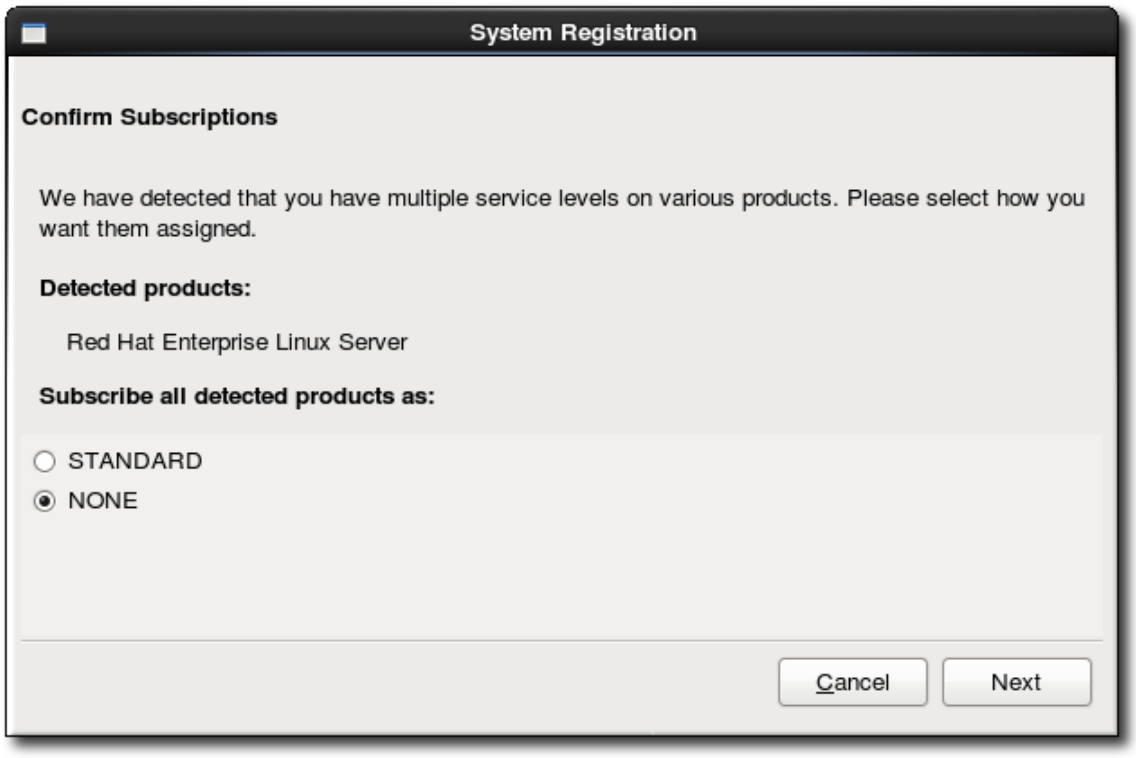
カスタマーポータルのサブスクリプション管理を使用する IT 環境には 1 つの組織しかないため、追加設定は必要ありません。Subscription Asset Manager などのローカルのサブスクリプションサービスを使用する IT インフラストラクチャーには複数の組織が設定されている場合があり、それらの組織内に複数の環境が設定されている場合があります。

複数の組織が検出されると、Subscription Manager は参加する組織を選択するよう要求します。



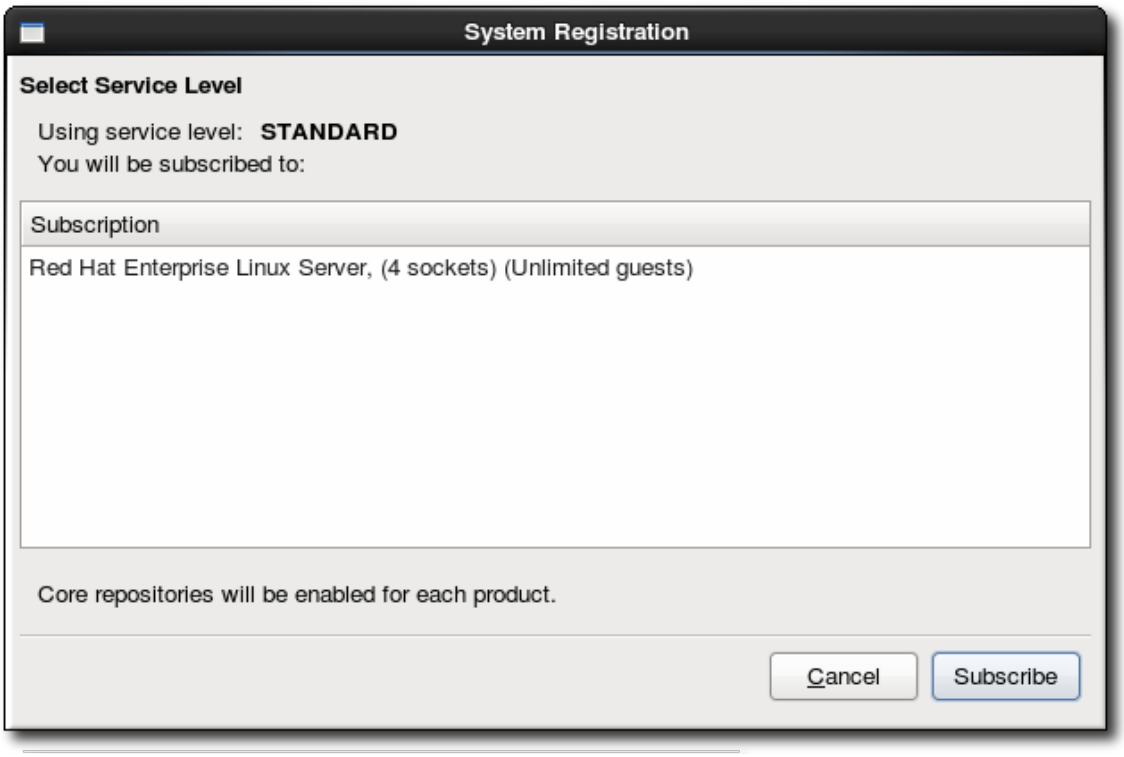
7. *デフォルト設定では、サブスクリプションが自動的に選択され、システムにアタッチされます。システムにアタッチするサブスクリプションを確認し、確定します。*

   a. *要求されたら、検出されたサブスクリプションに使用するサービスレベルを選択します。*



b.

Subscription Manager は、選択したサブスクリプションを表示します。ウィザードを完了するには、Subscribe ボタンをクリックして、このサブスクリプションの選択を確認する必要があります。



15.2.2. コマンドラインからの登録

マシンを登録する最も簡単な方法は、カスタマーポータルのサブスクリプション管理への認証に必要なユーザーアカウント情報と共にregister コマンドを渡すことです。システムが正常に認証されると、新しく割り当てられたシステムインベントリー ID と、登録したユーザーアカウント名をエコーバックします。

register コマンドのオプションは 表15.2「登録オプション」 に一覧表示されています。

例15.1 カスタマーポータルへのシステムの登録

```
[root@server1 ~]# subscription-manager register --username admin-example --password secret

The system has been registered with id: 7d133d55-876f-4f47-83eb-0ee931cb0a97
```

例15.2 登録時の自動サブスクライブ

register コマンドには --autosubscribe オプションがあります。これにより、システムをサブスクリプションサービスに登録し、1 回のステップでシステムのアーキテクチャーに最適なサブスク

リプションを即座にアタッチします。

```
[root@server1 ~]# subscription-manager register --username admin-example --password secret --autosubscribe
```

これは、Subscription Manager UI でデフォルト設定で登録する場合と同じ動作になります。

例15.3 Subscription Asset Manager へのシステムの登録

Subscription Asset Managr または CloudForms System Engine を使用すると、アカウントには organizationst と呼ばれる複数の独立したサブ区分を含めることができ、システムに参加する組織（通常は独立したグループまたはユニット）を指定する必要があります。これは、ユーザー名とパスワードに加えて --org オプションを使用して行います。指定のユーザーには、その組織にシステムを追加するためのアクセス権限も必要です。

カスタマーポータルサブスクリプション管理以外のサブスクリプションサービスに登録するには、システムが登録されている環境および組織の部門を特定するために、いくつかの追加オプションを使用する必要があります。

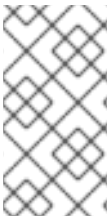
- サブスクリプションサービス自体を持つユーザーアカウントのユーザー名およびパスワード
- --serverURL - サブスクリプションサービスのホスト名を指定します。
- --baseurl - コンテンツ配信サービスのホスト名を指定します(CloudForms System Engine のみ)。
- --org - システムを登録する組織の名前を指定します。
- --environment - システムを追加する組織内の環境（グループ）の名前を指定します。デフォルトの環境がどの組織にも設定されているため、これはオプションです。

システムは、登録時にのみ環境に追加できます。

```
[root@server1 ~]# subscription-manager register --username=admin-example --password=secret --org="IT Department" --environment="dev" --serverurl=sam-
```

server.example.com

The system has been registered with id: 7d133d55-876f-4f47-83eb-0ee931cb0a97



注記

システムがマルチ組織環境にあり、組織が指定されていない場合、register コマンドは Remote Server エラーを返します。

表15.2 登録オプション

オプション	*説明*	*必須*
***--username*=name**	*コンテンツサーバーのユーザーアカウント名を指定します。*	*必須*
***--password*=password**	*ユーザーアカウントのパスワードを指定します。*	*必須*
***--serverurl*=hostname**	*使用するサブスクリプションサービスのホスト名を指定します。デフォルトはカスタマーポータルのサブスクライブ管理 subscription.rhn.redhat.com です。このオプションを使用しない場合、システムはカスタマーポータルのサブスクリプション管理に登録されます。*	*Subscription Asset Manager または CloudForms System Engine で必須*
***--baseurl*=URL**	*更新を受け取るのに使用するコンテンツ配信サーバーのホスト名を指定します。カスタマーポータルのサブスクリプション管理および Subscription Asset Manager はいずれも、URL https://cdn.redhat.com で Red Hat がホストするコンテンツ配信サービスを使用します。CloudForms System Engine は独自のコンテンツをホストするため、この URL は System Engine に登録されているシステムに使用する必要があります。*	*CloudForms System Engine で必須*

オプション	説明	必須
--org=name	システムに参加する組織を指定します。	必須 (ホスト型環境を除く)
--environment=name	システムを組織内の環境に登録します。	任意
--name=machine_name	登録するシステムの名前を設定します。デフォルトは **hostname** と同じです。	任意
--autosubscribe	最適な互換性のあるサブスクリプションを自動的に実施します。システムは 1 つのステップで設定できるため、自動セットアップ操作に適しています。	任意
--activationkey=key	登録プロセスの一環として既存のサブスクリプションをアタッチします。サブスクリプションは、ベンダーまたはシステム管理者によって、**Subscription Asset Manager** を使用して事前に割り当てられます。	任意
--servicelevel=None|Standard|Premium	そのマシンのサブスクリプションに使用するサービスレベルを設定します。これは、**--autosubscribe** オプションとのみ使用されます。	任意
--release=NUMBER	システムのサブスクリプションに使用するオペレーティングシステムのマイナーリリースを設定します。製品および更新は、その特定のマイナーリリースバージョンに限定されます。これは、**--autosubscribe** オプションとのみ使用されます。	任意
--force	すでに登録済みでもシステムを登録します。通常、マシンがすでに登録されている場合は、レジスター操作は失敗します。	任意

15.2.3. 登録解除

マシンの登録解除に必要なのは、unregister コマンドの実行のみです。これにより、サブスクリプションサービスからシステムのエントリーが削除され、サブスクリプションが削除され、ローカルでその ID 証明書とサブスクリプション証明書が削除されます。

コマンドラインでは、unregister コマンドのみが必要になります。

例15.4 システムの登録解除

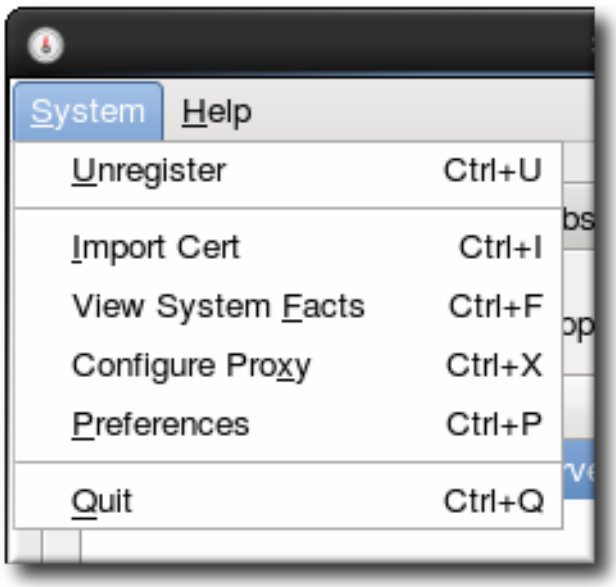
```
[root@server1 ~]# subscription-manager unregister
```

Subscription Manager GUI から登録解除するには、以下を実行します。

1. Subscription Manager UI を開きます。

```
[root@server ~]# subscription-manager-gui
```

2. System メニューを開き、Unregister 項目を選択します。



3. システムの登録解除を確認します。

15.3. サブスクリプションのアタッチと削除

システムにサブスクリプション を割り当てると、システムにそのサブスクリプション内の Red Hat 製品をインストールし、更新できます。サブスクリプションは、一度に購入されたすべての製品の全バリアントの一覧であり、製品とサブスクリプションが使用できる回数の両方を定義します。これらのライセンスのいずれかがシステムに関連付けられている場合に、そのサブスクリプションがシステムに アタッチされます。

15.3.1. GUI によるサブスクリプションのアタッチと削除

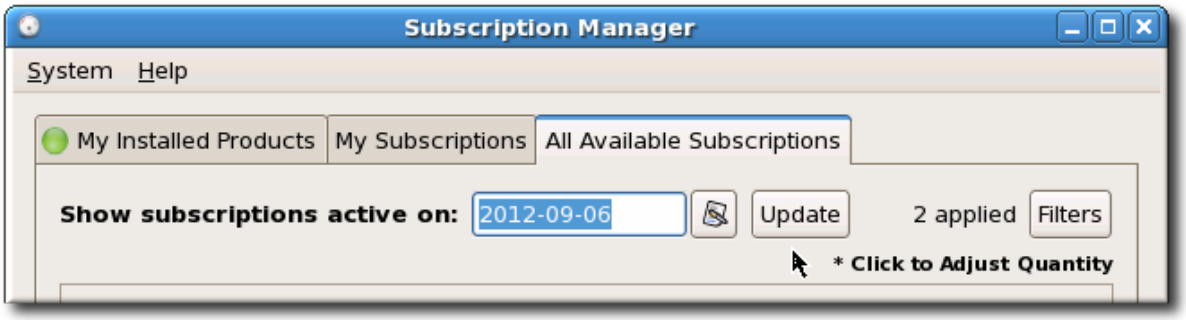
15.3.1.1. サブスクリプションのアタッチ

1. Subscription Manager を起動します。以下は例になります。

```
[root@server ~]# subscription-manager-gui
```

2. All Available Subscriptions タブを開きます。

3. 必要に応じて、日付の範囲を設定してFilters ボタンをクリックし、利用可能なサブスクリプションの検索に使用するフィルターを設定します。



サブスクリプションは、アクティブ化の日付と名前でフィルターリングできます。このチェックボックスで、より細かなフィルターリングが可能です。

- match my systemを選択すると、システムのアーキテクチャーに適合するサブスクリプションのみが表示されます。

- match my installed products を選択すると、システムに現在インストールされている製品で機能するサブスクリプションが表示されます。

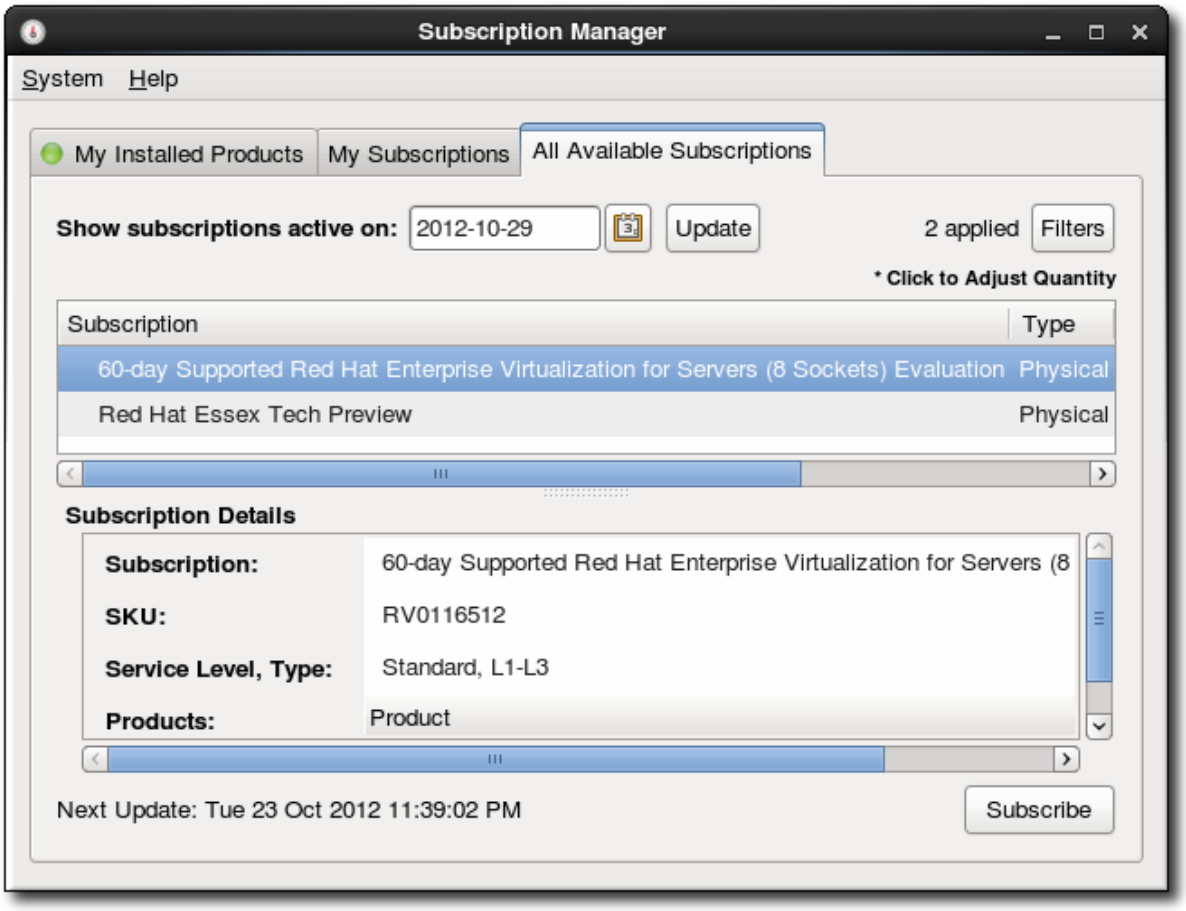
- have no overlap with existing subscriptionsを選択すると、重複した製品を持つサブスクリプションが除外されます。特定の製品について、サブスクリプションがすでにの

システムにアタッチされている場合や、複数のサブスクリプションが同じ製品を提供する場合は、サブスクリプションサービスがこれらのサブスクリプションをフィルターし、最適なもののみを表示します。

- **contain the text**は、*サブスクリプションまたはプール内の製品名などの文字列を検索します。*

*日付およびフィルターの設定後に、**Update** ボタンをクリックしてそれらを適用します。*

4. *利用可能なサブスクリプションの中から 1 つ選択します。*



5. *Subscribe ボタンをクリックします。*

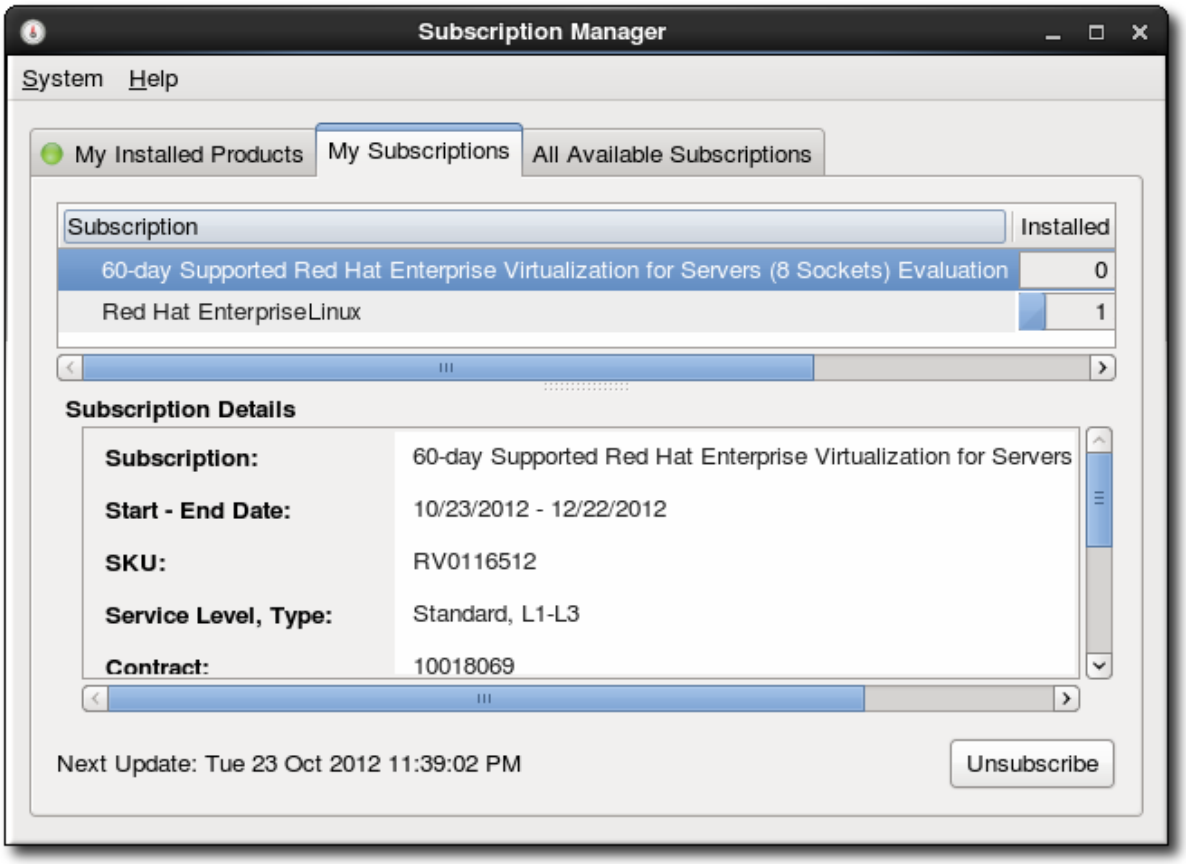
15.3.1.2. サブスクリプションの削除

1. *Subscription Manager を起動します。以下は例になります。*

```
[root@server ~]# subscription-manager-gui
```

2. *My Subscriptions タブを開きます。*

システムが現在アタッチされているアクティブなサブスクリプションがすべて表示されます。(サブスクリプションで利用可能な製品は、インストールされている場合とされていない場合があります。)



3. *削除するサブスクリプションを選択します。*

4. *ウィンドウ右下にある 登録解除 ボタンをクリックします。*

15.3.2. コマンドラインでのサブスクリプションのアタッチと削除

15.3.2.1. サブスクリプションのアタッチ

システムにサブスクリプションをアタッチする場合は、--pool オプションを使用して、個別の製品またはアタッチするサブスクリプションを指定する必要があります。

```
[root@server1 ~]# subscription-manager subscribe --pool=XYZ01234567
```

subscribe コマンドのオプションは、表15.3「サブスクライブオプション」 に記載されています。

購入した製品のサブスクリプションプールの ID を指定する必要があります。プール ID は、list コマンドを実行すると製品のサブスクリプション情報と共に一覧表示されます。

```
[root@server1 ~]# subscription-manager list --available
```

```
+-------------------------------------------+
   Available Subscriptions
+-------------------------------------------+
ProductName:          RHEL for Physical Servers
ProductId:            MKT-rhel-server
PoolId:               ff8080812bc382e3012bc3845ca000cb
Quantity:             10
Expires:              2011-09-20
```

または、サブスクリプションサービスで識別される最適なサブスクリプションは、--auto オプションを使用してシステムにアタッチすることができます(register コマンドの --autosubscribe オプションと類似しています)。

```
[root@server1 ~]# subscription-manager subscribe --auto
```

表15.3 サブスクライブオプション

オプション	説明	必須
--pool=pool-id	システムにアタッチするサブスクリプションの ID を指定します。	--auto が使用されていない限り必須
--auto	システムを最も適するサブスクリプションに自動的にアタッチします。	任意
--quantity=number	複数のサブスクリプションをシステムにアタッチします。これは、数の上限が定義されるサブスクリプションに対応するのに使用されます (例 : 2 ソケットサーバー用のサブスクリプションを 2 つ使用して 4 ソケットマシンに対応する)。	任意
--servicelevel=None|Standard|Premium	そのマシンのサブスクリプションに使用するサービスレベルを設定します。これは、--auto オプションとだけ併用されます。	任意

15.3.2.2. コマンドラインからのサブスクリプションの削除

システムは、複数のサブスクリプションおよび製品にアタッチすることができます。同様に、1 つのサブスクリプションまたはすべてのサブスクリプションをシステムから削除することができます。

--all オプションを指定して unsubscribe コマンドを実行すると、システムに現在アタッチされているすべての製品サブスクリプションとサブスクリプションプールが削除されます。

```
[root@server1 ~]# subscription-manager unsubscribe --all
```

1 つの製品サブスクリプションを削除することもできます。各製品には、識別用の X.509 証明書がインストールされています。削除する製品サブスクリプションは、サブスクライブ解除 されたコマンドで、その X.509 証明書の ID 番号を参照して識別されます。

1. 1 つの製品サブスクリプションを削除する場合は、製品証明書のシリアル番号を取得します。シリアル番号は、subscription#.pem ファイル (例 :3927295555856979O7.pem) から、または list コマンドを使用して取得できます。以下は例になります。

```
[root@server1 ~]# subscription-manager list --consumed

+-------------------------------------------+
   Consumed Product Subscriptions
+-------------------------------------------+


ProductName:        High availability (cluster suite)
ContractNumber:     0
SerialNumber:       11287514358600162
Active:             True
Begins:             2010-09-18
Expires:            2011-11-18
```

2. --serial オプションを指定して subscription-manager ツールを実行し、証明書を指定します。

```
[root@server1 ~]# subscription-manager unsubscribe --serial=11287514358600162
```

15.4. ベンダーサブスクリプションの引き換え

システムは、そのシステムですでに利用可能な既存のサブスクリプションで設定できます。サードパーティーベンダーから購入した一部のシステムでは、Red Hat 製品へのサブスクリプションがマシンの購入に含まれている場合があります。

Red Hat Subscription Manager は、システムハードウェアと BIOS に関する情報をシステムファクトにプルし、ハードウェアベンダーを認識します。ベンダーと BIOS 情報が特定の設定と一致する場合は、サブスクリプションを 引き換えることができ、サブスクリプションを自動的にシステムにアタッチすることができます。

15.4.1. GUI によるサブスクリプションの引き換え

注記

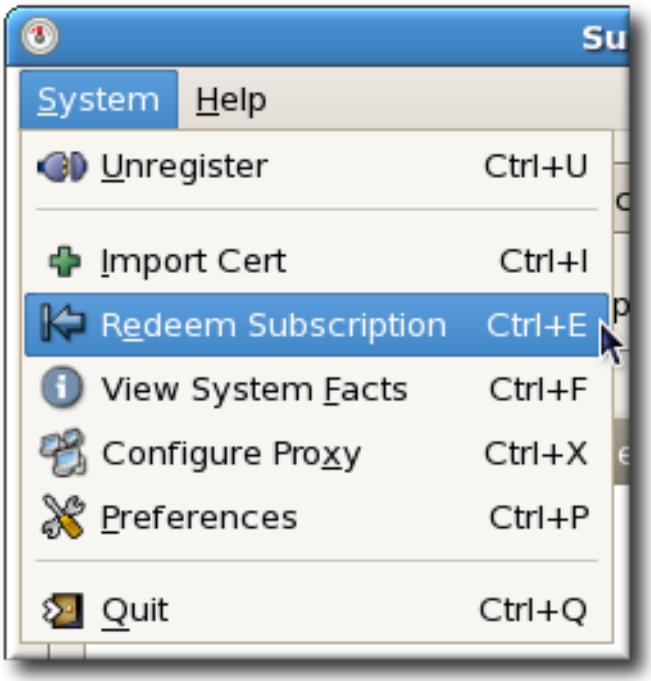
マシンに引き換えるサブスクリプションがない場合、Redeem メニュー項目は表示されません。

1. Subscription Manager を起動します。以下は例になります。

```
[root@server ~]# subscription-manager-gui
```

2. 必要に応じて、「GUI からの登録」の説明に従ってシステムを登録します。

3. ウィンドウ左上にある System メニューを開き、Redeem 項目をクリックします。



4. ダイアログウィンドウで、引き換えが完了した時に通知を送信するメールアドレスを入力します。引き換えプロセスでは、ベンダーに連絡して事前設定されたサブスクリプションに関する情報を受信するのに数分かかる可能性があるため、通知メッセージは Subscription Manager のダイアログウィンドウではなくメールで送信されます。

5. *Redeem ボタンをクリックします。*

確認メールが到着するまでに最長 10 分かかる場合があります。

15.4.2. コマンドラインを使用したサブスクリプションの引き換え

注記

サブスクリプションサービスがシステムとそのサブスクリプションを適切に特定できるように、先に マシンを登録する必要があります。

マシンのサブスクリプションは、プロセスの完了時に引き換えメールを送信するメールアドレスと共に、redeem コマンドを実行して引き換えられます。

```
# subscription-manager redeem --email=jsmith@example.com
```

15.5. SUBSCRIPTION ASSET MANAGER のアクティベーションキーからのサブスクリプションのアタッチ

ローカルの Subscription Asset Manager は、システムに使用するサブスクリプションを事前に設定することができます。その事前に設定されたサブスクリプションのセットは、アクティベーションキーで特定されます。そのキーを使用して、ローカルシステムにこれらのサブスクリプションをアタッチすることができます。

Subscription Asset Manager のアクティベーションキーは、新規システムの登録プロセスの一部として使用できます。

-

```
# subscription-manager register --username=jsmith --password=secret --org="IT Dept" --
activationkey=abcd1234
```

複数の組織がある場合は、システムの組織を指定する必要があります。この情報はアクティベーションキーでは定義されません。

15.6. システムの優先条件の設定

サブスクリプションの自動割り当てと修復（更新）は、現在インストールされている製品、ハードウェア、アーキテクチャーなど、さまざまな基準に基づいてシステムにアタッチするサブスクリプションを選択します。Subscription Manager が使用する優先条件を、さらに 2 つ設定できます。

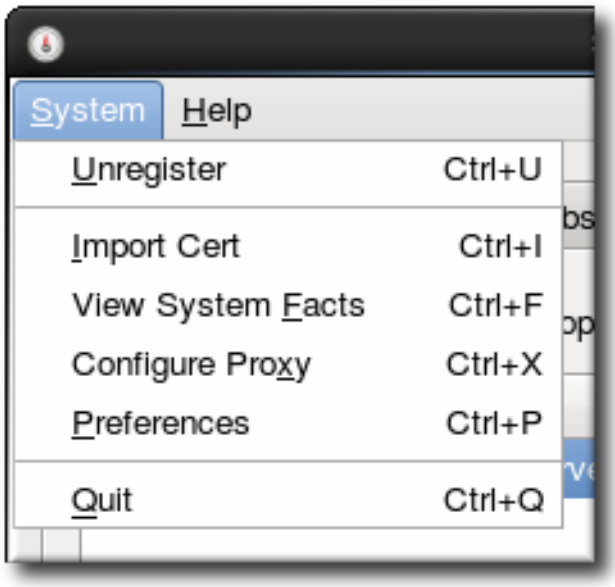
- サブスクリプションのサービスレベル
- 使用するオペレーティングシステムのマイナーバージョン (X.Y)

これは、インストールされているすべての製品と現在のサブスクリプションがアクティブな状態を維持するために毎日実行される修復に特に便利です。

15.6.1. UI での優先条件の設定

サービスレベルの優先条件とオペレーティングシステムのリリースバージョンの優先条件はいずれも、Subscription Manager の System Preferences ダイアログボックスで設定されます。

1. Subscription Manager を開きます。
2. System メニューを開きます。
3. System Preferences メニュー項目を選択します。

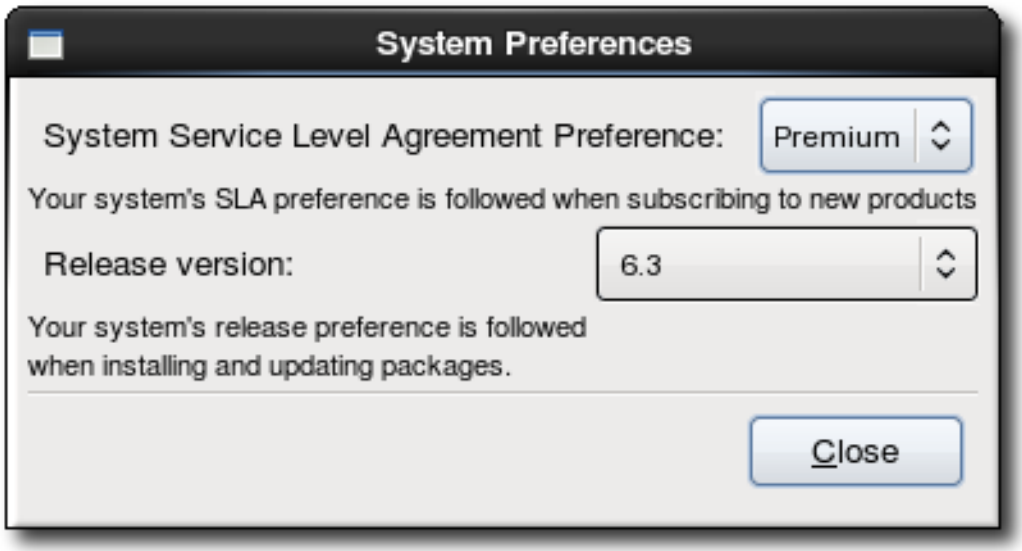


4.

ドロップダウンメニューから、希望のサービスレベルアグリーメントの優先条件を選択します。すべてのアクティブなサブスクリプション基づいて、Red Hat アカウントで利用可能なサービスレベルのみが一覧表示されます。

5.

Release version ドロップダウンメニューで、オペレーティングシステムのリリースの優先条件を選択します。一覧表示されているバージョンだけが、アカウントにアクティブなサブスクリプションがある Red Hat Enterprise Linux バージョンです。



6.

優先条件は保存され、今後のサブスクリプション操作に適用されます。ダイアログを閉じるには、Close をクリックします。

15.6.2. コマンドラインでのサービスレベルの設定

一般的なサービスレベルの優先条件は、service-level --set コマンドを使用して設定できます。

例15.5 サービスレベルの優先条件の設定

まず、service-level コマンドで --list オプションを使用して、システムで利用可能なサービスレベルを一覧表示します。

```
[root@server ~]# subscription-manager service-level --list
+-------------------------------------------+
        Available Service Levels
+-------------------------------------------+
Standard
None
Premium
Self-Support
```

次に、システムで必要なレベルを設定します。

```
[root@server ~]# subscription-manager service-level --set=self-support
Service level set to: self-support
```

ローカルシステムの現在の設定は、--show オプションで表示されます。

```
[root#server ~]# subscription-manager service-level --show
Current service level: self-support
```

サービスレベルの優先条件は、サブスクリプション操作の実行時 (システムの登録、登録後のサブスクリプションのアタッチなど) に定義できます。これは、システム設定を上書きするために使用できます。register コマンドおよび subscribe コマンドの両方には、--servicelevel オプションがあり、そのアクションの優先度を設定します。

例15.6 プレミアムサービスレベルのサブスクリプションの自動アタッチ

```
[root#server ~]# subscription-manager subscribe --auto --servicelevel Premium
Service level set to: Premium
Installed Product Current Status:
ProductName:          Red Hat Enterprise Linux 5 Server
Status:               Subscribed
```

注記

--servicelevel オプションには、--autosubscribe オプション(register 用)または --auto オプション（サブスクライブ用）が必要です。指定したプールをアタッチする場合や、サブスクリプションをインポートする場合には使用できません。

15.6.3. コマンドラインでの優先オペレーティングシステムリリースバージョンの設定

多くの IT 環境は、特定レベルのセキュリティーまたはその他の基準を満たすために認定が必要です。この場合、メジャーアップグレードは注意して計画し、制御する必要があります。そのため、管理者は単に yum update を実行してあるバージョンから別のバージョンに移行することができません。

リリースバージョンの優先条件を設定すると、最新バージョンのリポジトリーを自動的に使用する代わりに、システムのアクセスがそのオペレーティングシステムのバージョンに関連付けられたコンテンツリポジトリーに制限されます。

たとえば、優先するオペレーティングシステムのバージョンが 5.9 の場合、他のリポジトリーが利用できる場合でも、インストールされているすべての製品とシステムに割り当てられたサブスクリプションに対して 5.9 コンテンツリポジトリーが優先されます。

例15.7 登録時のオペレーティングシステムリリースの設定

リリースバージョンの優先条件は、register に --release オプションを使用して、システムの登録時に設定できます。これにより、リリースの優先条件が、登録時にシステムに選択されたサブスクリプションおよび自動アタッチされたサブスクリプションに適用されます。

設定を設定するには、--autosubscribe オプションが必要です。これは、自動アタッチするサブスクリプションを選択するために使用される基準の 1 つであるためです。

```
[root#server ~]# subscription-manager register --autosubscribe --release=5.9 --
username=admin@example.com...
```

注記

サービスレベルの優先条件を設定するのとは異なり、リリースの優先条件は登録時にしか使用できません。あるいは、優先条件として設定します。subscribe コマンドで指定することはできません。

例15.8 オペレーティングシステムリリースの優先条件の設定

release コマンドは、組織で利用可能な購入済みの (アタッチされているとは限らない) サブスクリプションに基づいて、利用可能なオペレーティングシステムのリリースを表示できます。

```
[root#server ~]# subscription-manager release --list
+-------------------------------------------+
        Available Releases
+-------------------------------------------+
5.8
5.9
5Server
```

次に、--set は、利用可能なリリースバージョンの 1 つに優先条件を設定します。

```
[root#server ~]# subscription-manager release --set=5.9
Release version set to: 5.9
```

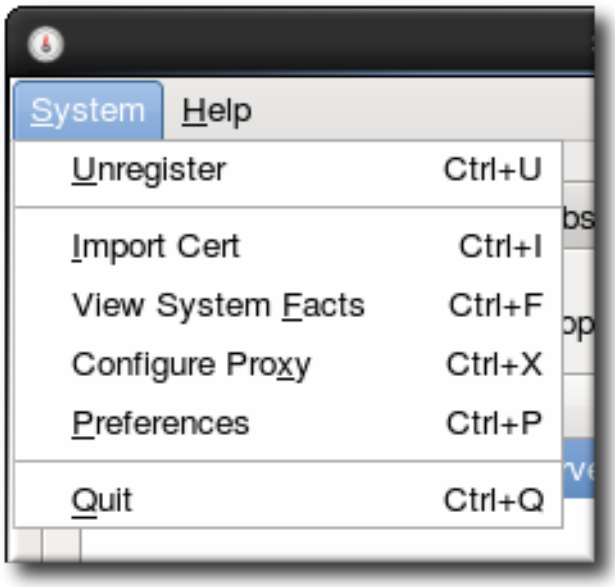
15.6.4. Preference の削除

コマンドラインで設定を削除するには、適切なコマンドで --unset を使用します。たとえば、リリースバージョンの設定を解除するには、以下を実行します。

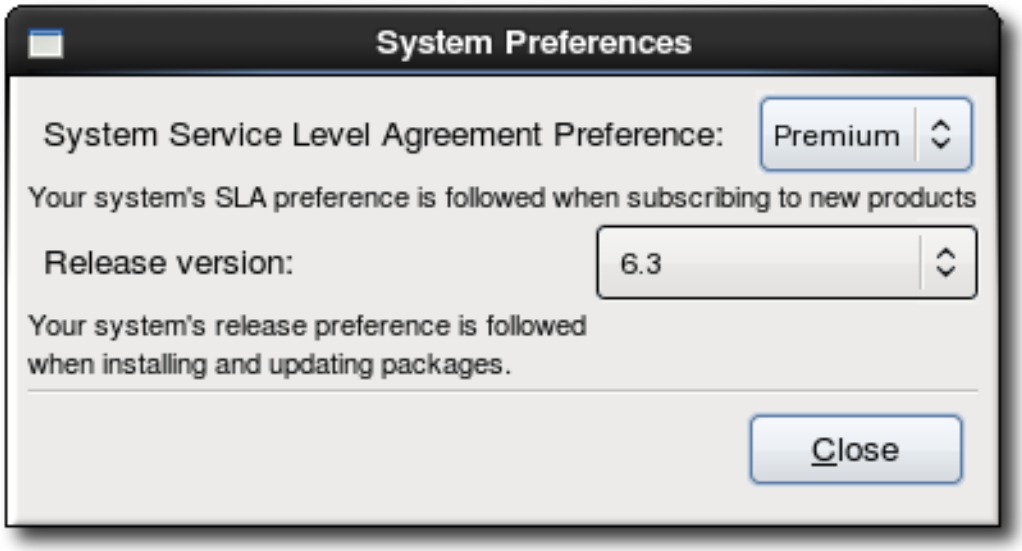
```
[root#server ~]# subscription-manager release --unset
Release version set to:
```

UI で設定を削除または設定解除するには、以下を実行します。

1. Subscription Manager を開きます。
2. System メニューを開きます。
3. System Preferences メニュー項目を選択します。



4. *対応するドロップダウンメニューで、サービスレベルまたはリリースバージョンの値を空白行に設定します。*



5. ***Close** をクリックします。*

15.7. サブスクリプションの有効期限および通知の管理

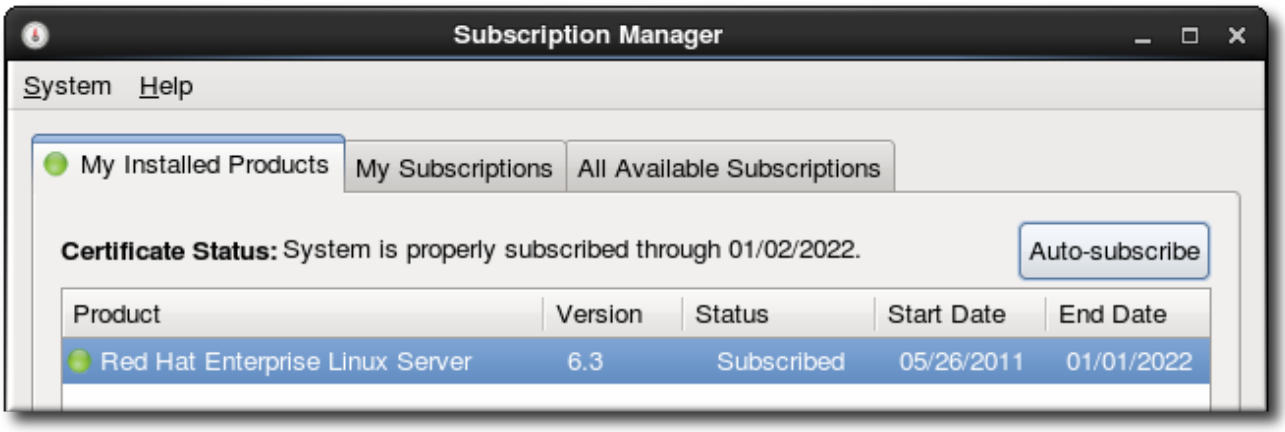
サブスクリプションは、有効期間 と呼ばれる一定期間アクティブになります。サブスクリプションの購入時に、契約の開始日と終了日が設定されます。

システムには、複数のサブスクリプションをアタッチすることができます。各製品にはそれぞれのサブスクリプションが必要です。さらに、一部の製品では、完全にサブスクライブするために、複数のサブスクリプションが必要になる場合があります。たとえば、16 ソケットマシンでは、ソケット数を

カバーするために、4 ソケットオペレーティングシステムサブスクリプションが 4 つ必要になる場合があります。

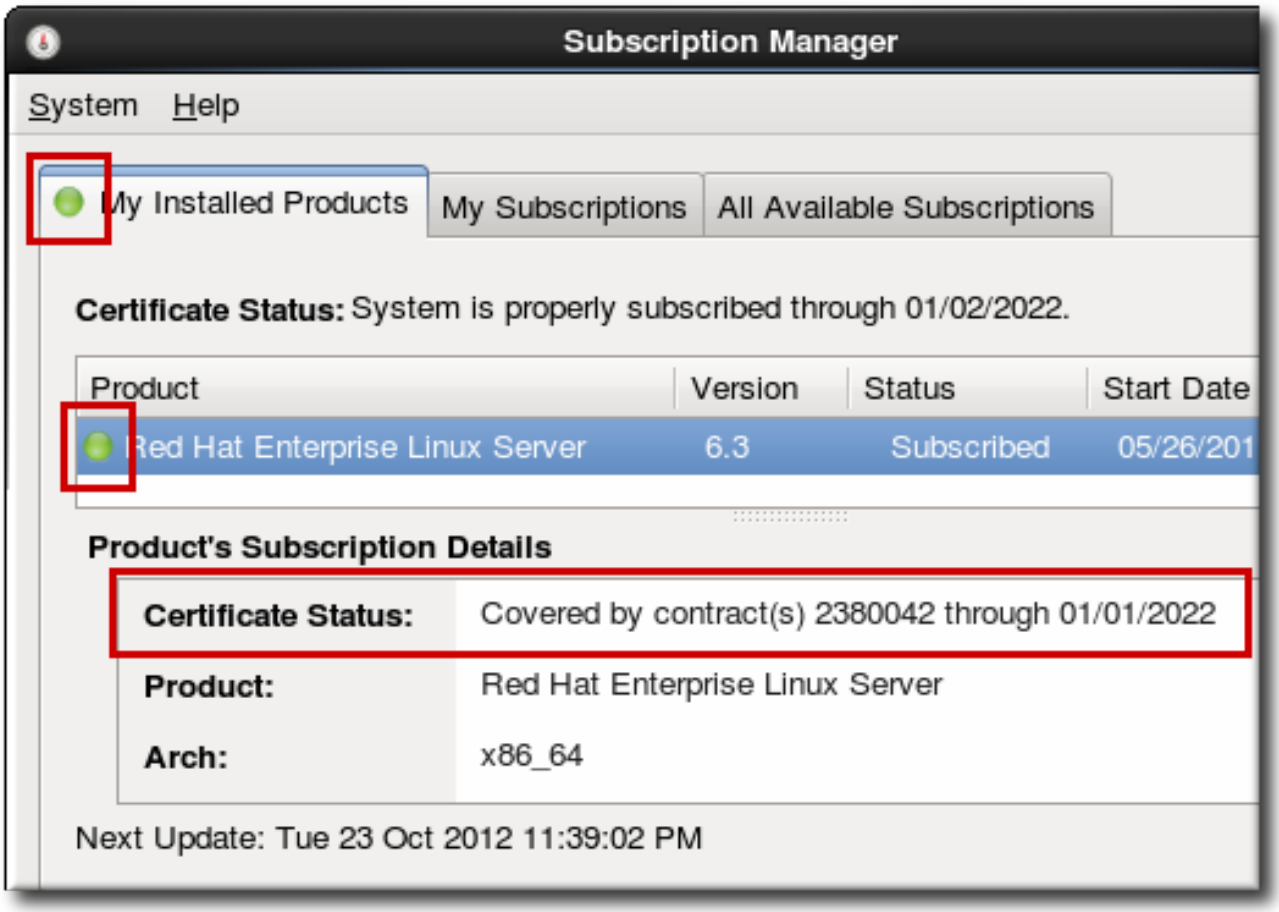
My Installed Software タブには、システム全体のサブスクリプションステータスが表示されます。また、製品サブスクリプションが有効から無効になる (つまり、期限切れになる) 最初の日付も表示されます。

図15.2 有効期限



Red Hat Subscription Manager は、システムにインストール済みの製品の有効な証明書への変更を示す一連のログおよび UI メッセージを提供します。Subscription Manager GUI では、システムサブスクリプションのステータスは色分けされます。green はすべての製品が完全にサブスクライブされていることを意味し、黄 色は一部の製品はサブスクライブされず、更新がまだ有効であることを意味します。赤 は更新が無効になっていることを意味します。

図15.3 色分けされステータスビュー

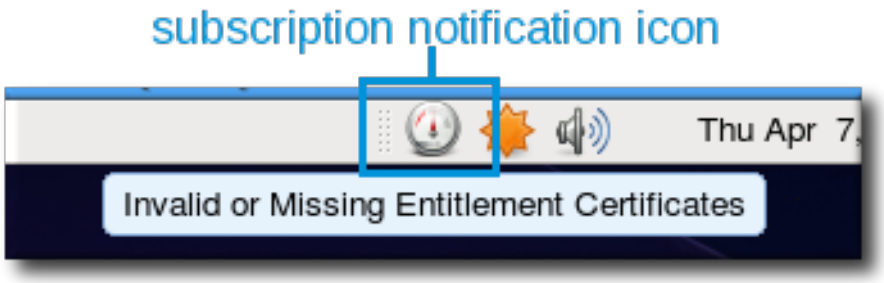


*コマンドラインツールも、マシンのそのステータスを表示します。緑、黄色、および赤のコードは、それぞれ***subscribed**、**partially subscribed**、*および* **expired/not subscribed** *というテキストのステータスメッセージに変換されます。*

```
[root@server ~]# subscription-manager list
+-------------------------------------------+
   Installed Product Status
+-------------------------------------------+

ProductName:          Red Hat Enterprise Linux Server
Status: Not Subscribed
Expires:
SerialNumber:
ContractNumber:
AccountNumber:
```

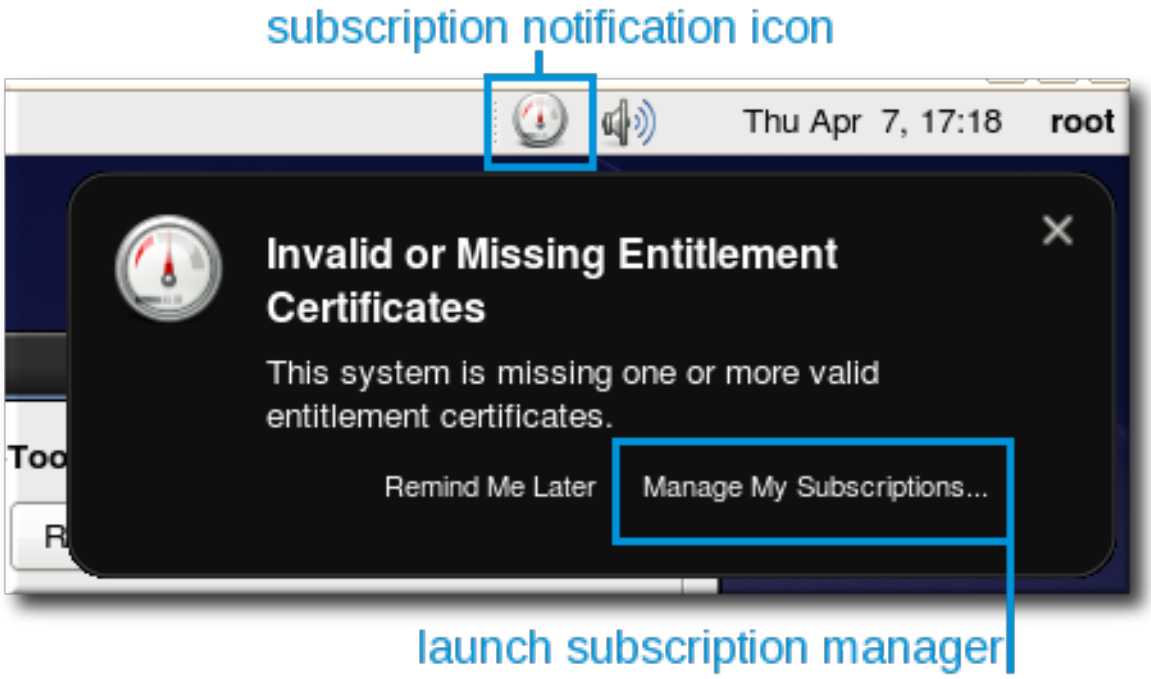
サブスクリプションの変更に関する警告がある場合は、必ずトップメニューバーに燃料計に似た小さなアイコンが表示されます。

図15.4 サブスクリプション通知アイコン



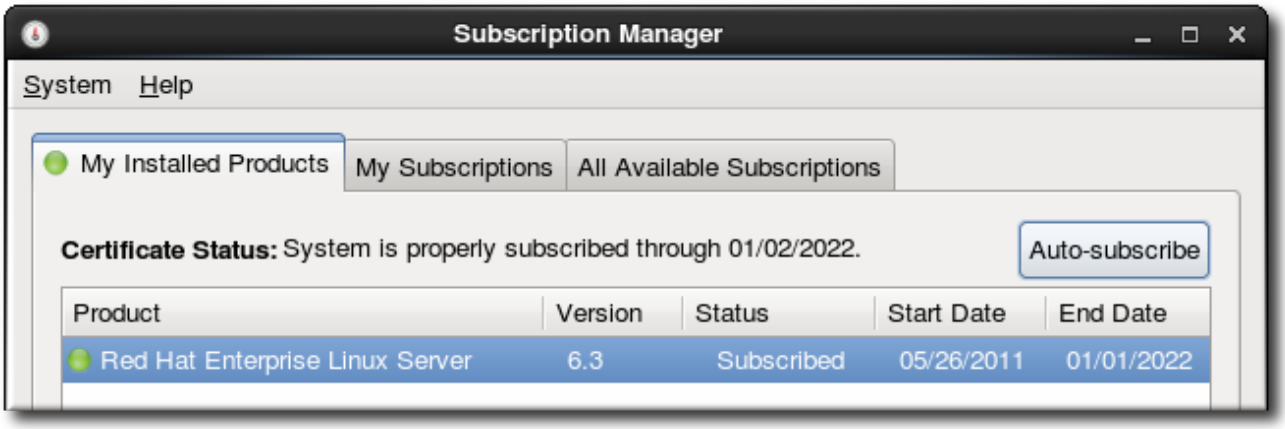
*インストールされているいずれかの製品がサブスクリプションの有効期限に近づくと、**SubscriptionManager** デーモンが警告を発行します。システムに有効な証明書がない製品がある場合に同様のメッセージが表示されます。つまり、その製品をカバーするサブスクリプションが実施されていないか、サブスクリプションの有効期限を過ぎて製品がインストールされます。サブスクリプション通知ウィンドウで **Manage My Subscriptions...** ボタンをクリックすると、**Red Hat Subscription Manager GUI** が開き、サブスクリプションを表示および更新します。*

図15.5 サブスクリプション警告メッセージ



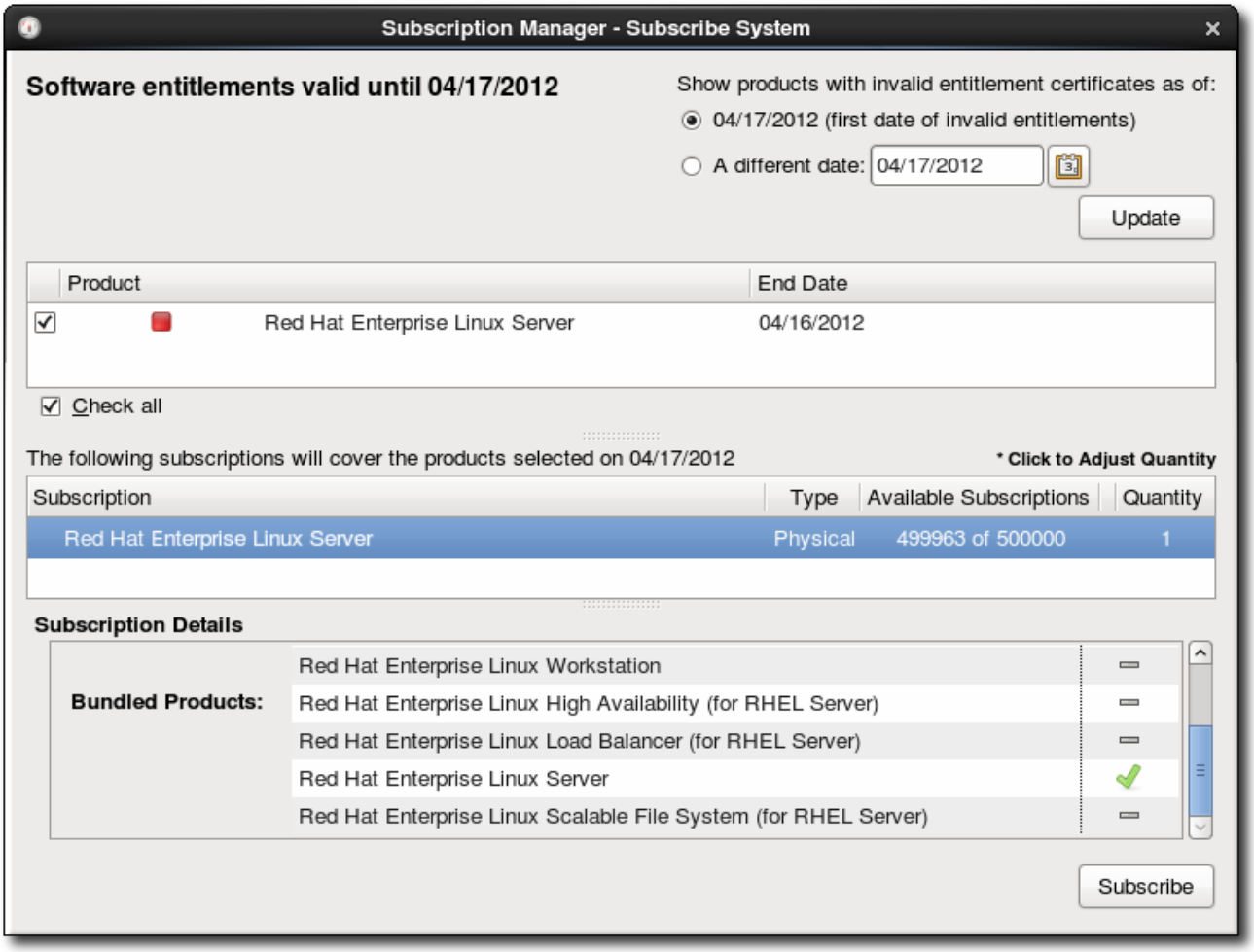
***Subscription Manager UI** を開くと、通知によって開かれたか、通常どおり開かれたかに関係なく、製品に有効な証明書が不足しているかどうかを示すアイコンが左上隅に表示されます。無効な製品に一致するサブスクリプションをアタッチする最も簡単な方法は、**Autosubscribe** ボタンをクリックすることです。*

図15.6 自動サブスクライブボタン



***Subscribe System** ダイアログには、有効な証明書を持たない特定の製品に適用される利用可能なサブスクリプションのターゲット一覧が表示されます (サブスクリプションが利用可能であることを前提とします)。*

図15.7 システムのサブスクライブ



パート III. ネットワーク関連の設定

ネットワークの設定方法を説明した後、このパートではリモートログインの許可、ネットワーク上でのファイルおよびディレクトリーの共有方法、Web サーバーのセットアップ方法など、ネットワークに関連するトピックについて説明します。

第16章 NETWORK INTERFACES

Red Hat Enterprise Linux では、設定済みのソフトウェア インターフェイス と、システムに接続されている 物理ネットワークデバイスとの間で、すべてのネットワーク通信が発生します。

ネットワークインターフェイスの設定ファイルは /etc/sysconfig/network-scripts/ ディレクトリーにあります。これらのネットワークインターフェイスをアクティブまたは非アクティブにするために使用されるスクリプトもここにあります。インターフェイスファイルの数とタイプはシステムによって異なる場合がありますが、このディレクトリーには 3 つのカテゴリーがあります。

1. インターフェイス設定ファイル
2. インターフェイス制御スクリプト
3. ネットワーク機能ファイル

各カテゴリーのファイルは連携して、さまざまなネットワークデバイスを有効にします。

本章では、これらのファイル間の関係とそれらがどのように使用されるかを説明します。

16.1. ネットワーク設定ファイル

インターフェイス設定ファイルを解決する前に、まずネットワーク設定で使用される主要な設定ファイルを項目化できるようにします。ネットワークスタックを設定する際にこれらのファイルが果たすロールを理解すると、Red Hat Enterprise Linux システムをカスタマイズする際に役立ちます。

プライマリーネットワーク設定ファイルは以下のとおりです。

/etc/hosts

このファイルの主な目的は、他の方法で解決できないホスト名を解決することです。また、これを使用して、DNS サーバーのない小規模なネットワークでホスト名を解決することもできます。コンピューターが存在するネットワークのタイプに関係なく、このファイルには、ループバックデバイス(127.0.0.1)の IP アドレスを localhost.localdomain として指定する行が含まれている必要があります。詳細は、man ページの hosts (5) を参照してください。

/etc/resolv.conf

このファイルは、DNS サーバーおよび検索ドメインの IP アドレスを指定します。特に設定されていない限り、ネットワーク初期化スクリプトにこのファイルが入力されます。このファイルの詳細は、resolv.conf (5) man ページを参照してください。

/etc/sysconfig/network

このファイルは、すべてのネットワークインターフェイスのルーティングおよびホスト情報を指定します。これは、インターフェイス固有のものではなく、グローバル効果を持ち、ディレクティブを含めるために使用されます。このファイルと、ファイルで使用できるディレクティブの詳細は、「/etc/sysconfig/network」を参照してください。

/etc/sysconfig/network-scripts/ifcfg-<interface-name>

ネットワークインターフェイスごとに、対応するインターフェイス設定スクリプトがあります。これらの各ファイルは、特定のネットワークインターフェイスに固有の情報を提供します。このタイプのファイルと、許可するディレクティブの詳細は、「インターフェイス設定ファイル」を参照してください。

警告

/etc/sysconfig/networking/ ディレクトリーは、ネットワーク 管理ツール (system-config-network)によって使用され、その内容を手動で編集し ない でください。設定を削除するリスクがあるため、ネットワーク設定には 1 つの方法のみを使用することが強く推奨されます。

Network Administration Tool を使用したネットワークインターフェイスの設定に関する詳細は、を参照してください。17章Network Configuration

16.2. インターフェイス設定ファイル

インターフェイス設定ファイルは、個々のネットワークデバイスのソフトウェアインターフェイスを制御します。これは、システムの起動時に、このファイルを使用して、どのインターフェイスを起動するかと、どのように設定するかを決定します。通常、これらのファイルの名前は ifcfg- <name > です。ここで、<name> は設定ファイルが制御するデバイスの名前を指します。

16.2.1. イーサネットインターフェイス

最も一般的なインターフェイスファイルの 1 つは、システム内の最初のイーサネットネットワークインターフェイスカードまたは NIC を制御する /etc/sysconfig/network -scripts/ifcfg-eth0 です。複数の NIC を持つシステムでは、複数の ifcfg-eth <X > ファイルがあります。< X > は、特定のインターフェイスに対応する一意の番号になります。各デバイスには独自の設定ファイルがあるため、管理者は各インターフェイスの機能を個別に制御できます。

以下は、固定 IP アドレスを使用するシステムの ifcfg-eth0 ファイルの例です。

```
DEVICE=eth0
BOOTPROTO=none
ONBOOT=yes
NETMASK=255.255.255.0
IPADDR=10.0.1.27
USERCTL=no
```

インターフェイス設定ファイルに必要な値は、他の値に基づいて変更される可能性があります。たとえば、DHCP を使用するインターフェイスの ifcfg-eth0 ファイルは、 DHCP サーバーにより IP 情報が提供されるため、異なるように見えます。

```
DEVICE=eth0
BOOTPROTO=dhcp
ONBOOT=yes
```

ネットワーク 管理ツール(system-config-network)は、さまざまなネットワークインターフェイス設定ファイルに変更を加える簡単な方法です（このツールの使用に関する詳細な説明は、17章Network Configuration を参照してください）。

ただし、特定のネットワークインターフェイスの設定ファイルを手動で編集することもできます。

イーサネットインターフェイスの設定ファイルにある設定可能なパラメーターの一覧を以下に示します。

BONDING_OPTS=<parameters>

ボンディングデバイスの設定パラメーターを設定し、/etc/sysconfig/network-scripts/ifcfg-bond <N > で使用されます(「チャンネルボンディングインターフェイス」を参照してください)。これらのパラメーターは、/sys/class/net/ <bonding device> /bonding のボンディングデバイスに使用されるパラメーターと、ボンディング モジュールのディレクティブで説明されているように、ボンディングドライバーのモジュールパラメーターと同じです。

この設定方法を使うと、複数のボンディングデバイスに異なる設定をすることが可能になります。ifcfg- <name > で BONDING_OPTS を使用する場合は、/etc/modprobe.conf を使用してボンディングデバイスのオプションを指定し ない でください。

BOOTPROTO=<protocol>

ここで、<protocol& gt; は以下のいずれかになります。

- *none - ブートタイムプロトコルは使用しないでください。*
- *BOOTP - BOOTP プロトコルを使用する必要があります。*
- *DHCP - DHCP プロトコルを使用する必要があります。*

BROADCAST=<address>

ここで、<address > はブロードキャストアドレスです。値は自動的に ipcalc で計算されるため、このディレクティブは非推奨になりました。

DEVICE=<name>

ここで、&lt ;name> は物理デバイスの名前です(論理名は となる動的に割り当てられた PPP デバイスを除く)。

DHCP_HOSTNAME=<name>

ここで、<name > は DHCP サーバーに送信される短いホスト名です。このオプションは、DHCP サーバーが IP アドレスを受け取る前にホスト名を指定する必要がある場合にのみ使用します。

DNS{1,2}=<address>

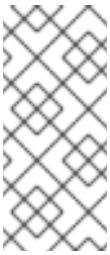
ここで、<address > は、PEERDNS ディレクティブが yes に設定されている場合に /etc/resolv.conf に配置されるネームサーバーアドレスです。

ETHTOOL_OPTS=<options>

&lt ;options& gt; は、ethtool がサポートするデバイス固有のオプションです。たとえば、100Mb に強制する場合は、完全なデュプレックスを行います。

```
ETHTOOL_OPTS="autoneg off speed 100 duplex full"
```

カスタムの init スクリプトの代わりに、ETHTOOL_OPTS を使用してインターフェイスの速度とデュプレックスの設定を行います。カスタム initscripts run は、ネットワーク init スクリプト以外で実行すると、ブート後のネットワークサービスの再起動時に予期しない結果になります。



注記

速度またはデュプレックス設定を変更するには、ほとんどの場合、オート ネガオフ オプションでオートネゴシエーションを無効にする必要があります。オプションエントリーは順序に依存するため、最初にこれを指定する必要があります。

GATEWAY=<address>

ここで、<address > は、ネットワークルーターまたはゲートウェイデバイス（存在する場合）の IP アドレスです。

HOTPLUG=<answer>

ここで、<answer& gt; は以下のいずれかになります。

- *yes - このデバイスは、ホットプラグ時にアクティベートする必要があります（これはデフォルトのオプションです）。*
- *No - このデバイスは、ホットプラグ時にアクティブ化しないでください。*

HOTPLUG=no オプションを使用すると、ボンディングカーネルモジュールがロードされたときにチャンネルボンディングインターフェイスがアクティブにならないようにすることができます。

チャンネルボンディングインターフェイスの詳細は、「チャンネルボンディングインターフェイス」 を参照してください。

HWADDR=<MAC-address>

& lt;MAC-address > は、AA:BB:CC:DD:EE:FF 形式のイーサネットデバイスのハードウェアアドレスです。このディレクティブは、各 NIC のモジュールに設定されたロード順序に関係なく、インターフェイスに正しいデバイス名が割り当てられているように、複数の NIC を含むマシンで使用する必要があります。このディレクティブは、MACADDR と併用し ない でください。

IPADDR=<address>

ここで、<address > は IP アドレスです。

LINKDELAY=<time>

& lt;time& gt; は、デバイスを設定する前にリンクネゴシエーションを待機する秒数です。

MACADDR=<MAC-address>

& lt;MAC-address > は、AA:BB:CC:DD:EE:FF 形式のイーサネットデバイスのハードウェアアドレスです。このディレクティブは、MAC アドレスをインターフェイスに割り当てるために使用され、物理 NIC に割り当てられたアドレスを上書きします。このディレクティブは、HWADDR と併用し ない でください。

MASTER=<bond-interface>

ここで、<bond-interface > は、イーサネットインターフェイスがリンクされるチャンネルボンディングインターフェイスです。

このディレクティブは、SLAVE ディレクティブとともに使用されます。

チャンネルボンディングインターフェイスの詳細は、「チャンネルボンディングインターフェイス」 を参照してください。

NETMASK=<mask>

ここで、<mask& gt; はネットマスクの値です。

NETWORK=<address>

ここで、<address > はネットワークアドレスです。値は自動的に ipcalc で計算されるため、このディレクティブは非推奨になりました。

ONBOOT=<answer>

ここで、<answer& gt; は以下のいずれかになります。

- *yes - このデバイスは、システムの起動時にアクティベートする必要があります。*
- *no - このデバイスは、システムの起動時にアクティブ化しないでください。*

PEERDNS=<answer>

ここで、<answer& gt; は以下のいずれかになります。

- *yes - DNS ディレクティブが設定されている場合は、/etc/resolv.conf を変更します。DHCP を使用している場合は、yes がデフォルトになります。*
- *No - /etc/resolv.conf は変更しないでください。*

SLAVE=<answer>

ここで、<answer& gt; は以下のいずれかになります。

- *yes - このデバイスは、MASTER ディレクティブで指定されたチャンネルボンディングインターフェイスによって制御されます。*
- *no: このデバイスは、MASTER ディレクティブで指定されたチャンネルボンディングインターフェイスで制御され ません。*

このディレクティブは、MASTER ディレクティブとともに使用されます。

チャンネルボンディングインターフェイスの詳細は、「チャンネルボンディングインターフェイス」 を参照してください。

SRCADDR=<address>

&lt ;address&gt ; は、送信パケットの指定されたソース IP アドレスです。

USERCTL=<answer>

ここで、<answer& gt; は以下のいずれかになります。

- はい - root 以外のユーザーはこのデバイスを制御できます。
- no: root 以外のユーザーはこのデバイスを制御することができません。

16.2.2. IPsec インターフェイス

以下の例は、LAN A の network-to-network IPsec 接続の ifcfg ファイルを示しています。この例では接続を識別する一意の名前は ipsec1 であるため、生成されるファイルには /etc/sysconfig/network-scripts/ifcfg-ipsec1 という名前が付けられます。

```
TYPE=IPsec
ONBOOT=yes
IKE_METHOD=PSK
SRCNET=192.168.1.0/24
DSTNET=192.168.2.0/24
DST=X.X.X.X
```

上記の例では、X.X.X.X は宛先 IPsec ルーターの公開されている IP アドレスです。

以下は、IPsec インターフェイスの設定可能なパラメーターの一覧です。

DST=<address>

ここで、<address > は IPsec 宛先ホストまたはルーターの IP アドレスです。これは、ホスト間およびネットワーク間 IPsec 設定の両方に使用されます。

DSTNET=<network>

ここで、<network > は IPsec 宛先ネットワークのネットワークアドレスです。これは、ネットワーク間 IPsec 設定にのみ使用されます。

SRC=<address>

ここで、<address > は IPsec ソースホストまたはルーターの IP アドレスです。この設定はオプションで、ホスト間の IPsec 設定にのみ使用されます。

SRCNET=<network>

ここで、<network > は IPsec ソースネットワークのネットワークアドレスです。これは、ネットワーク間 IPsec 設定にのみ使用されます。

TYPE=<interface-type>

ここで、<interface-type> は IPSEC です。どちらのアプリケーションも ipsec-tools パッケージに含まれます。

IPsec を使用した手動キー暗号化を使用する場合は、設定パラメーターについて /usr/share/doc/initscripts- <version-number> /sysconfig.txt (< version-number > を initscripts パッケージのバージョンに置き換えます)を参照してください。

racoon IKEv1 キー管理デーモンは、IPSec のパラメーターセットをネゴシエートして設定します。事前共有キー、RSA 署名、または GSS-API を使用できます。キーの暗号化を自動的に管理するために racoon が使用される場合は、以下のオプションが必要です。

IKE_METHOD=<encryption-method>

ここで、<encryption-method& gt; は PSK、X509、または GSSAPI のいずれかです。PSK を指定する場合は、IKE_PSK パラメーターも設定する必要があります。X509 を指定する場合は、IKE_CERTFILE パラメーターも設定する必要があります。

IKE_PSK=<shared-key>

ここ で、<shared-key > は PSK （事前共有鍵）メソッドの共有秘密値です。

IKE_CERTFILE=<cert-file>

ここで、<cert-file > は、ホストの有効な X.509 証明書ファイルです。

IKE_PEER_CERTFILE=<cert-file>

ここで、<cert-file > は、リモート ホストの有効な X.509 証明書ファイルです。

IKE_DNSSEC=<answer>

ここで、<answer> は yes になります。racoon デーモンは、DNS 経由でリモートホストの X.509 証明書を取得します。IKE_PEER_CERTFILE が指定されている場合は、このパラメーターを含め ない でください。

IPsec で利用可能な暗号化アルゴリズムの詳細は、setkey の man ページを参照してください。racoon の詳細は、racoon および racoon.conf の man ページを参照してください。

16.2.3. チャンネルボンディングインターフェイス

Red Hat Enterprise Linux では、管理者は ボンディングカーネルモジュールと、チャンネルボンディング インターフェイス と呼ばれる特別なネットワークインターフェイスを使用して、複数のネットワークインターフェイスを 1 つの チャンネル にバインドできます。このチャンネルボンディングにより、複数のネットワークインターフェイスが 1 つとして機能できるようになり、また同時に帯域幅が増加し、冗長性を提供します。

チャンネルボンディングインターフェイスを作成するには、/etc/sysconfig/network-scripts/ ディレクトリーに ifcfg-bond <N > という名前のファイルを作成し、< N > をそのインターフェイスの番号（例： 0 ）に置き換えます。

ファイルのコンテンツは、イーサネットインターフェイスなどボンディングされるインターフェイスのタイプと同じにすることができます。唯一の違いは、DEVICE= ディレクティブは bond <N> で、& lt; N> を インターフェイスの番号に置き換えることです。

以下は、チャンネルボンディングの設定ファイル **ifcfg-bond0** の例です。

```
DEVICE=bond0
IPADDR=192.168.1.1
NETMASK=255.255.255.0
ONBOOT=yes
BOOTPROTO=none
USERCTL=no
BONDING_OPTS="<bonding parameters separated by spaces>"
```

チャンネルボンディングインターフェイスを作成したら、設定ファイルに **MASTER=** ディレクティブおよび **SLAVE=** ディレクティブを追加して、一緒にバインドするネットワークインターフェイスを設定する必要があります。各チャンネルボンディングされたインターフェイスの設定ファイルは、ほぼ同じです。

たとえば、**2** つのイーサネットインターフェイスがチャンネルボンディングされている場合、**eth0** と **eth1** の両方の例を以下に示します。

```
DEVICE=eth<N>
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
USERCTL=no
```

この例では、**&lt ;N& gt;** をインターフェイスの数値に置き換えます。

チャンネルボンディングインターフェイスを有効にするには、カーネルモジュールを読み込む必要があります。チャンネルボンディングインターフェイスの起動時にモジュールがロードされるようにするには、以下の行を **/etc/modprobe.conf** に追加します。

```
alias bond<N> bonding
```

& lt;N& gt; を、**0** などのインターフェイス番号に置き換えます。

重要

Red Hat Enterprise Linux 5.10 では、ボンディングカーネルモジュールのインターフェイス固有のパラメーターは、ifcfg-bondN インターフェイスファイルの BONDING_OPTS="ボンディングパラメーター" ディレクティブでスペース区切りリストとして指定する必要があります。/etc/modprobe.conf ファイルでボンディングデバイスのオプションを指定し ない でください。

debug パラメーターおよび max_bonds パラメーターはインターフェイス固有ではないため、必要な場合は以下のように /etc/modprobe.conf で指定する必要があります。

```
options bonding debug=1 max_bonds=1
```

ただし、BONDING_OPTS ディレクティブで ifcfg-bondN ファイルを使用する場合は、max_bonds パラメーターを設定し ない でください。このディレクティブにより、ネットワークスクリプトが必要に応じてボンドインターフェイスを作成するためです。

/etc/modprobe.conf への変更は、モジュールが次に読み込まれるまで反映されません。実行中のモジュールを最初にアンロードする必要があります。ボンディングモジュールの設定に関する指示やアドバイス、およびボンディングパラメーターの一覧は、「Channel Bonding モジュール」 を参照してください。

16.2.4. エイリアスとクローンファイル

あまり使用されないインターフェイス設定ファイルのタイプは、エイリアス と クローン ファイルの 2 つです。

複数のアドレスを単一のインターフェイスにバインドするために使用されるエイリアスインターフェイスの設定ファイルでは、ifcfg- <if-name > : <alias-value > 命名スキームを使用します。

たとえば、ifcfg-eth0: 0 ファイルは、DEVICE=eth 0:0 と静的 IP アドレス 10.0.0.2 を指定するように設定でき、ifcfg-eth0 で DHCP を介して IP 情報を受け取るように設定済みのイーサネットインターフェイスのエイリアスとして機能します。この設定では、eth0 は動的 IP アドレスにバインドされますが、同じ物理ネットワークカードは固定の 10.0.0 .2 IP アドレスを介して要求を受信できます。

警告

エイリアスインターフェイスは DHCP をサポートしません。

クローンインターフェイス設定ファイルは、ifcfg- <if-name> - <clone-name> の命名規則を使用する必要があります。エイリアスファイルは既存のインターフェイスに対して複数のアドレスを許可しますが、クローンファイルを使用してインターフェイスの追加オプションを指定します。たとえば、eth0 と呼ばれる標準の DHCP イーサネットインターフェイスは以下のようになります。

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=dhcp
```

USERCTL ディレクティブのデフォルト値は、指定されていない場合は no であるため、ユーザーはこのインターフェイスを起動することができません。ユーザーがインターフェイスを制御できるようにするには、ifcfg-eth0 を ifcfg-eth0 -user にコピーしてクローンを作成し、以下の行を ifcfg-eth0-user に追加します。

```
USERCTL=yes
```

これにより、ifcfg-eth0 と ifcfg- eth0 -user の設定オプションが組み合わされているため、/sbin/ifup eth0- user コマンドを使用して eth0 インターフェイスを起動できます。これは非常に基本的な例ですが、この方法はさまざまなオプションおよびインターフェイスで使用できます。

エイリアスとインターフェイス設定ファイルのクローンを作成する最も簡単な方法は、グラフィカル ネットワーク管理ツール を使用することです。このツールの使用方法は、17章Network Configuration を参照してください。

16.2.5. Dialup インターフェイス

ダイアルアップ接続経由でインターネットに接続する場合は、インターフェイスに設定ファイルが必要になります。

PPP インターフェイスファイルは、以下の形式で名前が付けられます。

ifcfg-ppp<X>

ここで、<X> は特定のインターフェイスに対応する一意の番号になります。

PPP インターフェイス設定ファイルは、wvdial、Network Administration Tool、または Kppp を使用してダミーアカウントを作成すると自動的に作成されます。このファイルを手動で作成して編集することもできます。

以下は、典型的な ifcfg-ppp0 ファイルです。

```
DEVICE=ppp0
NAME=test
WVDIALSECT=test
MODEMPORT=/dev/modem
LINESPEED=115200
PAPNAME=test
USERCTL=true
ONBOOT=no
PERSIST=no
DEFROUTE=yes
PEERDNS=yes
DEMAND=no
IDLETIMEOUT=600
```

SLIP (Serial Line Internet Protocol) は別のダミーインターフェイスですが、使用頻度は低くなります。SLIP ファイルには、ifcfg-sl0 などのインターフェイス設定ファイル名があります。

これらのファイルで使用できるその他のオプション には、以下が含まれます。

DEFROUTE=<answer>

ここで、<answer& gt; は以下のいずれかになります。

- *Yes: このインターフェイスをデフォルトルートとして設定します。*
- *no: このインターフェイスをデフォルトルートとして設定しません。*

DEMAND=<answer>

ここで、<answer& gt; は以下のいずれかになります。

- *Yes : このインターフェイスを使用すると、あるユーザーが接続を使用しようとすると、pppd が接続を開始できます。*
- *No: このインターフェイスに対して手動で接続を確立する必要があります。*

IDLETIMEOUT=<value>

ここで、<value > は、インターフェイスが切断されるまでのアイドルアクティビティーの秒数です。

INITSTRING=<string>

ここで、<string > は、モデムデバイスに渡される初期化文字列です。このオプションは主に SLIP インターフェイスと併用されます。

LINESPEED=<value>

ここで、<value > はデバイスのボーレートです。使用できる標準値には、57600、38400、19200、および 9600 があります。

MODEMPORT=<device>

& lt;device& gt; は、インターフェイスの接続を確立するために使用されるシリアルデバイスの名前です。

MTU=<value>

& lt;value& gt; は、インターフェイスの Maximum Transfer Unit (MTU) 設定です。MTU は、ヘッダー情報をカウントしない、フレームが伝送できるデータの最大バイト数を指します。ダミーの状況では、この値を 576 に設定すると、パケットが破棄され、接続のスループットがわずかに向上します。

NAME=<name>

ここで、<name > は、ダイヤルアップ接続設定のコレクションに指定されたタイトルへの参照です。

PAPNAME=<name>

ここで、<name > は、リモートシステムへの接続を許可するために発生する Password Authentication Protocol (PAP) 交換中に指定されるユーザー名です。

PERSIST=<answer>

ここで、<answer& gt; は以下のいずれかになります。

- *yes: モデムのハング後に非アクティブ化された場合でも、このインターフェイスはいつでもアクティブな状態に維持する必要があります。*
- *No: このインターフェイスは、常にアクティブな状態に維持しないでください。*

REMIP=<address>

ここで、<address > はリモートシステムの IP アドレスです。通常、これは指定されないままになります。

WVDIALSECT=<name>

ここで、<name > はこのインターフェイスを /etc/wvdial.conf のダイヤル設定に関連付けます。このファイルには、ダイアル対象の電話番号と、そのインターフェイスのその他の重要な情報が含まれています。

16.2.6. その他のインターフェイス

その他の一般的なインターフェイス設定ファイルには以下が含まれます。

ifcfg-lo

ローカル ループバックインターフェイス は、多くの場合、テスト内で使用され、同じシステムを参照する IP アドレスを必要とするさまざまなアプリケーションで使用されます。ループバックデバイスに送信されたデータはすべて、ホストのネットワーク層に即座に返されます。

警告

ループバックインターフェイススクリプト /etc/sysconfig/network-scripts/ifcfg-lo は手動で編集しないでください。これを行うと、システムが正常に動作できなくなる可能性があります。

ifcfg-irlan0

infrared インターフェイス を使用すると、ラップトップやプリンターなどのデバイス間の情報を infrared リンクで流れることができます。これは、イーサネットデバイスと同様の方法で機能します。ただし、これは、一般的にピアツーピア接続を介して発生する点が異なります。

ifcfg-plip0

Parallel Line Interface Protocol (PLIP) 接続は、イーサネットデバイスと同じように機能しますが、並列ポートを使用する点が異なります。

ifcfg-tr0

トークンリング トポロジーは、以前と同様に ローカルエリアネットワーク (LAN)で共通ではありません。イーサネットによって Eclipse を持っています。

16.3. インターフェイス制御スクリプト

インターフェイス制御スクリプトは、システムインターフェイスをアクティブまたは非アクティブにします。/etc/sysconfig/network-scripts/ ディレクトリーにある制御スクリプトを呼び出す主なインターフェイス制御スクリプトは、/sbin/ifdown および /sbin/ifup の 2 つです。

ifup および ifdown インターフェイススクリプトは、/sbin/ ディレクトリーのスクリプトへのシンボリックリンクです。これらのスクリプトのいずれかが呼び出されると、以下のようなインターフェイス

の値を指定する必要があります。

```
ifup eth0
```

警告

ifup および ifdown インターフェイススクリプトは、ユーザーがネットワークインターフェイスを起動してダウンさせるために使用する唯一のスクリプトです。

以下のスクリプトは、参照の目的でのみ説明されています。

ネットワークインターフェイスの起動プロセス中にさまざまなネットワーク初期化タスクを実行するために使用される 2 つのファイルは、/etc/rc.d/init.d/functions および /etc/sysconfig/network-scripts/network-functions です。詳細は、「ネットワーク機能仮想化ファイル」 を参照してください。

インターフェイスが指定され、要求を実行しているユーザーがインターフェイスを制御できることを確認したら、正しいスクリプトによりインターフェイスが起動します。以下は、/etc/sysconfig/network-scripts/ ディレクトリーにある一般的なインターフェイス制御スクリプトです。

ifup-aliases

複数の IP アドレスがインターフェイスに関連付けられている場合は、インターフェイス設定ファイルから IP エイリアスを設定します。

ifup-ippp および ifdown-ippp

ISDN インターフェイスをアップまたはダウンにします。

ifup-ipsec および ifdown-ipsec

IPsec インターフェイスをアップおよびダウンにします。

ifup-ipv6 および ifdown-ipv6

IPv6 インターフェイスをアップまたはダウンにします。

ifup-ipx

IPX インターフェイスを起動します。

ifup-plip

PLIP インターフェイスを起動します。

ifup-plusb

ネットワーク接続用の USB インターフェイスを起動します。

ifup-post および ifdown-post

インターフェイスの起動またはダウン後に実行されるコマンドが含まれます。

ifup-ppp および ifdown-ppp

PPP インターフェイスをアップまたはダウンにします。

ifup-routes

デバイスの静的ルートを、インターフェイスの起動時に追加します。

ifdown-sit および ifup-sit

IPv4 接続内での IPv6 トンネルのアップとダウンに関連する関数呼び出しが含まれます。

ifup-sl および ifdown-sl

SLIP インターフェイスをアップまたはダウンにします。

ifup-wireless

ワイヤレスインターフェイスを起動します。

警告

/etc/sysconfig/network-scripts/ ディレクトリーのスクリプトを削除または変更すると、インターフェイス接続が不規則に機能するか、失敗する可能性があります。ネットワークインターフェイスに関連するスクリプトを変更する必要があるのは、上級ユーザーのみです。

すべてのネットワークスクリプトを同時に操作する最も簡単な方法は、以下のコマンドに示すように、ネットワークサービス(/etc/rc.d/init.d/network)で /sbin/service コマンドを使用することです。

```
service network <action>
```

ここで、&lt ;action> は を 起動 するか、を停止 するか、を 再起動 することができます。

設定されているデバイスの一覧と現在アクティブなネットワークインターフェイスを表示するには、以下のコマンドを使用します。

```
service network status
```

16.4. 静的ルートおよびデフォルトゲートウェイ

静的ルートは、デフォルトゲートウェイを通過してはならないトラフィック用です。ルーティングは、しばしば、ルーティング専用のネットワーク上で、デバイスにより処理されます (ただし、デバイスはルーティングを行うように設定できます)。したがって、Red Hat Enterprise Linux サーバーまたはクライアントで静的ルートを設定する必要がない場合もしばしばあります。例外は、暗号化された VPN トンネルを通過する必要があるトラフィックや、コストやセキュリティー上の理由から、特定のルートを通過する必要があるトラフィックが含まれます。デフォルトゲートウェイは、ローカルネット

ワーク宛ではなく、ルーティングテーブルで優先ルートが指定されていないすべてのトラフィックに適用されます。デフォルトゲートウェイは、従来専用のネットワークルーターです。

コマンドラインを使用した静的ルートの設定

静的ルートが必要な場合は、**ip route add** コマンドでルーティングテーブルに追加し、**ip route del** コマンドを使用して削除できます。より頻繁に使用される **ip route** コマンドは、

```
ip route [ add | del | change | append | replace ] destination-address
```

の形式を取ります。オプションおよび形式の詳細は、**man** ページの **ip-route (8)** を参照してください。

ip route コマンドをオプションなしで使用して、**IP** ルーティングテーブルを表示します。以下に例を示します。

```
~]$ ip route
default via 192.168.122.1 dev eth0  proto static  metric 1024
192.168.122.0/24 dev ens9  proto kernel  scope link  src 192.168.122.107
192.168.122.0/24 dev eth0  proto kernel  scope link  src 192.168.122.126
```

ホストアドレス（つまり単一の **IP** アドレス）に静的ルートを追加するには、**root** で以下のコマンドを実行します。

```
~]# ip route add 192.0.2.1 via 10.0.0.1 [dev ifname]
```

ここでの **192. 0.2.1** は、ドット付き **10** 進数表記のホストの **IP** アドレスに、**10.0.0.1** はネクストホップアドレスに、**ifname** は次のホップにつながる終了インターフェイスです。

ネットワーク（つまり **IP** アドレスの範囲を表す **IP** アドレス）に静的ルートを追加するには、**root** で以下のコマンドを実行します。

```
~]# ip route add 192.0.2.0/24 via 10.0.0.1 [dev ifname]
```

ここでの **192.0.2.0** はドット形式 **10** 進法での宛先ネットワークの **IP** アドレスに、**/24** はネットワーク接頭辞になります。ネットワーク接頭辞は、サブネットマスク内の有効なビット数です。ネットワークアドレスにスラッシュ、ネットワーク接頭辞長を続けるこの形式は、**classless inter-domain routing (CIDR)** 表記と呼ばれることもあります。

静的ルート設定は、インターフェイスごとに **/etc/sysconfig/network-scripts/route-**インターフェイス ファイルに保存できます。たとえば、の静的ルートです。 **eth0** インターフェイスは **/etc/sysconfig/network-scripts/route-eth0** ファイルに保存されます。**route-**インターフェイス ファイ

ルには、ip コマンド引数とネットワーク/ネットマスクディレクティブの 2 つの形式があります。これについては、以下で説明します。

ip route コマンドに関する詳細情報は、ip-route(8) man ページを参照してください。

デフォルトゲートウェイの設定

デフォルトゲートウェイは、ネットワークスクリプトにより決定されます。これは、最初に /etc/sysconfig/network を解析し、「up」 状態のインターフェイスについてネットワークインターフェイス ifcfg ファイルを解析します。ifcfg ファイルは数字の小さい順に解析され、最後に読み取られる GATEWAY ディレクティブがルーティングテーブルのデフォルトルートを作成するために使用されます。

そのため、デフォルトのルートは GATEWAY ディレクティブで指定でき、グローバルまたはインターフェイス固有の設定ファイルで指定することができます。ゲートウェイをグローバルに指定すると、特に複数のネットワークインターフェイスが存在する場合に、静的ネットワーク環境では特定の利点があります。一貫して適用すると、障害の検索が簡単になります。グローバルオプションである GATEWAYDEV ディレクティブもあります。複数のデバイスが GATEWAY を指定し、1 つのインターフェイスが GATEWAYDEV ディレクティブを使用する場合は、そのディレクティブが優先されます。このオプションは、インターフェイスがダウンし、障害検出が複雑になる可能性があるため、予期せぬ結果を招く可能性があるため推奨されません。

グローバルデフォルトゲートウェイ設定は /etc/sysconfig/network ファイルに保存されます。このファイルは、すべてのネットワークインターフェイスのゲートウェイおよびホスト情報を指定します。このファイルと、ファイルで使用できるディレクティブの詳細は、「/etc/sysconfig/network」 を参照してください。

16.5. IFCFG ファイルでの静的ルートの設定

コマンドプロンプトで ip コマンドを使用して設定した静的ルートは、システムがシャットダウンまたは再起動すると失われます。静的ルートの設定を、システムを再起動した後も持続するようにするには、/etc/sysconfig/network-scripts/ ディレクトリーに保存されているインターフェイス別の設定ファイルに追加する必要があります。ファイル名は、route-ifname の形式で指定する必要があります。設定ファイルで使用するコマンドには 2 種類あります。「IP コマンド引数形式を使用した静的ルート」 で説明されている ip コマンドと、「ネットワーク/ネットマスクディレクティブの形式」 で説明しているように Network/Netmask フォーマットの 2 つのタイプがあります。

16.5.1. IP コマンド引数形式を使用した静的ルート

インターフェイスごとの設定ファイル（例： /etc/sysconfig/network-scripts/route-eth0 ）で必要な場合は、1 行目でデフォルトゲートウェイへのルートを定義します。これは、ゲートウェイが DHCP 経由で設定されておらず、/etc/sysconfig/network ファイルでグローバルに設定されていない場合にのみ必要です。

> ***default via 192.168.1.1 dev interface***

ここでの 192.168.1.1 は、デフォルトゲートウェイの IP アドレスになります。interface は、デフォルトゲートウェイに接続されている、または到達可能なインターフェイスになります。dev オプションは省略できます。これはオプションです。この設定は、/etc/sysconfig/network ファイルの設定よりも優先されます。

リモートネットワークへのルートが必要な場合は、静的ルートは以下のように指定できます。各行は、個別のルートとして解析されます。

> ***10.10.10.0/24 via 192.168.1.1 [dev interface]***

ここでの 10.10.10.0/24 は、リモートもしくは宛先ネットワークのネットワークアドレスおよび接頭辞長です。アドレス 192.168.1.1 は、リモートネットワークに続く IP アドレスです。ネクストホップアドレス の方が好ましいですが、出口インターフェイスのアドレスでも機能します。「ネクストホップ」とは、ゲートウェイやルーターなどリンクのリモート側を意味します。dev オプションを使用して、終了 インターフェイス を指定できますが、必須ではありません。必要に応じて静的ルートを追加します。

以下は、ip コマンド引数形式を使用した route-interface ファイルの例です。デフォルトゲートウェイは 192.168.0.1 です。 eth0 リース行または WAN 接続が 192.168.0.10 で利用できます。2 つの静的ルートは、10.10.10.0/24 ネットワークおよび 172.16.1.10/32 ホストに到達するためのものです。

```
default via 192.168.0.1 dev eth0
10.10.10.0/24 via 192.168.0.10 dev eth0
172.16.1.10/32 via 192.168.0.10 dev eth0
```

上記の例では、ローカルの 192.168.0.0/24 ネットワークに向かうパケットはそのネットワークに接続されているインターフェイスに移動します。10.10.10.0/24 ネットワークおよび 172.16.1.10/32 ホストに向かうパケットは、192.168.0.10 に移動します。既知でないリモートネットワークに向かうパケットはデフォルトゲートウェイを使用するので、デフォルトルートが適切でない場合は、静的ルートはリモートネットワークもしくはホスト用のみに設定すべきです。ここでのリモートとは、システムに直接繋がれていないネットワークやホストを指します。

出口インターフェイスの指定は、オプションです。特定のインターフェイスからトラフィックを強制的に締め出したい場合は、これが便利です。たとえば、VPN の場合、リモートネットワークへのトラフィックが通過するように強制できます。 tun0 インターフェイスが宛先ネットワークとは別のサブネットにある場合でも、インターフェイス。

重複するデフォルトゲートウェイ

デフォルトゲートウェイがすでに DHCP から割り当てられている場合、IP コマンドの引数形式により、起動中に起動中または ifup コマンドを使用して down 状態からインターフェイスを起動する際に、RTNETLINK 応答の 2 つのエラーが発生する可能性があります。 ファイルが存在するか、Error: either "to" が重複しているか、またはX.X.X.Xはガベージコレクションです。X.X.X.X はゲートウェイ、または別の IP アドレスです。これらのエラーは、デフォルトゲートウェイを使用して別のネットワークへの別のルートがある場合にも発生する可能性があります。これらのエラーはいずれも無視しても安全です。

16.5.2. ネットワーク/ネットマスクディレクティブの形式

ネットワーク/ネットマスクディレクティブの形式を route-interface ファイルに使用することも可能です。以下は、ネットワーク/ネットマスク形式のテンプレートで、後に説明が続きます。

```
ADDRESS0=10.10.10.0
NETMASK0=255.255.255.0
GATEWAY0=192.168.1.1
```

- *ADDRESS0=10.10.10. 0 は、到達するリモートネットワークまたはホストのネットワークアドレスです。*
- *NETMASK0=255.255.255. 0 は、ADDRESS0=10.10.10.0 で定義されているネットワークアドレスのネットマスク です。*
- *GATEWAY0=192.168. 1.1 はデフォルトゲートウェイ、または ADDRESS0=10.10.10.0 に到達するために使用できる IP アドレスです。*

以下は、ネットワーク/ネットマスクディレクティブの形式を使用した route-interface ファイルの例です。デフォルトゲートウェイは 192.168.0.1 ですが、専用回線または WAN 接続が 192.168.0.10 で利用可能です。2 つの静的ルートは、10.10.10.0/24 および 172.16.1.0/24 のネットワークに到達するためのものです。

```
ADDRESS0=10.10.10.0
NETMASK0=255.255.255.0
GATEWAY0=192.168.0.10
ADDRESS1=172.16.1.10
NETMASK1=255.255.255.0
GATEWAY1=192.168.0.10
```

後に続く静的ルートは、順番に番号付けされる必要があり、いずれの値もスキップしてはいけませ

ん。たとえば、ADDRESS0、ADDRESS1、ADDRESS2 などです。

16.6. ネットワーク機能仮想化ファイル

Red Hat Enterprise Linux は、インターフェイスをアップまたはダウンするのに使用する重要な関数を含む複数のファイルを利用します。各インターフェイス制御ファイルにこれらの関数を含めるように強制するのではなく、必要に応じて呼び出されるいくつかのファイルでグループ化されます。

/etc/sysconfig/network-scripts/network-functions ファイルには、最も一般的に使用される IPv4 機能が含まれています。これは、多くのインターフェイス制御スクリプトに役立ちます。これらの関数には、で変更に関する情報を要求する実行中のプログラムへの問い合わせ、ホスト名の設定、ゲートウェイデバイスの検索、a 特定のデバイスがダウンしているかどうかの確認、およびデフォルトルートの追加が含まれます。

IPv6 インターフェイスに必要な機能は IPv4 インターフェイスとは異なるため、/etc/sysconfig/network-scripts/network-functions-ipv6 ファイルはこの情報を保持するために特別に存在します。このファイルの関数は、静的 IPv6 ルートの設定および削除、トンネルの作成と削除、IPv6 アドレスのインターフェイスへの追加および削除、インターフェイスに IPv6 アドレスの存在のテストを行います。

16.7. 関連情報

以下は、ネットワークインターフェイスの詳細を説明するリソースです。

16.7.1. インストールされているドキュメント

/usr/share/doc/initscripts-<version>/sysconfig.txt

本章で説明されていない IPv6 オプションなど、ネットワーク設定ファイルで利用可能なオプションについてのガイド。

/usr/share/doc/iproute-<version>/ip-cref.ps

このファイルには、ip コマンドに関する多くの情報が含まれています。これはルーティングテーブルの操作に使用できます。ggv または kghostview アプリケーションを使用してこのファイルを表示します。

第17章 NETWORK CONFIGURATION

相互に通信するには、コンピューターにネットワーク接続が必要です。これは、オペレーティングシステムがインターフェイスカード（イーサネット、ISDN モデム、またはトークンリングなど）を認識し、インターフェイスをネットワークに接続するように設定することによって実現されます。

Network Administration Tool を使用すると、以下の種別のネットワークインターフェイスを設定できます。

- イーサネット
- ISDN
- modem
- xDSL
- トークンリング
- CIPE
- ワイヤレスデバイス

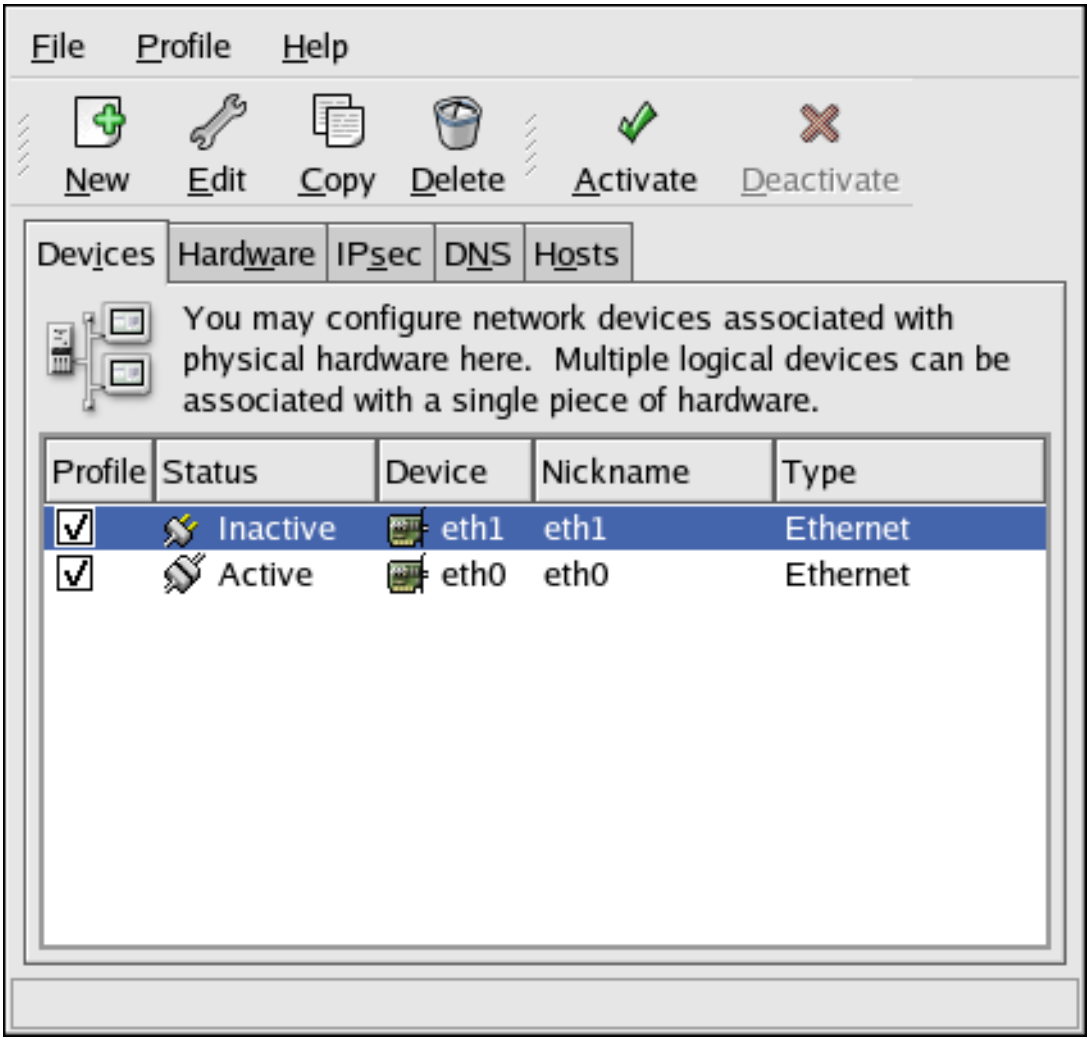
また、IPsec 接続の設定、DNS 設定の管理、追加のホスト名と IP アドレスの組み合わせの保存に使用される /etc/hosts ファイルの管理にも使用できます。

Network Administration Tool を使用するには、root 権限が必要です。アプリケーションを起動するには、アプリケーション（パネルのメインメニュー）> System Settings > Network に移動します。または、シェルプロンプトでコマンド system-config-network を入力します（例： XTerm または GNOME ターミナル）。コマンドを入力すると、X が実行されている場合、グラフィカルバージョンが表示されます。それ以外の場合は、テキストベースのバージョンが表示されます。

コマンドラインバージョンを使用するには、root でコマンド system-config-network-cmd --help

を実行して、すべてのオプションを表示します。

図17.1 ネットワーク管理ツール



[D]

ヒント

Red Hat Hardware Compatibility List (http://hardware.redhat.com/hcl/)を使用して、Red Hat Enterprise Linux がハードウェアデバイスをサポートしているかどうかを確認します。

17.1. 概要

Network Administration Tool でネットワーク接続を設定するには、以下の手順を実行します。

1. *物理ハードウェアデバイスに関連付けられたネットワークデバイスを追加します。*

2. 物理ハードウェアデバイスがまだ存在しない場合は、ハードウェア一覧に追加します。

3. ホスト名および DNS 設定を設定します。

4. DNS 経由で検索できないホストを設定します。

本章では、ネットワーク接続の種類ごとにこれらの手順について説明します。

17.2. イーサネット接続の確立

イーサネット接続を確立するには、ネットワークインターフェイスカード(NIC)、ネットワークケーブル（通常は CAT5 ケーブル）、および接続するネットワークが必要です。異なるネットワークが異なるネットワーク速度を使用するように設定されている。NIC が接続するネットワークと互換性があることを確認してください。

イーサネット接続を追加するには、以下の手順に従います。

1. Devices タブをクリックします。

2. ツールバーの New ボタンをクリックします。

3. デバイスタイプ 一覧から イーサネット接続 を 選択し、Forward をクリックします。

4. ネットワークインターフェイスカードをハードウェア一覧に追加した場合は、イーサネットカード リストから選択します。それ以外の場合は、Other Ethernet Card を選択してハードウェアデバイスを追加します。

注記

インストールプログラムは、対応しているイーサネットデバイスを検出し、それらを設定するよう要求します。インストール中にイーサネットデバイスを設定した場合は、ハードウェア タブのハードウェア一覧に表示されます。

5.

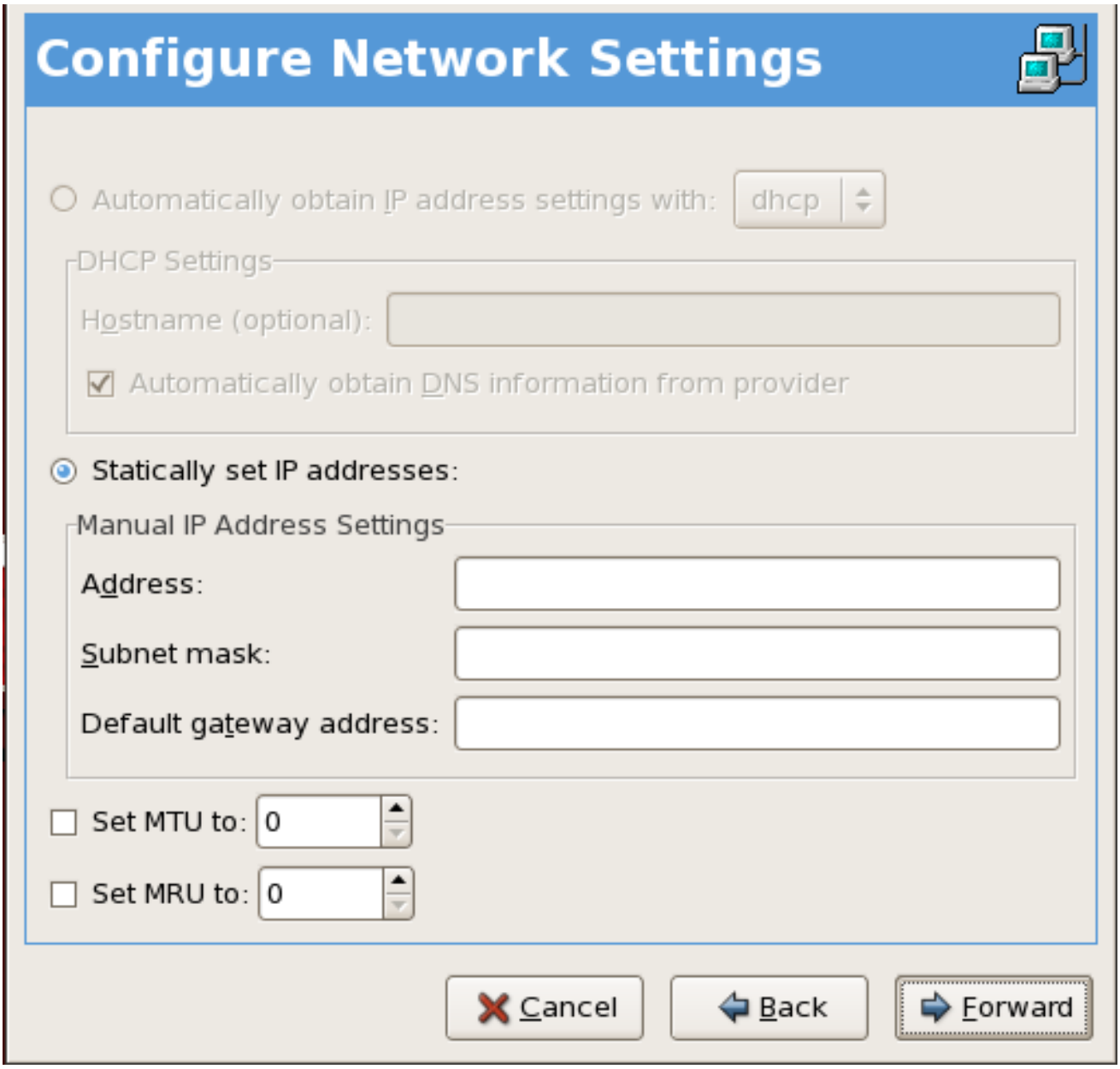
Other Ethernet Card を選択した場合は、Select Ethernet Adapter ウィンドウが表示されます。イーサネットカードの製造元とモデルを選択します。デバイス名を選択します。これがシステムの最初のイーサネットカードの場合は、デバイス名として eth0 を選択します。2 番目のイーサネットカードの場合は、eth1 （など）を選択します。ネットワーク管理ツール では、NIC のリソースを設定することもできます。進む をクリックして続けます。

6. *の Configure Network Settings ウィンドウで、DHCP と静的 IP アドレスを選択します。図17.2「イーサネット設定」ネットワークが起動されるたびにデバイスが異なる IP アドレスを受信する場合は、ホスト名を指定しないでください。*

7. *Set MTU の値を に指定しないでください。または MRU を フィールドに設定 しないでください。MTU は Maximum Transmission Unit を、Maximum Receive Unit の MRU を表します。ネットワーク設定ツールは、これらのパラメーターの両方に適切な値を選択します。進むをクリックして続けます。*

8. *Create Ethernet Device ページで Apply をクリックします。*

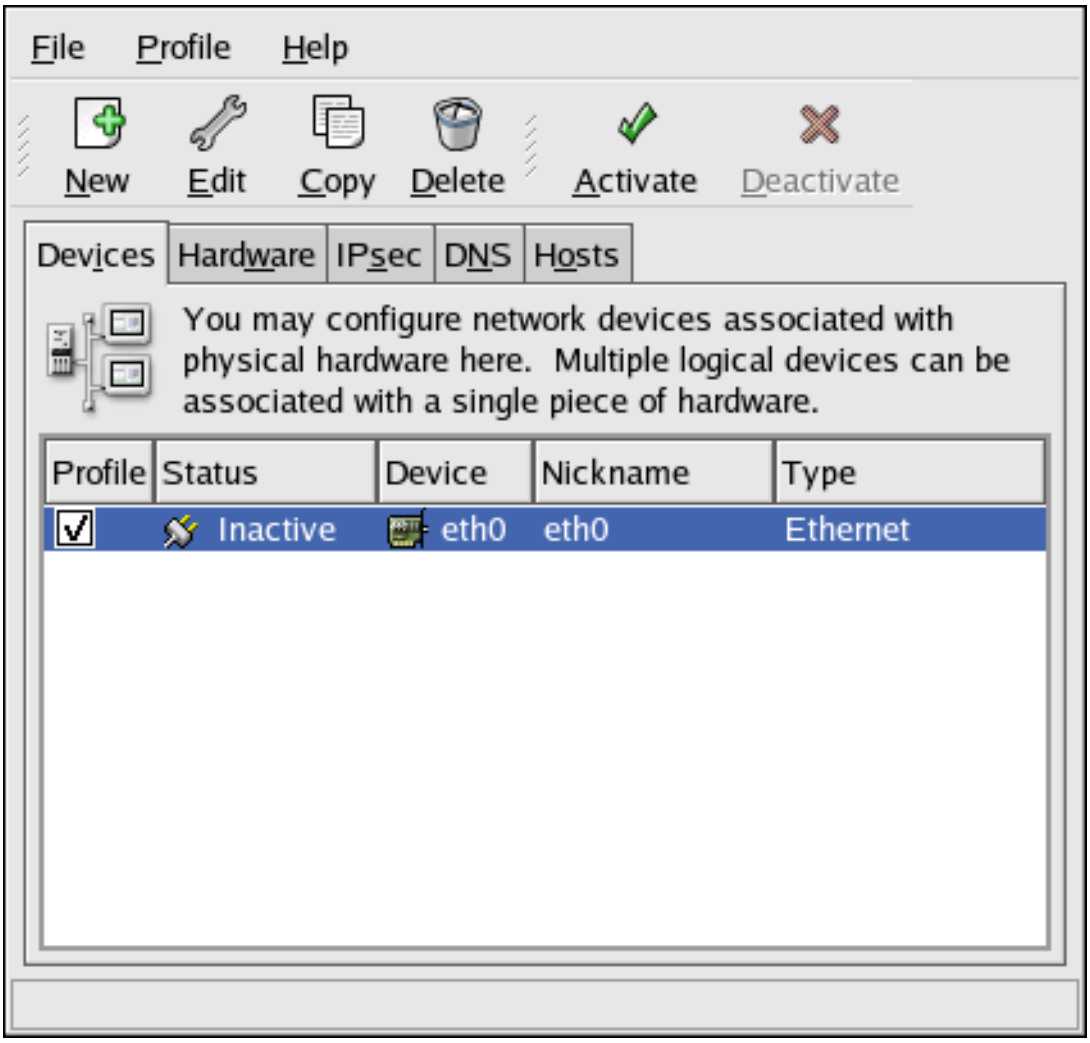
図17.2 イーサネット設定



[D]

イーサネットデバイスの設定後、図17.3「イーサネットデバイス」に示されるように、デバイス一覧に表示されます。

図17.3 イーサネットデバイス



[D]

***File > Save** を選択して変更を保存します。*

*イーサネットデバイスを追加した後、デバイス一覧からデバイスを選択し、**Edit** をクリックして設定を編集できます。たとえば、デバイスが追加されると、デフォルトでブート時に起動するように設定されています。この設定を変更するには、を選択してデバイスを編集し、コンピューターの起動時に **Activate device** を変更し、変更を保存します。*

*デバイスが追加されると、**Inactive** ステータスにあるように、すぐにアクティブ化されません。デバイスを有効にするには、デバイスリストから選択し、**Activate** ボタンをクリックします。コンピューターの起動時にデバイスをアクティベートするようにシステムが設定されている場合（デフォルト）、この手順を再度実行する必要はありません。*

複数のデバイスをイーサネットカードに関連付けると、後続のデバイスは デバイスエイリアス になります。デバイスエイリアスを使用すると、1 つの物理デバイスに複数の仮想デバイスを設定できるた

め、1 つ以上の IP アドレスを割り当てることができます。たとえば、eth1 デバイスと eth1:1 デバイスを設定できます。詳細は、「デバイスエイリアス」 を参照してください。

17.3. ISDN 接続の確立

ISDN 接続は、電話会社によってインストールされた特別な電話ラインを介して ISDN モデムカードで確立されたインターネット接続です。ISDN 接続はヨーロッパで人気があります。

ISDN 接続を追加するには、以下の手順に従います。

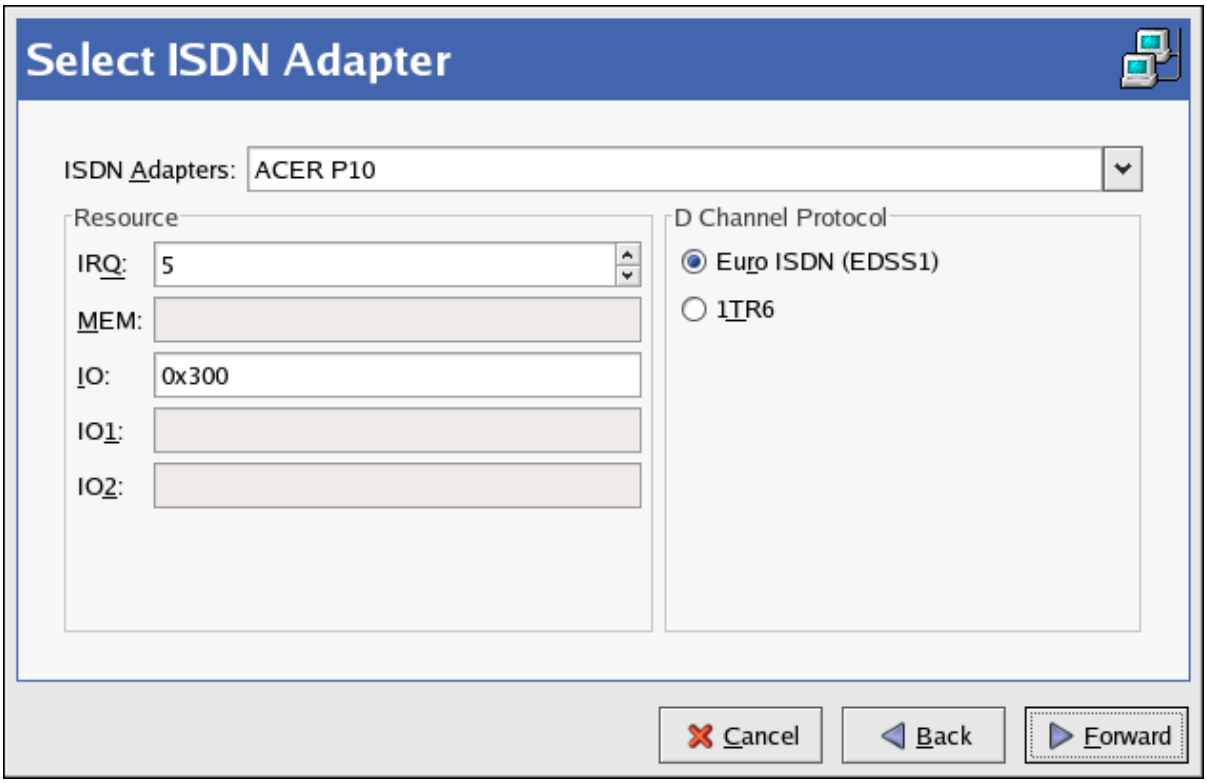
1. Devices タブをクリックします。

2. ツールバーの New ボタンをクリックします。

3. Device Type 一覧から ISDN 接続 を選択し、Forward をクリックします。

4. プルダウンメニューから ISDN アダプターを選択します。次に、アダプターのリソースおよび D チャネルプロトコルを設定します。進む をクリックして続けます。

図17.4 ISDN 設定



[D]

5.

インターネットサービスプロバイダー(ISP)が事前設定された一覧にある場合は、これを選択します。それ以外の場合は、ISP アカウントに必要な情報を入力します。値が分からない場合は、ISP にお問い合わせください。Forward をクリックします。

6.

IP Settings ウィンドウで Encapsulation Mode を選択し、IP アドレスを自動的に取得するか、または a static IP を設定するかどうかを選択します。終了したら Forward （進む）をクリックします。

7.

Create Dialup Connection ページで Apply をクリックします。

ISDN デバイスの設定後に、図17.5「ISDN デバイス」 に示されるように、デバイス一覧にタイプ ISDN が設定されたデバイスとして表示されます。

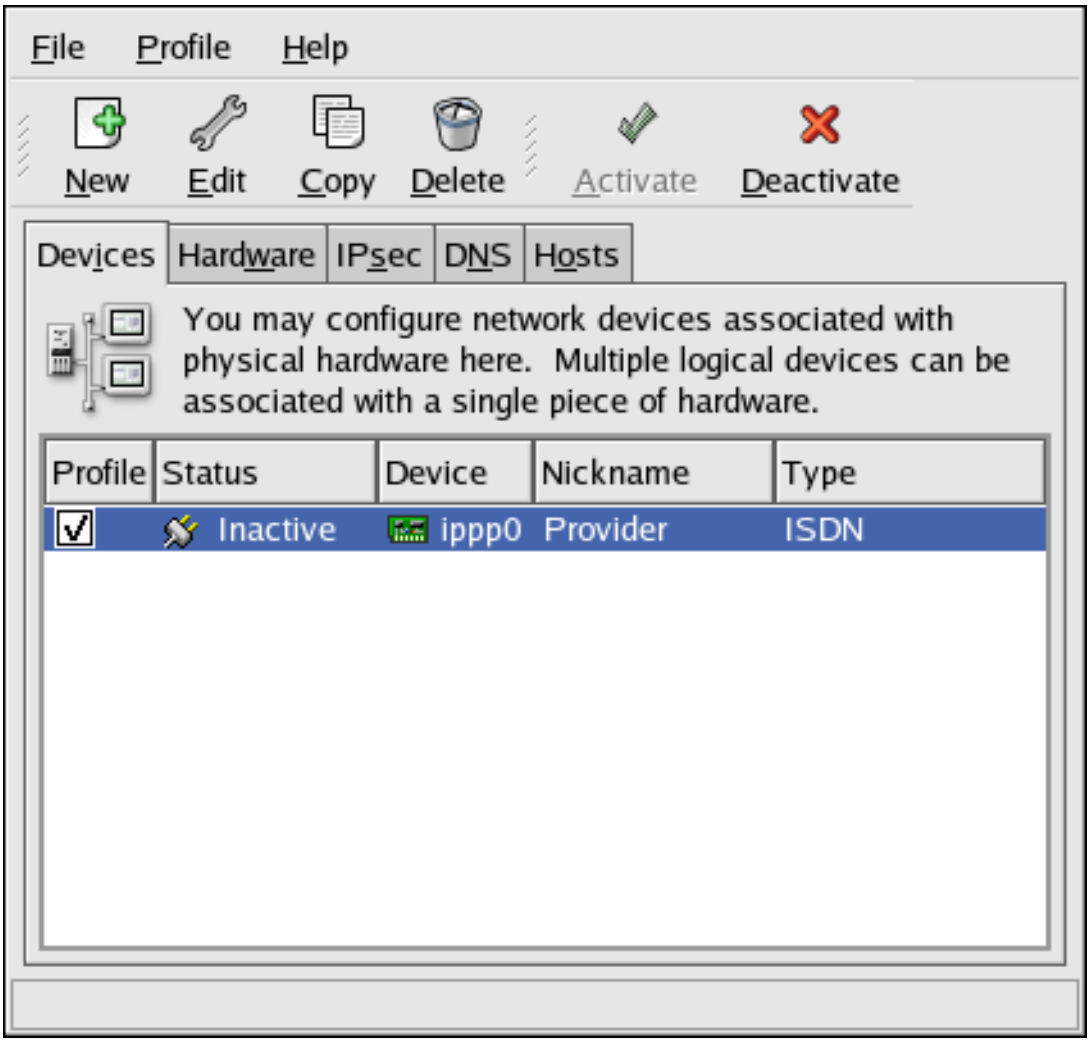
File > Save を選択して変更を保存します。

ISDN デバイスの追加後に、デバイス一覧からデバイスを選択し、Edit をクリックして設定を編集できます。たとえば、デバイスが追加されると、デフォルトではブート時に起動しないように設定されて

います。設定を編集してこの設定を変更します。圧縮、PPP オプション、ログイン名、パスワードなどを変更できます。

デバイスが追加されると、Inactive ステータスにあるように、すぐにアクティブ化されません。デバイスを有効にするには、デバイスリストから選択し、Activate ボタンをクリックします。コンピューターの起動時にデバイスをアクティベートするようにシステムが設定されている場合（デフォルト）、この手順を再度実行する必要はありません。

図17.5 ISDN デバイス



[D]

17.4. モデム接続の確立

モデムを使用して、アクティブな電話でインターネット接続を設定できます。インターネットサービスプロバイダー(ISP)アカウント（別名ダイヤルアップアカウント）が必要です。

モデム接続を追加するには、以下の手順に従います。

1.
Devices タブをクリックします。

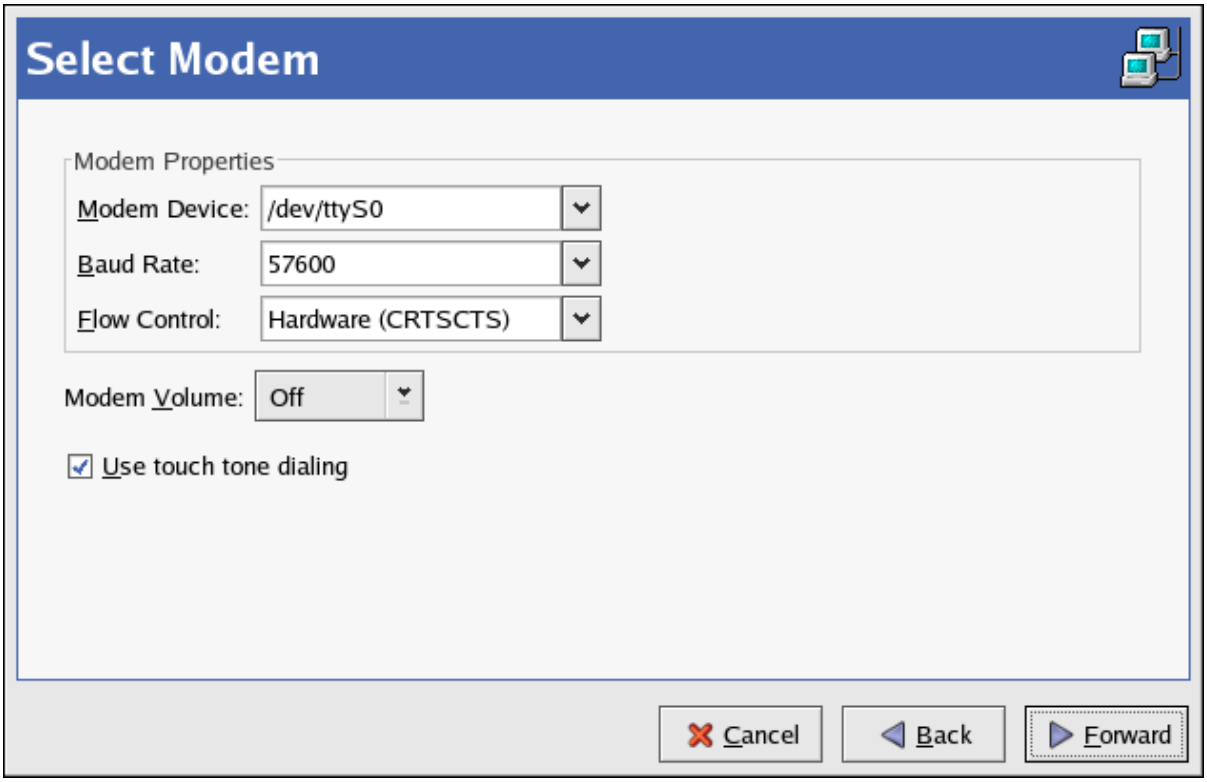
2.
ツールバーの New ボタンをクリックします。

3.
Device Type 一覧から Modem connection を選択し、Forward をクリックします。

4.
ハードウェア一覧にモデムが設定されている場合（ハードウェアタブ上）、Network Administration Tool は、これを使用してモデム接続を確立することを想定します。すでにモードが設定されていない場合は、システム内のモデムを検出しようとします。このプローブには時間がかかる場合があります。モデムが見つからない場合、表示される設定がプローブから見つからないことを警告するメッセージが表示されます。

5.
プローブ後、図17.6「モデム設定」 のウィンドウが表示されます。

図17.6 モデム設定



[D]

6.
モデムデバイス、ボーレート、フロー制御、およびモデムボリュームを設定します。これらの値が分からない場合は、モデムが正常にプローブされた場合は、デフォルトを受け入れます。ダイヤルトが必要ない場合は、対応するチェックボックスの選択を解除します。Forward をクリックします。

7.

ISP が事前設定された一覧にある場合は選択します。それ以外の場合は、ISP アカウントに必要な情報を入力します。これらの値が分からない場合は、ISP にお問い合わせください。Forward をクリックします。

8.

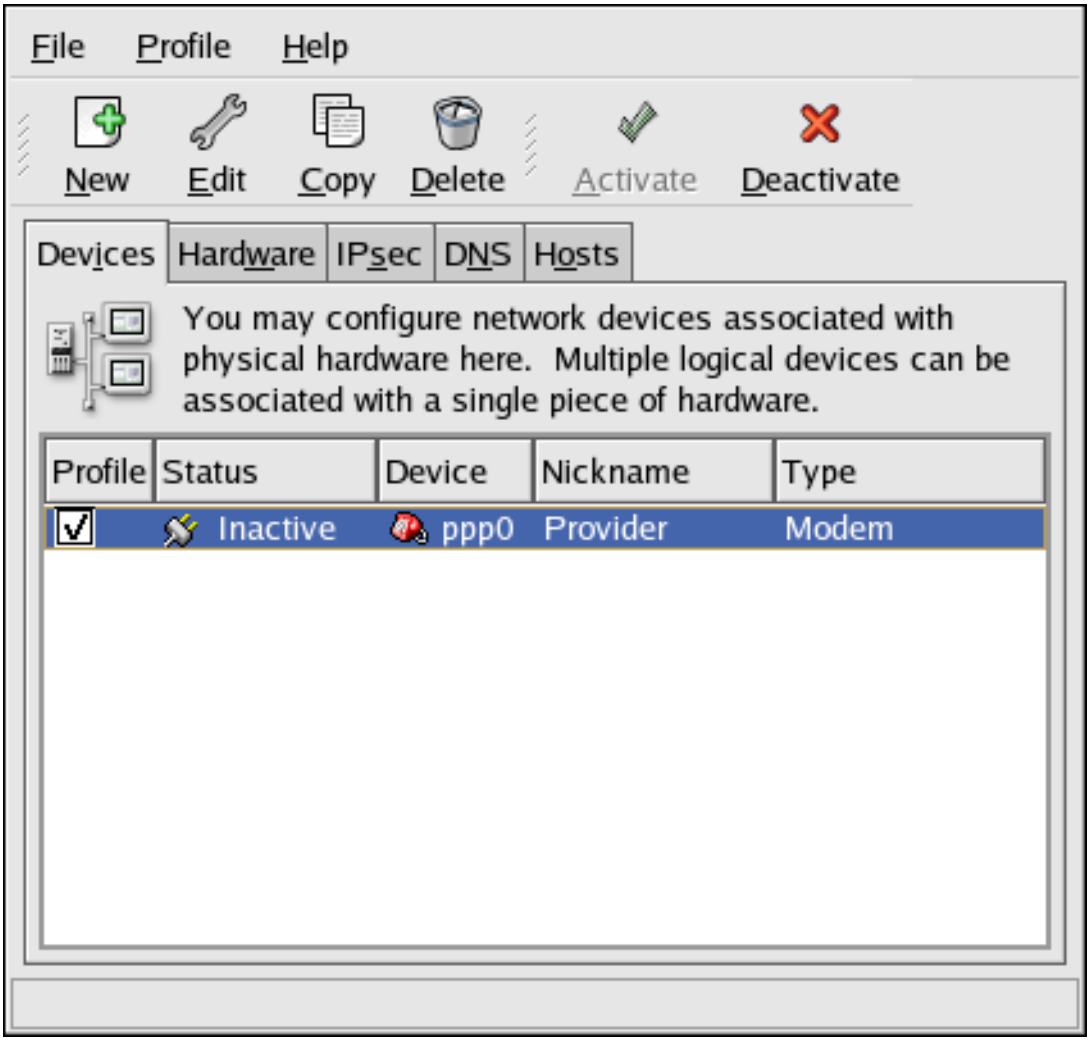
IP Settings ページで、IP アドレスを自動的に取得するか、1 つの静的に設定するかどうかを選択します。終了したら Forward （進む）をクリックします。

9.

Create Dialup Connection ページで Apply をクリックします。

mm デバイスの設定後、図17.7「Modem Device」 に示されているように、Modem タイプのデバイス一覧に表示されます。

図17.7 Modem Device



[D]

File > Save を選択して変更を保存します。

mm デバイスを追加したら、デバイス一覧からデバイスを選択し、Edit をクリックして設定を編集できます。たとえば、デバイスが追加されると、デフォルトではブート時に起動しないように設定されています。設定を編集してこの設定を変更します。圧縮、PPP オプション、ログイン名、パスワードなどを変更することもできます。

デバイスが追加されると、Inactive ステータスにあるように、すぐにアクティブ化されません。デバイスを有効にするには、デバイスリストから選択し、Activate ボタンをクリックします。コンピューターの起動時にデバイスをアクティベートするようにシステムが設定されている場合（デフォルト）、この手順を再度実行する必要はありません。

17.5. XDSL 接続の確立

DSL は、Digital Subscriber Lines を表します。ADSL、IDSL、および SDSL などの異なる DSL があります。ネットワーク管理ツール は、xDSL という用語を使用して、すべてのタイプの DSL 接続を意味します。

一部の DSL プロバイダーでは、イーサネットカードを使用して DHCP を介して IP アドレスを取得するようにシステムを設定する必要があります。一部の DSL プロバイダーでは、イーサネットカードを使用して PPPoE (Point-to-Point Protocol over Ethernet)接続を設定する必要があります。使用する DSL プロバイダーに依頼します。

DHCP を使用する必要がある場合は、「イーサネット接続の確立」 でイーサネットカードを設定します。

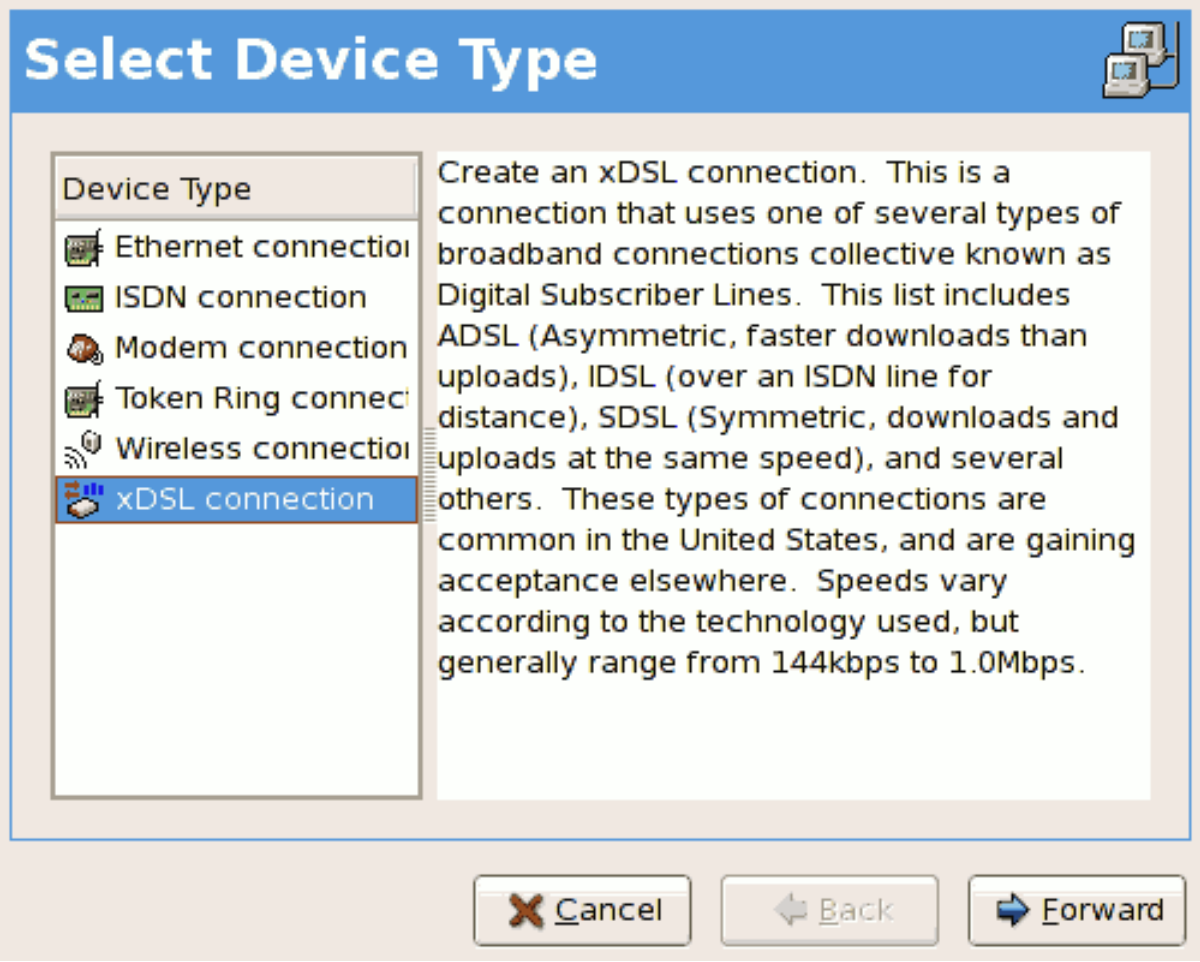
PPPoE を使用する必要がある場合は、以下の手順に従います。

1. Devices タブをクリックします。

2. New ボタンをクリックします。

3. Device Type 一覧から xDSL 接続 を選択し、図17.8「デバイスタイプの選択」 に示されるように Forward をクリックします。

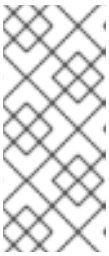
図17.8 デバイスタイプの選択



[D]

4.

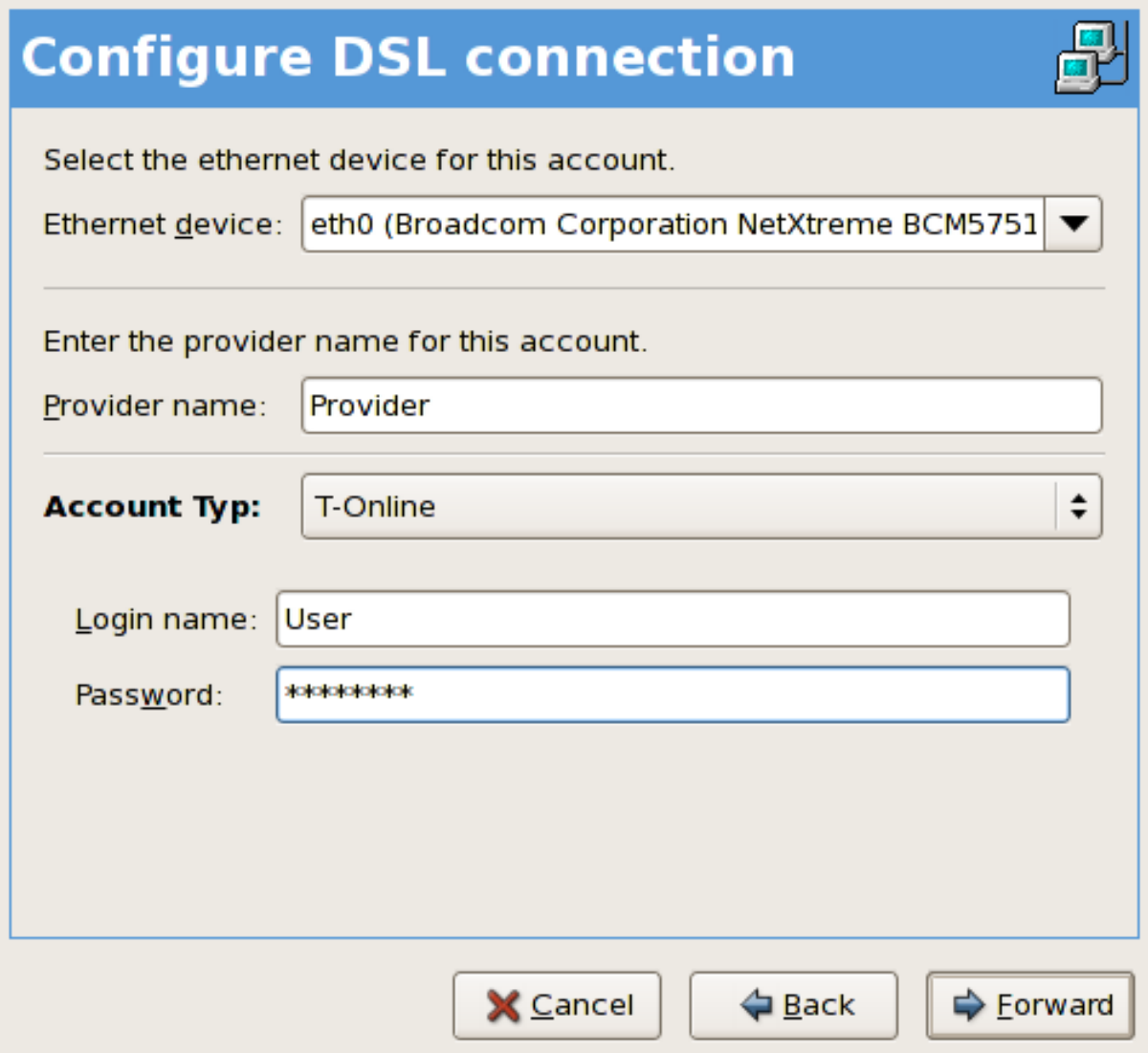
イーサネットカードがハードウェア一覧に存在する場合は、図17.9「xDSL 設定」 に記載されているページから イーサネットデバイス を選択します。それ以外の場合は、Select Ethernet Adapter ウィンドウが表示されます。



注記

インストールプログラムは、対応しているイーサネットデバイスを検出し、それらを設定するよう要求します。インストール中にイーサネットデバイスを設定した場合は、ハードウェア タブのハードウェア一覧に表示されます。

図17.9 xDSL 設定



[D]

5.

Provider Name、Login Name、および Password を入力します。T-On ラインアカウントを設定していない場合は、Account Type プルダウンメニューから Normal を選択します。

T-On ラインアカウントを設定する場合は、Account Type プルダウンメニューから T-Online を選択し、Login name および Password フィールドに任意の値を入力します。DSL 接続が完全に設定されたら、T-On のアカウント設定をさらに設定できます(T-On オンラインアカウントの設定を参照してください)。

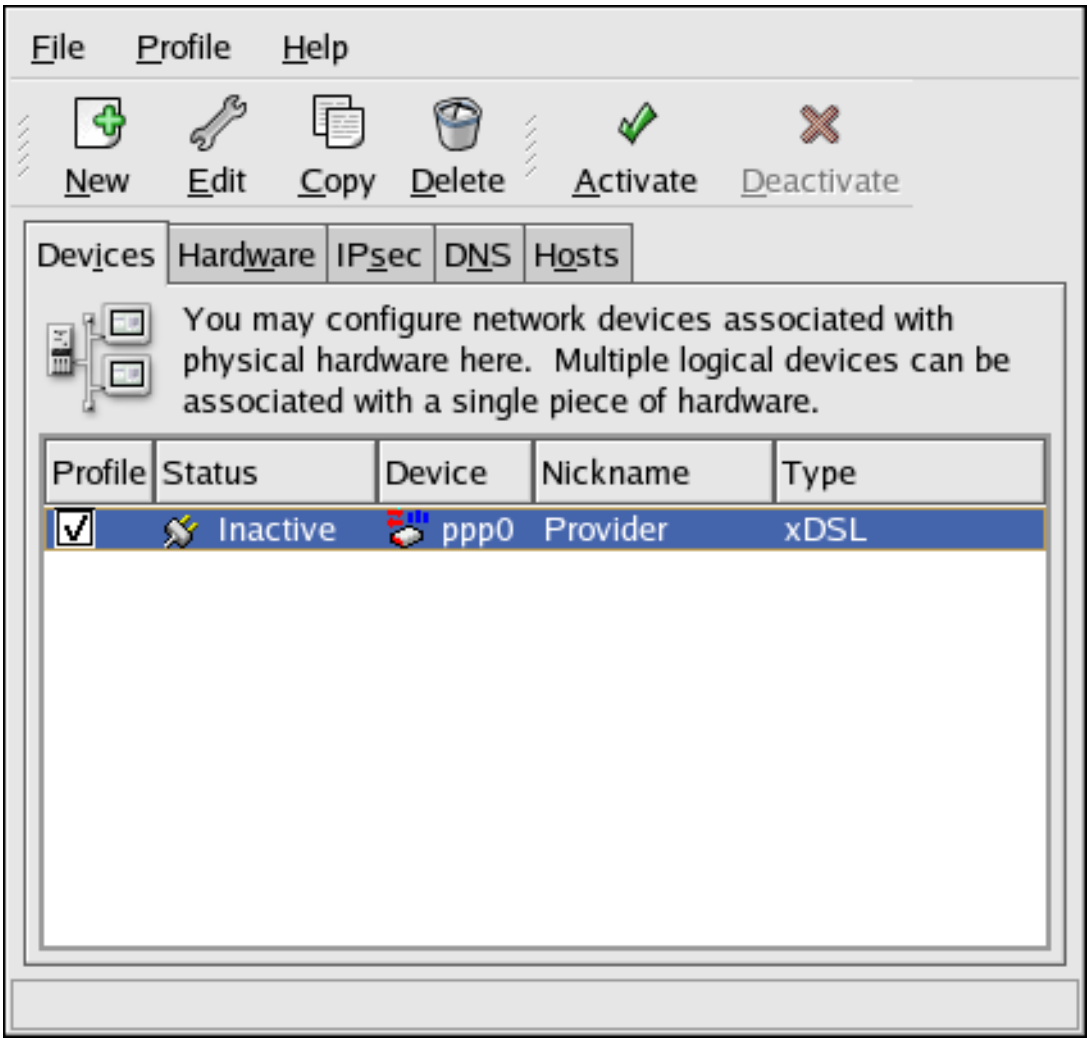
6.

Forward をクリックして Create DSL Connection メニューに移動します。設定を確認し、Apply をクリックして終了します。

7.

DSL 接続の設定後、図17.10「xDSL デバイス」 に示されるように、デバイス一覧に表示されます。

図17.10 xDSL デバイス

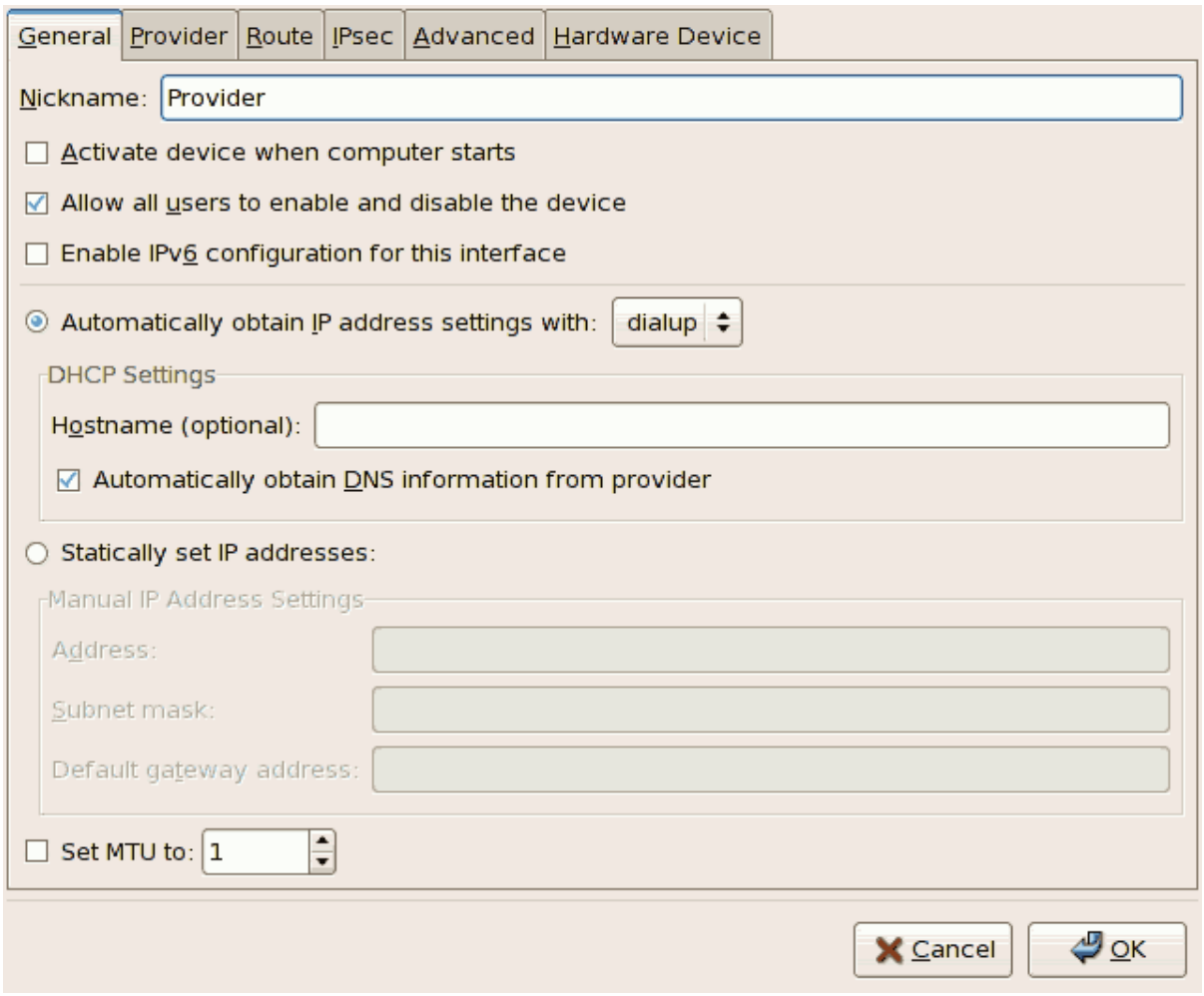


[D]

8.

xDSL 接続を追加した後、デバイス一覧からデバイスを選択し、Edit をクリックして設定を編集できます。

図17.11 xDSL 設定



[D]

たとえば、デバイスが追加されると、デフォルトではブート時に起動しないように設定されています。設定を編集してこの設定を変更します。終了したら OK をクリックします。

9. *xDSL 接続の設定が適切になったら、ファイル > 保存 を選択 し て変更を保存します。*

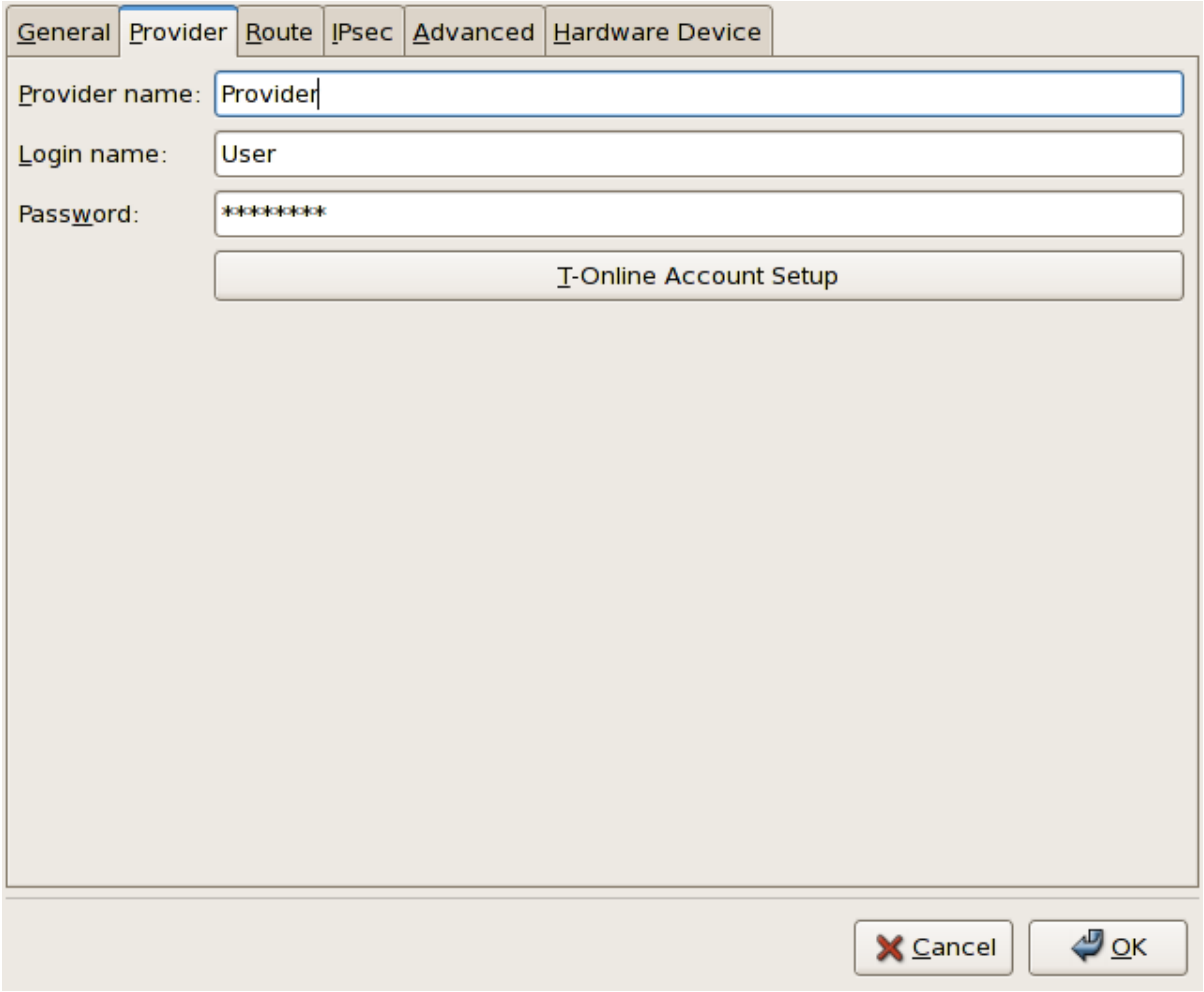
T-On オンラインアカウントの設定

T-On オンラインアカウントを設定する場合は、以下の追加の手順に従います。

1. *デバイス一覧からデバイスを選択し、編集 をクリックします。*

2. *図17.12「xDSL 設定：プロバイダータブ」 に示すように、xDSL 設定 メニューから Provider タブを選択します。*

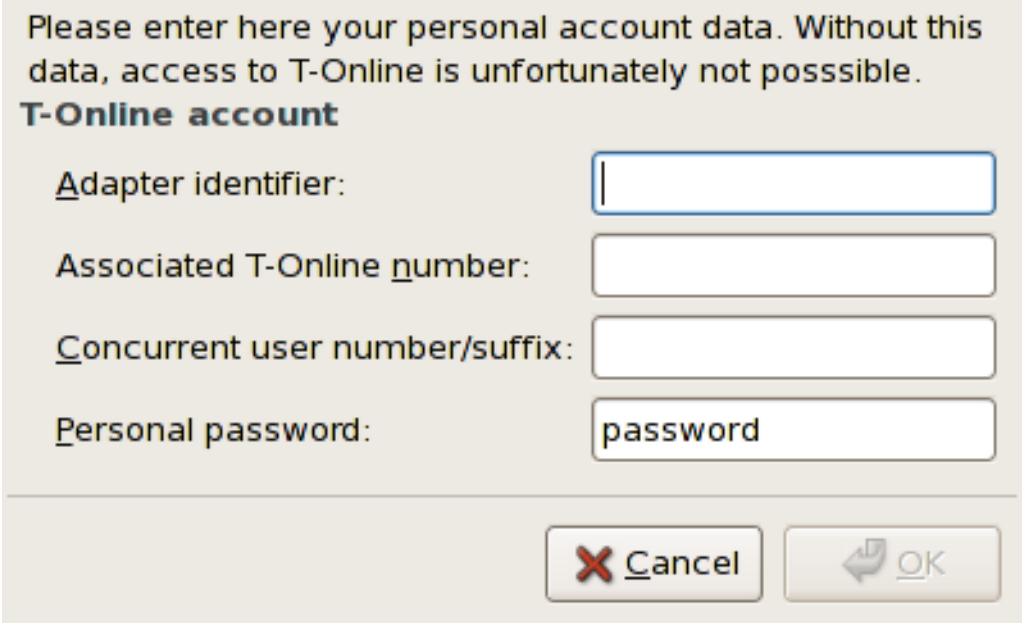
図17.12 xDSL 設定：プロバイダータブ



[D]

3. *T-Online Account Setup ボタンをクリックします。図17.13「アカウントの設定」 に示すように、T-Online アカウントの Account Setup ウィンドウが開きます。*

図17.13 アカウントの設定



[D]

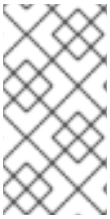
4. *Adapter ID、AssoatedT-Online number、Concurrent user number/suffix、および Personal パスワードを入力します。終了したら OK をクリックして、Account Setup ウィンドウを閉じます。*

5. *xDSL 設定 ウィンドウで、OK をクリックします。Network Administration Tool から File > Save を選択して変更を保存します。*

デバイスが追加されると、Inactive ステータスにあるように、すぐにアクティブ化されません。デバイスを有効にするには、デバイスリストから選択し、Activate ボタンをクリックします。コンピューターの起動時にデバイスをアクティベートするようにシステムが設定されている場合（デフォルト）、この手順を再度実行する必要はありません。

17.6. トークンリング接続の確立

トークンリングネットワーク は、すべてのコンピューターが循環パターンで接続されているネットワークです。トークン、または特別なネットワークパケットは、トークンリングの周りに移動し、コンピューターが互いに情報を送信できるようにします。



ヒント

Linux でトークンリングを使用する方法は、http://www.linuxtr.net/ から 『Linux Token Ring Project』 の Web サイトを参照してください。

トークンリング接続を追加するには、以下の手順に従います。

1. Devices タブをクリックします。

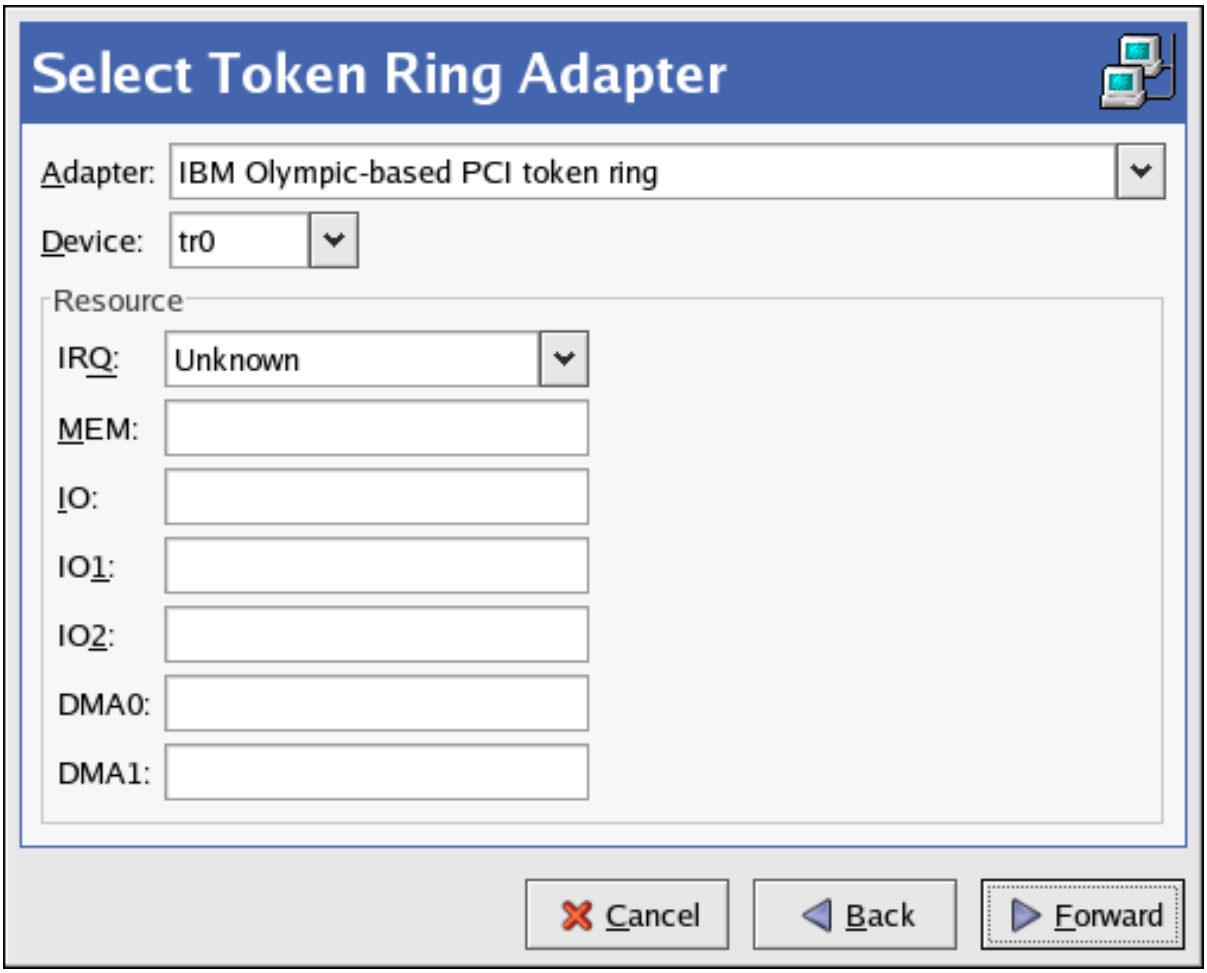
2. ツールバーの New ボタンをクリックします。

3. Device Type 一覧から Token Ring connection を選択し、Forward をクリックします。

4. トークンリングカードをハードウェア一覧に追加した場合は、Tokenring カード リストから選択します。それ以外の場合は、Other Tokenring Card を選択してハードウェアデバイスを追加します。

5. Other Tokenring Card を選択した場合は、図17.14「トークンリング設定」 に示されるように、Select Token Ring Adapter ウィンドウが表示されます。アダプターの製造元およびモデルを選択します。デバイス名を選択します。これがシステムの最初のトークンリングカードの場合は、tr0 を選択します。これが 2 番目のトークンリングカードの場合は、tr1 （等）を選択します。Network Administration Tool を使用すると、ユーザーはアダプターのリソースを設定することもできます。進む をクリックして続けます。

図17.14 トークンリング設定



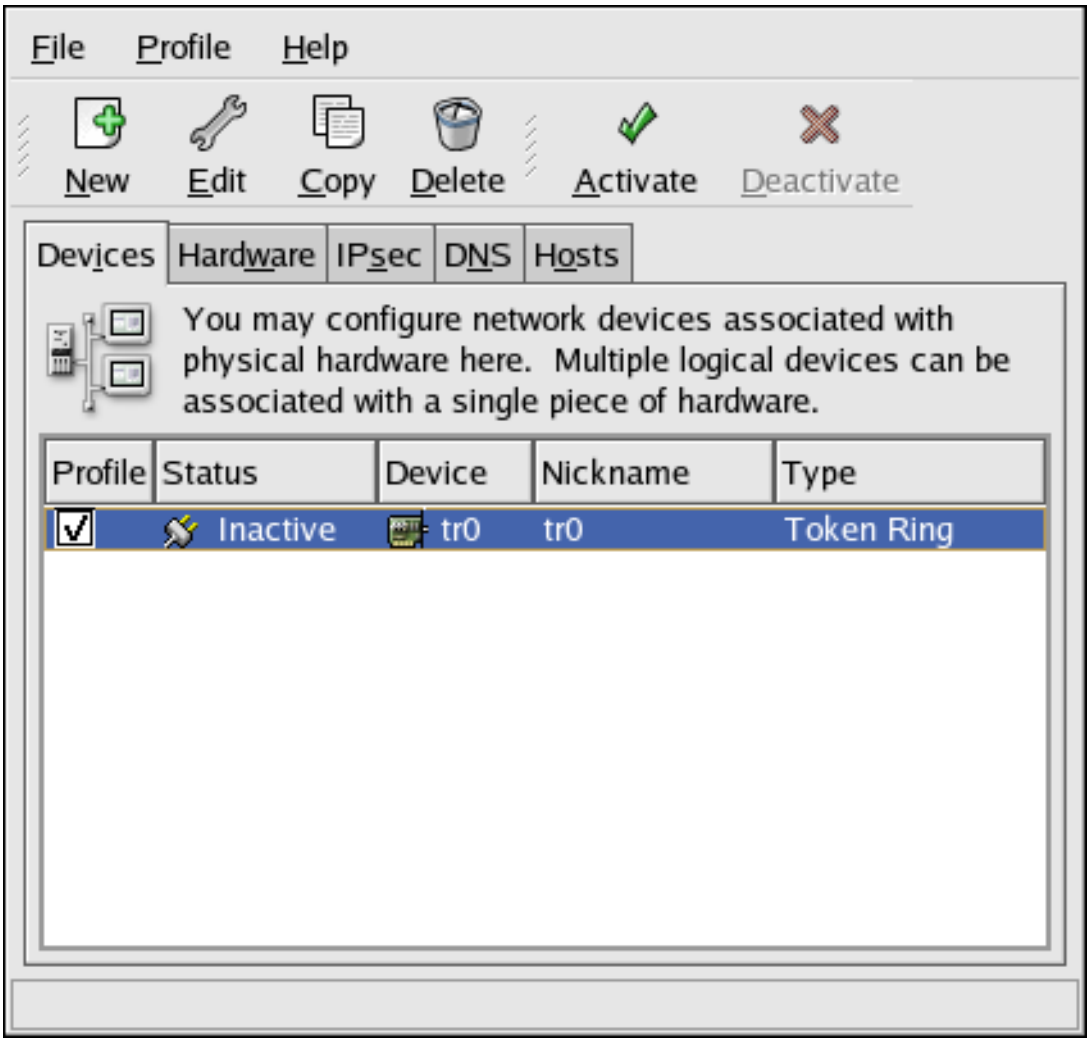
[D]

6. *Configure Network Settings ページで、DHCP と静的 IP アドレスのいずれかを選択します。デバイスのホスト名を指定できます。ネットワークが起動されるたびにデバイスが動的 IP アドレスを受信する場合は、ホスト名を指定しないでください。進む をクリックして続けます。*

7. *Create Tokenring Device ページで Apply をクリックします。*

トークンリングデバイスの設定後に、図17.15「トークンリングデバイス」 に示されるように、デバイス一覧に表示されます。

図17.15 トークンリングデバイス



[D]

File > Save を選択して変更を保存します。

デバイスの追加後、デバイス一覧からデバイスを選択し、Edit をクリックして設定を編集できます。たとえば、システムの起動時にデバイスを起動するかどうかを設定できます。

デバイスが追加されると、Inactive ステータスにあるように、すぐにアクティブ化されません。デバイスを有効にするには、デバイスリストから選択し、Activate ボタンをクリックします。コンピューターの起動時にデバイスをアクティベートするようにシステムが設定されている場合（デフォルト）、この手順を再度実行する必要はありません。

17.7. ワイヤレス接続の確立

ワイヤレスイーサネットデバイスはますます人気が高まっています。この設定はイーサネット設定と似ていますが、ワイヤレスデバイスの SSID やキーなどの設定を行うことができる点が異なります。

ワイヤレスイーサネット接続を追加するには、以下の手順に従います。

1. Devices タブをクリックします。

2. ツールバーの New ボタンをクリックします。

3. Device Type 一覧から Wireless connection を選択し、Forward をクリックします。

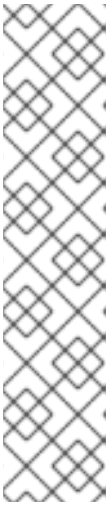
4. ワイヤレスネットワークインターフェイスカードをハードウェア一覧に追加した場合は、ワイヤレスカード リストから選択します。それ以外の場合は、Other Wireless Card を選択してハードウェアデバイスを追加します。

注記

インストールプログラムは通常、対応しているワイヤレスイーサネットデバイスを検出し、設定を求めるプロンプトを表示します。インストール時に設定すると、ハードウェア タブのハードウェア一覧に表示されます。

5. Other Wireless Card を選択した場合は、Select Ethernet Adapter ウィンドウが表示されます。イーサネットカードとデバイスの製造元とモデルを選択します。システムの最初のイーサネットカードの場合は、eth0 を選択します。これがシステムの 2 番目のイーサネットカードである場合は、eth1 （など）を選択します。 Network Administration Tool を使用すると、ユーザーはワイヤレスネットワークインターフェイスカードのリソースを設定することもできます。進む をクリックして続けます。

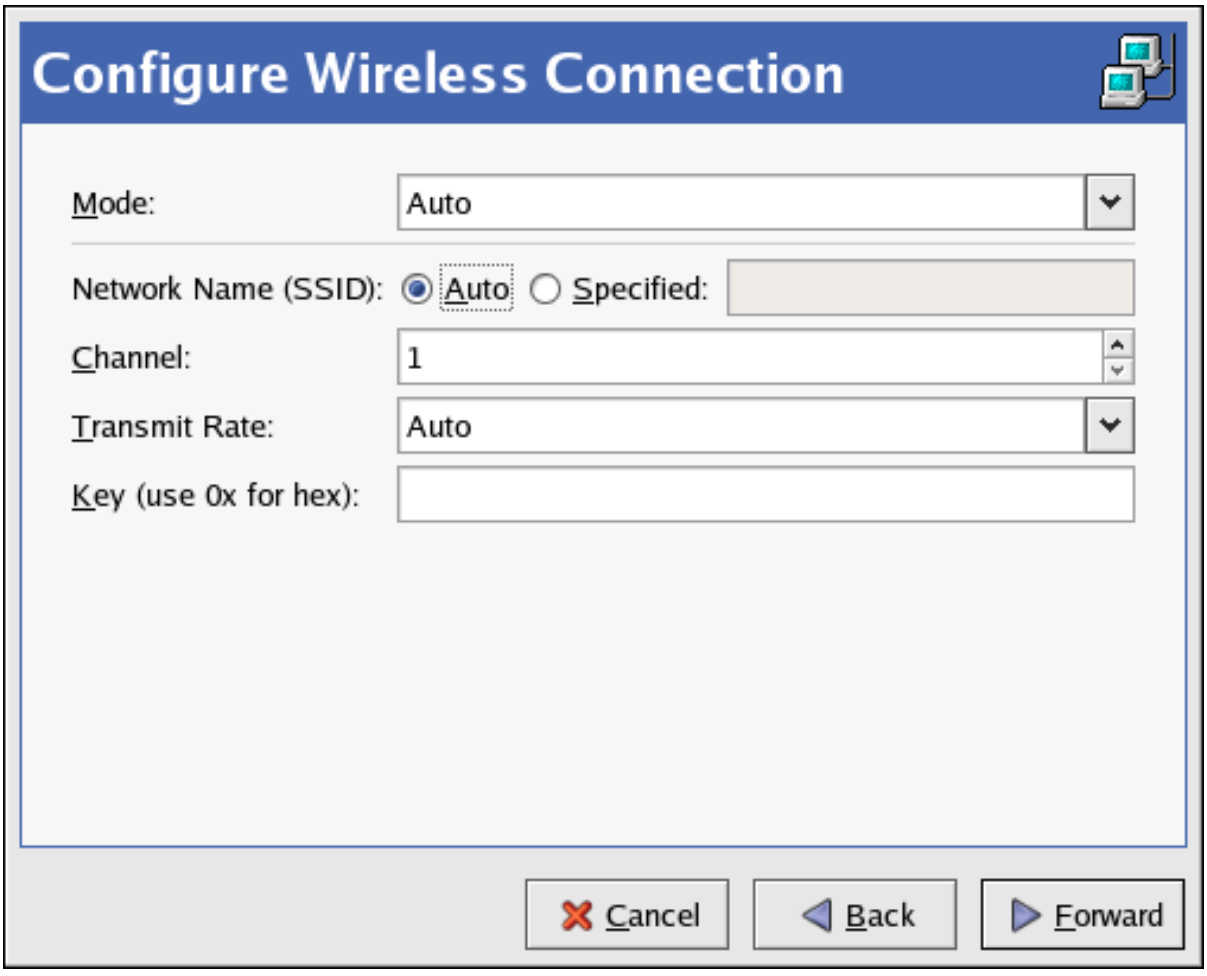
6. 図17.16「ワイヤレス設定」 に示されるように Configure Wireless Connection ページで、ワイヤレスデバイスの設定を設定します。



注記：OPEN SYSTEM AND SHARED KEY AUTHENTICATION

Authentication ドロップダウンでは、WEP 暗号化を使用したワイヤレスアクセスポイントは、オープンシステムと共有鍵認証の使用を選択することに注意してください。共有鍵認証では、関連付けプロセスでクライアントとアクセスポイント間の交換が必要です。これは、クライアントに正しい WEP キーがあることを証明します。システム認証を開くと、すべてのワイヤレスクライアントが接続できるようになります。直感的に、共有鍵認証はオープンシステムよりも安全ではないため、広く展開されません。そのため、アクセスポイントが必要とする方法が分からない場合は、認証方法として Open System (open) を選択することが推奨されます。open system を使用してアクセスポイントへの接続に失敗した場合は、共有鍵認証への切り替えを試みてください。

図17.16 ワイヤレス設定



[D]

7.

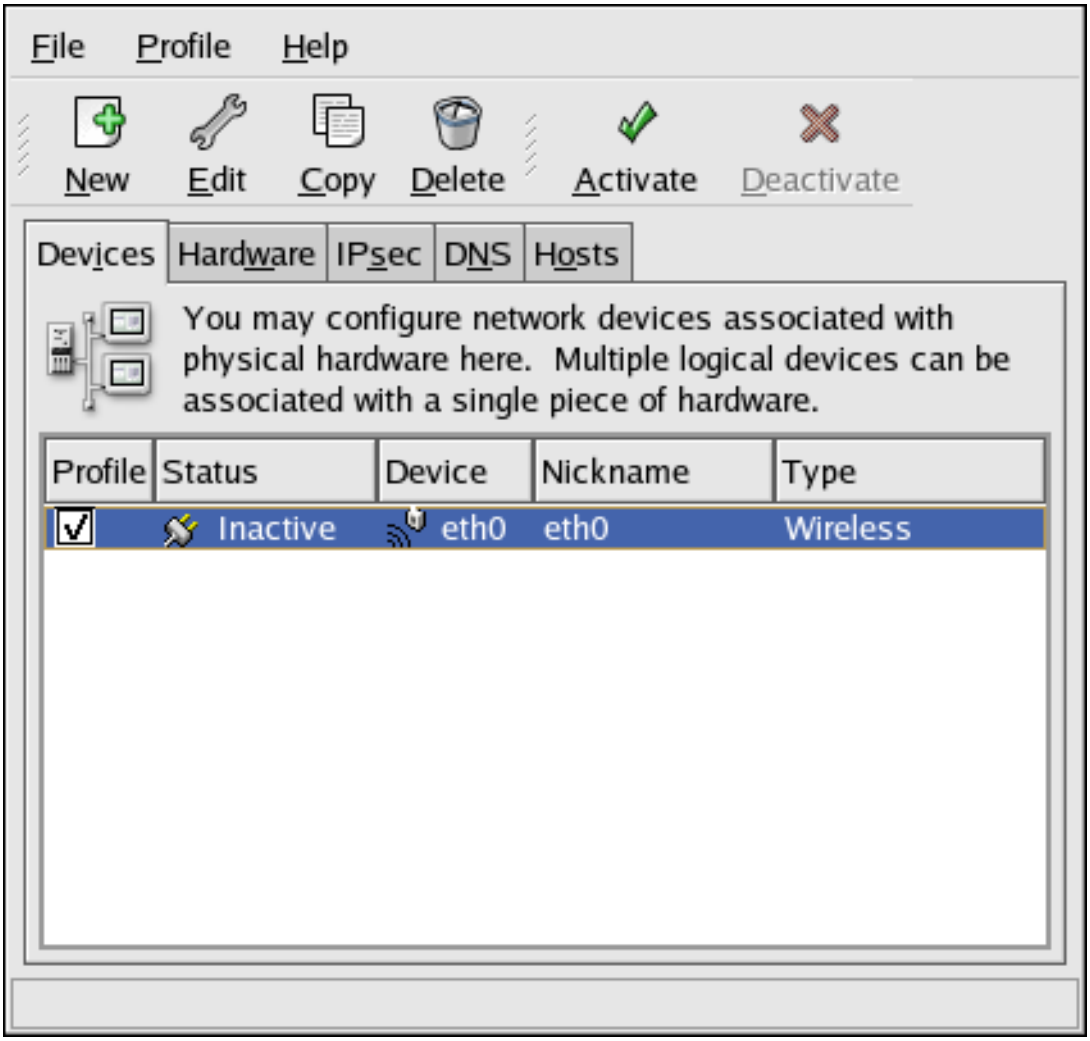
Configure Network Settings ページで、DHCP と静的 IP アドレスのいずれかを選択します。デバイスのホスト名を指定できます。ネットワークが起動されるたびにデバイスが動的 IP アドレスを受信する場合は、ホスト名を指定しないでください。進む をクリックして続けます。

8.

Create Wireless Device ページで *Apply* をクリックします。

ワイヤレスデバイスを設定すると、図17.17「ワイヤレスデバイス」に示されるように、デバイス一覧に表示されます。

図17.17 ワイヤレスデバイス



[D]

File > Save を選択して変更を保存します。

ワイヤレスデバイスを追加したら、デバイス一覧からデバイスを選択し、*Edit* をクリックして設定を編集できます。たとえば、システムの起動時にアクティブにするようにデバイスを設定できます。

デバイスが追加されると、*Inactive* ステータスにあるように、すぐにアクティブ化されません。デバイスを有効にするには、デバイスリストから選択し、*Activate* ボタンをクリックします。コンピュー

ターの起動時にデバイスをアクティベートするようにシステムが設定されている場合（デフォルト）、この手順を再度実行する必要はありません。

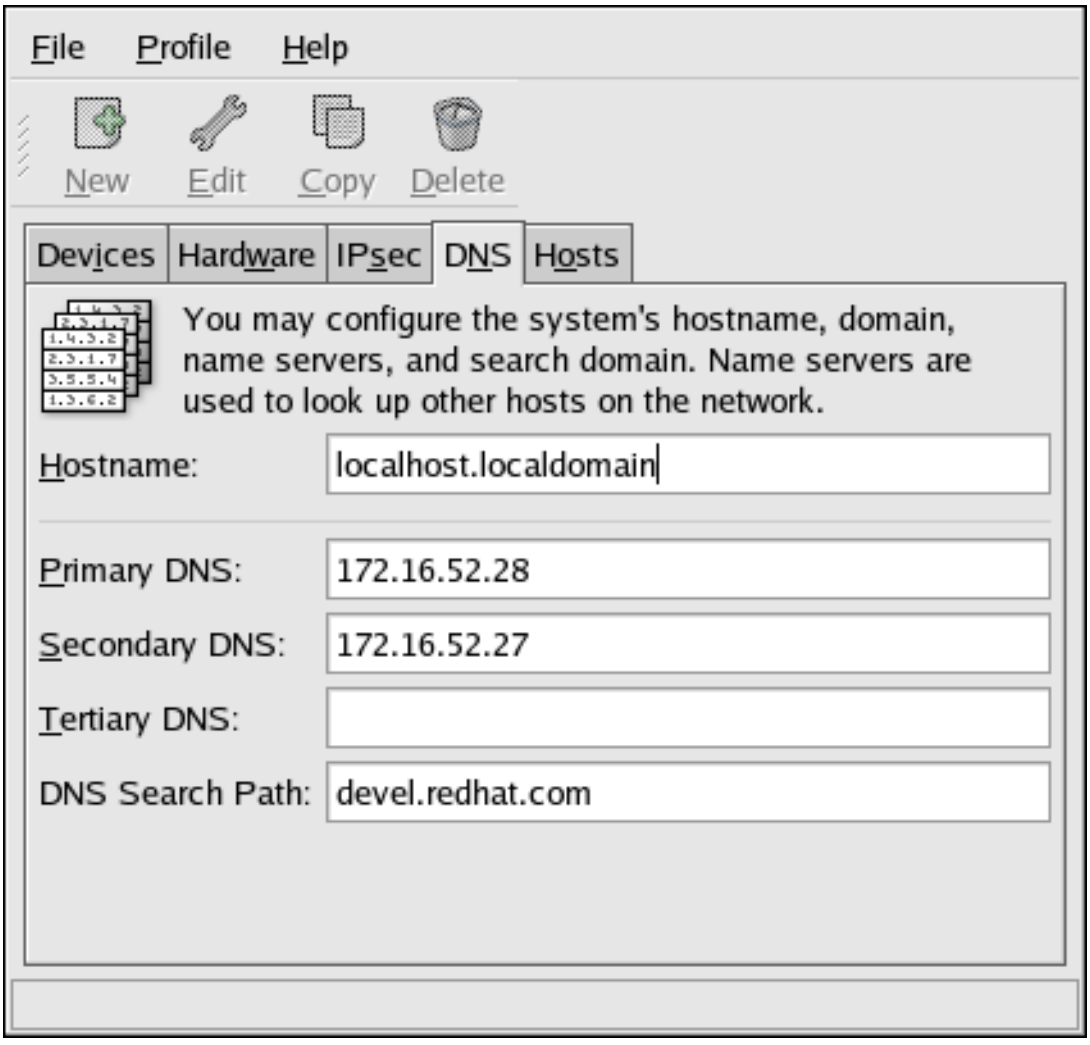
17.8. DNS 設定の管理

DNS タブでは、システムのホスト名、ドメイン、ネームサーバー、および検索ドメインを設定できます。ネームサーバーは、ネットワーク上の他のホストを検索するために使用されます。

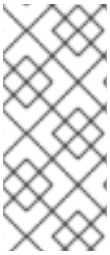
DNS サーバー名が DHCP または PPPoE （または ISP から取得）から取得されている場合は、プライマリー、セカンダリー、または 3 番目の DNS サーバーを追加しないでください。

ホスト名が DHCP または PPPoE （または ISP から取得）から動的に取得されている場合は、変更しないでください。

図17.18 DNS 設定



[D]



注記

ネームサーバーのセクションでは、システムがネームサーバーとして設定されません。代わりに、IP アドレスをホスト名に解決する際に使用するネームサーバーを設定します。

WARNING

ホスト名が変更され、system-config-network がローカルホストで起動している場合は、別の X11 アプリケーションを起動できない場合があります。そのため、新しいデスクトップセッションに再度ログインする必要がある場合があります。

17.9. ホストの管理

Hosts タブでは、/etc/hosts ファイルからホストを追加、編集、または削除できます。このファイルには、IP アドレスと対応するホスト名が含まれます。

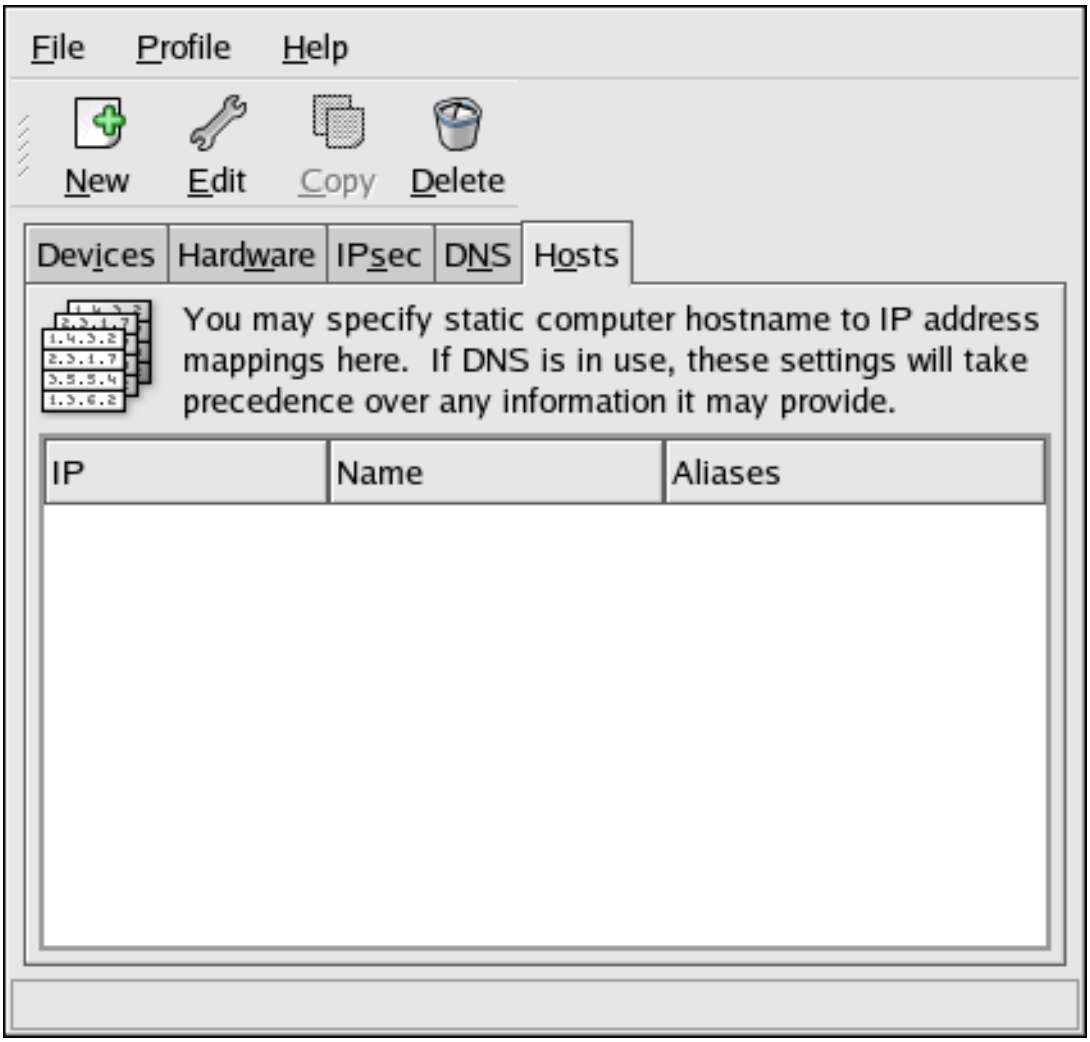
システムが IP アドレスに対してホスト名を解決しようとする場合や、IP アドレスのホスト名を決定しようとすると、ネームサーバーを使用する前に /etc/hosts ファイルを参照します（デフォルトの Red Hat Enterprise Linux 設定を使用している場合）。IP アドレスが /etc/hosts ファイルにリストされている場合は、ネームサーバーは使用されません。ネットワークに IP アドレスが DNS に記載されていないコンピューターが含まれている場合は、/etc/hosts ファイルに追加することが推奨されます。

/etc/hosts ファイルにエントリーを追加するには、ホスト タブに移動し、ツールバーの New ボタンをクリックして、要求された情報を入力して OK をクリックします。File > Save を選択するか、Ctrl+S を押して変更を /etc/hosts ファイルに保存します。現行バージョンのファイルはアドレスが解決されるたびに参照されるため、ネットワークサービスを再起動する必要はありません。

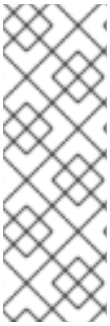
WARNING

localhost エントリーを削除しないでください。システムにネットワーク接続がない場合や、ネットワーク接続が継続的に実行されている場合でも、一部のプログラムは localhost ループバックインターフェイスを介してシステムに接続する必要があります。

図17.19 ホストの設定



[D]



ヒント

ルックアップの順序を変更するには、/etc/host.conf ファイルを編集します。行の 順序ホストである bind は、 ネームサーバーよりも /etc/hosts の優先度を指定します。行を 順序バインドに変更すると、ホスト はまずネームサーバーを使用してホスト名と IP アドレスを解決するよう設定します。ネームサーバーで IP アドレスを解決できない場合、システムは /etc/hosts ファイルで IP アドレスを検索します。

17.10. プロファイルの使用

各物理ハードウェアデバイスに複数の論理ネットワークデバイスを作成できます。たとえば、システム(eth0)にイーサネットカードが 1 つある場合は、異なるニックネームと設定オプションで論理ネットワークデバイスを作成できます。これは、すべて eth0 に関連付けられます。

論理ネットワークデバイスは、デバイスのエイリアスとは異なります。同じ物理デバイスに関連付けられた論理ネットワークデバイスは、異なるプロファイルに存在する必要があり、同時にアクティブにすることはできません。デバイスエイリアスは同じ物理ハードウェアデバイスに関連付けられていま

すが、同じ物理ハードウェアに関連付けられているデバイスのエイリアスを同時にアクティブにすることができます。デバイスエイリアスの作成に関する詳細は、「デバイスエイリアス」 を参照してください。

プロファイル を使用して、異なるネットワークに複数の設定セットを作成できます。設定セットには、ホストおよび DNS 設定に加えて論理デバイスを含めることができます。プロファイルの設定後に、Network Administration Tool を使用してそれらの間で送受信を切り替えることができます。

デフォルトでは、Common と呼ばれるプロファイルが 1 つあります。新規プロファイルを作成するには、プルダウンメニューから Profile > New を選択して、プロファイルに一意の名前を入力します。

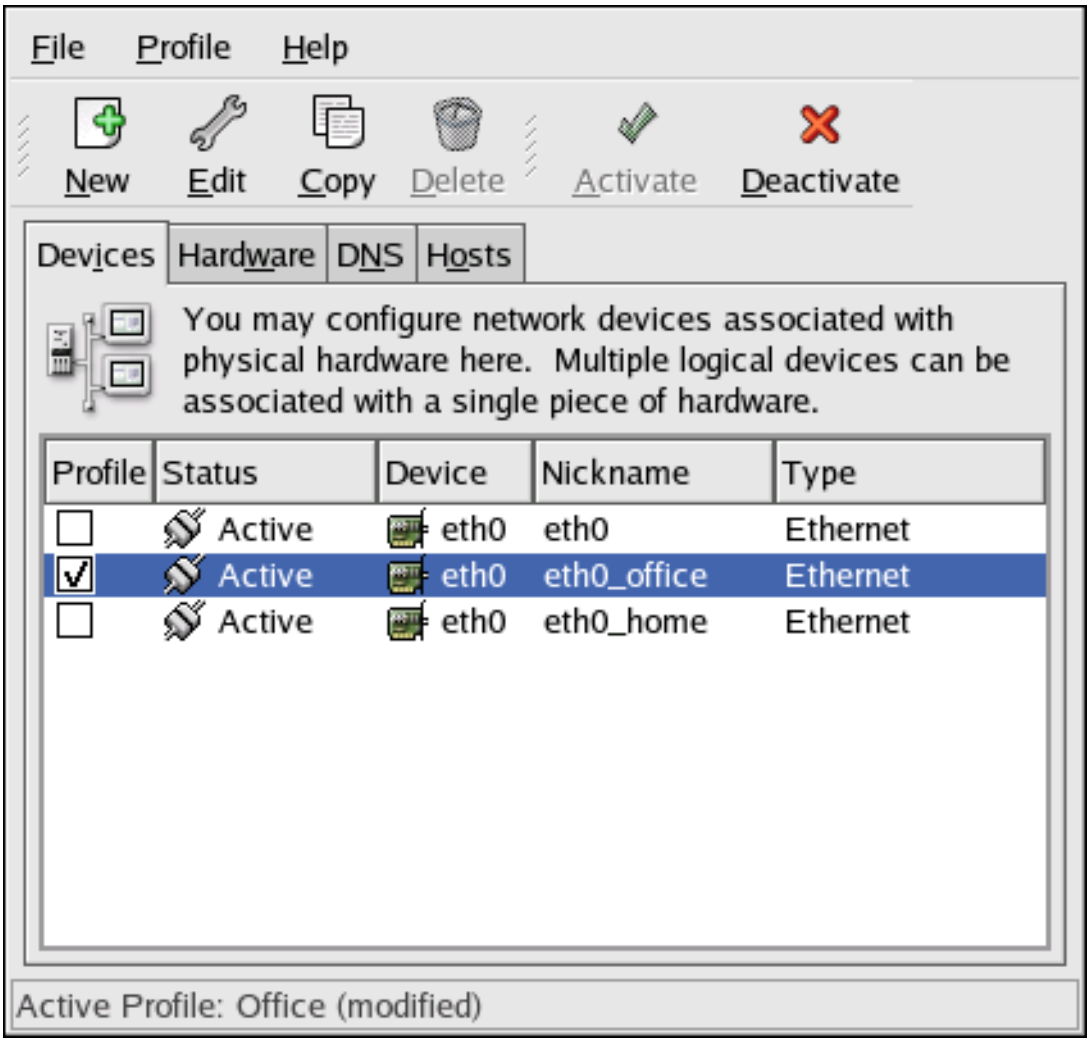
これで、メインウィンドウの下部のステータスバーで示されているように、新しいプロファイルを変更します。

一覧にすでにある既存のデバイスをクリックし、Copy ボタンをクリックして、既存のデバイスを論理ネットワークデバイスにコピーします。New ボタンを使用すると、ネットワークエイリアスが作成されますが、これは正しくありません。論理デバイスのプロパティーを変更するには、一覧から選択し、Edit をクリックします。たとえば、Nickname は、eth0_office など、より簡単に認識できるように、より分かりやすい名前に変更できます。新しいプロファイルの編集が終了したら、File メニューから Save をクリックして保存します。プロファイルの作成後に保存を忘れると、そのプロファイルが失われます。

デバイスの一覧には、Profile というラベルが付いたチェックボックスの列があります。各プロファイルで、デバイスのチェックまたは選択解除を行うことができます。現在選択されているプロファイルにチェックされているデバイスのみが含まれます。たとえば、Office というプロファイルに eth0_office という名前の論理デバイスを作成し、プロファイルが選択された場合に論理デバイスをアクティベートする場合は、eth0 デバイスの選択を解除して、eth0_office デバイスを確認します。

たとえば、図17.20「オフィスプロファイル」 は、論理デバイス eth0_office を持つ Office という名前のプロファイルを示しています。DHCP を使用して最初のイーサネットカードをアクティベートするように設定されます。

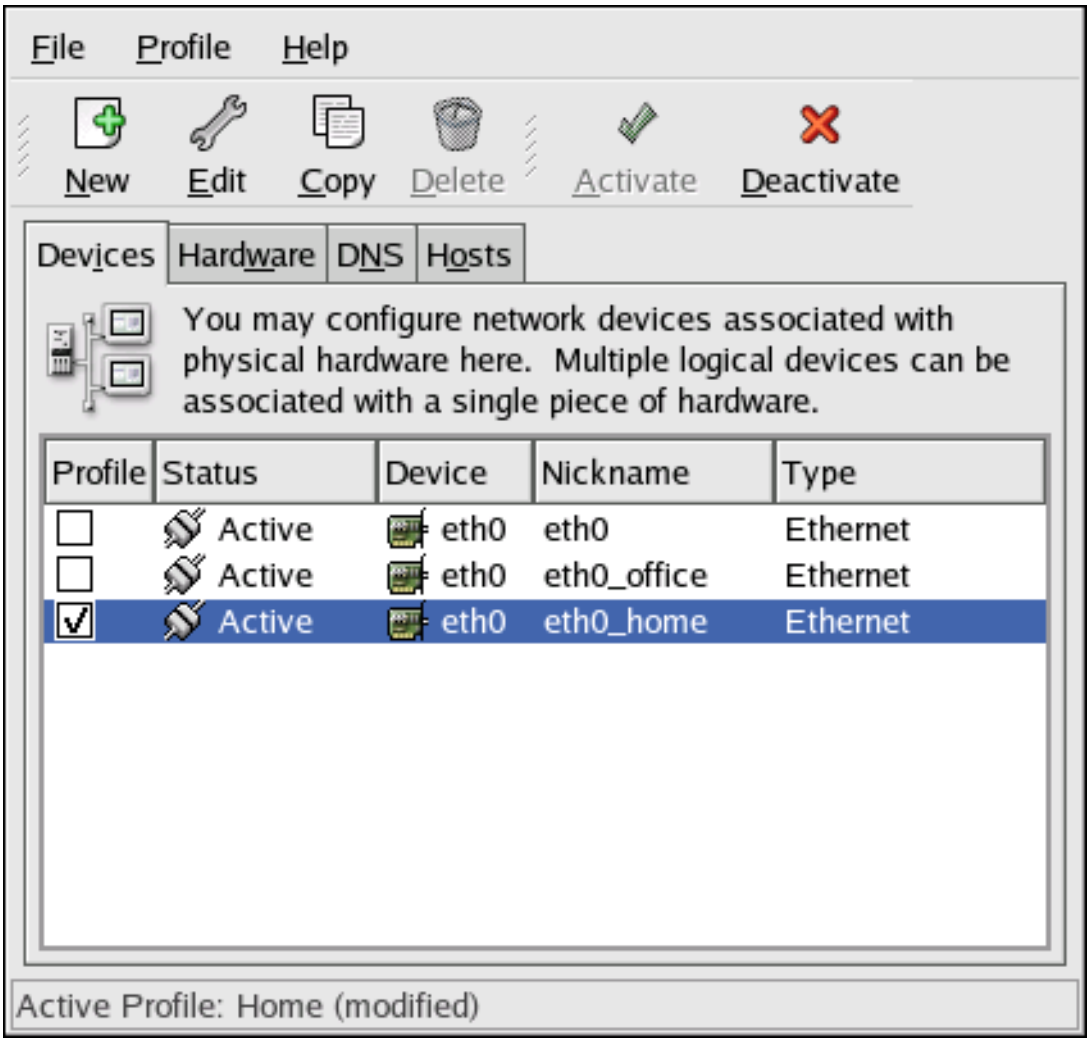
図17.20 オフィスプロファイル



[D]

図17.21「ホームプロファイル」 に示されるように Home プロファイルが eth0_home 論理デバイスをアクティベートし、eth0 に関連付けられていることに注意してください。

図17.21 ホームプロファイル



[D]

また、Office プロファイルでのみアクティベートするように eth0 を設定し、Home プロファイルでのみ PPP (modem)デバイスをアクティブにすることもできます。もう 1 つの例は、Common プロファイルが eth0 をアクティブにし、移動中に PPP デバイスをアクティベートすることです。

起動時にプロファイルを有効にするには、ブートローダー設定ファイルを変更して netprofile=<profilename> オプションを 含めます。たとえば、システムがブートローダーとして GRUB を使用し、/boot/grub/grub.conf には以下が含まれます。

```
title Red Hat Enterprise Linux (2.6.9-5.EL)
        root (hd0,0)
  kernel /vmlinuz-2.6.9-5.EL ro root=/dev/VolGroup00/LogVol00 rhgb quiet
  initrd /initrd-2.6.9-5.EL.img
```

これを次のように変更します(ここで、<profile name> は起動時にアクティベートするプロファイルの名前です)。

```
title Red Hat Enterprise Linux (2.6.9-5.EL)
        root (hd0,0)
  kernel /vmlinuz-2.6.9-5.EL ro root=/dev/VolGroup00/LogVol00 \
   netprofile=<profilename>  \    rhgb quiet
  initrd /initrd-2.6.9-5.EL.img
```

システムの起動後にプロファイルを切り替えるには、Applications （パネルのメインメニュー）> System Tools > Network Device Control （または command system-control-network）に移動してプロファイルを選択し、アクティベートします。activate profile セクションは、デフォルトの Common Interface が複数存在する場合に Network Device Control インターフェイスにのみ表示されます。

または、以下のコマンドを実行してプロファイルを有効にします(& lt;profilename> をプロファイルの名前に置き換えます)。

```
system-config-network-cmd --profile <profilename> --activate
```

17.11. デバイスエイリアス

デバイスエイリアス は、同じ物理ハードウェアに関連付けられた仮想デバイスですが、異なる IP アドレスを持つために同時にアクティブにすることができます。これらは一般的にデバイス名で表現され、その後にコロンと番号(eth0:1 など)が続きます。これらは、1 つのシステムに複数の IP アドレスがあり、1 つのネットワークカードしか持たない場合に便利です。

eth0 などのイーサネットデバイスを設定したら、静的 IP アドレス(DHCP がエイリアスでは機能しません)を使用するよう、Devices タブに移動し、New をクリックします。エイリアスで設定する Ethernet カードを選択し、エイリアスの静的 IP アドレスを設定し、Apply をクリックして作成します。イーサネットカードにデバイスがすでに存在しているため、作成したデバイスは eth0:1 などのエイリアスになります。

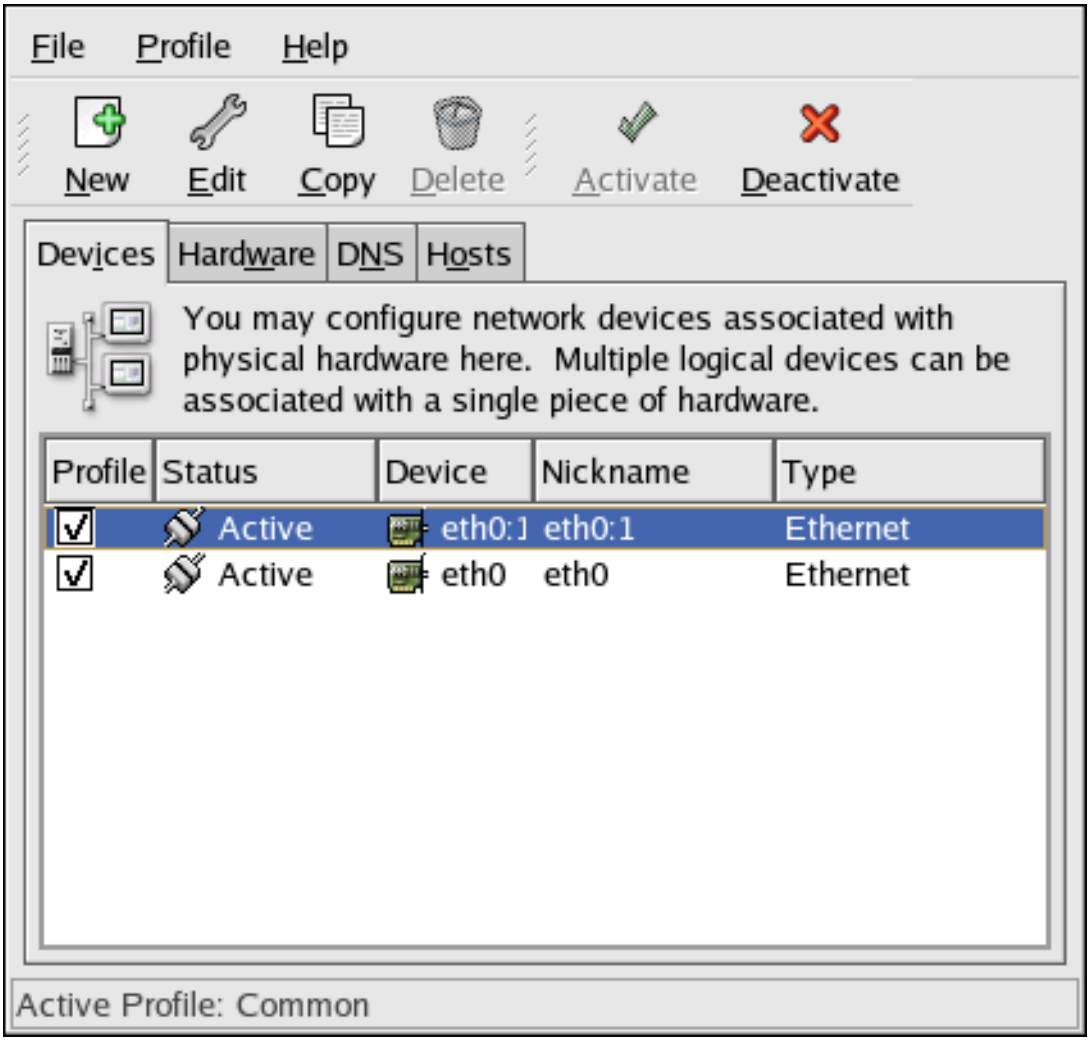
WARNING

イーサネットデバイスにエイリアスを設定する場合は、デバイスもエイリアスを持つように設定することもできません。IP アドレスを手動で設定する必要があります。

図17.22「ネットワークデバイスエイリアスの例」 は、eth0 デバイスに対する 1 つのエイリアスの例を示しています。eth0 :1 デバイス(eth 0 の最初のエイリアス)に注意してください。eth0 の 2 番目のエイリアスは、デバイス名 eth0:2 などになります。起動時にデバイスエイリアスをアクティブにす

るかどうかやエイリアス番号の設定を変更する場合は、リストから選択し、編集 ボタンをクリックします。

図17.22 ネットワークデバイスエイリアスの例



[D]

```
eth0      Link encap:Ethernet
 HWaddr 00:A0:CC:60:B7:G4
 inet addr:192.168.100.5  Bcast:192.168.100.255  Mask:255.255.255.0
 UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
 RX packets:161930 errors:1 dropped:0 overruns:0 frame:0
 TX packets:244570 errors:0 dropped:0 overruns:0 carrier:0
 collisions:475 txqueuelen:100
 RX bytes:55075551 (52.5 Mb)  TX bytes:178108895 (169.8 Mb)
 Interrupt:10 Base address:0x9000  eth0:1    Link encap:Ethernet  HWaddr 00:A0:CC:60:B7:G4
 inet addr:192.168.100.42  Bcast:192.168.100.255  Mask:255.255.255.0
 UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

```
Interrupt:10 Base address:0x9000  lo
Link encap:Local Loopback
inet addr:127.0.0.1  Mask:255.0.0.0
UP LOOPBACK RUNNING  MTU:16436  Metric:1
RX packets:5998 errors:0 dropped:0 overruns:0 frame:0
TX packets:5998 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:1627579 (1.5 Mb)  TX bytes:1627579 (1.5 Mb)
```

17.12.

```
system-config-network-cmd -e > /tmp/network-config
```

```
system-config-network-cmd -i -c -f /tmp/network-config
```

第18章

ntsysv

chkconfig

重要な影響

18.1.

-
-
-
-
-
-
-

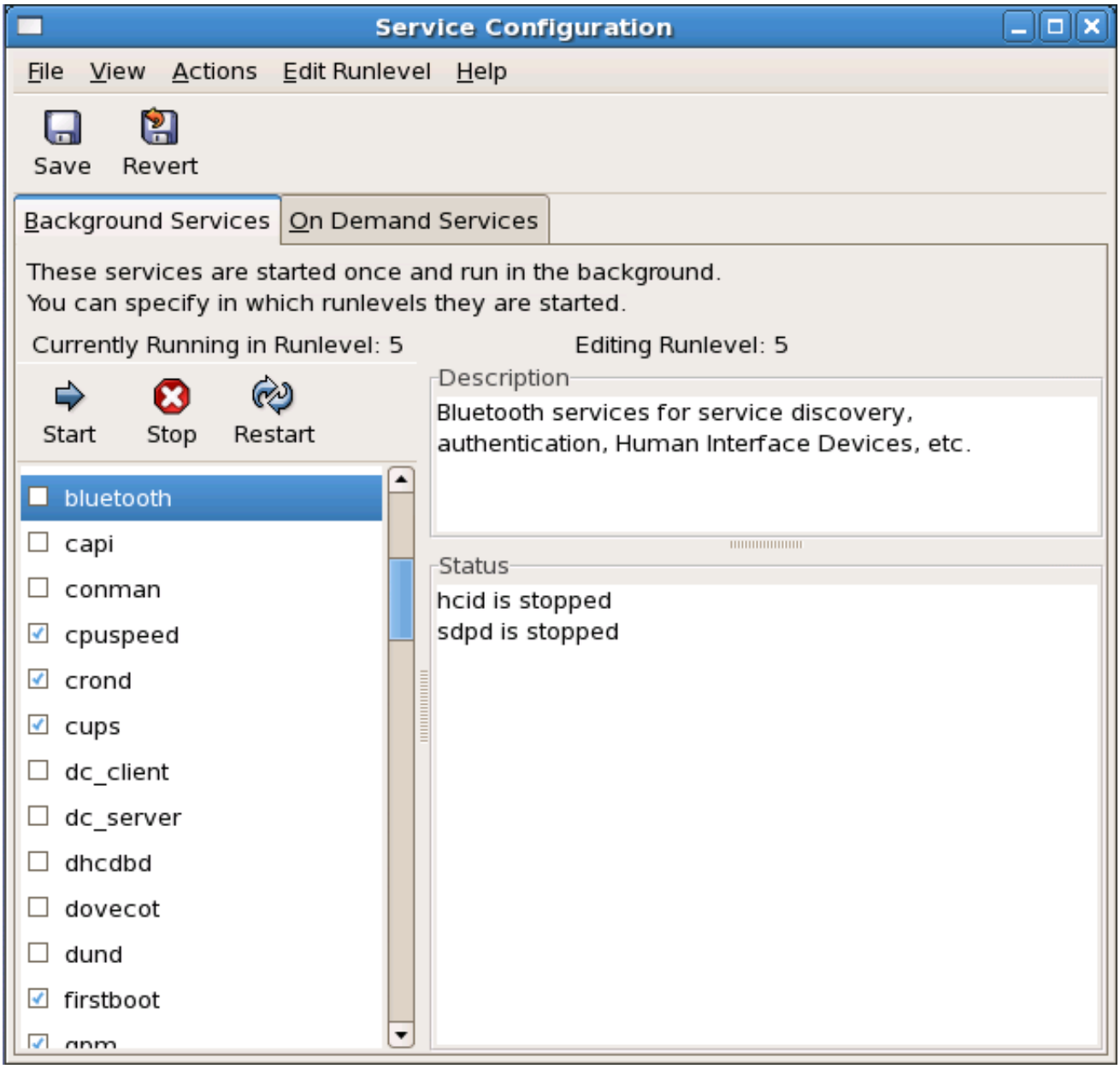
id:5:initdefault:

18.2.

18.2.1. xinetd

18.3.

図18.1


Service Configuration
File View Actions Edit Runlevel Help
Save
Revert
Background Services
On Demand Services
These services are started once and run in the background.
You can specify in which runlevels they are started.
Currently Running in Runlevel: 5
Editing Runlevel: 5
Start
Stop
Restart
Description
Bluetooth services for service discovery, authentication, Human Interface Devices, etc.
Status
hcid is stopped
sdpd is stopped
bluetooth
capi
conman
cpuspeed
crond
cups
dc_client
dc_server
dhcdbd
dovecot
dund
firstboot

[D]

注記

1.

2.

3.

18.4. NTSYSV

図18.2

[D]

WARNING

18.5. CHKCONFIG

```
rsync           on
```

```
httpd           0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

```
chkconfig --level 345 nscd off
```

WARNING

18.6. 関連情報

18.6.1. インストールされているドキュメント

-
-

18.6.2. 便利な Web サイト

-

第19章

注記

19.1. DNS の概要

19.1.1.

> bob.sales.example.com

19.1.2.

master

slave

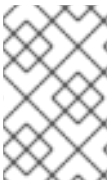
転送

19.1.3.

/etc/named.conf

/var/named/ directory

注記



ヒント

19.2. /ETC/NAMED.CONF

```
<statement-1> ["<statement-1-name>"] [<statement-1-class>] {
 <option-1>;
 <option-2>;
 <option-N>;
};
<statement-2> ["<statement-2-name>"] [<statement-2-class>] {
 <option-1>;
 <option-2>;
 <option-N>;
};
<statement-N> ["<statement-N-name>"] [<statement-N-class>] {
 <option-1>;
 <option-2>;
 <option-N>;
};
```

19.2.1. 一般的なステートメントのタイプ

/etc/named.conf では、通常、以下のタイプのステートメントが使用されます。

19.2.1.1.

```
acl <acl-name> {
 <match-element>;
 [<match-element>; ...]
};
```

-
-
-
-

```
acl black-hats {
 10.0.2.0/24;
 192.168.0.0/24;
};
acl red-hats {
 10.0.1.0/24;
};
options {
 blackhole { black-hats; };
```

```
 allow-query { red-hats; };
 allow-recursion { red-hats; };
};
```

19.2.1.2.

```
include "<file-name>"
```

19.2.1.3.

```
options {
 <option>;
 [<option>; ...]
};
```

allow-query

allow-recursion

blackhole

directory

forwarders

forward

-
-

listen-on

```
options {
 listen-on { 10.0.1.1; };
};
```

notify

以下のオプションを取ります。

-
-
-

pid-file

root-delegation-only

```
options {
 root-delegation-only exclude { "ad"; "ar"; "biz"; "cr"; "cu"; "de"; "dm"; "id";
  "lu"; "lv"; "md"; "ms"; "museum"; "name"; "no"; "pa";
  "pf"; "se"; "sr"; "to"; "tw"; "us"; "uy"; };
};
```

statistics-file

統計ファイルの代替の場所を指定します。

19.2.1.4.

```
zone <zone-name> <zone-class> {
 <zone-options>;
 [<zone-options>; ...]
};
```

注記

allow-query

allow-transfer

allow-update

file

masters

notify

-
-
-

type

-
-
-
-
-

zone-statistics

19.2.1.5.

```
zone "example.com" IN {
 type master;
 file "example.com.zone";
 allow-update { none; };
};
```

-

```
zone "example.com" {
 type slave;
 file "example.com.zone";
 masters { 192.168.0.1; };
};
```

19.2.2. その他のステートメントタイプ

controls

key "<key-name>"

キーは、安全な更新や、あるいは、rndc コマンドの使用など各種動作を認証するために使用されます。

-
-

logging

ロギングプロセスのカスタマイズは詳細なプロセスとなるため、本章の範囲外になります。

server

ステートメントを使うと、安全な DNS (DNSSEC) に使用される各種パブリックキーを指定できるようになります。

view "<view-name>"

これにより、他のホストが全く異なる情報を受け取る間、ゾーンに関する応答が 1 つの応答を受け取ることができます。また、信頼されないホスト以外のホストでは他のゾーンに対するクエリーしか実行できません。

19.2.3. コメントタグ

-
-

-

19.3.

注記

このため、ゾーンファイルは /var/named/chroot/var/named にあります。

各ゾーンファイルには、ディレクティブと リソースレコード を含めることができます。ディレクティブは、タスクを実行するか、特別な設定をゾーンに適用するようにネームサーバーに指示します。リソースレコードはゾーンのパラメーターを定義し、個々のホストに ID を割り当てます。ディレクティブはオプションですが、ゾーンにネームサービスを提供するためにリソースレコードが必要です。

ディレクティブとリソースレコードはすべて、個別の行で記入します。

コメントは、ゾーンファイルのセミコロン文字(;)の後に置くことができます。

19.3.1. ゾーンファイルのディレクティブ

ディレクティブはドル記号($)で始まり、その後にディレクティブの名前が続きます。これらは通常、ゾーンファイルの上部に表示されます。

一般的に使用されるディレクティブを以下に示します。

$INCLUDE

ディレクティブが表示される場所に、このゾーンファイルに別のゾーンファイルを含めるように named を設定します。これにより、メインのゾーンファイルとは別に、追加のゾーン設定を保存できます。

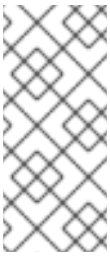
$ORIGIN

ホスト名などの修飾されていないレコードにドメイン名を追加します。

たとえば、ゾーンファイルには次の行を含めることができます。

```
$ORIGIN example.com.
```

末尾のピリオド(.)で終了しないリソースレコードで使用される名前には、example.com が追加されます。



注記

ゾーン名が $ORIGIN ディレクティブの値として使用されるため、/etc/named.conf にゾーンが指定されている場合は、$ORIGIN ディレクティブの使用は必要ありません。

$TTL

ゾーンのデフォルトの Time to Live (TTL) 値を設定します。これは、ゾーンリソースレコードが有効である期間（秒単位）です。各リソースレコードはそれ自身の TTL 値を含むことができるため、それがこのディレクティブを上書きします。

この値を増やすと、リモートのネームサーバーはより長い期間ゾーン情報をキャッシュし、ゾーンのクエリー数を減らし、リソースレコードの変更の延長に必要な時間を長くすることができます。

19.3.2. ゾーンファイルリソースレコード

ゾーンファイルのプライマリーコンポーネントは、リソースレコードです。

ゾーンファイルリソースレコードには多くのタイプがあります。以下は、最も頻繁に使用されます。

A

以下の例のように、名前に割り当てる IP アドレスを指定する Address レコードを参照します。

```
<host> IN A <IP-address>
```

&lt ;host> 値を省略すると、A レコードは namespace の上部にあるデフォルトの IP アドレスを参照します。このシステムは、FQDN 以外のすべての要求のターゲットです。

example.com ゾーンファイルの以下の A レコードの例を見てみましょう。

```
server1  IN A 10.0.1.3
 IN A 10.0.1.5
```

example.com の要求は 10.0.1.3 または 10.0.1.5 を参照します。

CNAME

これは、名前を別の名前にマップする正規名レコードを参照します。このタイプのレコードは、エイリアスレコード とも呼ばれます。

以下の例では、< alias-name> に送信された要求がホスト < real-name > を参照する必要があることを named に指示し ます。CNAME レコードは、Web サーバーの www など、共通の命名スキームを使用するサービスを参照するために最も一般的に使用されます。

```
<alias-name> IN CNAME <real-name>
```

以下の例では、A レコードはホスト名を IP アドレスにバインドし、CNAME レコードが一般的に使用される www ホスト名をこれにポイントします。

```
server1  IN A 10.0.1.5
www  IN CNAME server1
```

MX

これはメール eXchange レコードを指し、このゾーンによって制御される特定の名前空間に送信されたメールの移動先を指示します。

```
 IN MX <preference-value> <email-server-name>
```

ここで、& lt;preference-value > は namespace のメールサーバーの数値ランクを許可し、一部のメールシステムを他のシステムよりも優先されます。最も低い < preference-value> を 持つ

MX リソースレコードは、他よりも優先されます。しかし複数メールサーバーが同じ値を持つ可能性があり、その場合はメールトラフィックをサーバー間で均等に分配することになります。

&lt ;email-server-name> はホスト名または FQDN です。

```
IN    MX    10    mail.example.com.
IN    MX    20    mail2.example.com.
```

この例では、example.com ドメイン宛てのメールを受信する際に、最初の mail.example.com メールサーバーは mail2. example.com メールサーバーよりも優先されます。

NS

これは、特定ゾーンの権威ネームサーバーをアナウンスする NameServer レコードを参照します。

以下は、NS レコードのレイアウトを示しています。

```
IN NS <nameserver-name>
```

ここで、& lt;nameserver-name> は FQDN である必要があります。

次に、2 つのネームサーバーがドメインに対して権威として一覧表示されます。これらのネームサーバーがスレーブであるか、マスターであるかが重要ではなく、いずれも権威があると見なされます。

```
IN    NS    dns1.example.com.
IN    NS    dns2.example.com.
```

PTR

これは、名前空間の別の部分をポイントするように設計された PoinTeR レコードを参照します。

PTR レコードは、主に IP アドレスを特定の名前に戻すため、逆引き名前解決に使用されます。使用中の PTR レコードの他の例については、「逆引き名前解決ゾーンファイル」 を参照してください。

SOA

これは Start Of Authority リソースレコードを参照します。これは、名前空間に関する重要な信頼できる情報をネームサーバーに提案します。

SOA リソースレコードは、ディレクティブの後に配置され、ゾーンファイルの最初のリソースレコードです。

以下の SOA リソースレコードの基本構造を表示します。

```
@ IN SOA <primary-name-server> <hostmaster-email> (
 <serial-number>
 <time-to-refresh>
 <time-to-retry>
 <time-to-expire>
 <minimum-TTL> )
```

@ 記号は、$ORIGIN ディレクティブ（または $ORIGIN ディレクティブが設定されていない場合はゾーンの名前）をこの SOA リソースレコードで定義されている名前空間として配置します。このドメインに対して権威のあるプライマリーネームサーバーのホスト名は < primary-name-server > ディレクティブで、この namespace と通信するユーザーの電子メールは < hostmaster-email> ディレクティブ です。

&lt ;serial-number> ディレクティブは、named がゾーンを再ロードする時間であることを示すためにゾーンファイルが変更されるたびに増加する数値です。& lt;time-to-refresh > ディレクティブは、ゾーンに変更を加えたかどうかをマスターネームサーバーに尋ねるまでの待機時間を判断するために使用する数値スレーブサーバーです。&lt ;serial-number > ディレクティブは、古いゾーンデータを使用しているかどうかを判断するためにスレーブサーバーが使用する数値であるため、更新する必要があります。

& lt;time-to-retry > ディレクティブは、マスターネームサーバーが応答しない場合に更新要求を発行するまでの待機時間を判断するためにスレーブサーバーが使用する数値です。マスターが <time-to -expire> ディレクティブで指定された時間が経過する前に更新要求に 応答しなかった場合、スレーブサーバーはその名前空間に関する要求の権威として応答しなくなります。

BIND 4 および 8 では、& lt;minimum-TTL& gt; ディレクティブは amount of time のネームサーバーでゾーンの情報をキャッシュします。ただし、BIND 9 では、< minimum-TTL> ディレクティブ は、負の応答がキャッシュされる期間を定義します。負の回答のキャッシュは、最大 3 時間 (3H) に設定できます。

BIND の設定時には、すべての時間は秒で指定されます。ただし、秒以外の時間単位を指定する場合は省略形を使用できます。たとえば、分(M)、時間(H)、日(D)、および週(W)などです。表 19.1「秒表示とその他の時間単位」 の表は、秒単位で、同等の時間を別の形式で示しています。

表19.1 秒表示とその他の時間単位

秒	他の時間単位
60	1M
1800	30M
3600	1H
10800	3H
21600	6H
43200	12H
86400	1D
259200	3D
604800	1W
31536000	365D

以下の例は、実際の値が入力されると SOA リソースレコードが取る可能性のあるフォームを示しています。

```
@ IN SOA dns1.example.com. hostmaster.example.com. (
  2001062501 ; serial
  21600      ; refresh after 6 hours
  3600       ; retry after 1 hour
  604800     ; expire after 1 week
  86400 )    ; minimum TTL of 1 day
```

19.3.3. ゾーンファイルの例

ディレクティブとリソースレコードは個別に確認できない可能性があります。ただし、1 つのファイルにまとめて配置すると、理解が容易になります。

以下の例は、非常に基本的なゾーンファイルを示しています。

```
$ORIGIN example.com.
$TTL 86400
@  IN SOA dns1.example.com. hostmaster.example.com. (
    2001062501 ; serial
    21600      ; refresh after 6 hours
    3600       ; retry after 1 hour
    604800     ; expire after 1 week
    86400 )    ; minimum TTL of 1 day
;
;
  IN NS dns1.example.com.
  IN NS dns2.example.com.
dns1  IN A 10.0.1.1
  IN AAAA aaaa:bbbb::1
dns2  IN A 10.0.1.2
  IN AAAA aaaa:bbbb::2
;
;
@  IN MX 10 mail.example.com.
  IN MX 20 mail2.example.com.
mail  IN A 10.0.1.5
  IN AAAA aaaa:bbbb::5
mail2  IN A 10.0.1.6
  IN AAAA aaaa:bbbb::6
;
;
; This sample zone file illustrates sharing the same IP addresses
; for multiple services:
;
services IN A 10.0.1.10
  IN AAAA aaaa:bbbb::10
  IN A 10.0.1.11
  IN AAAA aaaa:bbbb::11
ftp  IN CNAME services.example.com.
www  IN CNAME services.example.com.
;
;
```

この例では、標準のディレクティブと SOA の値が使用されています。権威ネームサーバーは dns1 .example.com および dns2.example.com として設定されます。これには、それぞれ 10. 0.1.1 と 10.0.1. 2 に関連付ける A レコードがあります。

DNS レコードで設定されたメールサーバーは、A レコードを介して mail および mail2 を参照します。mail と mail2 名は末尾のピリオド(.)ではないので、$ORIGIN ドメインは後に置かれ、mail.example.com および mail 2.example.com に拡張されます。関連する A リソースレコードを使用して、それらの IP アドレスを特定できます。

www.example.com (WWW) などの標準名で利用可能なサービスは、CNAME レコードを使用して適切なサーバーを参照します。

このゾーンファイルは、以下のような zone ステートメントが named.conf 内にあるサービスに対して呼び出されます。

```
zone "example.com" IN {
 type master;
 file "example.com.zone";
 allow-update { none; };
};
```

19.3.4. 逆引き名前解決ゾーンファイル

逆引き名前解決ゾーンファイルは、特定の名前空間の IP アドレスを FQDN に変換するために使用されます。これは標準のゾーンファイルに非常に似ていますが、PTR リソースレコードを使用して IP アドレスを完全修飾ドメイン名にリンクする点が異なります。

以下は、PTR レコードのレイアウトを示しています。

```
<last-IP-digit> IN PTR <FQDN-of-system>
```

& lt;last-IP-digit& gt; は、特定のシステムの FQDN を参照する IP アドレスの最後の番号です。

以下の例では、10.0.1.1 から 10.0.1 .6 までの IP アドレスは、対応する FQDN を指してい ます。/var/named/example.com.rr.zone にあります。

```
$ORIGIN 1.0.10.in-addr.arpa.
$TTL 86400
@ IN SOA dns1.example.com. hostmaster.example.com. (
   2001062501 ; serial
   21600      ; refresh after 6 hours
   3600       ; retry after 1 hour
   604800     ; expire after 1 week
   86400 )    ; minimum TTL of 1 day
;
@ IN NS dns1.example.com.
;
1 IN PTR dns1.example.com.
2 IN PTR dns2.example.com.
```

```
;
5 IN PTR server1.example.com.
6 IN PTR server2.example.com.
;
3 IN PTR ftp.example.com.
4 IN PTR ftp.example.com.
```

このゾーンファイルは、以下のような zone ステートメントが named.conf ファイルにあるサービスに対して呼び出されます。

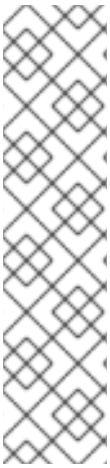
```
zone "1.0.10.in-addr.arpa" IN {
 type master;
 file "example.com.rr.zone";
 allow-update { none; };
};
```

ゾーン名以外は、この例と標準の zone ステートメントにはほとんど違いがありません。逆引き名前解決ゾーンには、IP アドレスの最初の 3 つのブロックが逆になっていてから .in-addr.arpa が続く必要があることに注意してください。これにより、逆引き名前解決ゾーンファイルで使用される IP 番号の単一ブロックがゾーンに関連付けることができます。

19.4. RNDCの使用

BIND には、localhost または a リモートホストから named デーモンのコマンドライン管理を可能にする rndc と呼ばれるユーティリティーが含まれています。

named デーモンへの不正アクセスを防ぐために、BIND は共有秘密鍵認証方法を使用して、ホストに特権を付与します。つまり、同じキーを /etc/named.conf と rndc 設定ファイル /etc/rndc.conf の両方に存在する必要があります。



注記

bind-chroot パッケージをインストールしている場合、BIND サービスは /var/named/chroot 環境で実行されます。すべての設定ファイルはそこに移動します。そのため、rndc.conf ファイルは /var/named/chroot/etc/rndc.conf にあります。

rndc ユーティリティーは chroot 環境で実行されないため、/etc/rndc.conf は /var/named/chroot/etc/rndc.conf へのシンボリックリンクであることに注意してください。

19.4.1. /etc/named.confの設定

rndc が named サービスに接続するには、BIND サーバーの /etc/named.conf ファイルに control ステートメントが必要です。

以下の例のように、controls ステートメントにより、rndc がローカルホストから接続できるようになります。

```
controls {
 inet 127.0.0.1
  allow { localhost; } keys { <key-name>; };
};
```

このステートメントは、ループバックアドレスのデフォルトの TCP ポート 953 をリッスンするように named に指示します。適切なキーが指定されている場合は、localhost からの rndc コマンドを許可します。&lt ;key-name> は、/etc/named.conf ファイル内の キー ステートメントの名前を指定します。次の例は、サンプルの キー ステートメントを示しています。

```
key "<key-name>" {
 algorithm hmac-md5;
 secret "<key-value>";
};
```

この場合、< key-value> は HMAC-MD5 アルゴリズムを使用します。HMAC-MD5 アルゴリズムを使用して鍵を生成するには、以下のコマンドを使用します。

```
dnssec-keygen -a hmac-md5 -b <bit-length> -n HOST <key-file-name>
```

256 ビットの長さ以上の鍵が適しています。<key -value> 領域に配置する実際のキー は、このコマンドによって生成された < key-file-name > ファイルにあります。

WARNING

/etc/named.conf は誰でも読み取り可能であるため、キー ステートメントを別のファイルに置き、root のみが読み取り可能にし、include ステートメントを使用して参照することができます。以下に例を示します。

```
include "/etc/rndc.key";
```

19.4.1.1. ファイアウォールによる通信のブロック

ファイアウォールが named デーモンから他のネームサーバーへの接続をブロックしている場合は、可能な限りファイアウォール設定を変更することが推奨されます。

警告：固定 UDP ソースポートの使用を回避

DNSSEC 検証を実行するように設定されていない、または DNSSEC によって保護されていない DNS ゾーンをクエリーする必要がある DNS リゾルバーは、16 ビットのトランザクション識別子(TXID)と宛先の UDP ポート番号を使用して、DNS データをクエリーするサーバーによって DNS 応答が送信されたかどうかを確認します。

以前は、BIND は DNS クエリーの送信時に、固定 UDP ソースポートを常に使用していました。BIND は、query-source （および query-source -v6）ディレクティブを使用して設定されたポート、または起動時にランダムに選択されたポートを使用していました。静的クエリーソースポートが使用されると、TXID はなりすましの応答に対する保護が十分にないため、攻撃者は キャッシュポイズニング攻撃を効率的に実行できるよう になります。この問題に対処するために、BIND は各 DNS クエリーにランダムに選択されたソースポートを使用できるように更新され、クエリーパケットが検出できない場合に攻撃者が偽装するのがより困難になりました。セキュリティー更新 [3] 影響を受けるすべての Red Hat Enterprise Linux バージョンに対してリリースされました。さらに、caching-nameserver パッケージが提供するデフォルト設定が更新され、固定クエリーソースポートが指定されなくなりました。

BIND を DNS リゾルバーとしてデプロイする場合は、前述の設定ディレクティブで BIND が強制的に実行されないようにし、固定クエリーソースポートを使用します。ファイアウォール設定では、ランダムなクエリーソースポートの使用も許可される必要があります。以前は、BIND がポート 53 をクエリーソースポートとして使用し、ファイアウォール上のそのポートからの DNS クエリーのみを許可するように BIND を設定することが一般的でした。

19.4.2. /etc/rndc.confの設定

キー は、/etc/rndc.conf の最も重要なステートメントです。

```
key "<key-name>" {
 algorithm hmac-md5;
 secret "<key-value>";
};
```

& lt;key-name& gt; および <key-value > は、/etc/named.conf の設定と全く同じです。

ターゲットサーバーの /etc/named.conf で指定した鍵と一致するには、以下の行を /etc/rndc.conf に追加します。

```
options {
 default-server  localhost;
 default-key     "<key-name>";
};
```

このディレクティブは、グローバルデフォルトキーを設定します。ただし、rndc 設定ファイルは、以下の例のように異なるサーバーに異なるキーを指定することもできます。

```
server localhost {
 key  "<key-name>";
};
```

重要な影響

root ユーザーのみが /etc/rndc.conf ファイルに対して読み取りまたは書き込みを行うことができることを確認してください。

/etc/rndc.conf ファイルの詳細は、rndc.conf の man ページを参照してください。

19.4.3. コマンドラインオプション

rndc コマンドは以下の形式になります。

```
rndc <options> <command> <command-options>
```

適切に設定された localhost で rndc を実行すると、以下のコマンドが利用可能になります。

- halt: named サービスをすぐに停止します。
- querylog: このネームサーバーに対して行われたすべてのクエリーをログに記録します。

- *refresh: ネームサーバーのデータベースを更新します。*

- *reload - ゾーンファイルを再読み込みしますが、以前にキャッシュされた応答をすべて保持します。このコマンドでは、保存された名前解決をすべて失わずにゾーンファイルに変更を加えることもできます。*

  変更が特定のゾーンのみに影響する場合は、reload コマンドの後にゾーンの名前を追加して、その特定のゾーンのみを再読み込みします。

- *stats: 現在の名前付き統計を /var/ named /named.stats ファイルにダンプします。*

- *stop - サーバーを正常に停止し、終了する前に動的更新および 増分ゾーンの転送 (IXFR) データを保存します。*

場合によっては、/etc/rndc.conf ファイルのデフォルト設定を上書きする必要がある場合があります。以下のオプションを設定できます。

- *-c & lt;configuration-file& gt;: 設定ファイルの代替の場所を指定します。*

- *-p <port-number > - デフォルトのポート 953 以外の rndc 接続に使用するポート番号を指定します。*

- *-s & lt;server& gt;: /etc/rndc.conf に一覧表示されている default-server 以外のサーバーを指定します。*

- *-y & lt;key-name > - /etc/rndc.conf の default-key オプション以外のキーを指定します。*

これらのオプションの詳細は、rndc の man ページを参照してください。

19.5. BIND の高度な機能

ほとんどの BIND 実装は、named を使用して名前解決サービスを提供したり、特定のドメインまたはサブドメインの認証局として機能します。ただし、BIND バージョン 9 には、より安全で効率的な

DNS サービスを可能にする多くの高度な機能があります。

注意

DNSSEC、TSIG、IXFR （以下のセクションで定義されている）などの一部の高度な機能は、この機能をサポートするネームサーバーを持つネットワーク環境でのみ使用する必要があります。ネットワーク環境に BIND 以外のまたは古い BIND ネームサーバーが含まれている場合は、使用する前に各高度な機能がサポートされていることを確認してください。

上記のすべての機能は、「インストールされているドキュメント」 で参照されている 『BIND 9 Administrator Reference Manual』 でより詳細に説明されています。

19.5.1. DNS プロトコルの機能強化

BIND は増分ゾーン転送(IXFR)をサポートします。この場合、スレーブネームサーバーはマスターネームサーバーで変更されたゾーンの更新部分のみをダウンロードします。標準の転送プロセスでは、最小変更のためにゾーン全体を各スレーブネームサーバーに転送する必要があります。非常に長いゾーンファイルと多くのスレーブネームサーバーを持つ非常に一般的なドメインでは、IXFR により、通知および更新プロセスがリソースを大量に消費することになります。

IXFR は、動的更新 を使用してマスターゾーンレコードに変更を加える場合にのみ使用できることに注意してください。ゾーンファイルを手動で編集して変更する場合は、自動ゾーン転送(AXFR)が使用されます。動的更新の詳細は、「インストールされているドキュメント」 で参照されている 『BIND 9 Administrator Reference Manual』 を参照してください。

19.5.2. 複数表示

named.conf の view ステートメントを使用すると、BIND は、要求が発信するネットワークに応じて異なる情報を提供できます。

これは主に、ローカルネットワーク外のクライアントからの機密性の高い DNS エントリーを拒否する一方で、ローカルネットワーク内のクライアントからのクエリーを許可するために使用されます。

view ステートメントは、match-clients オプションを使用して IP アドレス全体またはネットワーク全体を照合し、特別なオプションとゾーンデータを提供します。

19.5.3. セキュリティー

BIND は、マスターネームサーバーとスレーブネームサーバーの両方で、ゾーンの更新と転送を保護するさまざまな方法をサポートしています。

DNSSEC

DNS SECurity の略です。この機能により、ゾーンを ゾーンキー で暗号で署名できます。

これにより、受信者がそのネームサーバーの公開鍵を持っている限り、特定のゾーンに関する情報を、特定の秘密鍵で署名したネームサーバーからの情報を確認できます。

BIND バージョン 9 は、メッセージ認証の SIG (0)パブリック/秘密鍵メソッドもサポートします。

TSIG

Transaction SIGnatures の略でこの機能は、共有の秘密鍵が両方のネームサーバーに存在することを確認した後のみ、マスターからスレーブへ転送できるようにします。

この機能は、転送承認の標準の IP アドレスベースの方法を強化します。攻撃者は、ゾーンを転送するために IP アドレスにアクセスする必要があるだけでなく、秘密鍵も知っておく必要があります。

BIND バージョン 9 は、ゾーン転送を承認するもう 1 つの共有秘密鍵メソッドである TKEY にも対応しています。

19.5.4. IP バージョン 6

BIND バージョン 9 は、A6 ゾーンレコードを使用して IP バージョン 6 (IPv6)環境での name サービスをサポートします。

ネットワーク環境に IPv4 と IPv6 の両方のホストが含まれている場合は、すべてのネットワーククライアントで Lw resd 軽量リゾルバーデーモンを使用します。このデーモンは、IPv6 で使用される新しい A6 レコードおよび DNAME レコードを理解するための、非常に効率的なキャッシュ専用ネームサーバーです。詳細は、man ページの lwresd を参照してください。

19.6. 回避すべき一般的な間違い

BIND 設定ファイルの編集時に、初心者にとって非常に一般的です。以下の問題は使用しないでください。

- ゾーンファイルを編集するときは、シリアル番号を増やすようにしてください。

  シリアル番号がインクリメントされない場合、マスターネームサーバーは正しく新しい情報を持ちますが、スレーブネームサーバーには変更は通知されず、そのゾーンのデータの更新は試行されません。

- /etc/named.conf ファイルで、ellipses と semi-colons を正しく使用するようにしてください。

  セミコロンまたは閉じていない省略セクションを使用すると、named の起動を拒否する場合があります。

- すべての FQDN の後にゾーンファイルにピリオド(.)を配置するのを忘れて、ホスト名で省略するようにしてください。

  ドメイン名の最後にあるピリオドは、完全修飾ドメイン名を示します。ピリオドを省略すると、named はゾーンの名前または $ORIGIN 値を追加して完了します。

- ファイアウォールが named デーモンから他のネームサーバーへの接続をブロックしている場合は、可能な限りファイアウォール設定を変更することが推奨されます。固定 UDP ソースポートに関する重要なセキュリティー情報については、を参照してください。 「ファイアウォールによる通信のブロック」

19.7. 関連情報

以下の資料は、BIND に関するその他のリソースを提供します。

19.7.1. インストールされているドキュメント

BIND は、多種多様なトピックを網羅した広範囲に及ぶインストール済みのドキュメントを特徴としています。以下の各項目について、&lt ;version-number> を、システムにインストールされている

バインド のバージョンに置き換えます。

/usr/share/doc/bind-<version-number>/

このディレクトリーには、最新の機能が一覧表示されます。

/usr/share/doc/bind-<version-number>/arm/

このディレクトリーには、HTML および SGML 形式の 『BIND 9 Administrator Reference Manual』 が含まれています。このマニュアルでは、BIND リソース要件、異なるタイプのネームサーバーの設定方法、負荷分散の実行方法、およびその他の高度なトピックについて詳しく説明します。BIND のほとんどの新規ユーザーでは、これを開始するのが最適な場所です。

/usr/share/doc/bind-<version-number>/draft/

このディレクトリーには、DNS サービスに関連する問題をレビューし、それらに対応する方法を提案するさまざまな技術ドキュメントが含まれています。

/usr/share/doc/bind-<version-number>/misc/

このディレクトリーには、特定の高度な問題に対処するために設計されたドキュメントが含まれています。BIND バージョン 8 のユーザーは、BIND 9 に 移行する際に必要な特定の変更については、移行 ドキュメントを参照してください。options ファイルには、/etc/named.conf で使用される BIND 9 に実装されたすべてのオプションが一覧表示されます。

/usr/share/doc/bind-<version-number>/rfc/

このディレクトリーは、BIND に関連するすべての RFC ドキュメントを提供します。

BIND に関連するさまざまなアプリケーションおよび設定ファイルの man ページも多数あります。以下は、重要な man ページの一部を示しています。

管理アプリケーション

- man rndc: rndc コマンドを使用して BIND ネームサーバーを制御する際に使用できるさまざまなオプションを説明します。

サーバーアプリケーション

- *man named - BIND nameserver デーモンの制御に使用できるさまざまな引数を調べます。*
- *man lwresd: 軽量リゾルバーデーモンで利用可能な および オプションの目的を説明します。*

設定ファイル

- *man named.conf: named 設定ファイル内で利用可能なオプションの包括的な一覧です。*
- *man rndc.conf: rndc 設定ファイル内で利用可能なオプションの包括的な一覧。*

19.7.2. 便利な Web サイト

- *http://www.isc.org/index.pl?/sw/bind/ - BIND プロジェクトのホームページには、現在のリリースに関する情報と 『BIND 9 Administrator Reference Manual』 の PDF バージョンが含まれています。*
- *http://www.redhat.com/mirrors/LDP/HOWTO/DNS-HOWTO.html - BIND を解決、キャッシュネームサーバーとして使用すること、およびドメインのプライマリーネームサーバーとして機能するために必要なさまざまなゾーンファイルの設定を説明します。*

19.7.3. 関連書籍

- *Paul Albitz 『および Cricket Liu による DNS および BIND』: O'Reilly & Associates - 一般的な BIND 設定オプションと、DNS サーバーのセキュリティーを保護するストラテジーを提供する一般的なリファレンスです。*
- *Nicolai Langfeldt 『による DNS と BIND の Concise Guide』、Que - 複数のネットワークサービスと BIND 間の接続を見て、タスク指向の技術トピックを中心としています。*

[3]

セキュリティー更新は RHSA-2008:0533 でした。

第20章 OPENSSH

SSH™（または Secure SHell）は、クライアント/サーバーアーキテクチャーを使用した 2 つのシステム間のセキュアな通信を容易にし、ユーザーがリモートでサーバーホストシステムにログインできるようにするプロトコルです。FTP や Telnet などの他のリモート通信プロトコルとは異なり、SSH はログインセッションを暗号化するため、侵入者が接続して暗号化されていないパスワードを収集するのが困難になります。

SSH は、telnet や rsh などのリモートホストへのログインに使用される、以前の、安全ではない端末アプリケーションを置き換えるように設計されています。scp と呼ばれる関連プログラムは、ホスト間でファイルをコピーするために設計された rcp などの古いプログラムに代わるものです。このような旧式アプリケーションは、クライアントとサーバーとの間で送信するパスワードを暗号化しないため、可能な限り使用しないようにしてください。リモートシステムへのログインにセキュアな方法を使用すると、クライアントシステムとリモートホストの両方のリスクが軽減されます。

20.1. SSH の機能

SSH プロトコルは、以下のような保護手段を提供します。

- クライアントは、初回接続後に、以前接続したサーバーと同じサーバーに接続していることを確認できます。
- クライアントは、強力な 128 ビット暗号化を使用して、サーバーへ認証情報を送信します。
- セッション中に送受信された全データは、128 ビット暗号化を使用して転送されるため、傍受された送信データの暗号解読と読み取りは非常に困難になります。
- クライアントは X11 を転送できます。[4] サーバーからのアプリケーション。X11 転送 と呼ばれるこの手法は、ネットワーク上でグラフィカルアプリケーションを使用する安全な手段を提供します。

SSH プロトコルは送受信するものをすべて暗号化するため、セキュアでないプロトコルをセキュアにするために使用できます。SSH サーバーは、ポート転送 と呼ばれる技術を使用して、POP などのセキュリティー保護されていないプロトコルをセキュアにし、システム全体のシステムおよびデータセキュリティーを強化できます。

OpenSSH サーバーおよびクライアントは、サーバーマシンとクライアントマシン間のトラフィックの仮想プライベートネットワークと同様のトンネルを作成するように設定することもできます。

最後に、OpenSSH サーバーおよびクライアントは、Kerberos ネットワーク認証プロトコルの GSSAPI 実装を使用して認証するように設定できます。Kerberos 認証サービスの設定に関する詳細は、「Kerberos」 を参照してください。

Red Hat Enterprise Linux には、一般的な OpenSSH パッケージ(openssh)と、OpenSSH サーバー(openssh-server)およびクライアント(openssh-clients)のパッケージが含まれます。OpenSSH パッケージには、OpenSSL パッケージ(openssl)が必要です。このパッケージは、重要な暗号化ライブラリーを複数インストールし、OpenSSH が暗号化された通信を提供するようにします。

20.1.1. SSH を使用する理由

Nefarious コンピューターユーザーは、システムにアクセスするためにネットワークトラフィックの中断、傍受、および再ルーティングを可能にするさまざまなツールを利用できます。一般的には、これらの脅威は以下のとおり分類できます。

- 2 つのシステム間の通信の 傍受：このシナリオでは、攻撃者は通信者間で渡される情報をネットワーク上のどこかに置くことができます。攻撃者は情報を傍受して維持したり、情報を変更したり、目的の受信者に送信したりする可能性があります。

  この攻撃は、一般的なネットワークユーティリティーであるパケットスニッファーを使用してマウントできます。

- 特定ホストの偽装 - このストラテジーを使用すると、攻撃者のシステムは、送信の対象となる受信者として機能するように設定されています。このストラテジーが機能すると、ユーザーのシステムは間違ったホストと通信していることを認識しません。

  この攻撃は DNS ポイズニングと呼ばれる手法でマウントできます。[5] または IP スプーフィング[6]をクリックします。

いずれの手法でも、潜在的な機密情報を傍受することが可能です。その傍受が悪意のある理由で行われる場合には、多大な損害をもたらしかねません。

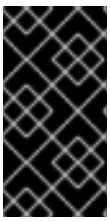
リモートシェルログインとファイルコピー用に SSH を使用すると、こうしたセキュリティーの脅威を大幅に軽減できます。これは、SSH クライアントとサーバーがデジタル署名を使用してそれぞれの ID を確認するためです。さらに、クライアントシステムとサーバーシステムとの間の通信はすべて暗号

化されます。各パケットはローカルシステムとリモートシステムのみに知られている鍵を使用して暗号化されるため、通信のいずれか一方の ID をスプーフィングする試みは成功しません。

20.2. SSH プロトコルのバージョン

SSH プロトコルを使用すると、プロトコルの仕様で構築されたクライアントおよびサーバープログラムに安全と通信し、相互に置き換え可能なものとして使用できます。

現在、2 種類の SSH （バージョン 1 とバージョン 2）があります。Red Hat Enterprise Linux の OpenSSH スイートは、SSH バージョン 2 を使用します。バージョン 1 の悪用に対して脆弱ではない、強化された鍵交換アルゴリズムを備えています。ただし、OpenSSH スイートはバージョン 1 の接続をサポートします。



重要な影響

可能な場合は、SSH バージョン 2 互換のサーバーとクライアントのみを使用することが推奨されます。

20.3. SSH 接続のイベントシーケンス

以下に挙げる一連のイベントは、2 つのホスト間で行われる SSH 通信の整合性を保護するのに役立ちます。

1. 暗号化ハンドシェイクが行われ、クライアントが正しいサーバーと通信していることを確認できます。

2. クライアントとリモートホストとの間の接続のトランスポート層が、対称暗号方式を使用して暗号化されます。

3. クライアントが、サーバーに対して自己認証します。

4. リモートクライアントは、暗号化された接続でリモートホストと対話します。

20.3.1. トランスポート層

トランスポート層の主なロールは、認証時とその後の通信中に、2 つのホスト間の通信を簡単に安全でセキュアなものにすることです。トランスポート層は、データの暗号化と復号を処理し、データパ

ケットの送受信時にその整合性を保護することでそのロールを果たします。また、トランスポート層は、情報を圧縮して転送を高速にします。

SSH クライアントがサーバーに接続すると鍵情報が交換されるため、両システムでトランスポート層が適正に構築できます。以下は、こうした鍵情報の交換中に発生する手順です。

- 鍵を交換する
- 公開鍵暗号化アルゴリズムが決定する
- 対称暗号化アルゴリズムが決定する
- メッセージ認証アルゴリズムが決定する
- ハッシュアルゴリズムが決定する

キー交換時に、サーバーは一意の ホストキー でクライアントに対して自己識別します。クライアントがこの特定のサーバーと通信していない場合、サーバーのホストキーはクライアントに認識されず、接続されません。OpenSSH は、サーバーのホスト鍵を受け入れることでこの問題を回避します。これは、ユーザーが通知を受けて新規のホスト鍵を受け取り、検証した後に行われます。後続の接続では、クライアントのホストキーがクライアントに保存されているバージョンに対してチェックされ、クライアントが実際に目的のサーバーと通信していることが信頼されます。今後、ホストキーが一致しなくなった場合、ユーザーは接続を行う前に、クライアントに保存されているバージョンを削除する必要があります。

注意

ローカルシステムは、対象サーバーと攻撃者が設定した偽サーバーとの違いを認識しないため、攻撃者は初回コンタクト中に SSH サーバーをマスカレードすることが可能です。この問題を防ぐために、初回接続の前かホスト鍵の不一致が発生した場合には、サーバー管理者へ連絡して新しい SSH サーバーの整合性を確認してください。

SSH は、ほとんどすべての公開鍵アルゴリズムまたはエンコード形式に対応するように設計されています。初回の鍵交換で、交換に使用されるハッシュ値と共有秘密値が作成されると、2 つのシステムは新しい鍵とアルゴリズムの計算を直ちに開始して、認証と、今後の接続で送信されるデータを保護します。

所定の鍵とアルゴリズムを使用して一定量のデータ (正確な量は SSH 実装により異なる) が送信された後に、もう 1 回鍵交換が行われてハッシュ値と新しい共有秘密値の別のセットが生成されます。攻撃者がハッシュ値と共有秘密値を判別できたとしても、その情報が役に立つのは限られた時間のみです。

20.3.2. 認証

トランスポート層が、2 つのシステム間で情報を渡すためのセキュアなトンネルを構築すると、サーバーは、秘密鍵でエンコードされた署名の使用やパスワードの入力など、サポートされている別の認証方法をクライアントに伝えます。次に、クライアントが、対応しているいずれかの方法を使用して、サーバーに対して自己認証を試みます。

SSH サーバーとクライアントは、異なるタイプの認証を採用するように設定できるため、双方の制御が最適化されます。サーバーは、そのセキュリティーモデルに基づいて、対応する暗号化方法を決定できます。クライアントは、利用可能なオプションの中から、試行する認証方法の順番を選択できます。

20.3.3. チャネル

SSH トランスポート層での認証に成功すると、多重化と呼ばれる手法により複数のチャンネルが開きます。[7].これらの各チャンネルは、異なるターミナルセッションと、転送された X11 セッションの通信を処理します。

クライアントとサーバーの両方で、新しいチャンネルを作成できます。その後、各接続の両端に、別々の番号が割り当てられます。クライアントが新しいチャンネルを開こうとする際、要求と共にチャンネル番号を送信します。この情報はサーバーにより保存され、そのチャンネルに通信を移動するのに使用されます。これは、異なるタイプのセッションが相互に影響しないように、あるセッションの終了時にそのチャンネルが SSH による一次接続を停止せずに閉じることができるようにするためです。

チャネルは、フロー制御 もサポートします。これにより、順番通りにデータを送受信できます。この方法では、チャンネルが開いているというメッセージをクライアントが受信するまで、チャンネルでデータが送信されません。

クライアントが要求するサービスのタイプと、ユーザーがネットワークに接続される方法に応じて、クライアントとサーバーは、各チャンネルの特性を自動的にネゴシエートします。これにより、プ

ロトコルの基本インフラストラクチャーを変更しなくても、異なるタイプのリモート接続を非常に柔軟に処理できます。

20.4. OPENSSH サーバーの設定

OpenSSH サーバーを実行するには、まず適切な RPM パッケージがインストールされている必要があります。openssh-server パッケージが必要で、openssh パッケージに依存します。

OpenSSH デーモンは、設定ファイル /etc/ssh/sshd_config を使用します。デフォルトの設定ファイルは、ほとんどの目的で十分です。デフォルトの sshd_config では提供されない方法で デーモンを設定する場合は、設定ファイルで定義できるキーワードの一覧については、sshd の man ページを参照してください。

OpenSSH サービスを起動するには、コマンド /sbin/service sshd start を使用します。OpenSSH サーバーを停止するには、コマンド /sbin/service sshd stop を使用します。システムの起動時にデーモンが自動的に起動するようにするには、18章 を参照してください。

再インストールすると、再インストールされたシステムは新しい ID キーのセットを作成します。再インストールの前にいずれかの OpenSSH ツールを使用してシステムに接続したクライアントには、以下のメッセージが表示されます。

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that the RSA host key has just been changed.
```

システムに生成されたホストキーを保持する場合は、/etc/ssh/ssh_host*key* ファイルをバックアップし、再インストール後に復元します。このプロセスはシステムのアイデンティティーを保持し、再インストール後にクライアントがシステムに接続しようとすると、警告メッセージを受信しません。

20.4.1. リモート接続に必要な SSH

SSH を本当の意味で有効なものにするには、Telnet や FTP などの安全ではない接続プロトコルを使用する必要があります。それ以外の場合は、ユーザーのパスワードが SSH を使用して 1 つのセッションで保護され、Telnet を使用したログイン時に後でキャプチャーされます。

無効にするサービスには以下が含まれます。

- *telnet*
- *rsh*
- *rlogin*
- *vsftpd*

システムに対するセキュアでない接続メソッドを無効にするには、コマンドラインプログラム chkconfig、ncurses ベースのプログラム /usr/sbin/ntsysv、または Services Configuration Tool (system-config-services)グラフィカルアプリケーションを使用します。これらのツールはすべて、ルートレベルのアクセスを必要とします。

chkconfig、/usr/sbin/ntsysv、および Services Configuration Tool を使用してランレベルとサービスを設定する方法の詳細は、18章 を参照してください。

20.5. OPENSSH 設定ファイル

OpenSSH には、クライアントプログラム(ssh、scp、および sftp)用とサーバーデーモン(sshd)の 2 つの異なる設定ファイルセットがあります。

システム全体の SSH 設定情報は /etc/ssh/ ディレクトリーに保存されます。

- *moduli - 安全なトランスポート層を構築する際に重要な Diffie-Hellman 鍵交換に使用される Diffie-Hellman グループが含まれます。SSH セッションの初めに鍵が交換される時、共有秘密値が作成されますが、どちらか一方の当事者だけでは決定できません。この値は、ホスト認証を行うのに使用されます。*
- *ssh_config: システム全体のデフォルトの SSH クライアント設定ファイル。これは、ユーザーのホームディレクトリー(~/.ssh/config)にもある場合に上書きされます。*
- *sshd_config - sshd デーモンの設定ファイルです。*

- *ssh_host_dsa_key: sshd デーモンが使用する DSA 秘密鍵です。*
- *ssh_host_dsa_key.pub: sshd デーモンが使用する DSA 公開鍵です。*
- *ssh_host_key - sshd デーモンが使用する SSH プロトコルのバージョン 1 用の RSA 秘密鍵です。*
- *ssh_host_key.pub: sshd デーモンが使用する SSH プロトコルのバージョン 1 用の RSA 公開鍵です。*
- *ssh_host_rsa_key: sshd デーモンが使用する SSH プロトコルのバージョン 2 用の RSA 秘密鍵です。*
- *ssh_host_rsa_key.pub: sshd が使用する SSH プロトコルのバージョン 2 用の RSA 公開鍵です。*

ユーザー固有の SSH 設定情報は、~/.ssh/ ディレクトリー内のユーザーのホームディレクトリーに保存されます。

- *authorized_keys: このファイルは、サーバーの認証済み公開鍵の一覧を保持します。クライアントがサーバーに接続すると、サーバーが、このファイル内に格納されている署名済み公開鍵を確認してクライアントを認証します。*
- *id_dsa: ユーザーの DSA 秘密鍵が含まれます。*
- *id_dsa.pub: ユーザーの DSA 公開鍵です。*
- *id_rsa: ssh が使用する SSH プロトコルのバージョン 2 用の RSA 秘密鍵です。*
- *id_rsa.pub: SSH プロトコルのバージョン 2 用の ssh が使用する RSA 公開鍵です。*

- *identity - ssh が使用する SSH プロトコルのバージョン 1 用の RSA 秘密鍵です。*

- *identity.pub: ssh が使用する SSH プロトコルのバージョン 1 用の RSA 公開鍵です。*

- *known_hosts: このファイルには、ユーザーがアクセスする SSH サーバーの DSA ホストキーが含まれます。このファイルは、SSH クライアントが正しい SSH サーバーに接続していることを確認するために非常に重要です。*

重要な影響

SSH サーバーのホストキーが変更された場合、クライアントは、テキストエディターを使用してサーバーのホストキーが known_hosts ファイルから削除されるまで接続が続行できないことをユーザーに通知します。ただし、これを実行する前に、SSH サーバーのシステム管理者に連絡して、サーバーが被害を受けていないことを確認してください。

SSH 設定ファイルで利用可能な各種ディレクティブの詳細は、ssh_config および sshd_config の man ページを参照してください。

20.6. OPENSSH クライアントの設定

クライアントマシンから OpenSSH サーバーに接続するには、クライアントマシンに openssh-clients パッケージおよび openssh パッケージをインストールする必要があります。

20.6.1. ssh コマンドの使用

ssh コマンドは、rlogin コマンド、rsh コマンド、および telnet コマンドに代わる安全なコマンドです。これにより、リモートマシンにログインし、リモートマシンでコマンドを実行できます。

ssh でリモートマシンにログインすることは、telnet の使用と似ています。penguin.example.net という名前のリモートマシンにログインするには、シェルプロンプトで以下のコマンドを入力します。

```
ssh penguin.example.net
```

リモートマシンに初めて ssh を実行すると、以下のようなメッセージが表示されます。

```
The authenticity of host 'penguin.example.net' can't be established.
DSA key fingerprint is 94:68:3a:3a:bc:f3:9a:9b:01:5d:b3:07:38:e2:11:0c.
Are you sure you want to continue connecting (yes/no)?
```

続行するには **yes** と入力します。これにより、以下のメッセージに示されるように、サーバーが既知のホスト(**~/.ssh/known_hosts**)の一覧に追加されます。

```
Warning: Permanently added 'penguin.example.net' (RSA) to the list of known hosts.
```

次に、リモートマシンのパスワードを要求するプロンプトが表示されます。パスワードを入力すると、リモートマシンのシェルプロンプトが表示されます。ユーザー名を指定しないと、ローカルクライアントマシンでログインしているユーザー名がリモートマシンに渡されます。別のユーザー名を指定する場合は、以下のコマンドを使用します。

```
ssh username@penguin.example.net
```

構文 **ssh -l username penguin.example.net** を使用することもできます。

ssh コマンドを使用すると、シェルプロンプトにログインせずに、リモートマシンでコマンドを実行できます。構文は **ssh hostname** コマンド です。たとえば、リモートマシン **penguin.example.net** で **ls /usr/share/doc** コマンドを実行する場合は、シェルプロンプトで以下のコマンドを入力します。

```
ssh penguin.example.net ls /usr/share/doc
```

正しいパスワードを入力すると、リモートディレクトリー **/usr/share/doc** の内容が表示され、ローカルのシェルプロンプトに戻ります。

20.6.2. scp コマンドの使用

scp コマンドを使用すると、暗号化されたセキュアな接続でマシン間でファイルを転送することができます。これは **rcp** に似ています。

ローカルファイルをリモートシステムに転送する一般的な構文は、以下のとおりです。

```
scp <localfile> username@tohostname:<remotefile>
```

&lt ;localfile> は、**/var/log/maillog** などのファイルへのパスを含むソースを指定します。**&lt ;remotefile>** は宛先を指定します。これは、**/tmp/hostname-maillog** などの新しいファイル名にす

ることができます。リモートシステムでは、前に / がない場合、パスはユーザー名（通常は /home/username / ）のホームディレクトリーと相対的になります。

ローカルファイルの shadowman を penguin.example.net 上のアカウントのホームディレクトリーに転送するには、シェルプロンプトで以下を入力します(username はユーザー名に置き換えてください)。

```
scp shadowman username@penguin.example.net:shadowman
```

これにより、ローカルファイル shadowman が penguin.example.net の /home/username/shadowman に転送されます。または、scp コマンドで最後の shadowman をオフにすることもできます。

リモートファイルをローカルシステムに転送する一般的な構文は、以下のとおりです。

```
scp username@tohostname:<remotefile> <newlocalfile>
```

&lt ;remotefile> はパスを含むソースを指定し、 &lt ;newlocalfile> はパスを含む宛先を指定します。

複数のファイルをソースファイルとして指定できます。たとえば、ディレクトリー downloads/ の内容を、リモートマシン penguin.example.net の upload / と呼ばれる既存のディレクトリーに転送するには、シェルプロンプトで以下を入力します。

```
scp downloads/* username@penguin.example.net:uploads/
```

20.6.3. sftp コマンドの使用

sftp ユーティリティーを使用して、セキュアでインタラクティブな FTP セッションを開くことができます。これは、セキュアで暗号化された接続を使用する点を除いて ftp と似ています。一般的な構文は sftp username@hostname.com です。認証が終わると、FTP で使用されるコマンドと同様のコマンドセットを使用できます。これらのコマンドの一覧は、sftp の man ページを参照してください。man ページを読むには、シェルプロンプトでコマンド man sftp を実行します。sftp ユーティリティーは、OpenSSH バージョン 2.5.0p1 以降でのみ使用できます。

20.7. セキュアなシェルの追加

セキュアなコマンドラインインターフェイスは、SSH を使用する多くの方法の先頭にすぎません。十分な帯域幅があれば、X11 セッションは SSH チャンネル上で送信できます。あるいは、TCP/IP 転送

を使用することで、以前はセキュリティー保護されていなかったシステム間のポート接続を特定の SSH チャンネルにマッピングすることができます。

20.7.1. X11 転送

SSH 接続で X11 セッションを開くことは、-Y オプションを使用し、ローカルマシンで X プログラムを実行して SSH サーバーへの接続を簡単に実行できます。

```
ssh -Y <user>@example.com
```

セキュアなシェルプロンプトから X プログラムを実行すると、SSH クライアントとサーバーは新しいセキュアなチャンネルを作成し、X プログラムデータはそのチャンネル上で透過的にクライアントマシンに送信されます。

X11 転送は非常に便利なものです。たとえば、X11 転送を使用して、プリンター設定ツールのセキュアでインタラクティブなセッションを作成できます。これを行うには、ssh を使用してサーバーに接続し、以下のコマンドを入力します。

```
system-config-printer &
```

サーバーの root パスワードを指定すると、Printer Configuration Tool が表示され、リモートユーザーがリモートシステムで印刷を安全に設定できるようになります。

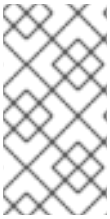
20.7.2. ポート転送

SSH は、ポート転送によりセキュリティー保護されていない TCP/IP プロトコルをセキュアにすることができます。この手法を使用する場合、SSH サーバーは SSH クライアントをつなぐ暗号化された経路となります。

ポート転送は、クライアント上のローカルポートをサーバー上のリモートポートにマッピングすることで機能します。SSH は、サーバーからクライアントの任意のポートにマップできます。この手法が機能するにはポート番号を一致させる必要はありません。

localhost で接続をリッスンする TCP/IP ポート転送チャンネルを作成するには、以下のコマンドを使用します。

```
ssh -L local-port:remote-hostname:remote-port username@hostname
```



注記

1024 未満のポートでリッスンするようにポート転送を設定するには、root レベルのアクセスが必要です。

暗号化された接続で POP3 を使用して mail.example.com という名前のサーバーで電子メールを確認するには、以下のコマンドを使用します。

```
ssh -L 1100:mail.example.com:110 mail.example.com
```

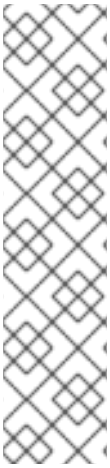
ポート転送チャンネルがクライアントマシンとメールサーバー間に配置されたら、POP3 メールクライアントに localhost 上のポート 1100 を使用して新しいメールを確認するように指示します。クライアントシステム上のポート 1100 に送信されたリクエストは、安全に mail.example.com サーバーに転送されます。

mail.example.com が SSH サーバーを実行しておらず、同じネットワーク上の別のマシンが でも、SSH を使用して接続の一部をセキュアにすることができます。ただし、若干異なるコマンドが必要になります。

```
ssh -L 1100:mail.example.com:110 other.example.com
```

この例では、クライアントマシン上のポート 1100 からの POP3 リクエストは、ポート 22 の SSH 接続を介して SSH サーバー other.example.com に転送されます。次に、other.example.com は mail.example.com のポート 110 に接続して、新しいメールを確認します。この手法を使用する場合、クライアントシステムと other.example.com SSH サーバー間の接続のみがセキュアである点に注意してください。

ポート転送は、ネットワークのファイアウォール経由でセキュアに情報を取得する場合にも使用できます。ファイアウォールが標準ポート(22)経由の SSH トラフィックを許可するように設定されていて、他のポートへのアクセスをブロックする場合は、確立された SSH 接続を介して通信をリダイレクトすることで、ブロックされたポートを使用する 2 つのホスト間の接続は引き続き可能です。



注記

この方法でポート転送を使って接続を転送すると、クライアントシステム上のどのユーザーもそのサービスに接続できます。クライアントシステムが侵害された場合、攻撃者は転送されたサービスにアクセスすることもできます。

ポート転送に関するシステム管理者は、/etc/ssh/sshd_config の AllowTcpForwarding 行に No パラメーターを指定して sshd サービスを再起動することで、この機能をサーバー上で無効にできます。

20.7.3. 鍵ペアの生成

ssh、scp、または sftp を使用してリモートマシンに接続するたびにパスワードを入力する必要がない場合は、認証キーペアを生成できます。

キーはユーザーごとに生成する必要があります。ユーザーのキーを生成するには、リモートマシンに接続するユーザーとして以下の手順を使用します。root で手順を完了すると、鍵を使用できるのは root のみです。

OpenSSH バージョン 3.0 以降では、~/.ssh/authorized_keys2、~/ssh/known_hosts2、および /etc/ssh_known_hosts2 は廃止されました。SSH プロトコル 1 と 2 は、~/.ssh/authorized_keys、~/.ssh/known_hosts ファイル、および /etc/ssh/ssh_known_hosts ファイルを共有します。

Red Hat Enterprise Linux 5.10 は、デフォルトで SSH プロトコル 2 および RSA 鍵を使用します。

ヒント

生成されたキーペアを再インストールして保存する場合は、ホームディレクトリーに .ssh ディレクトリーのバックアップを作成します。再インストール後に、このディレクトリーをホームディレクトリーにコピーします。このプロセスは、システムの全ユーザー(root など)で実行できます。

20.7.3.1. バージョン 2 用の RSA 鍵ペアの生成

以下の手順に従って、SSH プロトコルのバージョン 2 用の RSA 鍵ペアを生成します。これは、OpenSSH 2.9 以降のデフォルトです。

1.

プロトコルのバージョン 2 と連携するように RSA 鍵ペアを生成するには、シェルプロンプトで以下のコマンドを入力します。

```
ssh-keygen -t rsa
```

~/.ssh/id_rsa のデフォルトのファイルの場所を受け入れます。アカウントのパスワードとは別のパスフレーズを入力し、再度入力して確認します。

公開鍵は ~/.ssh/id_rsa.pub に書き込まれます。秘密鍵は ~/.ssh/id_rsa に書き込まれます。秘密鍵を誰にも配布しないでください。

2. 以下のコマンドを使用して、.ssh ディレクトリーのパーミッションを変更します。

```
chmod 755 ~/.ssh
```

3. ~/.ssh/id_rsa.pub の内容を、接続するマシンの ~/.ssh/authorized_keys ファイルにコピーします。~/.ssh/authorized_keys ファイルが存在する場合は、~/.ssh/id_rsa.pub ファイルの内容を他のマシンの ~/.ssh/authorized_keys ファイルに追加します。

4. 以下のコマンドを使用して authorized_keys ファイルのパーミッションを変更します。

```
chmod 644 ~/.ssh/authorized_keys
```

5. GNOME を実行しているか、GTK2+ ライブラリーがインストールされているグラフィカルデスクトップで実行している場合は、「GUI を使用した ssh-agent の設定」 に進みます。X Window System を実行していない場合は、「ssh-agentの設定」 に進みます。

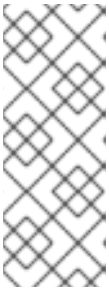
20.7.3.2. バージョン 2 用の DSA キーペアの生成

以下の手順に従って、SSH プロトコルのバージョン 2 用の DSA キーペアを生成します。

1. プロトコルのバージョン 2 と連携するように DSA キーペアを生成するには、シェルプロンプトで以下のコマンドを入力します。

```
ssh-keygen -t dsa
```

~/.ssh/id_dsa のデフォルトファイルの場所を受け入れます。アカウントのパスワードとは別のパスフレーズを入力し、再度入力して確認します。



ヒント

パスフレーズは、ユーザーの認証に使用される単語と文字の文字列です。パスフレーズは、パスフレーズのスペースまたはタブを使用できるパスワードとは異なります。通常、パスフレーズは単一の単語ではなくフレーズであるため、通常はパスワードよりも長い時間がかかります。

公開鍵は ~/.ssh/id_dsa.pub に書き込まれます。秘密鍵は ~/.ssh/id_dsa に書き込まれます。秘密鍵には決して付与しないことが重要です。

2.
以下のコマンドを使用して、.ssh ディレクトリーのパーミッションを変更します。

```
chmod 755 ~/.ssh
```

3.
~/.ssh/id_dsa.pub の内容を、接続するマシンの ~/.ssh/authorized_keys ファイルにコピーします。~/.ssh/authorized_keys ファイルが存在する場合は、~/.ssh/id_dsa.pub ファイルの内容を他のマシンの ~/.ssh/authorized_keys ファイルに追加します。

4.
以下のコマンドを使用して authorized_keys ファイルのパーミッションを変更します。

```
chmod 644 ~/.ssh/authorized_keys
```

5.
GTK2+ ライブラリーがインストールされている GNOME またはグラフィカルデスクトップ環境を実行している場合は、「GUI を使用した ssh-agent の設定」 に進みます。X Window System を実行していない場合は、「ssh-agentの設定」 に進みます。

20.7.3.3. バージョン 1.3 および 1.5 の RSA 鍵ペアの生成

SSH プロトコルのバージョン 1 で使用される RSA 鍵ペアを生成するには、以下の手順に従います。DSA を使用するシステム間のみを接続する場合は、RSA バージョン 1.3 または RSA バージョン 1.5 のキーペアは必要ありません。

1.
RSA （バージョン 1.3 および 1.5 プロトコル用）のキーペアを生成するには、シェルプロンプトで以下のコマンドを入力します。

```
ssh-keygen -t rsa1
```

デフォルトのファイルの場所(~/.ssh/identity)を受け入れます。アカウントパスワードとは異なるパスフレーズを入力します。パスフレーズを再度入力して確認します。

公開鍵は ~/.ssh/identity.pub に書き込まれます。秘密鍵は ~/.ssh/identity に書き込まれます。秘密鍵には付与しないでください。

2. chmod 755 ~/.ssh コマンドおよび chmod 644 ~/.ssh/identity.pub コマンドを使用して、.ssh ディレクトリーおよびキーのパーミッションを変更します。

3. ~/.ssh/identity.pub の内容を、接続するマシンの ~/.ssh/authorized_keys ファイルにコピーします。~/.ssh/authorized_keys ファイルが存在しない場合は、~/.ssh/identity.pub ファイルをリモートマシンのファイル ~/.ssh/authorized_keys にコピーします。

4. GNOME を実行している場合は、「GUI を使用した ssh-agent の設定」 に進みます。GNOME を実行していない場合は、「ssh-agentの設定」 に進みます。

20.7.3.4. GUI を使用した ssh-agent の設定

ssh-agent ユーティリティーを使用してパスフレーズを保存することで、ssh または scp の接続を開始するたびに入力する必要がないようにできます。GNOME を使用している場合は、gnome-ssh-askpass パッケージには、GNOME にログインする際にパスフレーズの入力を要求するために使用されるアプリケーションが含まれ、GNOME にログインするまで保存します。GNOME セッション中に作成された ssh または scp 接続のパスワードやパスフレーズを入力する必要はありません。GNOME を使用していない場合は、「ssh-agentの設定」 を参照してください。

GNOME セッション中にパスフレーズを保存するには、以下の手順に従います。

1. パッケージ gnome-ssh-askpass をインストールする必要があります。rpm -q openssh-askpass コマンドを使用して、インストールされているかどうかを判断できます。インストールされていない場合は、Red Hat Enterprise Linux CD-ROM セット、Red Hat FTP ミラーサイト、または Red Hat Network を使用してインストールします。

2. Main Menu Button （パネル上）> Preferences > More Preferences > Sessions の順に選択し、Startup Programs タブをクリックします。Add をクリックして、Startup Command テキストエリアに /usr/bin/ssh-add と入力します。最後に実行されるように、既存のコマンドよりも高い数値に優先度を設定します。ssh-add の適切な優先度番号は 70 以上です。優先度

番号が高いほど優先度は低くなります。他のプログラムの一覧を表示している場合は、優先度が最も低いはずです。Close をクリックしてプログラムを終了します。

3.
GNOME からログアウトしてからログインし直します。つまり、X を再起動します。GNOME が起動すると、ダイアログボックスが表示され、パスフレーズの入力が求められます。要求されたパスフレーズを入力します。DSA キーペアと RSA キーペアの両方が設定されている場合は、両方の入力を求められます。この時点から、ssh、scp、または sftp によるパスワードの入力を要求されることはありません。

20.7.3.5. ssh-agentの設定

ssh-agent は、ssh または scp の接続時に毎回入力する必要がないように、パスフレーズを保存するために使用できます。X Window System を実行していない場合は、シェルプロンプトから以下の手順を実行します。GNOME を実行しているが、ログイン時にパスフレーズの入力を求めるよう設定しない場合は(「GUI を使用した ssh-agent の設定」を参照)、この手順は XTerm などのターミナルウィンドウで機能します。X を実行しているが GNOME ではない場合、この手順はターミナルウィンドウで動作します。ただし、パスフレーズはそのターミナルウィンドウにのみ記憶され、グローバル設定ではありません。

1.
シェルプロンプトで、以下のコマンドを入力します。

```
exec /usr/bin/ssh-agent $SHELL
```

2.
次に、コマンドを入力します。

```
ssh-add
```

パスフレーズを入力します。複数のキーペアが設定されている場合は、それぞれにプロンプトが表示されます。

3.
ログアウトすると、パスフレーズは忘れられます。仮想コンソールにログインするか、またはターミナルウィンドウを開くたびに、これらの 2 つのコマンドを実行する必要があります。

20.8. 関連情報

OpenSSH および OpenSSL プロジェクトは常に開発中であり、最新の情報は Web サイトから入手できます。OpenSSH および OpenSSL ツールの man ページも、詳細な情報源です。

20.8.1. インストールされているドキュメント

- *man ページの ssh、scp、sftp、sshd、および ssh-keygen には、これらのコマンドの使用方法と、そのコマンドで使用できるすべてのパラメーターに関する情報が記載されています。*

20.8.2. 便利な Web サイト

- *http://www.openssh.com/ - OpenSSH の FAQ ページ、バグレポート、メーリングリスト、プロジェクトの目標、およびセキュリティー機能のより詳細な説明があります。*

- *http://www.openssl.org/: OpenSSL FAQ ページ、メーリングリスト、およびプロジェクトゴールの説明。*

- *http://www.freesshd.com/: 他のプラットフォーム用の SSH クライアントソフトウェアです。*

---

[4] *X11 は、従来は X Window System または X と呼ばれる X11R7 ウィンドウ表示システムを指します。Red Hat Enterprise Linux には、オープンソースの X Window System である X11R7 が含まれています。*

[5] *DNS ポイズニングは、侵入者が DNS サーバーをクラッキングし、クライアントシステムを悪意のある重複したホストを参照したときに発生します。*

[6] *IP スプーフィングは、侵入者がネットワーク上の信頼できるホストから誤って表示されるネットワークパケットを送信すると発生します。*

[7] *多重接続は、共有されている共通のメディアで送信されるいくつかのシグナルで設定されます。SSH により、異なるチャンネルが共通のセキュアな接続で送信されます。*

第21章 NETWORK FILE SYSTEM (NFS)

ネットワークファイル システム(NFS)により、リモートホストはネットワーク経由でファイルシステムをマウントし、そのファイルシステムをローカルにマウントされているかのように対話できます。また、システム管理者は、リソースをネットワーク上の中央サーバーに統合することができるようになります。

この章では、基本的な NFS の概念と補足的な情報に焦点を絞って説明します。

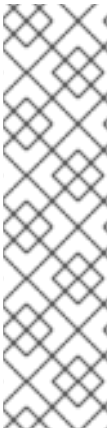
21.1. 仕組み

現在、NFS には 3 つのバージョンがあります。NFS バージョン 2 (NFSv2)は古く、広くサポートされています。NFS バージョン 3 (NFSv3)には、64 ビットファイルハンドル、Safe Async 書き込み、より堅牢なエラー処理など、より多くの機能があります。NFS バージョン 4 (NFSv4)はファイアウォールやインターネットを介して動作し、portmapper を必要とせず、ACL に対応し、ステートフルな操作を利用します。Red Hat Enterprise Linux は NFSv2、NFSv3、および NFSv4 クライアントをサポートしており、NFS 経由でファイルシステムをマウントする場合、Red Hat Enterprise Linux はデフォルトで NFSv3 を使用します（サーバーが対応している場合）。

NFS のすべてのバージョンは、IP ネットワーク上で実行中のTCP(Transmission Control Protocol)を使用することができます。この場合、NFSv4 ではこれが必要です。NFSv2 および NFSv3 では、IP ネットワークで実行している User Datagram Protocol (UDP)を使用して、クライアントとサーバー間のステートレスなネットワーク接続を提供できます。

UDP で NFSv2 または NFSv3 を使用する場合、通常の条件下でステートレスな UDP 接続では、TCP よりもプロトコルのオーバーヘッドが少なく、非常にクリーンで調整されていないネットワークでより優れたパフォーマンスに変換できます。NFS サーバーは、クライアントが共有ボリュームにアクセスすることを許可されている後に、ファイルハンドルをクライアントに送信します。このファイルハンドルはサーバー側に保存される不透明なオブジェクトであり、クライアントからの RPC 要求と共に渡されます。NFS サーバーはクライアントに影響を与えずに再起動でき、クッキーはそのまま残ります。ただし、UDP はステートレスのため、予期しないサーバーダウンなどが発生すると、UDP クライアントはサーバーの要求でネットワークを飽和させ続けます。このため、NFS サーバーへの接続時に TCP が推奨されるプロトコルになります。

プロトコルサポートは v4 プロトコルに組み込まれているため、NFSv4 では portmap、rpc.lockd、および rpc.statd デーモンとの対話はありません。NFSv4 は、既知の TCP ポート 2049 をリッスンし、ポート マップ との対話が不要になります。マウントプロトコルおよびロックプロトコルが V4 プロトコルに組み込まれ、rpc.lockd および rpc.statd との対話が不要になりました。rpc.mountd デーモンは依然としてサーバーで必要ですが、ネットワーク上の操作には関与しません。



注記

TCP は、Red Hat Enterprise Linux における NFS のデフォルトのトランスポートプロトコルです。UDP は互換性に必要となる場合は使用できますが、その使用範囲についてはできるだけ限定することを推奨しています。

すべての RPC/NFS デーモンには、ポートを設定する -p コマンドラインオプションがあり、ファイアウォールの設定が容易になります。

クライアントが TCP ラッパーによるアクセスが付与されると、NFS サーバーは設定ファイル /etc/exports を参照して、クライアントがエクスポートしたファイルシステムにアクセスできるかどうかを判断します。アクセスが許可されると、ユーザーはファイルおよびディレクトリーの全操作を使用できます。

重要

NFS がファイアウォールを有効にして Red Hat Enterprise Linux のデフォルトインストールと連携するには、デフォルトの TCP ポート 2049 の IPTables を設定する必要があります。IPTables を正しく設定しないと、NFS が正しく機能しません。

NFS の初期化スクリプトおよび rpc.nfsd プロセスでは、システム起動中の指定ポートへのバインドが可能になりました。ただし、ポートが利用できない場合や、別のデーモンと競合すると、エラーが発生しやすくなります。

21.1.1. 必要なサービス

Red Hat Enterprise Linux は、カーネルレベルのサポートとデーモンプロセスの組み合わせを使用して、NFS ファイル共有を提供します。すべての NFS バージョンは、クライアントとサーバー間のRPC(Remote Procedure Call)に依存します。Linux の RPC サービスは、portmap サービスによって制御されます。NFS ファイルシステムの共有またはマウントには、実装されている NFS のバージョンに応じて、以下のサービスが連携します。

- *nfs -(/sbin/service nfs start)は、NFS サーバーと、共有 NFS ファイルシステムの要求を処理する適切な RPC プロセスを起動します。*
- *nfslock -(/sbin/service nfslock start)は、適切な RPC プロセスを開始して、NFS クライアントがサーバー上のファイルをロックできるようにする必須のサービスです。*
- *portmap: ローカルの RPC サービスからのポート予約を受け入れます。その後、これらの*

ポートは公開され、対応するリモートの RPC サービスはそれらにアクセスします。Port map は RPC サービスの要求に応答し、要求された RPC サービスへの接続を設定します。

以下の RPC プロセスは、NFS サービスを容易にします。

- rpc.mountd: このプロセスは NFS クライアントからマウント要求を受け取り、要求されたファイルシステムが現在エクスポートされていることを確認します。このプロセスは、nfs サービスにより自動的に起動されるため、ユーザー設定は必要ありません。

- rpc.nfsd: サーバーが公開している明示的な NFS バージョンとプロトコルを定義できます。NFS クライアントが接続するたびにサーバースレッドを提供するなど、NFS クライアントの動的な要求に対応するため、Linux カーネルと連携して動作します。このプロセスは、nfs サービスに対応します。

- rpc.lockd: NFS クライアントがサーバー上のファイルをロックできるようにします。rpc.lockd が起動しないと、ファイルのロックに失敗します。rpc.lockd は Network Lock Manager (NLM) プロトコルを実装します。このプロセスは、nfslock サービスに対応します。このプロセスは NFSv4 では使用されません。

- rpc.statd: このプロセスは、Network Status Monitor (NSM) RPC プロトコルを実装します。このプロトコルは、NFS サーバーが正常に停止せずに再起動すると NFS クライアントに通知します。このプロセスは、nfslock サービスにより自動的に起動されるため、ユーザー設定は必要ありません。このプロセスは NFSv4 では使用されません。

- rpc.rquotad: このプロセスは、リモートユーザーのユーザークォータ情報を提供します。このプロセスは、nfs サービスにより自動的に起動されるため、ユーザー設定は必要ありません。

- rpc.idmapd: このプロセスは、ネットワーク上の NFSv4 名(user@domain 形式の文字列)とローカル UID と GID の間のマッピングを行う NFSv4 クライアントおよびサーバーアップコールを提供します。idmapd が NFSv4 で機能するには、/etc/idmapd.conf を設定する必要があります。このサービスは、NFSv4 での使用に必要です。

システムで NFS を使用するには、nfs-utils パッケージ、nfs-utils -lib パッケージ、および portmap パッケージがインストールされていることを確認します。

21.2. NFS クライアント設定

NFS 共有は、mount コマンドを使用してクライアント側に マウント されます。コマンドの形式は

以下のとおりです。

```
mount -t <nfs-type> -o <options> <host>:</remote/export> </local/directory>
```

&lt ;nfs-type> を、NFSv2 または NFSv3 サーバーの場合は nfs、NFSv4 サーバーの場合は nfs4 のいずれかに置き換えます。&lt ;options> を、NFS ファイルシステムのオプションのコンマ区切りリストに置き換えます（詳細は、「一般的な NFS マウントオプション」 を参照してください）。&lt ;host > をリモートホスト </ remote/export > がマウントされているリモートディレクトリーに置き換え、</ local/directory > を、リモートファイルシステムがマウントされるローカルディレクトリーに置き換えます。

詳細は mount の man ページを参照してください。

mount コマンドを手動で実行して NFS 共有にアクセスする場合は、システムを再起動してファイルシステムを手動で再マウントする必要があります。Red Hat Enterprise Linux では、システムの起動時にリモートファイルシステムを自動的にマウントする方法を 2 つ(/etc/fstab ファイルまたは autofs サービス)提供します。

21.2.1. /etc/fstab を使用した NFS ファイルシステムのマウント

別のマシンから NFS 共有をマウントする別の方法は、/etc/fstab ファイルに行を追加することです。/etc/fstab ファイルは起動時に netfs サービスによって参照されるため、NFS 共有を参照する行は、起動プロセス中に手動で mount コマンドを入力する場合と同じ効果が得られます。このファイルの各行には、NFS サーバーのホスト名、エクスポートされるサーバーの ディレクトリー、および NFS 共有がマウントされるローカルマシンの ディレクトリーを指定する必要があります。/etc/fstab ファイルを変更するには、root でなければなりません。

/etc/fstab の行の一般的な構文は以下のとおりです。

```
<server>:</remote/export> </local/directory> <nfs-type> <options> 0 0
```

&lt ;server& gt; を、ファイルシステムをエクスポートするサーバーのホスト名、IP アドレス、または完全修飾ドメイン名に置き換えます。&lt ;/remote/export > をエクスポートされたディレクトリーのパスに置き換え、</ local/directory > を、エクスポートしたディレクトリーがマウントされるローカルファイルシステムに置き換えます。&lt ;nfs-type> を、NFSv2 または NFSv3 サーバーの場合は nfs、NFSv4 サーバーの場合は nfs4 のいずれかに置き換えます。最後に、&lt ;options> を NFS ファイルシステムのオプションのコンマ区切りリストに置き換えます（詳細は、「一般的な NFS マウントオプション」 を参照してください）。マウントポイントは、/etc/fstab が読み取られる前に存在しておく必要があります。存在していないとマウントに失敗します。

以下は、NFS エクスポートをマウントする /etc/fstab 行の例です。

```
server:/usr/local/pub    /pub   nfs    defaults 0 0
```

クライアントシステムの /etc/fstab にこの行を追加した後、シェルプロンプトでコマンド mount /pub を入力し、マウントポイント /pub をサーバーからマウントします。このコマンドを実行する前に、マウントポイント /pub がクライアントマシンに存在している必要があります。

/etc/fstab 設定ファイルとそのコンテンツの詳細は、fstab の man ページを参照してください。

21.3. AUTOFS

/etc/fstab を使用する場合の欠点の 1 つは、NFS マウントされたファイルシステムにユーザーがアクセスする頻度に関わらず、マウントされたファイルシステムを所定の場所で維持するために、システムがリソースを割り当てる必要があることです。これは 1 つまたは 2 つのマウントでは問題になりませんが、システムが一度に多くのシステムへのマウントを維持している場合、システム全体のパフォーマンスに影響を与える可能性があります。/etc/fstab の代替は、カーネルベースの automount ユーティリティーを使用することです。自動マウント機能は 2 つのコンポーネントで設定されます。1 つはファイルシステムを実装するカーネルモジュールで、もう 1 つは他のすべての機能を実行するユーザー空間デーモンです。automount ユーティリティーは、（オンデマンドでマウント）NFS ファイルシステムを自動的にマウントおよびアンマウントできるため、システムリソースを節約できます。automount ユーティリティーを使用すると、AFS、SMBFS、CIFS、およびローカルファイルシステムなどの他のファイルシステムをマウントできます。

autofs は、デフォルトのプライマリー設定ファイルとして /etc/auto.master (マスターマップ) を使用します。これは、Name Service Switch メカニズムとともに autofs 設定(/etc/sysconfig/autofs内) を使用して別のネットワークソースと名前を使用するように変更できます。バージョン 4 デーモンのインスタンスはマスターマップに設定された各マウントポイントに対して実行されるため、指定のマウントポイントに対してコマンドラインから手動で実行できました。バージョン 5 では、設定されたすべてのマウントポイントの管理に単一のデーモンが使用されるため、すべての自動マウントをマスターマップで設定する必要があります。これは、他の業界標準の自動マウント機能の通常の要件と一致しています。マウントポイント、ホスト名、エクスポートしたディレクトリー、および各種オプションは各ホストに対して手動で設定するのではなく、すべて 1 つのファイルセット (またはサポートされている別のネットワークソース) 内に指定することができます。このサービスを使用する場合は、autofs 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。

21.3.1. autofs バージョン 5 の新機能

ダイレクトマップのサポート

autofs ダイレクトマップは、ファイルシステム階層の任意の時点でファイルシステムを自動的にマウントするメカニズムを提供します。ダイレクトマップは、マスターマップの /- のマウントポイントによって示されます。ダイレクトマップのエントリーには、(間接マップで使用される相対パス名の代わりに) 絶対パス名がキーとして含まれています。

レイジーマウントとアンマウントのサポート

マルチマウントマップエントリーは、1 つのキーの下にあるマウントポイントの階層を記述します。この良い例として、-hosts マップがあります。これは通常、/net/<host> 下のホストからのすべてのエクスポートをマルチマウントマップエントリーとして自動マウントするために使用されます。"-hosts"マップを使用すると、'ls' of "/net/<host> " は autofs トリガーを < host > から各エクスポートのマウントをマウントし、マウントしてアクセスする際に期限切れにします。これにより、エクスポートが多数あるサーバーにアクセスする際に必要なアクティブなマウントの数を大幅に減らすことができます。

強化された LDAP サポート

autofs バージョン 5 での Lightweight Directory Access Protocol (LDAP)のサポートが、autofs バージョン 4 に関する複数の方法で強化されました。autofs 設定ファイル (/etc/sysconfig/autofs)は、サイトが実装する autofs スキーマを指定するメカニズムを提供します。そのため、アプリケーション自体でトライアルとエラーでこれを判断する必要がなくなります。さらに、共通の LDAP サーバー実装でサポートされるほとんどのメカニズムを使用して、LDAP サーバーへの認証済みバインドがサポートされるようになりました。このサポートには、新しい設定ファイル (/etc/autofs_ldap_auth.conf) が追加されました。デフォルトの設定ファイルは自己文書化されており、XML 形式を使用します。

Name Service Switch (nsswitch) 設定の適切な使用

Name Service Switch 設定ファイルは、特定の設定データがどこから来るのかを判別する手段を提供するために存在します。この設定の理由は、データにアクセスするための統一されたソフトウェアインターフェイスを維持しながら、管理者が最適なバックエンドデータベースを柔軟に使用できるようにするためです。バージョン 4 の自動マウント機能は、ネームサービスのスイッチ設定の処理にますます厳しくなっていますが、まだ完了していません。一方、autofs バージョン 5 は完全な実装です。このファイルのサポートされる構文の詳細は、nsswitch.conf の man ページを参照してください。すべての nss データベースが有効なマップソースであるわけではなく、パーサーは無効なデータベースを拒否することに注意してください。有効なソースは、ファイル、yp、nis、nisplus、ldap および hesiod です。

autofs マウントポイントごとの複数のマスターマップエントリー

頻繁に使用されますが、まだ記述されていないのは、ダイレクトマウントポイント/-の複数のマスターマップエントリーの処理です。各エントリーのマップキーはマージされ、1 つのマップとして機能します。

以下は、ダイレクトマウントの connectathon テストマップの例です。

```
/- /tmp/auto_dcthon
/- /tmp/auto_test3_direct
/- /tmp/auto_test4_direct
```

21.3.2. autofs 設定

自動マウント機能の主な設定ファイルは /etc/auto.master です。マスターマップとも呼ばれます。マスターマップは、上記の紹介セクションで説明されているように変更される可能性があります。マスターマップには、システム上の autofs 制御マウントポイントと、それに対応する設定ファイルまたは自動マウントマップと呼ばれるネットワークソースが一覧表示されます。マスターマップの形式は次のとおりです。

```
<mount-point> <map-name> <options>
```

ここでは、以下のようになります。

- mount-point は、/home などの autofs マウントポイントです。
- map-name は、マウントポイントの一覧と、マウントポイントがマウントされるファイルシステムの場所を含むマップソースの名前です。マップエントリーの構文を以下に説明します。
- オプション は、それ自体にオプションが指定されていない限り、指定のマップのすべてのエントリーに適用されます。この動作は、累積するオプションがある autofs バージョン 4 とは異なります。これは、混合環境互換性の主な目的を達成するために変更されました。

以下は、/etc/auto.master ファイルの例です。

```
~]$ cat /etc/auto.master
/home /etc/auto.misc
```

マップの一般的な形式はそのマスターマップと同じですが、マスターマップではエントリーの末尾に表示されるオプション (options) がマウントポイント (mount-point) と場所 (location) の間に表示されます。

```
<mount-point>  [<options>]  <location>
```

ここでは、以下のようになります。

- *<mount-point& gt; は autofs マウントポイントです。これは 1 つのインダイレクトマウント用の 1 つのディレクトリー名にすることも、複数のダイレクトマウント用のマウントポイントの完全パスにすることもできます。ダイレクトマップと間接マップの各エントリーキー（上記の <mount-point >）の後に、スペースで区切られたオフセットディレクトリー（それぞれ/で始まるサブディレクトリー名）のリストが続き、ミューリマウントエントリーと呼ばれるものが続きます。*

- *<options> が指定されている場合は、独自のオプションを指定しないマップエントリーのマウントオプションです。*

- *<location> は、ローカルファイルシステムのパス(Sun マップ形式のエスケープ文字 : が先頭に付き、マップ名が / で始まります)、NFS ファイルシステム、その他の有効なファイルシステムの場所などのファイルシステムの場所です。*

マップファイルの例を以下に示します。

```
~]$ cat /etc/auto.misc
payroll -fstype=nfs personnel:/dev/hda3
sales -fstype=ext3 :/dev/hda4
```

マップファイルの最初の列は、autofs マウントポイント(personnel と呼ばれるサーバーのsales および payroll)を示し ます。2 番目のコラムは autofs マウントのオプションを示し、3 番目のコラムはマウントのソースを示しています。任意の設定に基づき、autofs マウントポイントは、/home/payroll と /home/sales になります。-fstype= オプションは省略されることが多く、通常は正しい操作には必要ありません。

ディレクトリーが存在しない場合、自動マウント機能はディレクトリーを作成します。ディレクトリーが存在している状況で自動マウント機能が起動した場合は、自動マウント機能の終了時にディレクトリーが削除されることはありません。以下のコマンドを実行して、自動マウントデーモンを起動または再起動できます。

```
service autofs start
```

または

```
service autofs restart
```

上記の設定を使用して、プロセスが /home/payroll/2006/July.sxc などのアンマウントされた autofs ディレクトリーにアクセスする必要がある場合、自動マウントデーモンはディレクトリーを自動的にマウントします。タイムアウトを指定した場合は、タイムアウト期間中ディレクトリーにアクセスしないと、ディレクトリーが自動的にアンマウントされます。

ターミナルで以下のコマンドを実行し、自動マウントデーモンのステータスを表示できます。

```
/sbin/service/autofs status
```

21.3.3. autofs の一般的なタスク

21.3.3.1. サイト設定ファイルの上書きまたは拡張

クライアントシステム上の特定マウントポイントのサイトデフォルト値を無効にする場合に便利です。たとえば、自動マウント機能マップが NIS に保存され、/etc/nsswitch.conf ファイルに以下のディレクティブがあるとします。

```
automount:  files nis
```

NIS の auto.master マップファイルには以下が含まれています。

```
/home auto.home
```

また、NIS の auto.home マップに以下が含まれていることを前提としています。

```
beth     fileserver.example.com:/export/home/beth
joe      fileserver.example.com:/export/home/joe
*        fileserver.example.com:/export/home/&
```

また、ファイルマップ /etc/auto.home は存在しません。

上記の例では、クライアントシステムが別のサーバーからホームディレクトリーをマウントする必要があると仮定します。この場合、クライアントは以下の /etc/auto.master マップを使用する必要があります。

```
/home /etc/auto.home2
+auto.master
```

また、/etc/auto.home2 マップにはエントリーが含まれます。

```
*   labserver.example.com:/export/home/&
```

マウントポイントの最初の出現のみが処理されるため、/home には NIS auto.home マップではなく /etc/auto.home2 の内容が含まれます。

または、サイト全体の auto.home マップをいくつかのエントリーで拡張する場合は、/etc/auto.home ファイルマップを作成して、新しいエントリーと最後に NIS auto.home マップを追加します。次に、/etc/auto.home ファイルマップは以下のようになります。

```
mydir someserver:/export/mydir
+auto.home
```

上記の NIS の auto.home マップを指定すると、/home の ls により以下が提供されます。

```
~]$ ls /home
beth  joe  mydir
```

autofs は、読み取り中のファイルマップと同じ名前のファイルマップの内容を含まないことを認識しているため、この最後の例は期待どおりに動作します。したがって、nsswitch 設定の次のマップソースに移動します。

21.3.3.2. LDAP を使用した自動マウント機能マップの格納

LDAP から自動マウント機能マップを取得するすべてのシステムに LDAP クライアントライブラリーをインストールする必要があります。RHEL 5 では、openldap パッケージは自動マウント機能 の依存関係として自動的にインストールされ ます。LDAP アクセスを設定する際は /etc/openldap/ldap.conf ファイルを編集します。BASE と URI がサイトに適切に設定されていることを確認します。スキーマが設定に設定されていることも確認してください。

自動マウント機能のマップを LDAP に格納するために既定された最新のスキーマが rfc2307bis に記載されています。このスキーマを使用するには、スキーマ定義からコメント文字を削除して、autofs 設定 (/etc/sysconfig/autofs) に設定する必要があります。以下に例を示します。

```
DEFAULT_MAP_OBJECT_CLASS="automountMap"
DEFAULT_ENTRY_OBJECT_CLASS="automount"
DEFAULT_MAP_ATTRIBUTE="automountMapName"
DEFAULT_ENTRY_ATTRIBUTE="automountKey"
DEFAULT_VALUE_ATTRIBUTE="automountInformation"
```

設定内でコメントされていないスキーマエントリーが上記だけであることを確認します。また、automountKey は rfc2307bis スキーマの cn 属性に代わることに注意してください。サンプル設定の LDIF について以下に説明します。

```
# extended LDIF
#
# LDAPv3
# base <> with scope subtree
# filter: (&(objectclass=automountMap)(automountMapName=auto.master))
# requesting: ALL
#

# auto.master, example.com
dn: automountMapName=auto.master,dc=example,dc=com
objectClass: top
objectClass: automountMap
automountMapName: auto.master

# extended LDIF
#
# LDAPv3
# base <automountMapName=auto.master,dc=example,dc=com> with scope subtree
# filter: (objectclass=automount)
# requesting: ALL
#

# /home, auto.master, example.com
dn: automountMapName=auto.master,dc=example,dc=com
objectClass: automount
cn: /home

automountKey: /home
automountInformation: auto.home

# extended LDIF
#
# LDAPv3
# base <> with scope subtree
# filter: (&(objectclass=automountMap)(automountMapName=auto.home))
# requesting: ALL
#

# auto.home, example.com
dn: automountMapName=auto.home,dc=example,dc=com
objectClass: automountMap
automountMapName: auto.home

# extended LDIF
#
# LDAPv3
# base <automountMapName=auto.home,dc=example,dc=com> with scope subtree
# filter: (objectclass=automount)
# requesting: ALL
#
```

```
# foo, auto.home, example.com
dn: automountKey=foo,automountMapName=auto.home,dc=example,dc=com
objectClass: automount
automountKey: foo
automountInformation: filer.example.com:/export/foo

# /, auto.home, example.com
dn: automountKey=/,automountMapName=auto.home,dc=example,dc=com
objectClass: automount
automountKey: /
automountInformation: filer.example.com:/export/&
```

21.3.3.3. Autofs v4 マップの Autofs v5 への適合

v4 マルチマップエントリー

autofs バージョン 4 では、マスターマップにマルチマップエントリーの概念が導入されました。マルチマップエントリーの形式は以下のとおりです。

```
<mount-point> <maptype1> <mapname1> <options1> -- <maptype2> <mapname2> <options2>
-- ...
```

このように任意の数のマップを 1 つのマップに統合できます。この機能は v5 では提供されなくなりました。これは、バージョン 5 が、同じ結果を含めるために使用できるマップが含まれているためです。以下のマルチマップの例を見てみましょう。

```
/home file /etc/auto.home -- nis auto.home
```

これは、v5 の以下の設定に置き換えることができます。

/etc/nsswitch.conf は以下を一覧表示する必要があります。

```
automount: files nis
```

/etc/auto.master には以下が含まれている必要があります。

```
/home  auto.home
```

/etc/auto.home には以下が含まれている必要があります。

```
<entries for the home directory>
+auto.home
```

これにより、/etc/auto.home からのエントリーと nis auto.home マップが組み合わされます。

複数マスターマップ

autofs バージョン 4 では、ファイル、nis、Hesiod、LDAP など、各ソースからマスターマップのコンテンツをマージできます。バージョン 4 の自動マウント機能は、/etc/nsswitch.conf に記載されている各ソースのマスターマップを検索します。マップは存在する場合は読み取られ、その内容が 1 つの大きな auto.master マップにマージされます。

バージョン 5 では、これは動作ではなくなりました。nsswitch.conf のソース一覧にある最初のマスターマップのみが参照されます。複数のマスターマップの内容をマージすることが望ましい場合は、含まれるマップを使用できます。以下の例を考慮してください。

```
/etc/nsswitch.conf:
automount: files nis
```

```
/etc/auto.master:
/home  /etc/auto.home
+auto.master
```

上記の設定は、ファイルベースの auto.master と NIS ベースの auto.master の内容を統合します。ただし、含まれるマップエントリーはファイルマップでのみ許可されるため、NIS auto.master と LDAP auto.master の両方を含める方法はありません。

この制限は、ソースに異なる名前を持つマスターマップを作成することで解決できます。上記の例では、auto.master.ldap という名前の LDAP マスターマップがある場合は、"+auto.master.ldap" をファイルベースのマスターマップに追加し、"ldap" が nsswitch 設定のソースとしてリストされている場合は、それも含まれます。

21.4. 一般的な NFS マウントオプション

リモートホストに NFS 経由でファイルシステムをマウントする以外に、マウント時に他のオプションを指定して、簡単に使用できるようになります。これらのオプションは、手動の mount コマンド、/etc/fstab 設定、autofs と併用できます。

以下に NFS マウントに一般的に使用されているオプションを示します。

-

hard または soft: NFS 接続を介してファイルを使用しているプログラムが、サーバーがオンラインに戻るのを停止および待機する(ハード)するか、エクスポートしたファイルシステムを提供するホストが利用できない場合、またはエラー(ソフト)を報告するかどうかを指定します。

hard を指定すると、intr オプションも指定されていない限り、NFS 通信が再開するプロセスを終了できません。

ソフト が指定されている場合、ユーザーは追加の timeo= <value > オプションを設定できます。ここで、< value > はエラーが報告されるまでの経過秒数を指定します。

注記

非常に輻輳したネットワークで I/O エラーを生成するか、非常にビジーなサーバーを使用する場合は、ソフトマウントを使用することは推奨されません。

- *intr - サーバーがダウンした場合やサーバーに到達できない場合に、NFS 要求が中断されます。*
- *nfsvers=2 または nfsvers=3: 使用する NFS プロトコルのバージョンを指定します。これは、複数の NFS サーバーを実行するホストに役立ちます。バージョンを指定しないと、NFS はカーネルおよび mount コマンドで対応している最新バージョンを使用します。このオプションは NFSv4 では対応していないため、使用しないでください。*
- *noacl - すべての ACL 処理をオフにします。古いバージョンの Red Hat Enterprise Linux、Red Hat Linux、Solaris と連動させる場合に必要となることがあります。こうした古いシステムには、最新の ACL テクノロジーに対する互換性がないためです。*
- *nolock - ファイルのロックを無効にします。この設定は、古い NFS サーバーに接続するときに必要になる場合があります。*
- *noexec - マウントされたファイルシステムでバイナリーの実行を行いません。これは、互換性のないバイナリーを含む NFS 経由で、Linux 以外のファイルシステムをマウントしている場合に便利です。*
- *nosuid - set-user-identifier ビットまたは set-group-identifier ビットを無効にします。これにより、リモートユーザーは、setuid プログラムを実行してより高い権限を取得できなくな*

ります。

- *port=num — NFS サーバーポートの数値を指定します。num が 0 （デフォルト）の場合、mount は、使用するポート番号のリモートホストの portmapper のクエリーを実行します。リモートホストの NFS デーモンがポートマッパーに登録されていない場合は、代わりに TCP 2049 の標準 NFS ポート番号が使用されます。*

- *rsize=num および wsize=num: この設定により、読み取り(rsize)および書き込み(wsize)の NFS 通信を高速化します。これにより、大きなデータブロックサイズをバイト単位で一度に転送することができます。古い Linux カーネルやネットワークカードの中には、ブロックサイズを大きくするとうまく動作しないものがあるので、これらの値を変更する場合は注意が必要です。NFSv2 または NFSv3 では、両方のパラメーターのデフォルト値が 8192 に設定されます。NFSv4 の場合、両方のパラメーターのデフォルト値は 32768 に設定されます。*

- *sec=mode: NFS 接続の認証時に使用するセキュリティーのタイプを指定します。*

  sec=sys はデフォルト設定で、AUTH_SYS を使用して NFS 操作を認証するのにローカルの UNIX UID および GID を使用します。

  sec=krb5 は、ユーザー認証に、ローカルの UNIX の UID と GID ではなく、Kerberos V5 を使用します。

  sec=krb5i は、ユーザー認証に Kerberos V5 を使用し、データの改ざんを防ぐ安全なチェックサムを使用して、NFS 操作の整合性チェックを行います。

  sec=krb5p は、ユーザー認証に Kerberos V5 を使用し、整合性チェックを実行し、トラフィックの傍受を防ぐため NFS トラフィックの暗号化を行います。これは最も安全な設定ですが、関連するパフォーマンスのオーバーヘッドも最も高くなります。

- *tcp: NFS マウントが TCP プロトコルを使用するように指定します。*

- *UDP: NFS マウントが UDP プロトコルを使用するように指定します。*

mount および nfs の man ページに、さらに多くのオプションが一覧表示されています。

21.5. NFS の開始と停止

NFS サーバーを実行するには、portmap サービスが実行されている必要があります。portmap がアクティブであることを確認するには、root で以下のコマンドを入力します。

```
service portmap status
```

portmap サービスが実行中の場合は、nfs サービスを開始できます。NFS サーバーを起動するには、root で以下を入力します。

```
service nfs start
```

注記

NFS クライアントとサーバーの両方が適切に機能するには、nfslock を起動する必要もあります。root タイプで NFS ロックを開始するには、/sbin/service nfslock start を実行します。NFS が起動時に開始するように設定されている場合は、chkconfig --list nfslock を実行して nfslock も起動するようにしてください。nfslock が の に設定されていない場合、これはコンピューターが起動するたびに /sbin/service nfslock の起動 を手動で実行する必要があることを意味します。システムの起動時に nfslock を自動的に起動するように設定するには、の端末 chkconfig nfslock に以下のコマンドを入力し ます。

サーバーを停止するには、root で以下を入力します。

```
service nfs stop
```

restart オプションは、NFS を停止して起動する簡単な方法です。これは、NFS の設定ファイルを編集した後に設定変更を有効にする最も効率的な方法です。

サーバーを再起動するには、root で以下を入力します。

```
service nfs restart
```

condrestart (conditional restart) オプションは、現在実行中の場合にのみ nfs を開始します。このオプションは、デーモンが実行されていない場合はデーモンを起動しないため、スクリプトに便利です。

条件付きでサーバーを再起動するには、root で以下を入力します。

```
service nfs condrestart
```

サービスを再起動せずに NFS サーバー設定ファイルを再読み込みするには、root で以下のコマンドを入力します。

```
service nfs reload
```

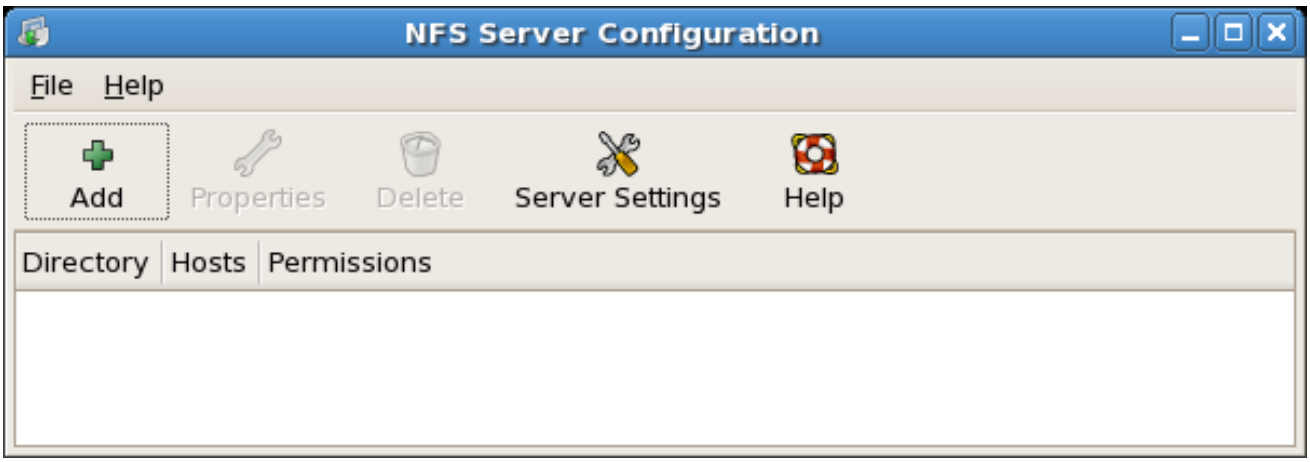
デフォルトでは、nfs サービスはシステムの起動時に自動的に起動し ません。システムの起動時に NFS が起動するように設定するには、/sbin/chkconfig、/usr/sbin/ntsysv、または Services Configuration Tool プログラムなどの initscript ユーティリティーを使用します。これらのツールの詳細は、18章 を参照してください。

21.6. NFS サーバーの設定

Red Hat Enterprise Linux で NFS サーバーを設定する方法は、3 つの方法があります。NFS サーバー設定ツール (system-config-nfs)、設定ファイルを手動で編集(/etc/exports)、または /usr/sbin/exportfs コマンドを使用します。

NFS Server Configuration Tool を使用するには、X Windows を実行し、root 権限を持ち、system-config-nfs RPM パッケージがインストールされている必要があります。アプリケーションを起動するには、System > Administration > Server Settings > NFS の順にクリックします。ターミナルでコマンド system-config-nfs を入力することもできます。NFS サーバー設定ツールのウィンドウを以下に示します。

図21.1 NFS サーバー設定ツール

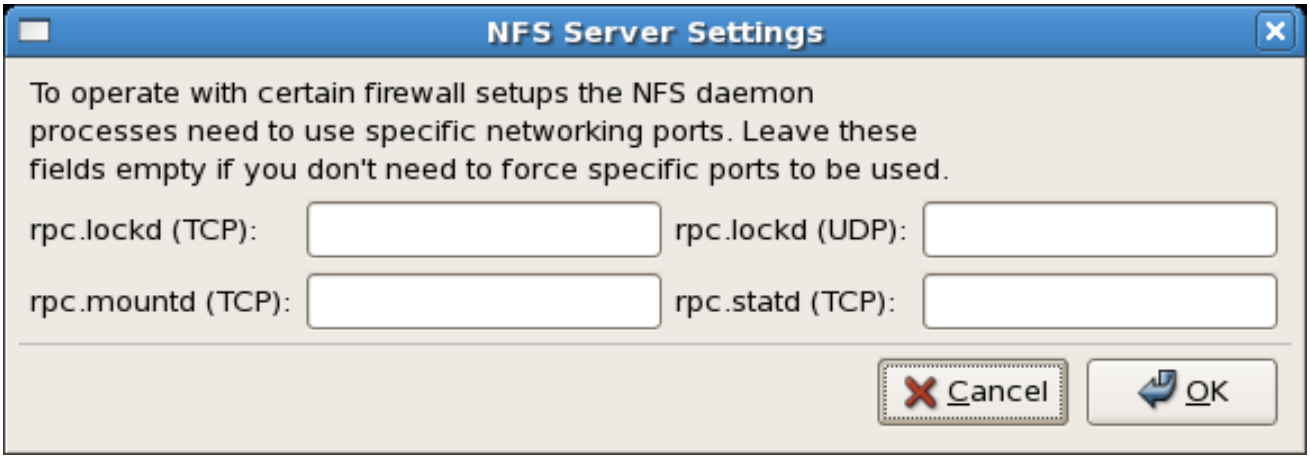


[D]

特定のファイアウォール設定に基づいて、特定のネットワークポートを使用するように NFS デーモ

ンプロセスを設定する必要がある場合があります。NFS サーバー設定では、portmapper によって割り当てられたランダムポートを使用する代わりに、各プロセスのポートを指定できます。Server Settings ボタンをクリックして、NFS Server の設定を設定できます。以下の図は、NFS サーバーの設定ウィンドウを示しています。

図21.2 NFS サーバーの設定



[D]

21.6.1. NFS ファイルシステムのエクスポートまたは共有

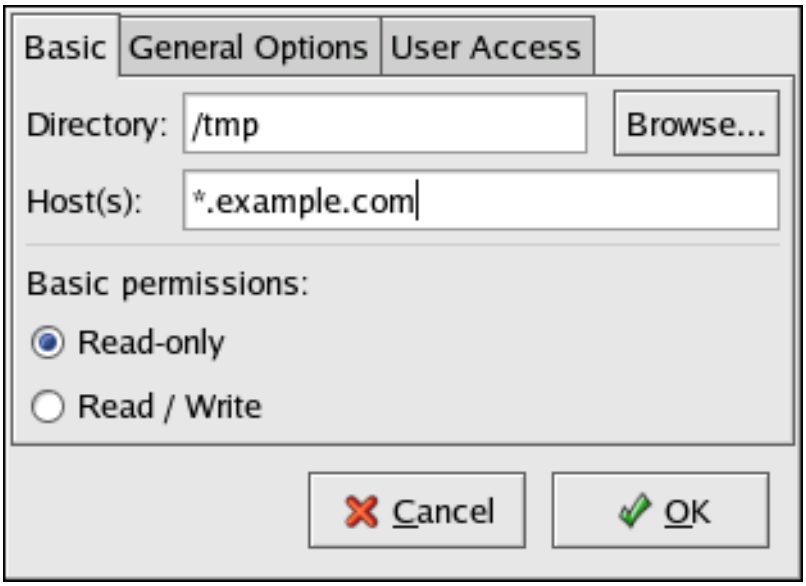
NFS サーバーからのファイルの共有または提供は、ディレクトリーのエクスポートと呼ばれます。NFS サーバー設定ツール を使用すると、システムを NFS サーバーとして設定できます。

NFS 共有を追加するには、Add ボタンをクリックします。図21.3「共有の追加」 に表示されるダイアログボックスが表示されます。

Basic タブには以下の情報が必要です。

- *directory: /tmp など、共有するディレクトリーを指定します。*
- *Host (s) - ディレクトリーを共有するホストを指定します。使用できる形式の説明は、「ホスト名の形式」 を参照してください。*
- *基本パーミッション - ディレクトリーに読み取り専用パーミッションと読み取り/書き込みパーミッションを付与するかどうかを指定します。*

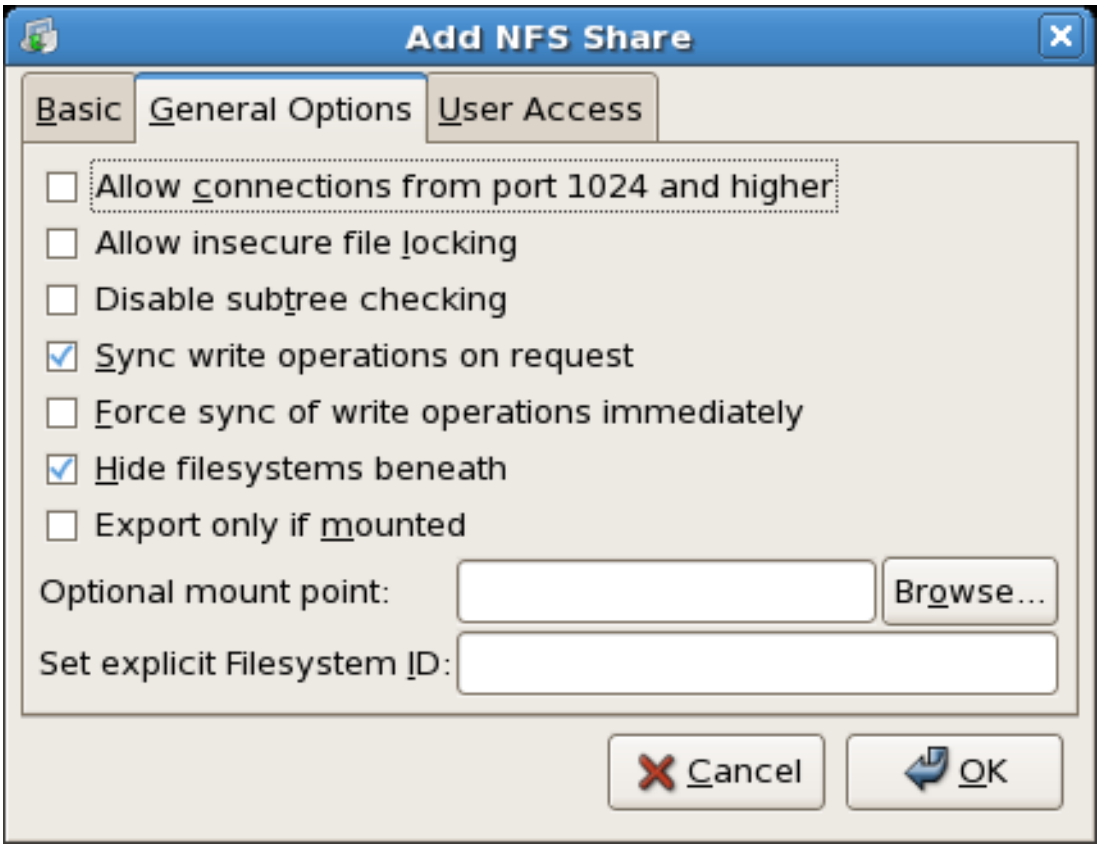
図21.3 共有の追加



[D]

General Options タブでは、以下のオプションを設定できます。

図21.4 NFS の一般的なオプション



[D]

- *ポート 1024 以上からの接続を許可する - 1024 未満のポート番号で起動するサービスは、root で起動する必要があります。このオプションを選択して、root 以外のユーザーが NFS*

サービスを起動できるようにします。このオプションは、セキュアでない に対応します。

- *Allow insecure file locking - ロック要求は必要ありません。このオプションは insecure_locks に対応します。*

- *Disable subtree checking - ファイルシステムのサブディレクトリーがエクスポートされていても、ファイルシステム全体がエクスポートされていない場合、サーバーは要求されたファイルがエクスポートされたサブディレクトリーにあるかどうかを確認します。このチェックは、サブツリーチェック と呼ばれます。このオプションを選択して、サブツリーのチェックを無効にします。ファイルシステム全体をエクスポートしている場合は、サブツリーチェックを無効にすると、転送率が増大する可能性があります。このオプションは no_subtree_check に対応します。*

- *Sync write operations on request - デフォルトでは有効になっており、このオプションでは、要求による変更がディスクに書き込まれる前に、サーバーが要求に応答することができません。このオプションは、同期 に対応します。これを選択しないと、async オプションが使用されます。*

  - *Reforce sync of write operations immediately - ディスクへの書き込みを遅延させないでください。このオプションは no_wdelay に対応します。*

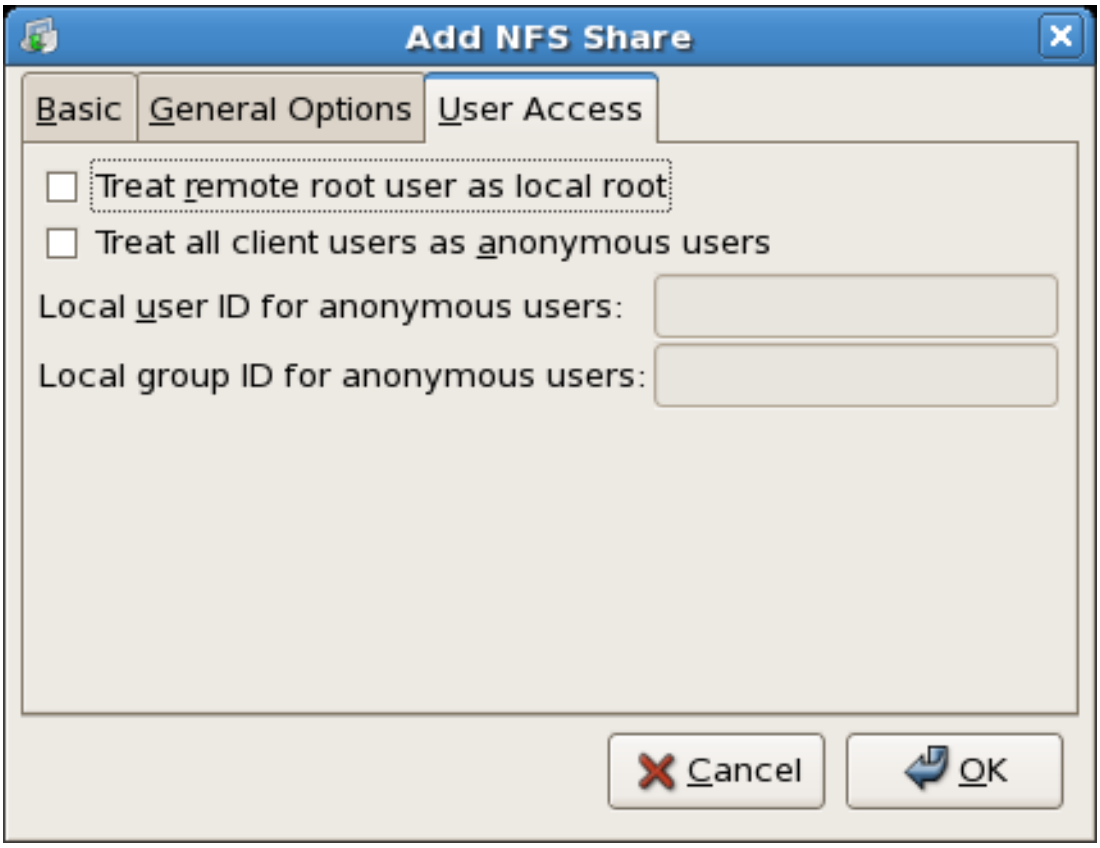
- *以下 のファイルシステムを非表示にする と、nohide オプションがオンまたはオフになります。nohide オプションをオフにすると、ネストされたディレクトリーが表示されます。したがって、クライアントは変更を通知せずに親からファイルシステムを介して移動できます。*

- *マウントされている場合にのみエクスポート すると、mountpoint オプションが設定され、マウントされている場合にのみディレクトリーをエクスポートできます。*

- *オプションのマウントポイント は、オプションのマウントポイントへのパスを指定します。Browse をクリックして、優先マウントポイントに移動するか、または既知のパスを入力します。*

- *明示的なファイルシステム ID を設定し ます。は fsid=X オプションを設定します。これは主にクラスター設定で使用されます。すべてのクラスターで一貫したファイルシステム ID を使用すると、古い NFS ファイル処理が回避されます。*

図21.5 NFS ユーザーアクセス



[D]

User Access タブでは、以下のオプションを設定できます。

- *リモートの root ユーザーをローカル root として扱います。デフォルトでは、root ユーザーのユーザーおよびグループ ID は両方とも 0 です。root squashing は、ユーザー ID 0 とグループ ID 0 を anonymous のユーザー ID とグループ ID にマッピングし、クライアントの root が NFS サーバーで root 権限を持たないようにします。このオプションを選択すると、root は匿名にマッピングされず、クライアントの root にはエクスポートされたディレクトリーへの root 権限があります。このオプションを選択すると、システムのセキュリティーが大幅に低下する可能性があります。絶対に必要な場合を除き、選択しないでください。このオプションは、no_root_squash に対応します。*

- *すべてのクライアントユーザーを匿名ユーザーと して扱います。このオプションを選択すると、すべてのユーザー ID とグループ ID が匿名ユーザーにマッピングされます。このオプションは、all_squash に対応します。*

  - *匿名ユーザー のローカルユーザー ID を指定します。匿名 ユーザーが選択されると、すべてのクライアントユーザー が選択されている場合は、匿名ユーザーのユーザー ID を指定できます。このオプションは anonuid に対応します。*

  -

匿名ユーザー のローカルグループ ID を指定します。匿名 ユーザー が選択されると、すべてのクライアントユーザー が選択されている場合は、匿名ユーザーのグループ ID を指定できます。このオプションは、ongid に対応します。

既存の NFS 共有を編集するには、一覧から共有を選択し、Properties ボタンをクリックします。既存の NFS 共有を削除するには、一覧から共有を選択し、Delete ボタンをクリックします。

OK をクリックしてリストから NFS 共有を追加、編集、または削除した後に、直ちに変更が行われます。サーバーデーモンは再起動し、古い設定ファイルは /etc/exports.bak として保存されます。新しい設定は /etc/exports に書き込まれます。

NFS サーバー設定ツール は、/etc/exports 設定ファイルに直接読み取りおよび書き込みします。そのため、ツールを使用して手動でファイルを変更できます。また、ファイルを手動で修正した後は、ツールを使用できます（ファイルが正しい構文で変更された場合）。

次のセクションでは、/etc/exports を手動で編集し、/usr/sbin/exportfs コマンドを使用して NFS ファイルシステムをエクスポートする方法を説明します。

21.6.2. コマンドラインからの設定

テキストエディターを使用して設定ファイルを編集したい場合や、X Window System がインストールされていない場合は、設定ファイルを直接変更できます。

/etc/exports ファイルは、NFS サーバーがエクスポートするディレクトリーを制御します。形式は以下のようになります。

```
directory hostname(options)
```

指定する必要がある唯一のオプションは sync または async のいずれかです(同期 が推奨されます)。sync が指定されている場合、サーバーは要求による変更がディスクに書き込まれる前に要求に応答しません。

以下に例を示します。

```
/misc/export speedy.example.com(sync)
```

これにより、ユーザーは speedy.example.com がデフォルトの読み取り専用パーミッションで /misc/export をマウントできます。

```
/misc/export speedy.example.com(rw,sync)
```

これにより、ユーザーが読み取り/書き込み権限 を持つ /misc/export をマウントできるよう になります。

可能なホスト名形式の説明は、「ホスト名の形式」 を参照してください。

注意

/etc/exports ファイルのスペースに注意してください。ホスト名とオプションの間のスペースが括弧でない場合、オプションはホスト名にのみ適用されます。ホスト名とオプションの間にスペースがある場合、オプションは残りの世界に適用されます。たとえば、以下の行を調べます。

```
/misc/export speedy.example.com(rw,sync) /misc/export
speedy.example.com (rw,sync)
```

最初の行は、speedy.example.com の読み取り/書き込みアクセスからユーザーに付与し、他のすべてのユーザーを拒否します。2 番目の行は、speedy.example.com の読み取り専用アクセス（デフォルト）からのユーザーに付与し、残りのユーザーに読み取り/書き込みアクセスを許可します。

/etc/exports を変更するたびに、以下のコマンドを実行して NFS デーモンに変更を通知したり、設定ファイルを再読み込みする必要があります。

```
service nfs reload
```

21.6.3. ファイアウォール背後での NFS の実行

NFS には portmap が必要です。これは、RPC サービスのポートを動的に割り当て、ファイアウォールルールの設定に問題が発生する可能性があるため、/etc/sysconfig/nfs 設定ファイルを編集して、必要な RPC サービスが実行されるポートを制御できます。NFS を許可するようにファイアウォールを設定する方法は、「/etc/sysconfig/nfs」 を参照してください。

21.6.4. ホスト名の形式

ホストは以下の形式にすることができます。

- 単一マシン：完全修飾ドメイン名（サーバーによって解決可能）、ホスト名（サーバーで解決可能）、または IP アドレス。
- ワイルドカードで指定された一連のマシン - * または？文字を使用して文字列の一致を指定します。ワイルドカードは IP アドレスでは使用しないことになっていますが、逆引き DNS ルックアップが失敗した場合には誤って動作する可能性があります。完全修飾ドメイン名でワイルドカードを指定する場合、ドット(.)はワイルドカードに含まれません。たとえば、*.example.com には one.example.com が含まれていますが one.two.example.com は含まれません。
- IP ネットワーク： a.b.c.d/z を使用します。ここで、a.b.c.d はネットワークであり、z はネットマスクのビット数です（例：0/24）。もう 1 つの使用可能な形式は a.b.c.d/netmask です。ここで、a.b.c.d はネットワークであり、netmask はネットマスクです（例：192.168.100.8/255.255.255.0）。
- netgroups - @group-name の形式で、group-name は NIS netgroup 名です。

21.7. /ETC/EXPORTS 設定ファイル

/etc/exports ファイルは、リモートホストにどのファイルシステムをエクスポートするかを制御し、オプションを指定します。空白行は無視され、ハッシュ記号(#)で行を開始することでコメントを作成できます。長い行はバックスラッシュ(\)でラップできます。エクスポートする各ファイルシステムは個別の行にする必要があり、エクスポートされたファイルシステムの後に配置された許可されたホストの一覧は、スペースで区切る必要があります。各ホストのオプションは、ホストの識別子の直後に括弧を追加し、その中に指定する。ホストと最初の括弧の間には空白を使用しない。有効なホストタイプは、gss/krb5、gss/krb5i、および gss/krb5p です。

エクスポートするファイルシステムの行の構造は次のとおりです。

```
<export> <host1>(<options>) <hostN>(<options>)...
```

この構造 **<export>** で をエクスポートされるディレクトリーに置き換え、**< host1 >** をエクスポート先のホストまたはネットワークに置き換え、**< options >** をそのホストまたはネットワークのオプションに置き換えます。追加のホストは、スペース区切りの一覧で指定できます。

以下の方法を使用して、ホスト名を指定できます。

- 単一ホスト: 特定のホストが完全修飾ドメイン名、ホスト名、または IP アドレスで指定されている場所。

- ワイルドカード - 特定の文字列に一致する完全修飾ドメイン名をグループ化するために * 文字または ? 文字を使用する場所。ワイルドカードは IP アドレスと併用しないでください。ただし、逆引き DNS ルックアップが失敗した場合、誤って動作する可能性があります。

  ワイルドカードを完全修飾ドメイン名で使用する場合は想定よりも正確である可能性があるため、注意してください。たとえば、ワイルドカードとして *.example.com を使用すると、sales.example.com はエクスポートされたファイルシステムにアクセスできますが、bob.sales.example.com はアクセスできません。* .example.com と *. *.example. com の両方に該当する必要があります。

- IP ネットワーク: 大規模なネットワーク内の IP アドレスに基づくホストの一致を許可します。たとえば、192.168.0. 0/28 では、最初の 16 の IP アドレスが 192.168.0.0 から 192.168.0.15 までで、エクスポートしたファイルシステムにアクセスできますが、192.168.0.16 以降にはアクセスできません。

- netgroups - 使用する NIS netgroup 名 @<group-name>書き込まれます。これにより、NIS サーバーが、このエクスポートしたファイルシステムのアクセス制御を担当することになります。この場合、/etc/exports に影響を及ぼさずに NIS グループでユーザーを追加および削除できます。

最も簡単な方法は、/etc/exports ファイルに、エクスポートするディレクトリーと、そのディレクトリーへのアクセスを許可するホストを指定するだけです。以下の例のようになります。

```
/exported/directory bob.example.com
```

この例では、bob.example.com は /exported/directory/ をマウントできます。この例ではオプションが指定されていないため、以下のデフォルトの NFS オプションが有効になります。

- ro: エクスポートしたファイルシステムのマウントは読み取り専用です。リモートホストは、ファイルシステムで共有されているデータに変更を加えることができません。ホストがファイルシステムに変更を加えることを許可するには、read/write (rw)オプションを指定する必要があります。

- wdelay: 別の書き込み要求が不満に疑われる場合は、NFS サーバーがディスクへの書き込みを遅らせます。これにより、書き込みコマンドによるディスクへのアクセス回数が減り、書

き込みオーバーヘッドを削減することでパフォーマンスが向上します。no_wdelay オプションはこの機能をオフにしますが、sync オプションを使用する場合にのみ利用できます。

- root_squash - リモートで接続した root ユーザーが root 権限を持たないようにし、ユーザー nfsnobody のユーザー ID を割り当てます。これにより、リモート root ユーザーの権限を最も低いローカルユーザーに効果的に拒否し、リモートサーバーのファイルの不正な変更を防ぎます。または、no_root_squash オプションは、root squashing をオフにします。root を含むすべてのリモートユーザーを非表示にするには、all_squash オプションを使用します。特定のホストからリモートユーザーで使用するユーザーおよびグループ ID を指定するには、anonuid オプションおよび anongid オプションを指定します。この場合、リモート NFS ユーザーが共有して指定 (anonuid=<uid-value>,anongid=<gid-value>) するために特別なユーザーアカウントを作成できます。<uid-value> is the user ID number and <gid-value> はグループ ID 番号です。

重要

デフォルトでは、アクセス制御リスト (ACL)は、Red Hat Enterprise Linux の NFS でサポートされています。この機能を無効にするには、ファイルシステムをエクスポートする際に no_acl オプションを指定します。

エクスポートするすべてのファイルシステムの各デフォルトは、明示的に上書きする必要があります。たとえば、rw オプションを指定しないと、エクスポートするファイルシステムが読み取り専用として共有されます。以下は、/etc/exports の例になりますが、ここでは 2 つのデフォルトオプションを上書きします。

```
/another/exported/directory 192.168.0.3(rw,sync)
```

この例では、192.168.0.3 は /another/exported/directory/ の読み取り/書き込みをマウントでき、クライアントによる書き込み要求の完了前にディスクへのすべての転送がディスクにコミットされます。

さらに、デフォルト値が指定されていない他のオプションも利用できます。たとえば、サブツリーチェックを無効にする、安全でないポートからのアクセスの許可する、安全でないファイルロックを許可する (一部の初期 NFS クライアント実装で必要) などの機能があります。これらのあまり使用されないオプションの詳細は、exports の man ページを参照してください。

警告

/etc/exports ファイルの形式では、特に空白文字の使用が非常に厳しく扱われます。ホストからエクスポートするファイルシステムの間、そしてホスト同士の間には、必ず空白文字を挿入してください。また、それ以外の場所 (コメント行を除く) には、空白文字を追加しないでください。

たとえば、以下の 2 つの行は意味が異なります。

```
/home bob.example.com(rw)
/home bob.example.com (rw)
```

最初の行は、bob.example.com のユーザーにのみ、/home ディレクトリーへの読み取り/書き込みアクセスを許可します。2 番目の行では、bob.example.com からのユーザーにディレクトリーを読み取り専用 (デフォルト) でマウントすることを許可し、その他のユーザーに読み取り/書き込みでマウントすることを許可します。

21.7.1. exportfs コマンド

NFS 経由でリモートユーザーにエクスポートされるすべてのファイルシステム、およびそれらのファイルシステムのアクセスレベルは /etc/exports ファイルに一覧表示されます。nfs サービスが開始すると、/usr/sbin/exportfs コマンドが起動してこのファイルを読み込み、実際のマウントプロセスのために制御を rpc.mountd (NFSv2 および NFSv3 の場合) に渡してから、rpc.nfsd に渡します。この時点でリモートユーザーがファイルシステムを使用できるようになります。

/usr/sbin/exportfs コマンドを手動で発行すると、root ユーザーは NFS サービスを再開始せずにディレクトリーをエクスポートするか、しないかを選択できるようになります。適切なオプションが指定されると、/usr/sbin/exportfs コマンドは、エクスポートされたファイルシステムを /var/lib/nfs/xtab に書き込みます。ファイルシステムへのアクセス権を決定する際には、rpc.mountd がxtab ファイルを参照するため、エクスポートしたファイルシステムの一覧への変更がすぐに反映されます。

以下は、/usr/sbin/exportfs で利用可能な一般的に使用されるオプションの一覧です。

- *-r - /etc/exports に一覧表示されているすべてのディレクトリーが、/etc/lib/nfs/xtab に新しいエクスポートリストを作成することにより、エクスポートします。このオプションは、/etc/exports に加えた変更でエクスポートリストを効果的に更新します。*

- *-a - /usr/sbin/exportfs に渡される他のオプションに応じて、すべてのディレクトリーをエクスポートするか、エクスポート解除します。他のオプションが指定されない場合、/usr/sbin/exportfs は、/etc/exports 内に指定してあるすべてのファイルシステムをエクスポートします。*

- *-o file-systems - /etc/exports に記載されていないディレクトリーを指定します。file-systems の部分を、エクスポートされる追加のファイルシステムに置き換えます。これらのファイルシステムは、/etc/exports で指定されたものと同じフォーマットでなければなりません。/etc/exports 構文の詳細は、「/etc/exports 設定ファイル」 を参照してください。このオプションは、エクスポートするファイルシステムのリストに永続的に追加する前に、エクスポートするファイルシステムをテストするためによく使用されます。*

- *-i - /etc/exports を無視します。コマンドラインから指定されたオプションのみが、エクスポートされるファイルシステムの定義に使用されます。*

- *-u - すべての共有ディレクトリーのエクスポートを解除します。コマンド /usr/sbin/exportfs -ua は、すべての NFS デーモンを稼働状態に維持しながら、NFS ファイル共有を保留します。NFS 共有を再度有効にするには、exportfs -r と入力します。*

- *-v - exportfs コマンドの実行時にエクスポートまたはエクスポートされていないファイルシステムがより詳細に表示されます。*

/usr/sbin/exportfs コマンドにオプションが渡されていない場合は、現在エクスポートされているファイルシステムの一覧が表示されます。

/usr/sbin/exportfs コマンドの詳細は、man ページの exportfs を参照してください。

21.7.1.1. NFSv4 で exportfs の使用

exportfs コマンドは、エクスポートしたファイルシステムの NFS テーブルの管理に使用されます。引数なしでターミナルに入力すると、exportfs コマンドはエクスポートされたすべてのディレクトリーを表示します。

NFSv4 は NFSv2 プロトコルおよび NFSv3 プロトコルで使用されていた MOUNT プロトコルを使用しなくなったため、ファイルシステムのマウントが変更されました。

NFSv4 クライアントは、NFSv4 サーバーによって提供されるすべてのエクスポートを、NFSv4 擬似ファイルシステムと呼ばれる単一のファイルシステムとして確認できるようになりました。Red Hat

Enterprise Linux では、擬似ファイルシステムが、fsid=0 オプションでエクスポート時に識別される単一の実際のファイルシステムとして識別されます。

たとえば、次のコマンドは NFSv4 サーバーで実行できます。

```
mkdir /exports
mkdir /exports/opt
mkdir /exports/etc
mount --bind /usr/local/opt /exports/opt
mount --bind /usr/local/etc /exports/etc
exportfs -o fsid=0,insecure,no_subtree_check gss/krb5p:/exports
exportfs -o rw,nohide,insecure,no_subtree_check gss/krb5p:/exports/opt
exportfs -o rw,nohide,insecure,no_subtree_check gss/krb5p:/exports/etc
```

この例では、中断不可能なリンクを作成する --bind オプションを使用して、マウントする複数のファイルシステムを持つクライアントが提供されます。

擬似ファイルシステム機能により、NFS バージョン 2、3、および 4 エクスポート設定は常に互換性があるとは限りません。たとえば、以下のディレクトリーツリーがあるとします。

```
/home
/home/sam
/home/john
/home/joe
```

また、エクスポートは以下のようになります。

```
/home *(rw,fsid=0,sync)
```

NFS バージョン 2、3、および 4 を使用すると、以下が機能します。

```
mount server:/home /mnt/home
ls /mnt/home/joe
```

v4 を使用すると、以下が機能します。

```
mount -t nfs4 server:/ /mnt/home
ls /mnt/home/joe
```

相違点は、server:/homeおよびserver:/です。すべてのバージョンでエクスポート設定を互換性を

持たせるには、fsid=0 で root ファイルシステムをエクスポート（読み取り専用）する必要があります。fsid=0 は、このエクスポートがルートであることを NFS サーバーに指示します。

```
/ *(ro,fsid=0)
/home *(rw,sync,nohide)
```

これらのエクスポートを使用すると、mount server:/home /mnt/home" と "mount -t nfs server:/home /mnt/homeの両方が期待どおりに機能します。

21.8. NFS のセキュア化

NFS は、ファイルシステム全体を多数の既知のホストと透過的に共有する場合に適しています。ただし、使いやすくなると、さまざまな潜在的なセキュリティー問題があります。

サーバーで NFS ファイルシステムをエクスポートする場合や、クライアントにマウントする場合は、以下の点を考慮する必要があります。そうすることで、NFS のセキュリティーリスクが最小限に抑えられ、サーバー上のデータがより適切に保護されます。

21.8.1. ホストアクセス

実装予定の NFS のバージョンに応じて、既存のネットワーク環境に応じて、セキュリティーの懸念事項により異なります。以下のセクションでは、NFSv2、NFSv3、および NFSv4 でセキュリティー対策を実装する際の相違点を説明します。可能な限り、他のバージョンの NFS よりも NFSv4 の使用が推奨されます。

21.8.1.1. NFSv2 または NFSv3 の使用

NFS は、実際にファイルシステムを使用するユーザーではなく、マウント要求を行うホストに基づいてエクスポートされたファイルシステムをマウントできるユーザーを制御します。ホストには、エクスポートしたファイルシステムをマウントするための明示的な権限を付与する必要があります。ファイルおよびディレクトリーのパーミッション以外のユーザーにはアクセス制御はできません。つまり、NFS 経由でファイルシステムがエクスポートされると、NFS サーバーに接続されているリモートホストの任意のユーザーが共有データにアクセスできるようになります。こうしたリスクを抑えるため、管理者によって共通のユーザーおよびグループ ID へのユーザー権限が取り消されたり、読み取り専用のアクセスに制限されたりすることがよくあります。ただし、このソリューションにより、NFS 共有が元々想定されている方法では使用されなくなります。

また、NFS ファイルシステムをエクスポートしているシステムで使用している DNS サーバーのコントロールが攻撃者に奪われると、特定のホスト名または完全修飾ドメイン名に関連付けられているシステムが、未承認のマシンに向かう可能性があります。この時、NFS マウントには、これ以上の安全確保を目的としたユーザー名やパスワード情報の交換が行われないため、この未承認のマシンが NFS 共有のマウントを許可されたシステムに なってしまいます。

ワイルドカードの範囲が意図したよりも多くのシステムを含める可能性があるため、NFS 経由でディレクトリーをエクスポートする場合は、ワイルドカードを慎重に使用する必要があります。

TCP ラッパーを介して portmap サービスへのアクセスを制限することも可能です。また、iptables でファイアウォールルールを作成することで、portmap、rpc.mountd、および rpc.nfsd が使用するポートにアクセスすることもできます。

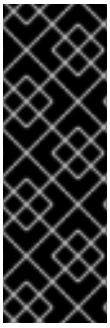
NFS および portmap のセキュリティー保護に関する詳細は、「iptables」 を参照してください。

21.8.1.2. NFSv4 の使用

NFSv4 のリリースにより、NFS エクスポートへの認証とセキュリティーが向上します。NFSv4 では、RPCSEC_GSS カーネルモジュールと Kerberos バージョン 5 GSS-API メカニズムの実装が義務付けられています。NFSv4 では、必須のセキュリティーメカニズムは個々のユーザーの認証を目的としており、NFSv2 および NFSv3 で使用されるクライアントマシンではありません。

注記

NFSv4 サーバーを設定する前に、Kerberos ticket-granting server (KDC) が正しくインストールおよび設定されていることを前提としています。Kerberos はネットワーク認証システムであり、対称暗号化と、信頼できるサードパーティー (KDC) を使用してクライアントとサーバーが相互に認証できるようにします。



重要

NFSv4 サーバーが Kerberos バージョン 5 GSS-API メカニズムを使用するように設定されている場合、NFS over UDP はサポートされず、クライアントシステムで NFS エクスポートのファイルシステムのマウントが失敗する可能性があります。したがって、この状況では TCP を使用することが推奨されます。詳細は、「TCP での NFS の使用」 を参照してください。

NFSv4 には、POSIX モデルではなく、Microsoft Windows NT モデルをベースとした ACL サポートが含まれています。これは、機能が広くデプロイされているためです。NFSv2 および NFSv3 には、ネイティブ ACL 属性のサポートはありません。

NFSv4 のもう 1 つの重要なセキュリティー機能は、ファイルシステムのマウントに MOUNT プロトコルの使用を削除することです。このプロトコルは、ファイルハンドルの処理方法が原因で、セキュリティーホールの可能性を示していました。

rpc.svcgssd と rpc.gssd の相互動作など、RPCSEC_GSS フレームワークの詳細は、http://www.citi.umich.edu/projects/nfsv4/gssd/ を参照してください。

21.8.2. ファイル権限

リモートホストにより NFS ファイルシステムを読み取り/書き込みとしてマウントした場合は、パーミッションが、各共有ファイルに対する唯一の保護となります。同じユーザー ID 値を共有する 2 つのユーザーが同じ NFS ファイルシステムをマウントすると、他のユーザーがお互いのファイルを変更できます。さらに、クライアントシステムに root としてログインしているユーザーは、su - コマンドを使用して、NFS 共有を介して特定のファイルにアクセスできるユーザーになります。

デフォルトでは、アクセス制御リスト (ACL) は、Red Hat Enterprise Linux では NFS が対応しています。この機能を無効にすることは推奨されません。

NFS 経由でファイルシステムをエクスポートする場合のデフォルトの動作は、root squashing を使用します。これにより、NFS 共有にアクセスするユーザーのユーザー ID が、ローカルマシンの root ユーザーとしてサーバーの nfsnobody アカウントの値に設定されます。root squashing を無効にしないでください。

NFS 共有を読み取り専用としてエクスポートする場合は、all_squash オプションの使用を検討してください。これにより、エクスポートしたファイルシステムにアクセスするすべてのユーザーが nfsnobody ユーザーのユーザー ID を取得します。

21.9. NFS と PORTMAP

注記

以下のセクションは、後方互換性のために portmap サービスを必要とする NFSv2 または NFSv3 実装にのみ適用されます。

portmapper は、RPC サービスをリッスンしているポートにマッピングします。RPC プロセスは、起動時に portmap に通知します。これは、リッスンしているポートと、そのプロセスが提供することが予想される RPC プログラム番号を登録します。次に、クライアントシステムは、特定の RPC プログラム番号でサーバーの portmap に接続します。portmap サービスは、クライアントを適切なポート番号にリダイレクトし、要求されたサービスと通信できるようにします。

RPC ベースのサービスは portmap に依存して受信クライアント要求ですべての接続を確立するため、ポートマップはこれらのサービスのいずれかが開始する前に、ポートマップを利用できるようにする必要があります。

portmap サービスはアクセス制御に TCP ラッパーを使用し、portmap のアクセス制御ルールはすべて の RPC ベースのサービスに影響します。あるいは、NFS RPC デーモンごとにアクセス制御ルールを指定することもできます。rpc.mountd および rpc.statd の man ページには、これらのルールの正確な構文に関する情報が記載されています。

21.9.1. NFS と portmapのトラブルシューティング

portmap は RPC サービスとそれらとの通信に使用されるポート番号を調整するため、トラブルシューティング時に portmap を使用して現在の RPC サービスのステータスを表示すると便利です。rpcinfo コマンドを使用すると RPC ベースの各サービスとそのポート番号、RPC プログラム番号、バージョン番号、および IP プロトコルタイプ (TCP または UDP) が表示されます。

portmap に対して適切な RPC ベースの NFS サービスが有効になっていることを確認するには、root で以下のコマンドを実行します。

```
rpcinfo -p
```

以下に、上記コマンドの出力例を示します。

```
program vers proto   port
100000    2   tcp    111  portmapper
100000    2   udp    111  portmapper
100021    1   udp  32774  nlockmgr
100021    3   udp  32774  nlockmgr
100021    4   udp  32774  nlockmgr
100021    1   tcp  34437  nlockmgr
100021    3   tcp  34437  nlockmgr
100021    4   tcp  34437  nlockmgr
100011    1   udp    819  rquotad
100011    2   udp    819  rquotad
100011    1   tcp    822  rquotad
100011    2   tcp    822  rquotad
100003    2   udp   2049  nfs
100003    3   udp   2049  nfs
100003    2   tcp   2049  nfs
100003    3   tcp   2049  nfs
100005    1   udp    836  mountd
100005    1   tcp    839  mountd
100005    2   udp    836  mountd
100005    2   tcp    839  mountd
100005    3   udp    836  mountd
100005    3   tcp    839  mountd
```

NFS サービスの 1 つが正しく起動しないと、portmap は、そのサービスのクライアントからの RPC 要求を正しいポートにマッピングできません。多くの場合、NFS が rpcinfo の出力に存在しない

場合は、NFS を再起動すると、サービスが portmap に正しく登録され、機能し始めます。NFS の開始方法は、「NFS の開始と停止」 を参照してください。

rpcinfo コマンドでは、その他の便利なオプションを使用できます。詳細は、man ページの rpcinfo を参照してください。

21.10. TCP での NFS の使用

NFSv4 のデフォルトのトランスポートプロトコルは TCP です。TCP を使用する利点は次のとおりです。

- 接続の持続性が向上し、NFS の古いファイルがメッセージを処理 するのが少なくなります。
- TCP は、完了のみを確認する UDP とは異なり、すべてのパケットを確認するため、負荷の高いネットワークでパフォーマンスが向上しています。
- TCP の輻輳制御は UDP よりも優れています。非常に輻輳したネットワークでは、UDP パケットはドロップされる最初のパケットです。つまり、NFS がデータを書き込む場合(8K チャンク)、その 8K はすべて UDP で再送信する必要があります。TCP の信頼性により、その 8K データの一部のみが一度に送信されます。
- エラー検出。（サーバーが利用不可であることが原因で）TCP 接続が切断されると、クライアントはデータの送信を停止し、サーバーが利用可能になると接続プロセスを再起動します。UDP では、接続なしであるため、クライアントはサーバーが接続を再確立するまで、データでネットワークを保留し続けます。

主な欠点は、TCP プロトコルに関連するオーバーヘッドが原因で、パフォーマンスが大幅に低下することです。

21.11. 関連情報

NFS サーバーの管理は難しい課題となる場合があります。本章では言及していませんが、NFS 共有のエクスポートやマウントに利用できるオプションは多数あります。詳細は、以下のソースを参照してください。

21.11.1. インストールされているドキュメント

-

/usr/share/doc/nfs-utils- <version-number> / - < version-number > を、インストールされている NFS パッケージのバージョン番号に置き換えます。このディレクトリーには、Linux 用の NFS 実装に関する情報が含まれています。これには、さまざまな NFS 設定やファイル転送パフォーマンスへの影響が含まれます。

- man mount — NFS のサーバー設定およびクライアント設定に使用するマウントオプションに関して総合的に説明しています。

- man fstab — 起動時にファイルシステムをマウントするために使用される /etc/fstab ファイルの形式の詳細を提供します。

- man nfs — NFS 固有のファイルシステムのエクスポートおよびマウントオプションについて詳細に説明しています。

- man exports — NFS ファイルシステムのエクスポート時に /etc/exports ファイル内で使用する一般的なオプションを表示します。

21.11.2. 便利な Web サイト

- http://nfs.sourceforge.net/: Linux NFS プロジェクトのホームおよびプロジェクトステータスの更新に最適な場所です。

- http://www.citi.umich.edu/projects/nfsv4/linux/ — Linux 2.6 カーネル用 NFSv4 のリソースです。

- http://www.nfsv4.org - NFS バージョン 4 のホームおよび関連するすべての標準。

- http://www.vanemery.com/Linux/NFSv4/NFSv4-no-rpcsec.html — 2.6 カーネルを含む Fedora Core 2 での NFSv4 の詳細について説明しています。

- http://www.sane.nl/events/sane2000/papers/pawlowski.pdf: NFS バージョン 4 プロトコルの機能と機能拡張に関する優れた協力です。

- http://wiki.autofs.net - Autofs wiki、議論、ドキュメント、および機能拡張。

21.11.3. 関連書籍

- 『Managing NFS and NIS』 *(Hal Stern、Mike Eisler および Ricardo Labiaga 著、O'Reilly & Associates 出版) — 利用可能な各種の NFS エクスポートやマウントオプションについて記載している優れた参考ガイドです。*

- 『NFS Illustrated』 *(Brent Callaghan 著、Addison-Wesley Publishing Company 出版): NFS と他のネットワークファイルシステムとの比較、NFS 通信がどのように発生するかなどが詳細に紹介されています。*

第22章 SAMBA

Samba は、Server Message Block (SMB)プロトコルのオープンソース実装です。これにより、Microsoft Windows®、Linux、UNIX、およびその他のオペレーティングシステムのネットワークが一緒に提供され、Windows ベースのファイルおよびプリンター共有にアクセスできます。Samba の SMB を使用すると、Windows クライアントに Windows サーバーとして表示できます。

22.1. SAMBA の概要

Samba のバージョン 3.0.0 の 3 番目のメジャーリリースでは、以前のバージョンから以下のような多くの改善が行われました。

- LDAP および Kerberos を使用して Active Directory ドメインに参加する機能
- 国際化のための Unicode サポートが組み込まれています。
- ローカルレジストリーのハッキングを必要とせずに、Samba サーバーへの Microsoft Windows XP Professional クライアント接続のサポート
- 400+ ページ参照マニュアルを含む Samba.org チームによって開発された 2 つの新しいドキュメントと、300 以上のページ実装と統合マニュアルが含まれています。これらのタイトルの詳細については、「関連書籍」 を参照してください。

22.1.1. Samba の機能

Samba は、強力で汎用のサーバーアプリケーションです。分離したシステム管理者は、インストールと設定を試行する前に、その機能および制限を把握しておく必要があります。

Samba の機能は次のとおりです。

- Linux、UNIX、Windows クライアントへのディレクトリーツリーとプリンターの提供
- ネットワーク参照を支援する(NetBIOS の有無にかかわらず)

- *Windows ドメインログインの認証*
- *Windows Internet Name Service (WINS)ネームサーバー解決の提供*
- *Windows NT®-style Primary Domain Controller (PDC)として動作*
- *Samba ベースの PDC のバックアップドメインコントローラー(BDC)として動作する*
- *Active Directory ドメインメンバーサーバーとして動作する。*
- *Windows NT/2000/2003 PDC に参加*

Samba が実行できないこと：

- *Windows PDC の BDC として機能する（逆も同様）*
- *Active Directory ドメインコントローラーとして動作する*

22.2. SAMBA デーモンと関連サービス

以下は、個々の Samba デーモンおよびサービスの概要です。

22.2.1. Samba デーモン

Samba は、3 つのデーモン(smbd、nmbbd、および winbindd)で設定されています。2 つのサービス(smb および windbind)は、デーモンの開始、停止、およびその他のサービス関連の機能を制御します。各デーモンの詳細と、そのデーモンを制御する特定のサービスが表示されます。

smbd

smbd サーバーデーモンは、Windows クライアントにファイル共有および印刷サービスを提供します。さらに、SMB プロトコルを使用したユーザー認証、リソースロック、およびデータ共有を行い

ます。サーバーが SMB トラフィックをリッスンするデフォルトのポートは TCP ポート 139 および 445 です。

smbd デーモンは smb サービスによって制御されます。

nmbd

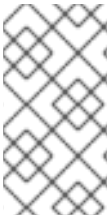
nmbd サーバーデーモンは、Windows ベースのシステムの SMB/CIFS によって生成されたものなど、NetBIOS ネームサービス要求を理解して応答します。これらのシステムには、Windows 95/98/ME、Windows NT、Windows 2000、Windows XP、および LanManager クライアントが含まれます。また、Windows Network Neighborhood ビューを設定する参照プロトコルに参加します。サーバーが NMB トラフィックをリッスンするデフォルトのポートは UDP ポート 137 です。

nmbd デーモンは、smb サービスによって制御されます。

winbindd

winbind サービスは、Windows NT 2000 または Windows Server 2003 を実行しているサーバーのユーザーおよびグループ情報を解決します。これにより、UNIX プラットフォームが理解できる Windows ユーザー/グループ情報が使用できるようになります。これは、Microsoft RPC 呼び出し、Pluggable Authentication Modules (PAM)、および Name Service Switch (NSS)を使用して実行されます。これにより、Windows NT ドメインユーザーが UNIX マシン上で UNIX ユーザーとして表示および操作できるようになります。Samba ディストリビューションにバンドルされていますが、winbind サービスは smb サービスとは別に制御されます。

winbindd デーモンは winbind サービスによって制御され、操作に smb サービスを起動する必要はありません。winbindd は、Samba が Active Directory メンバーである場合にも使用され、Samba ドメインコントローラーでも使用できます（ネストされたグループやドメイン間の信頼を実装するため）。winbind は Windows NT ベースのサーバーへの接続に使用されるクライアント側のサービスであるため、winbind の詳細はこのマニュアルの範囲外になります。



注記

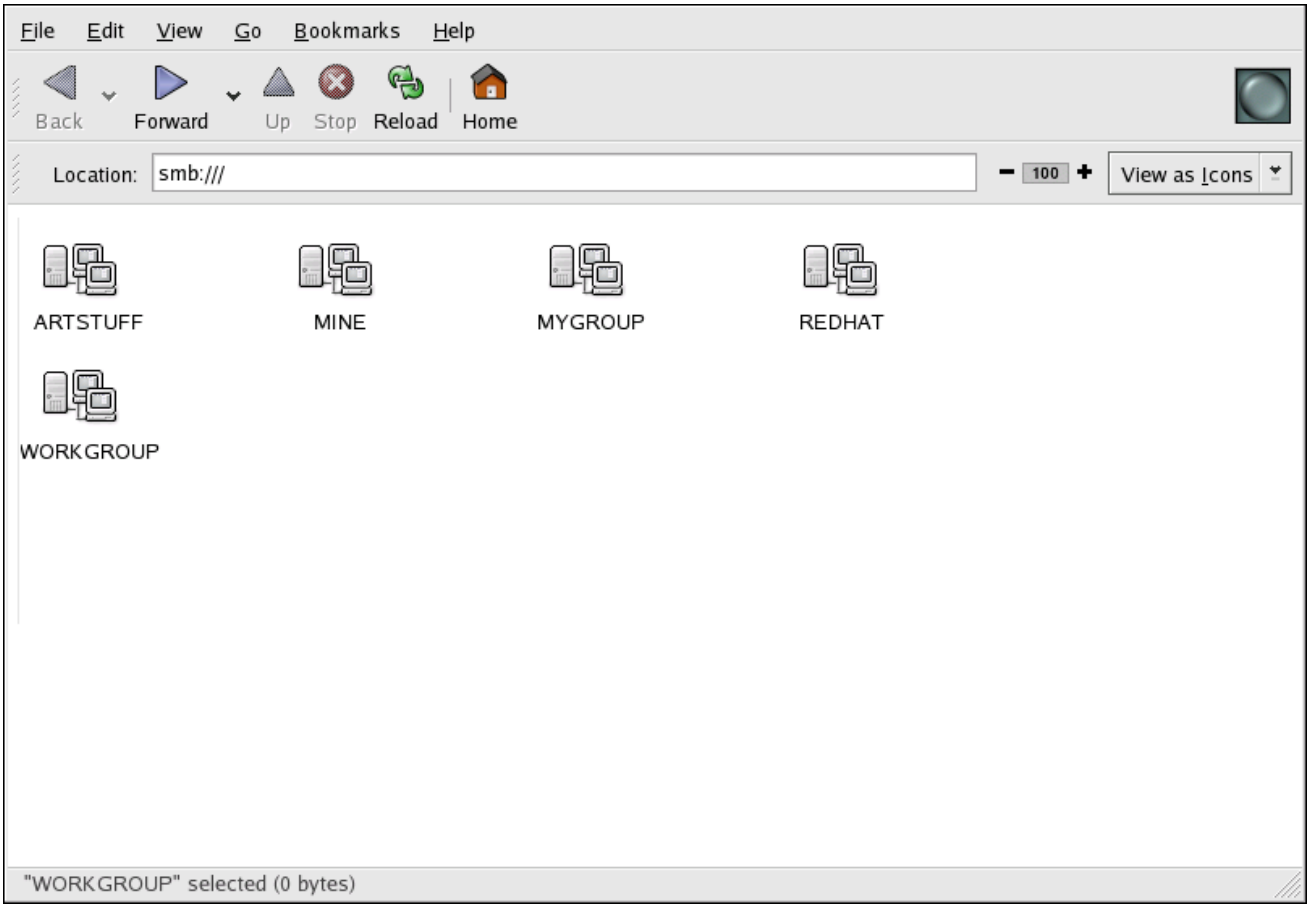
Samba ディストリビューションに含まれるユーティリティーの一覧は、「Samba ディストリビューションプログラム」 を参照してください。

22.3. SAMBA 共有への接続

Nautilus を使用して、ネットワークで利用可能な Samba 共有を表示できます。Places （パネル上）> Network Servers を選択して、ネットワーク上の Samba ワークグループの一覧を表示します。また、Nautilus の File > Open Location バーに smb: を入力してワークグループを表示することもできます。

図22.1「Nautilus の SMB ワークグループ」 に示すように、ネットワーク上の利用可能な SMB ワークグループごとにアイコンが表示されます。

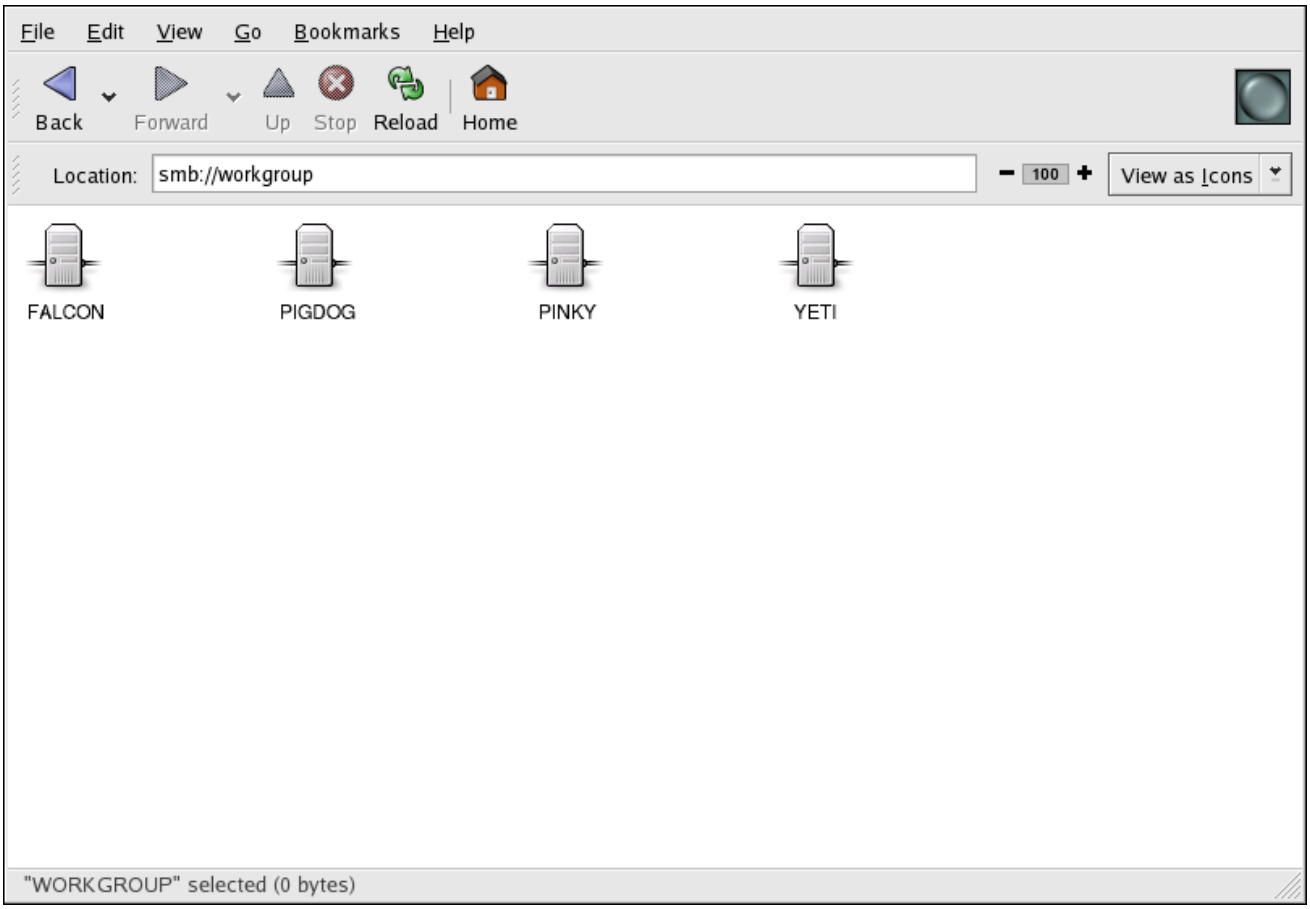
図22.1 Nautilus の SMB ワークグループ



[D]

ワークグループアイコンのいずれかをダブルクリックして、ワークグループ内のコンピューターの一覧を表示します。

図22.2 Nautilus の SMB マシン



[D]

図22.2「Nautilus の SMB マシン」から分かるように、workgroup 内に各マシンにアイコンがあります。アイコンをダブルクリックして、マシン上の Samba 共有を表示します。ユーザー名とパスワードの組み合わせが必要な場合は、プロンプトが出されます。

または、以下の構文を使用して、場所 に Samba サーバーおよび sharename を指定することもできます (< servername > および < sharename > を適切な値に置き換えます)。

```
smb://<servername>/<sharename>
```

22.3.1. コマンドライン

Samba サーバーのネットワークをクエリーするには、findsmb コマンドを使用します。見つかった各サーバーについて、IP アドレス、NetBIOS 名、ワークグループ名、オペレーティングシステム、および SMB サーバーのバージョンが表示されます。

シェルプロンプトから Samba 共有に接続するには、以下のコマンドを入力します。

```
smbclient //<hostname>/<sharename> -U <username>
```

& lt;hostname > を、接続する Samba サーバーのホスト名または IP アドレスに置き換えます。< sharename > は参照する共有ディレクトリーの名前に、< username > はシステムの Samba ユーザー名に置き換えます。正しいパスワードを入力するか、ユーザーにパスワードが必要ない場合は Enter を押します。

smb:\> プロンプトが表示されている場合は、正常にログインしました。ログインしたら、コマンドのリストに help と入力します。ホームディレクトリーの内容を参照する場合は、sharename をユーザー名に置き換えます。-U スイッチを使用しない場合、現行ユーザーのユーザー名が Samba サーバーに渡されます。

smbclient を終了するには、smb:\> プロンプトで exit と入力します。

22.3.2. 共有のマウント

Samba 共有をディレクトリーにマウントして、ディレクトリーのファイルをローカルファイルシステムの一部であるかのように扱う方が便利な場合があります。

Samba 共有をディレクトリーにマウントするには、（存在しない場合）マウントするディレクトリーを作成し、root で以下のコマンドを実行します。

```
mount -t cifs -o <username>,<password> //<servername>/<sharename> /mnt/point/
```

このコマンドは、& lt; servername > からローカルディレクトリー /mnt/point/ に <sharename > をマウントします。samba 共有のマウントの詳細は、man mount.cifs を参照してください。

22.4. SAMBA サーバーの設定

デフォルトの設定ファイル(/etc/samba/smb.conf)を使用すると、ユーザーはホームディレクトリーを Samba 共有として表示できます。また、システム用に設定されたすべてのプリンターを Samba 共有プリンターとして共有します。つまり、システムにプリンターを接続し、ネットワーク上の Windows マシンからそのプリンターに印刷できます。

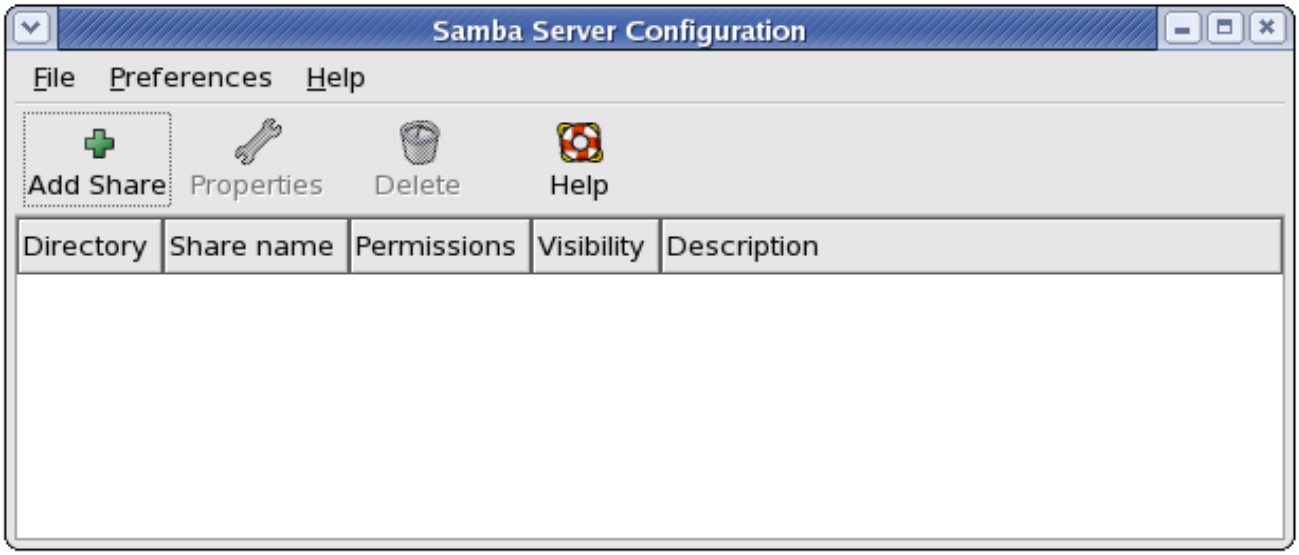
22.4.1. グラフィカル設定

グラフィカルインターフェイスを使用して Samba を設定するには、Samba Server Configuration Tool を使用します。コマンドラインの設定の場合は、「コマンドラインからの設定」 に進みます。

Samba Server Configuration Tool は、Samba 共有、ユーザー、および基本的なサーバー設定を管理するグラフィカルインターフェイスです。/etc/samba/ ディレクトリーの設定ファイルを変更します。アプリケーションを使用していないこれらのファイルへの変更は保持されます。

このアプリケーションを使用するには、X Window System を実行し、root 権限を持ち、system-config-samba RPM パッケージがインストールされている必要があります。デスクトップから Samba Server Configuration Tool を起動するには、システム（パネル上）> Administration > Server Settings > Samba に移動するか、シェルプロンプトでコマンド system-config-samba を入力します (XTerm や GNOME 端末など)。

図22.3 Samba サーバー設定ツール



[D]

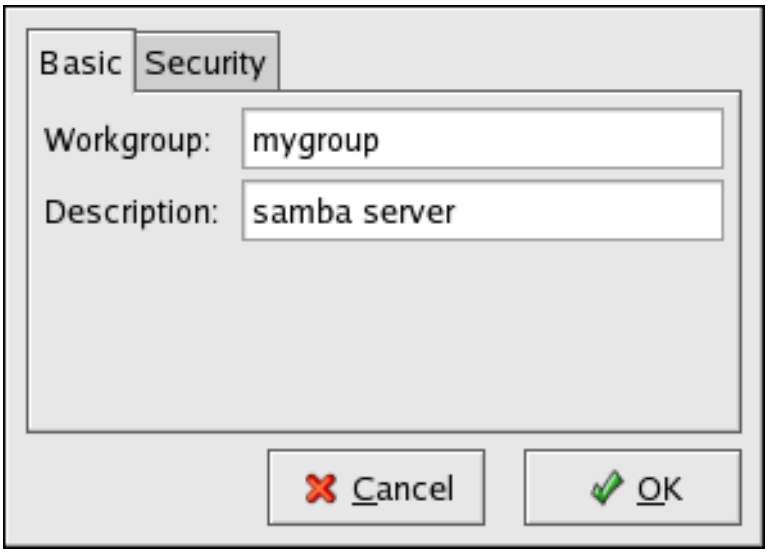
注記

Samba Server Configuration Tool は、共有プリンターや、ユーザーが Samba サーバー上の独自のホームディレクトリーを表示できるようにするデフォルトのスタンザを表示しません。

22.4.1.1. サーバー設定の設定

Samba サーバーを設定する最初のステップは、サーバーの基本設定とセキュリティーオプションを設定することです。アプリケーションの起動後に、プルダウンメニューから Preferences > Server Settings を選択します。Basic タブは、図22.4「基本的なサーバー設定の設定」 のように表示されます。

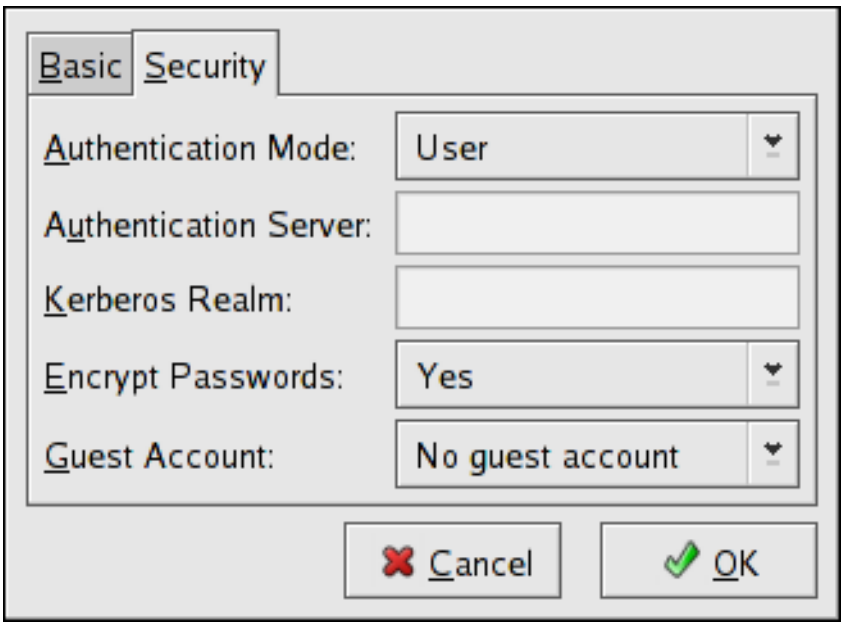
図22.4 基本的なサーバー設定の設定



[D]

Basic タブで、コンピューターに含まれるワークグループと、コンピューターの簡単な説明を指定します。smb.conf の workgroup および server 文字列 オプションに対応します。

図22.5 セキュリティーサーバー設定の設定

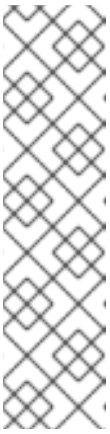


[D]

Security タブには以下のオプションが含まれます。

- *Authentication Mode: これは セキュリティー オプションに対応します。以下のタイプの認証のいずれかを選択します。*
  -

ADS - Samba サーバーは、Active Directory Domain (ADS)レルムのドメインメンバーとして機能します。このオプションでは、Kerberos がサーバーにインストールおよび設定され、Samba は、samba-client パッケージに含まれる net ユーティリティーを使用して ADS レルムのメンバーになる必要があります。詳細は、net の man ページを参照してください。このオプションでは、Samba を ADS コントローラーとして設定することはできません。Kerberos Realm フィールドに Kerberos サーバーのレルムを指定します。



注記

Kerberos Realm フィールドは、EXAMPLE.COM などの大文字で指定する必要があります。

Samba サーバーを ADS レルムのドメインメンバーとして使用すると、/etc/krb5.conf ファイルを含む Kerberos が適切に設定されていることを前提としています。

- *domain - Samba サーバーは、Windows NT Primary または Backup Domain Controller に依存してユーザーを検証します。サーバーはユーザー名とパスワードをコントローラーに渡し、返すまで待機します。Authentication Server フィールドに、プライマリーまたはバックアップドメインコントローラーの NetBIOS 名を指定します。*

  これを選択する場合は、Encrypted Passwords オプションを Yes に設定する必要があります。

- *server - Samba サーバーは、別の Samba サーバーに渡すことで、ユーザー名とパスワードの組み合わせを検証しようとします。そうでない場合、サーバーはユーザー認証モードを使用して検証を試みます。Authentication Server フィールドに他の Samba サーバーの NetBIOS 名を指定します。*

- *share - Samba ユーザーは、Samba サーバーごとにユーザー名とパスワードの組み合わせを入力する必要はありません。Samba サーバーから特定の共有ディレクトリーに接続しようとするまで、ユーザー名とパスワードの入力は求められません。*

- *user -（デフォルト）Samba ユーザーは、Samba サーバーごとに有効なユーザー名とパスワードを提供する必要があります。Windows ユーザー名 オプションを使用する場合は、このオプションを選択します。詳細は、「Samba ユーザーの管理」 を参照してください。*

- *encrypt Passwords - クライアントが Windows 98 のシステムから接続している場合は、Service Pack 3 を使用する Windows NT 4.0、またはその他の最新バージョンの*

Microsoft Windows を有効にする必要があります。パスワードは、傍受できるプレーンテキストの単語としてではなく、暗号化された形式でサーバーとクライアント間で転送されます。これは、暗号化されたパスワード オプションに対応します。暗号化された Samba パスワードの詳細は、「暗号化されたパスワード」 を参照してください。

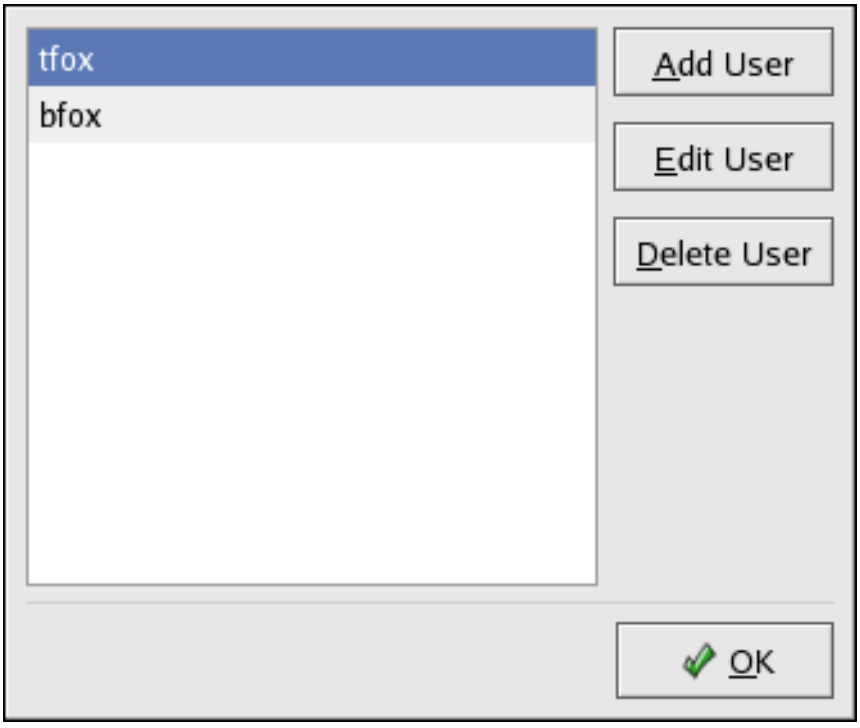
- *ゲストアカウント - ユーザーまたはゲストユーザーが Samba サーバーにログインする場合は、サーバー上の有効なユーザーにマッピングされる必要があります。ゲストの Samba アカウントとして使用するシステムの既存のユーザー名の中から 1 つ選択します。ゲストが Samba サーバーにログインすると、このユーザーと同じ権限があります。これは、ゲストアカウント オプションに対応します。*

OK をクリックすると、変更が設定ファイルに書き込まれ、デーモンが再起動されます。変更がすぐに反映されます。

22.4.1.2. Samba ユーザーの管理

Samba Server Configuration Tool では、Samba ユーザーを追加する前に、Samba サーバーとして動作するシステムで既存のユーザーアカウントを有効にする必要があります。Samba ユーザーが既存のユーザーアカウントに関連付けられている。

図22.6 Samba ユーザーの管理



[D]

Samba ユーザーを追加するには、プルダウンメニューから Preferences > Samba Users を選択して、Add User ボタンをクリックします。Create New Samba User ウィンドウで、ローカルシステムの既存ユーザーの一覧から Unix ユーザー名 を選択します。

ユーザーが Windows マシンに別のユーザー名を持ち、Windows マシンから Samba サーバーにログインする必要がある場合は、Windows ユーザー名 フィールドに Windows ユーザー名を指定 し ます。このオプションを機能させるには、Server Settings 設定の Security タブの Authentication Mode を User に設定する必要があります。

また、Samba ユーザーの Samba パスワード を設定し、再度入力して確認します。Samba に暗号化されたパスワードを使用することを選択した場合でも、すべてのユーザーの Samba パスワードはシステムパスワードとは異なることが推奨されます。

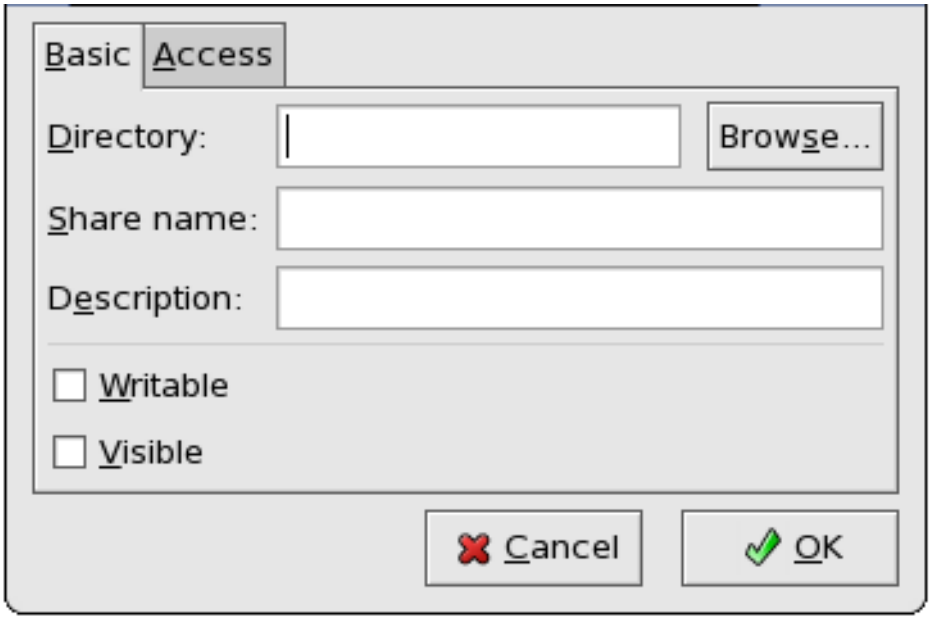
既存のユーザーを編集するには、一覧からユーザーを選択し、Edit User をクリックします。既存の Samba ユーザーを削除するには、ユーザーを選択し、Delete User ボタンをクリックします。Samba ユーザーを削除しても、関連付けられたシステムユーザーアカウントは削除されません。

ユーザーは、OK ボタンをクリックした直後に変更されます。

22.4.1.3. 共有の追加

Samba 共有を作成するには、メインの Samba 設定ウィンドウから Add ボタンをクリックします。

図22.7 共有の追加



[D]

Basic タブでは、以下のオプションを設定します。

-

directory - Samba 経由で共有するディレクトリー。ディレクトリーは、ここに入力する前に存在している必要があります。

- *共有名 - リモートマシンから表示される共有の実際の名前。デフォルトでは、これは Directory と同じ値ですが、設定できます。*
- *説明: ファイル共有の簡単な説明*
- *writable: ユーザーが共有ディレクトリーの読み取りと書き込みを可能にします。*
- *visible - 共有ディレクトリーのユーザーに読み取り専用権限を付与します。*

Access タブで、指定したユーザーのみが共有にアクセスできるようにするか、またはすべての Samba ユーザーが共有にアクセスできるようにするかを選択します。特定のユーザーへのアクセスを許可する場合は、利用可能な Samba ユーザーのリストからユーザーを選択します。

OK をクリックすると、共有が即座に追加されます。

22.4.2. コマンドラインからの設定

Samba は、設定ファイルとして /etc/samba/smb.conf を使用します。この設定ファイルを変更しても、コマンド service smb が再起動して Samba デーモンを再起動するまで変更は反映されません。

Windows ワークグループと Samba サーバーの簡単な説明を指定するには、smb.conf ファイルの次の行を編集します。

```
workgroup = WORKGROUPNAME
server string = BRIEF COMMENT ABOUT SERVER
```

WORKGROUPNAME は、このマシンが属する Windows ワークグループの名前に置き換えます。BRI EF COMMENT ABOUT SERVER はオプションで、 Samba システムに関する Windows コメントとして使用されます。

Linux システムで Samba 共有ディレクトリーを作成するには、以下のセクションを smb.conf ファイルに追加します（ニーズとシステムを反映するように変更した後）。

```
[sharename]
comment = Insert a comment here
path = /home/share/
valid users = tfox carole
public = no
writable = yes
printable = no
create mask = 0765
```

上記の例では、tfox および carole ユーザーが Samba クライアントから Samba サーバーのディレクトリー /home/share を読み書きできます。

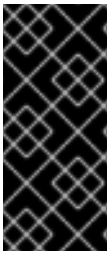
22.4.3. 暗号化されたパスワード

暗号化されたパスワードは、より安全であるため、デフォルトで有効になっています。暗号化されたパスワードでユーザーを作成するには、smbpasswd -a < username> コマンドを使用し ます。

22.5. SAMBA の起動および停止

Samba サーバーを起動するには、root としてログインしているシェルプロンプトで以下のコマンドを入力します。

```
service smb start
```



重要な影響

ドメインメンバーサーバーを設定するには、最初に net join コマンドを使用してドメインまたは Active Directory に参加してから、smb サービスを開始する必要があります。

サーバーを停止するには、root でログインした状態でシェルプロンプトで以下のコマンドを入力します。

```
service smb stop
```

restart オプションは、Samba を停止してから簡単に起動する方法です。これは、Samba の設定ファイルを編集した後に設定変更を有効にする最も信頼できる方法です。restart オプションは、最初に実行していない場合でもデーモンを起動することに注意してください。

サーバーを再起動するには、root でログインしている間にシェルプロンプトで以下のコマンドを入力します。

```
service smb restart
```

condrestart (conditional restart)オプションは、現在実行している条件でのみ smb を起動します。このオプションは、デーモンが実行されていない場合はデーモンを起動しないため、スクリプトに便利です。

注記

smb.conf ファイルが変更されると、Samba は数分後に自動的に再読み込みされます。手動 再起動 または リロード の実行は、効果的な方法と同じです。

条件付きでサーバーを再起動するには、root で以下のコマンドを入力します。

```
service smb condrestart
```

smb.conf ファイルの手動リロードは、smb サービスによる自動再読み込みに失敗した場合に役立ちます。サービスを再起動せずに Samba サーバー設定ファイルがリロードされるようにするには、root で以下のコマンドを入力します。

```
service smb reload
```

デフォルトでは、smb サービスは起動時に自動的に起動し ません。起動時に Samba が開始するように設定するには、/sbin/chkconfig、/usr/sbin/ntsysv、または Services Configuration Tool プログラムなどの initscript ユーティリティーを使用します。これらのツールの詳細は、18章 を参照してください。

22.6. SAMBA サーバータイプと SMB.CONF ファイル

Samba 設定は簡単です。Samba へのすべての変更は、/etc/samba/smb.conf 設定ファイルで行われます。デフォルトの smb.conf ファイルは適切に文書化されていますが、LDAP、Active Directory、多数のドメインコントローラーの実装などの複雑なトピックについては対応していません。

以下のセクションでは、Samba サーバーを設定するさまざまな方法を説明します。設定を成功させるために smb.conf ファイルに必要なニーズと変更に注意してください。

22.6.1. スタンドアロンサーバー

スタンドアロンサーバーは、ワークグループサーバーまたはワークグループ環境のメンバーになります。スタンドアロンサーバーはドメインコントローラーではなく、ドメインに参加しません。以下の例には、いくつかの匿名共有レベルのセキュリティー設定と 1 つのユーザーレベルのセキュリティー設定が含まれます。共有レベルおよびユーザーレベルのセキュリティーモードの詳細は、「Samba のセキュリティーモード」 を参照してください。

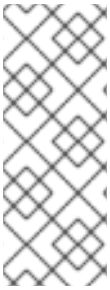
22.6.1.1. Anonymous Read-Only

以下の smb.conf ファイルは、匿名の読み取り専用ファイル共有を実装するために必要な設定例を示しています。security = share パラメーターは匿名を共有します。1 つの Samba サーバーのセキュリティーレベルを混在させることはできません。security ディレクティブは、smb.conf ファイルの [global] 設定セクションにあるグローバル Samba パラメーターです。

```
[global]
workgroup = DOCS
netbios name = DOCS_SRV
security = share
[data]
comment = Documentation Samba Server
path = /export
read only = Yes
guest only = Yes
```

22.6.1.2. Anonymous Read/Write

以下の smb.conf ファイルは、匿名の読み取り/書き込みファイル共有を実装するために必要な設定例を示しています。匿名の読み取り/書き込みファイル共有を有効にするには、読み取り専用 ディレクティブを no に設定します。force user ディレクティブおよび force group ディレクティブも追加され、共有で指定された新たに配置されたファイルの所有権を強制します。



注記

匿名の読み取り/書き込みサーバーが可能ですが、推奨されません。ユーザーに関係なく、共有領域に置かれたファイルはすべて、smb.conf ファイルの汎用ユーザー（強制的にユーザー）およびグループ(強制的にグループ)によって指定されるユーザーおよびグループの組み合わせが割り当てられます。

```
[global]
workgroup = DOCS
netbios name = DOCS_SRV
security = share
[data]
comment = Data
path = /export
force user = docsbot
```

```
force group = users
read only = No
guest ok = Yes
```

22.6.1.3. Anonymous Print Server

以下の smb.conf ファイルは、匿名プリントサーバーを実装するために必要な設定例を示しています。以下に示すように、参照可能な を no に設定しても、Windows Network Neighborhood のプリンターはリストされません。プリンターのブラウズから非表示にすることができますが、明示的にプリンターを設定することができます。NetBIOS を使用して DOCS_SRV に接続すると、クライアントが DOCS ワークグループにも含まれている場合は、クライアントはプリンターにアクセスできます。また、使用クライアントドライバーのディレクティブが Yes に設定されているため、クライアント のローカルプリンタードライバーがインストールされていることを前提とします。この場合、Samba サーバーはプリンタードライバーをクライアントと共有する必要はありません。

```
[global]
workgroup = DOCS
netbios name = DOCS_SRV
security = share
printcap name = cups
disable spools= Yes
show add printer wizard = No
printing = cups
[printers]
comment = All Printers
path = /var/spool/samba
guest ok = Yes
printable = Yes
use client driver = Yes
browseable = Yes
```

22.6.1.4. セキュアな読み取り/書き込みファイルおよびプリントサーバー

以下の smb.conf ファイルは、セキュアな読み取り/書き込みプリントサーバーを実装するために必要な設定例を示しています。セキュリティー ディレクティブを user に設定すると、Samba がクライアント接続を認証するように強制されます。[homes] 共有には、[public] 共有に force ユーザー または force group ディレクティブが含まれていないことに注意してください。[homes] 共有は、[public] の force user および force グループ ではなく、作成されたファイルに、認証されたユーザーの詳細を使用します。

```
[global]
workgroup = DOCS
netbios name = DOCS_SRV
security = user
printcap name = cups
disable spools = Yes
show add printer wizard = No
printing = cups
[homes]
comment = Home Directories
```

```
valid users = %S
read only = No
browseable = No
[public]
comment = Data
path = /export
force user = docsbot
force group = users
guest ok = Yes
[printers]
comment = All Printers
path = /var/spool/samba
printer admin = john, ed, @admins
create mask = 0600
guest ok = Yes
printable = Yes
use client driver = Yes
browseable = Yes
```

22.6.2. ドメインメンバーサーバー

スタンドアロンサーバーに類似したドメインメンバーは、ドメインコントローラー(Windows または Samba のいずれか)にログインし、ドメインのセキュリティールールに従います。ドメインメンバーサーバーの例としては、プライマリードメインコントローラー(PDC)上のマシンアカウントを持つ Samba を実行している部署サーバーが挙げられます。部門の全クライアントは、引き続き PDC とデスクトッププロファイルで認証され、すべてのネットワークポリシーファイルが含まれます。相違点は、部署サーバーではプリンターとネットワーク共有を制御できることです。

22.6.2.1. Active Directory ドメインメンバーサーバー

以下の smb.conf ファイルは、Active Directory ドメインメンバーサーバーを実装するために必要な設定例を示しています。この例では、Samba はローカルで実行されているサービスのユーザーを認証しますが、Active Directory のクライアントでもあります。kerberos レルムパラメーターがすべての上限（例： realm = EXAMPLE.COM）に表示されることを確認します。Windows 2000/2003 では Active Directory 認証に Kerberos が必要なため、realm ディレクティブが必要です。Active Directory と Kerberos が異なるサーバーで実行されている場合は、パスワードサーバー のディレクティブを区別するために必要になることがあります。

```
[global]
realm = EXAMPLE.COM
security = ADS
encrypt passwords = yes
# Optional. Use only if Samba cannot determine the Kerberos server automatically.
password server = kerberos.example.com
```

メンバーサーバーを Active Directory ドメインに参加させるには、以下の手順を実行する必要があります。

- メンバーサーバー上の smb.conf ファイルの設定
- メンバーサーバーの /etc/krb5.conf ファイルを含む Kerberos の設定
- Active Directory ドメインサーバーでのマシンアカウントの作成
- メンバーサーバーと Active Directory ドメインの関連付け

マシンアカウントを作成し、Windows 2000/2003 Active Directory に参加するには、最初に Active Directory ドメインに参加するメンバーサーバー用に Kerberos を初期化する必要があります。管理 Kerberos チケットを作成するには、メンバーサーバーで root で以下のコマンドを入力します。

```
kinit administrator@EXAMPLE.COM
```

kinit コマンドは、Active Directory 管理者アカウントおよび Kerberos レルムを参照する Kerberos 初期化スクリプトです。Active Directory には Kerberos チケットが必要なため、kinit はクライアント/サーバー認証用の Kerberos チケット保証チケットを取得してキャッシュします。Kerberos、/etc/krb5.conf ファイル、および kinit コマンドの詳細は、「Kerberos」 を参照してください。

Active Directory サーバー(windows1.example.com)に参加するには、メンバーサーバーで root で以下のコマンドを入力します。

```
net ads join -S windows1.example.com -U administrator%password
```

マシン windows1 は対応する Kerberos レルム(kinit コマンド成功)で自動的に確認されるため、net コマンドは必要な管理者アカウントとパスワードを使用して Active Directory サーバーに接続します。これにより、Active Directory に適切なマシンアカウントが作成され、Samba ドメインメンバーサーバーにドメインに参加するパーミッションが付与されます。

注記

security = ads and not security = user が使用されるため、smbpasswd などのローカルパスワードバックエンドは必要ありません。security = ads をサポートしない古いクライアントは、security = domain が設定されたかのように認証されます。この変更は機能に影響を与えず、ドメインにないローカルユーザーを許可します。

22.6.2.2. Windows NT4 ベースのドメインメンバーサーバー

以下の smb.conf ファイルは、Windows NT4 ベースのドメインメンバーサーバーを実装するために必要な設定例を示しています。NT4 ベースのドメインのメンバーサーバーになるのは、Active Directory への接続に似ています。主な違いは、NT4 ベースのドメインは認証方法で Kerberos を使用しないことです。これにより、smb.conf ファイルが容易になります。この場合、Samba メンバーサーバーは NT4 ベースのドメインサーバーへのパススルーとして機能します。

```
[global]
workgroup = DOCS
netbios name = DOCS_SRV
security = domain
[homes]
comment = Home Directories
valid users = %S
read only = No
browseable = No
[public]
comment = Data
path = /export
force user = docsbot
force group = users
guest ok = Yes
```

Samba をドメインメンバーサーバーとして使用することは、多くの状況で役に立ちます。Samba サーバーに、ファイルやプリンターの共有以外に、他の用途がある場合もあります。ドメイン環境で Linux のみのアプリケーションを使用する必要があるインスタンスで、Samba をドメインメンバーサーバーに設定することが有益です。管理者は、Windows ベースではない場合でも、ドメイン内のすべてのマシンを追跡します。Windows ベースのサーバーハードウェアが非推奨になると、smb.conf ファイルを変更してサーバーを Samba ベースの PDC に変換できます。Windows NT ベースのサーバーが Windows 2000/2003 にアップグレードされると、smb.conf ファイルは必要に応じてインフラストラクチャーの変更を組み込むように簡単に変更可能です。

重要な影響

smb.conf ファイルを設定したら、root で以下のコマンドを入力して、Samba を起動する前 にドメインに参加します。

```
net rpc join -U administrator%password
```

ドメインサーバーのホスト名を指定する -S オプションは、net rpc join コマンドで記述する必要はないことに注意してください。Samba は、明示的に記載するのではなく、smb.conf ファイルの workgroup ディレクティブで指定されたホスト名を使用します。

22.6.3. ドメインコントローラー

Windows NT のドメインコントローラーは、Linux 環境の Network Information Service (NIS)サーバーと機能的に似ています。ドメインコントローラーと NIS サーバーは、ホストユーザー/グループ情報データベースと関連サービスの両方を提供します。ドメインコントローラーは主にセキュリティーに使用されます。これには、ドメインリソースにアクセスするユーザーの認証が含まれます。ユーザー/グループデータベースの整合性を維持するサービスは、Security Account Manager (SAM)と呼ばれます。SAM データベースは Windows と Linux の Samba ベースのシステム間で異なる方法で保存されるため、SAM レプリケーションは実現できず、プラットフォームを PDC/BDC 環境で混在させることはできません。

Samba 環境では、1 つの PDC と 0 個以上の BDC しか存在できません。

重要な影響

Samba は、Samba/Windows ドメインコントローラーの混合環境では存在できません(Samba は Windows PDC の BDC にすることも、その逆もできません)。または、Samba PDC と BDC を共存させる こともでき ます。

22.6.3.1. tdbsamを使用したプライマリードメインコントローラー(PDC)

Samba PDC の最も単純な実装で最も一般的な実装は、tdbsam パスワードデータベースバックエンドを使用します。エージング smbpasswd バックエンドを置き換えることが予定されています。tdbsam には、「Samba アカウント情報データベース」 で詳細に説明されている多くの改良点があります。passdb backend ディレクティブは、PDC に使用するバックエンドを制御します。

```
[global]
workgroup = DOCS
netbios name = DOCS_SRV
passdb backend = tdbsam
security = user
add user script = /usr/sbin/useradd -m "%u"
delete user script = /usr/sbin/userdel -r "%u"
add group script = /usr/sbin/groupadd "%g"
delete group script = /usr/sbin/groupdel "%g"
add user to group script = /usr/sbin/usermod -G "%g" "%u"
add machine script = /usr/sbin/useradd -s /bin/false -d /dev/null  -g machines "%u"
# The following specifies the default logon script
# Per user logon scripts can be specified in the user
# account using pdbedit logon script = logon.bat
# This sets the default profile path.
# Set per user paths with pdbedit
logon drive = H:
domain logons = Yes
os level = 35
preferred master = Yes
domain master = Yes
[homes]
 comment = Home Directories
```

```
 valid users = %S
 read only = No
[netlogon]
 comment = Network Logon Service
 path = /var/lib/samba/netlogon/scripts
 browseable = No
 read only = No
# For profiles to work, create a user directory under the
# path shown.
mkdir -p /var/lib/samba/profiles/john
[Profiles]
 comment = Roaming Profile Share
 path = /var/lib/samba/profiles
 read only = No
 browseable = No
 guest ok = Yes
 profile acls = Yes
# Other resource shares ... ...
```

tdbsam を使用する機能 PDC システムを提供するには、以下の手順に従います。

1. 上記の例で示すように、smb.conf ファイルの設定を使用します。

2. root ユーザーを Samba パスワードデータベースに追加します。

```
smbpasswd -a root
Provide the password here.
```

3. smb サービスを起動します。

4. すべてのプロファイル、ユーザー、および netlogon ディレクトリーが作成されていることを確認します。

5. ユーザーがメンバーになることができるグループを追加します。

```
groupadd -f users
groupadd -f nobody
groupadd -f ntadmins
```

6. UNIX グループをそれぞれの Windows グループに関連付けます。

```
net groupmap add ntgroup="Domain Users" unixgroup=users
net groupmap add ntgroup="Domain Guests" unixgroup=nobody
net groupmap add ntgroup="Domain Admins" unixgroup=ntadmins
```

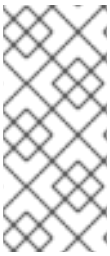
7.

ユーザーまたはグループにアクセス権限を付与します。たとえば、Samba ドメインコントローラーのドメインにクライアントマシンを追加する権利を Domain Admins グループのメンバーに付与するには、以下のコマンドを実行します。

```
net rpc rights grant 'DOCS\Domain Admins' SetMachineAccountPrivilege -S PDC -U root
```

Windows システムは、ドメインユーザー などのドメイングループにマッピングされるプライマリーグループを使用することが推奨されます。

Windows グループとユーザーは同じ名前空間を使用するため、UNIX の場合のようにグループと同じ名前のユーザーが存在することはできません。



注記

複数のドメインコントローラーが必要な場合や、250 以上のユーザーがある場合は、tdbsam 認証バックエンドを使用し ない でください。この場合、LDAP が推奨されます。

22.6.3.2. Active Directory を使用したプライマリードメインコントローラー(PDC)

Samba を Active Directory のメンバーにすることが可能ですが、Samba が Active Directory ドメインコントローラーとして動作することはできません。

22.7. SAMBA のセキュリティーモード

Samba には、共有レベル と ユーザー レベルの 2 つのタイプがあり、まとめてセキュリティーレベル として知られています 。共有レベルのセキュリティーは 1 つの方法でのみ実装できますが、ユーザーレベルのセキュリティーは 4 つの方法のいずれかで実装できます。セキュリティーレベルを実装するさまざまな方法は、セキュリティーモード と呼ばれ ます 。

22.7.1. ユーザーレベルのセキュリティー

ユーザーレベルのセキュリティーは Samba のデフォルト設定です。security = user ディレクティブが smb.conf ファイルにリストされていない場合でも、Samba により使用されます。サーバーがクライアントのユーザー名/パスワードを受け入れると、クライアントは各インスタンスにパスワードを指定せずに複数の共有をマウントできます。Samba は、セッションベースのユーザー名/パスワード要求を受け入れることもできます。クライアントは、ログオンごとに一意の UID を使用して、複数の認証コンテキストを維持します。

smb.conf では、ユーザーレベルのセキュリティーを設定する security = user ディレクティブは次のとおりです。

```
[GLOBAL]
...
security = user
...
```

以下のセクションでは、ユーザーレベルのセキュリティーの他の実装を説明します。

22.7.1.1. ドメインセキュリティーモード（ユーザーレベルのセキュリティー）

ドメインセキュリティーモードでは、Samba サーバーにマシンアカウント（ドメインセキュリティー信頼アカウント）があり、すべての認証要求がドメインコントローラーに渡されます。smb.conf で以下のディレクティブを使用して、Samba サーバーがドメインメンバーサーバーに追加されます。

```
[GLOBAL]
...
security = domain
workgroup = MARKETING
...
```

22.7.1.2. Active Directory セキュリティーモード（ユーザーレベルのセキュリティー）

Active Directory 環境をお持ちの場合は、ドメインをネイティブの Active Directory メンバーとして参加させることができます。セキュリティーポリシーで NT 互換認証プロトコルの使用が制限されていても、Samba サーバーは Kerberos を使用して ADS に参加できます。Active Directory メンバーモードの Samba は、Kerberos チケットを受け入れることができます。

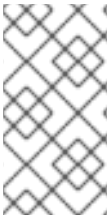
smb.conf では、以下のディレクティブによって Samba が Active Directory メンバーサーバーになります。

```
[GLOBAL]
...
security = ADS
```

```
realm = EXAMPLE.COM
password server = kerberos.example.com
...
```

22.7.1.3. サーバーセキュリティーモード（ユーザーレベルのセキュリティー）

Samba がドメインメンバーサーバーとして機能できない場合、サーバーセキュリティーモードが使用されていました。



注記

セキュリティーの欠点が多数あるため、このモードを使用しない ことを強く推奨します。

smb.conf では、次のディレクティブにより、Samba がサーバーセキュリティーモードで動作できるようになります。

```
[GLOBAL]
...
encrypt passwords = Yes
security = server
password server = "NetBIOS_of_Domain_Controller"
...
```

22.7.2. 共有レベルのセキュリティー

共有レベルのセキュリティーでは、サーバーはクライアントからの明示的なユーザー名なしでパスワードのみを受け入れます。サーバーは、ユーザー名とは関係なく、各共有のパスワードを想定します。Microsoft Windows クライアントが共有レベルのセキュリティーサーバーとの互換性の問題があるという最近の報告があります。Samba 開発者は、共有レベルのセキュリティーの使用を強く推奨していません。

smb.conf では、共有レベルのセキュリティーを設定する security = share ディレクティブは次のとおりです。

```
[GLOBAL]
...
security = share
...
```

22.8. SAMBA アカウント情報データベース

Samba の最新リリースには、以前は利用できない新しいパスワードデータベースバックエンドなど、多くの新機能が提供されています。Samba バージョン 3.0.0 は、以前のバージョンの Samba で使用されたすべてのデータベースに完全に対応します。ただし、サポートされるバックエンドの多くは、実稼働環境での使用には適していない場合があります。

以下は、Samba で使用できる異なるバックエンドの一覧です。ここに記載されていないその他のバックエンドも利用できます。

プレインテキスト

プレーンテキストのバックエンドは、/etc/passwd タイプのバックエンドではありません。プレーンテキストのバックエンドでは、クライアントとサーバー間ですべてのユーザー名とパスワードが暗号化されずに送信されます。この方法は安全性が非常に低く、いずれの方法でも使用することは推奨されません。プレーンテキストのパスワードを使用して Samba サーバーに接続する異なる Windows クライアントは、このような認証方法をサポートできません。

smbpasswd

以前の Samba パッケージで使用される一般的なバックエンドでは、smbpasswd バックエンドは MS Windows LanMan アカウントと NT アカウントを含むプレーンテキストの ASCII テキストレイアウトを使用し、暗号化されたパスワード情報を使用します。smbpasswd バックエンドには、Windows NT/2000/2003 SAM 拡張制御のストレージがありません。smbpasswd バックエンドは、NT ベースのグループの RID などの Windows 情報を適切に拡張したり、Windows 情報を保持することができないため推奨されません。tdbsam バックエンドは、小規模なデータベース(250 ユーザー)で使用するためにこれらの問題を解決しますが、まだエンタープライズクラスソリューションではありません。

ldapsam_compat

ldapsam_compat バックエンドを使用すると、アップグレードされたバージョンの Samba で使用する OpenLDAP サポートを継続できます。このオプションは通常、Samba 3.0 に移行する際に使用されます。

tdbsam

tdbsam バックエンドは、ローカルサーバー、ビルトインデータベースレプリケーションを必要としないサーバー、および LDAP のスケーラビリティーや複雑さを必要としないサーバーに最適なデータベースバックエンドを提供します。tdbsam バックエンドには、smbpasswd データベース情報と、以前に除外された SAM 情報が含まれます。拡張 SAM データを含めることで、Samba は Windows NT/2000/2003 ベースのシステムと同じアカウントおよびシステムアクセス制御を実装できます。

250 ユーザーには、最大で tdbsam バックエンドが推奨されます。大規模な組織では、スケー

ラビリティーとネットワークインフラストラクチャーに関する懸念があるため、Active Directory または LDAP の統合が必要です。

ldapsam

ldapsam バックエンドは、Samba に最適な分散アカウントインストール方法を提供します。LDAP は、OpenLDAP slurpd デーモンを使用して、データベースを任意の数のサーバーに複製できることが最適です。LDAP データベースは軽量でスケーラブルなため、大規模な企業により推奨されます。

以前のバージョンの Samba から 3.0 にアップグレードする場合は、/usr/share/doc/samba-<version> /LDAP/samba.schema が変更されたことに注意してください。このファイルには、属性構文の定義 と、ldapsam バックエンドが適切に機能するために必要な objectclass 定義 が含まれます。

そのため、Samba サーバーに ldapsam バックエンドを使用している場合は、このスキーマファイルを含めるように slapd を設定する必要があります。これを行う方法については、「/etc/openldap/schema/ ディレクトリー」 を参照してください。

注記

ldapsam バックエンドを使用する場合は、openldap-server パッケージがインストールされている必要があります。

mysqlsam

mysqlsam バックエンドは MySQL ベースのデータベースバックエンドを使用します。これは、MySQL をすでに実装しているサイトに便利です。現時点では、mysqlsam は Samba とは別のモジュールにパックされるため、Samba では正式にサポートされていません。

22.9. SAMBA ネットワークブラウザー

ネットワーク参照に より、Windows サーバーと Samba サーバーを Windows Network Neighborhood に表示できるようになります。ネットワーク Neighborhood 内では、アイコンはサーバーとして表され、開いている場合は、利用可能なサーバーの共有とプリンターが表示されます。

ネットワーク参照機能には、TCP/IP での NetBIOS が必要です。NetBIOS ベースのネットワークはブロードキャスト(UDP)メッセージングを使用して、参照リストの管理を行います。NetBIOS と WINS

を TCP/IP ホスト名解決のプライマリーメソッドとしてない場合、静的ファイル(/etc/hosts)や DNS などの他の方法を使用する必要があります。

ドメインマスターブラウザーは、すべてのサブネット上のローカルマスターブラウザーからのブラウズリストを照合し、ワークグループとサブネット間で参照できるようにします。また、ドメインマスターブラウザーは、独自のサブネット用のローカルマスターブラウザーを使用することが推奨されます。

22.9.1. ドメインのブラウジング

デフォルトでは、ドメインの Windows サーバー PDC もそのドメインのドメインマスターブラウザーです。このタイプの状況では、Samba サーバーをドメインマスターサーバーとして設定することはできません。

Windows サーバー PDC を含まないサブネットでは、Samba サーバーをローカルマスターブラウザーとして実装できます。ドメインコントローラー環境でローカルマスターブラウザー（またはまったく参照しない）に smb.conf を設定することは、ワークグループ設定と同じです。

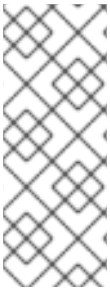
22.9.2. WINS (Windows Internetworking Name Server)

Samba サーバーまたは Windows NT サーバーのいずれかが WINS サーバーとして機能します。NetBIOS が有効な状態で WINS サーバーを使用すると、UDP ユニキャストをルーティングでき、ネットワーク全体で名前解決が可能になります。WINS サーバーがないと、UDP ブロードキャストはローカルサブネットに制限されるため、他のサブネット、ワークグループ、またはドメインにルーティングすることはできません。WINS レプリケーションが必要な場合は、Samba が現在 WINS レプリケーションをサポートしていないため、Samba をプライマリー WINS サーバーとして使用しないでください。

NT/2000/2003 サーバーと Samba 環境では、Microsoft WINS 機能を使用することが推奨されます。Samba のみの環境では、WINS に Samba サーバーを 1 つだけ 使用することが推奨されます。

以下は、Samba サーバーが WINS サーバーとして機能する smb.conf ファイルの例です。

```
[global]
wins support = Yes
```



ヒント

すべてのサーバー(Samba を含む)は、NetBIOS 名を解決するために WINS サーバーに接続する必要があります。WINS を使用しないと、ローカルサブネットでのみ参照が行われます。さらに、ドメイン全体のリストを取得した場合でも、WINS を使用せずにホスト をクライアントで解決できません。

22.10. CUPS 印刷サポートのある SAMBA

Samba を使用すると、クライアントマシンは Samba サーバーに接続されたプリンターを共有できます。さらに、Samba を使用すると、クライアントマシンは Linux に組み込まれているドキュメントを Windows プリンター共有に送信できます。Red Hat Enterprise Linux で機能する印刷システムもありますが、Samba と密接に統合されるため、CUPS (Common UNIX Print System)が推奨される印刷システムです。

22.10.1. 単純な smb.conf 設定

以下の例は、CUPS サポート向けの非常に基本的な smb.conf 設定を示しています。

```
[global]
load printers = Yes
printing = cups
printcap name = cups
[printers]
comment = All Printers
path = /var/spool/samba/print
printer = IBMInfoP
browseable = No
public = Yes
guest ok = Yes
writable = No
printable = Yes
printer admin = @ntadmins
[print$]
comment = Printer Drivers Share
path = /var/lib/samba/drivers
write list = ed, john
printer admin = ed, john
```

他の印刷設定も可能です。機密ドキュメントを印刷するためのセキュリティーおよびプライバシーを追加するために、ユーザーは、パブリックパスにない独自の印刷スプールを持つことができます。ジョブが失敗すると、他のユーザーはファイルにアクセスできません。

print$ 共有には、ローカルで利用できない場合にクライアントがアクセスするプリンタードライバーが含まれます。print$ 共有はオプションであり、組織によっては必要ない可能性があります。

browseable を Yes に設定すると、Samba サーバーがドメイン/ワークグループに正しく設定されていると、Windows Network Neighborhood でプリンターを表示できます。

22.11. SAMBA ディストリビューションプログラム

findsmb

findsmb <subnet_broadcast_address>

findsmb プログラムは、特定のサブネット上の SMB 対応システムに関する情報を報告する Perl スクリプトです。サブネットを指定しないと、ローカルサブネットが使用されます。表示される項目には、IP アドレス、NetBIOS 名、ワークグループ、またはドメイン名、オペレーティングシステム、およびバージョンが含まれます。

以下の例は、システムで有効なユーザーとして findsmb を実行する出力を示しています。

```
~]# findsmb
IP ADDR       NETBIOS NAME  WORKGROUP/OS/VERSION
------------------------------------------------------------------
10.1.59.25    VERVE         [MYGROUP] [Unix] [Samba 3.0.0-15]
10.1.59.26    STATION22     [MYGROUP] [Unix] [Samba 3.0.2-7.FC1]
10.1.56.45    TREK         +[WORKGROUP] [Windows 5.0] [Windows 2000 LAN Manager]
10.1.57.94    PIXEL         [MYGROUP] [Unix] [Samba 3.0.0-15]
10.1.57.137   MOBILE001     [WORKGROUP] [Windows 5.0] [Windows 2000 LAN Manager]
10.1.57.141   JAWS         +[KWIKIMART] [Unix] [Samba 2.2.7a-security-rollup-fix]
10.1.56.159   FRED         +[MYGROUP] [Unix] [Samba 3.0.0-14.3E]
10.1.59.192   LEGION       *[MYGROUP] [Unix] [Samba 2.2.7-security-rollup-fix]
10.1.56.205   NANCYN        +[MYGROUP] [Unix] [Samba 2.2.7a-security-rollup-fix]
```

net

net <protocol> <function> <misc_options> <target_options>

net ユーティリティーは、Windows および MS-DOS に使用される net ユーティリティーと似ています。最初の引数は、コマンドの実行時に使用するプロトコルを指定するために使用されます。&lt ;protocol&gt ; オプションは、サーバー接続のタイプを指定するための ads、rap、または rpc にすることができます。Active Directory は ads を使用し、Win9x/NT3 は rap を使用し、Windows NT4/2000/2003 は rpc を使用します。プロトコルを省略すると、net は自動的にそれを判断しようとします。

以下の例では、wakko という名前のホストで利用可能な共有の一覧を表示します。

```
~]# net -l share -S wakko
Password:

Enumerating shared resources (exports) on remote server:

Share name   Type     Description
----------   ----     -----------
data         Disk     Wakko data share
tmp          Disk     Wakko tmp share
IPC$         IPC      IPC Service (Samba Server)
ADMIN$       IPC      IPC Service (Samba Server)
```

以下の例では、wakko という名前のホストの Samba ユーザーの一覧を表示します。

```
~]# net -l user -S wakko
root password:
User name            Comment
-----------------------------
andriusb             Documentation
joe                  Marketing
lisa                 Sales
```

nmblookup

nmblookup <options> <netbios_name>

nmblookup プログラムは、NetBIOS 名を IP アドレスに解決します。プログラムは、ターゲットマシンが応答するまで、ローカルサブネットでクエリーをブロードキャストします。

以下に例を示します。

```
~]# nmblookup trek
querying trek on 10.1.59.255
10.1.56.45 trek<00>
```

pdbedit

pdbedit <options>

pdbedit プログラムは、SAM データベースにあるアカウントを管理します。smbpasswd、LDAP、NIS+、tdb データベースライブラリーなど、すべてのバックエンドがサポートされます。

以下は、ユーザーの追加、削除、および一覧表示の例です。

```
~]# pdbedit -a kristin
new password:
retype new password:
Unix username:        kristin
NT username:
Account Flags:        [U          ]
User SID:             S-1-5-21-1210235352-3804200048-1474496110-2012
Primary Group SID:    S-1-5-21-1210235352-3804200048-1474496110-2077
Full Name: Home Directory:        \\wakko\kristin
HomeDir Drive:
Logon Script:
Profile Path:         \\wakko\kristin\profile
Domain:               WAKKO
Account desc:
Workstations: Munged
dial:
Logon time:           0
Logoff time:          Mon, 18 Jan 2038 22:14:07 GMT
Kickoff time:         Mon, 18 Jan 2038 22:14:07 GMT
Password last set:    Thu, 29 Jan 2004 08:29:28
GMT Password can change:  Thu, 29 Jan 2004 08:29:28 GMT
Password must change: Mon, 18 Jan 2038 22:14:07 GMT

~]# pdbedit -v -L kristin
Unix username:        kristin
NT username:
Account Flags:        [U          ]
User SID:             S-1-5-21-1210235352-3804200048-1474496110-2012
Primary Group SID:    S-1-5-21-1210235352-3804200048-1474496110-2077
Full Name:
Home Directory:       \\wakko\kristin
HomeDir Drive:
Logon Script:
Profile Path:         \\wakko\kristin\profile
Domain:               WAKKO
Account desc:
Workstations: Munged
dial:
Logon time:           0
Logoff time:          Mon, 18 Jan 2038 22:14:07 GMT
Kickoff time:         Mon, 18 Jan 2038 22:14:07 GMT
Password last set:    Thu, 29 Jan 2004 08:29:28 GMT
Password can change:  Thu, 29 Jan 2004 08:29:28 GMT
Password must change: Mon, 18 Jan 2038 22:14:07 GMT

~]# pdbedit -L
andriusb:505:
joe:503:
lisa:504:
kristin:506:

~]# pdbedit -x joe
```

```
~]# pdbedit -L

andriusb:505: lisa:504: kristin:506:
```

rpcclient

rpcclient <server> <options>

rpcclient プログラムは、システム管理用の Windows 管理グラフィカルユーザーインターフェイス (GUI)へのアクセスを提供する Microsoft RPC を使用して管理コマンドを実行します。これは、Microsoft RPC の完全な複雑さを理解する上級ユーザーが最もよく使用されます。

smbcacls

smbcacls <//server/share> <filename> <options>

smbcacls プログラムは、Samba サーバーが共有するファイルおよびディレクトリーの Windows ACL を変更します。

smbclient

smbclient <//server/share> <password> <options>

smbclient プログラムは、ftp と同様の機能を提供する汎用 UNIX クライアントです。

smbcontrol

smbcontrol -i <options>

smbcontrol <options> <destination> <messagetype> <parameters>

smbcontrol プログラムは、smbd デーモンまたは nmbd デーモンを実行する制御メッセージを送信します。smbcontrol -i を実行すると、空の行または q が入力されるまで、コマンドを対話的に実行します。

smbpasswd

smbpasswd <options> <username> <password>

smbpasswd プログラムは、暗号化されたパスワードを管理します。このプログラムはスーパーユーザーが実行し、ユーザーのパスワードを変更したり、通常のユーザーが独自の Samba パスワードを変更したりできます。

smbspool

smbspool <job> <user> <title> <copies> <options> <filename>

smbspool プログラムは、Samba への CUPS 互換の印刷インターフェイスです。CUPS プリンターで使用するために設計された smbspool は CUPS 以外のプリンターでも機能します。

smbstatus

smbstatus <options>

smbstatus プログラムは、Samba サーバーへの現在の接続のステータスを表示します。

smbtar

smbtar <options>

smbtar プログラムは、Windows ベースの共有ファイルおよびディレクトリーのバックアップおよび復元をローカルテープアーカイブに対して実行します。tar コマンドと同様ですが、これら 2 つは互換性がありません。

testparm

testparm <options> <filename> <hostname IP_address>

testparm プログラムは、smb.conf ファイルの構文をチェックします。smb.conf ファイルがデフォルトの場所(/etc/samba/smb.conf)にある場合は、場所を指定する必要はありません。testparm プログラムにホスト名および IP アドレスを指定すると、hosts.allow および host.deny ファイルが正しく設定されていることを確認します。testparm プログラムは、テスト後に smb.conf ファイルとサーバーのロール（スタンバイ、ドメインなど）の概要も表示します。これは、コメントを除外し、経験のある管理者が読み取るための情報を簡潔に表示するため、デバッグに役立ちます。

以下に例を示します。

```
~]# testparm
```

```
Load smb config files from /etc/samba/smb.conf
Processing section "[homes]"
Processing section "[printers]"
Processing section "[tmp]"
Processing section "[html]"
Loaded services file OK.
Server role: ROLE_STANDALONE
Press enter to see a dump of your service definitions

<enter>
# Global parameters
[global]
 workgroup = MYGROUP
 server string = Samba Server
 security = SHARE
 log file = /var/log/samba/%m.log
 max log size = 50
 socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
 dns proxy = No
[homes]
 comment = Home Directories
 read only = No
 browseable = No
[printers]
 comment = All Printers
 path = /var/spool/samba
 printable = Yes
 browseable = No
[tmp]
 comment = Wakko tmp
 path = /tmp
 guest only = Yes
[html]
 comment = Wakko www
 path = /var/www/html
 force user = andriusb
 force group = users
 read only = No
 guest only = Yes
```

wbinfo

wbinfo <options>

wbinfo プログラムは、winbindd デーモンからの情報を表示します。wbinfo が機能するには、winbindd デーモンを実行する必要があります。

22.12. 関連情報

以下のセクションでは、Samba をより詳細に調べる方法を説明します。

22.12.1. インストールされているドキュメント

- /usr/share/doc/samba-<version-number& gt;/ - Samba ディストリビューションに含まれるすべての追加ファイル。これには、すべてのヘルパースクリプト、サンプル設定ファイル、およびドキュメントが含まれます。

  また、このディレクトリーに『は、公式 Samba-3 HOWTO-Collection および『Samba-』 3 by Example』（どちらも以下で引用）のオンラインバージョンも含まれます。

22.12.2. 関連書籍

- John H. Terpstra および Jelmer R. Vernooij; Prentice Hall 『- The official Samba-3 HOWTO-Collection』 by John H. Terpstra and Jelmer R. Vernooij; Prentice Hall - The official Samba-3 HOWTO-Collection by John H. Terpstra and Jelmer R. Vernooij; Prentice Hall - The official Samba-3 documentation as issued by the Samba development teamこれは、ステップ別のガイドよりも多くのリファレンスガイドです。

- 『Samba-3 by John H. Terpstra; Prentice Hall - これは、OpenLDAP、DNS、DHCP、および印刷設定ファイルの詳細な例』 を説明する Samba 開発チームが発行する別の公式リリースです。これには、実際の実装に役立つステップごとの関連情報があります。

- 『Samba の 2nd Edition by Jay T's、Robert Eckstein、David Collier-Brown、O'Reilly - 包括的な参考資料を含む上級ユーザー向けの優れた』 リソースです。

22.12.3. 便利な Web サイト

- http://www.samba.org/ - Samba ディストリビューションのホームページと、Samba 開発チームが作成したすべての公式ドキュメントです。HTML 形式や PDF 形式では多くのリソースが利用できますが、購入のみが可能です。これらのリンクの多くは Red Hat Enterprise Linux 固有のものではありませんが、一部の概念が適用される場合があります。

- http://samba.org/samba/archives.html - Samba コミュニティーのアクティブな電子メール一覧。リストアクティビティーのレベルが高いため、ダイジェストモードを有効にすることが推奨されます。

- Samba newsgroups - NNTP プロトコルを使用する gmane.org などの Samba スレッドの newsgroups も利用可能です。これは、メーリングリストのメールを受信する代替手段です。

第23章 DYNAMIC HOST CONFIGURATION PROTOCOL (DHCP)

DHCP (Dynamic Host Configuration Protocol: 動的ホスト設定プロトコル) は、クライアントマシンに TCP/IP 情報を自動的に割り当てるネットワークプロトコルです。各 DHCP クライアントは、中央の中央の DHCP サーバーに接続し、クライアントのネットワーク設定(IP アドレス、ゲートウェイ、DNS サーバーなど)を返します。

23.1. DHCP を使用する理由

DHCP は、クライアントネットワークインターフェイスの自動設定に便利です。クライアントシステムを設定する際に、管理者は IP アドレス、ネットマスク、ゲートウェイ、または DNS サーバーを指定する代わりに、DHCP を選択します。クライアントはこの情報を DHCP サーバーから取得します。DHCP は、管理者が多数のシステムの IP アドレスを変更する場合にも便利です。すべてのシステムを再設定する代わりに、サーバー上の 1 つの DHCP 設定ファイルを編集して、新しい IP アドレスセットを編集することができます。組織の DNS サーバーが変更されると、DHCP クライアントではなく、DHCP サーバーで変更が行われます。管理者がネットワークを再起動するか、クライアントを再起動すると、変更が有効になります。

機能している DHCP サーバーをネットワークに正しく接続している場合は、ラップトップおよびその他のモバイルコンピューターユーザーがこれらのデバイスをオフィスからオフィスに移動できます。

23.2. DHCP サーバーの設定

dhcp パッケージには、ISC DHCP サーバーが含まれています。まず、スーパーユーザーとして パッケージをインストールします。

```
~]# yum install dhcp
```

dhcp パッケージをインストールすると、/etc/dhcpd.conf ファイルが作成されます。これは単に空の設定ファイルになります。

```
~]# cat /etc/dhcpd.conf
#
# DHCP Server Configuration file.
#   see /usr/share/doc/dhcp*/dhcpd.conf.sample
```

サンプル設定ファイルは、/usr/share/doc/dhcp-<version>/dhcpd.conf.sample にあります。このファイルは、/etc/dhcpd.conf の設定に役立ちます。これについては、以下で説明します。

また、DHCP は /var/lib/dhcpd/dhcpd.leases ファイルを使用してクライアントのリースデータベースを保存します。詳細は、「リースデータベース」 を参照してください。

23.2.1. 設定ファイル

DHCP サーバーを設定する最初のステップは、クライアントのネットワーク情報を保存する設定ファイルを作成します。このファイルを使用して、クライアントシステムのオプションとグローバルオプションを宣言します。

設定ファイルには追加のタブや空白行が含まれているため、簡単に書式を整えることができます。キーワードは大文字と小文字を区別せず、ハッシュ記号(#)で始まる行はコメントとみなされます。

現在、2 つの DNS 更新スキームが実装されています。アドホック DNS 更新モードと、DHCP-DNS インタラクションのドラフト更新モードです。これら 2 つが Internet Engineering Task Force (IETF) 標準プロセスの一部として許可される場合、3 番目のモード（標準の DNS 更新メソッド）があります。これらのスキームとの互換性を確保するために、DNS サーバーを設定する必要があります。バージョン 3.0b2pl11 以前のバージョンでは、アドホックモードを使用していましたが、非推奨になっています。同じ動作を維持するには、以下の行を設定ファイルの上部に追加します。

```
ddns-update-style ad-hoc;
```

推奨されるモードを使用するには、以下の行を設定ファイルの上部に追加します。

```
ddns-update-style interim;
```

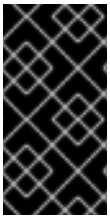
さまざまなモードの詳細は、dhcpd.conf の man ページを参照してください。

設定ファイルのステートメントには、次のような 2 つのタイプがあります。

- パラメーター: タスクの実行方法、タスクを実行するかどうか、クライアントに送信するネットワーク設定のオプションを規定します。
- 宣言 - ネットワークトポロジーの記述、クライアントの記述、クライアントのアドレス指定、宣言グループへのパラメーターグループの適用を行います。

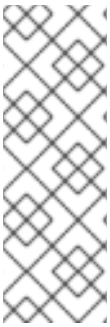
キーワードオプションから始まるパラメーターは、オプション と呼ばれます。これらのオプションは DHCP オプションを制御しますが、パラメーターはオプションではない値を設定するか、DHCP サーバーの動作を制御します。

中括弧({ })で囲まれたセクションの前に宣言されたパラメーター（オプションを含む）はグローバルパラメーターとみなされます。グローバルパラメーターは、これ以降のすべてのセクションに適用されます。



重要な影響

設定ファイルが変更された場合、コマンド service dhcpd によって DHCP デーモンが再起動されるまで変更は反映されません。



ヒント

毎回 DHCP 設定ファイルを変更してサービスを再起動する代わりに、omshell コマンドを使用すると、DHCP サーバーへの接続、クエリー、設定の変更をインタラクティブに行うことができます。omshell を使用すると、DHCP サーバーの実行中でも変更を行うことができます。omshell の詳細は、omshell の man ページを参照してください。

例23.1「サブネットの宣言」 では、ルーター、subnet-mask、domain-name、domain-name-servers、および time-offset オプションは、その下に宣言された host ステートメントに使用されます。

また、サブネット を宣言することもできます。サブネット 宣言は、ネットワーク内のすべてのサブネットに含める必要があります。そうでない場合、DHCP サーバーは起動に失敗します。

この例では、サブネット内のすべての DHCP クライアントに対してグローバルオプションと 範囲 が宣言されています。クライアントには、の 範囲 内の IP アドレスが割り当てられます。

例23.1 サブネットの宣言

```
subnet 192.168.1.0 netmask 255.255.255.0 {
        option routers                  192.168.1.254;
        option subnet-mask              255.255.255.0;

        option domain-name              "example.com";
        option domain-name-servers      192.168.1.1;

        option time-offset              -18000;     # Eastern Standard Time

 range 192.168.1.10 192.168.1.100;
}
```

同じ物理ネットワークを共有するすべてのサブネットは、例23.2「Shared-network 宣言」 に示されるように shared-network 宣言内で宣言する必要があります。shared-network 内のパラメーター（ただし囲まれた subnet 宣言の外）は、グローバルパラメーターとみなされます。shared-network の名前は、'test-lab' というタイトルを使用してテストラボ環境のすべてのサブネットを説明するなど、ネットワークの説明的なタイトルである必要があります。

例23.2 Shared-network 宣言

```
shared-network name {
   option domain-name              "test.redhat.com";
   option domain-name-servers      ns1.redhat.com, ns2.redhat.com;
   option routers                  192.168.0.254;
   more parameters for EXAMPLE shared-network
   subnet 192.168.1.0 netmask 255.255.252.0 {
      parameters for subnet
      range 192.168.1.1 192.168.1.254;
   }
   subnet 192.168.2.0 netmask 255.255.252.0 {
      parameters for subnet
      range 192.168.2.1 192.168.2.254;
   }
}
```

例23.3「Group 宣言」 で示すように、group 宣言は、宣言のグループにグローバルパラメーターを適用するために使用されます。たとえば、共有ネットワーク、サブネット、ホストをグループ化することができます。

例23.3 Group 宣言

```
group {
  option routers                  192.168.1.254;
  option subnet-mask              255.255.255.0;

  option domain-name              "example.com";
  option domain-name-servers      192.168.1.1;

  option time-offset              -18000;     # Eastern Standard Time

  host apex {
    option host-name "apex.example.com";
    hardware ethernet 00:A0:78:8E:9E:AA;
    fixed-address 192.168.1.4;
  }

  host raleigh {
    option host-name "raleigh.example.com";
    hardware ethernet 00:A1:DD:74:C3:F2;
    fixed-address 192.168.1.6;
  }
}
```

サブネット内のシステムに動的 IP アドレスをリースする DHCP サーバーを設定するには、例 23.4「Range パラメーター」 を実際の値で変更します。これにより、クライアントのデフォルトのリース時間、最大リース時間、ネットワークの設定値を宣言します。この例では、192.168.1.10 および 192.168.1.100 の 範囲 の IP アドレスをクライアントシステムに割り当てます。

例23.4 Range パラメーター

```
default-lease-time 600;
max-lease-time 7200;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.1.255;
option routers 192.168.1.254;
option domain-name-servers 192.168.1.1, 192.168.1.2;
option domain-name "example.com";

subnet 192.168.1.0 netmask 255.255.255.0 {
  range 192.168.1.10 192.168.1.100;
}
```

ネットワークインターフェイスカードの MAC アドレスに基づいてクライアントに IP アドレスを割り当てるには、host 宣言内の hardware ethernet パラメーターを使用します。例23.5「DHCP を使用した静的 IP アドレス」 で示すように、host apex 宣言は、MAC アドレス 00:A0:78:8E:9E:AA を持つネットワークインターフェイスカードを常に IP アドレス 192.168.1.4 を受け取るように指定します。

オプションのパラメーター host-name を使用して、クライアントにホスト名を割り当てることもできます。

例23.5 DHCP を使用した静的 IP アドレス

```
host apex {
  option host-name "apex.example.com";
  hardware ethernet 00:A0:78:8E:9E:AA;
  fixed-address 192.168.1.4;
}
```

ヒント

提供される設定ファイルのサンプルは開始点として使用でき、カスタム設定オプションをこれに追加できます。これを適切な場所にコピーするには、以下のコマンドを使用します。

```
cp /usr/share/doc/dhcp-<version-number>/dhcpd.conf.sample /etc/dhcpd.conf
```

< version-number> は DHCP バージョン番号に置き換えてください。

オプションステートメントの完全なリストと機能については、**dhcp-options** の **man** ページを参照してください。

23.2.2. リースデータベース

DHCP サーバーでは、**/var/lib/dhcpd/dhcpd.leases** ファイルが **DHCP** クライアントのリースデータベースを保存します。このファイルは変更しないでください。最近割り当てられた各 **IP** アドレスの **DHCP** リース情報は、リースデータベースに自動的に保存されます。この情報には、リースの長さ、**IP** アドレスが割り当てられている場所、リースの開始日と終了日、リースの取得に使用されたネットワークインターフェイスカードの **MAC** アドレスが含まれます。

リースデータベースの時刻はすべて、現地時間でなく協定世界時 **(UTC)** を使用します。

リースデータベースは、サイズが大きくなり過ぎるのを避けるために適宜再作成されます。最初に、すべての既知のリースは一時的なリースデータベースに保存されます。**dhcpd.leases** ファイルの名前は **dhcpd.leases~** に変更され、一時的なリースデータベースが **dhcpd.leases** に書き込まれます。

リースデータベースがバックアップファイルに変更された後、新規ファイルが書き込まれる前に、**DHCP** デーモンが強制終了されるか、システムがクラッシュする可能性があります。この場合、**dhcpd.leases** ファイルは存在しませんが、サービスを起動する必要があります。この際、新規のリースファイルを作成しないでください。作成すると、それまでのリースはすべて失われ、多くの問題が発生します。これを解決する方法は、**dhcpd.leases~** バックアップファイルの名前を **dhcpd.leases** に変更して、デーモンを起動することです。

23.2.3. サーバーの起動と停止

重要な影響

DHCP サーバーの初回起動時には、dhcpd.leases ファイルが存在しないと失敗します。ファイルが存在しない場合には、touch /var/lib/dhcpd/dhcpd.leases コマンドを使用して作成します。

同じサーバーが DNS サーバーとして BIND も実行している場合は、named サービスを開始すると自動的に dhcpd.leases ファイルをチェックするため、この手順は必要ありません。

DHCP サービスを起動するには、コマンド /sbin/service dhcpd start を使用します。DHCP サーバーを停止するには、コマンド /sbin/service dhcpd stop を使用します。

デフォルトでは、DHCP サービスはブート時に起動しません。システムの起動時にデーモンが自動的に起動するように設定するには、18章 を参照してください。

複数のネットワークインターフェイスがシステムに接続されているものの、DHCP サーバーはいずれかのインターフェイスでのみ起動する必要がある場合は、DHCP サーバーがそのデバイスでのみ起動するように設定します。/etc/sysconfig/dhcpd で、DHCPDARGS 一覧にインターフェイスの名前を追加します。

```
# Command line options here
DHCPDARGS=eth0
```

これは、ネットワークカードが 2 つあるファイアウォールマシンで役立ちます。1 つのネットワークカードを DHCP クライアントとして設定して、インターネットへの IP アドレスを取得することができます。他のネットワークカードは、ファイアウォールの背後にある内部ネットワークの DHCP サーバーとして使用できます。内部ネットワークに接続されたネットワークカードのみを指定すると、ユーザーはインターネット経由でデーモンに接続できないので、システムの安全性が向上します。

/etc/sysconfig/dhcpd で指定できるその他のコマンドラインオプションには以下が含まれます。

- *-p & lt;portnum& gt; - dhcpd がリッスンする UDP ポート番号を指定します。デフォルト値はポート 67 です。DHCP サーバーは、指定された UDP ポートよりも大きなポート番号で DHCP クライアントに応答を送信します。たとえば、デフォルトのポート 67 を使用する場合、サーバーはポート 67 でポート 67 でリッスンし、要求とポート 68 のクライアントへの応答をリッスンします。ポートが指定され、DHCP リレーエージェントが使用される場合は、DHCP リレーエージェントがリッスンするのと同じポートを指定する必要があります。詳細は、「DHCP リレーエージェント」 を参照してください。*

- *-f: フォアグラウンドプロセスとしてデーモンを実行します。これは主にデバッグ用に使用されます。*

- *-d: 標準エラー記述子に DHCP サーバーデーモンをログに記録します。これは主にデバッグ用に使用されます。このオプションを指定しないと、ログは /var/log/messages に書き込まれます。*

- *-cf & lt;filename& gt; - 設定ファイルの場所を指定します。デフォルトの場所は /etc/dhcpd.conf です。*

- *-lf & lt;filename& gt; - リースデータベースファイルの場所を指定します。リースデータベースファイルがすでに存在する場合は、DHCP サーバーが起動するたびに同じファイルが使用されることが非常に重要です。このオプションは、実稼働環境以外のマシンでデバッグする目的にのみ使用することが強く推奨されます。デフォルトの場所は /var/lib/dhcpd/dhcpd.leases です。*

- *-q - デーモンの起動時に著作権に関するメッセージ全体を表示しません。*

23.2.4. DHCP リレーエージェント

DHCP リレーエージェント(dhcrelay)を使用すると、DHCP サーバーのないサブネットから他のサブネットにある 1 つ以上の DHCP サーバーに DHCP および BOOTP 要求のリレーを行うことができます。

DHCP クライアントが情報を要求すると、DHCP リレーエージェントは起動時に指定された DHCP サーバーの一覧に要求を転送します。DHCP サーバーが応答を返すと、元の要求を送信したネットワーク上で応答がブロードキャストまたはユニキャストになります。

DHCP リレーエージェントは、インターフェイスが INTERFACES ディレクティブで /etc/sysconfig/dhcrelay に指定されていない限り、すべてのインターフェイスの DHCP 要求をリッスンします。

DHCP リレーエージェントを起動するには、コマンド service dhcrelay start を使用します。

23.3. DHCP クライアントの設定

DHCP クライアントを設定する最初のステップは、カーネルがネットワークインターフェイスカー

ドを認識するようにすることです。ほとんどのカードはインストールプロセス時に認識され、システムはカード用に正しいカーネルモジュールを使用するように設定されています。インストール後にカードが追加されると、Kudzu [8] これを認識し、適切なカーネルモジュールの入力を求めます(http://hardware.redhat.com/hcl/ でハードウェア互換性一覧を確認してください)。インストールプログラムまたは kudzu のいずれかがネットワークカードを認識しない場合は、正しいカーネルモジュールを読み込むことができます（詳細は、45章一般的なパラメーターおよびモジュール を参照してください）。

DHCP クライアントを手動で設定するには、/etc/sysconfig/network ファイルを変更して、/etc/sysconfig/network-scripts ディレクトリー内の各ネットワークデバイスのネットワークおよび設定ファイルを有効にします。このディレクトリーでは、各デバイスに ifcfg-eth0 という名前の設定ファイルがなければなりません。eth0 はネットワークデバイス名です。

/etc/sysconfig/network ファイルには次の行が含まれている必要があります。

```
NETWORKING=yes
```

起動時にネットワークを開始する場合は、NETWORKING 変数を yes に設定する必要があります。

/etc/sysconfig/network-scripts/ifcfg-eth0 ファイルには、次の行が含まれている必要があります。

```
DEVICE=eth0
BOOTPROTO=dhcp
ONBOOT=yes
```

各デバイスを DHCP を使用するように設定するには、設定ファイルが必要です。

network スクリプトのその他のオプションには、以下が含まれます。

- DHCP_HOSTNAME: DHCP サーバーが IP アドレスを受け取る前にホスト名を指定する必要がある場合にのみこのオプションを使用します。(Red Hat Enterprise Linux の DHCP サーバーデーモンは、この機能をサポートしません。)
- PEERDNS= <answer> です。&lt ;answer&gt ; は以下のいずれかになります。
  - はい - /etc/resolv.conf をサーバーの情報で変更します。DHCP を使用している場合は、yes がデフォルトになります。

- No - /etc/resolv.conf は変更しないでください。

- SRCADDR= <address> 。ここで、<address > は送信パケットの指定されたソース IP アドレスです。

- USERCTL= <answer >。ここで、& lt;answer& gt; は以下のいずれかになります。

  - はい - root 以外のユーザーはこのデバイスを制御できます。

  - no: root 以外のユーザーはこのデバイスを制御することができません。

グラフィカルインターフェイスを使用する場合は、17章Network Configuration を参照してください。Network Administration Tool を使用して、DHCP を使用するようにネットワークインターフェイスを設定する手順を参照してください。

ヒント

プロトコルタイミング、リース要件、要求、動的 DNS サポート、エイリアス、およびクライアント側の設定に上書き、先頭に追加するさまざまな値など、クライアントの DHCP オプションの詳細設定については、dhclient および dhclient.conf の man ページを参照してください。

23.4. マルチホーム DHCP サーバーの設定

マルチホーム DHCP サーバーは、複数のネットワーク（つまり複数のサブネット）を提供します。以下の項の例では、複数のネットワークを提供するように DHCP サーバーを設定する方法、リッスンするネットワークインターフェイスを選択する方法、ネットワークを移動するシステムのネットワーク設定の定義方法について詳しく説明します。

変更を行う前に、既存の /etc/sysconfig/dhcpd ファイルおよび /etc/dhcpd.conf ファイルのバックアップを作成してください。

DHCP デーモンは、特に指定がない限り、すべてのネットワークインターフェイスでリッスンします。/etc/sysconfig/dhcpd ファイルを使用して、DHCP デーモンがリッスンするネットワークインターフェイスを指定します。以下の /etc/sysconfig/dhcpd の例では、DHCP デーモンが eth0 インターフェイスおよび eth1 インターフェイスでリッスンするように指定します。

```
DHCPDARGS="eth0 eth1";
```

システムに 3 つのネットワークインターフェイスカード(eth0、eth1、および eth2)があり、DHCP デーモンが eth0 でリッスンすることが望ましい場合は、/etc/sysconfig/dhcpd で eth0 のみを指定します。

```
DHCPDARGS="eth0";
```

以下は、2 つのネットワークインターフェイスが 10.0.0.0/24 ネットワークで、 eth1 が 172.16.0.0 /24 ネットワークにあるサーバー用の基本的な /etc/dhcpd.conf ファイルです。複数の subnet 宣言を使用すると、複数のネットワークに異なる設定を定義できます。

```
ddns-update-style interim;
default-lease-time 600;
max-lease-time 7200;

subnet 10.0.0.0 netmask 255.255.255.0 {
 option subnet-mask 255.255.255.0;
 option routers 10.0.0.1;
 range 10.0.0.5 10.0.0.15;
}

subnet 172.16.0.0 netmask 255.255.255.0 {
 option subnet-mask 255.255.255.0;
 option routers 172.16.0.1;
 range 172.16.0.5 172.16.0.15;

}
```

サブネット 10.0.0.0 ネットマスク 255.255.255.0

DHCP サーバーが提供しているすべてのネットワークには、subnet 宣言が必要です。複数のサブネットには、複数の サブネット 宣言が必要です。DHCP サーバーに サブネット 宣言の範囲にネットワークインターフェイスがない場合、DHCP サーバーはそのネットワークを提供しません。

サブネット宣言が 1 つしかなく、そのサブネット の範囲内にネットワークインターフェイスがない場合、DHCP デーモンは起動に失敗し、以下のようなエラーが /var/log/messages に記録されます。

```
dhcpd: No subnet declaration for eth0 (0.0.0.0).
dhcpd: ** Ignoring requests on eth0.  If this is not what
dhcpd:    you want, please write a subnet declaration
dhcpd:    in your dhcpd.conf file for the network segment
dhcpd:    to which interface eth1 is attached. **
dhcpd:
dhcpd:
dhcpd: Not configured to listen on any interfaces!
```

オプション subnet-mask 255.255.255.0

subnet-mask オプション は、サブネットマスクを定義し、subnet 宣言の netmask 値を上書きします。簡単なケースでは、サブネットとネットマスクの値は同じです。

オプションルーター 10.0.0.1

option routers オプションは、サブネットのデフォルトゲートウェイを定義します。これは、システムが異なるサブネット上の内部ネットワーク、さらには外部ネットワークに届くために必要です。

range 10.0.0.5 10.0.0.15;

range オプションは、利用可能な IP アドレスのプールを指定します。指定した IP アドレスの範囲からアドレスが割り当てられます。

詳細は、dhcpd.conf (5) man ページを参照してください。

エイリアスインターフェイス

エイリアスインターフェイスは DHCP ではサポートされていません。エイリアスインターフェイスが唯一のインターフェイスである場合、/etc/dhcpd.conf で指定された唯一のサブネットでは、DHCP デーモンが起動に失敗します。

23.4.1. ホストの設定

変更を行う前に、既存の /etc/sysconfig/dhcpd ファイルおよび /etc/dhcpd.conf ファイルのバックアップを作成してください。

複数ネットワーク用の単一システムの設定

以下の /etc/dhcpd.conf の例では、2 つのサブネットを作成し、接続するネットワークに応じて同じシステムに IP アドレスを設定します。

```
ddns-update-style interim;
default-lease-time 600;
max-lease-time 7200;

subnet 10.0.0.0 netmask 255.255.255.0 {
 option subnet-mask 255.255.255.0;
 option routers 10.0.0.1;
 range 10.0.0.5 10.0.0.15;
}

subnet 172.16.0.0 netmask 255.255.255.0 {
 option subnet-mask 255.255.255.0;
 option routers 172.16.0.1;
 range 172.16.0.5 172.16.0.15;

}

host example0 {
 hardware ethernet 00:1A:6B:6A:2E:0B;
 fixed-address 10.0.0.20;
}

host example1 {
 hardware ethernet 00:1A:6B:6A:2E:0B;
 fixed-address 172.16.0.20;
}
```

host example0

host 宣言は、IP アドレスなどの単一システムの特定のパラメーターを定義します。複数のホストに特定のパラメーターを設定するには、複数の host 宣言を使用します。

ほとんどの DHCP クライアントは host 宣言の名前を無視するため、他の host 宣言に固有の名前であれば、この名前はどのようなものでも構いません。複数のネットワークに同じシステムを設定するには、host 宣言ごとに異なる名前を使用します。そうしないと、DHCP デーモンが起動に失敗します。システムは、host 宣言の名前ではなく、ハードウェアイーサネット オプションで識別されます。

hardware ethernet psycA:6B:6A:2E:0B;

hardware ethernet オプションは、システムを識別します。このアドレスを見つけるには、必要なシステムで ifconfig コマンドを実行し、HWaddr アドレスを見つけます。

fixed-address 10.0.0.20;

fixed-address オプションは、ハードウェアイーサネット オプションで指定されたシステムに、有効な IP アドレスを割り当てます。このアドレスは、range オプションで指定された IP アドレスプール外である必要があります。

option ステートメントがセミコロンで終了しない場合、DHCP デーモンは起動に失敗し、以下のようなエラーが /var/log/messages に記録されます。

```
/etc/dhcpd.conf line 20: semicolon expected.
dhcpd: }
dhcpd: ^
dhcpd: /etc/dhcpd.conf line 38: unexpected end of file
dhcpd:
dhcpd: ^
dhcpd: Configuration file errors encountered -- exiting
```

複数のネットワークインターフェイスを持つシステムの設定

以下の host 宣言は、複数のネットワークインターフェイスを持つ単一のシステムを設定し、各インターフェイスが同じ IP アドレスを受け取るようにします。両方のネットワークインターフェイスが同じネットワークに同時に接続されている場合には、この設定は機能しません。

```
host interface0 {
 hardware ethernet 00:1a:6b:6a:2e:0b;
 fixed-address 10.0.0.18;
}

host interface1 {
 hardware ethernet 00:1A:6B:6A:27:3A;
 fixed-address 10.0.0.18;
}
```

この例では、interface0 が最初のネットワークインターフェイスで、interface1 は 2 番目のインターフェイスです。異なる ハードウェアイーサネット オプションは、各インターフェイスを識別します。

このようなシステムが別のネットワークに接続されている場合は、ホスト 宣言をさらに追加します。

- *ホストが接続し ているネットワークに有効な Fix-address を割り当てます。*
- *host 宣言の名前を一意にします。*

host 宣言で指定された名前が一意ではない場合、DHCP デーモンは起動に失敗し、以下のようなエラーが /var/log/messages に記録されます。

```
dhcpd: /etc/dhcpd.conf line 31: host interface0: already exists
dhcpd: }
dhcpd: ^
dhcpd: Configuration file errors encountered -- exiting
```

このエラーは、/etc/dhcpd.conf に複数の host interface0 宣言が定義されているために生じました。

23.5. 関連情報

その他の設定オプションについては、以下のリソースを参照してください。

23.5.1. インストールされているドキュメント

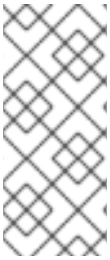
- dhcpd の man ページ：DHCP デーモンがどのように機能するかを説明しています。
- dhcpd.conf の man ページ：DHCP 設定ファイルの設定方法を説明し、例をいくつか紹介します。
- dhcpd.leases の man ページ：DHCP リースファイルの設定方法を説明し、いくつかの例が含まれています。
- dhcp-options の man ページ： dhcpd.conf で DHCP オプションを宣言するための構文の説明と例が含まれています。
- dhcrelay の man ページ：DHCP リレーエージェントおよびその設定オプションについて説明しています。
- /usr/share/doc/dhcp-<バージョン& gt;/: 現行バージョンの DHCP サービスのサンプルファイル、README ファイル、およびリリースノートが含まれます。

---

[8] kudzu は、システムの起動時に実行するハードウェアプロービングツールで、システムから追加または削除されているハードウェアを特定します。

第24章 MYSQL 5.0 から MYSQL 5.5 への移行

MySQL 5.0 から MySQL 5.5 に移行する前に、MySQL データベースを含むすべてのデータのバックアップを作成します。MySQL 5.1 および MySQL 5.5 の詳細は、および http://dev.mysql.com/doc/relnotes/mysql/5.5/en/ から入手 http://dev.mysql.com/doc/relnotes/mysql/5.1/en/ できるリリースノートを参照してください。



注記

Red Hat は、MySQL 5.0 パッケージ(mysql-5.0.* および関連パッケージ)のセキュリティーアドバイザリーを発行しません。セキュリティーアドバイザリーは MySQL 5.5 でのみ提供されます。

24.1. MYSQL 5.0 から MYSQL 5.5 へのアップグレード

MySQL 5.0 から MySQL 5.5 へのアップグレードで唯一サポートされている方法は、MySQL 5.1 を中間ステップとして使用することです。このため、mysql51* Software Collection パッケージが提供されます。MySQL 5.1 パッケージはサポートされておらず、MySQL 5.5 への移行の目的でのみ提供されることに注意してください。実稼働システムでは mysql51* パッケージを使用しないでください。

mysql51 および mysql 55 Software Collections は相互または mysql パッケージと競合しないため、ユーザーは mysql パッケージとともに mysql51 および mysql55 Software Collections をインストールできます。すべてのバージョンの MySQL を同時に実行することもできます。ただし、特定のリソースが競合しないようにするには、my.cnf 設定ファイルでポートとソケット番号を変更する必要があります。

MySQL 5.0 から MySQL 5.5 にアップグレードする方法は 2 つあります。

- mysqldump ユーティリティーおよび mysql ユーティリティーを使用すると、ダンプおよび復元のアップグレードにより、1 つのデータベースからすべてのデータベースの完全に新しいダンプが生成されます。次に、MySQL コマンドラインインターフェイスは、他のデータベース内で入力としてダンプファイルで実行されます（実際には、mysqlimport ユーティリティーまたは LOAD DATA INFILE SQL コマンドを使用します）。ダンプと復元の両方で、適切なデーモンを実行する必要があります。mysqldump コマンドの実行時に --all-databases オプションを使用して、生成されるダンプにすべてのデータベースを追加します。必要に応じて、--routines、--triggers、および --events (MySQL 5.1 以降にのみ有効)オプションを使用できます。

例24.1「ダンプおよびリストアのアップグレード」は、dump メソッドおよび restore メソッドを使用してアップグレードするために使用される特定のコマンドを示しています。

- インプレースアップグレード：データファイルをあるデータベースディレクトリーから別のデータベースディレクトリーにコピーし、両方の MySQL デーモンが停止します。コピーされたファイルに適切な権限と SELinux コンテキストを設定する必要があります。インプレースアップグレードは通常、大規模なデータベースでは高速で簡単ですが、リスクと既知の問題があります。詳細は、本章の最初にリンクされている MySQL 5.1 および MySQL 5.5 のリリースノートを参照してください。

例24.2「インプレースアップグレード」 は、インプレースアップグレードの実行に使用される特定のコマンドを示しています。

アップグレード後、dump および restore を使用するか、インプレースアップグレードを実行して MySQL サーバーを起動し、mysql_upgrade コマンドを実行します。内部テーブルをチェックし、修復するには、mysql_upgrade を実行する必要があります。Software Collection （特に mysql_upgrade スクリプト）から MySQL サーバーと対話するすべてのスクリプトは、scl enable 環境内で実行する必要があります。

注記

mysql_upgrade コマンドの実行中に、以下のエラーが発生する可能性があります。

```
You can't use locks with log tables.
```

これは既知の問題で(http://bugs.mysql.com/bug.php?id=30487で報告)、アップグレードプロセスには影響を与えません。

データのアップグレード時に以外は、新しい環境を反映するように my.cnf 設定ファイルの適切な設定を変更することを検討してください。

例24.1 ダンプおよびリストアのアップグレード

MySQL 5.0 から MySQL 5.5 Software Collection へのアップグレードのダンプおよび復元の例：

```
~]# service mysqld start
Starting mysqld:                                           [  OK  ]
~]# mysqldump --all-databases --routines > dump.sql
~]# service mysqld stop
Stopping mysqld:                                           [  OK  ]
~]# service mysql51-mysqld start
Starting mysql51-mysqld:                                   [  OK  ]
~]# scl enable mysql51 'mysql' < dump.sql
```

```
~]# scl enable mysql51 'mysql_upgrade'
Looking for 'mysql' as: mysql
Looking for 'mysqlcheck' as: mysqlcheck
Running 'mysqlcheck with default connection arguments
Running 'mysqlcheck with default connection arguments
a.t1                                OK
mysql.columns_priv                  OK
⋮
mysql.user                          OK
Running 'mysql_fix_privilege_tables'...
OK
~]# scl enable mysql51 'mysqldump --all-databases --routines --events' > dump2.sql
~]# service mysql51-mysqld stop
Stopping mysqld:                                [  OK  ]
~]# service mysql55-mysqld start
Starting mysql55-mysqld:                        [  OK  ]
~]# scl enable mysql55 'mysql' < dump2.sql
~]# scl enable mysql55 'mysql_upgrade'
Looking for 'mysql' as: mysql
Looking for 'mysqlcheck' as: mysqlcheck
Running 'mysqlcheck with default connection arguments
Running 'mysqlcheck with default connection arguments
a.t1                                OK
mysql.columns_priv                  OK
⋮
mysql.user                          OK
Running 'mysql_fix_privilege_tables'...
OK
```

例24.2 インプレースアップグレード

MySQL 5.0 から MySQL 5.5 Software Collection へのインプレースアップグレードの例 :

```
~]# service mysqld stop
Stopping mysqld:                                [  OK  ]
~]# service mysql51-mysqld stop
Stopping mysql51-mysqld:                        [  OK  ]
~]# rm -rf /opt/rh/mysql51/root/var/lib/mysql/
~]# cp -r /var/lib/mysql/ /opt/rh/mysql51/root/var/lib/mysql/
~]# chown -R mysql:mysql /opt/rh/mysql51/root/var/lib/mysql/
~]# restorecon -R /opt/rh/mysql51/root/var/lib/mysql/
~]# service mysql51-mysqld start
Starting mysql51-mysqld:                        [  OK  ]
~]# scl enable mysql51 'mysql_upgrade'
Looking for 'mysql' as: mysql
Looking for 'mysqlcheck' as: mysqlcheck
Running 'mysqlcheck with default connection arguments
Running 'mysqlcheck with default connection arguments
a.t1                                OK
mysql.columns_priv                  OK
⋮
mysql.user                          OK
```

```
Running 'mysql_fix_privilege_tables'...
OK
~]# service mysql51-mysqld stop
Stopping mysql51-mysqld:                                   [  OK  ]
~]# service mysql55-mysqld stop
Stopping mysql55-mysqld:                                   [  OK  ]
~]# rm -rf /opt/rh/mysql55/root/var/lib/mysql/
~]# cp -r /opt/rh/mysql51/root/var/lib/mysql/ /opt/rh/mysql55/root/var/lib/mysql/
~]# chown -R mysql:mysql /opt/rh/mysql55/root/var/lib/mysql/
~]# restorecon -R /opt/rh/mysql55/root/var/lib/mysql/
~]# service mysql55-mysqld start
Starting mysql55-mysqld:                                   [  OK  ]
~]# scl enable mysql55 'mysql_upgrade'
Looking for 'mysql' as: mysql
Looking for 'mysqlcheck' as: mysqlcheck
Running 'mysqlcheck with default connection arguments
Running 'mysqlcheck with default connection arguments
a.t1                                               OK
mysql.columns_priv                                 OK
 ⋮
mysql.user                                         OK
Running 'mysql_fix_privilege_tables'...
OK
```

第25章 APACHE HTTP サーバー

Apache HTTP Server は、Apache Software Foundation (http://www.apache.org/)が開発した、堅牢で商用レベルのオープンソース Web サーバーです。Red Hat Enterprise Linux には、Apache HTTP Server 2.2 と、その機能を強化するために設計されたサーバーモジュールが多数含まれています。

Apache HTTP Server でインストールされるデフォルトの設定ファイルは、ほとんどの状況で変更せずに動作します。本章では、設定ファイル内の多くのディレクティブ(/etc/httpd/conf/httpd.conf)を、カスタム設定を必要とするか、以前の Apache HTTP Server 1.3 形式から設定ファイルを変換する必要があるものを支援します。

WARNING

グラフィカル HTTP Configuration Tool (system-config-httpd)を使用する場合は、Apache HTTP Server の設定ファイルを手動で編集し ない でください。これは、HTTP Configuration Tool が使用されるたびにこのファイルを再生成するためです。

25.1. APACHE HTTP SERVER 2.2

Apache HTTP Server 2.2 とバージョン 2.0 には、Red Hat Enterprise Linux 4 に同梱されていたバージョン 2.0 には重要な違いがあります。本セクションでは、Apache HTTP Server 2.2 の機能の一部を確認し、重要な変更を説明します。バージョン 1.3 からアップグレードする場合は、バージョン 1.3 からバージョン 2.0 に移行する手順も読み取る必要があります。バージョン 1.3 設定ファイルを 2.0 形式に移行する手順は、「Apache HTTP Server 1.3 設定ファイルの 2.0 への移行」 を参照してください。

25.1.1. Apache HTTP Server 2.2 の機能

Apache HTTP Server 2.2 では、バージョン 2.0 に対する以下の改善点があります。

- キャッシュモジュール(mod_cache、mod_disk_cache、mod_mem_cache)が改善されました。
- 以前のバージョンで提供された認証モジュールを置き換える、認証および承認サポートの新しい構造。

- プロキシー負荷分散のサポート(mod_proxy_balancer)

- 32 ビットプラットフォームでの大規模なファイル（つまり 2GB 以上）を処理するサポート

デフォルトの httpd 設定に以下の変更が加えられました。

- mod_cern_meta モジュールおよび mod_asis モジュールはデフォルトで読み込まれなくなりました。

- mod_ext_filter モジュールはデフォルトで読み込まれるようになりました。

Red Hat Enterprise Linux の以前のリリースからアップグレードする場合は、httpd 2.2 の httpd 設定を更新する必要があります。詳細は、http://httpd.apache.org/docs/2.2/upgrading.html を参照してください。

25.2. APACHE HTTP サーバー設定ファイルの移行

25.2.1. Apache HTTP Server 2.0 設定ファイルの移行

本セクションでは、バージョン 2.0 から 2.2 への移行について説明します。バージョン 1.3 から移行する場合は、「Apache HTTP Server 1.3 設定ファイルの 2.0 への移行」 を参照してください。

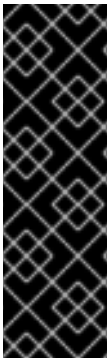
- バージョン 2.0 の設定ファイルと起動スクリプトには、特に変更された可能性のあるモジュール名でマイナーな調整が必要になります。バージョン 2.0 で動作していたサードパーティーモジュールはバージョン 2.2 でも機能しますが、読み込む前に再コンパイルする必要があります。記述する必要がある主なモジュールは、認証および承認モジュールです。名前が変更された各モジュールに対して、LoadModule 行を更新する必要があります。

- mod_userdir モジュールは、ディレクトリー名を示す UserDir ディレクティブを指定すると、要求にのみ動作します。バージョン 2.0 で使用される手順を維持する場合は、設定ファイルに UserDir public_html ディレクティブを追加します。

- SSL を有効にするには、必要な mod_ssl ディレクティブを追加して httpd.conf ファイルを編集します。バージョン 2.2 では apachectl start ssl が利用できないため、apachectl start を使用します。httpd の SSL 設定の例は conf/extra/httpd-ssl.conf にあります。

- 設定をテストするには、設定エラーを検出する service httpd configtest を使用することが推奨されます。

バージョン 2.0 から 2.2 へのアップグレードに関する詳細は、http://httpd.apache.org/docs/2.2/upgrading.html を参照してください。



重要

『POODLE: SSLv3 脆弱性(CVE-2014-3566)で説明されている脆弱性』 により、Red Hat は SSL を無効にし、TLSv1. 1 または TLSv1.2 のみを使用することを推奨します。後方互換性は、TLSv1.0 を使用して実現できます。Red Hat がサポートする多くの製品は SSLv2 プロトコルまたは SSLv3 プロトコルを使用するか、デフォルトでそれらのプロトコルを有効にできます。ただし、SSLv2 または SSLv3 を使用することが強く推奨されます。

25.2.2. Apache HTTP Server 1.3 設定ファイルの 2.0 への移行

本セクションでは、Apache HTTP Server 2.0 で使用される Apache HTTP Server 1.3 設定ファイルの移行について詳しく説明します。

Red Hat Enterprise Linux 2.1 から Red Hat Enterprise Linux 5 にアップグレードする場合、Apache HTTP Server 2.0 パッケージの新しいストック設定ファイルは /etc/httpd/conf/httpd.conf.rpmnew としてインストールされ、元のバージョン 1.3 httpd.conf は変更されない点に注意してください。新しい設定ファイルを使用するか、古い設定をベースとして移行するか、既存のファイルをベースとして使用したり、適切に変更したりしても、ファイルの一部が他の部分よりも変更され、通常は混合アプローチが最適です。バージョン 1.3 と 2.0 の両方のストック設定ファイルは、3 つのセクションに分かれています。

/etc/httpd/conf/httpd.conf ファイルが、新しくインストールされたデフォルトのバージョンであり、元の設定ファイルのコピーを保存すると、以下の例のように diff コマンドを呼び出すのが最も簡単な方法です(root でログイン)。

```
diff -u httpd.conf.orig httpd.conf | less
```

このコマンドは、変更内容を強調表示します。元のファイルのコピーが利用できない場合は、以下の例のように rpm2cpio および cpio コマンドを使用して RPM パッケージから抽出します。

```
rpm2cpio apache-<version-number>.i386.rpm | cpio -i --make
```

上記のコマンドで、< version-number> を apache パッケージのバージョン番号に置き換えてくだ

さい。

最後に、Apache HTTP Server に設定エラーをチェックするテストモードがあることを確認すると便利です。アクセスを使用するには、以下のコマンドを入力します。

```
apachectl configtest
```

25.2.2.1. グローバル環境設定

設定ファイルのグローバル環境セクションには、処理する同時要求の数や各種ファイルの場所など、Apache HTTP Server の全体的な操作に影響するディレクティブが含まれています。本セクションでは多くの変更が必要で、古い設定をその設定ファイルに移行する際に Apache HTTP Server 2.0 設定ファイルをベースとする必要があります。

25.2.2.1.1. インターフェイスおよびポートバインディング

BindAddress ディレクティブおよび Port ディレクティブは存在しなくなり、それらの機能はより柔軟な Listen ディレクティブで提供されるようになりました。

1.3 バージョン設定ファイルで Port 80 が設定されている場合は、2.0 設定ファイルで Listen 80 に変更します。Port が 80 以外 の値に設定されている場合は、ポート番号を ServerName ディレクティブの内容に追加します。

Apache HTTP Server 1.3 ディレクティブの例を以下に示します。

```
Port 123
ServerName www.example.com
```

この設定を Apache HTTP Server 2.0 に移行するには、以下の構造を使用します。

```
Listen 123
ServerName www.example.com:123
```

このトピックの詳細は、Apache Software Foundation の Web サイトを参照してください。

- http://httpd.apache.org/docs-2.0/mod/mpm_common.html#listen

- *http://httpd.apache.org/docs-2.0/mod/core.html#servername*

25.2.2.1.2. Server-Pool Size Regulation

Apache HTTP Server が要求を受け入れると、子プロセスまたはスレッドをディスパッチしてそれらを処理します。子プロセスまたはスレッドのこのグループは、サーバープール と呼ばれます。Apache HTTP Server 2.0 では、これらのサーバープールを作成および維持する責任は、Multi-Processing Modules (MPMs)と呼ばれるモジュールのグループに抽象化されています。他のモジュールとは異なり、Apache HTTP Server は MPM グループの 1 つのモジュールのみをロードできます。2.0 に同梱される MPM モジュールは、prefork、worker、および perchild の 3 つです。現在、prefork および worker MPM のみが利用できますが、perchild MPM は後日で利用できます。

元の Apache HTTP Server 1.3 の動作は、prefork MPM に移動しました。prefork MPM は Apache HTTP Server 1.3 と同じディレクティブを受け入れるため、以下のディレクティブを直接移行できます。

- *StartServers*
- *MinSpareServers*
- *MaxSpareServers*
- *MaxClients*
- *MaxRequestsPerChild*

worker MPM は、マルチプロセスのマルチスレッドサーバーを実装し、スケーラビリティーを向上させます。この MPM を使用する場合、要求はスレッドによって処理され、システムリソースを監視し、多数の要求を効率的に処理できるようにします。worker MPM で使用できるディレクティブの一部は、prefork MPM で受け入れられるディレクティブと同じですが、これらのディレクティブの値は Apache HTTP Server 1.3 インストールから直接転送しないでください。代わりにデフォルト値をガイドとして使用し、最適な値を判断してみてください。

重要な影響

worker MPM を使用するには、/etc/sysconfig/httpd ファイルを作成し、以下のディレクティブを追加します。

```
HTTPD=/usr/sbin/httpd.worker
```

MPM のトピックの詳細は、Apache Software Foundation の Web サイトを参照してください。

- http://httpd.apache.org/docs-2.0/mpm.html

25.2.2.1.3. Dynamic Shared Object (DSO)のサポート

ここで必要な変更は多数あります。Apache HTTP Server 1.3 の設定をバージョン 2.0 に合わせて修正しようとする場合は（バージョン 2.0 設定への変更ではなく）、ストック Apache HTTP Server 2.0 設定ファイルからこのセクションをコピーすることが強く推奨されます。

ストック Apache HTTP Server 2.0 設定から セクションをコピーしたくない場合は、以下に注意してください。

- AddModule ディレクティブおよび ClearModuleList ディレクティブは存在しなくなりました。これらのディレクティブは、を使用してモジュールが正しい順序で有効にされるようにします。Apache HTTP Server 2.0 API を使用すると、モジュールは順序を指定でき、これら 2 つのディレクティブが不要になります。
- LoadModule 行の順序は、ほとんどの場合、関連性がなくなりました。
- 多くのモジュールは、追加、削除、名前変更、分割、または他に組み込まれています。
- 独自の RPM にパッケージ化されたモジュールの LoadModule 行 (mod_ssl、php、mod_perl など)は、/etc/httpd/conf.d/ ディレクトリー内の関連ファイルにあるため、不要になりました。
- さまざまな HAVE_XXX 定義が定義されなくなりました。

重要な影響

元のファイルを変更する場合は、httpd.conf に以下のディレクティブが含まれていることが重要です。

```
Include conf.d/*.conf
```

このディレクティブを省略すると、独自の RPM にパッケージ化されたすべてのモジュール(mod_perl、php、mod_sslなど)が失敗します。

25.2.2.1.4. その他のグローバル環境の変更

以下のディレクティブは、Apache HTTP Server 2.0 の設定から削除されました。

- *serverType: Apache HTTP Server は* ServerType *スタンドアロン としてのみ実行でき、このディレクティブは無関係です。*

- AccessConfig *および* ResourceConfig*: これらのディレクティブは Include ディレクティブの機能をミラーリングするため削除されました。AccessConfig ディレクティブおよび ResourceConfig ディレクティブが設定されている場合は、Include ディレクティブに置き換えます。*

  ファイルが古いディレクティブによって暗示される順序で読み取られるようにするには、Include ディレクティブを httpd.conf の末尾に配置し、AccessConfig に対応するものの前に ResourceConfig に対応するものを追加する必要があります。デフォルト値を使用する場合は、conf/srm.conf および conf/access.conf ファイルとして明示的に組み込みます。

25.2.2.2. メインサーバー設定

設定ファイルの主なサーバー設定セクションは、メインサーバーを設定します。これは、< VirtualHost > コンテナー内で定義される仮想ホストで処理されない要求に応答します。ここでの値により、定義された < VirtualHost> コンテナーのデフォルトも提供されます。

このセクションで使用されるディレクティブは、Apache HTTP Server 1.3 とバージョン 2.0 の間ではほとんど変更されていません。メインサーバー設定が大幅にカスタマイズされている場合は、Apache HTTP Server 2.0 に合わせて既存の設定ファイルを変更する方が簡単な場合があります。若干カスタマイズされたメインサーバーセクションのみを持つユーザーは、変更をデフォルトの 2.0 設定に移行する必要があります。

25.2.2.2.1. UserDir マッピング

UserDir ディレクティブは、http://example.com/~bob/ などの URL を有効にし、/home/bob/public_html/ などのユーザー bob 内のサブディレクトリーにマッピングします。この機能の副次的な影響により、潜在的な攻撃者は特定のユーザー名がシステムに存在するかどうかを判断できます。このため、Apache HTTP Server 2.0 のデフォルト設定はこのディレクティブを無効にします。

UserDir マッピングを有効にするには、httpd.conf の ディレクティブを以下のように変更します。

```
UserDir disable
```

次のように変更します。

```
UserDir public_html
```

このトピックの詳細は、Apache Software Foundation の Web サイトを参照してください。

- http://httpd.apache.org/docs-2.0/mod/mod_userdir.html#userdir

25.2.2.2.2. ロギング

以下のロギングディレクティブが削除されました。

- AgentLog
- RefererLog
- RefererIgnore

ただし、agent および referrer ログは、CustomLog ディレクティブおよび LogFormat ディレクティブで引き続き利用できます。

このトピックの詳細は、Apache Software Foundation の Web サイトを参照してください。

- http://httpd.apache.org/docs-2.0/mod/mod_log_config.html#customlog
- http://httpd.apache.org/docs-2.0/mod/mod_log_config.html#logformat

25.2.2.2.3. ディレクトリーのインデックス作成

非推奨の FancyIndexing ディレクティブが削除されました。同じ機能は、IndexOptions ディレクティブ内の FancyIndexing オプション で利用できます。

IndexOptions ディレクティブの VersionSort オプションにより、バージョン番号を含むファイルをより自然な方法でソートします。たとえば、httpd-2.0.6.tar は、ディレクトリーインデックスページに httpd-2.0.36.tar の前に表示されます。

ReadmeName ディレクティブおよび HeaderName ディレクティブのデフォルトが、README および HEADER から README.html および HEADER.html に変更になりました。

このトピックの詳細は、Apache Software Foundation の Web サイトを参照してください。

- http://httpd.apache.org/docs-2.0/mod/mod_autoindex.html#indexoptions
- http://httpd.apache.org/docs-2.0/mod/mod_autoindex.html#readmename
- http://httpd.apache.org/docs-2.0/mod/mod_autoindex.html#headername

25.2.2.2.4. コンテンツネゴシエーション

CacheNegotiatedDocs ディレクティブは、の引数(on または off)を取るようになりました。CacheNegotiatedDocs の既存インスタンスは、の CacheNegotiatedDocs に置き換える必要があり ます。

このトピックの詳細は、Apache Software Foundation の Web サイトを参照してください。

- http://httpd.apache.org/docs-2.0/mod/mod_negotiation.html#cachenegotiateddocs

25.2.2.2.5. エラードキュメント

ErrorDocument ディレクティブでハードコードされたメッセージを使用するには、Apache HTTP Server 1.3 で必要な二重引用符 の前に二重引用符 を付けるのではなく、メッセージを二重引用符 " のペアで囲む必要があります。

Apache HTTP Server 1.3 ディレクティブの例を以下に示します。

```
ErrorDocument 404 "The document was not found
```

ErrorDocument 設定を Apache HTTP Server 2.0 に移行するには、以下の構造を使用します。

```
ErrorDocument 404 "The document was not found"
```

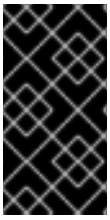
前の ErrorDocument ディレクティブ例の最後の二重引用符に注意してください。

このトピックの詳細は、Apache Software Foundation の Web サイトを参照してください。

- http://httpd.apache.org/docs-2.0/mod/core.html#errordocument

25.2.2.3. 仮想ホストの設定

すべての < VirtualHost> コンテナー のコンテンツは、「メインサーバー設定」 で説明されているメインサーバーセクションと同じ方法で移行する必要があります。



重要な影響

SSL/TLS 仮想ホスト設定は、メインのサーバー設定ファイルと /etc/httpd/conf.d/ssl.conf に移動されていることに注意してください。

- http://httpd.apache.org/docs-2.0/vhosts/

25.2.2.4. モジュールおよび Apache HTTP Server 2.0

Apache HTTP Server 2.0 では、モジュールシステムが変更され、モジュールをチェーン化したり、新しいや関心のある方法で組み合わせたりできるようになりました。Common Gateway Interface (CGI)スクリプトは、たとえば、mod_include で処理できるサーバー解析された HTML ドキュメントを生成できます。これにより、モジュールを組み合わせて特定の目的を達成する方法に関して、非常に多くの可能性が開かれます。

これが機能する仕組みは、各リクエストに 1 つの ハンドラー モジュールと、その後に 0 個以上の フィルター モジュールが提供されることです。

Apache HTTP Server 1.3 では、たとえば Perl スクリプトは Perl モジュール(mod_perl)によって完全に処理されます。Apache HTTP Server 2.0 では、リクエストは最初にコアモジュールによって 処理 されます。これは静的ファイルを提供し、mod_perl によってフィルターリング されます。

この使用方法、および Apache HTTP Server 2.0 のその他の新機能はすべてこのドキュメントの範囲外になります。ただし、PATH_INFO ディレクティブがフィルターとして実装されるモジュールで処理されるドキュメントに PATH_INFO ディレクティブが使用されている場合、それぞれが true ファイル名の後に続くパス情報が含まれているため、変更はこの変更の影響を受けます。最初に要求を処理するコアモジュールは、デフォルトでは PATH_INFO を理解せず、このような情報を含むリクエストに対して 404 Not Found エラーを返します。別の方法として、AcceptPathInfo ディレクティブを使用してコアモジュールを照合し、PATH_INFO でリクエストを受け入れることもできます。

以下は、このディレクティブの例です。

```
AcceptPathInfo on
```

このトピックの詳細は、Apache Software Foundation の Web サイトを参照してください。

- http://httpd.apache.org/docs-2.0/mod/core.html#acceptpathinfo
- http://httpd.apache.org/docs-2.0/handler.html
- http://httpd.apache.org/docs-2.0/filter.html

25.2.2.4.1. suexec モジュール

Apache HTTP Server 2.0 では、mod_suexec モジュールは仮想ホストの設定に使用される User ディレクティブおよび Group ディレクティブではなく、SuexecUserGroup ディレクティブを使用します。User ディレクティブおよび Group ディレクティブは一般的に使用できますが、仮想ホストの設定には非推奨となっています。

Apache HTTP Server 1.3 ディレクティブの例を以下に示します。

```
<VirtualHost vhost.example.com:80>
  User someone
  Group somegroup
</VirtualHost>
```

この設定を Apache HTTP Server 2.0 に移行するには、以下の構造を使用します。

```
<VirtualHost vhost.example.com:80>
  SuexecUserGroup someone somegroup
</VirtualHost>
```

25.2.2.4.2. mod_ssl モジュール

mod_ssl の設定は、httpd.conf ファイルから /etc/httpd/conf.d/ssl.conf ファイルに移動しました。このファイルを読み込み、mod_ssl を機能させるには、「Dynamic Shared Object (DSO)のサポート」 の説明に従って Include conf.d/*.conf というステートメントが httpd.conf ファイルに含まれている必要があります。

SSL 仮想ホストの ServerName ディレクティブは、ポート番号を明示的に指定する必要があります。

Apache HTTP Server 1.3 ディレクティブの例を以下に示します。

```
<VirtualHost _default_:443>
  # General setup for the virtual host
  ServerName ssl.example.name
  ...
</VirtualHost>
```

この設定を Apache HTTP Server 2.0 に移行するには、以下の構造を使用します。

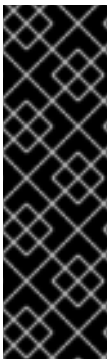
```
<VirtualHost _default_:443>
```

```
# General setup for the virtual host
ServerName ssl.host.name:443
...
</VirtualHost>
```

SSLLog ディレクティブと SSLLogLevel ディレクティブの両方が削除されていることにも注意してください。mod_ssl モジュールは ErrorLog ディレクティブおよび LogLevel ディレクティブに従うようになりました。これらのディレクティブの詳細は、ErrorLog および LogLevel を参照してください。

このトピックの詳細は、Apache Software Foundation の Web サイトを参照してください。

- http://httpd.apache.org/docs-2.0/mod/mod_ssl.html
- http://httpd.apache.org/docs-2.0/vhosts/



重要

『POODLE: SSLv3 脆弱性(CVE-2014-3566)で説明されている脆弱性』 により、Red Hat は SSL を無効にし、TLSv1. 1 または TLSv1.2 のみを使用することを推奨します。後方互換性は、TLSv1.0 を使用して実現できます。Red Hat がサポートする多くの製品は SSLv2 プロトコルまたは SSLv3 プロトコルを使用するか、デフォルトでそれらのプロトコルを有効にできます。ただし、SSLv2 または SSLv3 を使用することが強く推奨されます。

25.2.2.4.3. mod_proxy モジュール

プロキシーアクセス制御ステートメントは、< Directory proxy: > ではなく < Proxy > ブロック内に配置されるようになりました。

古い mod_proxy のキャッシング機能は、以下の 3 つのモジュールに分割されました。

- mod_cache
- mod_disk_cache

- *mod_mem_cache*

これらは通常、古いバージョンの mod_proxy モジュールと同様のディレクティブを使用しますが、キャッシュ設定を移行する前に各ディレクティブを検証することが推奨されます。

このトピックの詳細は、Apache Software Foundation の Web サイトを参照してください。

- http://httpd.apache.org/docs-2.0/mod/mod_proxy.html

25.2.2.4.4. mod_include モジュール

mod_include モジュールはフィルターとして実装されるため、異なる方法で有効化されるようになりました。フィルターの詳細は、「モジュールおよび Apache HTTP Server 2.0」 を参照してください。

Apache HTTP Server 1.3 ディレクティブの例を以下に示します。

```
AddType text/html .shtml
AddHandler server-parsed .shtml
```

この設定を Apache HTTP Server 2.0 に移行するには、以下の構造を使用します。

```
AddType text/html .shtml
AddOutputFilter INCLUDES .shtml
```

オプション +Includes ディレクティブは、引き続き < Directory> コンテナーまた は .htaccess ファイルに必要です。

このトピックの詳細は、Apache Software Foundation の Web サイトを参照してください。

- http://httpd.apache.org/docs-2.0/mod/mod_include.html

25.2.2.4.5. mod_auth_dbm モジュールおよび mod_auth_db モジュール

Apache HTTP Server 1.3 は、それぞれ Berkeley Databases と DBM データベースを使用する mod_auth_db と mod_auth_dbm の 2 つの認証モジュールをサポートしていました。これらのモジュールは、Apache HTTP Server 2.0 の mod_auth_dbm という名前の単一のモジュールに統合され、複数の異なるデータベース形式にアクセスできます。mod_auth_db から移行するには、AuthDBUserFile と AuthDBGroupFile を、AuthDBMUserFile および AuthDBMGroupFile の mod_auth_dbm に置き換えて調整する必要があります。また、使用中のデータベースファイルのタイプを指定するには、ディレクティブ AuthDBMType DB を追加する必要があります。

以下の例は、Apache HTTP Server 1.3 の mod_auth_db 設定のサンプルを示しています。

```
<Location /private/>
  AuthType Basic
  AuthName "My Private Files"
  AuthDBUserFile /var/www/authdb
  require valid-user
</Location>
```

この設定を Apache HTTP Server のバージョン 2.0 に移行するには、以下の構造を使用します。

```
<Location /private/>
  AuthType Basic
  AuthName "My Private Files"
  AuthDBMUserFile /var/www/authdb
  AuthDBMType DB
  require valid-user
</Location>
```

AuthDBMUserFile ディレクティブは、.htaccess ファイルでも使用できることに注意してください。

Apache HTTP Server 2.0 では、ユーザー名とパスワードのデータベースを操作するために使用される dbmmanage Perl スクリプトは htdbm に置き換えられました。htdbm プログラムは同等の機能を提供し、mod_auth_dbm のようにさまざまなデータベース形式を操作できます。コマンドラインで -T オプションを使用して、使用する形式を指定できます。

表25.1「dbmmanage から htdbmへの移行」 では、dbmmanage を使用して DBM 形式のデータベースから htdbm 形式に移行する方法を示します。

表25.1 dbmmanage から htdbmへの移行

アクション	dbmmanage コマンド(1.3)	同等の htdbm コマンド(2.0)
データベースにユーザーを追加する（指定のパスワードを使用）	dbmmanage authdb add username password	htdbm -b -TDB authdb username password
ユーザーをデータベースに追加する（パスワードのプロンプト）	dbmmanage authdb adduser username	htdbm -TDB authdb username
データベースからユーザーを削除します。	dbmmanage authdb delete username	htdbm -x -TDB authdb username
データベースのユーザーを一覧表示します。	dbmmanage authdb view	htdbm -l -TDB authdb
パスワードの確認	dbmmanage authdb check username	htdbm -v -TDB authdb username

-m オプションおよび -s オプションは dbmmanage と htdbm の両方と連携し、ハッシュパスワードに MD5 アルゴリズムまたは SHA1 アルゴリズムをそれぞれ使用できます。

htdbm で新しいデータベースを作成する場合は、-c オプションを使用する必要があります。

このトピックの詳細は、Apache Software Foundation の Web サイトを参照してください。

- http://httpd.apache.org/docs-2.0/mod/mod_auth_dbm.html

25.2.2.4.6. mod_perl モジュール

mod_perl の設定は、httpd.conf から /etc/httpd/conf.d/perl.conf ファイルに移動しました。このファイルを読み込むには、mod_perl が機能するには、「Dynamic Shared Object (DSO)のサポート」の説明に従って Include conf.d/*.conf というステートメントを httpd.conf に含める必要があります。

httpd.conf の Apache:: は ModPerl:: に置き換える必要があります。また、ハンドラーの登録方法も変更になりました。

以下は、Apache HTTP Server 1.3 mod_perl の設定例です。

```
<Directory /var/www/perl>
  SetHandler perl-script
  PerlHandler Apache::Registry
  Options +ExecCGI
</Directory>
```

これは、Apache HTTP Server 2.0 の mod_perl と同じです。

```
<Directory /var/www/perl>
  SetHandler perl-script
  PerlResponseHandler ModPerl::Registry
  Options +ExecCGI
</Directory>
```

mod_perl 1.x のほとんどのモジュールは、mod_perl 2.x で変更せずに動作するはずです。XS モジュールには再コンパイルが必要で、Makefile のマイナーな変更が必要になる場合があります。

25.2.2.4.7. mod_python モジュール

mod_python の設定は、httpd.conf から /etc/httpd/conf.d/python.conf ファイルに移動しました。このファイルを読み込むには、mod_python が機能するには、「Dynamic Shared Object (DSO) のサポート」 の説明に従って Include conf.d/*.conf というステートメントを httpd.conf に配置する必要があります。

25.2.2.4.8. PHP

PHP の設定は、httpd.conf からファイル /etc/httpd/conf.d/php.conf に移動しました。このファイルを読み込むには、「Dynamic Shared Object (DSO)のサポート」 の説明に従って Include conf.d/*.conf というステートメントを httpd.conf に配置する必要があります。

注記

Apache HTTP Server 1.3 で使用される PHP 設定ディレクティブは、Red Hat Enterprise Linux 5 で Apache HTTP Server 2.0 に移行する際に完全に互換性を持つようになりました。

PHP バージョン 4.2.0 以降では、グローバルスコープで利用可能な事前定義された変数のデフォルトセットが変更されました。個々の入力変数およびサーバー変数は、デフォルトではグローバルスコー

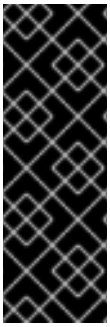
プに直接配置されなくなりました。この変更により、スクリプトが破損する可能性があります。/etc/php.ini ファイルで register_globals を On に設定して、以前の動作に戻します。

このトピックの詳細は、グローバルスコープの変更に関する詳細は、以下の URL を参照してください。

- http://www.php.net/release_4_1_0.php

25.2.2.4.9. mod_authz_ldap モジュール

Red Hat Enterprise Linux には、Apache HTTP Server 用の mod_authz_ldap モジュールが同梱されています。このモジュールは、サブジェクトの識別名の短い形式とクライアント SSL 証明書の発行者を使用して、LDAP ディレクトリー内のユーザーの識別名を決定します。また、ユーザーの LDAP ディレクトリーエントリーの属性に基づいてユーザーを作成したり、アセットのユーザーおよびグループ権限に基づいてアセットへのアクセスを判断したり、パスワードの期限が切れたユーザーのアクセスを拒否したりすることもできます。mod_authz_ldap モジュールを使用する場合は、mod_ssl モジュールが必要です。



重要な影響

mod_authz_ldap モジュールは、暗号化されたパスワードハッシュを使用して LDAP ディレクトリーに対してユーザーを認証しません。この機能は、実験的な mod_auth_ldap モジュールによって提供されます。このモジュールのステータスの詳細は、http://httpd.apache.org/docs-2.0/mod/mod_auth_ldap.html でオンラインの mod_auth_ldap モジュールのドキュメントを参照してください。

/etc/httpd/conf.d/authz_ldap.conf ファイルは、mod_authz_ldap モジュールを設定します。

mod_authz_ldap サードパーティーモジュールの設定に関する詳細は、/usr/share/doc/mod_authz_ldap- <version> /index.html（パッケージのバージョン番号に <version> を置換）または http://authzldap.othello.ch/ を参照してください。

25.3. HTTPDの起動と停止

httpd パッケージをインストールしたら、http://httpd.apache.org/docs/2.2/ でオンラインで利用できる Apache HTTP Server のドキュメントを確認してください。

httpd RPM は、/etc/init.d/httpd スクリプトをインストールします。これは、/sbin/service コマンドを使用してアクセスできます。

apachectl control スクリプトを使用して httpd を起動すると、/etc/sysconfig/httpd に環境変数が設定され、httpd が起動します。init スクリプトを使用して環境変数を設定することもできます。

root で apachectl 制御スクリプトを使用してサーバーを起動するには、次のコマンドを実行します。

```
apachectl start
```

/sbin/service httpd start を使用して httpd を起動 することもできます。これにより httpd が起動しますが、環境変数は設定されません。httpd.conf （ポート 80）でデフォルトの Listen ディレクティブを使用している場合は、apache サーバーを起動するために root 権限が必要になります。

サーバーを停止するには、root で以下を入力します。

```
apachectl stop
```

/sbin/service httpd stop を使用して httpd を停止 することもできます。restart オプションは、Apache HTTP Server を停止して起動する簡単な方法です。

以下を入力して、root でサーバーを再起動することができます。

```
apachectl restart
```

または

```
service httpd restart
```

起動時にエラーが発生した場合は、Apache がコンソールまたは ErrorLog にメッセージを表示します。

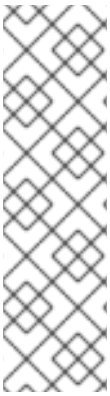
デフォルトでは、httpd サービスは起動時に自動的に起動し ません。起動時に Apache を起動する場合は、rc.N ディレクトリー内の起動ファイルに apachectl への呼び出しを追加する必要があります。典型的な使用されるファイルは rc.local です。これにより Apache が root として起動されるため、この呼び出しを追加する前にセキュリティーと認証を適切に設定することが推奨されます。

/sbin/chkconfig、/usr/sbin/ntsysv、または Services Configuration Tool プログラムなどの initscript ユーティリティーを使用して、システムの起動時に httpd サービスが起動するように設定することもできます。

また、以下を入力して httpd サーバーのステータスを表示することもできます。

```
apachectl status
```

ただし、これを機能させるには、status モジュール mod_status を httpd.conf 設定ファイルで有効にする必要があります。mod_status の詳細は、http://httpd.apache.org/docs/2.2/mod/mod_status.html を参照してください。



注記

Apache HTTP Server をセキュアなサーバーとして実行する場合は、暗号化されたプライベート SSL キーを使用するときにマシンが起動した後にセキュアなサーバーのパスワードが必要になります。

詳細は、http://httpd.apache.org/docs/2.2/sslを参照してください。

25.4. APACHE HTTP サーバーの設定

HTTP Configuration Tool を使用すると、Apache HTTP Server の /etc/httpd/conf/httpd.conf 設定ファイルを設定できます。古い srm.conf または access.conf 設定ファイルは使用されず、空のままにします。グラフィカルインターフェイスを使用して、仮想ホスト、ロギング属性、接続の最大数などのディレクティブを設定できます。HTTD 設定ツールを起動するには、システム > 管理 > サーバー設定 > HTTP を クリックします。

HTTP Configuration Tool で設定できるのは、Red Hat Enterprise Linux で提供されるモジュールのみです。追加のモジュールがインストールされている場合、このツールを使用して設定することはできません。

注意

このツールを使用する場合は、/etc/httpd/conf/httpd.conf 設定ファイルを手動で編集しないでください。HTTP Configuration Tool は、変更を保存してプログラムを終了すると、このファイルを生成します。HTTP Configuration Tool で利用できないモジュールまたは設定オプションを追加する場合は、このツールを使用することはできません。

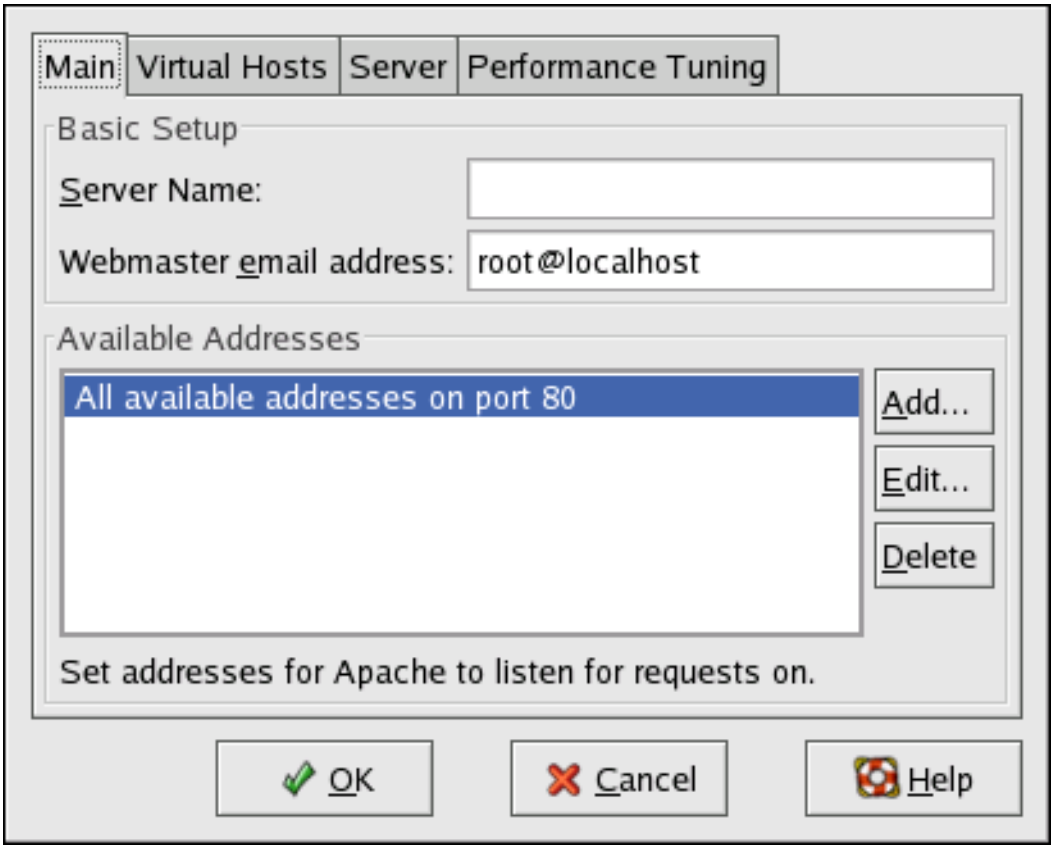
HTTP Configuration Tool を使用して Apache HTTP Server を設定する一般的な手順は以下のとおりです。

1. *Main タブで基本設定を行います。*
2. *Virtual Hosts タブをクリックし、デフォルト設定を設定します。*
3. *Virtual Hosts タブで、Default Virtual Host を設定します。*
4. *複数の URL または仮想ホストを提供するには、仮想ホストを追加します。*
5. *Server タブでサーバー設定を設定します。*
6. *Performance Tuning タブで接続を設定します。*
7. *必要なファイルをすべて DocumentRoot ディレクトリーおよび cgi-bin ディレクトリーにコピーします。*
8. *アプリケーションを終了し、 を選択して設定を保存します。*

25.4.1. 基本設定

Main タブを使用して基本的なサーバー設定を設定します。

図25.1 基本設定



[D]

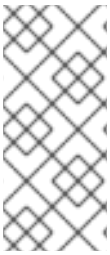
Server Name テキスト領域で使用する権限を持つ完全修飾ドメイン名を入力します。このオプションは、httpd.conf の ServerName ディレクティブに対応します。ServerName ディレクティブは、Web サーバーのホスト名を設定します。リダイレクト URL の作成時に使用されます。サーバー名を定義しない場合、Web サーバーはシステムの IP アドレスから解決を試みます。サーバー名は、サーバーの IP アドレスから解決されるドメイン名である必要はありません。たとえば、サーバーの実際の DNS 名は foo.example.com で、サーバー名を www.example.com に設定します。

Web マスターのメールアドレステキストエリアに、Web サーバーを管理するユーザーのメールアドレス を入力します。このオプションは、httpd.conf の ServerAdmin ディレクティブに対応します。サーバーのエラーページをメールアドレスが含まれるように設定すると、ユーザーがサーバーの管理者に問題を報告できるようにこのメールアドレスが使用されます。デフォルト値は root@localhost です。

Available Addresses 領域を使用して、サーバーが受信要求を受け入れるポートを定義します。このオプションは、httpd.conf の Listen ディレクティブに対応します。デフォルトでは、Red Hat は、セキュアではない Web 通信のポート 80 をリッスンするように Apache HTTP Server を設定します。

Add ボタンをクリックして、要求を受け入れる追加のポートを定義します。図25.2「利用可能なアドレス」 に示されるようにウィンドウが表示されます。Listen to all addresses オプションを選択して定義されたポートですべての IP アドレスをリッスンするか、またはサーバーが接続を受け入れる特定の IP アドレスを Address フィールドで指定します。ポート番号ごとに 1 つの IP アドレスのみを指定します。同じポート番号で複数の IP アドレスを指定するには、各 IP アドレスにエントリーを作成します。可能な場合、DNS ルックアップが失敗しないように、ドメイン名の代わりに IP アドレスを使用します。『DNS および Apache の問題』 については、http://httpd.apache.org/docs/2.2/dns-caveats.html を参照してください。

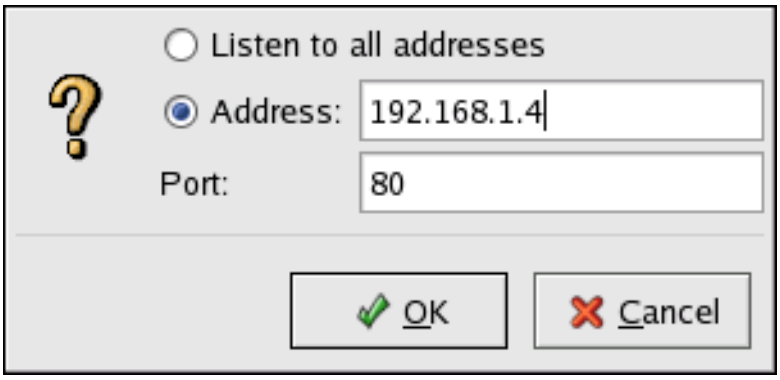
Address フィールドにアスタリスク(*)を入力することは、すべてのアドレス の Listen を選択することと同じです。Available Addresses フレームの Edit ボタンをクリックすると、選択したエントリーに入力されたフィールドを除き、Add ボタンと同じウィンドウが表示されます。エントリーを削除するには、エントリーを選択し、Delete ボタンをクリックします。



ヒント

1024 未満のポートをリッスンするようにサーバーを設定した場合は、root で起動する必要があります。ポート 1024 以降では、通常のユーザーとして httpd を起動できます。

図25.2 利用可能なアドレス

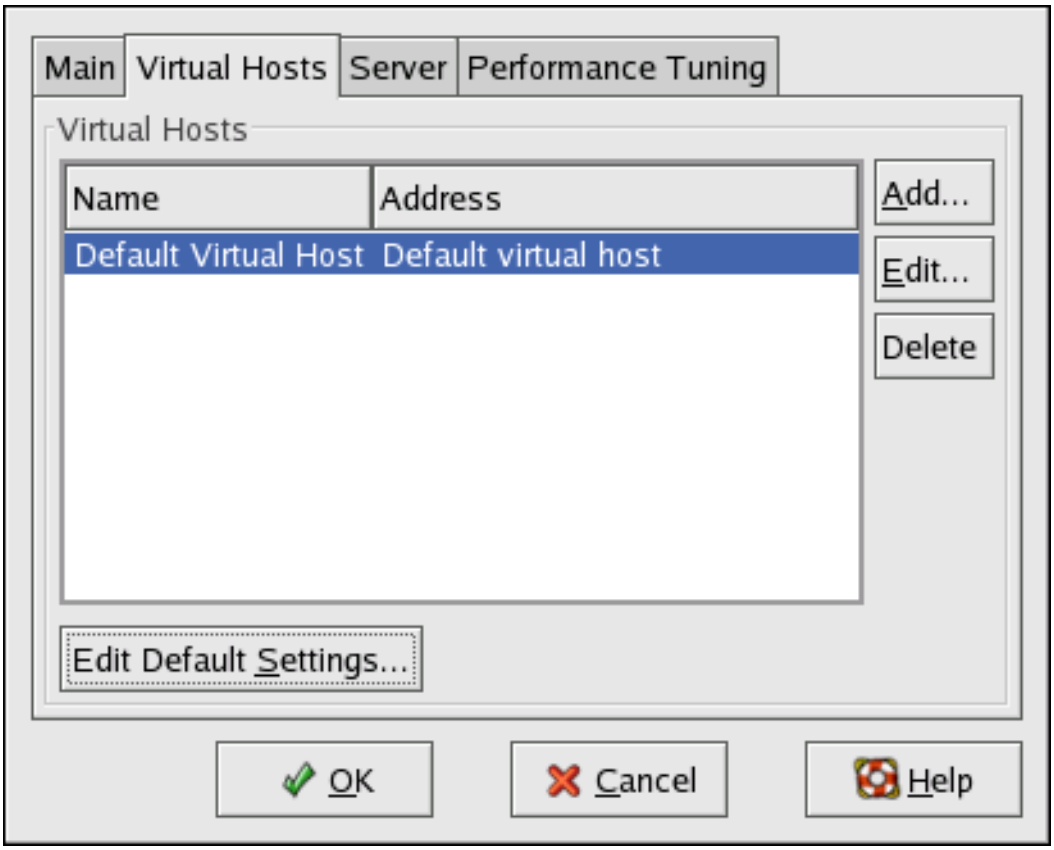


[D]

25.4.2. デフォルトの設定

Server Name、Webmaster メールアドレス、および Available Addresses を定義したら、仮想ホスト タブをクリック します。以下の図は、仮想ホスト タブを 示しています。

図25.3 仮想ホストタブ

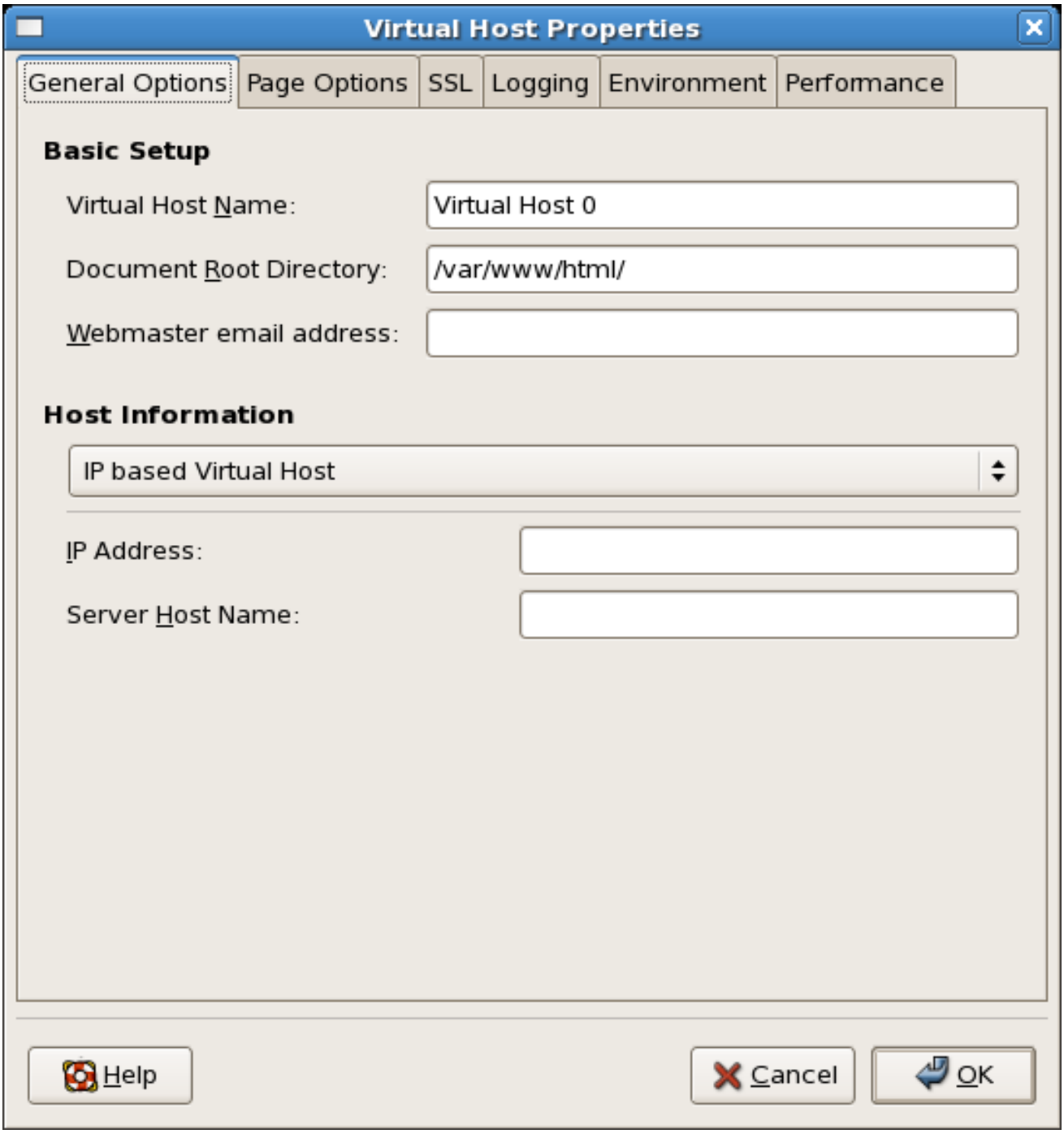


[D]

Edit をクリックすると、好みの設定が可能な 仮想ホストプロパティー ウィンドウが表示されます。新しい設定を追加するには、Add ボタンをクリックします。これにより、Virtual Host Properties ウィンドウも表示されます。Edit Default Settings ボタンをクリックし、General Options タブがない Virtual Host Properties ウィンドウを表示します。

General Options タブで、hostname、ドキュメント root ディレクトリー、および webmaster のメールアドレスを設定できます。ホスト情報では、仮想ホストの IP アドレスとホスト名を設定できます。以下の図は、General Options タブを示しています。

図25.4 一般的なオプション



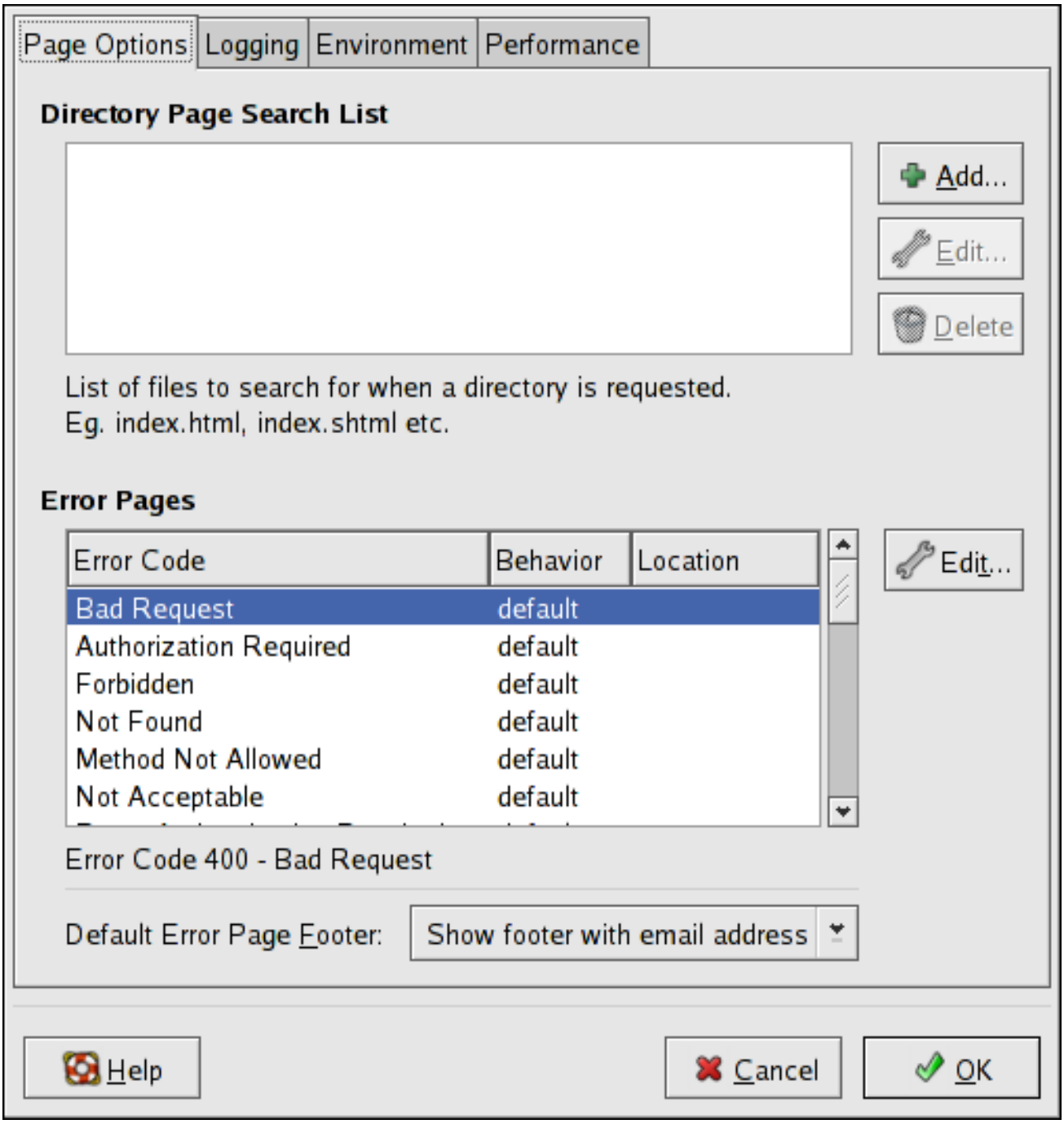
[D]

仮想ホストを追加すると、仮想ホストの設定が、その仮想ホストに優先されます。仮想ホスト設定内で定義されていないディレクティブでは、デフォルト値が使用されます。

25.4.2.1. サイト設定

以下の図は、Directory Page Search Listおよび Error Pages を設定するページオプション タブを示しています。これらの設定が分からない場合は、変更しないでください。

図25.5 サイト設定



[D]

Directory Page Search List に一覧表示されているエントリーは、DirectoryIndex ディレクティブを定義します。DirectoryIndex は、ディレクトリー名の最後にスラッシュ(/)を指定して、ユーザーがディレクトリーのインデックスを要求する際にサーバーによって提供されるデフォルトページです。

たとえば、ユーザーが http://www.example.com/this_directory/ ページを要求すると、DirectoryIndex ページ（存在する場合）またはサーバー生成ディレクトリー一覧のいずれかを取得します。サーバーは、DirectoryIndex ディレクティブにリストされているファイルの 1 つを見つけ、最初に見つかったファイルを返します。 これらのファイルが見つからない場合や、Options Indexes

がそのディレクトリーに設定されている場合、サーバーは ディレクトリー内のサブディレクトリーおよびファイルのリストを生成して HTML 形式で返します。

エラー コード セクションを 使用して、問題やエラーが発生した場合にクライアントをローカルまたは外部 URL にリダイレクトするように Apache HTTP Server を設定します。このオプションは ErrorDocument ディレクティブに対応します。クライアントが Apache HTTP Server に接続しようとすると問題やエラーが発生した場合、デフォルトのアクションは Error Code 列に表示される短いエラーメッセージを表示することです。このデフォルト設定を上書きするには、エラーコードを選択し、Edit ボタンをクリックします。Default を選択して、デフォルトの短いエラーメッセージを表示します。クライアントを外部 URL にリダイレクトする URL を選択し、Location フィールドに http:// を含む完全な URL を入力します。クライアントを内部 URL にリダイレクトするには File を選択し、Web サーバーのドキュメントルートの下にファイルの場所を入力します。場所はスラッシュ(/)を開始し、ドキュメントルートと相対的である必要があります。

たとえば、404 Not Found エラーコードを 404.html という名前のファイルで作成した Web ページにリダイレクトするには、404.html を DocumentRoot/../error/404.html にコピーします。この場合、DocumentRoot は定義したドキュメントルートディレクトリーです（デフォルトは /var/www/html/です）。Document Root がデフォルトの場所のままである場合は、ファイルを /var/www/error/404.html にコピーする必要があります。次に、404 - Not Found エラーコードの動作として File を選択し、/error/404.html を Location として入力します。

Default Error Page Footer メニューから、以下のオプションのいずれかを選択できます。

- メールアドレスでフッター を表示します：すべてのエラーページの下部にあるデフォルトのフッターと、ServerAdmin ディレクティブで指定された Web サイトメンテナーのメールアドレスを表示します。
- フッター の表示：エラーページの下部にあるデフォルトのフッターのみを表示します。
- No footer: エラーページの下部にフッターは表示されません。

25.4.2.2. SSL サポート

mod_ssl は、SSL 経由で HTTP プロトコルの暗号化を有効にします。SSL (Secure Sockets Layer)プロトコルは、TCP/IP ネットワークを介した通信および暗号化に使用されます。SSL タブを使用すると、サーバーの SSL を設定できます。SSL を設定するには、以下へのパスを指定する必要があります。

- 証明書ファイル：PEM (Privacy Enhanced Mail)でエンコードされたサーバー証明書ファイルへのパスを参照する SSLCertificateFile ディレクティブの使用と同じです。

- キーファイル：PEM でエンコードされたサーバーの秘密鍵ファイルへのパスを参照する SSLCertificateKeyFile ディレクティブの使用と同じです。

- 証明書チェーンファイル：証明書のすべてのサーバーのチェーンを含む証明書ファイルへのパスを参照する SSLCertificateChainFile ディレクティブの使用と同じです。

- 認証局ファイル：サーバーと通信する信頼性またはアイデンティティーを確認するために使用される暗号化されたファイルです。

SSL の設定ディレクティブについては、http://httpd.apache.org/docs/2.2/mod/directives.html#S を参照してください。また、有効にする SSL オプションを決定する必要もあります。これらは、以下のオプションで SSLOptions を使用するのと同じです。

- FakeBasicAuth: Apache で使用される標準の認証方法を有効にします。これは、Client X509 証明書の Subject Distinguished Name (DN)が基本的な HTTP ユーザー名に変換されることを意味します。

- ExportCertData - SSL_SERVER_CERT、SSL_CLIENT_CERT、および SSL_CLIENT_CERT _CHAIN_n に CGI 環境変数を作成します。n は数字 0,1,2,3,4. です。これらのファイルは、CGI スクリプトによるより多くの証明書チェックに使用されます。

- CompatEnvVars - CGI 環境変数を追加して、Apache SSL の後方互換性を有効にします。

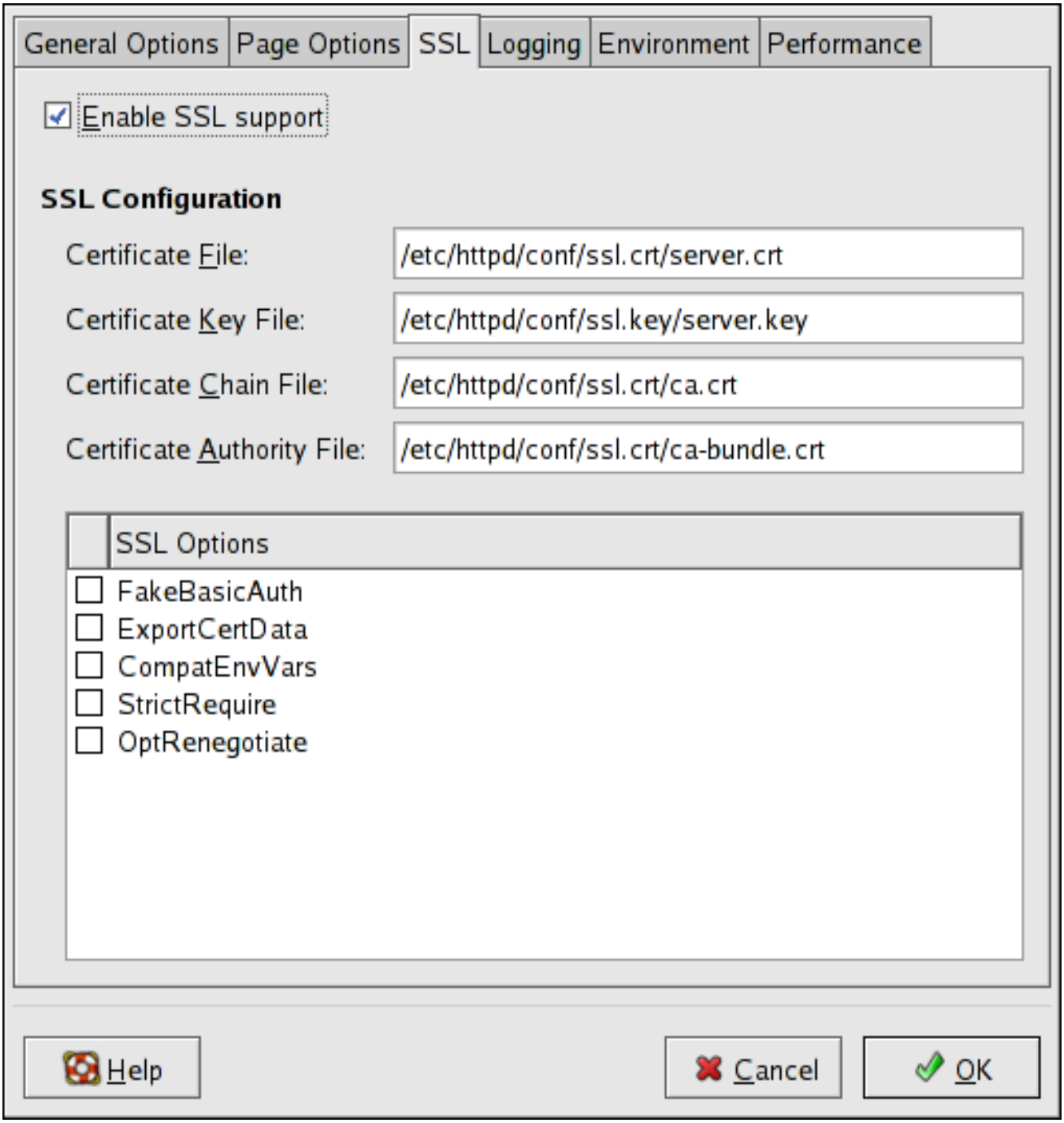
- StrictRequire: SSLRequireSSL ディレクティブおよび SSLRequire ディレクティブがアクセスが禁止されていることが示唆されるたびに、アクセスの拒否を強制する厳密なアクセスを有効にします。

- OptRenegotiate: mod_ssl による不要なハンドシェイクの回避を有効にし、安全なパラメーターチェックも実行します。ディレクトリーごとに OptRenegotiate を有効にすることが推奨されます。

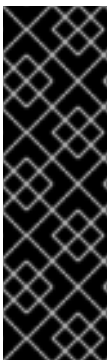
上記の SSL オプションの詳細は、http://httpd.apache.org/docs/2.2/mod/mod_ssl.html#ssloptions を参照してください。以下の図は、

SSL タブと、上記で説明したオプションを示しています。

図25.6 SSL



[D]



重要

『POODLE: SSLv3 脆弱性(CVE-2014-3566)で説明されている脆弱性』 により、Red Hat は SSL を無効にし、TLSv1. 1 または TLSv1.2 のみを使用することを推奨します。後方互換性は、TLSv1.0 を使用して実現できます。Red Hat がサポートする多くの製品は SSLv2 プロトコルまたは SSLv3 プロトコルを使用するか、デフォルトでそれらのプロトコルを有効にできます。ただし、SSLv2 または SSLv3 を使用することが強く推奨されます。

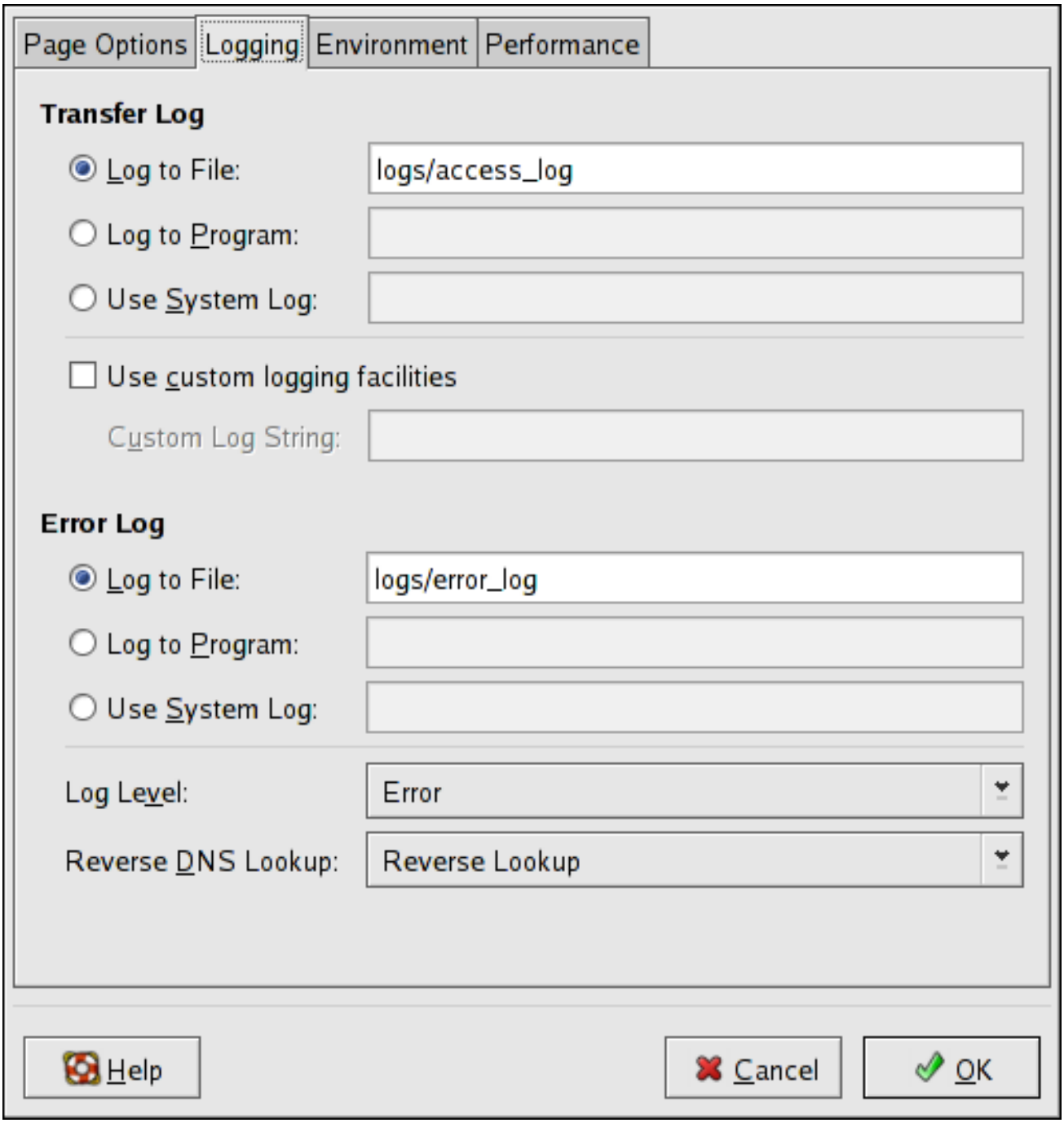
25.4.2.3. ロギング

Logging タブを使用して、特定の転送およびエラーログのオプションを設定します。

デフォルトでは、サーバーは転送ログを /var/log/httpd/access_log ファイルに書き込み、エラーログを /var/log/httpd/error_log ファイルに書き込みます。

転送ログには、Web サーバーへのアクセス試行の全一覧が含まれます。接続しようとしているクライアントの IP アドレス、試行の日時、取得しようとしている Web サーバー上の ファイルを記録します。この情報を保存するパスおよびファイルの名前を入力します。パスおよびファイル名がスラッシュ (/)で始まりない場合、パスは設定されたサーバールートディレクトリーへの相対パスになります。このオプションは TransferLog ディレクティブに対応します。

図25.7 ロギング



[D]

Use custom logging 機能をチェックし、Custom Log String フィールドにカスタム ログ文字列を入力して、カスタムログ形式を設定できます。これにより、LogFormat ディレクティブが設定されます。このディレクティブの形式に関する詳細は、http://httpd.apache.org/docs/2.2/mod/mod_log_config.html#logformat を参照してください。

エラーログには、発生するサーバーエラーの一覧が含まれます。この情報を保存するパスおよびファイルの名前を入力します。パスおよびファイル名がスラッシュ(/)で始まりない場合、パスは設定されたサーバールートディレクトリーへの相対パスになります。このオプションは ErrorLog ディレクティブに対応します。

Log Level メニューを使用して、エラーログにエラーメッセージの詳細度を設定します。（最も詳細度の低いものから最も詳細度の高いものから詳細まで）設定して、Alerting、alert、cit、error、warn、notice、info、または debug に設定できます。このオプションは LogLevel ディレクティブに対応します。

Reverse DNS Lookup メニューで選択した値は、HostnameLookups ディレクティブを定義します。No Reverse Lookup を選択すると、値を off に設定します。Reverse Lookup を選択すると、値を on に設定します。Double Reverse Lookup を選択すると、値が double に設定されます。

Reverse Lookup を選択すると、サーバーは Web サーバーからドキュメントを要求する各接続の IP アドレスを自動的に解決します。IP アドレスを解決すると、特定の IP アドレスに対応するホスト名を見つけるためにサーバーが DNS に 1 つ以上の接続を行います。

Double Reverse Lookup を選択すると、サーバーはダブル逆引き DNS を実行します。つまり、リバースルックアップを実行すると、結果に対して転送ルックアップが実行されます。正引きルックアップの IP アドレスの少なくとも 1 つが、最初の逆引き参照のアドレスと一致する必要があります。

DNS 要求がサーバーに負荷を追加し、速度が遅くなる可能性があるため、このオプションを No Reverse Lookup に設定したままにする必要があります。サーバーがビジー状態になると、これらのリバースルックアップや 2 倍のリバースルックアップを実行しようとする影響は非常に顕著になる可能性があります。

リバースルックアップと 2 倍のリバースルックアップも、インターネット全体として問題となります。各ホスト名を検索するために確立された各接続が追加されます。したがって、独自の Web サーバーの利点とインターネットの利点については、このオプションを No Reverse Lookup に設定したままにする必要があります。

25.4.2.4. 環境変数

Environment タブを使用して、CGI スクリプトで設定、合格、または未設定にする特定の変数のオプションを設定します。

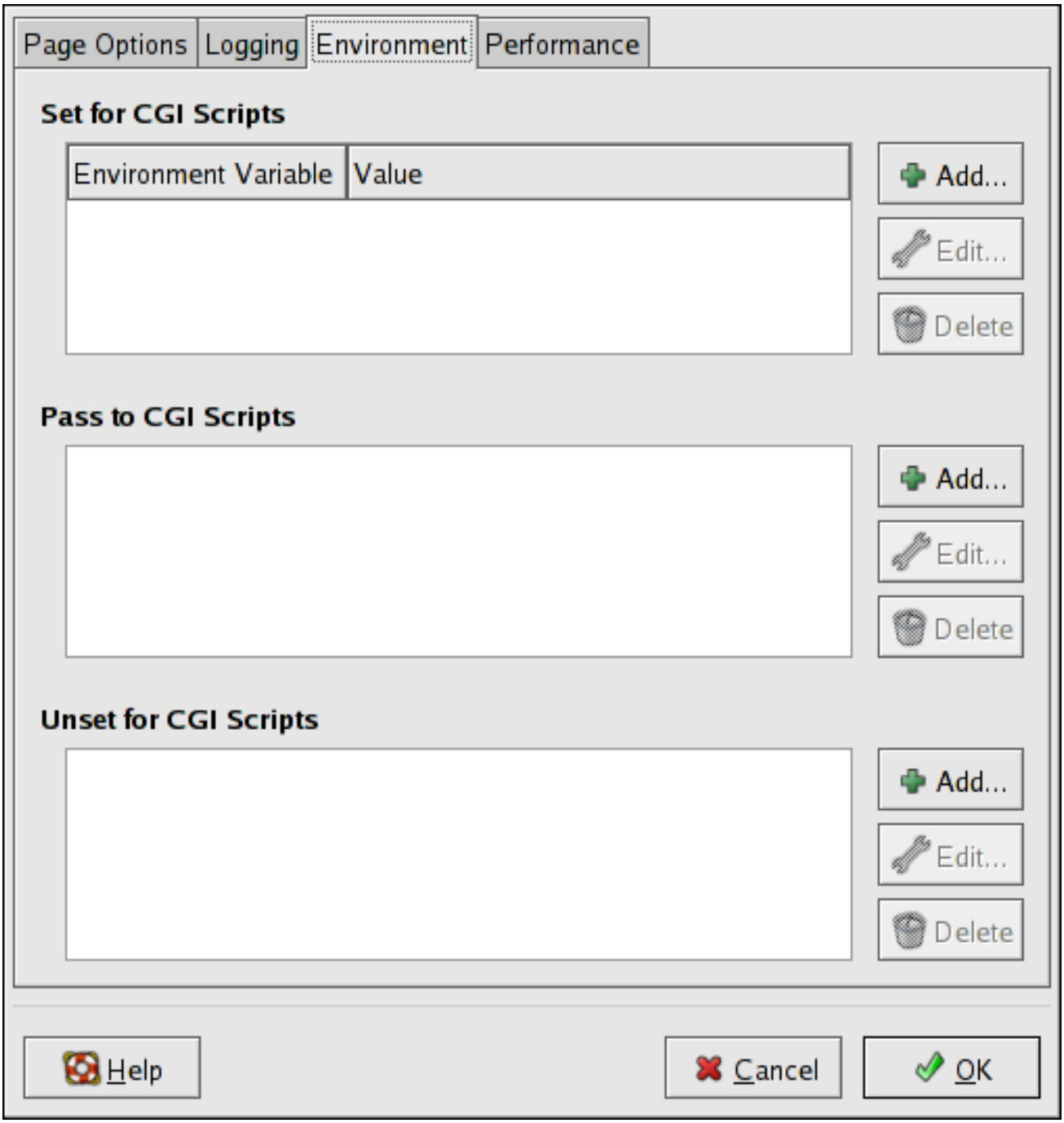
CGI スクリプトまたはサーバー側の include (SSI)ページの環境変数を変更する必要がある場合があります。Apache HTTP Server は mod_env モジュールを使用して、CGI スクリプトおよび SSI ページに渡される環境変数を設定できます。Environment Variables ページを使用して、このモジュールのディレクティブを設定します。

CGI Scripts の設定 セクションを使用して、CGI スクリプトおよび SSI ページに渡される環境変数を設定します。たとえば、環境変数 MAXNUM を 50 に設定するには、図25.8「環境変数」 に示すよう

に、CGI スクリプトの設定 セクションにある Add ボタンをクリックしてから、Value に Environment Variable テキストフィールドに MAXNUM と入力し、テキストフィールドを設定 します。50OK をクリックして一覧に追加します。Set for CGI Scripts セクションは、SetEnv ディレクティブを設定します。

サーバーが最初に CGI スクリプトで起動されたときに、Pass to CGI Scripts セクションを使用して環境変数の値を渡します。この環境変数を表示するには、シェルプロンプトでコマンド env を入力します。Pass to CGI Scripts セクション内の Add ボタンをクリックし、表示されるダイアログボックスに環境変数の名前を入力します。OK をクリックして一覧に追加します。Pass to CGI Scripts セクションでは、PassEnv ディレクティブを設定します。

図25.8 環境変数



[D]

値が CGI スクリプトおよび SSI ページに渡されないように環境変数を削除するには、CGI スクリプトの Unset for CGI Scripts セクションを使用します。Unset for CGI Scripts セクションで Add をクリックし、設定を解除する環境変数の名前を入力します。OK をクリックして一覧に追加します。これは UnsetEnv ディレクティブに対応します。

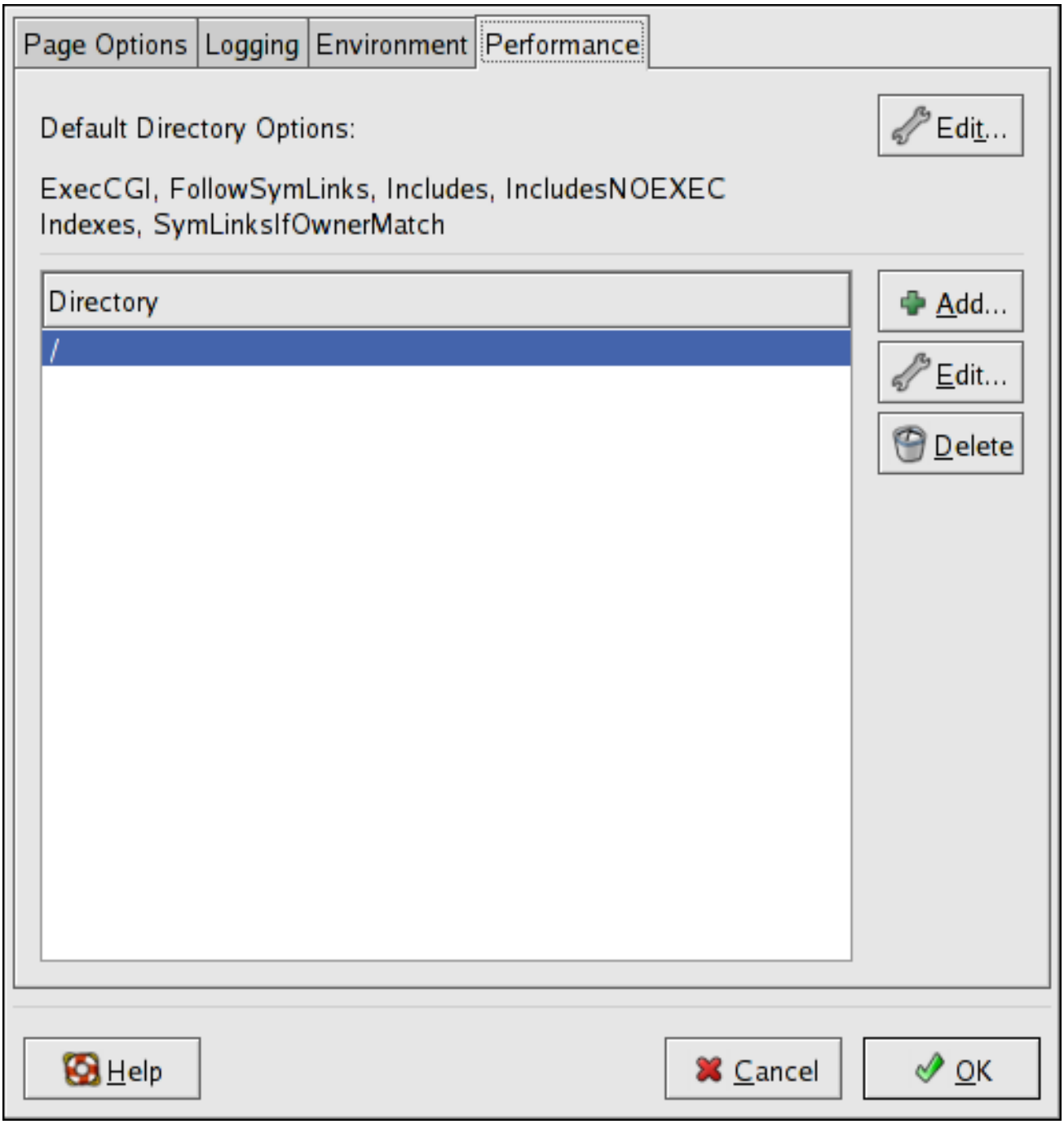
これらの環境値を編集するには、一覧から選択し、対応する 編集 ボタンをクリックします。一覧からエントリーを削除するには、エントリーを選択し、対応する Delete ボタンをクリックします。

Apache HTTP Server の環境変数の詳細は、http://httpd.apache.org/docs/2.2/env.htmlを参照してください。

25.4.2.5. ディレクトリー

Performance タブの directories ページを使用して、特定のディレクトリーのオプションを設定します。これは < Directory> ディレクティブに対応します。

図25.9 ディレクトリー



[D]

右上の Edit ボタンをクリックして、その下の Directory リストで指定されていないすべてのディレクトリーの Default Directory Options を設定します。選択したオプションは、< Directory > ディレクティブ内の Options ディレクティブとして一覧表示されます。以下のオプションを設定できます。

- *ExecCGI - CGI スクリプトの実行を許可します。このオプションが選択されていない場合、CGI スクリプトは実行されません。*

- *FollowSymLinks: シンボリックリンクの後に続くことを許可します。*

- *includes: サーバー側のインクルードを許可します。*

- *IncludesNoExec - サーバー側のインクルードを許可しますが、CGI スクリプトで #exec および #include コマンドを無効にします。*

- *Indexes - DirectoryIndex (index.htmlなど)が要求されたディレクトリーに存在しない場合、ディレクトリーのコンテンツのフォーマットされたリストを表示します。*

- *Multiview - content-negotiated multiviews をサポートします。このオプションはデフォルトで無効にされています。*

- *SymLinksIfOwnerMatch - ターゲットファイルまたはディレクトリーがリンクと同じ所有者を持つ場合にのみシンボリックリンクをたどります。*

特定のディレクトリーにオプションを指定するには、Directory リストボックスの横にある Add ボタンをクリックします。図25.10「ディレクトリーの設定」 に示されるようにウィンドウが表示されます。ウィンドウ下部の Directory テキストフィールドに、設定するディレクトリーを入力します。右側の一覧でオプションを選択し、左側のオプションで Order ディレクティブを設定します。Order ディレクティブは、allow ディレクティブと deny ディレクティブが評価される順序を制御します。Allow hosts from and Deny hosts from text フィールドで、以下のいずれかを指定できます。

- *Allow all hosts: all と入力して、すべてのホストへのアクセスを許可します。*

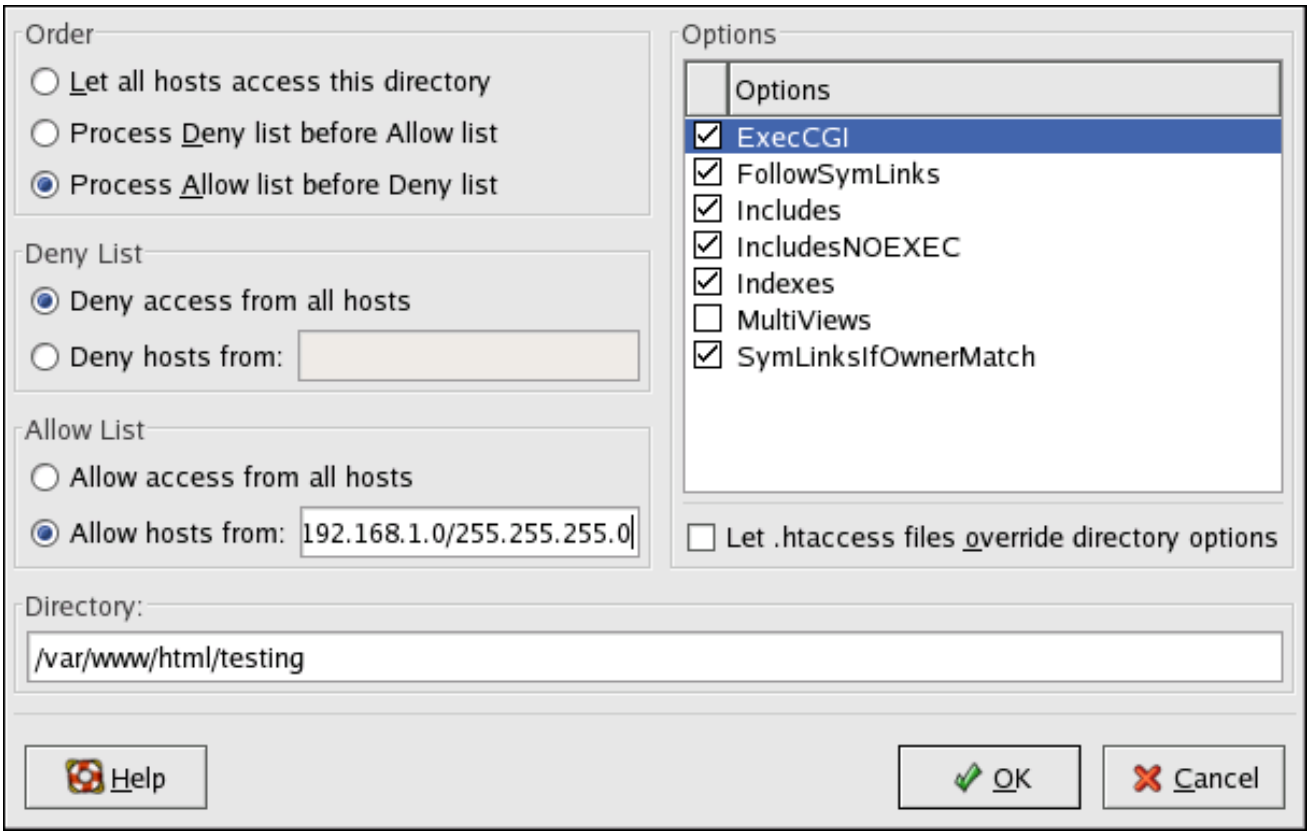
- *パーシャルドメイン名：名前が指定の文字列と一致するか、または終了されるすべてのホストを許可します。*

- *完全な IP アドレス：特定の IP アドレスへのアクセスを許可します。*

- *サブネット：の場合は以下のようになります。 192.168.1.0/255.255.255.0*

- *ネットワーク CIDR 仕様（例：） 10.3.0.0/16*

図25.10 ディレクトリーの設定



[D]

Let .htaccess ファイルがディレクトリーオプション を上書き する場合は、.htaccess ファイルの設定ディレクティブが優先されます。

25.5. HTTPD.CONFの設定ディレクティブ

Apache HTTP Server 設定ファイルは /etc/httpd/conf/httpd.conf です。httpd.conf ファイルは十分にコメント化されており、ほとんどは自己計画的です。デフォルト設定はほとんどの状況で機能しますが、より重要な設定オプションの一部を理解することが推奨されます。

WARNING

Apache HTTP Server 2.2 のリリースに伴い、多くの設定オプションが変更されました。バージョン 1.3 から 2.2 に移行する場合は、まず「Apache HTTP Server 1.3 設定ファイルの 2.0 への移行」をお読みください。

25.5.1. 一般的な設定のヒント

Apache HTTP Server を設定する場合は、/etc/httpd/conf/httpd.conf を編集し、「httpdの起動と停止」 で概説するように、/etc/httpd/conf/httpd.conf を編集し、httpd プロセスを開始します。

httpd.conf を編集する前に、元の ファイルのコピーを作成します。バックアップを作成すると、設定ファイルの編集中に間違いからの復旧が容易になります。

間違いが発生し、Web サーバーが正しく機能しない場合は、最初に httpd.conf で編集したパスを確認し、誤字がないことを確認します。

次に、Web サーバーのエラーログ /var/log/httpd/error_log を確認します。変更ログは、専門知識のレベルによっては、解釈が簡単ではない可能性があります。ただし、エラーログの最後のエントリーは、有用な情報を提供します。

以下のサブセクションには、httpd.conf に含まれるディレクティブの多くについての簡単な説明の一覧が記載されています。これらの説明は網羅的なものではありません。詳細は、オンラインの Apache ドキュメンテーション(http://httpd.apache.org/docs/2.2/)を参照してください。

mod_ssl ディレクティブの詳細は、http://httpd.apache.org/docs/2.2/mod/mod_ssl.html でオンラインのドキュメントを参照してください。

AccessFileName

AccessFileName は、サーバーが各ディレクトリーのアクセス制御情報に使用するファイルに名前を付けます。デフォルトは .htaccess です。

AccessFileName ディレクティブの直後に、Files タグのセットは、.ht で始まるファイルにアクセス制御を適用します。これらのディレクティブは、セキュリティー上の理由から、すべての .htaccess ファイル（または .htで始まるその他のファイル）への Web アクセスを拒否します。

アクション

action は MIME コンテンツタイプと CGI スクリプトのペアを指定して、そのメディアタイプのファイルが要求されたときに特定の CGI スクリプトが実行されるようにします。

AddDescription

FancyIndexing を IndexOptions パラメーターとして使用する場合、AddDescription ディレクティブを使用して、サーバーが生成したディレクトリー一覧に特定のファイルまたはファイルタイプの

ユーザー指定の説明を表示できます。AddDescription ディレクティブは、特定のファイル、ワイルドカード式、またはファイル拡張子の一覧表示をサポートします。

AddEncoding

AddEncoding は、特定のエンコーディングタイプを指定するファイル名の拡張子です。AddEncoding を使用して、一部のブラウザーに、ダウンロード時に特定のファイルを圧縮解除するように指示することもできます。

AddHandler

addhandler は、 ファイル拡張子を特定のハンドラーにマッピングします。たとえば、cgi-script ハンドラーを拡張子 .cgi に一致させると、.cgi で終わるファイルを CGI スクリプトとして自動的に扱います。以下は、.cgi 拡張の AddHandler ディレクティブの例です。

```
AddHandler cgi-script .cgi
```

このディレクティブにより、cgi-bin 外の CGI が、ディレクトリーコンテナー内の ExecCGI オプションを持つサーバー上の任意のディレクトリーで機能できるようになります。ディレクトリーの ExecCGI オプションの設定に関する詳細は、ディレクトリー を参照してください。

CGI スクリプトの他に、AddHandler ディレクティブを使用して、サーバー解析された HTML およびイメージマップファイルを処理します。

AddIcon

AddIcon は、特定の拡張子を持つファイルのサーバー生成ディレクトリー一覧に表示するアイコンを指定します。たとえば、Web サーバーは、拡張子が .bin または .exe のファイルのアイコン binary.gif を表示するように設定されます。

AddIconByEncoding

このディレクティブは、サーバーが生成したディレクトリー一覧の MIME エンコーディングのあるファイルによって表示されるアイコンの名前。たとえば、デフォルトでは、Web サーバーは、サーバー生成されたディレクトリー一覧にある MIME でエンコードされた x 圧縮ファイルと x-gzip ファイルの横に、compress.gif アイコンを表示します。

AddIconByType

このディレクティブの名前アイコンは、サーバー生成ディレクトリー一覧に MIME タイプのファイルの横に表示されます。たとえば、サーバーは、サーバーが生成したディレクトリー一覧に mime-type のテキストタイプのファイルの横にあるアイコン text.gif を表示します。

AddLanguage

AddLanguage は、ファイル名の拡張子を特定の言語に関連付けます。このディレクティブは、クライアントの Web ブラウザーの言語設定に基づいて複数の言語でコンテンツを提供する Apache HTTP Server に便利です。

AddType

AddType ディレクティブを使用して、デフォルトの MIME タイプとファイル拡張子のペアを定義または上書きします。以下のサンプルディレクティブは、.tgz ファイル拡張子を認識するように Apache HTTP Server に指示します。

```
AddType application/x-tar .tgz
```

エイリアス

Alias 設定により、DocumentRoot ディレクトリー以外のディレクトリーにアクセスできます。エイリアスで終わる URL は、自動的にエイリアスのパスに対して解決されます。デフォルトでは、アイコン/ ディレクトリーにエイリアスが 1 つ設定されています。icons/ ディレクトリーには Web サーバーからアクセスできますが、ディレクトリーは DocumentRoot には含まれません。

許可

allow は、指定のディレクトリーにアクセスできるクライアントを指定します。クライアントは、すべて、ドメイン名、IP アドレス、部分的な IP アドレス、ネットワーク/ネットマスクのペアなどにすることができます。DocumentRoot ディレクトリーは、すべての からのリクエストを 許可 するよう設定されます。つまり、すべて のユーザーがアクセスできます。

AllowOverride

AllowOverride ディレクティブは、任意の オプション が .htaccess ファイルの宣言で上書きできるかどうかを設定します。デフォルトでは、root ディレクトリーと DocumentRoot の両方が、.htaccess の上書きを許可しないように設定されています。

BrowserMatch

BrowserMatch ディレクティブにより、サーバーは環境変数を定義し、クライアントの Web ブラウザータイプを識別する User-Agent HTTP ヘッダーフィールドに基づいて適切なアクションを実行できます。デフォルトでは、Web サーバーは BrowserMatch を使用して、既知の問題のある特定のブラウザーへの接続を拒否し、またこれらのアクションに問題があることがわかっているブラウザーの keepalive および HTTP ヘッダーのフラッシュを無効にします。

キャッシュディレクティブ

デフォルトの Apache HTTP Server 設定ファイルでは、コメント化されたキャッシュディレクティブが多数提供されています。ほとんどの場合、行頭からハッシュ記号(#)を削除して、このような行

をコメント解除すれば十分です。ただし、以下は、より重要なキャッシュ関連のディレクティブのリストです。

- *CacheEnable: キャッシュがディスク、メモリー、またはファイル記述子キャッシュであるかを指定します。デフォルトでは、CacheEnable は / の下にある URL のディスクキャッシュを設定します。*

- *CacheRoot - キャッシュされたファイルを含むディレクトリーの名前を指定します。デフォルトの CacheRoot は /var/httpd/proxy/ ディレクトリーです。*

- *cache size: キャッシュが使用できる領域をキロバイト単位で指定します。デフォルトの CacheSize は 5 KB です。*

以下は、他の一般的なキャッシュ関連のディレクティブの一覧です。

- *CacheMaxExpire: キャッシュに HTML ドキュメントが保持される期間（元の Web サーバーからのリロードなし）を指定します。デフォルトは 24 時間(86400 秒)です。*

- *CacheLastModifiedFactor - 独自の有効期限が設定された元のサーバーから行われなかったドキュメントの期限切れ（有効期限）の日付を指定します。デフォルトの CacheLastModifiedFactor は 0.1 に設定されています。つまり、このようなドキュメントの有効期限は、ドキュメントが最後に変更された時点の 1 回目と等しくなります。*

- *CacheDefaultExpire: 有効期限をサポートしないプロトコルを使用して受信されたドキュメントの有効期限を時間単位で指定します。デフォルトは 1 時間(3600 秒)に設定されます。*

- *noProxy: コンテンツがキャッシュされないサブネット、IP アドレス、ドメイン、またはホストのスペース区切りの一覧を指定します。この設定は、イントラネットサイトで最も役立ちます。*

CacheNegotiatedDocs

デフォルトでは、Web サーバーは、コンテンツベースでネゴシエートされたドキュメントをキャッシュしないようにプロキシーサーバーに要求します（つまり、時間の経過とともに、または要求元からの入力が変わる可能性があります）。で CacheNegotiatedDocs が に設定されている場合、この機能は無効になり、プロキシーサーバーはそのようなドキュメントをキャッシュできます。

CustomLog

CustomLog は、ログファイルとログファイルの形式を識別します。デフォルトでは、アクセスログは /var/log/httpd/access_log ファイルに記録され、エラーは /var/log/httpd/error_log ファイルに記録されます。

デフォルトの CustomLog 形式は、以下に示すように 結合された ログファイル形式です。

```
remotehost rfc931 user date "request" status bytes referrer user-agent
```

DefaultIcon

DefaultIcon は、他のアイコンが指定されていないファイルのサーバー生成ディレクトリーリストに表示されるアイコンを指定します。unknown.gif イメージファイルがデフォルトです。

DefaultType

DefaultType は、MIME タイプを判断できないドキュメントに使用する Web サーバーのデフォルトコンテンツタイプを設定します。デフォルトは text/plain です。

却下

deny は Allow と同様に機能しますが、アクセスが拒否されるユーザーを指定します。DocumentRoot は、デフォルトで誰からでも要求を 拒否 するように設定されていません。

ディレクトリー

& lt;directory /path/to/directory > タグおよび </ Directory > タグにより、特定のディレクトリーとそのサブディレクトリーにのみ適用される設定ディレクティブのグループを囲むために使用されるコンテナーが作成されます。ディレクトリーに適用されるディレクティブは、ディレクトリータグ内で使用できます。

デフォルトでは、非常に制限のあるパラメーターはルートディレクトリー(/)に適用され、Options (オプションを参照)および AllowOverride (AllowOverrideを参照)ディレクティブを使用します。この設定では、より多くの許容設定を必要とするシステム上の任意のディレクトリーを明示的に指定する必要があります。

デフォルト設定では、別の Directory コンテナーは、ディレクトリーツリーに柔軟性の低いパラメーターを割り当てる DocumentRoot 用に設定され、Apache HTTP Server がそこにあるファイルにアクセスできるようにします。

ディレクトリーコンテナー は、ScriptAlias ディレクティブで指定されたディレクトリー外にあるサーバー側のアプリケーションの追加 cgi-bin ディレクトリーを設定するために使用することもできま

す（詳細は、ScriptAlias を参照してください）。

これを実行するには、Directory コンテナーは、そのディレクトリーに ExecCGI オプションを設定する必要があります。

たとえば、CGI スクリプトが /home/my_cgi_directory にある場合は、以下のディレクトリーコンテナー を httpd.conf ファイルに追加します。

```
<Directory /home/my_cgi_directory>
  Options +ExecCGI
</Directory>
```

次に、AddHandler ディレクティブのコメントを解除して、.cgi 拡張子を持つファイルを CGI スクリプトとして識別する必要があります。AddHandler の設定手順については、AddHandler を参照してください。

これを機能させるには、CGI スクリプトのパーミッションとスクリプトへのパス全体を 0755 に設定する必要があります。

DirectoryIndex

DirectoryIndex は、ディレクトリー名の最後にスラッシュ(/)を指定して、ユーザーがディレクトリーのインデックスを要求する際にサーバーによって提供されるデフォルトページです。

ユーザーが http://example/this_directory/ ページを要求すると、DirectoryIndex ページ（存在する場合）またはサーバー生成ディレクトリー一覧のいずれかを取得します。DirectoryIndex のデフォルトは index.html で、index.html.var タイプマップです。サーバーはこれらのファイルのいずれかを検索し、最初に見つかったファイルを返します。これらのファイルのいずれかが見つからず、Options Indexes がそのディレクトリーに設定されている場合は、ディレクトリーリスト機能がオフになっていない限り、サーバーはサブディレクトリーとファイルの一覧を HTML 形式で生成し、返します。

DocumentRoot

DocumentRoot は、リクエストに応答して提供される HTML ファイルのほとんどが含まれるディレクトリーです。セキュアではないセキュアな Web サーバー両方のデフォルトの DocumentRoot は /var/www/html ディレクトリーです。たとえば、サーバーは以下のドキュメントのリクエストを受け取る場合があります。

```
http://example.com/foo.html
```

サーバーは、デフォルトの ディレクトリーで以下のファイルを検索します。

```
/var/www/html/foo.html
```

セキュアおよびセキュアではない Web サーバーで共有されないように DocumentRoot を変更するには、「仮想ホスト」 を参照してください。

errorDocument

ErrorDocument ディレクティブは、HTTP 応答コードをクライアントに送信するメッセージまたは URL に関連付けます。デフォルトでは、Web サーバーはエラーが発生した場合に単純なエラーメッセージおよび通常は暗号的なエラーメッセージを出力します。ErrorDocument ディレクティブは、代わりにカスタマイズされたメッセージまたはページを出力するよう Web サーバーを強制します。

重要な影響

有効なようにするには、メッセージを二重引用符 " のペアで囲む 必要があり ます。

ErrorLog

errorlog は、サーバーエラーがログ に記録されるファイルを指定します。デフォルトでは、このディレクティブは /var/log/httpd/error_log に設定されます。

ExtendedStatus

ExtendedStatus ディレクティブは、server-status ハンドラーが呼び出されたときに Apache が基本(off)または詳細なサーバーステータス情報（）を生成するかどうかを制御します。server-status ハンドラーは Location タグを使用して呼び出されます。server-status の呼び出しに関する詳細情報は、場所 に含まれています。

Group

Apache HTTP Server プロセスのグループ名を指定します。

このディレクティブは、仮想ホストの設定で非推奨になりました。

デフォルトでは、Group は apache に設定されます。

HeaderName

HeaderName は、ディレクトリーに存在するファイルに名前を付けます。このファイルが、サーバーが生成したディレクトリー一覧の先頭に追加されます。ReadmeName と同様に、サーバーは可能な場合、またはプレーンテキストで HTML ドキュメントとして追加しようとします。

HostnameLookups

HostnameLookups は、on、off、または double に設定できます。で HostnameLookups が に設定されている場合、サーバーは各接続の IP アドレスを自動的に解決します。IP アドレスを解決すると、サーバーは DNS サーバーへの 1 つ以上の接続を行い、処理のオーバーヘッドが追加されます。HostnameLookups が double に設定されている場合、サーバーはダブルリバース DNS 検索を実行し、処理オーバーヘッドをさらに追加します。

サーバー上のリソースを節約するために、HostnameLookups はデフォルトで off に設定されます。

サーバーログファイルにホスト名が必要な場合は、Web サーバーのログファイルのローテーション時に DNS ルックアップをより効率的にかつ一括で実行する多くのログアナライザーツールの 1 つを実行することを検討してください。

IfDefine

IfDefine タグは、IWDefine タグが true と指定された test の場合に適用される設定ディレクティブを囲む。テストが false の場合、ディレクティブは無視されます。

IfDefine タグのテストはパラメーター名（例： HAVE_PERL）です。パラメーターが定義されている場合、これはサーバーの start-up コマンドに引数として提供されることを意味します。これは、テストが true であることを意味します。この場合、Web サーバーが起動すると、テストは true で、if Define タグに含まれるディレクティブが 適用されます。

IfModule

& lt;IfModule > タグおよび </ IfModule > タグは、指定されたモジュールが読み込まれている場合にのみアクティベートされる条件付きコンテナーを作成します。IfModule コンテナー内のディレクティブは、2 つの条件のいずれかで処理されます。ディレクティブは、開始 <If Module> タグ内に含まれるモジュールがロードされている場合 に処理されます。または、感嘆符 ! がモジュール名の前に表示される場合、ディレクティブは <If Module> タグで指定されたモジュールが読み込まれて いない 場合にのみ処理されます。

Apache HTTP Server モジュールの詳細は、「モジュールの追加」 を参照してください。

包含

include を使用すると、起動時に他の設定ファイルを含めることができます。

これらの設定ファイルへのパスは、ServerRoot に対する絶対または相対パスになります。

重要な影響

サーバーが個別にパッケージ化されたモジュール(mod_ssl、mod_perl、php など) を使用するには、以下のディレクティブを セクション 1: httpd.conf のグローバル環境 に含める必要があります。

```
Include conf.d/*.conf
```

IndexIgnore

index ignore は、ファイルの拡張子、部分的なファイル名、ワイルドカード式、または完全なファイル名を一覧表示します。Web サーバーには、サーバーが生成したディレクトリー一覧内のこれらのパラメーターに一致するファイルは含まれません。

IndexOptions

IndexOptions は、アイコン、ファイルの説明などを追加して、生成されたサーバーダイレクトリストの外観を制御します。Options Indexes が設定されている場合(オプションを参照)、Web サーバーは、Web サーバーがインデックスなしでディレクトリーの HTTP 要求を受信するとディレクトリー一覧を生成します。

まず、Web サーバーは、DirectoryIndex ディレクティブ（通常は index.html）に一覧表示されている名前に一致するファイルを検索します。index.html ファイルが見つからない場合、Apache HTTP Server は要求されたディレクトリーの HTML ディレクトリーのリストを作成します。このディレクトリー一覧の表示は、一部では IndexOptions ディレクティブによって制御されます。

デフォルト設定は FancyIndexing をオンにします。つまり、ユーザーは列ヘッダーをクリックするとディレクトリーの一覧をソートできます。別の方法では、同じヘッダースイッチを昇順から降順に切り替えます。FancyIndexing は、ファイル拡張子に基づいて、異なるファイルの異なるアイコンも表示します。

AddDescription オプションは、FancyIndexing と併用すると、サーバーが生成したディレクトリー一覧にファイルの簡単な説明を表示します。

IndexOptions には、サーバーが生成したディレクトリーの外観を制御するために設定できる他のパ

ラメーターが多数あります。IconHeight パラメーターおよび IconWidth パラメーターでは、サーバーが生成した Web ページのアイコンに HTML HEIGHT および WIDTH タグを含める必要があります。IconsAreLinks パラメーターは、グラフィカルアイコンと URL リンクターゲットを含む HTML リンクアンカーを組み合わせます。

KeepAlive

keepalive は、サーバーが接続ごとに複数の要求を許可するかどうかを設定します。また、1 つのクライアントがサーバーのリソースを過剰に消費できないようにするために使用できます。

デフォルトでは、Keepalive は off に設定されています。Keepalive が on に設定され、サーバーが非常にビジー状態になると、サーバーは子プロセスの最大数を迅速に生成できます。このような場合、サーバーが大幅に低下します。Keepalive が有効になっている場合は、KeepAliveTimeout low を設定し(KeepAliveTimeout ディレクティブの詳細は KeepAliveTimeout を参照してください)、サーバー上の /var/log/httpd/error_log ログファイルを監視することが推奨されます。このログは、サーバーが子プロセスが不足すると報告されます。

KeepAliveTimeout

KeepAliveTimeout は、リクエストが提供されてから接続を閉じるまでにサーバーが待機する秒数を設定します。サーバーがリクエストを受信すると、代わりに Timeout ディレクティブが適用されます。KeepAliveTimeout ディレクティブは、デフォルトで 15 秒に設定されています。

LanguagePriority

LanguagePriority は、クライアントの Web ブラウザーに言語設定が設定されていない場合に、異なる言語の優先順位を設定します。

listen

Listen コマンドは、Web サーバーが受信要求を受け入れるポートを識別します。デフォルトでは、Apache HTTP Server はセキュアではない Web 通信ではポート 80 をリッスンするように設定されています。セキュアな Web 通信のためにポート 443 番のポート（セキュアなサーバーを定義する /etc/httpd/conf.d/ssl.conf ファイル）です。

Apache HTTP Server が 1024 未満のポートをリッスンするように設定されている場合、root ユーザーのみがこれを起動できます。ポート 1024 以降では、通常のユーザーとして httpd を起動できます。

Listen ディレクティブを使用して、サーバーが接続を受け入れる特定の IP アドレスを指定することもできます。

LoadModule

LoadModule は、Dynamic Shared Object (DSO)モジュールを読み込むために使用されます。LoadModule ディレクティブの使用手順など、Apache HTTP Server の DSO サポートの詳細は、「モジュールの追加」 を参照してください。モジュールの読み込み順序は、Apache HTTP Server 2.0 では 重要ではなくなりました。Apache HTTP Server 2.0 DSO サポートの詳細は、「Dynamic Shared Object (DSO)のサポート」 を参照してください。

場所

&lt ;Location& gt; および </Location > タグは、URL に基づくアクセス制御を指定するコンテナーを作成します。

たとえば、サーバーのドメイン内から接続したユーザーがステータスレポートを表示できるようにするには、以下のディレクティブを使用します。

```
<Location /server-status>
  SetHandler server-status
  Order deny,allow
  Deny from all
  Allow from <.example.com>
</Location>
```

& lt;.example.com& gt; を Web サーバーの 2 次ドメイン名に置き換えます。

ドメイン内からの要求にサーバー設定レポート（インストールされたモジュールおよび設定ディレクティブを含む）を提供するには、以下のディレクティブを使用します。

```
<Location /server-info>
  SetHandler server-info
  Order deny,allow
  Deny from all
  Allow from <.example.com>
</Location>
```

ここでも、<.example.com > を Web サーバーの 2 次ドメイン名に置き換えます。

LogFormat

LogFormat ディレクティブは、さまざまな Web サーバーログファイルの形式を設定します。使用される実際の LogFormat は、CustomLog ディレクティブで指定された設定によって異なります(CustomLogを参照してください)。

CustomLog ディレクティブが combined に設定されている場合の形式オプションを以下に示しま

す。

%H （リモートホストの IP アドレスまたはホスト名）

要求しているクライアントのリモート IP アドレスを一覧表示します。で HostnameLookups が に設定されている場合、クライアントのホスト名は DNS で利用できない限り記録されます。

%l (rfc931)

使用されていません。このフィールドのログファイルにハイフンが表示されます。

%u （認証ユーザー）

認証が必要な場合に記録されたユーザーのユーザー名を一覧表示します。通常、これは使用されないため、ハイフン - がこのフィールドのログファイルに表示されます。

%t (date)

リクエストの日時を一覧表示します。

%r （要求文字列）

ブラウザーまたはクライアントからの要求文字列を正確に一覧表示します。

%s （ステータス）

クライアントホストに返された HTTP ステータスコードを一覧表示します。

%b （バイト）

ドキュメントのサイズを一覧表示します。

%\"%{Referer}i\" (referrer)

クライアントホストを Web サーバーを参照する Web ページの URL を一覧表示します。

%\"%{user-Agent}i\" (user-agent)

リクエストを行う Web ブラウザーのタイプを一覧表示します。

LogLevel

logLevel は、エラーログのエラーメッセージの詳細を設定します。ログ レベルは、（最も詳細度の低いものから詳細まで）設定して、アラート、crit、error、warn、notice、info、または debug を設定できます。デフォルトの LogLevel は warn です。

MaxKeepAliveRequests

このディレクティブは、永続接続ごとに許可されるリクエストの最大数を設定します。Apache プロジェクトでは、サーバーのパフォーマンスが向上する高度な設定を推奨しています。MaxKeepAliveRequests はデフォルトで 100 に設定されています。これはほとんどの状況に適しています。

NameVirtualHost

NameVirtualHost ディレクティブは、必要に応じて名前ベースの仮想ホストに IP アドレスとポート番号を関連付けます。名前ベースの仮想ホストを使用すると、1 つの Apache HTTP Server が複数の IP アドレスを使用せずに異なるドメインを提供できるようになります。

注記

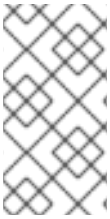
名前ベースの仮想ホストは、セキュアでない HTTP 接続でのみ 機能します。セキュアなサーバーで仮想ホストを使用する場合は、代わりに IP アドレスベースの仮想ホストを使用してください。

名前ベースの仮想ホストを有効にするには、NameVirtualHost 設定ディレクティブのコメントを解除し、正しい IP アドレスを追加します。次に、設定に必要なとおり、各仮想ホストの VirtualHost コンテナーを追加します。

オプション

Options ディレクティブは、特定のディレクトリーで利用可能なサーバー機能を制御します。たとえば、root ディレクトリーに指定される制限的なパラメーターでは、Options は FollowSymLinks ディレクティブにだけ設定されます。サーバーがルートディレクトリーのシンボリックリンクをたどることができることを除いて、機能は有効になっていません。

デフォルトでは、DocumentRoot ディレクトリーの Options は Indexes および FollowSymLinks が含まれるように設定されています。インデックス を使用すると、DirectoryIndex (index.htmlなど) が指定されていない場合、サーバーはディレクトリーのディレクトリー一覧を生成できます。FollowSymLinks を使用すると、サーバーはそのディレクトリー内のシンボリックリンクをたどることができます。



注記

メインサーバー設定セクションの options ステートメントは、各 VirtualHost コンテナーに個別に複製する必要があります。詳細は、VirtualHost を参照してください。

順序

Order ディレクティブは、allow ディレクティブと deny ディレクティブが評価される順序を制御します。サーバーは、DocumentRoot ディレクトリーの Deny ディレクティブの前に Allow ディレクティブを評価するように設定されています。

PidFile

pidfile は、 サーバーがプロセス ID (PID)を記録するファイルに名前を付けます。デフォルトでは、PID は /var/run/httpd.pid に一覧表示されます。

Proxy

<proxy *& gt ; および </Proxy > タグは、プロキシーサーバーにのみ適用される設定ディレクティブのグループを囲むコンテナーを作成します。< Directory > コンテナー内で許可されるディレクティブの多くは、< Proxy> コンテナー内 でも使用できます。

ProxyRequests

Apache HTTP Server がプロキシーサーバーとして機能するように設定するには、< IfModule mod_proxy.c > 行、ProxyRequests、および <Proxy> スタンザの各行からハッシュ マーク(#)を削除します。ProxyRequests ディレクティブを On に設定し、< Proxy> スタンザの Allow from ディレクティブでサーバーへのアクセスが許可されるドメインを設定し ます。

ReadmeName

ReadmeName は、ディレクトリー内に存在する場合は、サーバーが生成したディレクトリー一覧の最後に追加されるファイルに名前を付けます。Web サーバーは、最初に HTML ドキュメントとしてファイルを含めようとし、次にこれをプレーンテキストとして追加しようとします。デフォルトでは、ReadmeName は README.html に設定されています。

リダイレクト

Web ページを移動すると、リダイレクト を使用してファイルの場所を新しい URL にマップできま

す。形式は以下のとおりです。

```
Redirect /<old-path>/<file-name> http://<current-domain>/<current-path>/<file-name>
```

この例では、< old-path > を < file-name> および < current- domain > の古いパス情報に置き換え、< current-path > を、< file-name > の現在のドメインおよびパス情報に置き換えます。

この例では、古い場所にある & lt;file-name& gt; のリクエストは自動的に新しい場所にリダイレクトされます。

高度なリダイレクト手法については、Apache HTTP Server に含まれる mod_rewrite モジュールを使用します。mod_rewrite モジュールの設定に関する詳細は、http://httpd.apache.org/docs/2.2/mod/mod_rewrite.html でオンラインの Apache Software Foundation ドキュメントを参照してください。

ScriptAlias

ScriptAlias ディレクティブは、CGI スクリプトの場所を定義します。一般に、CGI スクリプトを DocumentRoot 内に残しておくことは適切ではありません。ここでは、テキストドキュメントとして参照できます。このため、サーバー側の実行ファイルとスクリプトを含む DocumentRoot ディレクトリー外の特別なディレクトリーは ScriptAlias ディレクティブによって指定されます。このディレクトリーは cgi-bin と呼ばれ、デフォルトで /var/www/cgi-bin/ に設定されます。

cgi-bin/ ディレクトリー外に実行ファイルを保存するディレクトリーを確立できます。手順は、AddHandler および ディレクトリー を参照してください。

ServerAdmin

ServerAdmin ディレクティブを Web サーバー管理者のメールアドレスに設定します。このメールアドレスは、サーバーが生成した Web ページのエラーメッセージを表示するため、ユーザーはサーバー管理者にメールを送信して問題を報告できます。

デフォルトでは、ServerAdmin は root@localhost に設定されます。

ServerAdmin を設定する一般的な方法は、webmaster@example.com に設定することです。設定が完了したら、エイリアス webmaster を /etc/aliases で Web サーバーを担当する人に実行し、/usr/bin/newaliases を実行します。

ServerName

ServerName は、サーバーのホスト名およびポート番号(Listen ディレクティブと一致する)を指定します。ServerName はマシンの実際のホスト名に一致する必要はありません。たとえば、Web サーバーは www.example.com ですが、サーバーのホスト名は実際には foo.example.com になります。ServerName で指定する値は、システムで解決できる有効な Domain Name Service (DNS)名である必要があります。何も作成しないでください。

以下は ServerName ディレクティブの例です。

```
ServerName www.example.com:80
```

ServerName を指定する場合は、IP アドレスとサーバー名のペアが /etc/hosts ファイルに含まれていることを確認してください。

ServerRoot

ServerRoot ディレクティブは、Web サイトコンテンツを含む最上位のディレクトリーを指定します。デフォルトでは、ServerRoot はセキュア なサーバーと非セキュアサーバーの両方で /etc/httpd に設定されています。

ServerSignature

ServerSignature ディレクティブは、Apache HTTP Server サーバーバージョンと ServerName を含む行を、クライアントに送信されたエラーメッセージなどのサーバー生成ドキュメントに追加します。ServerSignature はデフォルトで on に設定されます。

ServerSignature を EMail に設定すると、自動生成される応答の署名行に mailto:ServerAdmin HTML タグが追加されます。ServerSignature を Off に設定すると、Apache がバージョン番号とモジュール情報を送信しないようにすることもできます。ServerTokens 設定も確認してください。

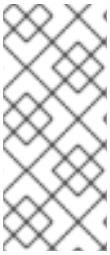
ServerTokens

ServerTokens ディレクティブは、クライアントに返信する Server 応答ヘッダーフィールドに、オペレーティングシステムのタイプおよびコンパイル済みモジュールの詳細が含まれるべきかどうかを決定します。デフォルトでは、ServerTokens は Full に設定され、オペレーティングシステムの種類とコンパイル済みモジュールに関する情報を送信します。ServerTokens を Prod に設定すると製品名のみが送信され、脆弱性をスキャンする際にサーバーヘッダーのハッカー確認情報がいくつでも推奨されます。ServerTokens を Min （最小）または OS （オペレーティングシステム）に設定することもできます。

SuexecUserGroup

mod_suexec モジュールから発信される SuexecUserGroup ディレクティブは、CGI プログラムに対してユーザーおよびグループの実行権限を指定できるようにします。CGI 以外の要求は、User

ディレクティブおよび Group ディレクティブで指定されたユーザーおよびグループで処理されます。



注記

バージョン 2.0 以降、SuexecUserGroup ディレクティブは、VirtualHosts セクションの設定内で User ディレクティブおよび Group ディレクティブを使用する Apache HTTP Server 1.3 設定に置き換わりました。

Timeout

timeout は、サーバーが通信中に受信および送信を待つ時間を秒単位で定義します。タイムアウトはデフォルトで 300 秒に設定されています。これはほとんどの状況に適しています。

TypesConfig

TypesConfig は、MIME タイプマッピングのデフォルト一覧（コンテンツタイプへのファイル名拡張）を設定するファイルに名前を付けます。デフォルトの TypesConfig ファイルは /etc/mime.types です。/etc/mime.types を編集する代わりに、MIME タイプマッピングを追加する方法として、AddType ディレクティブを使用することが推奨されます。

AddType の詳細は、AddType を参照してください。

UseCanonicalName

で に設定すると、このディレクティブは ServerName ディレクティブおよび Port ディレクティブで指定された値を使用して Apache HTTP Server がそれ自体を参照するように設定します。UseCanonicalName が off に設定されている場合、サーバーは代わりに、それ自体を参照する際に要求元のクライアントによって使用される値を使用します。

UseCanonicalName はデフォルトで off に設定されています。

User

User ディレクティブは、サーバープロセスのユーザー名を設定し、サーバーがアクセスできるファイルを決定します。このユーザーがアクセスできないファイルは、Apache HTTP Server に接続するクライアントにもアクセスできません。

デフォルトでは、User は apache に設定されています。

このディレクティブは、仮想ホストの設定で非推奨になりました。

注記

セキュリティー上の理由から、Apache HTTP Server は root ユーザーとしては実行されません。

UserDir

UserDir は、Web サーバーによって提供される個人 HTML ファイルを配置する各ユーザーのホームディレクトリー内のサブディレクトリーです。このディレクティブは、デフォルトで 無効 にするように設定されています。

サブディレクトリーの名前は、デフォルト設定で public_html に設定されます。たとえば、サーバーは以下のリクエストを受け取る場合があります。

```
http://example.com/~username/foo.html
```

サーバーは ファイルを検索します。

```
/home/username/public_html/foo.html
```

上記の例では、/home/username/ がユーザーのホームディレクトリーです（ユーザーのホームディレクトリーのデフォルトパスは異なる場合があることに注意してください）。

ユーザーのホームディレクトリーのパーミッションが正しく設定されていることを確認してください。ユーザーのホームディレクトリーを 0711 に設定する必要があります。読み取り(r)および実行(x)ビットは、ユーザーの public_html ディレクトリー(0755 も機能)に設定する必要があります。ユーザーの public_html ディレクトリーで提供されるファイルは、少なくとも 0644 に設定する必要があります。

VirtualHost

<VirtualHost &gt ; および </VirtualHost > タグは、仮想ホストの特性を示すコンテナーを作成します。VirtualHost コンテナーはほとんどの設定ディレクティブを受け入れます。

コメント付き VirtualHost コンテナーは httpd.conf で提供され、各仮想ホストに必要な最低限の設定ディレクティブのセットを示しています。仮想ホストの詳細は、「仮想ホスト」 を参照してくださ

い。

注記

デフォルトの SSL 仮想ホストコンテナーは、ファイル /etc/httpd/conf.d/ssl.conf に置かれるようになりました。

25.5.2. SSL の設定ディレクティブ

/etc/httpd/conf.d/ssl.conf ファイルのディレクティブは、TLS を使用したセキュアな Web 通信を有効にするように設定できます。TLS の有効化中 『の SSL の無効化に関する重要な情報は、httpd の POODLE SSLv3.0 脆弱性(CVE-2014-3566)の解決』 を参照してください。

重要

『POODLE: SSLv3 脆弱性(CVE-2014-3566)で説明されている脆弱性』 により、Red Hat は SSL を無効にし、TLSv1. 1 または TLSv1.2 のみを使用することを推奨します。後方互換性は、TLSv1.0 を使用して実現できます。Red Hat がサポートする多くの製品は SSLv2 プロトコルまたは SSLv3 プロトコルを使用するか、デフォルトでそれらのプロトコルを有効にできます。ただし、SSLv2 または SSLv3 を使用することが強く推奨されます。

SetEnvIf

SetEnvIf は、受信接続のヘッダーに基づいて環境変数を設定します。これは SSL ディレクティブのみではなく、提供された /etc/httpd/conf.d/ssl.conf ファイルにあります。このコンテキストの目的は、HTTP keepalive を無効にし、クライアントブラウザーから終了通知なしに SSL が接続を閉じることを許可することです。この設定は、SSL 接続を確実にシャットダウンしない特定のブラウザーに必要です。

SSL 設定ファイル内の他のディレクティブに関する詳細は、以下の URL を参照してください。

- http://localhost/manual/mod/mod_ssl.html
- http://httpd.apache.org/docs/2.2/mod/mod_ssl.html

注記

ほとんどの場合、Red Hat Enterprise Linux のインストール時に SSL ディレクティブを適切に設定します。Apache HTTP Secure Server ディレクティブを変更する場合は注意してください。設定が間違っていると、セキュリティー脆弱性が発生する可能性があります。

25.5.3. MPM 固有のサーバープールディレクティブ

「Server-Pool Size Regulation」 で説明されているように、server-pool の特性を管理する責任は、Apache HTTP Server 2.0 の MPMs と呼ばれるモジュールグループに分類されます。server-pool の特性は、使用される MPM によって異なります。このため、使用中の MPM の server-pool を定義するのに IfModule コンテナーが必要です。

デフォルトでは、Apache HTTP Server 2.0 は、prefork と worker MPM の両方の server-pool を定義します。

次のセクションでは、MPM 固有のサーバープールコンテナー内にあるディレクティブを一覧表示します。

MaxClients

MaxClients は、一度に実行できるサーバープロセスの合計数または同時接続クライアントに制限を設定します。このディレクティブの主な目的は、Apache HTTP Server がオペレーティングシステムをクラッシュさせないようにすることです。ビジーなサーバーの場合、この値は高い値に設定する必要があります。サーバーのデフォルトは、使用中の MPM に関係なく 150 に設定されます。ただし、prefork MPM を使用する際に MaxClients の値が 256 を超えることは推奨されません。

MaxRequestsPerChild

MaxRequestsPerChild は、子終了の前に各子サーバープロセスが提供する要求の合計数を設定します。MaxRequestsPerChild を設定する主な理由は、有効期限の長いプロセスがメモリーリークを引き起こすことを回避するためです。prefork MPM のデフォルトの MaxRequestsPerChild は 4000 で、worker MPM の場合は 0 です。

MinSpareServers および MaxSpareServers

これらの値は、prefork MPM でのみ使用されます。Apache HTTP Server は、受信要求の数に基づいて適切な数のスペアサーバープロセスを維持することで、認識された負荷に動的に適応する方法を調整します。サーバーは要求を待機しているサーバーの数を確認し、MaxSpareServers を超える場合に を強制終了するか、サーバーの数が MinSpareServers 未満の場合は を作成します。

デフォルトの MinSpareServers 値は 5 です。デフォルトの MaxSpareServers 値は 20 です。これ

らのデフォルト設定は、ほとんどの状況に適しています。MinSpareServers を大きな数値に増やさないように注意してください。これは、トラフィックが軽量であってもサーバーに大量の処理負荷が発生するためです。

MinSpareThreads および MaxSpareThreads

これらの値は worker MPM でのみ使用されます。これらは、受信要求の数に基づいて適切な数のスペアサーバースレッドを維持することで、Apache HTTP Server が認識された負荷に動的に適応する方法を調整します。サーバーはリクエストを待機しているサーバースレッドの数を確認し、MaxSpareThreads を超える場合は一部を強制終了するか、サーバーの数が MinSpareThreads 未満の場合は作成します。

デフォルトの MinSpareThreads 値は 25 です。デフォルトの MaxSpareThreads 値は 75 です。これらのデフォルト設定は、ほとんどの状況に適しています。MaxSpareThreads の値は、MinSpareThreads および ThreadsPerChild の合計以上である必要があります。そうでない場合、Apache HTTP Server は自動的に修正します。

StartServers

StartServers ディレクティブは、起動時に作成されるサーバープロセスの数を設定します。Web サーバーは、トラフィックの負荷に基づいてサーバープロセスを動的に強制終了して作成するため、このパラメーターを変更する必要はありません。Web サーバーは、worker MPM の prefork MPM と 2 の起動時に 8 サーバープロセスを開始するように設定されています。

ThreadsPerChild

この値は、worker MPM でのみ使用されます。各子プロセス内のスレッド数を設定します。このディレクティブのデフォルト値は 25 です。

25.6. モジュールの追加

Apache HTTP Server には、多くのモジュールが同梱されています。Apache HTTP モジュールの詳細は、http://httpd.apache.org/docs/2.2/mod/ を参照してください。

Apache HTTP Server は、必要に応じてランタイム時に簡単にロードできる 動的共有オブジェクト (DSO)または モジュールをサポートします。

Apache Project は、http://httpd.apache.org/docs/2.2/dso.html で完全な DSO ドキュメントをオンラインに提供します。または、http-manual パッケージがインストールされている場合は、DSO に関するドキュメンテーションは http://localhost/manual/mod/ からオンラインで参照できます。

Apache HTTP Server が DSO を使用するには、/etc/httpd/conf/httpd.conf 内の LoadModule ディ

レクティブで指定する必要があります。モジュールが別のパッケージにより提供されている場合は、/etc/httpd/conf.d/ ディレクトリーのモジュール設定ファイル内に 行が表示されるはずです。詳細は、LoadModule を参照してください。

http.conf からモジュールを追加または削除する場合は、「httpdの起動と停止」 にあるように、Apache HTTP Server をリロードまたは再起動する必要があります。

新しいモジュールを作成する場合は、最初に、include ファイル、ヘッダーファイル、および APache eXtenSion (/usr/sbin/apxs)アプリケーションを含む httpd-devel パッケージをインストールします。このアプリケーションは、include ファイルおよびヘッダーファイルを使用して DSO をコンパイルします。

モジュールを作成したら、/usr/sbin/apxs を使用して、Apache ソースツリー外でモジュールソースをコンパイルします。/usr/sbin/apxs コマンドの使用方法は、オンラインの Apache ドキュメントの http://httpd.apache.org/docs/2.2/dso.html および apxs の man ページを参照してください。

コンパイルしたら、モジュールを /usr/lib/httpd/modules/ ディレクトリーに配置します。default-64 ビットユーザー空間(x86_64、ia64、?)を使用する RHEL プラットフォームの場合、このパスは /usr/lib64/httpd/modules/ になります。次に、以下の構造を使用して LoadModule 行を httpd.conf に追加します。

```
LoadModule <module-name> <path/to/module.so>
```

ここで、<module-name > はモジュールの名前で、< path/to/module.so > は DSO へのパスです。

25.7. 仮想ホスト

Apache HTTP Server の組み込み仮想ホストにより、サーバーはどの IP アドレス、ホスト名、またはポートが要求されるかに基づいて異なる情報を提供できます。仮想ホストの使用に関する完全なガイドは、http://httpd.apache.org/docs/2.2/vhosts/ でオンラインで利用できます。

25.7.1. 仮想ホストの設定

名前ベースの仮想ホストを作成するには、例として httpd.conf で提供される仮想ホストコンテナーを使用することが推奨されます。

仮想ホストの例は次のようになります。

```
#NameVirtualHost *:80
#
#<VirtualHost *:80>
# ServerAdmin webmaster@dummy-host.example.com
# DocumentRoot /www/docs/dummy-host.example.com
# ServerName dummy-host.example.com
# ErrorLog logs/dummy-host.example.com-error_log
# CustomLog logs/dummy-host.example.com-access_log common #</VirtualHost>
```

名前ベースの仮想ホストを有効にするには、ハッシュマーク(#)を削除して NameVirtualHost 行のコメントを解除し、アスタリスク(*)をマシンに割り当てられた IP アドレスに置き換えます。

次に、< VirtualHost > コンテナーのコメントを解除してカスタマイズして仮想ホストを設定します。

&lt ;VirtualHost> 行で、アスタリスク(*)をサーバーの IP アドレスに変更します。ServerName をマシンに割り当てられた 有効な DNS 名に切り替え、必要に応じて他のディレクティブを設定します。

&lt ;VirtualHost> コンテナーは高度なカスタマイズが可能で、メインサーバー設定で利用可能なほぼすべてのディレクティブを受け入れます。

ヒント

デフォルト以外のポートでリッスンするように仮想ホストを設定する場合は、そのポートを /etc/httpd/conf/httpd.conf ファイルのグローバル設定セクションの Listen ディレクティブに追加する必要があります。

新しく作成された仮想ホストを有効にするには、Apache HTTP Server をリロードまたは再起動する必要があります。詳細は、「httpdの起動と停止」 を参照してください。

名前ベースおよび IP アドレスベースの仮想ホストの両方の作成および設定に関する包括的な情報は、http://httpd.apache.org/docs/2.2/vhosts/ でオンラインで提供されます。

25.8. APACHE HTTP セキュアサーバー設定

本セクションでは、OpenSSL ライブラリーおよびツールキットを使用できるように mod_ssl セキュリティーモジュールが有効になっている Apache HTTP Server の基本情報を提供します。これら

の 3 つのコンポーネントの組み合わせは、このセクションではセキュアな Web サーバーまたはセキュアなサーバーとして参照されます。

mod_ssl モジュールは、Apache HTTP Server のセキュリティーモジュールです。mod_ssl モジュールは OpenSSL プロジェクトが提供するツールを使用して Apache HTTP Server に非常に重要な機能を追加します。これは通信を暗号化する機能です。一方、ブラウザーと Web サーバー間の通常の HTTP 通信はプレーンテキストで送信されます。

このセクションは、これらプログラムのドキュメントをすべて完了せず、ドキュメントは含まれません。このガイドは、可能であれば、特定のサブジェクトに関する詳細のドキュメントを見つけることができる適切な場所を指しています。

本セクションでは、これらのプログラムをインストールする方法を説明します。秘密鍵と証明書要求の生成に必要な手順、独自の自己署名証明書の生成方法、およびセキュアなサーバーで使用する証明書のインストール方法についても確認できます。

mod_ssl 設定ファイルは /etc/httpd/conf.d/ssl.conf にあります。このファイルを読み込むには、mod_ssl が機能するには、/etc/httpd/conf/httpd .conf ファイルに Include conf.d/*.conf というステートメントが必要です。このステートメントは、デフォルトでデフォルトの Apache HTTP Server 設定ファイルに含まれています。

25.8.1. セキュリティー関連パッケージの概要

セキュアなサーバーを有効にするには、少なくとも以下のパッケージがインストールされている必要があります。

httpd

httpd パッケージには、httpd デーモンおよび関連ユーティリティー、設定ファイル、アイコン、Apache HTTP Server モジュール、man ページ、Apache HTTP Server が使用するその他のファイルが含まれます。

mod_ssl

mod_ssl パッケージには mod_ssl モジュールが含まれており、Secure Sockets Layer (SSL) プロトコルおよび Transport Layer Security (TLS)プロトコルを介して Apache HTTP Server に対して強力な暗号化を提供します。

openssl

openssl パッケージには OpenSSL ツールキットが含まれています。OpenSSL ツールキットは SSL プロトコルおよび TLS プロトコルを実装し、汎用暗号化ライブラリーも含まれます。

さらに、他のソフトウェアパッケージは、特定のセキュリティー機能を提供します（ただし、セキュアなサーバーで機能する必要はありません）。

25.8.2. 証明書およびセキュリティーの概要

Secure Sockets Layer (SSL)プロトコルと（ほとんどの場合）認証局(CA)からのデジタル証明書の組み合わせを使用してセキュリティーを提供します。SSL は、暗号化された通信とブラウザーとセキュアなサーバー間の相互認証を処理します。CA が承認したデジタル証明書は、セキュアなサーバーに認証を提供します(CA は組織の ID の認定の背後で、その再構築を後継します)。ブラウザーが SSL 暗号化を使用して通信する場合は、ナビゲーションバーの URL (Uniform Resource Locator)の先頭に https:// 接頭辞が使用されます。

暗号化は、キーの使用によって異なります（データ形式でシークレットエンコーダー/デコーダーリングとして解釈されます）。従来の暗号または対称暗号では、トランザクションの両端に同じキーがあり、これを使用して相互の送信をデコードします。公開鍵または非対称暗号化では、公開鍵と秘密鍵の 2 つの鍵が共存します。個人または組織は秘密鍵を秘密にし、公開鍵を公開します。公開鍵でエンコードされたデータは、秘密鍵でのみデコードできます。秘密鍵でエンコードされたデータは、公開鍵でのみデコードできます。

セキュアなサーバーを設定するには、パブリック暗号を使用して公開鍵と秘密鍵のペアを作成します。ほとんどの場合、証明書要求（公開鍵を含む）、会社の ID の証明、および CA に支払いを行います。CA は、証明書要求と ID を検証し、安全なサーバーの証明書を返します。

セキュアなサーバーは、証明書を使用して Web ブラウザーに自己識別します。独自の証明書（自己署名証明書と呼ばれる）を生成するか、または CA から証明書を取得できます。信頼できる CA からの証明書は、Web サイトが特定の会社または組織に関連付けられていることを保証します。

または、独自の自己署名証明書を作成できます。ただし、自己署名証明書は、ほとんどの実稼働環境では使用しないでください。ユーザーのブラウザーでは、自己署名証明書は自動的に許可されません。ユーザーには、ブラウザーが証明書を受け入れてセキュアな接続を作成するように要求されます。自己署名証明書と CA 署名の証明書の違いに関する詳細は、「証明書の種類」 を参照してください。

選択した CA からの自己署名証明書または署名済み証明書を追加したら、セキュアなサーバーにインストールする必要があります。

25.8.3. 既存のキーおよび証明書の使用

既存の鍵と証明書がすでにある場合（たとえば、セキュアなサーバーをインストールして別の会社のセキュアなサーバー製品を置き換える場合など）、既存の鍵と証明書を安全なサーバーで使用する可能性があります。以下の 2 つの状況では、既存のキーと証明書を使用できないインスタンスが提供されます。

- IP アドレスまたはドメイン名を変更する場合：特定の IP アドレスとドメイン名 のペアに対して証明書が発行されます。IP アドレスまたはドメイン名を変更する場合は、新しい証明書を取得する必要があります。

- VeriSign からの証明書があり、サーバーソフトウェアを変更した場合 は、VeriSign が広く使用されている CA です。別の目的で VeriSign 証明書がすでにある場合は、既存の VeriSign 証明書を新しいセキュアなサーバーで使用することを検討している可能性があります。ただし、VeriSign は 1 つの特定のサーバーソフトウェアと IP アドレス/ドメイン名の組み合わせの証明書を発行するため、は許可されません。

  これらのパラメーターのいずれかを変更する場合（たとえば、以前に別のセキュアなサーバー製品を使用している場合など）、以前の設定で使用するために取得した VeriSign 証明書は、新しい設定では機能しません。新しい証明書を取得する必要があります。

使用できる既存の鍵と証明書がある場合は、新しい鍵を生成し、新しい証明書を取得する必要はありません。ただし、鍵と証明書を含むファイルを移動して名前を変更する必要がある場合があります。

既存のキーファイルを移動します。

```
/etc/pki/tls/private/server.key
```

既存の証明書ファイルを以下に移動します。

```
/etc/pki/tls/certs/server.crt
```

Red Hat Secure Web Server からアップグレードする場合は、古いキー(httpsd.key)および証明書(httpsd.crt)は /etc/httpd/conf/ にあります。セキュアなサーバーが使用できるように、鍵と証明書を移動して名前を変更します。以下の 2 つのコマンドを使用して、鍵と証明書ファイルを移動して名前を変更します。

```
mv /etc/httpd/conf/httpsd.key /etc/pki/tls/private/server.key
mv /etc/httpd/conf/httpsd.crt /etc/pki/tls/certs/server.crt
```

次に、コマンドを使用してセキュアなサーバーを起動します。

```
service httpd start
```

25.8.4. 証明書の種類

Red Hat が提供する RPM パッケージからセキュアなサーバーをインストールした場合は、ランダムに生成された秘密鍵とテスト証明書が生成され、適切なディレクトリーに配置されます。ただし、セキュアなサーバーの使用を開始する前に、独自の鍵を生成し、サーバーを正しく識別する証明書を取得する必要があります。

安全なサーバーを操作するための鍵と証明書が必要です。つまり、自己署名証明書を生成するか、CA から CA 署名の証明書を購入することができます。この 2 つの相違点は何ですか？

CA 署名の証明書は、サーバーに 2 つの重要な機能を提供します。

- ブラウザー（通常は）は証明書を自動的に認識し、ユーザーに要求せずにセキュアな接続を可能にします。
- CA が署名済み証明書を発行すると、ブラウザーに Web ページを提供している組織のアイデンティティーが保証されます。

セキュアなサーバーが大規模に公開されている場合、セキュアなサーバーには CA が署名した証明書が必要です。このため、Web サイトにアクセスするユーザーは、Web サイトの所有者を主張する組織によって所有されていることを認識できます。証明書を署名する前に、CA は、証明書を要求する組織が実際に要求したユーザーであることを確認します。

SSL をサポートするほとんどの Web ブラウザーには、証明書が自動的に受け入れる CA のリストがあります。ブラウザーが CA の承認が一覧にない証明書に遭遇すると、ブラウザーはユーザーに対して接続を許可または拒否するように要求します。

セキュアなサーバーの自己署名証明書を生成できますが、自己署名証明書は CA 署名の証明書と同じ機能を提供しないことに注意してください。自己署名証明書は、ほとんどの Web ブラウザーでは自動的に認識されず、Web サイトを提供している組織のアイデンティティーに関する保証はありません。CA 署名の証明書は、セキュアなサーバーにこれらの重要な機能を提供します。セキュアなサーバーを実稼働環境で使用する場合は、CA 署名の証明書が推奨されます。

CA から証明書を取得するプロセスはかなり簡単です。簡単な概要は以下のとおりです。

1. 暗号化秘密鍵と公開鍵のペアを作成します。

2. 公開鍵に基づいて証明書要求を作成します。証明書要求には、サーバーとそのサーバーをホストする会社に関する情報が含まれます。

3. アイデンティティーの承認ドキュメントとともに、証明書要求を CA に送信します。Red Hat は、どの認証局が選択するかについての推奨事項を行いません。決定は、過去の経験、フォンドア語、または通貨的要因に基づく場合があります。

   CA を決定したら、CA から証明書を取得する方法に関する指示に従う必要があります。

4. CA が満たされると、実際に依頼されると、デジタル証明書が提供されます。

5. この証明書を安全なサーバーにインストールし、セキュアなトランザクションの処理を開始します。

CA から証明書を取得するか、または独自の自己署名証明書を生成する場合でも、最初にキーを生成することです。手順は、「キーの生成」 を参照してください。

25.8.5. キーの生成

キーを生成するには、root である必要があります。

まず、cd コマンドを使用して /etc/pki/tls/ ディレクトリーに変更します。以下のコマンドを使用して、インストール時に生成された鍵および証明書を削除します。

```
rm private/server.key
rm certs/server.crt
```

crypto-utils パッケージには、名前が示すように鍵を生成するために使用できる genkey ユーティリティーが含まれています。独自の秘密鍵を作成するには、crypto-utils 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。ターミナルで man genkey と入力して、より多くのオプションを

表示できます。genkey ユーティリティーを使用して www.example.com のキーを生成する場合は、ターミナルで以下のコマンドを入力します。

```
genkey www.example.com
```

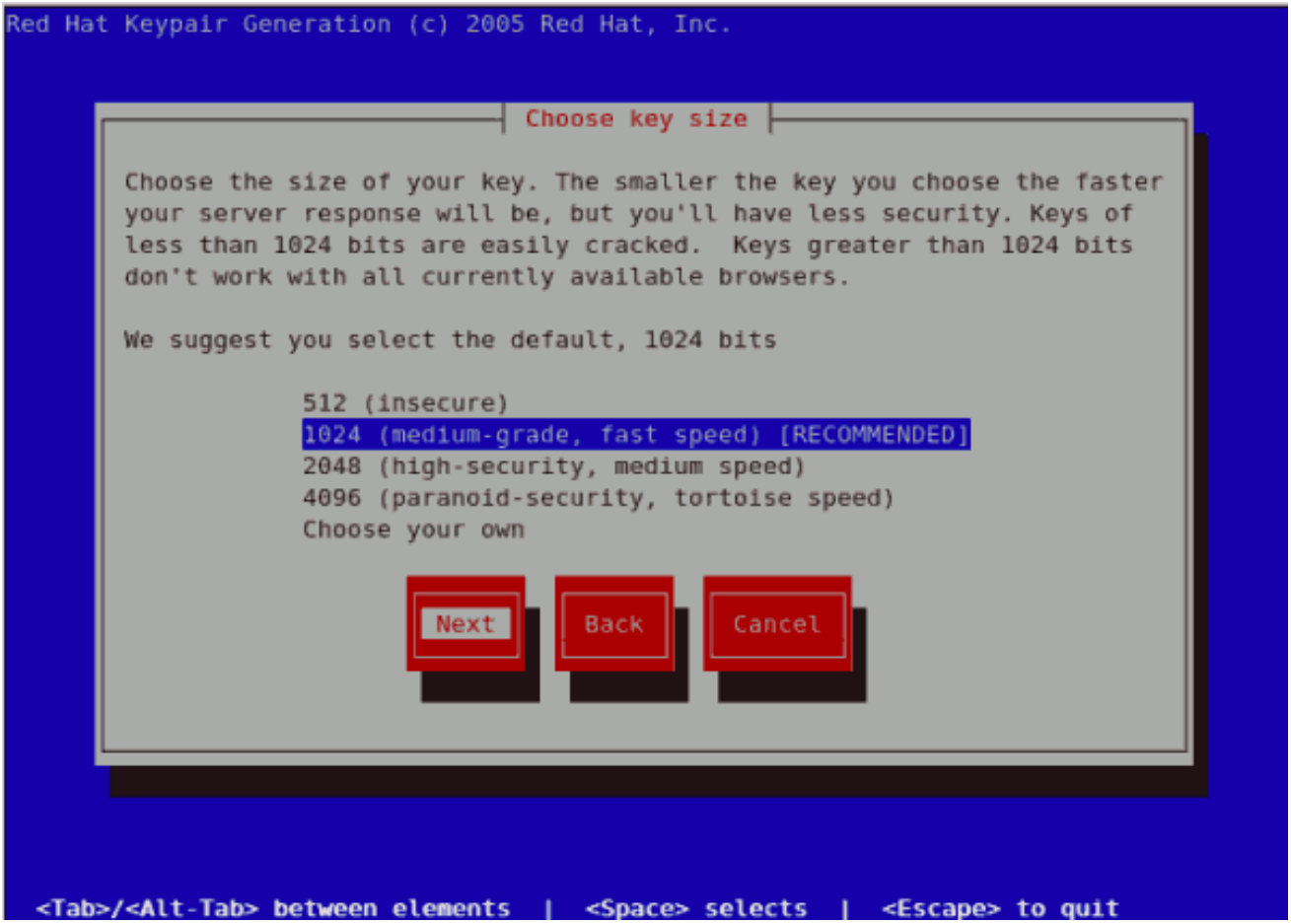
make ベースのプロセスは RHEL 5 に同梱されなくなった点に注意してください。これにより、genkey グラフィカルユーザーインターフェイスが起動します。以下の図は最初の画面を示しています。移動するには、キーボードの矢印とタブキーを使用します。このウィンドウは、キーの保存先を示し、操作を続行または取り消すプロンプトを表示します。次のステップに進むには、Next を選択して Return (Enter)キーを押します。

図25.11 キーペアの生成

[D]

次の画面では、キーのサイズを選択するように求められます。示されているように、キーのサイズが小さくなると、サーバーからの応答が速くなり、セキュリティーレベルが低くなります。矢印キーを使用して任意のキーサイズを選択し、Next を選択して次のステップに進みます。以下の図は、キーサイズ選択画面を示しています。

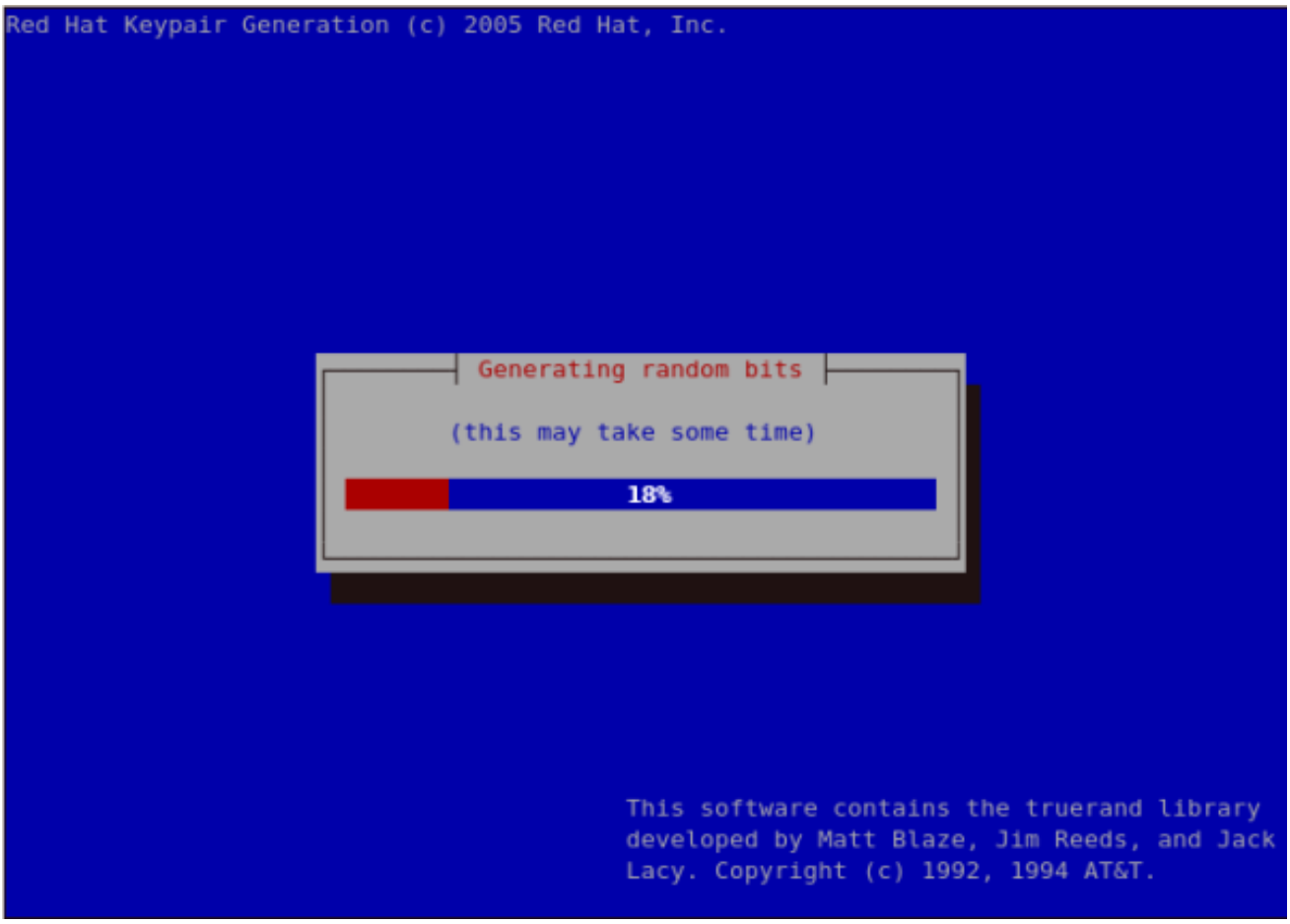
図25.12 キーサイズの選択



[D]

次のステップを選択すると、ランダムなビット生成プロセスが開始します。これには、選択したキーのサイズによっては時間がかかる場合があります。キーのサイズが大きいほど、生成にかかる時間が長くなります。

図25.13 ランダムなビットの生成



[D]

キーを生成する際に、証明書要求(CSR)を認証局(CA)に送信するように求められます。

図25.14 Generate CSR

[D]

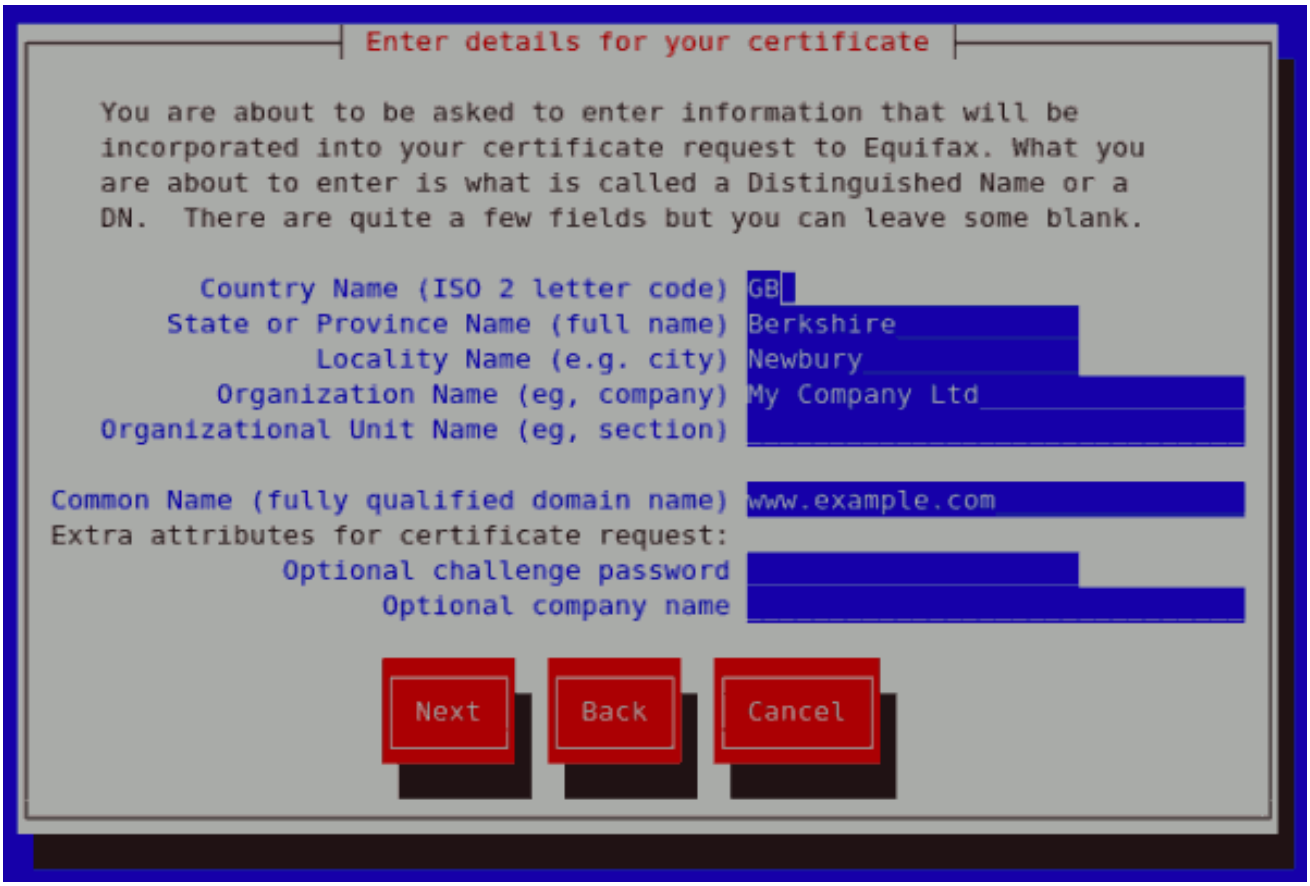
Yes を選択すると、リクエストの送信先となる認証局を選択するように求められます。No を選択すると、自己署名証明書を生成できます。この次のステップは、図25.17「サーバーの自己署名証明書の生成」 で説明されています。

図25.15 認証局(CA)を選択します。

[D]

Selecting your preferred option で Next を選択し、次のステップに進みます。次の画面では、証明書の詳細を入力できます。

図25.16 証明書の詳細を入力します。



[D]

自己署名証明書キーペアを生成する場合は、CSR を生成しないでください。これを行うには、Generate CSR 画面で、優先オプションとして No を選択します。これにより、以下の図が表示されます。この図から、証明書の詳細を入力できます。証明書の詳細を入力し、戻り値キーを押すと 図25.19「秘密鍵の保護」 が表示されます。そこから、秘密鍵の暗号化を選択できます。

図25.17 サーバーの自己署名証明書の生成

[D]

*証明書の詳細を入力し、**Next** を選択して続行します。以下の図は、**Equifax** に送信される証明書の詳細を完了した後に表示される次の画面の例です。サーバーの自己署名鍵を生成する場合は、この画面が表示されないことに注意してください。*

図25.18 証明書要求の開始

```
You now need to submit your CSR and documentation to your certificate
authority. Submitting your CSR may involve pasting it into an online
web form, or mailing it to a specific address. In either case, you
should include the BEGIN and END lines.

-----BEGIN CERTIFICATE REQUEST-----
MIIBTjCB+QIBADBmMQswCQYDVQQGEwJHQjESMBAGA1UECBMJQmVya3NoaXJlMRAw
DgYDVQQHEwdOZXdidXJ5MRcwFQYDVQQKEw5NeSBDb21wYW55IEx0ZDEYMBYGA1UE
AxMPd3d3LmV4YW1wbGUuY29tMFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAMbY0dqO
YlXsmstZ7L7C27TX7lyBQO7jayOc7mShlXemItJHoEjcSTge51G5EIm5sm5+5vNU
6NEkBNnW0aAoa4MCAwEAAaAuMBUGCSqGSIb3DQEJAjEIEwZyZWRoYXQwFQYJKoZI
hvcNAQkHMQgTBnJlZGhhdDANBgkqhkiG9w0BAQUFAANBAK1iOocPMET2Yy3t4ffb
uIERHGn6w0RhriJtCxkJBDGbwTXKUXYw0iWWX5WQpcwnn0LYTXj8X1c4KX29N5gm
LVs=
-----END CERTIFICATE REQUEST-----


A copy of this CSR has been saved in the file
/etc/pki/tls/certs/www.example.com.2.csr

Press return when ready to continue
```

[D]

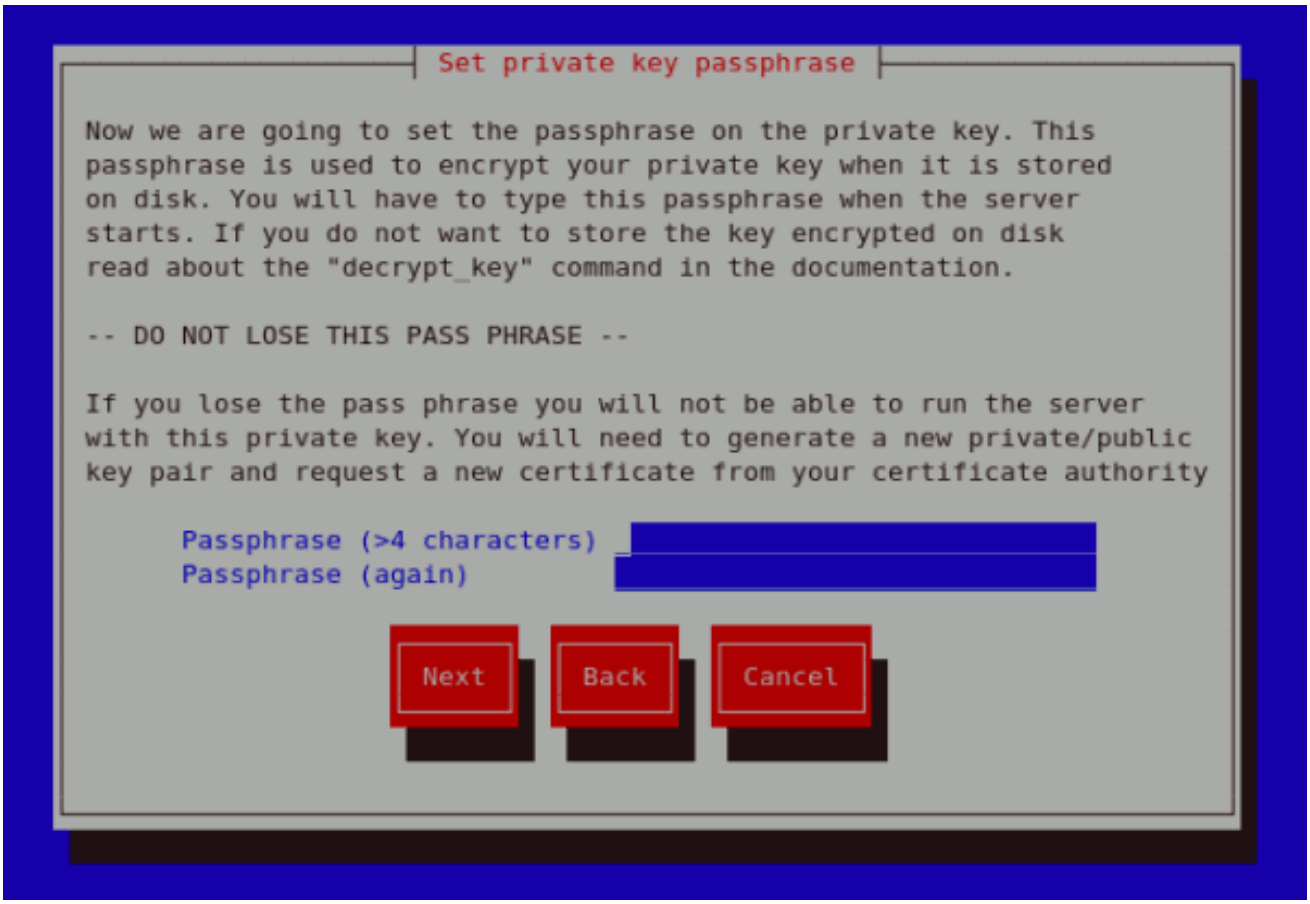
*戻り値のキーを押すと、秘密鍵の暗号化を有効または無効にする次の画面が表示されます。スペースバーを使用して、これを有効または無効にします。有効にすると、[*] 文字が表示されます。希望するオプションを選択する場合は、**Next** を選択して次のステップに進みます。*

図25.19 秘密鍵の保護

[D]

次の画面では、鍵パスフレーズを設定できます。このパスフレーズは、サーバーなしでは実行できないため、失われないでください。以下に示すように、新しい秘密鍵または公開鍵のペアを再生成し、CA から新しい証明書を要求する必要があります。セキュリティーを確保するため、パスフレーズは入力時に表示されません。希望するパスフレーズを入力し、次へ を選択してターミナルに戻ります。

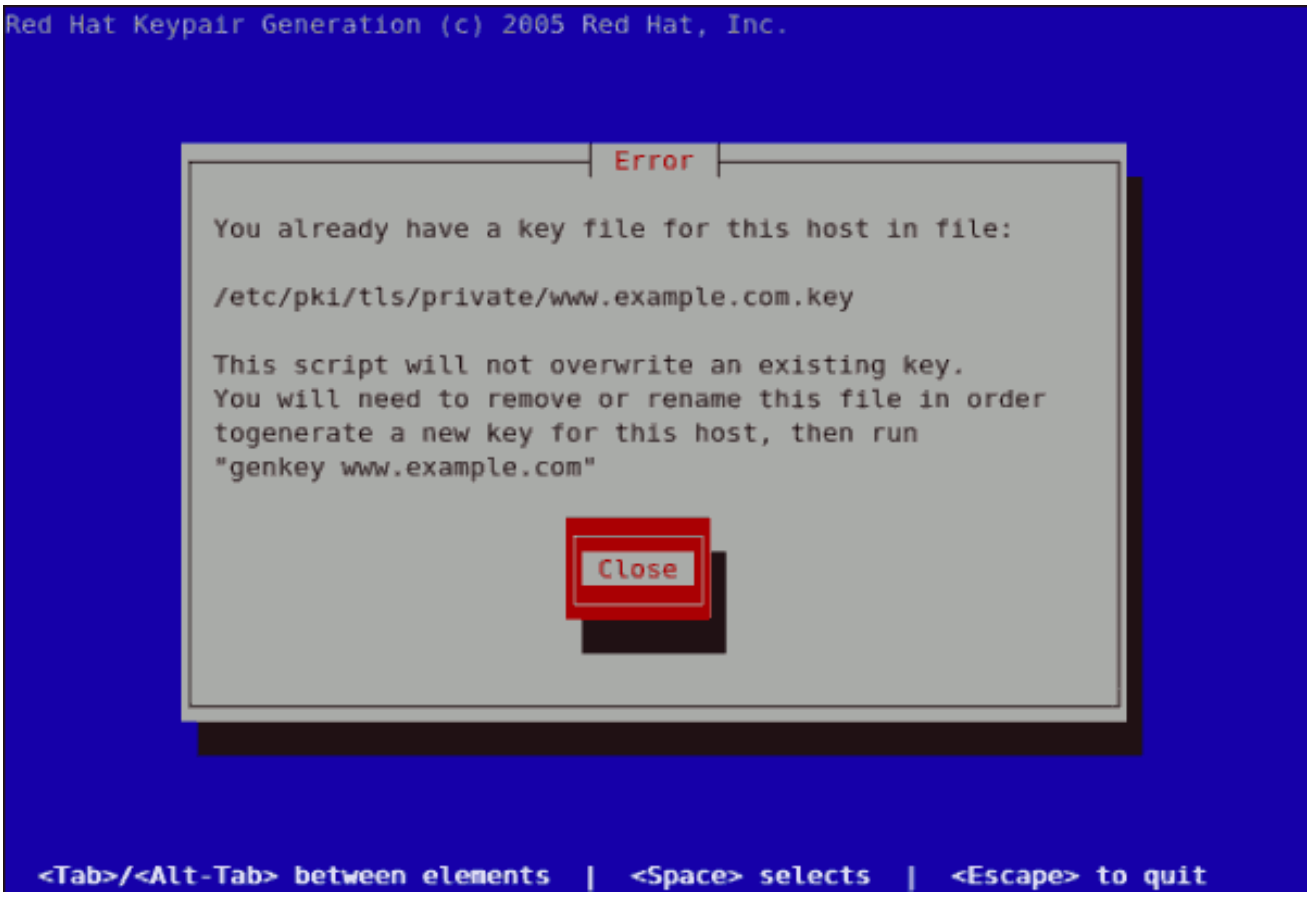
図25.20 パスフレーズの設定



[D]

特定のホスト名に対して既存のキーペアがあるサーバーで genkey www.example.com を実行しようとすると、以下のようにエラーメッセージが表示されます。新しいキーペアを生成するように、既存のキーファイルを削除する必要があります。

図25.21 genkey エラー



[D]

- *http://httpd.apache.org/docs/2.2/ssl/*
- *http://httpd.apache.org/docs/2.2/vhosts/*

25.8.6. 新しいキーを使用するようにサーバーを設定する方法

新しいキーを使用するように Apache HTTP Server を設定する手順は次のとおりです。

- *CSR の送信後に CA から署名済み証明書を取得します。*
- *証明書をパスにコピーします（例： /etc/pki/tls/certs/www.example.com.crt）。*
- */etc/httpd/conf.d/ssl.conf を編集します。SSLCertificateFile 行と SSLCertificateKey 行を に変更します。*

```
SSLCertificateFile /etc/pki/tls/certs/www.example.com.crt
SSLCertificateKeyFile /etc/pki/tls/private/www.example.com.key
```

「www.example.com」 の部分は、genkey コマンドで渡された引数と一致する必要があることに注意してください。

25.9. 関連情報

Apache HTTP Server の詳細は、以下のリソースを参照してください。

25.9.1. 便利な Web サイト

- http://httpd.apache.org/ - Apache HTTP Server の公式 Web サイトです。すべてのディレクティブおよびデフォルトモジュールのドキュメントを参照してください。
- http://www.modssl.org/ - mod_ssl の公式 Web サイトです。
- http://www.apacheweek.com/: Apache に関する包括的なオンライン週ごとの包括的な情報です。

第26章 FTP

ファイル転送プロトコル (FTP) は、今日インターネット上で見られる、最も古く、一般的に使用されているプロトコルです。この目的は、ユーザーがリモートホストに直接ログインしなくても、もしくはリモートシステムの使用法についての知識がなくとも、ネットワーク上のコンピューターホスト間で確実にファイルを転送することです。これにより、ユーザーは、標準の簡単なコマンドセットを使用してリモートシステム上のファイルにアクセスすることができます。

本章では、FTP プロトコルの基本と、Red Hat Enterprise Linux に同梱されるプライマリー FTP サーバーの設定オプションである vsftpd を概説します。

26.1. ファイル転送プロトコル (FTP)

FTP は、クライアントのサーバーアーキテクチャーを使用して、TCP ネットワークプロトコルを使用してファイルを転送します。FTP は古いプロトコルであるため、暗号化されていないユーザー名とパスワード認証を使用します。このため、安全でないプロトコルとみなされており、絶対的に必要でない限り、使用するべきではありません。FTP の代わりとなるのは、OpenSSH スイートからの sftp です。OpenSSH の設定に関する詳細は、20章OpenSSH を参照してください。

ただし、FTP はインターネット上で非常に評価されているため、ファイルをパブリックに共有する必要があることがよくあります。したがって、システム管理者は FTP プロトコルの一意の特性を認識しておく必要があります。

26.1.1. 複数のポート、複数モード

インターネットで使用されるほとんどのプロトコルとは異なり、FTP が適切に機能するには複数のネットワークポートが必要です。FTP クライアントアプリケーションが FTP サーバーへの接続を開始すると、コマンド ポート として知られるポート 21 をサーバー上で開きます。このポートは、すべてのコマンドをサーバーに発行するために使用されます。サーバーから要求されたデータはすべて、データ ポート を介してクライアントに返されます。データ接続のポート番号、およびデータ接続の初期化方法は、クライアントが アクティブ モードまたは パッシブ モードでデータを要求するかによって異なります。

これらのモードの定義は以下のとおりです。

アクティブモード

アクティブモードは、FTP プロトコルがデータをクライアントアプリケーションに転送するために使用する元の方法です。FTP クライアントがアクティブモードのデータ転送を開始すると、サーバーは、サーバー上のポート 20 から、クライアントが指定する IP アドレスと、ランダムで権限のないポート(1024 以上)への接続を開きます。この方法では、クライアントマシンがポート 1024 以上での接続を受け入れるように許可されている必要があります。インターネットなどのセキュリティー保護されていないネットワークの増加に伴い、ファイアウォールを使用したクライアントマ

シンの保護が明確になりました。このようなクライアント側のファイアウォールは、アクティブモードの FTP サーバーから着信接続を拒否することが多いため、パッシブモードが考案されました。

パッシブモード

パッシブモードはアクティブモードと同様に、FTP クライアントアプリケーションによって開始されます。サーバーからのデータを要求する際に、FTP クライアントはパッシブモードでデータにアクセスしたいことを示し、サーバーはサーバー上の IP アドレスとランダムで権限のないポート (1024 以上)を提供します。クライアントは、サーバー上のそのポートに接続して要求した情報をダウンロードします。

passive モードは、データ接続でのクライアント側のファイアウォール干渉の問題を解決しますが、サーバー側のファイアウォールの管理が複雑になる可能性があります。FTP サーバー上の特権のないポートの範囲を制限することで、サーバー上で開いているポートの数を減らすことができます。またこの方法により、サーバーを対象としたファイアウォールのルール設定の手順が簡略化されます。パッシブポートの制限に関する詳細は、「ネットワークオプション」 を参照してください。

26.2. FTP サーバー

Red Hat Enterprise Linux には、2 つの異なる FTP サーバーが同梱されています。

- Red Hat Content Accelerator: 高パフォーマンスの Web サーバーおよび FTP サービスを提供するカーネルベースの Web サーバー。速度は主要な設計目標であるため、機能が制限され、匿名 FTP サーバーとしてのみ実行されます。Red Hat Content Accelerator の設定および管理の詳細については、オンラインの http://www.redhat.com/docs/manuals/tux/ を参照してください。

- vsftpd: Red Hat Enterprise Linux に推奨される FTP サーバーである高速でセキュアな FTP デーモン。本章の残りの部分では、vsftpd に重点を置いています。

26.2.1. vsftpd

Very Secure FTP Daemon (vsftpd) は、高速で安定しており、最も重要なこととして安全であるようにゼロから設計されています。vsftpd は、多数の接続を効率的かつ安全に処理できるため、Red Hat Enterprise Linux とともに配布される唯一のスタンドアロン FTP サーバーです。

vsftpd で使用されるセキュリティーモデルには、主に 3 つの側面があります。

- 特権プロセス と非特権プロセスの強力な分離: 別個のプロセスが異なるタスクを処理し、各プロセスはタスクに必要な最小限の権限で実行されます。

- 高い権限を必要とするタスクを、必要最小限の特権を持つプロセスで処理: libcap ライブラリーにある互換性を利用して、通常は完全な root 権限を必要とするタスクを、権限が低いプロセスでより安全に実行できます。

- ほとんどのプロセスは chroot jail で実行 されます。可能な場合は、プロセスは root で共有されているディレクトリーに変更されます。その後、このディレクトリーは chroot jail とみなされます。たとえば、ディレクトリー /var/ftp/ がプライマリー共有ディレクトリーである場合、vsftpd は /var/ftp/ を / と呼ばれる新しいルートディレクトリーに再割り当てします。これにより、新しい root ディレクトリーに含まれていないディレクトリーに対する悪意のあるハッカーのアクティビティーが阻止されます。

これらのセキュリティープラクティスを使用すると、vsftpd によるリクエストの処理方法に以下のような影響があります。

- 親プロセスは、必要最小限の権限で稼働: 親プロセスは、リスクレベルを最低限に抑えるために必要とされる権限のレベルを動的に算出します。子プロセスは FTP クライアントとの直接的な対話を処理し、可能な限り権限なしに近い形で実行します。

- 高い 権限を必要とするすべての操作は、小さな親プロセスにより処理され ます。Apache HTTP Server と同様に、vsftpd は権限のない子プロセスを起動し、着信接続を処理します。これにより、権限のある親プロセスを最小限に抑えられ、比較的少ないタスクを処理することになります。

- 親プロセスでは、権限のない子プロセスからの要求はすべて信頼さ れません。子プロセスとの通信はソケット上で受信され、子プロセスからの情報の有効性は動作する前にチェックされます。

- FTP クライアントとの相互作用のほとんどは、chroot jail 内の権限のない子プロセスにより処理 されます。これらの子プロセスは権限がなく、共有ディレクトリーにのみアクセスできるため、クラッシュしたプロセスにより、攻撃者が共有ファイルにアクセスすることしかできません。

26.2.2. vsftpdでインストールされるファイル

vsftpd RPM はデーモン(/usr/sbin/vsftpd)、その設定および関連ファイル、および FTP ディレクトリーをシステムにインストールします。以下は、vsftpd 設定に関連するファイルおよびディレクトリーの一覧です。

- */etc/rc.d/init.d/vsftpd: /sbin/service コマンドが vsftpd を起動、停止、または再読み込みするために使用する 初期化スクリプト (initscript)。このスクリプトの使用についての詳細は、「vsftpdの起動と停止」 を参照してください。*

- */etc/pam.d/vsftpd: vsftpd の PAM （プラグ可能な認証モジュール）設定ファイルこのファイルは、FTP サーバーへのログインにユーザーが満たす必要のある要件を指定します。詳細は、「PAM (プラグ可能な認証モジュール)」 を参照してください。*

- */etc/vsftpd/vsftpd.conf: vsftpd の設定ファイル。このファイルに含まれる重要なオプションの一覧は、「vsftpd 設定オプション」 を参照してください。*

- */etc/vsftpd/ftpusers: vsftpd にログインすることができないユーザーの一覧です。デフォルトでは、このリストには、root、bin、および daemon ユーザーが含まれます。*

- */etc/vsftpd/user_list - このファイルは、userlist_deny ディレクティブが YES （デフォルト）に設定されているか、/etc/vsftpd/vsftpd.conf で NO に設定されているかによって、一覧表示されたユーザーへのアクセスを拒否または許可するように設定できます。/etc/vsftpd/user_list を使用してユーザーにアクセスを付与する場合、一覧表示されるユーザー名は /etc/vsftpd/ftpusers に表示され ない はずです。*

- */var/ftp/: vsftpd が提供するファイルが含まれるディレクトリーです。また、匿名ユーザー用の /var/ftp/pub/ ディレクトリーも含まれています。どちらのディレクトリーも誰でも読み取り可能ですが、root ユーザーのみが書き込み可能です。*

26.2.3. vsftpdの起動と停止

vsftpd RPM は、/etc/rc.d/init.d/vsftpd スクリプトをインストールします。これは、/sbin/service コマンドを使用してアクセスできます。

サーバーを起動するには、root で以下を入力します。

```
service vsftpd start
```

サーバーを停止するには、root で以下を入力します。

```
service vsftpd stop
```

restart オプションは、vsftpd を停止して起動する簡単な方法です。これは、vsftpd の設定ファイルを編集した後に設定変更を有効にする最も効率的な方法です。

サーバーを再起動するには、root で以下を入力します。

```
service vsftpd restart
```

condrestart (条件付き再起動)オプションは、現在実行している場合にのみ vsftpd を起動します。このオプションは、デーモンが実行されていない場合はデーモンを起動しないため、スクリプトに便利です。

条件付きでサーバーを再起動するには、root で以下を入力します。

```
service vsftpd condrestart
```

デフォルトでは、システムの起動時に vsftpd サービスが自動的に起動し ません。システムの起動時に vsftpd サービスが開始するように設定するには、/sbin/chkconfig、/usr/sbin/ntsysv、または Services Configuration Tool プログラムなどの initscript ユーティリティーを使用します。これらのツールの詳細は、18章 を参照してください。

26.2.3.1. vsftpdの複数コピーの起動

1 台のコンピューターが複数の FTP ドメインに使用される場合があります。これは、マルチホーミング と呼ばれる手法です。vsftpd を使用してマルチホーミングを行う 1 つの方法として、デーモンの複数のコピーを実行し、それぞれ独自の設定ファイルを使用します。

これを行うには、最初に、関連するすべての IP アドレスをシステム上のネットワークデバイスまたはエイリアスネットワークデバイスに割り当てます。ネットワークデバイスおよびデバイスエイリアスの設定に関する詳細は、17章Network Configuration を参照してください。ネットワーク設定スクリプトの詳細は、16章Network Interfaces を参照してください。

次に、FTP ドメインの DNS サーバーが正しいマシンを参照するように設定する必要があります。BIND とその設定ファイルの詳細は、19章 を参照してください。

vsftpd が異なる IP アドレスで要求に応答するには、デーモンの複数コピーが実行中である必要があります。「vsftpdの起動と停止」 で説明されているように、最初のコピーは vsftpd initscripts を使用して実行する必要があります。このコピーは、標準の設定ファイル /etc/vsftpd/vsftpd.conf を使用します。

追加の FTP サイトごとに、/etc/vsftpd/ ディレクトリーに一意の名前を持つ設定ファイル (/etc/vsftpd /vsftpd-site-2.conf など)が必要です。各設定ファイルは、root でのみ読み取り/書き込み可能でなければなりません。IPv4 ネットワーク上でリッスンする各 FTP サーバーの設定ファイル内で、以下のディレクティブは一意である必要があります。

```
listen_address=N.N.N.N
```

N.N.N.N を FTP サイトの一意 の IP アドレスに置き換えます。サイトが IPv6 を使用している場合は、代わりに listen_address6 ディレクティブを使用してください。

追加のサーバーに設定ファイルがある場合は、以下のコマンドを使用して root シェルプロンプトから vsftpd デーモンを起動する必要があります。

```
vsftpd /etc/vsftpd/<configuration-file> [amp   ]
```

上記のコマンドで、< configuration-file > をサーバーの設定ファイルの一意の名前に置き換えます（例： /etc/vsftpd/vsftpd-site-2.conf ）。

サーバーごとに変更するディレクティブには、以下のものがあります。

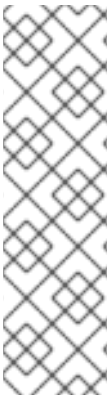
- *anon_root*
- *local_root*
- *vsftpd_log_file*
- *xferlog_file*

vsftpd の設定ファイルで利用可能なディレクティブの詳細な一覧は、「vsftpd 設定オプション」 を参照してください。

追加のサーバーが起動時に自動的に起動するように設定するには、上記のコマンドを /etc/rc.local ファイルの最後に追加します。

26.2.4. TLS を使用した vsftpd 接続の暗号化

ユーザー名、パスワード、およびデータを暗号化せずに送信する FTP の本質的に安全でない性質に対照するために、vsftpd デーモンは TLS プロトコルを使用して接続を認証し、すべての転送を暗号化するように設定できます。TLS をサポートする FTP クライアントは、TLS が有効になっている vsftpd と通信する必要があります。



注記

SSL (Secure Sockets Layer)は、セキュリティープロトコルの古い実装の名前です。新しいバージョンは TLS (Transport Layer Security)と呼ばれます。SSL には深刻なセキュリティー脆弱性があるため、新しいバージョン(TLS)のみを使用する必要があります。vsftpd サーバーに含まれるドキュメントや、vsftpd.conf ファイルで使用される設定ディレクティブは、セキュリティー関連の項目を参照する際に SSL 名を使用しますが、TLS はサポートされており、ssl_enable ディレクティブが YES に設定されている場合にデフォルトで使用されます。

vsftpd.conf ファイルの ssl_enable 設定ディレクティブを YES に設定して、TLS サポートを有効にします。ssl_enable オプションが有効な場合に自動的にアクティブになる他の TLS関連のディレクティブのデフォルト設定では、合理的に適切に設定された TLS のセットアップが提供されます。これには、全接続に TLS v1 プロトコルのみを使用する（安全でない SSL プロトコルバージョンの使用がデフォルトで無効）、匿名以外のすべてのログインがパスワードやデータ転送の送信に TLS を使用するように強制することなどが挙げられます。

例26.1 TLS を使用するように vsftpd の設定

以下の例では、設定ディレクティブは vsftpd.conf ファイルでセキュリティープロトコルの古い SSL バージョンを明示的に無効にします。

```
ssl_enable=YES
ssl_tlsv1=YES
ssl_sslv2=NO
ssl_sslv3=NO
```

設定を変更したら、vsftpd サービスを再起動します。

```
~]# service vsftpd restart
```

vsftpd による TLS の使用を調整するための他の TLS関連の設定ディレクティブについては、vsftpd.conf(5) の man ページを参照してください。また、一般的に使用されるその他の vsftpd.conf 設定ディレクティブの説明は、「vsftpd 設定オプション」 を参照してください。

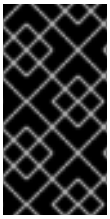
26.2.5. vsftpd 設定オプション

vsftpd では、広く利用可能な他の FTP サーバーにはカスタマイズレベルが提供されない場合がありますが、ほとんどの管理者のニーズに対応するのに十分なオプションが提供されます。機能が緩和された制限設定およびプログラムによるエラーの制限が過度に行われていないという事実。

vsftpd のすべての設定は設定ファイル /etc/vsftpd/vsftpd.conf によって処理されます。各ディレクティブは、ファイル内の独自の行にあり、以下の形式に従います。

```
<directive>=<value>
```

各ディレクティブで、< directive > を有効なディレクティブに置き換え、< value> を有効な値に置き換えます。

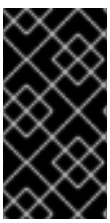


重要な影響

< directive>、同じ記号、およびディレクティブの & lt; value> の間には、スペースを 入れないでください。

コメント行は、ハッシュマーク(#)の前に付け、デーモンによって無視されます。

利用可能なすべてのディレクティブの一覧は、vsftpd.conf の man ページを参照してください。



重要な影響

vsftpd をセキュアにする方法の概要は、「サーバーセキュリティー」 を参照してください。

以下は、/etc/vsftpd/vsftpd.conf 内の重要なディレクティブの一覧です。vsftpd の設定ファイル内で明示的に見つからないディレクティブはすべて、デフォルト値に設定されます。

26.2.5.1. デーモンオプション

以下は、vsftpd デーモンの全体的な動作を制御するディレクティブの一覧です。

- listen: 有効にすると、vsftpd はスタンドアロンモードで実行されます。Red Hat Enterprise Linux は、この値を YES に設定します。このディレクティブは、listen_ipv6 ディレクティブと組み合わせて使用することはできません。

  デフォルト値は NO です。

- listen_ipv6: 有効にすると、vsftpd はスタンドアロンモードで実行されますが、IPv6 ソケットのみをリッスンします。このディレクティブは listen ディレクティブと併用できません。

  デフォルト値は NO です。

- session_support: 有効にすると、vsftpd は PAM (Pluggable Authentication Modules) を介して各ユーザーのログインセッションを維持しようとします。詳細は、「PAM (プラグ可能な認証モジュール)」 を参照してください。セッションロギングが必要ない場合は、このオプションを無効にすると、vsftpd のプロセスが少なくなります。

  デフォルト値は YES です。

26.2.5.2. ログインオプションとアクセス制御

ログイン動作とアクセス制御メカニズムを制御するディレクティブの一覧を以下に示します。

- anonymous_enable: 有効にすると、匿名ユーザーはログインできます。ユーザー名 anonymous および ftp が受け入れられます。

  デフォルト値は YES です。

  匿名ユーザーに影響するディレクティブの一覧は、「Anonymous User Options」 を参照してください。

- *banned_email_file - deny_email_enable ディレクティブが YES に設定されている場合、このディレクティブは、サーバーへのアクセスを許可しない匿名電子メールパスワードのリストを含むファイルを指定します。*

  デフォルト値は /etc/vsftpd.banned_emails です。

- *banner_file: サーバーへの接続が確立されたときに表示されるテキストを含むファイルを指定します。このオプションは、ftpd_banner ディレクティブで指定されたテキストを上書きします。*

  このディレクティブにはデフォルト値がありません。

- *cmds_allowed - サーバーが許可する FTP コマンドのコンマ区切りの一覧を指定します。その他のコマンドはすべて拒否されます。*

  このディレクティブにはデフォルト値がありません。

- *deny_email_enable: 有効にすると、/etc/vsftpd.banned_emails で指定された電子メールパスワードを使用する匿名ユーザーは、サーバーへのアクセスを拒否します。このディレクティブによって参照されるファイルの名前は、banned_email_file ディレクティブを使用して指定できます。*

  デフォルト値は NO です。

- *ftpd_banner: 有効にすると、このディレクティブ内で指定された文字列は、サーバーへの接続が確立されると表示されます。このオプションは banner_file ディレクティブで上書きできます。*

  デフォルトでは、vsftpd は標準バナーを表示します。

- *local_enable: 有効にすると、ローカルユーザーはシステムにログインできます。*

  デフォルト値は YES です。

ローカルユーザーに影響するディレクティブの一覧は、「ローカルユーザーオプション」を参照してください。

- pam_service_name - vsftpd の PAM サービス名を指定します。

  デフォルト値は ftp です。Red Hat Enterprise Linux 5.10 では、このオプションは 設定ファイルで vsftpd に設定されます。

- tcp_wrappers: 有効にすると、サーバーへのアクセスを付与するために TCP ラッパーが使用されます。FTP サーバーが複数の IP アドレスに設定されている場合、VSFTPD_LOAD_CONF オプションを使用して、クライアントが要求した IP アドレスに基づいて異なる設定ファイルを読み込むことができます。

  デフォルト値は NO です。Red Hat Enterprise Linux 5.10 では、このオプションは 設定ファイルで YES に設定されます。

  TCP ラッパーの詳細は、「TCP Wrapper および xinetd」 を参照してください。

- userlist_deny: userlist_enable ディレクティブと併用し、NO に設定すると、userlist_file ディレクティブで指定されたファイルにユーザー名がリストされていない限り、すべてのローカルユーザーはアクセスが拒否されます。クライアントがパスワードを要求する前にアクセスが拒否されるため、このディレクティブを NO に設定すると、ローカルユーザーが暗号化されていないパスワードをネットワーク経由で送信できなくなります。

  デフォルト値は YES です。

- userlist_enable: 有効にすると、userlist_file ディレクティブで指定されたファイルに一覧表示されるユーザーはアクセスが拒否されます。クライアントがパスワードを要求する前にアクセスが拒否されるため、ユーザーは暗号化されていないパスワードをネットワーク経由で送信できなくなります。

  デフォルト値は NO です。Red Hat Enterprise Linux 5.10 では、このオプションは 設定ファイルで YES に設定されます。

- userlist_file: userlist_enable ディレクティブが有効な場合に vsftpd が参照するファイルを指定します。

デフォルト値は /etc/vsftpd.user_list で、インストール時に作成されます。

26.2.5.3. Anonymous User Options

以下は、サーバーへの匿名ユーザーアクセスを制御するディレクティブの一覧です。このオプションを使用するには、anonymous_enable ディレクティブを YES に設定する必要があります。

- *anon_mkdir_write_enable: write_enable ディレクティブと組み合わせて使用すると、匿名ユーザーは書き込み権限を持つ親ディレクトリーに新しいディレクトリーを作成できます。*

  デフォルト値は NO です。

- *anon_root: 匿名ユーザーのログイン後に vsftpd が変更するディレクトリーを指定します。*

  このディレクティブにはデフォルト値がありません。

- *anon_upload_enable: write_enable ディレクティブと組み合わせて使用すると、匿名ユーザーは書き込み権限を持つ親ディレクトリーにファイルをアップロードできます。*

  デフォルト値は NO です。

- *anon_world_readable_only: 有効にすると、匿名ユーザーは誰でも読み取り可能なファイルのみをダウンロードできます。*

  デフォルト値は YES です。

- *ftp_username: 匿名 FTP ユーザーに使用されるローカルユーザーアカウント(/etc/passwdに記載されている)を指定します。ユーザーの /etc/passwd で指定したホームディレクトリーは、匿名 FTP ユーザーの root ディレクトリーです。*

デフォルト値は ftp です。

- no_anon_password: 有効にすると、匿名ユーザーはパスワードを要求されません。

  デフォルト値は NO です。

- secure_email_list_enable: 有効にすると、匿名ログインの指定された電子メールパスワードのリストのみが受け入れられます。これは、仮想ユーザーを必要とせずに、パブリックコンテンツへの制限されたセキュリティーを提供する便利な方法です。

  提供されたパスワードが /etc/vsftpd.email_passwords に表示されない限り、匿名ログインは阻止されます。ファイル形式は、1 行に 1 つのパスワードで、末尾の空白はありません。

  デフォルト値は NO です。

26.2.5.4. ローカルユーザーオプション

以下は、ローカルユーザーがサーバーにアクセスする方法を特徴付けるディレクティブの一覧です。これらのオプションを使用するには、local_enable ディレクティブを YES に設定する必要があります。

- chmod_enable - 有効にすると、FTP コマンド SITE CHMOD がローカルユーザーに許可されます。このコマンドを使用すると、ファイルのパーミッションを変更できます。

  デフォルト値は YES です。

- chroot_list_enable: 有効にすると、chroot_list_file ディレクティブで指定されたファイルに一覧表示されているローカルユーザーは、ログイン時に chroot jail に配置されます。

  chroot_local_user ディレクティブと組み合わせて使用すると、chroot_list_file ディレクティブで指定されたファイルに一覧表示されるローカルユーザーは、ログイン時に chroot jail に置かれ ません。

デフォルト値は NO です。

- *chroot_list_file: chroot_list_enable ディレクティブが YES に設定されている場合に、参照されるローカルユーザーの一覧を含むファイルを指定します。*

  デフォルト値は /etc/vsftpd.chroot_list です。

- *chroot_local_user: 有効にすると、ローカルユーザーはログイン後にホームディレクトリーに変更します。*

  デフォルト値は NO です。

WARNING

chroot_local_user を有効にすると、特にアップロード権限を持つユーザー向けに、多くのセキュリティー問題が開きます。このため、推奨されません。

- *guest_enable - 有効にすると、匿名以外のすべてのユーザーが guest (guest_username ディレクティブで指定されたローカルユーザー)としてログインされます。*

  デフォルト値は NO です。

- *guest_username: ゲスト ユーザーがマップされるユーザー名を指定します。*

  デフォルト値は ftp です。

- *local_root: ローカルユーザーのログイン後に vsftpd が変更するディレクトリーを指定します。*

このディレクティブにはデフォルト値がありません。

- local_umask - ファイル作成の umask 値を指定します。デフォルト値は 8 進法（ベースが 8 の数値システム）で、接頭辞 0 を含むことに注意してください。それ以外の場合は、値は base-10 整数として処理されます。

  デフォルト値は 022 です。

- passwd_chroot_enable - chroot_local_user ディレクティブと併用すると、/etc/passwd 内のホームディレクトリーフィールドにある /./ の発生に基づいて、vsftpd change-roots local users を実行します。

  デフォルト値は NO です。

- user_config_dir - そのユーザーの特定の設定を含むローカルシステムユーザーの名前を作成する設定ファイルを含むディレクトリーへのパスを指定します。ユーザーの設定ファイルのディレクティブは、/etc/vsftpd/vsftpd.conf にあるディレクティブを上書きします。

  このディレクティブにはデフォルト値がありません。

26.2.5.5. ディレクトリーオプション

以下は、ディレクトリーに影響するディレクティブの一覧です。

- dirlist_enable: 有効にすると、ユーザーはディレクトリー一覧を表示できます。

  デフォルト値は YES です。

- dirmessage_enable: 有効にすると、ユーザーがメッセージファイルを含むディレクトリーを入力すると常にメッセージが表示されます。このメッセージは、現在のディレクトリー内にあります。このファイルの名前は message_file ディレクティブで指定され、デフォルトでは .message です。

  デフォルト値は NO です。Red Hat Enterprise Linux 5.10 では、このオプションは 設定

ファイルで YES に設定されます。

- *force_dot_files: 有効にすると、ドット（）で始まるファイルはディレクトリーリストに一覧表示され ます(. および . . ファイルを除く)。*

  デフォルト値は NO です。

- *hide_ids: 有効にすると、すべてのディレクトリーの一覧に ftp が各ファイルのユーザーおよびグループとして表示されます。*

  デフォルト値は NO です。

- *message_file: dirmessage_enable ディレクティブを使用する際のメッセージファイルの名前を指定します。*

  デフォルト値は .message です。

- *text_userdb_names: 有効にすると、UID エントリーおよび GID エントリーの代わりにテキストユーザー名とグループ名が使用されます。このオプションを有効にすると、サーバーのパフォーマンスが低下する可能性があります。*

  デフォルト値は NO です。

- *use_localtime - 有効にすると、ディレクトリーの一覧が、GMT ではなくコンピューターのローカル時間を表示します。*

  デフォルト値は NO です。

26.2.5.6. ファイル転送オプション

以下は、ディレクトリーに影響するディレクティブの一覧です。

- *download_enable: 有効にすると、ファイルのダウンロードが許可されます。*

  デフォルト値は YES です。

- *chown_uploads: 有効にすると、匿名ユーザーがアップロードしたすべてのファイルは chown_username ディレクティブで指定されたユーザーによって所有されます。*

  デフォルト値は NO です。

- *chown_username: chown_uploads ディレクティブが有効な場合、匿名でアップロードされたファイルの所有権を指定します。*

  デフォルト値は root です。

- *write_enable: 有効にすると、DELE、RNFR、STOR など、ファイルシステムを変更できる FTP コマンドが許可されます。*

  デフォルト値は YES です。

26.2.5.7. ロギングのオプション

以下は、vsftpd のログ動作に影響するディレクティブの一覧です。

- *dual_log_enable - xferlog_enable と併用すると、vsftpd は、xferlog_file ディレクティブ（デフォルトでは/var/log/xferlog ）で指定されたファイルに wu-ftpdと互換性のあるログという 2 つのファイルを同時に書き込みます。 vsftpd _log_file ディレクティブで指定された標準の vsftpd ログファイル（デフォルトでは/var/log/vsftpd.log ）。*

  デフォルト値は NO です。

- *log_ftp_protocol - xferlog_enable と組み合わせて有効にし、xferlog_std_format を NO に設定すると、すべての FTP コマンドおよび応答がログに記録されます。このディレクティブはデバッグに役立ちます。*

デフォルト値は NO です。

- *syslog_enable: xferlog_enable と併用すると、通常は vsftpd_log_file ディレクティブで指定された標準の vsftpd ログファイル（デフォルトでは/var/log/vsftpd.log ）に書き込まれたすべてのロギングは、FTPD 機能ではなくシステムロガーに送信されます。*

  デフォルト値は NO です。

- *vsftpd_log_file: vsftpd ログファイルを指定します。このファイルを使用するには、xferlog_enable を有効にし、xferlog_std_format を NO に設定する必要があります。または、xferlog_std_format が YES に設定されている場合は、dual_log_enable を有効にする必要があります。syslog_enable が YES に設定されている場合、このディレクティブで指定されたファイルの代わりにシステムログが使用されることに注意してください。*

  デフォルト値は /var/log/vsftpd.log です。

- *xferlog_enable: 有効にすると、vsftpd は接続をログに記録し(vsftpd 形式のみ)、vsftpd_log_file ディレクティブ（デフォルトでは/var/log/vsftpd.log ）で指定されたログファイルに情報を転送します。xferlog_std_format が YES に設定されている場合、ファイル転送情報はログに記録されますが、接続はログに記録されず、xferlog_file （デフォルトでは/var/log/xferlog ）で指定されたログファイルが代わりに使用されます。dual_log_enable が YES に設定されている場合は、ログファイルとログ形式の両方が使用されることに注意してください。*

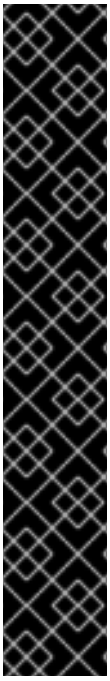
  デフォルト値は NO です。Red Hat Enterprise Linux 5.10 では、このオプションは 設定ファイルで YES に設定されます。

- *xferlog_file - wu-ftpd互換のログファイルを指定します。このファイルを使用するには、xferlog_enable を有効にし、xferlog_std_format を YES に設定する必要があります。また、dual_log_enable が YES に設定されている場合にも使用されます。*

  デフォルト値は /var/log/xferlog です。

- *xferlog_std_format - xferlog_enable と併用すると、wu-ftpdと互換性のあるファイル転送ログのみが xferlog_file ディレクティブで指定されたファイルに書き込まれます（デフォルトでは/var/log/xferlog ）。このファイルは、ファイル転送のみをログに記録し、サーバーへの接続はログに記録されないことに注意してください。*

デフォルト値は NO です。Red Hat Enterprise Linux 5.10 では、このオプションは 設定ファイルで YES に設定されます。



重要な影響

古い wu-ftpd FTP サーバーによって書き込まれたログファイルとの互換性を維持するために、Red Hat Enterprise Linux では、xferlog_std_format ディレクティブが YES に設定されます。ただし、この設定は、サーバーへの接続がログに記録されないことを意味します。

vsftpd 形式でログ接続を記録し、wu-ftpdと互換性のあるファイル転送ログを維持するには、dual_log_enable を YES に設定します。

wu-ftpdと互換性のあるファイル転送ログを維持することが重要でない場合は、xferlog_std_format を NO に設定するか、ハッシュマーク(#)で行をコメントするか、行を完全に削除します。

26.2.5.8. ネットワークオプション

以下は、vsftpd がネットワークと対話する方法に影響を与えるディレクティブの一覧です。

- accept_timeout: パッシブモードを使用して接続を確立するクライアントの時間を指定します。

  デフォルト値は 60 です。

- anon_max_rate: 匿名ユーザーの最大データ転送速度をバイト/秒単位で指定します。

  デフォルト値は 0 で、転送レートを制限しません。

- connect_from_port_20 有効にすると、vsftpd はアクティブモードのデータ転送中にサーバー上でポート 20 を開くのに十分な権限で実行されます。このオプションを無効にすると、vsftpd は権限が少なく実行できますが、一部の FTP クライアントと互換性がない場合があります。

デフォルト値は NO です。Red Hat Enterprise Linux 5.10 では、このオプションは 設定ファイルで YES に設定されます。

- *connect_timeout - アクティブモードを使用するクライアントがデータ接続に応答する最大時間を秒単位で指定します。*

  デフォルト値は 60 です。

- *data_connection_timeout - データ転送を停止できる最大時間を秒単位で指定します。トリガーされると、リモートクライアントへの接続は閉じられます。*

  デフォルト値は 300 です。

- *ftp_data_port - connect_from_port_20 が YES に設定されている場合に、アクティブなデータ接続に使用されるポートを指定します。*

  デフォルト値は 20 です。

- *idle_session_timeout - リモートクライアントからのコマンド間の最大時間を指定します。トリガーされると、リモートクライアントへの接続は閉じられます。*

  デフォルト値は 300 です。

- *listen_address: vsftpd がネットワーク接続をリッスンする IP アドレスを指定します。*

  このディレクティブにはデフォルト値がありません。

ヒント

異なる IP アドレスを提供する vsftpd のコピーを複数実行する場合は、vsftpd デーモンのコピーごとに、このディレクティブの値が異なる必要があります。マルチホーム FTP サーバーの詳細は、「vsftpdの複数コピーの起動」 を参照してください。

- *listen_address6: listen_ipv6 が YES に設定されている場合に、vsftpd がネットワーク接続をリッスンする IPv6 アドレスを指定します。*

  このディレクティブにはデフォルト値がありません。

  ヒント

  異なる IP アドレスを提供する vsftpd のコピーを複数実行する場合は、vsftpd デーモンのコピーごとに、このディレクティブの値が異なる必要があります。マルチホーム FTP サーバーの詳細は、「vsftpdの複数コピーの起動」 を参照してください。

- *listen_port: vsftpd がネットワーク接続をリッスンするポートを指定します。*

  デフォルト値は 21 です。

- *local_max_rate: サーバーにログインしているローカルユーザーの最大レートデータがバイト/秒単位で転送されるように指定します。*

  デフォルト値は 0 で、転送レートを制限しません。

- *max_clients: スタンドアロンモードで実行している場合に、サーバーに接続できる同時クライアントの最大数を指定します。追加のクライアント接続があると、エラーメッセージが表示されます。*

  デフォルト値は 0 で、接続を制限しません。

- *max_per_ip - 同じソース IP アドレスから接続可能なクライアントの最大数を指定します。*

  デフォルト値は 0 で、接続を制限しません。

- *pasv_address: ネットワークアドレス変換(NAT)ファイアウォールの背後にあるサーバー*

向けのサーバーのパブリック向け IP アドレスの IP アドレスを指定します。これにより、vsftpd はパッシブモード接続の正しいリターンアドレスを渡すことができます。

このディレクティブにはデフォルト値がありません。

- *pasv_enable: 有効にすると、パッシブモードの接続が許可されます。*

  デフォルト値は YES です。

- *pasv_max_port: パッシブモード接続のために FTP クライアントに送信される最大ポートを指定します。この設定はポート範囲を制限するために使用され、ファイアウォールルールの作成が容易になります。*

  デフォルト値は 0 で、パッシブポートの範囲の最大値は制限しません。値は 65535 を超えることができません。

- *pasv_min_port: パッシブモード接続の FTP クライアントに送信される最小ポートを指定します。この設定はポート範囲を制限するために使用され、ファイアウォールルールの作成が容易になります。*

  デフォルト値は 0 で、パッシブポート範囲を最小限に制限しません。値は 1024 未満にしないでください。

- *pasv_promiscuous: 有効にすると、データ接続は、同じ IP アドレスから発信されていることを確認するためにチェックされません。この設定は、特定のタイプのトンネリングにのみ役立ちます。*

  デフォルト値は NO です。

注意

絶対に必要なセキュリティー機能は有効にしないでください。パッシブモード接続がデータ転送を開始する制御接続と同じ IP アドレスから送信されることを検証する重要なセキュリティー機能を無効にするためです。

- port_enable: 有効にすると、アクティブモードの接続が許可されます。

  デフォルト値は YES です。

26.2.6. 関連情報

vsftpd の詳細は、以下の資料を参照してください。

26.2.6.1. インストールされているドキュメント

- /usr/share/doc/vsftpd- <version-number> / ディレクトリー : < version-number > を、インストールした vsftpd パッケージのバージョンに置き換えます。このディレクトリーには、ソフトウェアの基本情報を含む README が含まれています。TUNING ファイルには基本的なパフォーマンスチューニングのヒントと SECURITY/ ディレクトリーには、vsftpd が使用するセキュリティーモデルに関する情報が含まれています。

- vsftpd 関連の man ページ : デーモンおよび設定ファイルの man ページは多数あります。以下は、重要な man ページの一部を示しています。

  サーバーアプリケーション

  - man vsftpd: vsftpd で利用可能なコマンドラインオプションを説明しています。

  設定ファイル

  - man vsftpd.conf: vsftpd の設定ファイル内で利用可能なオプションの詳細な一覧が含まれています。

- *man 5 hosts_access : TCP ラッパー設定ファイルで利用可能な形式およびオプション(hosts.allow および hosts.deny)を説明しています。*

26.2.6.2. 便利な Web サイト

- *http://vsftpd.beasts.org/: vsftpd プロジェクトページは、最新のドキュメントを見つけ、ソフトウェアの作成者に問い合わせるのに適した場所です。*
- *http://slacksite.com/other/ftp.html: この Web サイトでは、アクティブモードとパッシブモード FTP の相違点を簡単に説明します。*
- *http://www.ietf.org/rfc/rfc0959.txt: IETF からの FTP プロトコルのオリジナルの Request for Comments (RFC)。*

第27章 メール

初期の 1960 秒で電子メール（電子メール）の生存日が発生しました。メールボックスは、そのユーザーのみが読み取り可能なユーザーのホームディレクトリー内のファイルでした。プリミティブメールアプリケーションは、ファイルの下部に新しいテキストメッセージを追加し、ユーザーが継続的に拡大したファイルをウォードし、特定のメッセージを見つけます。このシステムは、同じシステムのユーザーにのみメッセージを送信できました。

Ray Tomlinson という名前のコンピューターエンジニアが ARPANET を介して 2 台のマシン間でテストメッセージを送信した場合、電子メールメッセージファイルの最初のネットワーク転送は 1971 年にかかりました（インターネットへのプリキューター）。メールによる通信はまもなく一般的になり、ARPANET のトラフィックの 50% は 2 年未満に設定されています。

現在、標準化されたネットワークプロトコルに基づく電子メールシステムは、インターネット上で最も広く使用されているサービスの一部に進化しています。Red Hat Enterprise Linux は、メールを提供し、アクセスするための高度なアプリケーションを多数提供します。

本章では、現在使用している最新の電子メールプロトコルと電子メールの送受信用に設計されたプログラムについて説明します。

27.1. メールプロトコル

今日、電子メールはクライアント/サーバーのアーキテクチャーを使用して配信されています。電子メールのメッセージは、メールクライアントプログラムを使用して作成されます。次に、このプログラムがメッセージをサーバーに送信します。その後、サーバーはメッセージを受信者のメールサーバーに転送します。ここで、メッセージは受信者の電子メールクライアントに渡されます。

このプロセスを有効にするために、各種の標準のネットワークプロトコルが異なるマシンによる (多くの場合、異なるオペレーティングシステムで、異なる電子メールプログラムを使用) 電子メールの送受信を可能にしています。

以下は、電子メールの転送に最も一般的に使用されているプロトコルです。

27.1.1. メール転送プロトコル

クライアントアプリケーションからサーバーへのメール配信、および送信元サーバーから宛先サーバーへのメール配信は、SMTP(Simple Mail Transfer Protocol)によって処理されます。

27.1.1.1. SMTP

SMTP の第一の目的は、メールサーバー間における電子メールの転送です。ただし、これは、メールクライアントにも重要です。メールを送信するには、クライアントが送信メールサーバーにメッセージを送信し、配信先メールサーバーに接続します。このため、メールクライアントの設定時に SMTP サーバーを指定する必要があります。

Red Hat Enterprise Linux では、ユーザーはローカルマシンで SMTP サーバーを設定してメール配信を処理できます。ただし、送信メール用にリモート SMTP サーバーを設定することも可能です。

SMTP プロトコルに関して重要なのは認証が不要である点です。これにより、インターネット上の誰でも、個人や大規模なグループに対してでも電子メールを送信できます。ジュークメールや スパムを可能にする SMTP のこの特性です。リレー制限を課すと、インターネット上の任意のユーザーが、ご使用の SMTP サーバーを介してインターネット上の別のサーバーへ電子メールを送信することが制限されます。このような制限を課さないサーバーは、オープンリレー サーバーと呼ばれます。

デフォルトでは、Sendmail (/usr/sbin/sendmail)は、Red Hat Enterprise Linux におけるデフォルトの SMTP プログラムです。ただし、Postfix (/usr/sbin/postfix)と呼ばれるシンプルなメールサーバーアプリケーションも利用できます。

27.1.2. メールアクセスプロトコル

メールサーバーから電子メールを取得するために、電子メールクライアントアプリケーションが使用する主なプロトコルは 2 つあります。POP(Post Office Protocol)と Internet Message Access Protocol (IMAP)。

27.1.2.1. POP

Red Hat Enterprise Linux のデフォルトの POP サーバーは /usr/lib/cyrus-imapd/pop3d で、cyrus-imapd パッケージで提供されます。POP サーバーを使用する場合、電子メールメッセージは電子メールクライアントアプリケーションによってダウンロードされます。デフォルトでは、ほとんどの POP 電子メールクライアントでは、電子メールサーバーのメッセージが正常に転送された後に削除されるように自動的に設定されます。ただし、この設定は通常は変更できます。

POP は、電子メールのファイル添付を可能にするMIME(Multipurpose Internet Mail Extensions)などの重要なインターネットメッセージング標準と完全に互換性があります。

POP は、電子メールを読み取るシステムを 1 つ持つユーザーにとって最適に機能します。また、インターネットやメールサーバーを持つネットワークに常時接続していないユーザーにもうまく機能します。ネットワーク速度が遅いユーザーの場合は、POP は各メッセージのコンテンツ全体をダウンロードするために、認証時にクライアントプログラムを必要とします。このプロセスは、メッセージに大きなファイルが添付されている場合に長時間かかる場合があります。

標準 POP プロトコルの最新バージョンは POP3 です。

ただし、あまり使用されていない POP プロトコルのバリアントが複数あります。

- APOP: MDS 認証を使用した POP3。ユーザーのパスワードのエンコードされたハッシュは、暗号化されていないパスワードを送信するのではなく、電子メールクライアントからサーバーに送信されます。

- KPOP: Kerberos 認証を使用した POP3。詳細は、「Kerberos」 を参照してください。

- RPOP: RPOP 認証を使用した POP3 です。これは、パスワードに似たユーザーごとの ID を使用し、POP 要求を認証します。ただし、この ID は暗号化されないため、RPOP は標準の POP よりも安全ではありません。

セキュリティーを強化するには、クライアント認証およびデータ転送セッションにSSL(Secure Socket Layer)暗号化を使用できます。これは、ipop3s サービスを使用するか、/usr/sbin/stunnel プログラムを使用して有効にできます。詳細は、「通信のセキュリティー保護」 を参照してください。

27.1.2.2. IMAP

Red Hat Enterprise Linux のデフォルトの IMAP サーバーは /usr/lib/cyrus-imapd/imapd で、cyrus-imapd パッケージで提供されます。IMAP メールサーバーを使用する場合、電子メールメッセージはサーバーに残るので、ユーザーはメッセージの読み取りや削除が可能です。また、IMAP により、クライアントアプリケーションはサーバー上でメールディレクトリーを作成、名前変更、または削除して電子メールを整理および保存することもできます。

IMAP は、複数のマシンを使用して電子メールにアクセスするユーザーに特に便利です。このプロトコルでは、メッセージが開封されるまでは、電子メールのヘッダー情報しかダウンロードされず帯域幅を節減できるため、低速な接続でメールサーバーに接続するユーザーにも便利です。ユーザーは、メッセージを表示またはダウンロードすることなく削除することも可能です。

便宜上、IMAP クライアントアプリケーションはメッセージのコピーをローカルでキャッシュできるため、IMAP サーバーに直接接続していない場合に、ユーザーは以前に読み取りメッセージを閲覧できます。

IMAP は POP と同様に、電子メールのファイル添付を可能にする MIME などの重要なインターネットメッセージング標準と完全に互換性があります。

セキュリティーを強化するには、クライアント認証およびデータ転送セッションに SSL 暗号化を使用できます。これは、imaps サービスを使用するか、/usr/sbin/stunnel プログラムを使用して有効にできます。詳細は、「通信のセキュリティー保護」 を参照してください。

無償や商用の IMAP クライアントおよびサーバーは他にも提供されています。これらの多くは、IMAP プロトコルを拡張し、追加機能を提供します。包括的なリストは http://www.imap.org/products/longlist.htm でオンラインで参照できます。

27.1.2.3. Dovecot

IMAP および POP3 プロトコルを実装する imap- login デーモンおよび pop 3-login デーモンは、dovecot パッケージに含まれています。IMAP および POP の使用は、dovecot を使用して設定されます。デフォルトでは、dovecot は IMAP ポートと POP3 ポートの両方でリッスンし、さらに安全なバリアントもリッスンします。dovecot の使用を開始するには、以下を実行します。

1. デーモンを起動します。

```
service dovecot start
```

2. 次回の再起動後にデーモンが自動的に起動するようにします。

```
chkconfig dovecot on
```

dovecot は IMAP サーバーを起動したことを報告するだけで、POP3 サーバーも起動することに注意してください。

SMTP とは異なり、これらのプロトコルの両方で、ユーザー名とパスワードを使用してクライアントを認証する必要があります。デフォルトでは、両方のプロトコルのパスワードは、暗号化されていないネットワーク上で渡されます。

dovecot で SSL を設定するには、以下を実行します。

- 必要に応じて、dovecot 設定ファイル /etc/pki/dovecot/dovecot-openssl.cnf を編集します。ただし、一般的なインストールでは、このファイルを変更する必要はありません。
- /etc/pki/dovecot/certs/dovecot.pem ファイルおよび /etc/pki/dovecot/private/dovecot.pem ファイルの名前を変更、移動、または削除します。

- dovecot の自己署名証明書を作成する /usr/share/doc/dovecot-1.0/examples/mkcert.sh スクリプトを実行します。証明書は、/etc/pki/dovecot/certs および /etc/pki/dovecot/private ディレクトリーにコピーされます。変更を実装するには、dovecot を再起動します (/sbin/service dovecot を再起動します)。

dovecot の詳細は、オンライン(http://www.dovecot.org)を参照してください。

注記

デフォルトでは、SSLv2 および SSLv3 は Dovecot では許可されません。POODLE SSL 脆弱性 (CVE-2014-3566) の影響を受けないようにするためです。詳細は、Postfix および Dovecot の POODLE SSL 3.0 脆弱性(CVE-2014-3566)の解決 を参照してください。

27.2. 電子メールプログラムの分類

一般的に、すべての電子メールアプリケーションは 3 つのタイプのうち 1 つ以上に分類されます。それぞれの分類は、電子メールメッセージの移動および管理のプロセスにおいてそれぞれ特定のロールを果たします。大半のユーザーはメッセージの送受信に使用する特定の電子メールプログラムだけを認識しますが、電子メールを正しい送信先に届けるにはすべての電子メールプログラムが重要になります。

27.2.1. メール転送エージェント (Mail Transport Agent)

MTA(メール転送エージェント)は、SMTP を使用してホスト間で電子メールメッセージを転送します。メッセージは目的の送信先に移動する時、様々な MTA に関わることがあります。

マシン間のメッセージ配信は簡単に見えるかもしれませんが、配信のためにある MTA がメッセージを受け入れることが可能か、または受け入れるべきかを判断する過程全体は非常に複雑です。さらに、スパムの問題により、特定の MTA の使用は、通常 MTA の設定または MTA が置かれているネットワークのアクセス設定によって制限されます。

最新の電子メールクライアントプログラムの多くは、メール送信時に MTA として機能します。ただし、このアクションは、実際の MTA のロールと混同しないようにしてください。メールクライアントプログラムは MTA などのメールを送信できる唯一の理由は、アプリケーションを実行しているホストに独自の MTA がないためです。これは、UNIX ベース以外のオペレーティングシステムの電子メールクライアントプログラムに特に当てはまります。ただし、これらのクライアントプログラムは、アウトバウンドメッセージのみを MTA に送信し、メッセージを目的の受信者のメールサーバーに直接送信しません。

Red Hat Enterprise Linux は Sendmail と Postfix の 2 つの MTA をインストールするため、電子メールクライアントプログラムは MTA として機能する必要はありません。Red Hat Enterprise Linux には、Fetchmail と呼ばれる特別な目的の MTA も含まれています。

Sendmail、Postfix、Fetchmail の詳細は、「メール転送エージェント (MTA)」 を参照してください。

27.2.2. メール配信エージェント (MDA)

Mail Delivery Agent (MDA)は MTA によって呼び出され、適切なユーザーのメールボックスに受信メールをファイル送ります。多くの場合、MDA は実際には mail や Procmail などのLDA(Local Delivery Agent)です。

電子メールクライアントアプリケーションが読み取り可能なポイントに配信されるメッセージを実際に処理するプログラムは、いずれも MDA と見なすことができます。このため、一部の MTA (Sendmail や Postfix など)は、ローカルユーザーのメールスプールファイルに新しい電子メールメッセージを追加すると、MDA のロールを埋めることができます。通常、MDA はシステム間でのメッセージの転送やユーザーインターフェイスの提供は行いません。MDA は、ローカルマシン上でメッセージの配信と並べ替えを行い、電子メールクライアントアプリケーションがアクセスできるようにします。

27.2.3. メールユーザーエージェント

Mail User Agent (MUA)は、メールクライアントアプリケーションと同義語です。MustRunAs は、少なくとも非常に多くのプログラムで、電子メールメッセージの読み取りと作成を行うことができます。多くのMUA は、POP プロトコルまたは IMAP プロトコルを介してメッセージを取得し、メッセージを格納するメールボックスを設定し、送信メッセージを MTA に送信することができます。

MUA は、Evolution などのグラフィカルであったり、Mutt などの非常にシンプルなテキストベースのインターフェイスを持つ場合があり ます。

27.3. メール転送エージェント (MTA)

Red Hat Enterprise Linux には、Sendmail と Postfix の 2 つの主要 MTA が含まれています。Sendmail はデフォルトの MTA として設定されていますが、デフォルトの MTA を Postfix に切り替えることができます。

27.3.1. Sendmail

Sendmail の主な目的は、他の MTA と同様に、通常は SMTP プロトコルを使用してホスト間で電子メールを安全に転送することです。ただし、Sendmail は高度な設定が可能であるため、使用されるプロトコルを含め、電子メールの処理方法についてのほぼすべての側面を制御できます。多くのシステ

ム管理者は、電源とスケーラビリティーにより、Sendmail を MTA として使用することを選択しています。

27.3.1.1. 用途と制約

認識すべき重要な点は、Sendmail ができないことではなく、Sendmail が何であるか、何ができるのかということです。複数のロールを果たすモノリシックなアプリケーションの時代には、Sendmail は組織内で電子メールサーバーを稼働するために必要な唯一のアプリケーションと思われるかもしれません。Sendmail は各ユーザーのディレクトリーにメールを送信し、ユーザー用にアウトバウンドメールを送信できるため、技術的にはこれが当てはまります。Sendmail はメールを各ユーザーのディレクトリーにスプールして、ユーザーに送信メールを配信できるからです。ユーザーは通常、POP または IMAP を使用してメッセージをローカルマシンにダウンロードするMUA を使用して電子メールと対話したいです。ユーザーは通常、POP または IMAP を使用する MUA で電子メールとやりとりを行い、ローカルマシンにメッセージをダウンロードする方法を望みます。こうした他のアプリケーションを Sendmail と連動させることは可能ですが、実際、それらが存在する理由は異なり、独立して機能することができます。

Sendmail で設定すべき、また設定できるすべての用途の説明は、本セクションの対象外となります。Sendmail には文字どおり数百におよぶ様々なオプションやルールセットがあるため、Sendmail のあらゆる機能や問題修正方法に関する専門的な資料が多くあります。Sendmail リソースのリストについては、「関連情報」 を参照してください。

本セクションでは、Sendmail でデフォルトでインストールされたファイルを確認し、不要なメール（スパム）を停止する方法や、LDAP (Lightweight Directory Access Protocol)で Sendmail を拡張する方法など、基本的な設定変更を確認します。

27.3.1.2. Sendmail のデフォルトのインストール

Sendmail 実行可能ファイルは /usr/sbin/sendmail です。

Sendmail の長さと詳細な設定ファイルは /etc/mail/sendmail.cf です。sendmail.cf ファイルを直接編集しないでください。Sendmail の設定を変更するには、/etc/mail/sendmail.mc ファイルを編集し、元の /etc/mail/sendmail.cf のバックアップを作成し、以下の代替手段を使用して新しい設定ファイルを生成します。

- /etc/mail に含まれる makefile （すべての -C /etc/mail）を使用して、新しい /etc/mail/sendmail.cf 設定ファイルを作成します。必要に応じて、/etc/mail (db ファイル)で他のすべての生成されたファイルが再生成されます。古い makemap コマンドは引き続き使用できます。make コマンドは、make パッケージがインストールされて いる場合は、sendmail start | restart | reload により自動的に使用されます。

- または、同梱の m4 マクロプロセッサーを使用して、新しい /etc/mail/sendmail.cf を作

成することもできます。

Sendmail の設定に関する詳細は 「Sendmail の一般的な設定変更」 を参照してください。

各種の Sendmail 設定ファイルは、以下を含む /etc/mail/ ディレクトリーにインストールされます。

- access: 送信メールに Sendmail を使用できるシステムを指定します。
- domaintable - ドメイン名のマッピングを指定します。
- local-host-names: ホストのエイリアスを指定します。
- mailertable: 特定のドメインのルーティングを上書きする方法を指定します。
- virtusertable - ドメイン固有のエイリアス形式を指定し、1 台のマシンで複数の仮想ドメインをホストできるようにします。

/etc/mail/ にある設定ファイルの一部は、Sendmail が設定変更を使用できるようにするために、/etc/mail/ の設定ファイルの一部(ドメインテーブル、mailertable、virtusertable など)をデータベースファイルに保存する必要があります。データベースファイルの設定に変更を追加する場合は、以下のコマンドを実行します。

makemap hash /etc/mail/ <name> & lt; /etc/mail/ <name>

ここで、<name > は変換する設定ファイルの名前に置き換えられます。

たとえば、example.com ドメインにすべての電子メールを bob@other-example.com に配信するには、以下の行を virtusertable ファイルに追加します。

```
@example.com     bob@other-example.com
```

変更を完了するには、root で以下のコマンドを使用して virtusertable.db ファイルを更新する必要

があります。

```
makemap hash /etc/mail/virtusertable < /etc/mail/virtusertable
```

これにより、新しい設定を含む更新された virtusertable.db ファイルが作成されます。

27.3.1.3. Sendmail の一般的な設定変更

Sendmail 設定ファイルを変更する場合は、既存のファイルを編集するのではなく、完全に新しい /etc/mail/sendmail.cf ファイルを生成するのが最適です。

注意

sendmail.cf ファイルを変更する前に、バックアップコピーを作成することが推奨されます。

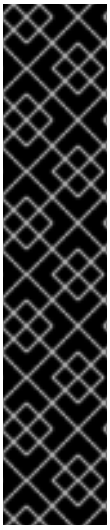
希望する機能を Sendmail に追加するには、root ユーザーとして /etc/mail/sendmail.mc ファイルを編集します。完了したら、以下のコマンドを実行して m4 マクロプロセッサーを使用して新しい sendmail.cf を生成します。

```
m4 /etc/mail/sendmail.mc > /etc/mail/sendmail.cf
```

デフォルトでは、m4 マクロプロセッサーは Sendmail でインストールされますが、m4 パッケージに含まれます。

新しい /etc/mail/sendmail.cf ファイルを作成したら、Sendmail を再起動して変更を反映します。これを行う最も簡単な方法は、以下のコマンドを入力することです。

```
service sendmail restart
```



重要な影響

デフォルトの sendmail.cf ファイルでは、Sendmail はローカルコンピューター以外のホストからのネットワーク接続を受け入れません。Sendmail を他のクライアント用のサーバーとして設定するには、/etc/mail/sendmail.mc ファイルを編集し、DAEMON_OPTIONS ディレクティブの Addr= オプションで指定されているアドレスを 127.0.0.1 からアクティブなネットワークデバイスの IP アドレスに変更するか、行頭に dnl を配置することで DAEMON_OPTIONS ディレクティブをすべてコメントアウトします。完了したら、以下のコマンドを実行して /etc/mail/sendmail.cf を再生成します。

```
m4 /etc/mail/sendmail.mc > /etc/mail/sendmail.cf
```

Red Hat Enterprise Linux に同梱されるデフォルト設定は、ほとんどの SMTP 専用サイトで機能します。ただし、UUCP (UNIX から UNIX Copy)サイトでは機能しません。UUCP メール転送を使用する場合は、/etc/mail/sendmail.mc ファイルを再設定し、新しい /etc/mail/sendmail.cf を生成する必要があります。

/usr/share/sendmail-cf ディレクトリー下のディレクトリーにあるファイルを編集する前に、/usr/share/sendmail-cf/README ファイルを参照してください。これは、/etc/mail/sendmail.cf ファイルの今後の設定に影響を及ぼす可能性があるためです。

27.3.1.4. マスカレーディング

一般的な Sendmail の設定の 1 つとして、1 台のマシンがネットワーク上の全マシンのメールのゲートウェイとして機能するように設定する方法があります。たとえば、ある企業が mail.example.com という名前のマシンですべての電子メールを処理し、すべての送信メールに一貫した返信アドレスを割り当てるとします。

この場合、Sendmail サーバーは、返信アドレスが user@host.example.com ではなく user@example.com となるように、会社ネットワーク上のマシン名をマスカレードする必要があります。

これを行うには、以下の行を /etc/mail/sendmail.mc に追加します。

```
FEATURE(always_add_domain)dnl
FEATURE(`masquerade_entire_domain')dnl
FEATURE(`masquerade_envelope')dnl
FEATURE(`allmasquerade')dnl
MASQUERADE_AS(`bigcorp.com.')dnl
MASQUERADE_DOMAIN(`bigcorp.com.')dnl
MASQUERADE_AS(bigcorp.com)dnl
```

m4 を使用して新しい sendmail.cf を生成した後、この設定は、ネットワーク内からのすべてのメールが bigcorp.com から送信されたかのように表示されます。

27.3.1.5. Spam の停止

電子メールのスパムは、通信を要求したことがないユーザーから受信した、不要な迷惑メールとして定義することができます。これは、破壊的でコストがかかる、広く蔓延したインターネット通信標準の悪用です。

Sendmail を使用すると、迷惑メールの送信に使用されている新たなスパム技術を比較的に簡単にブロックすることができます。さらに、数多くの一般的なスパム手法もデフォルトでブロックします。sendmail で利用可能な主要なアンチスパム機能は、ヘッダーチェック、リレー拒否（バージョン 8.9 からデフォルト）、アクセスデータベースおよび送信者情報チェックです。

たとえば、リレーとも呼ばれる SMTP メッセージの転送は、Sendmail バージョン 8.9 以降デフォルトでは無効になっています。この変更前には、Sendmail はメールホスト(x.edu)に、ある当事者(y.com)からのメッセージを受け入れて別の当事者(z.net)に送信するよう指示しました。しかし、現在は任意のドメインがサーバーを介してメールをリレーするよう Sendmail を設定する必要があります。リレードメインを設定するには、/etc/mail/relay-domains ファイルを編集して Sendmail を再起動します。

ただし、ユーザーがインターネット全体にある他のサーバーからのスパムによる低下が多くなります。このような場合、/etc/mail/access ファイルで利用可能な Sendmail のアクセス制御機能を使用して、不要なホストからの接続を防ぐことができます。以下の例は、このファイルを使用したブロックの方法と Sendmail サーバーへのアクセスを具体的に許可する方法を示しています。

```
badspammer.com       ERROR:550 "Go away and do not spam us anymore"
tux.badspammer.com   OK
10.0                 RELAY
```

この例では、basp ammer.com から送信される電子メールが 550 RFC-821 準拠のエラーコードでブロックされ、メッセージがスパムメーマーに戻されます。tux.badspammer.com サブドメインから送信される電子メールは受け入れられます。最後の行は、10.0.*.* ネットワークから送信された電子メールがメールサーバーを介してリレーできることを示しています。

/etc/mail/access.db はデータベースであるため、makemap を使用して変更を有効にします。これを行うには、root で以下のコマンドを実行します。

```
makemap hash /etc/mail/access < /etc/mail/access
```

メッセージヘッダーの分析により、ヘッダーの内容に基づいてメールを拒否することができます。SMTP サーバーは、メッセージヘッダーに電子メールに関する情報を保存します。メッセージがある MTA から別の MTA に移動すると、各メッセージは他のすべての Received ヘッダーの上に Received ヘッダーに配置されます。ただし、この情報をスパム対策が変更する可能性があることに注意してください。

上記の例は、アクセスの許可や阻止に関する Sendmail が持つ機能のほんの一部です。詳細と例については、/usr/share/sendmail-cf/README を参照してください。

Sendmail は、メールの配信時に Procmail MDA を呼び出すため、SpamAssassin のようなスパムフィルターリングプログラムを使用して、ユーザーに対してスパムを識別してファイルに保存することも可能です。SpamAssassin の使用に関する詳細は、「spam フィルター」 を参照してください。

27.3.1.6. LDAP での Sendmail の使用

LDAP (Lightweight Directory Access Protocol) の使用は、大規模なグループから特定のユーザーに関する特定の情報を検索する非常に迅速かつ強力な方法です。たとえば、LDAP サーバーを使用すると、一般的な企業ディレクトリーから特定のメールアドレスをユーザーの姓で検索できます。このような実装では、LDAP は Sendmail と大きく分離されており、LDAP は階層的なユーザー情報を保存し、Sendmail には事前にアドレスを指定したメールメッセージの LDAP クエリーの結果のみが付与されます。

ただし、Sendmail は LDAP とのはるかに優れた統合をサポートします。この場合、LDAP を使用して、中規模レベルの組織をサポートするさまざまなメールサーバーで、エイリアス や virtusertables などのメンテナーンスされるファイルを置き換えます。つまり、LDAP はメールルーティングレベルを Sendmail およびその個別の設定ファイルから、多くの異なるアプリケーションで活用できる強力な LDAP クラスターに抽象化します。

Sendmail の現行バージョンは LDAP に対応しています。LDAP を使用して Sendmail を拡張するには、最初に OpenLDAP などの LDAP サーバーを実行し、適切に設定します。次に、/etc/mail/sendmail.mc を編集して以下を追加します。

```
LDAPROUTE_DOMAIN('yourdomain.com')dnl
FEATURE('ldap_routing')dnl
```

注記

これは LDAP を使用した非常に基本的な Sendmail の設定にすぎません。この設定は、特に共通の LDAP サーバーを使用するように複数の Sendmail マシンを設定する場合など、LDAP の実装によって大幅に異なる可能性があります。

詳細な LDAP ルーティング設定の指示と例については、/usr/share/sendmail-cf/README を参照してください。

次に、m4 を実行して Sendmail を再起動して、/etc/mail/sendmail.cf ファイルを再作成します。手順は、「Sendmail の一般的な設定変更」 を参照してください。

LDAP の詳細は、28章Lightweight Directory Access Protocol (LDAP) を参照してください。

27.3.2. postfix

当初、IBM のセキュリティーエキスパートであるプログラマーの Wietse Venema 氏によって開発された Postfix は、Sendmail 互換の MTA で、セキュア、高速、かつ容易に設定できるように設計されています。

セキュリティーを向上させるために、Postfix はモジュラー設計を使用します。この場合、特権が限定された小さなプロセスは マスター デーモンにより起動されます。より小さく、権限の低いプロセスは、メール配信の様々な段階に関連する非常に特殊なタスクを実行してルートディレクトリーが変更された環境で稼働し、攻撃の影響を制限します。

Postfix がローカルコンピューター以外のホストからのネットワーク接続を受け入れるよう設定するには、設定ファイルを多少変更するだけでできます。しかし、より複雑なニーズのために、Postfix はさまざまな設定オプションを提供し、サードパーティーは、非常に汎用性が高くフル機能の MTA となるよう追加しています。

Postfix の設定ファイルは人間に解読可能で、250 以上のディレクティブに対応しています。Sendmail とは異なり、変更を反映するためにマクロ処理は必要なく、また最も一般的に使用されるオプションの大部分は、多数のコメントが付いたファイルで説明されています。

重要な影響

Postfix を使用する前に、デフォルトの MTA を Sendmail から Postfix に切り替える必要があります。

27.3.2.1. Postfix のデフォルトインストール

Postfix 実行可能ファイルは /usr/sbin/postfix です。このデーモンは、メール配信の処理に必要なすべての関連プロセスを起動します。

Postfix は設定ファイルを /etc/postfix/ ディレクトリーに保存します。以下は、一般的に使用されるその他のファイルの一覧です。

- access: アクセス制御に使用します。このファイルは、Postfix に接続できるホストを指定します。
- aliases: メールプロトコルで必要な設定可能なリスト。
- main.cf: グローバル Postfix 設定ファイル設定オプションの大部分がこのファイルで指定されています。
- master.cf: メール配信を完了するために Postfix がさまざまなプロセスと対話する方法を指定します。
- transport: 電子メールアドレスをリレーホストにマッピングします。

重要な影響

デフォルトの /etc/postfix/main.cf ファイルでは、Postfix はローカルコンピューター以外のホストからのネットワーク接続を受け入れることができません。Postfix を他のクライアントのサーバーとして設定する方法は、「Postfix の基本設定」 を参照してください。

/etc/postfix/ ディレクトリーのファイル内で一部のオプションを変更する場合は、変更を有効にするために postfix サービスを再起動する必要がある場合があります。これを行う最も簡単な方法は、以下のコマンドを入力することです。

```
service postfix restart
```

27.3.2.2. Postfix の基本設定

デフォルトでは、Postfix はローカルホスト以外のホストからのネットワーク接続を受け付けません。ネットワーク上の他のホストのメール配信を有効にするには、root として以下の手順を実行します。

- vi などのテキストエディターで /etc/postfix/main.cf ファイルを編集します。
- ハッシュマーク(#)を削除して mydomain 行のコメントを解除し、domain.tld をメールサーバーがサービスするドメイン(example.com など)に置き換えます。
- myorigin = $mydomain 行のコメントを解除します。
- myhostname 行のコメントを解除し、host.domain.tld をマシンのホスト名に置き換えます。
- mydestination = $myhostname, localhost.$mydomain 行のコメントを解除します。
- mynetworks 行のコメントを解除し、168.100.189.0/28 を、サーバーに接続可能なホスト用の有効なネットワーク設定に置き換えます。
- inet_interfaces = all 行のコメントを解除します。
- inet_interfaces = localhost 行をコメント化します。
- postfix サービスを再起動します。

これらの手順が完了したら、ホストは配信のため外部の電子メールを受け入れるようになります。

Postfix には様々な設定オプションがあります。Postfix の設定方法を学習する最適な方法の 1 つは、/etc/postfix/main.cf ファイル内のコメントを読むことです。また、Red Hat Enterprise Linux バージョン 5.9 では、Postfix は MySQL マップをサポートします。これにより、Postfix は MySQL データベースを使用し、MySQL データベースに対するさまざまな操作のためにさまざまなルックアップテーブルを設定することができます。たとえば、グローバルメールのリダイレクトを処理する 仮想テーブル、SMTP サーバーへの アクセス を制御するアクセステーブル、およびシステム全体のメールリダイレクトを管理するための aliases テーブルです。設定の詳細と例、および LDAP および

SpamAssassin インテグレーションに関する情報を含むその他のその他のリソースは、http://www.postfix.org/ でオンラインで利用できます。

27.3.3. Fetchmail

Fetchmail は、リモートサーバーから電子メールを取得してローカルの MTA に配信する MTA です。多くのユーザーは、リモートサーバー上にあるメッセージをダウンロードするプロセスと、MUA で電子メールを読み取り、整理するプロセスを別々にする機能性を評価してます。ダイアルアップユーザーのニーズを念頭に設計された Fetchmail は、POP3 や IMAP などのプロトコルを使用して、メールスプールファイルに接続し、すべての電子メールメッセージを迅速にダウンロードします。また、必要に応じて、電子メールメッセージを SMTP サーバーに転送することもできます。

Fetchmail は、各ユーザーのホームディレクトリー内の .fetchmailrc ファイルを使用して、ユーザーごとに設定されます。

Fetchmail は .fetchmailrc ファイルの設定を使用して、リモートサーバー上にある電子メールを確認し、ダウンロードします。ローカル MTA を使用して電子メールを正しいユーザーのスプールファイルに配置し、ローカルマシンのポート 25 に配信します。Procmail が利用できる場合は起動して電子メールをフィルターし、MUA が読み込むことができるようにメールボックスに配置します。

27.3.3.1. Fetchmail の設定オプション

Fetchmail の実行時にすべての必要なオプションをコマンドラインで渡し、リモートサーバーで電子メールを確認することは可能ですが、.fetchmailrc ファイルを使用する方がはるかに簡単です。希望の設定オプションを .fetchmailrc ファイルに配置し、それらのオプションが fetchmail コマンドが実行されるたびに使用されるようにします。Fetchmail の実行時にオプションを上書きしたい場合は、コマンドラインでそのオプションを指定します。

ユーザーの .fetchmailrc ファイルには、3 つのクラスの設定オプションが含まれています。

- *グローバルオプション: プログラムの動作を制御する、または電子メールを確認する全接続の設定を提供する指示を Fetchmail に指定します。*
- **server options**: *ポーリングされるサーバーに必要な情報（ホスト名など）を指定します。また、チェックするポートやタイムアウトするまで待機する秒数など、特定の電子メールサーバーの設定などです。こうしたオプションは、該当するサーバーを使用する全ユーザーに影響を及ぼします。*
- *ユーザーオプション: 指定したメールサーバーを使用して電子メールの認証およびチェックに必要なユーザー名とパスワードなどの情報が含まれています。*

グローバルオプションは .fetchmailrc ファイルの上部に表示され、その後に 1 つ以上のサーバーオプションが表示されます。各オプションは Fetchmail がチェックする異なるメールサーバーを指定します。ユーザーオプションは、そのメールサーバーをチェックする各ユーザーアカウントのサーバーオプションに従います。サーバーオプションと同様に、複数のユーザーオプションを指定することで特定のサーバーでの使用、同一サーバー上の複数の電子メールアカウントの確認を行うことができます。

サーバーオプションは、サーバー情報の前に ポーリング または skip などの特別なオプションの動詞を使用して .fetchmailrc ファイルでサービスを呼び出します。poll アクションは Fetchmail の実行時にこのサーバーオプションを使用するように Fetchmail に指示します。これは、指定したユーザーオプションを使用して電子メールをチェックします。ただし、skip アクションの後にあるサーバーオプションは、Fetchmail が呼び出されたときにこのサーバーのホスト名が指定されていない限り確認されません。skip オプションは、特定して呼び出された時にスキップされたサーバーのみをチェックし、現在稼働中の設定には影響しないため、.fetchmailrc の設定をテストする場合に便利です。

サンプルの .fetchmailrc ファイルは以下の例のようになります。

```
set postmaster "user1"
set bouncemail
 poll pop.domain.com proto pop3
  user 'user1' there with password 'secret' is user1 here
 poll mail.domain2.com
  user 'user5' there with password 'secret2' is user1 here
  user 'user7' there with password 'secret3' is user1 here
```

この例では、グローバルオプションにより、最終手段としてユーザーに電子メールが送信されるように指定されており(postmaster オプション)、すべての電子メールエラーは送信者ではなく、ポストマスターに送信されます(bouncemail オプション)。set アクションは、この行にグローバルオプションが含まれていることを Fetchmail に伝えます。次に、2 つのメールサーバーが指定されており、もう 1 つは POP3 を使用してチェックするように設定されます。2 番目のサーバーオプションを使用して 2 つのユーザーを確認しますが、ユーザーで見つかったすべての電子メールは user1 のメールスプールに送信されます。これにより、1 つの MUA 受信トレイに表示され、複数のサーバーで複数のクラスを確認できます。各ユーザーの固有の情報は、user アクションで始まります。

注記

ユーザーはパスワードを .fetchmailrc ファイルに配置する必要はありません。with password '<password>' セクション を省略すると、Fetchmail は起動時にパスワードを要求します。

Fetchmail には、グローバルオプション、サーバーオプション、ローカルオプションが多数あります。これらの多くのオプションは、ほとんど使用されないか、非常に特殊な状況にのみ適用されま

す。fetchmail の man ページには、各オプションの詳細が記載されていますが、最も一般的なものがここに記載されています。

27.3.3.2. グローバルオプション

グローバルオプションは、set アクションの後に、1 行ずつ配置する必要があります。

- *daemon <seconds >: Fetchmail がバックグラウンドに残るデーモンモードを指定します。&lt ;seconds& gt; を、Fetchmail がサーバーをポーリングするまで待機する秒数に置き換えます。*
- *postmaster: 配信に問題が発生した場合にローカルユーザーがメールを送信するように指定します。*
- *syslog: エラーおよびステータスメッセージのログファイルを指定します。デフォルトでは、これは /var/log/maillog です。*

27.3.3.3. サーバーオプション

サーバーオプションは、ポーリング または スキップ アクションの後に .fetchmailrc の各行に配置する必要があります。

- *auth & lt;auth-type& gt; - <auth-type& gt; を、使用する認証タイプに置き換えます。デフォルトでは パスワード 認証が使用されますが、一部のプロトコルは kerberos_v5、kerberos_v4、および ssh などの他のタイプの認証をサポートします。any 認証タイプを使用する場合、Fetchmail は、パスワードを必要としない方法を最初に試み、次にパスワードをマスクする方法を試み、最後にサーバーに暗号化されていないパスワードの送信を試みます。*
- *interval & lt;number& gt; - 指定されたサーバーを < number > 毎にポーリングし、設定されたすべてのサーバーで電子メールをチェックします。このオプションは、通常のユーザーがほとんどメッセージを受信しない電子メールサーバーに使用されます。*
- *port & lt;port-number&gt ; - &lt ;port-number> をポート番号に置き換えます。この値は、指定されたプロトコルのデフォルトのポート番号を上書きします。*
- *proto & lt;protocol > - < protocol > を、サーバー上のメッセージを確認するときに使用する pop3 や imap などのプロトコルに置き換えます。*

- *timeout & lt;seconds& gt;: & lt;seconds > を、Fetchmail が接続の試行をやめてからサーバーが非アクティブになる秒数に置き換えます。この値が設定されていない場合、デフォルトの 300 秒が想定されます。*

27.3.3.4. ユーザーオプション

ユーザーオプションは、サーバーオプションの下の各行に置かれる場合と、サーバーオプションと同じ行に置かれる場合があります。いずれの場合も、定義されるオプションは user オプション（以下で定義）に従う必要があります。

- *fetchall: 既読メッセージを含め Fetchmail がキュー内の全メッセージをダウンロードするように命令します。デフォルトでは、Fetchmail は新規メッセージのみをダウンロードするようになっています。*

- *fetchlimit & lt;number& gt;: &lt ;number> を、停止する前に取得するメッセージの数に置き換えます。*

- *flush: 新規メッセージを取得する前に、キューにあるすべての既読メッセージを削除します。*

- *limit & lt;max-number-bytes > - < max-number-bytes > を Fetchmail が取得した時にメッセージが許可される最大サイズ（バイト単位）に置き換えます。このオプションでは低速のネットワークリンクが提供されるため、サイズが大きいメッセージのダウンロードに時間がかかりすぎる場合に有用です。*

- *Password ' <password> ' - & lt;password& gt; をユーザーのパスワードに置き換えます。*

- *preconnect " <command> " - < command > を、ユーザーのメッセージを取得する前に実行するコマンドに置き換えます。*

- *postconnect " <command> " - < command > を、ユーザーのメッセージを取得した後に実行するコマンドに置き換えます。*

- *ssl: SSL 暗号化を有効にします。*

- *sslproto: 許可された SSL プロトコルまたは TLS プロトコルを定義します。設定可能な値は SSL2、SSL3、SSL23、および TLS1 です。ただし、POODLE: SSLv3 脆弱性(CVE-2014-3566)により、このオプションを TLS1 に設定してください。*

- *user " <username> ": &lt ;username> を Fetchmail がメッセージの取得に使用するユーザー名に置き換えます。このオプションは、他のすべてのユーザーオプションの前に付ける必要があります。*

27.3.3.5. Fetchmail のコマンドオプション

fetchmail コマンドの実行時にコマンドラインで使用される Fetchmail オプションの大半は、.fetchmailrc 設定オプションを反映します。この方法では、Fetchmail は設定ファイルの有無を問わず使用できます。これらのオプションは、.fetchmailrc ファイルに残る方が簡単なため、ほとんどのユーザーはコマンドラインでは使用されません。

fetchmail コマンドを、特定の目的の他のオプションを指定して実行したい場合もあります。コマンドラインで指定されるオプションが設定ファイルオプションを上書きするため、コマンドオプションを使用して、エラーの原因となっている .fetchmailrc 設定を一時的に上書きすることが可能です。

27.3.3.6. 情報提供またはデバッグのオプション

fetchmail コマンドの後に使用されるオプションの一部は、重要な情報を提供する場合があります。

- *--configdump: .fetchmailrc および Fetchmail のデフォルト値からの情報に基づいて、可能なオプションをすべて表示します。このオプションを使用すると、どのユーザーの電子メールも取得されません。*

- *-s: Fetchmail をサイレントモードで実行し、fetchmail コマンドの後にエラー以外のメッセージが表示されないようにします。*

- *-v: Fetchmail を verbose モードで実行し、Fetchmail とリモートの電子メールサーバー間の通信をすべて表示します。*

- *-v: 詳細なバージョン情報の表示、グローバルオプションの一覧表示、電子メールプロトコルや認証方法など、各ユーザーと使用する設定の表示を行います。このオプションを使用すると、どのユーザーの電子メールも取得されません。*

27.3.3.7. 特殊なオプション

これらのオプションは、.fetchmailrc ファイルにあるデフォルト値を上書きする際に役に立つ場合があります。

- -a: Fetchmail は、新規または既読を問わず、すべてのメッセージをリモートの電子メールサーバーからダウンロードします。デフォルトでは、Fetchmail は新規メッセージのみをダウンロードします。

- -k: Fetchmail はメッセージをダウンロードした後、リモートの電子メールサーバー上にメッセージを残します。このオプションを使用すると、メッセージをダウンロード後に削除するデフォルトの動作は上書きされます。

- -l & lt;max-number-bytes& gt;: Fetchmail は特定のサイズを超えるメッセージはダウンロードせず、リモートの電子メールサーバー上に残します。

- --quit: Fetchmail デーモンのプロセスを終了します。

その他のコマンドおよび .fetchmailrc オプションは、fetchmail の man ページを参照してください。

27.4. メール転送エージェント (MTA) の設定

メール送信には、MTA （メール 転送エージェント ）が不可欠です。Evolution、Thunderbird、Mutt などの Mail User Agent (MUA)を使用してメールの読み取りと作成を行います。ユーザーが MUA から電子メールを送信すると、メッセージは MTA に渡されます。MTA は一連の MTA を通じて、メッセージが送信先に届くまで送信します。

ユーザーがシステムからメールを送信する予定がなくても、一部の自動化されたタスクまたはシステムプログラムは /bin/mail コマンドを使用して、ログメッセージを含む電子メールをローカルシステムの root ユーザーに送信する場合があります。

Red Hat Enterprise Linux 5 は、Sendmail、Postfix、Exim の 3 つの MTA を提供します。3 つすべてがインストールされている場合、sendmail がデフォルトの MTA になります。Mail Transport Agent Switcher では、sendmail、postfix、または exim をシステムのデフォルト MTA として選択できます。

Mail Transport Agent Switcher プログラムのテキストベースのバージョンを使用するに

は、system-switch-mail RPM パッケージをインストールする必要があります。グラフィカルバージョンを使用する場合は、system-switch-mail-gnome パッケージもインストールする必要があります。

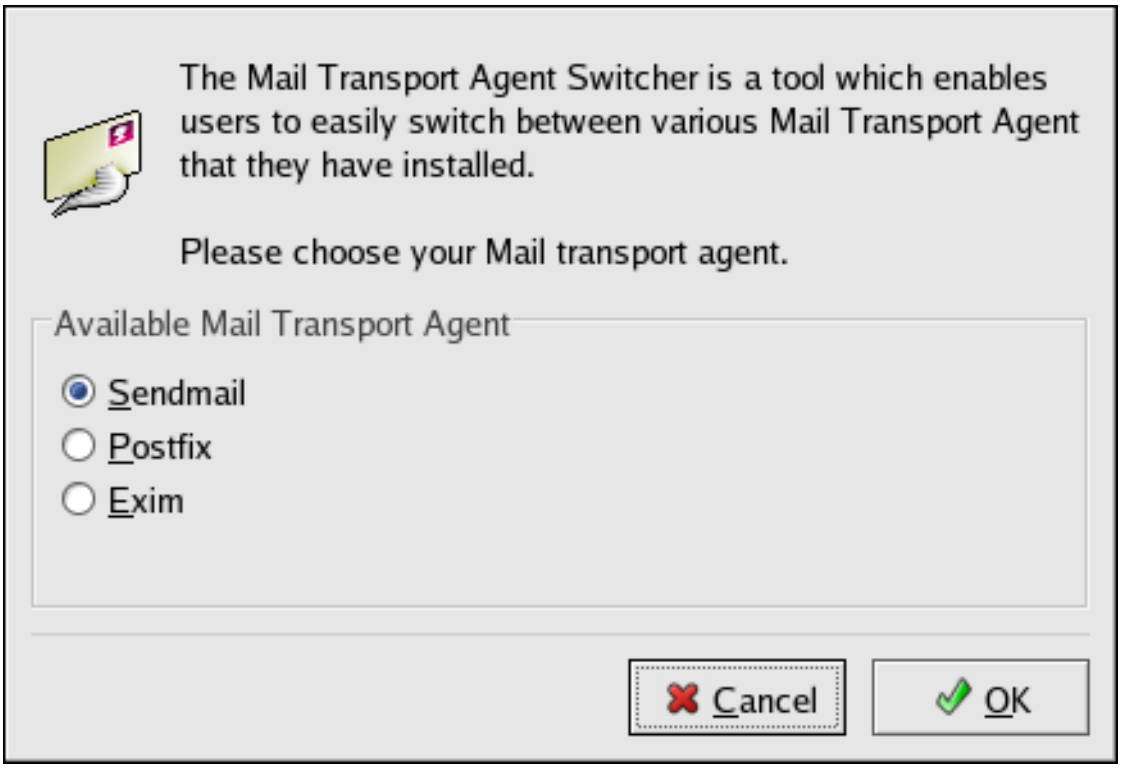
注記

RPM パッケージのインストールに関する詳細は、パートII「パッケージ管理」 を参照してください。

Mail Transport Agent Switcher を起動するには、System （パネルのメインメニュー）> Administration > Mail Transport Agent Switcher を選択するか、シェルプロンプトでコマンド system-switch-mail を入力します（例：XTerm または GNOME ターミナル）。

このプログラムは、X Window System が実行されているかどうかを自動的に検出します。実行中の場合、プログラムは 図27.1「メール転送エージェントスイッチャー」 のようにグラフィカルモードで起動します。X が検出されない場合は、テキストモードで起動します。Mail Transport Agent Switcher をテキストモードで実行させるには、system-switch-mail-nox コマンドを使用します。

図27.1 メール転送エージェントスイッチャー



[D]

OK を選択して MTA を変更すると、選択したメールデーモンが起動時に開始するように有効にな

り、選択したメールデーモンは起動時に起動しないように無効になります。選択したメールデーモンが起動し、その他のメールデーモンは停止されるため、変更が即座に行われます。

27.5. メール配信エージェント (MDA)

Red Hat Enterprise Linux には、Procmail と mail の 2 つの主要 MDA が含まれています。どちらのアプリケーションも LDA と見なされ、MTA のスプールファイルからユーザーのメールボックスにメールを移動します。ただし、Procmail の方が堅牢なフィルターリングシステムを提供します。

このセクションでは、Procmail についてのみ詳しく説明します。mail コマンドの詳細は、man ページを参照してください。

ローカルホストのメールスプールファイルに電子メールが置かれると、Procmail が配信とフィルターリングを行います。Procmail は強力な上、システムリソースの使用が低いため、幅広く利用されています。Procmail は、電子メールクライアントアプリケーションが読み取る電子メールを配信するという重要なロールを果たします。

Procmail は、様々な方法で呼び出すことができます。MTA が電子メールをメールスプールファイルの中に置くと常に Procmail が起動します。次に、Procmail は電子メールを MUA のためにフィルターリング、ファイル保存して、終了します。別の方法としては、メッセージを受信すると常に Procmail を実行するように MUA を設定し、メッセージが正しいメールボックスに移動するようにできます。デフォルトでは、ユーザーのホームディレクトリーに /etc/procmailrc または .procmailrc ファイル（別名 rc ファイル）が存在すると、MTA が新しいメッセージを受信するたびに Procmail が呼び出されます。

Procmail が電子メールメッセージに対応するかどうかは、メッセージが rc ファイルの特定の条件または レシピ と一致するかどうかによって異なります。あるメッセージが任意のレシピと適合する場合、電子メールは特定のファイルに置かれるか削除され、それ以外は処理されます。

Procmail が起動すると、電子メールメッセージを読み取り、ヘッダー情報から本文を切り離します。次に、Procmail は /etc/procmailrc s ディレクトリーの /etc/procmailrcs ファイルで、デフォルトのシステム全体の Procmail 環境変数とレシピを探します。その後、Procmail はユーザーのホームディレクトリーで .procmailrc ファイルを検索します。多くのユーザーは、Procmail 用に追加の rc ファイルを作成します。このファイルは、ホームディレクトリーの .procmailrc ファイル内で参照されます。

デフォルトでは、システム全体の rc ファイルが /etc/ ディレクトリーに存在せず、ユーザーのホームディレクトリーに .procmailrc ファイルが存在しません。そのため、Procmail を使用するには、各ユーザーが特定の環境変数とルールで .procmailrc ファイルを構築する必要があります。

27.5.1. Procmail の設定

Procmail の設定ファイルには、重要な環境変数が含まれています。これらの変数は、並べ替えするメッセージ、およびどのレシピとも適合しないメッセージの処理を指定します。

これらの環境変数は通常、以下の形式で .procmailrc の開始時に表示されます。

```
<env-variable>="<value>"
```

この例では、& lt;env-variable&gt ; は変数の名前に、< value> は 変数を定義します。

ほとんどの Procmail ユーザーが使用していない環境変数が多くあります。また、重要な環境変数の多くがデフォルト値で定義されています。重要な環境変数の多くは、既にデフォルト値で定義されています。大抵の場合は、以下のような変数が使用されます。

- *DEFAULT: どのレシピにも一致しないメッセージが配置されるデフォルトのメールボックスを設定します。*

  デフォルトの DEFAULT 値は、$ORGMAIL と同じです。

- *INCLUDERC: 照合するメッセージのレシピを含む追加の rc ファイルを指定します。これにより、Procmail レシピの一覧は、スパムのブロックや電子メール一覧の管理など、異なるロールを満たす個々のファイルに分割されます。このファイルは、ユーザーの .procmailrc ファイルのコメント文字を使用して、オンまたはオンにすることができます。*

  たとえば、ユーザーの .procmailrc ファイル内の行は以下のようになります。

  ```
  MAILDIR=$HOME/Msgs
  INCLUDERC=$MAILDIR/lists.rc
  INCLUDERC=$MAILDIR/spam.rc
  ```

  ユーザーが電子メールリストの Procmail フィルターをオフにしたいが、スパム制御をそのまま残す場合は、最初の INCLUDERC 行をハッシュマーク文字(#)でコメントアウトします。

- *LOCKSLEEP: Procmail が特定のロックファイルの使用を試みる間隔を秒単位で設定します。デフォルトは 8 秒です。*

- *LOCKTIMEOUT: ロックファイルが最後に変更された後、Procmail が古くて削除可能であるとみなすまでに経過する必要のある時間を秒単位で設定します。デフォルトは 1024 秒です。*

- *LOGFILE: Procmail の情報やエラーメッセージが書き込まれるファイルです。*

- *MAILDIR: Procmail の現在の作業用ディレクトリーを設定します。設定されると、他の Procmail のパスはすべてこのディレクトリーに対する相対パスになります。*

- *ORGMAIL: 元のメールボックス、またはデフォルトまたはレシピで必要な場所にメッセージを配置できない場合にメッセージを配置する別の場所を指定します。*

  デフォルトでは、/var/spool/mail/$LOGNAME の値が使用されます。

- *SUSPEND: スワップ領域など必要なリソースが利用できない場合に Procmail が一時停止する時間を秒単位で設定します。*

- *SWITCHRC: 追加の Procmail レシピを含む外部ファイルを指定できます。ただし、これは INCLUDERC オプションと同様です。ただし、レシピのチェックは参照された設定ファイルで実際に停止され、SWITCHRCの指定ファイルのレシピのみが使用される点が異なります。*

- *VERBOSE: Procmail が詳細情報をログに記録します。このオプションはデバッグに役立ちます。*

その他の重要な環境変数は、ログイン名である LOGNAME、ホームディレクトリーの場所である HOME、デフォルトのシェルである SHELL などのシェルからプルされます。

すべての環境変数およびデフォルト値に関する包括的な説明は、procmailrc man ページを参照してください。

27.5.2. Procmail レシピ

多くの場合、新規ユーザーが Procmail の使用法を学習するにあたって最も難しいと感じるのは、レシピの構築です。一部のエクステントでは、レシピは 正規表現 を使用してメッセージ照合を行うため、一致する文字列の条件を指定するために使用される特定の形式になります。ただ、正規表現の構築はそれほど難しくなく、読んで理解することも簡単です。その上、Procmail のレシピを書く方法は、

正規表現にかかわらず一貫性があるため、例を使って学習すると簡単です。Procmail のレシピの例は、「レシピの例」 を参照してください。

Procmail レシピは以下の形式を使用します:

```
:0<flags>: <lockfile-name>
* <special-condition-character>
   <condition-1>
* <special-condition-character>
   <condition-2>
* <special-condition-character>
   <condition-N>
   <special-action-character>
   <action-to-perform>
```

Procmail レシピの最初の 2 文字は、コロンとゼロです。ゼロの後に様々なフラグを追加して、Procmail がレシピを処理する方法を制御します。< flags> セクションの後ろにコロンを 付けると、このメッセージに対してロックファイルが作成されることを指定します。ロックファイルが作成されると、< lockfile-name> を置き換えて名前を指定できます 。

レシピには、メッセージと適合させる様々な条件を追加できます。条件がない場合は、すべてのメッセージがレシピと適合することになります。正規表現は、メッセージ照合を容易にするために、一部の条件で使用されます。複数の条件を使用する場合は、アクションが実行されるためにはすべてが適合しなければなりません。条件は、レシピの 1 行目に設定されたフラグに基づいてチェックされます。* 文字の後に置かれたオプションの特殊文字は、さらに条件を制御できます。

&lt ;action-to-perform > は、メッセージが条件の 1 つと一致する場合に実行するアクションを指定します。1 つのレシピに指定できるアクションは 1 つのみとなります。多くの場合、メールボックスの名前がここで使用され、適合するメッセージをファイルに誘導し、電子メールを効果的に並べ替えます。特別なアクションの文字は、アクションが指定される前に使用することもできます。詳細は、「特別な条件とアクション」 を参照してください。

27.5.2.1. 配信と非配信レシピ

レシピが特定のメッセージと一致する場合に使用されるアクションで、配信レシピと 非 配信 レシピと見なされるかどうかが判断されます。配信レシピには、ファイルへのメッセージの書き込み、別のプログラムへのメッセージ送信、別の電子メールアドレスへのメッセージ転送などのアクションが含まれています。非配信レシピは、ネストされたブロック などの他のアクションをカバーします。ネストされたブロックは、中括弧 { } に含まれるアクションのセットで、レシピの条件に一致するメッセージで実行されます。ネストされたブロックは、互いにネストさせることができるため、メッセージに対するアクションを特定して実行するにあたっての制御力が強化されます。

メッセージが配信レシピと適合すると、Procmail は指定されたアクションを実行し、その他のレシピとメッセージとの比較を停止します。非配信レシピと適合するメッセージの場合は、他のレシピに

対する照合は継続されます。

27.5.2.2. フラグ

フラグは、レシピの条件がメッセージと照合される方法、またはメッセージと照合されるかどうかを決定するために不可欠です。一般的に使用されるフラグは以下のとおりです。

- a: A や a のフラグのない以前のレシピもこのメッセージに適合する場合にのみ、このレシピが使用されることを指定します。
- A: A または a のフラグが付いた以前のレシピもこのメッセージと 一致し、正常に完了した場合にのみ、このレシピが使用されることを指定します。
- b: メッセージの本文を解析し、一致する条件を検索します。
- b: ファイルへのメッセージの書き込みや転送など、結果として生じるアクションでボディーを使用します。これはデフォルトの動作です。
- c - メールのカーボンコピーを生成します。必要なアクションをメッセージで実行し、メッセージのコピーは rc ファイルで引き続き処理できるため、レシピの配信に役立ちます。
- D: egrep の比較で大文字と小文字を区別します。デフォルトでは、照合プロセスでは大文字と小文字を区別していません。
- e: A フラグと類似していますが、レシピ内の条件は、すぐに E フラグなしのレシピの前のメッセージと一致しない場合にのみ、メッセージと比較されます。これは else アクションと類似しています。
- e: 直前のレシピで指定されたアクションが失敗した場合のみ、レシピがメッセージと比較されます。
- f: フィルターとしてパイプを使用します。
- H: メッセージのヘッダーを解析し、一致する条件を検索します。これはデフォルトで発生します。

- *h: 結果として生じるアクションでヘッダーを使用します。これはデフォルトの動作です。*

- *w: Procmail に対して、指定されたフィルターまたはプログラムが終了するのを待ち、メッセージがフィルターされたとみなす前に成功したかどうかを報告するよう指示します。*

- *w: プログラム障害のメッセージが抑制されている点を除いて w と同じです。*

その他のフラグの詳細な一覧は、procmailrc man ページを参照してください。

27.5.2.3. ローカルロックファイルの指定

ロックファイルは、Procmail で複数のプロセスが 1 つのメッセージを同時に変更しないようにするために非常に役立ちます。ローカルロックファイルを指定するには、レシピの 1 行目にフラグの後にコロン(:)を追加します。これにより、宛先ファイル名と LOCKEXT グローバル環境変数で設定されたものに基づいてローカルロックファイルが作成されます。

別の方法としては、このレシピで使用するローカルロックファイルの名前をコロンの後に指定します。

27.5.2.4. 特別な条件とアクション

Procmail レシピの条件とアクションの前に使用される特殊文字により、解釈の仕方が変わります。

*以下の文字は、レシピの条件行頭にある * 文字の後に使用できます。*

- *!: 条件の行では、この文字により条件が反転し、条件がメッセージに一致しない場合にのみ一致が発生するようになります。*

- *& lt;: メッセージが、指定されているバイト数に収まっているかどうかを確認します。*

- & gt; : メッセージが、指定されているバイト数を超えているかどうかを確認します。

以下の文字は、特別なアクションを実行するために使用されます。

- !: アクションの行では、この文字は、メッセージを指定されたメールアドレスに転送するように Procmail に指示します。

- $: rc ファイルで以前に設定された変数を参照します。多くの場合は、さまざまなレシピによって参照される共通のメールボックスを設定するために使用されます。

- | - 指定したプログラムを開始し、メッセージを処理します。

- { および }: 適合するメッセージに適用する追加のレシピを格納するために使用されるネストされたブロックを構築します。

アクションの行頭に特殊文字を使用しない場合、Procmail はアクションの行がメッセージを書き込むためのメールボックスを指定していると仮定します。

27.5.2.5. レシピの例

Procmail は極めて柔軟性の高いプログラムですが、この柔軟性が原因で、新規ユーザーが Procmail のレシピを一から作成するのが難しい場合があります。

Procmail レシピの条件を構築するスキルを向上させる最適な方法は、正規表現をしっかり理解し、他の人が構築した多くの例を参照することから始まります。正規表現に関する詳細な説明は、本セクションでは扱いません。Procmail レシピの構造と役立つ Procmail のサンプルレシピは、インターネット上のさまざまな場所（例： http://www.iki.fi/era/procmail/links.html）にあります。正規表現の適切な使用と調整方法は、これらのレシピ例を参照してください。また、基本的な正規表現ルールの概要は grep の man ページにあります。

以下にあげる簡単な例は、Procmail のレシピの基本構造を記載しており、構造をさらに複雑にするための基盤を示しています。

以下の例に示すように、基本的なレシピには条件さえも含まれていません。

-

```
:0:
new-mail.spool
```

最初の行は、ローカルロックファイルを作成することを指定しますが、名前を指定しないため、Procmail は宛先ファイル名を使用して、LOCKEXT 環境変数に指定された値を追加します。条件が指定されていないため、すべてのメッセージがこのレシピと一致し、MAILDIR 環境変数で指定されたディレクトリー内にある new-mail.spool という単一の spool ファイルに配置されます。その後、MUA はこのファイルでメッセージを表示できます。

このような基本レシピは、rc ファイルの末尾に置かれ、メッセージをデフォルトの場所に送ります。

以下の例では、特定の電子メールアドレスからのメッセージを照合して、削除します。

```
:0
* ^From: spammer@domain.com
/dev/null
```

この例では、spammer@domain.com から送信されたメッセージはすべて /dev/null デバイスに送信され、削除されます。

注意

永続的な削除のために /dev/null にメッセージを送信する前に、ルールが意図したとおりに機能していることを確認してください。レシピが間違えて目的以外のメッセージを対象にすると、それらのメッセージは消えてしまい、ルールのトラブルシューティングが困難になります。

より良い解決策は、レシピのアクションを特別なメールボックスにポイントすることです。これは、誤検出を探すために時折確認できます。メッセージを誤って一致していないと満足したら、メールボックスを削除し、メッセージを /dev/null に送信するようアクションに指示します。

以下のレシピでは、特定のメーリングリストから送信された電子メールを取得して、特定のフォルダーに配置します。

```
:0:
* ^(From|CC|To).*tux-lug
tuxlug
```

tux-lug@domain.com メーリングリストから送信されたメッセージはすべて、MUA の tuxlug メールボックスに自動的に配置されます。From、CC、または To 行にメーリングリストのメールアドレスがある場合に、この例の条件がメッセージと一致することに注意してください。

さらに詳しい強力なレシピについては、「関連情報」 の Procmail に関する多くのオンライン資料を参照してください。

27.5.2.6. spam フィルター

Procmail は、新規の電子メールを受信すると Sendmail、Postfix、Fetchmail によって呼び出されるため、スパム対策の強力なツールとして使用できます。

これは、Procmail が SpamAssassin と併用された場合に特に有効です。これらの 2 つのアプリケーションを併用すると、スパムメールを迅速に特定して、並び替えまたは破棄できます。

SpamAssassin は、ヘッダー分析、テキスト分析、ブラックリスト、スパム追跡データベース、自己学習型 Bayesian スパム分析を使用して、迅速かつ正確にスパムの特定とタグ付けを行います。

ローカルユーザーが SpamAssassin を使用する最も簡単な方法は、~/.procmailrc ファイルの最上部付近に以下の行を配置することです。

```
INCLUDERC=/etc/mail/spamassassin/spamassassin-default.rc
```

/etc/mail/spamassassin/spamassassin-default.rc には、すべての受信メールに対して SpamAssassin を有効にする簡単な Procmail ルールが含まれています。電子メールがスパムであると判断された場合には、ヘッダー内でタグ付けされ、タイトルの先頭には以下のようなパターンが追加されます。

```
*****SPAM*****
```

電子メールのメッセージ本文にも、スパム診断の理由となった要素の継続的な記録が先頭に追加されます。

スパムとしてタグ付けされた電子メールをファイル保存するには、以下と同様のルールを使用する

ことができます。

```
:0 Hw
* ^X-Spam-Status: Yes
spam
```

このルールにより、スパムとしてヘッダーにタグ付けされた電子メールはすべて、spam と呼ばれるメールボックスにファイルとして保存されます。

SpamAssassin は Perl スクリプトであるため、ビジー状態のサーバーではバイナリー SpamAssassin デーモン(spamd)およびクライアントアプリケーション(spamc)を使用する必要がある場合があります。ただし、SpamAssassin をこのように設定するには、ホストへの root アクセスが必要です。

spamd デーモンを起動するには、root で以下のコマンドを入力します。

```
service spamassassin start
```

システムの起動時に SpamAssassin デーモンを起動するには、Services Configuration Tool (system-config-services)などの initscript ユーティリティーを使用して spamassassin サービスをオンにします。initscript ユーティリティーの詳細は、18章 を参照してください。

Procmail が Perl スクリプトの代わりに SpamAssassin クライアントアプリケーションを使用するように設定するには、~/.procmailrc ファイルの最上部付近に以下の行を追加します。システム全体の設定の場合は、/etc/procmailrc に配置します。

```
INCLUDERC=/etc/mail/spamassassin/spamassassin-spamc.rc
```

27.6. メールユーザーエージェント

Red Hat Enterprise Linux では、メールプログラムスコアがあります。フル機能のグラフィカル電子メールクライアントプログラム(Ximian Evolution など)と、mutt などのテキストベースの電子メールプログラムがあります。

このセクションの残りの部分では、クライアントとサーバー間の通信のセキュリティー保護に重点を置いています。

27.6.1. 通信のセキュリティー保護

Red Hat Enterprise Linux に備わっている MustRunAss は、Ximian Evolution や mutt など、SSL で暗号化された電子メールセッションを提供します。

暗号化されていないネットワークを流れる他のサービスと同様に、ユーザー名、パスワード、メッセージ全体などの重要な電子メール情報は、ネットワーク上のユーザーが傍受して表示できます。また、標準の POP プロトコルおよび IMAP プロトコルは、認証情報を暗号化せずに渡すため、ユーザー名とパスワードはネットワーク経由で渡されるため、攻撃者がユーザーアカウントにアクセスできる可能性があります。

27.6.1.1. セキュアな電子メールクライアント

リモートサーバー上の電子メールを確認するように設計されている Linux MUA のほとんどは、SSL 暗号化に対応しています。メールの取得時に SSL を使用するには、メールのクライアントとサーバーの両方で SSL を有効にする必要があります。

SSL はクライアント側で簡単に有効にできます。多くの場合、MUA の設定ウィンドウでボタンをクリックするか、MUA の設定ファイルのオプションを使用して実行できます。セキュアな IMAP および POP には、MUA がメッセージの認証およびダウンロードに使用する既知のポート番号(993 と 995)があります。

27.6.1.2. 電子メールクライアントの通信のセキュリティー保護

メールサーバー上の IMAP および POP ユーザーに SSL 暗号化を提供することは簡単です。

最初に SSL 証明書を作成します。これには、SSL 証明書の 認証局 (CA)に適用するか、自己署名証明書を作成する方法の 2 つの方法があります。

注意

自己署名証明書は、テスト目的のみで使用することをお勧めします。実稼働環境で使用するサーバーは、CA によって付与された SSL 証明書を使用する必要があります。

IMAP の自己署名 SSL 証明書を作成するには、/etc/pki/tls/certs/ ディレクトリーに移動し、root で以下のコマンドを入力します。

■

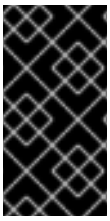
```
rm -f cyrus-imapd.pem make cyrus-imapd.pem
```

すべての質問に答えて、プロセスを完了してください。

POP の自己署名 SSL 証明書を作成するには、/etc/pki/tls/certs/ ディレクトリーに移動し、root で以下のコマンドを入力します。

```
rm -f ipop3d.pem make ipop3d.pem
```

もう一度、すべての質問に回答してプロセスを完了します。



重要な影響

各 make コマンドを実行する前に、デフォルトの imapd.pem ファイルおよび ipop3d.pem ファイルを必ず削除してください。

/etc/imapd.conf ファイルに以下の行を追加して、セキュアでない SSL プロトコルを無効にします。

```
tls_cipher_list: TLSv1+HIGH:!aNull:@STRENGTH
```

POODLE SSL 脆弱性 (CVE-2014-3566) の影響を受けないようにするためです。詳細は、POODLE: SSLv3 脆弱性(CVE-2014-3566)を参照してください。

完了したら、/sbin/service cyrus-imapd start コマンドを実行して、Cyrus IMAP および POP デーモンを起動します。

別の方法として、stunnel コマンドを標準の非セキュア IMAP プロトコルおよび POP プロトコルの SSL 暗号化ラッパーとして使用することもできます。ただし、Cyrus 設定ファイル /etc/cyrus.conf で IMAPS および POP3 を無効にする必要があります。これを行うには、imaps および pop3s を含む行をコメントアウトし、cyrus-imapd サービスを再起動します。

stunnel プログラムは、Red Hat Enterprise Linux に含まれる外部の OpenSSL ライブラリーを使用して、強力な暗号を提供し、接続を保護します。SSL 証明書を取得するために CA に適用することが推奨されますが、自己署名証明書を作成することも可能です。

自己署名の SSL 証明書を作成するには、/etc/pki/tls/certs/ ディレクトリーに移動し、以下のコマンドを入力します。

```
make stunnel.pem
```

もう一度、すべての質問に回答してプロセスを完了します。

証明書を入手したら、stunnel の設定ファイルを作成します。これは、すべての行がオプションまたはサービス定義の開始を指定するテキストファイルです。コメントと空の行をファイルに残して、コメントがセミコロンで始まる読みやすさを向上させることもできます。

stunnel RPM パッケージには、設定ファイルを保存できる /etc/stunnel/ ディレクトリーが含まれています。stunnel はファイル名やその拡張子の特別な形式を必要としませんが、/etc/stunnel/stunnel.conf を使用してください。以下のコンテンツは、stunnel をセキュアな IMAP および POP の TLS ラッパーとして設定します。

```
cert = /etc/pki/tls/certs/stunnel.pem
; Allow only TLS, thus avoiding SSL
options = NO_SSLv2
options = NO_SSLv3
chroot = /var/run/stunnel
setuid = nobody
setgid = nobody
pid = /stunnel.pid
socket = l:TCP_NODELAY=1
socket = r:TCP_NODELAY=1

[pop3s]
accept  = 995
connect = 110

[imaps]
accept  = 993
connect = 143
```

最後に、stunnel を起動します。

```
stunnel /etc/stunnel/stunnel.conf
```

stunnel の使用方法に関する詳細は、stunnel の man ページを参照するか、/usr/share/doc/stunnel - <version-number > / ディレクトリーのドキュメントを参照してください。< version-number > は stunnel のバージョン番号に置き換えてください。

27.7. 関連情報

以下は、電子メールアプリケーションに関する補足のドキュメントの一覧です。

27.7.1. インストールされているドキュメント

- Sendmail の設定に関する情報は、sendmail パッケージおよび sendmail -cf パッケージに含まれています。
  - /usr/share/sendmail-cf/README: m4、Sendmail のファイルの場所、サポートされるメーラー、強化機能へのアクセス方法などに関する情報が含まれています。

  さらに、sendmail および aliases の man ページには、Sendmail のさまざまなオプションと Sendmail /etc/mail/aliases ファイルの適切な設定に関する役立つ情報が含まれています。

- /usr/share/doc/postfix- <version-number>: Postfix の設定方法に関する多くの情報が含まれています。&lt ;version-number> を Postfix のバージョン番号に置き換えてください。

- /usr/share/doc/fetchmail- <version-number >: FEATURES ファイルの Fetchmail 機能の完全一覧と、入門的な FAQ ドキュメントが含まれています。&lt ;version-number> を Fetchmail のバージョン番号に置き換えてください。

- /usr/share/doc/procmail- <version-number >: Procmail の概要を提供する README ファイル、すべてのプログラム機能を調べる FEATURES ファイル、設定に関する多くの質問に対する回答が含まれる FAQ ファイルが含まれます。&lt ;version-number> を Procmail のバージョン番号に置き換えてください。

  Procmail の仕組みや新しいレシピの作成方法を学習する場合は、以下にあげる Procmail の man ページが非常に役立ちます。

  - Procmail: Procmail の仕組みと電子メールのフィルターリングに必要な手順を概説します。
  - procmailrc: レシピの構築に使用される rc ファイル形式を説明します。
  - procmailex: 実環境向けの役立つ Procmail レシピを多数紹介します。

- procmailsc - 特定のレシピとメッセージを適合するために Procmail が使用する重みのスコアリング手法を説明します。

- /usr/share/doc/spamassassin- <version-number> /: SpamAssassin に関する多くの情報が含まれています。&lt ;version-number> を、spamassassin パッケージのバージョン番号に置き換えてください。

27.7.2. 便利な Web サイト

- http://www.sendmail.org/: Sendmail の機能の詳細、ドキュメント、設定例を提供します。

- http://www.sendmail.com/: Sendmail に関する記事、概要、および記事が含まれています。これには、利用可能な多くのオプションの幅広いビューが含まれます。

- http://www.postfix.org/: Postfix プロジェクトのホームページには、Postfix に関する多くの情報が含まれています。メーリングリストは、特に情報検索に役立ちます。

- http://fetchmail.berlios.de/: Fetchmail のホームページ、オンラインマニュアル、および詳細な FAQ です。

- http://www.procmail.org/: Procmail のホームページでは、Procmail 専用の各種メーリングリストへのリンクと、さまざまな FAQ ドキュメントが記載されています。

- http://partmaps.org/era/procmail/mini-faq.html - 優れた Procmail FAQ で、トラブルシューティングのヒント、ファイルロックの詳細、ワイルドカード文字の使用について説明します。

- http://www.uwasa.fi/~ts/info/proctips.html - Procmail の使用がはるかに簡単にする多数のヒントが含まれています。.procmailrc ファイルをテストし、Procmail スクリプトレットを使用して、特定のアクションを実行する必要があるかどうかを判断する手順が含まれています。

- http://www.spamassassin.org/: SpamAssassin プロジェクトの公式サイトです。

27.7.3. 関連書籍

- 『Sendmail Milters: A Guide for Fighting Spam』 *by Bryan Costales and Marcia Flynt; Addison-Wesley:* メールフィルターのカスタマイズに役立つ優れた *Sendmail* ガイドです。

- 『Sendmail』 *(Bryan Costales、Eric Allman et al)、O'Reilly & Associates - Delivermail と Sendmail* のオリジナル作成者のサポートを受けている優れた *Sendmail* リファレンスです。

- 『Spam の削除：Geoff Mulligan による電子メール処理とフィルターリング』、*Addison-Wesley Publishing Company - Sendmail* や *Procmail* などの確立されたツールを使用してスパムの問題を管理する電子メール管理者が使用するさまざまな方法を調べるボリュームです。

- 『Internet Email Protocols: A Developer's Guide』 *by Kevin Johnson; Addison-Wesley Publishing Company -* 主要な電子メールプロトコルと、それが提供するセキュリティーについて非常に詳細なレビューを提供します。

- 『Managing IMAP』 *by Dianna Mullet and Kevin Mullet; O'Reilly & Associates - IMAP* サーバーの設定に必要な手順について詳しく説明します。

第28章 LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL (LDAP)

LDAP(Lightweight Directory Access Protocol)は、ネットワーク上で一元的に保存された情報にアクセスするために使用されるオープンプロトコルのセットです。これは、ディレクトリー共有の X.500 標準に基づいていますが、複雑でリソースを大量に消費します。このため、LDAP はX.500 Lite と呼ばれることもあります。 X.500 標準は、名前、アドレス、電話番号などの情報が含まれる階層およびカテゴリー化された情報が含まれるディレクトリーです。

X.500 と同様に、LDAP はディレクトリーを使用して階層的な方法で情報を整理します。これらのディレクトリーはさまざまな情報を保存し、Network Information Service (NIS)と同様に使用でき、LDAP 対応のネットワーク上の任意のマシンからアカウントにアクセスできるようにすることもできます。

多くの場合、LDAP は仮想電話ディレクトリーとして使用され、ユーザーは他のユーザーの連絡先情報に簡単にアクセスできます。ただし、LDAP は従来の phone ディレクトリーよりも柔軟性が高いため、世界全体で他の LDAP サーバーに対してクエリーを実行し、情報のアドホックのグローバルリポジトリーを提供することができるためです。ただし、現在、ユニバーサリ、政府部門、プライベート企業など、個々の組織内で LDAP を使用するのが一般的です。

LDAP はクライアント/サーバーシステムです。サーバーはさまざまなデータベースを使用してディレクトリーを保存できます。LDAP クライアントアプリケーションが LDAP サーバーに接続すると、ディレクトリーをクエリーするか、修正を試行できます。クエリーが発生した場合には、サーバーはクエリーにローカルで応答するか、応答のある LDAP サーバーにクエリーを参照できます。クライアントアプリケーションが LDAP ディレクトリー内の情報を変更しようとしている場合、サーバーは、ユーザーが変更を行い、情報を追加または更新するパーミッションを持っていることを確認します。

本章では、LDAPv2 プロトコルおよび LDAPv3 プロトコルのオープンソース実装である OpenLDAP 2.0 の設定および使用について説明します。

28.1. LDAP を使用する理由

LDAP を使用する主な利点は、組織全体の情報を中央リポジトリーに統合できることです。たとえば、組織内の各グループのユーザーリストを管理するのではなく、LDAP をネットワーク上のどこからでもアクセスできる中央ディレクトリーとして使用できます。また、LDAP は Transport Layer Security (TLS)をサポートしているため、機密データは prying eyes から保護できます。

重要

Resolution for POODLE SSLv3.0 vulnerability (CVE-2014-3566) for components that do not allow SSLv3 to be disabled via configuration settings(設定から SSLv3 を無効にできないコンポーネントで POODLE SSLv3.0 脆弱性 (CVE-2014-3566) を解決する方法) に説明されている脆弱性により、Red Hat はセキュリティー保護のために SSLv3 に依存しないことを推奨しています。OpenLDAP は、SSLv3 を効果的に無効にできるようにする設定パラメーターを提供しないシステムコンポーネントの 1 つです。リスクを軽減するには、stunnel コマンドを使用してセキュアなトンネルを提供し、SSLv3 の使用から stunnel を無効にすることが推奨されます。

LDAP は、ディレクトリーを格納する多くのバックエンドデータベースもサポートします。これにより、管理者は、サーバーが解読する情報の種類に最も適したデータベースを柔軟にデプロイできます。LDAP には明確に定義されたクライアントアプリケーションプログラミングインターフェイス (API)があるため、LDAP 対応アプリケーションの数は多く、数量と品質が増加します。

28.1.1. OpenLDAP の機能

OpenLDAP には、多くの重要な機能が含まれています。

- LDAPv3 サポート: OpenLDAP は、その他の改善の中で Simple Authentication and Security Layer (SASL)および TLS (Transport Layer Security)をサポートします。LDAPv2 は LDAP をよりセキュアにするように設計されています。
- IPv6 サポート: OpenLDAP は、次世代インターネットプロトコルバージョン 6 をサポートします。
- LDAP Over IPC: OpenLDAP はプロセス間通信(IPC)を使用してシステム内で通信できます。これにより、ネットワーク上で通信する必要がなくなるため、セキュリティーが強化されます。
- 更新された C API: プログラマーが LDAP ディレクトリーサーバーに接続し、使用する方法を改良します。
- LDIFv1 サポート: LDAP Data Interchange Format (LDIF)バージョン 1 への完全準拠を提供します。
- Enhanced Stand-Alone LDAP サーバー： 更新されたアクセス制御システム、スレッドプール、より良いツールなどが含まれています。

28.2. LDAP の用語

LDAP に関する議論には、LDAP 固有の用語の基本的な理解が必要です。

- エントリー: LDAP ディレクトリー内の単一のユニット。各エントリーは、一意の識別名 (DN)で識別 されます。

- 属性 - エントリーに直接関連付けられた情報。たとえば、組織は LDAP エントリーとして表示できます。組織に関連付けられた属性には、fax 番号、アドレスなどが含まれる場合があります。また、ユーザーの電話番号やメールアドレスなどの一般的な属性を使用して、LDAP ディレクトリーのエントリーとして表示できます。

  一部の属性は必須ですが、他の属性は任意です。objectclass 定義は、各エントリーに必要な属性を設定します。objectClass 定義は、/etc/openldap/schema/ ディレクトリーにあるさまざまなスキーマファイルにあります。詳細は、「/etc/openldap/schema/ ディレクトリー」を参照してください。

  属性とそれに対応する値のアサーションは、RDN (Relative Distinguished Name)とも呼ばれます。RDN はエントリーごとに一意ですが、DN はグローバルに一意です。

- LDIF: LDAP データ交換形式 (LDIF)は LDAP エントリーの ASCII テキスト表現です。LDAP サーバーへのデータのインポートに使用されるファイルは LDIF 形式である必要があります。LDIF エントリーは以下の例のようになります。

```
[<id>] dn: <distinguished name>
<attrtype>: <attrvalue>
<attrtype>: <attrvalue>
<attrtype>: <attrvalue>
```

  各エントリーには 、必要に応じて任意の数の <attrtype>: <attrvalue> ペア を含めることができます。空白行は、エントリーの最後を示します。

注意

この情報 を使用するには、すべての < attrtype > and <attrvalue > ペアを対応するスキーマファイルに定義する 必要があり ます。

< および > で囲まれた値は変数で、新しい LDAP エントリーが作成されるたびに設定できます。 ただし、このルールは < id > には適用されません。&lt ;id> は、エントリーの編集に使用されるアプリケーションによって決定される番号です。

28.3. OPENLDAP デーモンとユーティリティー

OpenLDAP ライブラリーおよびツールのスイートは、以下のパッケージに含まれています。

- *OpenLDAP: OpenLDAP サーバーおよびクライアントアプリケーションの実行に必要なライブラリーが含まれます。*
- *openldap-clients: LDAP サーバーのディレクトリーを表示および変更するコマンドラインツールが含まれます。*
- *openldap-servers: LDAP サーバーの設定および実行に必要なサーバーおよびその他のユーティリティーが含まれます。*

openldap-servers パッケージには、Standalone LDAP Daemon (/usr/sbin/slapd)と Standalone LDAP Update Replication Daemon (/usr/sbin/slurpd)の 2 つのサーバーが含まれています。

slapd デーモンはスタンドアロンの LDAP サーバーであり、slurpd デーモンは、ある LDAP サーバーからネットワーク上の他の LDAP サーバーに変更を同期するために使用されます。slurpd デーモンは、複数の LDAP サーバーを処理する場合にのみ使用されます。

管理タスクを実行するには、openldap-servers パッケージにより、以下のユーティリティーが /usr/sbin/ ディレクトリーにインストールされます。

- *slapadd: LDIF ファイルから LDAP ディレクトリーにエントリーを追加します。たとえ*

ば、/usr/sbin/slapadd -l ldif-input コマンドは、新しいエントリーを含む LDIF ファイル ldif-input で読み取ります。

重要な影響

root ユーザーのみが /usr/sbin/slapadd を使用できます。ただし、ディレクトリーサーバーは ldap ユーザーとして実行されます。したがって、ディレクトリーサーバーは slapadd が作成したファイルを変更できません。この問題を修正するには、slapadd を使用してから、以下のコマンドを入力します。

```
chown -R ldap /var/lib/ldap
```

- *slapcat - デフォルト形式の LDAP ディレクトリーからプルし、Sleepycat Software の Berkeley DB システムに保存し、LDIF ファイルに保存します。たとえば、コマンド /usr/sbin/slapcat -l ldif-output は、LDAP ディレクトリーからのエントリーを含む ldif-output という LDIF ファイルを出力します。*

- *slapindex - 現在のコンテンツに基づいて slapd ディレクトリーを再インデックスします。このツールは、/etc/openldap/slapd.conf 内のインデックスオプションが変更されるたびに実行する必要があります。*

- *slappasswd: slapd 設定ファイルの ldapmodify または rootpw 値で使用する暗号化されたユーザーパスワード値(/etc/openldap/slapd.conf)を生成します。/usr/sbin/slappasswd コマンドを実行してパスワードを作成します。*

WARNING

slapadd、slapcat、または slapindex を使用する前に、/sbin/service ldap stop コマンドを実行して slapd を停止する必要があります。それ以外の場合は、LDAP ディレクトリーの整合性が危険にさらされます。

これらのユーティリティーの使用方法は、それぞれの man ページを参照してください。

openldap-clients パッケージは、LDAP ディレクトリーのエントリーの追加、変更、および削除に使用される /usr/bin/ にツールをインストールします。これらのツールには、以下が含まれます。

- *ldapadd: ファイルまたは標準入力で入力を受け入れることで LDAP ディレクトリーにエントリーを追加します。ldapadd は、ldapmodify -a へのハードリンクです。*

- *ldapdelete: シェルプロンプトまたはファイルを介してユーザー入力を受け入れることで、LDAP ディレクトリーからエントリーを削除します。*

- *ldapmodify: LDAP ディレクトリーのエントリーを変更し、ファイルまたは標準入力による入力を受け入れます。*

- *ldappasswd: LDAP ユーザーのパスワードを設定します。*

- *ldapsearch: シェルプロンプトを使用して LDAP ディレクトリー内のエントリーを検索します。*

- *ldapcompare: LDAP サーバーへの接続を開き、指定のパラメーターを使用して比較を行い、実行します。*

- *ldapwhoami: LDAP サーバーへの接続を開き、バインドし、whoami 操作を実行します。*

- *ldapmodrdn: LDAP サーバーへの接続を開き、エントリーの RDN を変更し、変更します。*

ldapsearch の例外により、各ユーティリティーは、LDAP ディレクトリー内で変更する各エントリーに対してコマンドを入力するのではなく、変更を含むファイルを参照することで簡単に使用できます。このようなファイルの形式は、各ユーティリティーの man ページで説明されています。

28.3.1. NSS、PAM、および LDAP

OpenLDAP パッケージに加えて、Red Hat Enterprise Linux には nss_ldap と呼ばれるパッケージが含まれており、このパッケージにより、LDAP の Linux 環境や他の UNIX 環境への統合が強化されます。

nss_ldap パッケージは、以下のモジュールを提供します。&lt ;version> は、使用中の libnss_ldap のバージョンを参照します。

- /lib/libnss_ldap-<version>.so
- /lib/security/pam_ldap.so

nss_ldap パッケージは、Itanium アーキテクチャーまたは AMD64 アーキテクチャーに以下のモジュールを提供します。

- /lib64/libnss_ldap-<version>.so
- /lib64/security/pam_ldap.so

libnss_ldap- <version > .so モジュールを使用すると、アプリケーションは glibc の Nameservice Switch (NSS)インターフェイスを介して LDAP ディレクトリーを使用してユーザー、グループ、ホスト、およびその他の情報を検索できます。NSS を使用すると、アプリケーションは NIS ネームサービスとフラット認証ファイルとともに LDAP を使用して認証できます。

pam_ldap モジュールにより、PAM 対応のアプリケーションは LDAP ディレクトリーに保存されている情報を使用してユーザーを認証できます。PAM 対応アプリケーションには、コンソールログイン、POP および IMAP メールサーバー、および Samba が含まれます。ネットワークに LDAP サーバーをデプロイすることで、これらのアプリケーションはすべて同じユーザー ID とパスワードの組み合わせを使用して認証できるため、管理を大幅に簡素化できます。

PAM の設定に関する詳細は、「PAM (プラグ可能な認証モジュール)」 および PAM の man ページを参照してください。

28.3.2. PHP4、LDAP、および Apache HTTP Server

Red Hat Enterprise Linux には、PHP サーバー側のスクリプト言語の LDAP モジュールを含むパッケージが含まれています。

php-ldap パッケージは、/usr/lib/php4/ldap.so モジュールを介して PHP4 HTML 組み込みスクリプト言語に LDAP サポートを追加します。このモジュールにより、PHP4 スクリプトは LDAP ディレクトリーに保存されている情報にアクセスできます。

Red Hat Enterprise Linux には、Apache HTTP Server 用の mod_authz_ldap モジュールが同梱されています。このモジュールは、サブジェクトの識別名の短い形式とクライアント SSL 証明書の発

行者を使用して、LDAP ディレクトリー内のユーザーの識別名を決定します。また、ユーザーの LDAP ディレクトリーエントリーの属性に基づいてユーザーを作成したり、アセットのユーザーおよびグループ権限に基づいてアセットへのアクセスを判断したり、パスワードの期限が切れたユーザーのアクセスを拒否したりすることもできます。mod_authz_ldap モジュールを使用する場合は、mod_ssl モジュールが必要です。

重要な影響

mod_authz_ldap モジュールは、暗号化されたパスワードハッシュを使用して LDAP ディレクトリーに対してユーザーを認証しません。この機能は、Red Hat Enterprise Linux に含まれていない実験的な mod_auth_ldap モジュールにより提供されます。このモジュールのステータスの詳細は、Apache Software Foundation の Web サイト(http://www.apache.org/)を参照してください。

28.3.3. LDAP クライアントアプリケーション

ディレクトリーの作成および変更に対応するグラフィカル LDAP クライアントを利用できますが、Red Hat Enterprise Linux には同梱されていません。このようなアプリケーションの 1 つが LDAP ブラウザー/編集 - http://www.iit.edu/~gawojar/ldap/ でオンラインで利用可能な Java ベースのツールです。

他の LDAP クライアントは、ディレクトリーが読み取り専用としてアクセスします。これらは、組織全体の情報を参照しますが、変更はありません。このようなアプリケーションの例としては、Sendmail、Mozilla、Gnome Meeting、および Evolution があります。

28.4. OPENLDAP 設定ファイル

OpenLDAP 設定ファイルは /etc/openldap/ ディレクトリーにインストールされます。以下は、最も重要なディレクトリーおよびファイルを強調表示する簡単な一覧です。

- /etc/openldap/ldap.conf: これは、ldapsearch、ldapadd、Sendmail、Evolution、Gnome Meeting などの OpenLDAP ライブラリーを使用するすべてのクライアント アプリケーションの設定ファイルです。
- /etc/openldap/slapd.conf: これは、slapd デーモンの設定ファイルです。詳細は、「Editing /etc/openldap/slapd.conf」 を参照してください。
- /etc/openldap/schema/ ディレクトリー : このサブディレクトリーには、slapd デーモンが使用するスキーマが含まれます。詳細は、「/etc/openldap/schema/ ディレクトリー」 を参照してください。

注記

nss_ldap パッケージがインストールされている場合は、/etc/ldap.conf という名前のファイルが作成されます。このファイルは、nss_ldap パッケージが提供する PAM モジュールおよび NSS モジュールによって使用されます。詳細は、「OpenLDAP を使用してシステムを認証するためのシステムの設定」 を参照してください。

28.5. /ETC/OPENLDAP/SCHEMA/ ディレクトリー

/etc/openldap/schema/ ディレクトリーは、以前は slapd.at.conf ファイルおよび slapd.oc.conf ファイルにある LDAP 定義を保持します。/etc/openldap/schema/redhat/ ディレクトリーは、Red Hat Enterprise Linux 用に Red Hat が配信するカスタマイズされたスキーマを保持します。

すべての 属性構文定義 と objectclass 定義 が、異なるスキーマファイルに配置されるようになりました。以下の例に示すように、さまざまなスキーマファイルは、include 行を使用して /etc/openldap/slapd.conf で参照されます。

```
include  /etc/openldap/schema/core.schema
include  /etc/openldap/schema/cosine.schema
include  /etc/openldap/schema/inetorgperson.schema
include  /etc/openldap/schema/nis.schema
include  /etc/openldap/schema/rfc822-MailMember.schema
include  /etc/openldap/schema/redhat/autofs.schema
```

注意

OpenLDAP によってインストールされたスキーマファイルで定義されたスキーマアイテムを変更しないでください。

OpenLDAP で使用されるスキーマを拡張して、デフォルトのスキーマファイルをガイドとして使用して、追加の属性タイプとオブジェクトクラスをサポートすることができます。これを行うには、/etc/openldap/schema/ ディレクトリーに local.schema ファイルを作成します。デフォルトの include schema 行の下に以下の行を追加して、slapd.conf 内でこの新しいスキーマを参照します。

```
include  /etc/openldap/schema/local.schema
```

次に、local.schema ファイル内の新しい属性タイプとオブジェクトクラスを定義します。多くの組織は、デフォルトでインストールされているスキーマファイルから既存の属性タイプを使用し、新しい

オブジェクトクラスを local.schema ファイルに追加します。

特定の特別な要件に一致するようにスキーマを拡張することは、本章の範囲外となります。詳細は、http://www.openldap.org/doc/admin/schema.html を参照してください。

28.6. OPENLDAP 設定の概要

本セクションでは、OpenLDAP ディレクトリーをインストールおよび設定するための簡単な概要を説明します。詳細は、以下の URL を参照してください。

- http://www.openldap.org/doc/admin/quickstart.html: OpenLDAP Web サイトの『Quick-Start Guide』
- http://www.tldp.org/HOWTO/LDAP-HOWTO/index.html : Linux ドキュメントプロジェクトの 『LDAP Linux HOWTO』 です。

LDAP サーバーを作成する基本的な手順は以下のとおりです。

1. openldap、openldap-servers、および openldap-clients RPM をインストールします。
2. /etc/openldap/slapd.conf ファイルを編集し、LDAP ドメインおよびサーバーを指定します。詳細は、「Editing /etc/openldap/slapd.conf」 を参照してください。
3. コマンドで slapd を起動します。

```
service ldap start
```

LDAP の設定後に、chkconfig、/usr/sbin/ntsysv、または Services Configuration Tool を使用して、システムの起動時に LDAP が起動するように設定します。サービスの設定に関する詳細は、18章 を参照してください。

4. ldapadd を使用して LDAP ディレクトリーにエントリーを追加します。

5. ldapsearch を使用して、slapd が情報に正しくアクセスしているかどうかを判断します。

6. この時点で、LDAP ディレクトリーが適切に機能し、LDAP 対応のアプリケーションで設定できます。

28.6.1. Editing /etc/openldap/slapd.conf

slapd LDAP サーバーを使用するには、設定ファイル /etc/openldap/slapd.conf を変更して、正しいドメインおよびサーバーを指定します。

接尾辞 の行は、LDAP サーバーが情報を提供するドメインに名前を付け、以下から変更する必要があります。

```
suffix        "dc=your-domain,dc=com"
```

完全修飾ドメイン名を反映するように編集します。以下に例を示します。

```
suffix        "dc=example,dc=com"
```

rootdn エントリーは、LDAP ディレクトリーの操作に設定されたアクセス制御または管理制限パラメーターで無制限のユーザーの識別名(DN)です。rootdn ユーザーは、LDAP ディレクトリーの root ユーザーとして考えることができます。設定ファイルで、以下の例のように、rootdn 行をデフォルト値から変更します。

```
rootdn        "cn=root,dc=example,dc=com"
```

ネットワーク経由で LDAP ディレクトリーを入力する場合は、rootpw 行を変更します。デフォルト値を暗号化されたパスワード文字列に置き換えます。暗号化されたパスワード文字列を作成するには、以下のコマンドを入力します。

```
slappasswd
```

プロンプトが表示されたら、を入力してパスワードを再入力します。プログラムは、生成される暗号化されたパスワードをシェルプロンプトに出力します。

次に、rootpw 行のいずれかで、新しく作成された暗号化されたパスワードを /etc/openldap/slapd.conf にコピーし、ハッシュマーク(#)を削除します。

完了すると、この行は以下の例のようになります。

```
rootpw {SSHA}vv2y+i6V6esazrIv70xSSnNAJE18bb2u
```

WARNING

TLS 暗号化が有効でない限り、/etc/openldap/slapd.conf に指定された rootpw ディレクティブを含む LDAP パスワードは、暗号化されてい ないネットワークを介して送信されます。

TLS 暗号化を有効にするには、/etc/openldap/slapd.conf のコメントを確認し、slapd.conf の man ページを参照してください。

セキュリティーを強化するために、rootpw ディレクティブは、LDAP ディレクトリーの前にハッシュマーク(#)を追加してコメントアウトする必要があります。

/usr/sbin/slapadd コマンドラインツールをローカルで使用して LDAP ディレクトリーを入力する場合は、rootpw ディレクティブを使用する必要はありません。

重要な影響

/usr/sbin/slapadd を使用できるのは、root ユーザーのみです。ただし、ディレクトリーサーバーは ldap ユーザーとして実行されます。したがって、ディレクトリーサーバーは、slapadd が作成したファイルを変更できません。この問題を修正するには、slapadd を使用してから、以下のコマンドを入力します。

```
chown -R ldap /var/lib/ldap
```

28.7. OPENLDAP を使用してシステムを認証するためのシステムの設定

本セクションでは、OpenLDAP ユーザー認証を設定する方法の概要を説明します。OpenLDAP のエキスパートがない限り、ここに記載されているドキュメントよりも多くのドキュメントが必要になります。詳細は、「関連情報」 に記載されている参考資料を参照してください。

必要な LDAP パッケージをインストールします。

まず、適切なパッケージが LDAP サーバーと LDAP クライアントマシンの両方にインストールされていることを確認します。LDAP サーバーには openldap-servers パッケージが必要です。

openldap パッケージ、 openldap -clients パッケージ、および nss_ldap パッケージは、すべての LDAP クライアントマシンにインストールする必要があります。

設定ファイルを編集します。

- *サーバーで、LDAP サーバーの /etc/openldap/slapd.conf ファイルを編集して、組織の詳細と一致するようにします。slapd.conf の編集方法は、「Editing /etc/openldap/slapd.conf」 を参照してください。*

- *クライアントマシンで、/etc/ldap.conf と /etc/openldap/ldap.conf の両方に適切なサーバーが含まれ、組織のベース情報を検索する必要があります。*

  これを行うには、グラフィカル 認証設定ツール(system-config-authentication)を実行し、User Information タブで Enable LDAP Support を選択します。

  これらのファイルを手動で編集することもできます。

- *クライアントマシンで、LDAP を使用するように /etc/nsswitch.conf を編集する必要があります。*

  これを行うには、Authentication Configuration Tool (system-config-authentication)を実行し、User Information タブで Enable LDAP Support を選択します。

  /etc/nsswitch.conf を手動で編集する場合は、ldap を適切な行に追加します。

  以下に例を示します。

```
passwd: files ldap
shadow: files ldap
group: files ldap
```

-

28.7.1. PAM および LDAP

標準の PAM 対応アプリケーションが認証に LDAP を使用するには、Authentication Configuration Tool (system-config-authentication)を実行し、Authentication タブで Enable LDAP Support を選択します。PAM の設定に関する詳細は、「PAM (プラグ可能な認証モジュール)」 および PAM の man ページを参照してください。

28.7.2. 以前の認証情報の LDAP 形式への移行

/usr/share/openldap/migration/ ディレクトリーには、認証情報を LDAP 形式に移行するシェルおよび Perl スクリプトのセットが含まれます。

注記

これらのスクリプトを使用するには、システムに Perl がインストールされている必要があります。

まず、正しいドメインを反映するように migrate_common.ph ファイルを変更します。デフォルトの DNS ドメインはデフォルト値から以下のように変更する必要があります。

```
$DEFAULT_MAIL_DOMAIN = "example";
```

デフォルトのベースも、以下のように変更する必要があります。

```
$DEFAULT_BASE = "dc=example,dc=com";
```

ユーザーデータベースを LDAP で読み取り可能な形式に移行するジョブは、同じディレクトリーにインストールされた移行スクリプトのグループに分類されます。表28.1「LDAP 移行スクリプト」 を使用して、ユーザーデータベースを移行するために実行するスクリプトを決定します。

既存のネームサービスに基づいて適切なスクリプトを実行します。

/usr/share/openldap/migration/ ディレクトリーの README ファイルおよび migration-tools.txt ファイルは、情報の移行方法の詳細を提供します。

表28.1 LDAP 移行スクリプト

既存のネームサービス	LDAP が稼働しているか？	使用するスクリプト
/etc フラットファイル	はい	migrate_all_online.sh
/etc フラットファイル	いいえ	migrate_all_offline.sh
NetInfo	はい	migrate_all_netinfo_online.sh
NetInfo	いいえ	migrate_all_netinfo_offline.sh
NIS (YP)	はい	migrate_all_nis_online.sh
NIS (YP)	いいえ	migrate_all_nis_offline.sh

28.8. 以前のリリースからのディレクトリーの移行

Red Hat Enterprise Linux では、OpenLDAP は Sleepycat Software の Berkeley DB システムをディレクトリーのディスク上のストレージ形式として使用します。以前のバージョンの OpenLDAP は GNU Database Manager (gdbm)を使用していました。このため、Red Hat Enterprise Linux 5.10 に LDAP 実装をアップグレードする前に、元の LDAP データをアップグレード前に最初にエクスポートしてから再インポートする必要があります。これには、以下の手順を実行します。

1. オペレーティングシステムをアップグレードする前に、コマンド /usr/sbin/slapcat -l ldif-output を実行します。これにより、LDAP ディレクトリーからのエントリーを含む ldif-output という LDIF ファイルが出力されます。

2. オペレーティングシステムをアップグレードします。LDIF ファイルを含むパーティションを再フォーマットしないように注意してください。

3. コマンド /usr/sbin/slapadd -l ldif-output を実行して、LDAP ディレクトリーをアップグレードされた Berkeley DB 形式に再インポートします。

28.9. 関連情報

以下のリソースは、LDAP に関する追加情報を提供します。システムで LDAP を設定する前に、特に OpenLDAP Web サイトおよび LDAP HOWTO を確認することを強く推奨します。

28.9.1. インストールされているドキュメント

- /usr/share/docs/openldap- <versionnumber> / ディレクトリー：一般的な README ドキュメントとその他の情報が含まれています。

- LDAP 関連の man ページ：LDAP に関連するさまざまなアプリケーションおよび設定ファイルに関する man ページが多数あります。以下は、より重要な man ページの一部の一覧です。

  クライアントアプリケーション

  - man ldapadd: LDAP ディレクトリーにエントリーを追加する方法を説明します。
  - man ldapdelete: LDAP ディレクトリー内のエントリーを削除する方法を説明します。
  - man ldapmodify: LDAP ディレクトリー内のエントリーを変更する方法を説明します。
  - man ldapsearch: LDAP ディレクトリー内のエントリーを検索する方法を説明します。
  - man ldappasswd: LDAP ユーザーのパスワードを設定または変更する方法を説明します。
  - man ldapcompare - ldapcompare ツールの使用方法が説明されています。
  - man ldapwhoami - ldapwhoami ツールの使用方法について説明しています。
  - man ldapmodrdn - エントリーの RDN を変更する方法を説明します。

  サーバーアプリケーション

  - man slapd: LDAP サーバーのコマンドラインオプションを説明しています。

- man slurpd: LDAP レプリケーションサーバーのコマンドラインオプションを説明しています。

管理アプリケーション

- man slapadd: slapd データベースにエントリーを追加するために使用されるコマンドラインオプションを説明しています。

- man slapcat - slapd データベースから LDIF ファイルを生成するために使用されるコマンドラインオプションを説明しています。

- man slapindex - slapd データベースの内容に基づいてインデックスを再生成するために使用されるコマンドラインオプションを説明しています。

- man slappasswd: LDAP ディレクトリーのユーザーパスワードを生成するために使用されるコマンドラインオプションを説明しています。

設定ファイル

- man ldap.conf: LDAP クライアントの設定ファイル内で使用できるフォーマットおよびオプションを説明しています。

- man slapd.conf - LDAP サーバーアプリケーション(slapd および slurpd)と LDAP 管理ツール(slapadd、slapcat、および slapindex)の両方で参照される設定ファイル内で利用可能な形式とオプションを説明しています。

28.9.2. 便利な Web サイト

- http://www.openldap.org/ - OpenLDAP プロジェクトのホーム。この Web サイトには、OpenLDAP の設定に関する情報と、今後のロードマップとバージョン変更に関する情報が含まれています。

- http://www.padl.com/ - nss_ldap および pam_ldap の開発者は、その他の便利な LDAP ツールの開発者です。

- *http://www.kingsmountain.com/ldapRoadmap.shtml - Jeff Hodges' LDAP Road Map には、複数の便利な FAQ へのリンクと、LDAP プロトコルに関する注意事項が含まれています。*

- *http://www.ldapman.org/articles/: ディレクトリーツリーの設計およびディレクトリー構造のカスタマイズ方法など、LDAP の概要を提供する記事です。*

28.9.3. 関連書籍

- *『OpenLDAP by John』 Terpstra and Benjamin Coles; Prentice Hall.*

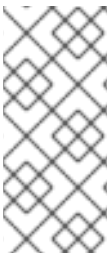
- *Mark Wilcox による 『LDAP の実装』、 Wrox Press, Inc.*

- *『Understanding and Deploying LDAP Directory Services』 by Tim Howes et al.; Macmillan Technical Publishing*

第29章 AUTHENTICATION-CONFIGURATION

ユーザーが Red Hat Enterprise Linux システムにログインする場合は、有効かつアクティブなユーザーとしてユーザー名とパスワードの組み合わせを検証するか、認証 する必要があります。ユーザーがローカルシステムにあることを検証するための情報や、システムがリモートシステムのユーザーデータベースに認証を延期する場合などです。

Authentication Configuration Tool は、NIS、LDAP、および Hesiod サーバーからユーザー情報の取得を設定するグラフィカルインターフェイスを提供します。このツールでは、LDAP、Kerberos、および SMB を認証プロトコルとして設定することもできます。

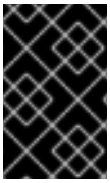


注記

インストール中にメディアまたは高いセキュリティーレベルを設定した場合（または Security Level Configuration Toolを使用して）、ファイアウォールは NIS (Network Information Service)認証を防ぎます。

本章では、各認証タイプについて詳細に説明しません。代わりに、Authentication Configuration Tool を使用して設定する方法を説明します。

デスクトップから Authentication Configuration Tool のグラフィカルバージョンを起動するには、システム（パネルで）> Administration > Authentication を選択するか、シェルプロンプトでコマンド system-config-authentication を入力します（例： XTerm または GNOME 端末）。



重要な影響

認証プログラムを終了すると、変更がすぐに有効になります。

29.1. ユーザー情報

User Information タブでは、ユーザーの認証方法を設定し、複数のオプションがあります。オプションを有効にするには、そのオプションの横にある空のチェックボックスをクリックします。オプションを無効にするには、そのオプションの横にあるチェックボックスをクリックしてチェックボックスの選択を解除します。OK をクリックしてプログラムを終了し、変更を適用します。

図29.1 ユーザー情報

[D]

以下の一覧では、各オプションの設定内容について説明します。

NIS

***Enable NIS Support** オプションは、ユーザーとパスワード認証のために NIS サーバーに接続するようにシステムを設定します(NIS クライアントとして)。NIS の設定 ボタンをクリックして、NIS ドメインおよび NIS サーバーを指定します。NIS サーバーが指定されていない場合、デーモンはブロードキャスト経由で検索を試みます。*

このオプションを機能させるには、ypbind パッケージをインストールする必要があります。NIS サポートを有効にすると、portmap サービスおよび ypbind サービスが起動し、システムの起動時に開始することもできます。

NIS の詳細は、「NIS のセキュア化」 を参照してください。

LDAP

Enable LDAP Support オプションは、システムに LDAP 経由でユーザー情報を取得するように指示します。Configure LDAP... ボタンをクリックして以下を指定します。

- LDAP 検索ベース DN: リストされた識別名(DN)を使用してユーザー情報を取得するように指定します。
- LDAP サーバー: LDAP サーバーの IP アドレスを指定します。
- TLS を使用して接続を暗号化 します。有効にすると、Transport Layer Security を使用して LDAP サーバーに送信されるパスワードを暗号化します。Download CA Certificate オプションを使用すると、有効な CA （認証局） 証明書をダウンロードする URL を指定できます。有効な CA 証明書は PEM (Privacy Enhanced Mail)形式である必要があります。

  CA 証明書の詳細は、「証明書およびセキュリティーの概要」 を参照してください。

このオプションを機能させるには、openldap-clients パッケージをインストールする必要があります。

LDAP の詳細は、28章Lightweight Directory Access Protocol (LDAP) を参照してください。

hesiod

Enable Hesiod Support オプションは、リモートの Hesiod データベースから情報（ユーザー情報を含む）を取得するようにシステムを設定します。Configure Hesiod... ボタンをクリックして以下を指定します。

- Hesiod LHS - Hesiod クエリーに使用されるドメイン接頭辞を指定します。

- *Hesiod RHS - デフォルトの Hesiod ドメインを指定します。*

このオプションを機能させるには、hesiod パッケージをインストールする必要があります。

Hesiod の詳細は、コマンド man hesiod を使用した man ページを参照してください。LHS および RHS の詳細は、hesiod.conf の man ページ(man hesiod.conf)を参照してください。

Winbind

Enable Winbind Support オプションは、システムが Windows Active Directory または Windows ドメインコントローラーに接続するように設定します。指定したディレクトリーまたはドメインコントローラーからのユーザー情報にアクセスできるようになり、サーバーの認証オプションを設定できます。Configure Winbind... ボタンをクリックして以下を指定します。

- *Winbind ドメイン: 接続する Windows Active Directory またはドメインコントローラーを指定します。*

- *Security Model - クライアントが Samba に応答する方法を設定するセキュリティーモデルを選択できます。ドロップダウンリストでは、以下のいずれかを選択できます。*

  - *user: これはデフォルトのモードです。このレベルのセキュリティーでは、クライアントは最初に有効なユーザー名とパスワードを使用してログインする必要があります。このセキュリティーモードでは、暗号化されたパスワードを使用することもできます。*

  - *サーバー - このモードでは、Samba は別の SMB サーバー(Windows NT Server など)で認証してユーザー名/パスワードの検証を試みます。試行に失敗すると、代わりにユーザー モードが有効になります。*

  - *ドメイン - このモードでは、Windows NT Server の仕組みと同様に、Samba は Windows NT Primary または Backup Domain Controller で認証してユーザー名/パスワードの検証を試みます。*

  - *ads - このモードは、Samba が Active Directory Server (ADS)レルムでドメインメンバーとして機能するように指示します。このモードで動作するには、krb5-server パッ*

ケージがインストールされ、Kerberos を適切に設定する必要があります。

- Winbind ADS レルム: ads セキュリティーモデルを選択すると、Samba サーバーがドメインメンバーとして機能させる ADS レルムを指定できます。

- Winbind ドメインコントローラー： このオプションを使用して、winbind が使用するドメインコントローラーを指定します。ドメインコントローラーの詳細は、「ドメインコントローラー」 を参照してください。

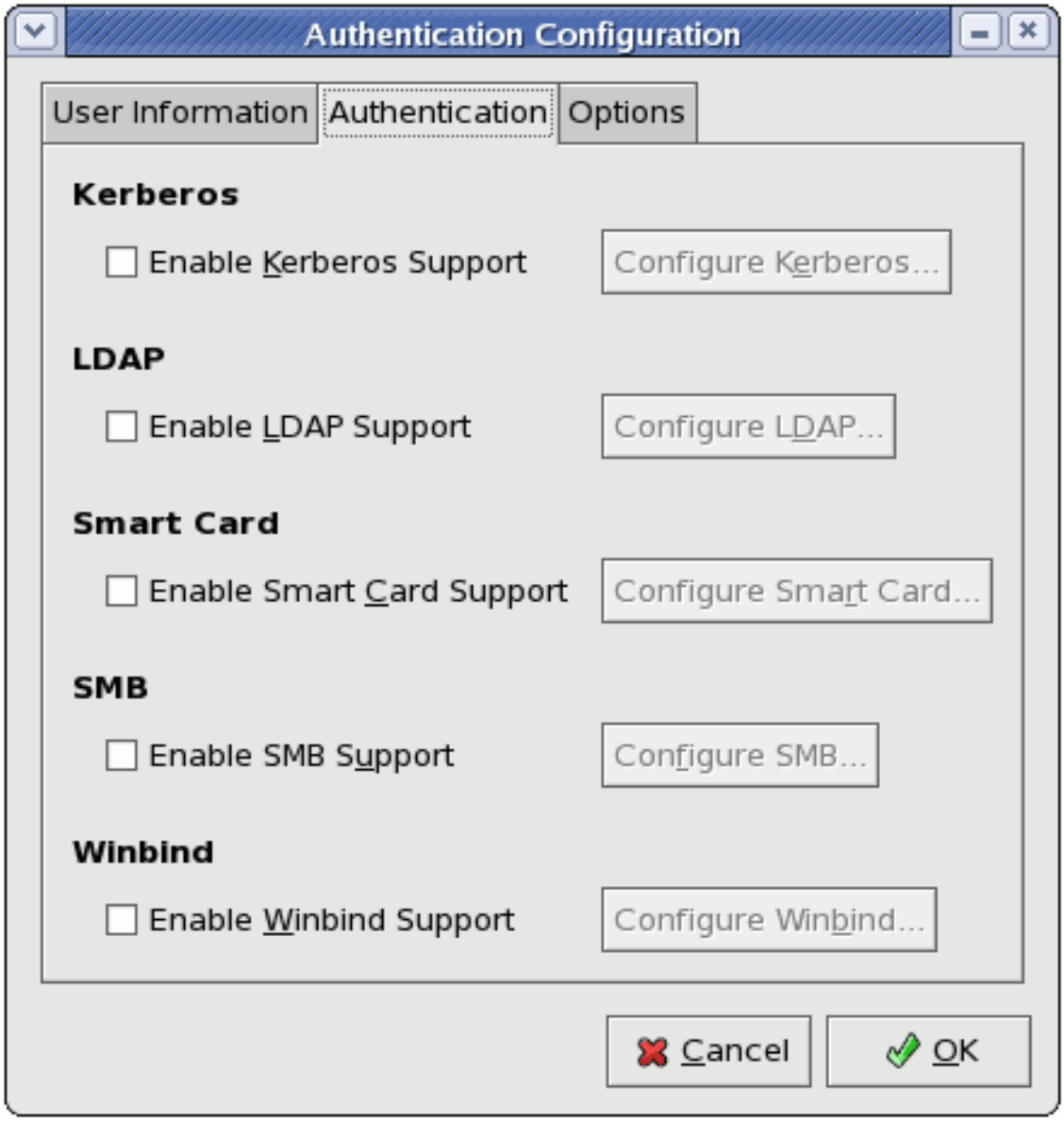
- テンプレートシェル: Windows NT ユーザーのユーザー情報を入力する際に、winbindd デーモンはここで選択した値を使用して、そのユーザーのログインシェルを指定します。

winbind サービスの詳細は、「Samba デーモンと関連サービス」 の winbindd を参照してください。

29.2. 認証

Authentication タブでは、ネットワーク認証方法を設定できます。オプションを有効にするには、そのオプションの横にある空のチェックボックスをクリックします。オプションを無効にするには、そのオプションの横にあるチェックボックスをクリックしてチェックボックスの選択を解除します。

図29.2 認証



[D]

以下では、各オプションの設定内容を説明します。

Kerberos

Kerberos サポートを有効 にする オプションは、Kerberos 認証を有効にします。Configure Kerberos... ボタンをクリックして Kerberos Settings ダイアログを開き、以下を設定します。

- *realm: Kerberos サーバーのレルムを設定します。レルムは、Kerberos を使用するネットワークで、1 つ以上の KDC と潜在的に多数のクライアントで設定されます。*

- *KDC: Kerberos チケットを発行するサーバーである Key Distribution Center (KDC)を定義します。*

- *管理サーバー - kadmind を実行する管理サーバーを指定します。*

Kerberos 設定ダイアログで は、DNS を使用してホストをレルムに解決し、レルムの KDC を見つけることもできます。

このオプションを機能 させるには、krb5-libs パッケージおよび krb5-workstation パッケージをインストールする必要があります。Kerberos の詳細は、「Kerberos」 を参照してください。

LDAP

Enable LDAP Support オプションは、認証に LDAP を使用するように標準の PAM 対応アプリケーションに指示します。Configure LDAP... ボタンを使用すると、Configure LDAP... の User Information タブにあるオプションと同じオプションで LDAP サポートを設定できます。これらのオプションの詳細は、「ユーザー情報」 を参照してください。

このオプションを機能させるには、openldap-clients パッケージをインストールする必要があります。

スマートカード

Enable Smart Card Support オプションは、スマートカード認証を有効にします。これにより、ユーザーはスマートカードに関連付けられた証明書とキーを使用してログインできます。その他のオプションについては、Configure Smart Card... ボタンをクリックします。

このオプションを機能させるには、pam_pkcs11 パッケージおよび coolkey パッケージをインストールする必要があります。スマートカードの詳細は、「対応するスマートカード」 を参照してください。「シングルサインオン(SSO)」

SMB

Enable SMB Support オプションは、SMB (Server Message Block)サーバーを使用してユーザーを認証するように PAM を設定します。SMB は、システム間の通信に使用されるクライアント/サーバープロトコルを参照します。また、Samba が Windows クライアントに Windows サーバーとして

表示するのに使用されるプロトコルでもあります。Configure SMB... ボタンをクリックして、以下を指定します。

- workgroup - 使用する SMB ワーク グループ を指定します。
- ドメインコントローラー: 使用する SMB ドメインコントローラーを指定します。

Winbind

Enable Winbind Support オプションは、システムが Windows Active Directory または Windows ドメインコントローラーに接続するように設定します。指定したディレクトリーまたはドメインコントローラーからのユーザー情報にアクセスできるようになり、サーバーの認証オプションを設定できます。

Configure Winbind... オプションは、User Information タブの Configure Winbind... ボタンにあるオプションと同じです。詳細は、Winbind (「ユーザー情報」の下)を参照してください。

29.3. オプション

このタブでは、以下のように他の設定オプションを使用できます。

図29.3 オプション

[D]

キャッシュユーザー情報

このオプションを選択して、ネームサービスキャッシュデーモン(nscd)を有効にし、ブート時に起動するように設定します。

このオプションを機能させるには、nscd パッケージをインストールする必要があります。nscd の詳細は、コマンドの man nscd を使用して man ページを参照してください。

シャドウパスワードの使用

このオプションを選択して、パスワードをシャドウパスワード形式で、/etc/passwd ではなく /etc/shadow ファイルに保存します。シャドウパスワードはインストール時にデフォルトで有効になっており、システムのセキュリティーを強化するのに強く推奨されます。

このオプションを機能させるには、shadow-utils パッケージをインストールする必要があります。シャドウパスワードの詳細は、「シャドウパスワード」 を参照してください。

MD5 パスワードの使用

このオプションを選択して MD5 パスワードを有効にします。これにより、パスワードは 8 文字以下ではなく最大 256 文字になります。これはインストール時にデフォルトで選択され、セキュリティーを強化するために強く推奨されます。

ローカルユーザーにはローカル認証で十分です。

このオプションを有効にすると、/etc/passwd ファイルに保持されるユーザーアカウントについて、システムはネットワークサービス(LDAP や Kerberos など)からの認証を確認しません。

ネットワークサービスによるシステムアカウントの認証

このオプションを有効にすると、ネットワークサービス(LDAP や Kerberos など)がマシンのシステムアカウント(root を含む)を認証できるようにシステムが設定されます。

29.4. コマンドラインバージョン

認証設定ツールは、 インターフェイスのないコマンドラインツールとして実行することもできます。コマンドラインバージョンは、設定スクリプトまたはキックスタートスクリプトで使用できます。認証オプションは、表29.1「コマンドラインオプション」 で要約されています。

ヒント

これらのオプションは authconfig の man ページか、シェルプロンプトで authconfig --help と入力しても確認できます。

表29.1 コマンドラインオプション

オプション	説明
--enableshadow	シャドウパスワードの有効化
--disableshadow	シャドウパスワードの無効化
--enablemd5	MD5 パスワードの有効化
--disablemd5	MD5 パスワードの無効化

オプション	説明
--enablenis	NIS の有効化
--disablenis	NIS の無効化
--nisdomain=<domain>	NIS ドメインの指定
--nisserver=<server>	NIS サーバーの指定
--enableldap	ユーザー情報の LDAP の有効化
--disableldap	ユーザー情報の LDAP の無効化
--enableldaptls	LDAP での TLS の使用の有効化
--disableldaptls	LDAP での TLS の使用の無効化
--enableldapauth	認証用の LDAP の有効化
--disableldapauth	認証の LDAP の無効化
--ldapserver=<server>	LDAP サーバーの指定
--ldapbasedn=<dn>	LDAP ベース DN の指定
--enablekrb5	Kerberos の有効化
--disablekrb5	Kerberos の無効化
--krb5kdc=<kdc>	Kerberos KDC の指定
--krb5adminserver=<server>	Kerberos 管理サーバーの指定
--krb5realm=<realm>	Kerberos レルムの指定
--enablekrb5kdcdns	DNS を使用して Kerberos KDC を検索
--disablekrb5kdcdns	DNS を使用した Kerberos KDC の検索の無効化

オプション	説明
--enablekrb5realmdns	DNS を使用した Kerberos レルムの検索の有効化
--disablekrb5realmdns	DNS を使用した Kerberos レルムの検索の無効化
--enablesmbauth	SMB の有効化
--disablesmbauth	SMB の無効化
--smbworkgroup=<workgroup>	SMB ワークグループの指定
--smbservers=<server>	SMB サーバーの指定
--enablewinbind	デフォルトでユーザー情報の winbind の有効化
--disablewinbind	デフォルトでユーザー情報の winbind の無効化
--enablewinbindauth	デフォルトで認証の winbindauth の有効化
--disablewinbindauth	デフォルトで認証の winbindauth を無効化
--smbsecurity=<user|server|domain|ads>	Samba および winbind に使用するセキュリティーモード
--smbrealm=<STRING>	security=adsの場合の Samba および winbind のデフォルトレルム
--smbidmapuid=<lowest-highest>	UID 範囲 winbind がドメインまたは ADS ユーザーに割り当てる
--smbidmapgid=<lowest-highest>	GID 範囲 winbind がドメインまたは ADS ユーザーに割り当てる
--winbindseparator=<\>	winbindusedefaultdomain が有効でない場合、winbind ユーザー名のドメインおよびユーザー部分を分離するために使用される文字

オプション	説明
--winbindtemplatehomedir=</home/%D/%U>	winbind ユーザーがホームディレクトリーとして持つディレクトリー
--winbindtemplateprimarygroup=<nobody>	winbind ユーザーがプライマリーグループとして持つグループ
--winbindtemplateshell=</bin/false>	winbind ユーザーがデフォルトのログインシェルとして持つシェル
--enablewinbindusedefaultdomain	ユーザー名にドメインのないユーザーがドメインユーザーであることを仮定するように winbind を設定します。
--disablewinbindusedefaultdomain	ユーザー名にドメインのないユーザーがドメインユーザーではないことを仮定するように winbind を設定します。
--winbindjoin=<Administrator>	管理者として winbind ドメインまたは ADS レルムを結合します。
--enablewins	ホスト名の解決に WINS を有効にします。
--disablewins	ホスト名解決の WINS の無効化
--enablehesiod	Hesiod の有効化
--disablehesiod	Hesiod の無効化
--hesiodlhs=<lhs>	Hesiod LHS の指定
--hesiodrhs=<rhs>	Hesiod RHS の指定
--enablecache	nscdの有効化
--disablecache	nscdの無効化
--nostart	portmap サービス、ypbind サービス、または nscd サービスが設定されている場合でも、それらを起動または停止しないでください。

オプション	説明
--kickstart	ユーザーインターフェイスを表示しません。
--probe	ネットワークのデフォルトをプローブおよび表示します。

第30章 SSSD での認証情報の使用およびキャッシュ

SSSD (System Security Services Daemon)は、さまざまな ID プロバイダーおよび認証プロバイダーへのアクセスを提供します。SSSD は、ローカルクライアントと設定済みのデータストアの仲介です。ローカルクライアントは SSSD に接続し、SSSD は外部プロバイダーに接続します。これにより、管理者には以下のような利点があります。

- ID/認証サーバーの負荷の削減。すべてのクライアントサービスが識別サーバーに直接接続しようとするのではなく、すべてのローカルクライアントは、識別サーバーへ接続したり、そのキャッシュを確認したりする SSSD と通信できます。
- オフライン認証を許可します。SSSD は、必要に応じて、リモートサービスから取得するユーザー ID および認証情報のキャッシュを保持できます。これにより、リモート ID サーバーがオフラインまたはローカルマシンがオフラインであっても、ユーザーはリソースに対して正常に認証できます。
- 単一ユーザーアカウントの使用 リモートユーザーには、ローカルシステム用や組織システム用など、2 つ（またはそれ以上の）ユーザーアカウントがあることがよくあります。これは、仮想プライベートネットワーク(VPN)に接続するために必要です。SSSD はキャッシュとオフライン認証をサポートしているため、リモートユーザーはローカルマシンに対して認証することでネットワークリソースに接続でき、SSSD はネットワーク認証情報を維持します。

System Security Services Daemon では、認証設定ツールと連携するために追加の設定やチューニングは必要ありません。ただし、SSSD は他のアプリケーションと連携でき、デーモンはこれらのアプリケーションのパフォーマンスを改善するために設定変更が必要になる場合があります。

30.1. SSSD.CONF ファイルについて

SSSD サービスおよびドメインは、.conf ファイルで設定されます。デフォルトのファイルは /etc/sssd/sssd.conf ですが、sssd コマンドで -c オプションを使用すると、代替ファイルを SSSD に渡すことができます。

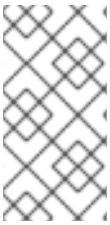
```
# sssd -c /etc/sssd/customfile.conf
```

[domain/LDAP] など、[type/name] で識別される設定の個別のセクションで、サービスとドメインの両方を個別に設定します。設定ファイルは、単純な key = value 行を使用して設定を設定します。コメント行は、ハッシュ記号(#)またはセミコロン(;)のいずれかで設定されます。

以下に例を示します。

```
[section]
# Comment line
key1 = val1
key10 = val1,val2
```

30.2. SSSD の起動および停止



注記

SSSD を初めて起動する前に、少なくとも 1 つのドメインを設定します。「ドメインの作成」を参照してください。

service コマンドまたは /etc/init.d/sssd スクリプトのいずれかが SSSD を開始できます。以下に例を示します。

```
# service sssd start
```

デフォルトでは、SSSD は自動的に起動しないように設定されています。この動作を変更するには、chkconfig コマンドを使用します。

```
[root@server ~]# chkconfig sssd on
```

30.3. システムサービスと連携するように SSSD を設定

SSSD は、SSSD プロセス自体と並行して実行する特殊なサービスと連携しました。SSSD および関連するサービスは、sssd.conf ファイルで設定されます。[sssd] セクションには、アクティブなサービスも一覧表示され、sssd が services ディレクティブ内で開始する際に起動する必要があります。

SSSD は現在、複数のサービスを提供します。

- sssd_nss モジュールからの name サービス要求に応答する Name Service Switch (NSS) プロバイダーサービス。これは、SSSD 設定の [nss] セクションで設定されます。
- sssd_pam モジュールを介して PAM 会話を管理する PAM プロバイダーサービス。これは、設定の [pam] セクションで設定されます。
- 監視 （他のすべての SSSD サービスを監視または再起動する特別なサービス）を監視しま

す。そのオプションは、/etc/sssd/sssd.conf 設定ファイルの [sssd] セクションで指定されます。

注記

DNS ルックアップがホスト名の IPv4 アドレスが返されない場合、SSSD は失敗を返す前に IPv6 アドレスを検索しようとします。これにより、非同期リゾルバーが正しいアドレスを識別することのみが保証されます。

ホスト名の解決動作は、sssd.conf 設定ファイルの lookup family order オプションで設定されます。

30.3.1. NSS サービスの設定

SSSD は、SSSD を使用してユーザー情報を取得するようにシステムに指示する NSS モジュール sssd_nss を提供します。NSS 設定には SSSD モジュールへの参照が含まれ、SSSD 設定 は SSSD が NSS と対話する方法を設定します。

30.3.1.1. NSS サービスマップおよび SSSD について

Name Service Switch (NSS)は、多くの設定および名前解決サービスを検索するサービスが中心的な設定を提供します。NSS は、設定ソースでシステム ID とサービスをマッピングする方法を 1 つ提供します。

SSSD は、NSS と複数のタイプの NSS マップのプロバイダーサービスとして機能します。

- *パスワード(パスワード)*
- *ユーザーグループ(シャドウ)*
- *グループ(グループ)*
- *netgroups (netgroups)*

30.3.1.2. SSSD を使用するように NSS サービスを設定する

NSS は、すべてサービスマップに対して、複数の ID および設定プロバイダーを使用できます。デフォルトでは、サービスにシステムファイルを使用します。SSSD を含めるには、希望のサービスタイプに nss_sss モジュールを含める必要があります。

Authentication Configuration ツールを使用して SSSD を有効にします。これは、SSSD をプロバイダーとして使用するように nsswitch.conf ファイルを自動的に設定します。

```
[root@server ~]# authconfig --enablesssd --update
```

これにより、SSSD モジュールを使用するように password、shadow、group、および netgroups サービスマップが自動的に設定されます。

```
passwd:     files sss
shadow:     files sss
group:      files sss

netgroup:   files sss
```

30.3.1.3. NSS と連携させる SSSD の設定

SSSD が NSS 要求の処理に使用するオプションおよび設定は、[nss] サービスセクションで SSSD 設定ファイルで設定されます。

1. sssd.conf ファイルを開きます。

```
[root@server ~]# vim /etc/sssd/sssd.conf
```

2. NSS が、SSSD と連携するサービスの 1 つとしてリストされていることを確認します。

```
[sssd]
config_file_version = 2
reconnection_retries = 3
sbus_timeout = 30
services = nss, pam
```

3. [nss] セクションで、NSS パラメーターのいずれかを変更します。これらは、表 30.1「sssd [nss] 設定パラメーター」 に記載されています。

```
[nss]
```

```
filter_groups = root
filter_users = root
reconnection_retries = 3
entry_cache_timeout = 300
entry_cache_nowait_percentage = 75
```

4. ***SSSD を再起動します。***

```
[root@server ~]# service sssd restart
```

表30.1 sssd [nss] 設定パラメーター

パラメーター	値の形式	[root@server ~] Description
enum_cache_timeout	integer	**sssd_nss** がすべてのユーザーに関する情報の要求をキャッシュする期間（秒単位）を指定します（列挙）。
entry_cache_nowait_percentage	integer	キャッシュを更新する前に **sssd_nss** がキャッシュされたエントリーを返す時間を指定します。これをゼロに設定すると、エントリーキャッシュの更新が無効になります。 これにより、次の更新前の特定の間隔が一定の割合である場合に、エントリーキャッシュがバックグラウンドで自動的にエントリーを更新するように設定されます。たとえば、間隔が 300 秒でキャッシュの割合が 75 の場合、要求が間隔の 225 秒 - 75% になると、エントリーキャッシュが更新を開始します。 このオプションに使用できる値は 0 から 99 で、**entry_cache_timeout** 値に基づいてパーセンテージを設定します。デフォルト値は 50% です。
entry_negative_timeout	integer	**sssd_nss** が **負** のキャッシュヒットをキャッシュする期間（秒単位）を指定します。負のキャッシュヒットは、存在しないエントリーを含む、無効なデータベースエントリーのクエリーです。

パラメーター	値の形式	[root@server ~] Description
filter_users、filter_groups	string	特定のユーザーが NSS データベースからフェッチされないように、SSSD に指示します。これは、**root** などのシステムアカウントに特に便利です。
filter_users_in_groups	ブール値	グループルックアップの実行時に、**filter_users** リストに一覧表示されているユーザーがグループメンバーシップに表示されるかどうかを設定します。**false** に設定すると、グループルックアップはそのグループのメンバーであるすべてのユーザーを返します。指定されていない場合、デフォルト値は **true** で、グループメンバーー覧をフィルターリングします。

30.3.2. PAM サービスの設定

警告

PAM 設定ファイルの間違いにより、ユーザーがシステムから完全にロックされる可能性があります。変更を実行する前に設定ファイルを常にバックアップし、セッションを開いたままにして、変更を元に戻すことができるようにします。

SSSD は、SSSD を使用してユーザー情報を取得するようにシステムに指示する PAM モジュール sssd_pam を提供します。PAM 設定には SSSD モジュールへの参照が含まれ、SSSD 設定 は SSSD が PAM と対話する方法を設定します。

PAM サービスを設定するには、以下を実行します。

1. *認証設定ツールは、/etc/pam.d/system-auth-ac ファイルに自動的に書き込みます。このファイルは /etc/pam.d/system-auth へのシンボリックリンクです。/etc/pam.d/system-auth に加えた変更は、次に authconfig が実行される際に上書きされます。*

したがって、/etc/pam.d/system-auth シンボリックリンクを削除します。

```
[root@server ~]# rm /etc/pam.d/system-auth
rm: remove symbolic link `/etc/pam.d/system-auth'? y
```

2. 新しい /etc/pam.d/system-auth-local ファイルを作成します。これを行う簡単な方法の 1 つは、単に /etc/pam.d/system-auth-ac ファイルをコピーすることです。

```
[root@server ~]# cp /etc/pam.d/system-auth-ac /etc/pam.d/system-auth-local
```

3. /etc/pam.d/system-auth-local ファイルと /etc/pam.d/system-auth の間に新しいシンボリックリンクを作成します。

```
[root@server ~]# ln -s /etc/pam.d/system-auth-local /etc/pam.d/system-auth
```

4. /etc/pam.d/system-auth-local ファイルを編集し、すべての SSSD モジュールを PAM 設定に追加します。

```
#%PAM-1.0
...
auth        sufficient    pam_sss.so use_first_pass
auth        required      pam_deny.so

...
account [default=bad success=ok user_unknown=ignore] pam_sss.so
account     required      pam_permit.so

...
password    sufficient    pam_sss.so use_authtok
password    required      pam_deny.so

...
session     sufficient    pam_sss.so
session     required      pam_unix.so
```

これらのモジュールは、必要に応じてステートメントを 含む ように設定できます。

5. sssd.conf ファイルを開きます。

```
# vim /etc/sssd/sssd.conf
```

6.

PAM が SSSD と連携するサービスの 1 つとして一覧表示されていることを確認してください。

```
[sssd]
config_file_version = 2
reconnection_retries = 3
sbus_timeout = 30
services = nss, pam
```

7. ***[pam] セクションで、PAM パラメーターのいずれかを変更します。これらは、表 30.2「SSSD [pam] 設定パラメーター」 に記載されています。***

```
[pam]
reconnection_retries = 3
offline_credentials_expiration = 2
offline_failed_login_attempts = 3
offline_failed_login_delay = 5
```

8. ***SSSD を再起動します。***

```
[root@server ~]# service sssd restart
```

表30.2 SSSD [pam] 設定パラメーター

パラメーター	値の形式	説明
offline_credentials_expiration	integer	認証プロバイダーがオフラインの場合にキャッシュしたログインを許可する期間を日数単位で設定します。この値は、最後に成功したオンラインログインから測定されます。指定しない場合、デフォルトは**0**(0)で、これは無制限です。
offline_failed_login_attempts	integer	認証プロバイダーがオフラインの場合に許可されるログイン試行の失敗回数を設定します。指定しない場合、デフォルトは**0**(0)で、これは無制限です。

パラメーター	値の形式	説明
offline_failed_login_delay	integer	ユーザーがログイン試行の失敗制限に達した場合にログイン試行を防ぐ時間を設定します。ゼロ(**0**)に設定すると、失敗した試行制限に達すると、プロバイダーはオフラインである間は認証できません。オンライン認証の成功のみが、オフライン認証を再度有効にできます。指定しない場合、デフォルトで 5 (5)に設定されます。

30.4. ドメインの作成

SSSD は、さまざま な ID サーバーに関連付けられているドメイン を認識します。ドメインは、アイデンティティープロバイダーと認証方法の組み合わせです。SSSD は LDAP アイデンティティープロバイダー(OpenLDAP、Red Hat Directory Server、Microsoft Active Directory を含む)と連携し、ネイティブ LDAP 認証または Kerberos 認証を使用できます。

SSSD は、異なるドメインに属する限り、同じユーザー名を持つ異なるユーザーを認識できます。たとえば、SSSD は ldap.example.com ドメインで jsmith と ldap.otherexample.com ドメインの jsmith の両方を正常に認証できます。SSSD は完全修飾ドメイン名を使用した要求を許可するため、jsmith@ldap.example.com の情報を要求すると、適切なユーザーアカウントが返されます。ユーザー名のみを指定すると、ドメインが検索順に最初に表示されるユーザーが返されます。

ヒント

SSSD には filter_users オプションがあり、指定したユーザーが検索で返されないのを除外します。

ドメインの設定は、ユーザー情報が保存される 場所 と、これらのユーザーがシステムへの認証を許可する 方法 を定義します。使用できる組み合わせは 表30.3「アイデンティティーストアおよび認証タイプの組み合わせ」 に記載されています。

- *「ドメインを設定するための一般的なルールとオプション」*
- *「LDAP ドメインの設定」*

- *「ドメインを使用した Kerberos 認証の設定」*
- *「プロキシードメインの設定」*

表30.3 アイデンティティーストアおよび認証タイプの組み合わせ

ID プロバイダー	認証プロバイダー
LDAP	LDAP
LDAP	Kerberos
proxy	LDAP
proxy	Kerberos
proxy	proxy

30.4.1. ドメインを設定するための一般的なルールとオプション

ドメイン設定は、アイデンティティープロバイダー、認証プロバイダー、およびそれらのプロバイダー の情報にアクセスするための特定の設定を定義します。アイデンティティープロバイダーには、LDAP とプロキシーの 2 つのタイプの認証プロバイダー(LDAP、Kerberos、およびプロキシー)があります。ID プロバイダーおよび認証プロバイダーは、ドメインエントリー内の任意の組み合わせで設定できます。

ドメインエントリー自体とともに、SSSD がクエリーするドメイン一覧にドメイン名を追加する必要があります。以下に例を示します。

```
domains = LOCAL,Name

[domain/Name]
id_provider = type
auth_provider = type
provider_specific = value
global = value
```

グローバル 属性は、キャッシュやタイムアウトの設定など、あらゆるタイプのドメインで使用できます。各アイデンティティープロバイダーおよび認証プロバイダーには、独自の必須および任意の設定パラメーターセットがあります。

表30.4 一般的な [domain] 設定パラメーター

パラメーター	値の形式	説明
id_provider	string	このドメインに使用するデータプロバイダーアイデンティティーバックエンドを指定します。サポート対象のバックエンドは以下のとおりです。 ● ldap ● ipa (FreeIPA バージョン 2.x との互換性) ● **nss_nis** などのレガシー NSS プロバイダーのプロキシー。プロキシー ID プロバイダーを使用するには、**proxy_lib_name** オプションで設定した、正常に起動するようにレガシー NSS ライブラリーを指定する必要があります。 ● local、SSSD 内部ローカルプロバイダー
auth_provider	string	ドメインに使用される認証プロバイダーを設定します。このオプションのデフォルト値は **id_provider** の値です。サポートされる認証プロバイダーは ldap、ipa、krb5 (Kerberos)、proxy、および none です。

パラメーター	値の形式	説明
min_id,max_id	integer	**(オプション)**ドメインの UID および GID の範囲を指定します。ドメインにその範囲外のエントリーが含まれる場合は、それらは無視されます。**min_id** のデフォルト値は **1** です。**max_id** のデフォルト値は **0** で、無制限です。  **重要** デフォルトの **min_id** 値は、すべてのタイプのアイデンティティープロバイダーで同じです。LDAP ディレクトリーが 1 から始まる UID 番号を使用している場合は、ローカルの **/etc/passwd** ファイルのユーザーと競合する可能性があります。これらの競合を回避するには、**min_id** を **1000** 以上に設定します。
列挙	ブール値	**(オプション)**ドメインのユーザーおよびグループを一覧表示するかどうかを指定します。列挙とは、リモートソースで利用可能なユーザーおよびグループ全体が、ローカルマシンにキャッシュされることを意味します。列挙が無効になっている場合、ユーザーおよびグループは要求時にのみキャッシュされます。

パラメーター	値の形式	説明
		 **警告** 列挙を有効にすると、クライアントを再初期化すると、リモートソースから利用可能なユーザーおよびグループのセット全体が完全に更新されます。同様に、SSSD が新しいサーバーに接続すると、リモートソースからの利用可能なユーザーおよびグループのセット全体が、ローカルマシンにプルおよびキャッシュされます。リモートソースに接続されているクライアントが多数あるドメインでは、クライアントからのクエリーが頻繁に行われるため、この更新プロセスではネットワークのパフォーマンスに悪影響を与える可能性があります。利用可能なユーザーおよびグループのセットが十分に大きくなると、クライアントのパフォーマンスも低下します。

パラメーター	値の形式	説明
		このパラメーターのデフォルト値は **false** で、列挙を無効にします。
cache_credentials	ブール値	**(オプション)**ローカルの SSSD ドメインデータベースキャッシュにユーザーの認証情報を保存するかどうかを指定します。このパラメーターのデフォルト値は **false** です。オフライン認証を有効にするには、この値を **true** に設定します。
entry_cache_timeout	integer	**(オプション)**SSSD が **正** のキャッシュヒットをキャッシュする期間を秒単位で指定します。正のキャッシュヒットはクエリーの成功です。
use_fully_qualified_names	ブール値	**(オプション)**このドメインへのリクエストに完全修飾ドメイン名が必要なかどうかを指定します。**true** に設定すると、このドメインへのすべてのリクエストは完全修飾ドメイン名を使用する必要があります。また、リクエストからの出力に完全修飾名が表示されることも意味します。完全修飾ユーザー名に要求を制限すると、SSSD はユーザー名が競合するユーザーのドメインを区別できます。 ***use_fully_qualified_names*** を **false** に設定すると、リクエストで完全修飾名を使用できますが、出力には簡素化されたバージョンのみが表示されます。 SSSD はレルム名ではなく、ドメイン名に基づいて名前のみを解析できます。ただし、ドメインとレルムの両方に同じ名前を使用できます。

30.4.2. LDAP ドメインの設定

LDAP ドメインは、SSSD が LDAP ディレクトリーをアイデンティティープロバイダー（およびオプションで認証プロバイダーとして使用する）として使用することを意味します。SSSD は、以下のよ

うな主要なディレクトリーサービスに対応します。

- Red Hat Directory Server
- OpenLDAP
- Microsoft Active Directory 2008 (UNIX ベースのアプリケーションのサブシステムあり)

注記

DNS サービス検出により、LDAP バックエンドは、特別な DNS クエリーを使用して自動的に接続する適切な DNS サーバーを検索できます。

- 「LDAP ドメインを設定するためのパラメーター」
- 「LDAP ドメインの例」
- 「Active Directory ドメインの例」
- 「証明書のサブジェクト名での IP アドレスの使用」

30.4.2.1. LDAP ドメインを設定するためのパラメーター

LDAP ディレクトリーは、アイデンティティープロバイダーと認証プロバイダーの両方として機能します。この設定には、LDAP サーバーのユーザーディレクトリーを特定して接続するための十分な情報が必要ですが、これらの接続パラメーターを定義する方法は柔軟です。

LDAP サーバーへの接続に使用するユーザーアカウントの指定や、パスワード操作に異なる LDAP サーバーの使用など、より詳細な制御を可能にする他のオプションを利用できます。最も一般的なオプションは、表30.5「LDAP ドメイン設定パラメーター」 に記載されています。LDAP ドメインでも、 「ドメインを設定するための一般的なルールとオプション」 に記載されているオプションをすべて利用できます。

ヒント

その他のオプションの多くは、LDAP ドメイン設定の man ページ sssd-ldap (5) に記載されています。

表30.5 LDAP ドメイン設定パラメーター

パラメーター	説明
ldap_uri	SSSD が接続する LDAP サーバーの URI のコンマ区切りリストを指定します。この一覧は優先順に指定されるため、リストの最初のサーバーは最初に試行されます。追加のサーバーを一覧表示すると、フェイルオーバー保護が提供されます。これは、DNS SRV レコードから検出できます（指定されていない場合）。
ldap_search_base	LDAP ユーザー操作の実行に使用するベース DN を指定します。

パラメーター	説明
ldap_tls_reqcert	TLS セッションで SSL サーバー証明書を確認する方法を指定します。4 つのオプションがあります。 • 証明書の要求を無効にし ないでください。 • Allow a certificate, but still still still even if no certificate is no no certificate or a bad certificate is provided. • 証明書を要求して、証明書が指定されていない場合には通常を続行します。不正な証明書が指定されている場合は、セッションは終了します。 • 需要 と ハード は同じオプションです。これには、有効な証明書またはセッションの終了が必要です。 デフォルトは hard です。
ldap_tls_cacert	SSSD が認識するすべての CA の CA 証明書が含まれるファイルへの完全パスおよびファイル名を指定します。SSSD は、これらの CA が発行する証明書を受け入れます。 明示的に指定されていない場合は、OpenLDAP システムのデフォルトを使用します。

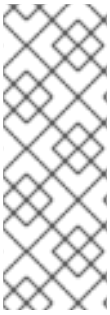
パラメーター	説明
ldap_referrals	SSSD が LDAP 参照を使用するかどうか、つまり、ある LDAP データベースから別の LDAP データベースからクエリーを転送するかどうかを設定します。SSSD は、データベースレベルとサブツリーの参照をサポートします。同じ LDAP サーバー内の参照では、SSSD はクエリーされるエントリーの DN を調整します。異なる LDAP サーバーに到達する参照では、SSSD は DN に完全一致します。この値を true に設定すると、参照が有効になります。これがデフォルトです。

ldap_schema	ユーザーエントリーの検索時に使用するスキーマのバージョンを設定します。これは、rfc2307 または rfc2307 bis のいずれかになります。デフォルトは rfc2307 です。
	
	RFC 2307 では、グループオブジェクトは多値属性 memberuid を使用します。これは、そのグループに属するユーザーの名前を一覧表示します。RFC 2307bis では、グループオブジェクトは、ユーザーまたはグループエントリーの完全な識別名(DN)を含む member 属性を使用します。RFC 2307bis を使用すると、member 属性を作成したネスト化されたグループを使用できます。これらのスキーマはグループメンバーシップに異なる定義を使用するため、SSSD で誤った LDAP スキーマを使用すると、適切なパーミッションが設定されている場合でも、ネットワークリソースの表示と管理の両方に影響する可能性があります。
	
	たとえば、RFC 2307bis では、ネストされたグループまたはプライマリー/セカンダリーグループを使用するときにすべてのグループが返されます。
	
	`$ id`
	`uid=500(myserver) gid=500(myserver)`
	`groups=500(myserver),510(myothergroup)`

パラメーター	説明
	SSSD が RFC 2307 スキーマを使用している場合は、プライマリーグループのみが返されます。 この設定は、SSSD がグループメンバーを決定する方法にのみ影響します。実際のユーザーデータは変更されません。
ldap_search_timeout	LDAP 検索がキャンセルされ、キャッシュされた結果が返される前に LDAP 検索を実行できる時間を秒単位で設定します。これは、列挙値が false で、列挙 が true の場合にデフォルト で 30 に設定されます。 LDAP 検索がタイムアウトすると、SSSD は自動的にオフラインモードに切り替わります。
ldap_network_timeout	接続の試行に失敗した後に SSSD が LDAP サーバーのポーリングを試行する時間を秒単位で設定します。デフォルトは 6 秒です。
ldap_opt_timeout	サーバーから応答が受信されない場合に、同期 LDAP 操作を中止するまで待機する時間を秒単位で設定します。このオプションは、SASL バインドの場合に KDC と通信する際のタイムアウトも制御します。デフォルトは 5 秒です。

30.4.2.2. LDAP ドメインの例

LDAP 設定は、特定の環境や SSSD の動作に応じて柔軟性が非常に高くなります。以下は LDAP ドメインの一般的な例ですが、SSSD 設定はこれらの例に限定されません。



注記

ドメインエントリーの作成に加えて、新しいドメインを SSSD のドメインの一覧に追加して、sssd.conf ファイルでクエリーします。以下に例を示します。

```
domains = LOCAL,LDAP1,AD,PROXYNIS
```

例30.1 基本的な LDAP ドメイン設定

LDAP ドメインには、以下の 3 つが必要です。

- *LDAP サーバー*
- *検索ベース*
- *セキュアな接続を確立する方法*

最後の項目は LDAP 環境によって異なります。SSSD は機密情報を処理するため、セキュアな接続が必要です。この接続は、専用の TLS/SSL 接続にすることも、Start TLS を使用できます。

専用の TLS/SSL 接続を使用すると、LDAPS 接続を使用してサーバーに接続するだけで、ldap_uri オプションの一部として設定されます。

```
# An LDAP domain
[domain/LDAP]
enumerate = false
cache_credentials = true

id_provider = ldap
auth_provider = ldap

ldap_uri = ldaps://ldap.example.com:636
ldap_search_base = dc=example,dc=com
```

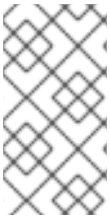
Start TLS を使用するには、安全でないポートでセキュア接続を確立するために証明書情報を入力する方法が必要です。これは、ldap_id_use_start_tls オプションを使用して Start TLS を使用し、ldap_tls_cacert を使用して SSL サーバー証明書を発行した CA 証明書を特定します。

```
# An LDAP domain
```

```
[domain/LDAP]
enumerate = false
cache_credentials = true

id_provider = ldap
auth_provider = ldap

ldap_uri = ldap://ldap.example.com
ldap_search_base = dc=example,dc=com
ldap_id_use_start_tls = true
ldap_tls_reqcert = demand
ldap_tls_cacert = /etc/pki/tls/certs/ca-bundle.crt
```

30.4.2.3. Active Directory ドメインの例

SSSD が Active Directory ドメインと連携するには、Active Directory ドメインとローカルシステムの両方を特別に相互通信できるように設定する必要があります。



注記

Microsoft Active Directory のドキュメントには、Active Directory ドメインの設定に関する完全な手順が記載されています。

1. authconfig を使用して、Active Directory を LDAP アイデンティティープロバイダーとして使用するように Linux クライアントを設定します。以下に例を示します。

```
authconfig --enableldap --enableldapauth --ldapserver=ldap://ad.example.com:389 --
enablekrb5 --krb5realm AD-REALM.EXAMPLE.COM --krb5kdc ad-kdc.example.com:88
--krb5adminserver ad-kdc.example.com:749 --update
```

authconfig コマンドは、「コマンドラインバージョン」 で説明されています。

2. Active Directory ドメインサービスロールを作成します。

3. UNIX サービスの Identity Management を Active Directory Domain Services ロールに追加します。Unix NIS ドメインを設定でドメイン名として使用します。

4. Active Directory サーバーで、Linux クライアントの名前で新しい Computer オブジェクトを作成します。

a.
Administrative Tools メニューで、Active Directory Users and Computers アプリケーションを選択します。

b.
ad.example.com などの Active Directory ルートオブジェクトを展開します。

c.
Computers を右クリックし、New および Computer アイテムを選択します。

d.
rhel-server などの Linux クライアントの名前を入力し、OK をクリックします。

e.
Computers オブジェクトを展開します。

f.
rhel-server オブジェクトを右クリックし、Properties を選択します。

g.
UNIX Attributes に、Linux NIS ドメインの名前と Linux サーバーの IP アドレスを入力します。

OK をクリックします。

5.
Active Directory サーバーのコマンドプロンプトから、Linux ホストプリンシパル用のマシンアカウント、パスワード、および UPN を作成します。

```
C:\> setspn -A host/rhel-server.example.com@AD-REALM.EXAMPLE.COM rhel-server
Registering ServicePrincipalNames for CN=rhel
server,CN=Computers,DC=ad,DC=example,DC=com
        host/rhel server.example.com@AD-REALM.EXAMPLE.COM
Updated object

C:\> setspn -L rhel-server
Registered ServicePrincipalNames for CN=rhel
server,CN=Computers,DC=ad,DC=example,DC=com:
        host/rhel server.example.com@AD-REALM.EXAMPLE.COM

C:\> ktpass /princ host/rhel-server.example.com@AD-REALM.EXAMPLE.COM /out
rhel-server.keytab /crypto all /ptype KRB5_NT_PRINCIPAL -desonly /mapuser AD\rhel-
server$ +rndPass

Targeting domain controller:
   ad.example.com
```

```
Using legacy password setting method
Successfully mapped host/rhel server.redhat.com
... 8< ...
```

6. Active Directory サーバーから Linux クライアントに keytab をコピーし、/etc/krb5.keytab として保存します。

7. Linux システムで、キータブファイルのパーミッションと所有者をリセットします。

```
[root@rhel-server ~]# chown root:root /etc/krb5.keytab

[root@rhel-server ~]# chmod 0600 /etc/krb5.keytab
```

8. キータブの SELinux ファイルパーミッションを復元します。

```
[root@rhel-server ~]# restorecon /etc/krb5.keytab
```

9. ホストが Active Directory ドメインに接続できることを確認します。

```
[root@rhel-server ~]# kinit -k -t /etc/krb5.keytab host/rhel-server.example.com@AD-REALM.EXAMPLE.COM
```

10. Active Directory サーバーで、Linux ユーザーのグループを作成します。

    a. unixusers という名前の新規グループを作成します。

    b. unixusers グループを開き、Unix Attributes タブを開きます。

    c. Unix 設定を設定します。

    - NIS ドメイン
    - UID

- */bin/bash*へのログインシェル
- */home/aduser*へのホームディレクトリー
- *unixusers*へのプライマリーグループ名

11. *次に、Linux マシンで SSSD ドメインを設定します。*

例30.2 Active Directory 2008 ドメイン

```
[root@rhel-server ~]# vim /etc/sssd/sssd.conf

[sssd]
config_file_version = 2
domains = ad.example.com
services = nss, pam

[nss]

[pam]

[domain/ad.example.com]
cache_credentials = true
enumerate = false

id_provider = ldap
auth_provider = krb5
chpass_provider = krb5
access_provider = ldap

ldap_sasl_mech = GSSAPI
ldap_sasl_authid = host/rhel-server.example.com@AD-REALM.EXAMPLE.COM


ldap_schema = rfc2307bis

ldap_user_search_base = ou=user accounts,dc=ad,dc=example,dc=com
ldap_user_object_class = user
ldap_user_home_directory = unixHomeDirectory
ldap_user_principal = userPrincipalName
ldap_user_name = sAMAccountName

ldap_group_search_base = ou=groups,dc=ad,dc=example,dc=com
ldap_group_object_class = group

ldap_access_order = expire
ldap_account_expire_policy = ad
ldap_force_upper_case_realm = true
```

```
ldap_disable_referrals = true

#krb5_server = server.ad.example.com
krb5_realm = AD-REALM.EXAMPLE.COM
```

これらのオプションは、LDAP ドメイン設定の man ページ sssd-ldap (5) で説明されています。

12. SSSD を再起動します。

```
[root@rhel-server ~]# service sssd restart
```

30.4.2.4. 証明書のサブジェクト名での IP アドレスの使用

サーバー名の代わりに ldap_uri オプションで IP アドレスを使用すると、TLS/SSL 接続が失敗する場合があります。TLS/SSL 証明書には、IP アドレスではなくサーバー名が含まれます。ただし、証明書の サブジェクト代替名 フィールドを使用してサーバーの IP アドレスを含めることができます。これにより、IP アドレスを使用したセキュアな接続に成功します。

1. 既存の証明書を証明書要求に変換します。署名鍵(-signkey)は、最初に証明書を発行した CA の発行者の鍵です。これを外部 CA で実行する場合は、別の PEM ファイルが必要になります。証明書が自己署名されている場合は、これが証明書自体になります。以下に例を示します。

```
openssl x509 -x509toreq -in old_cert.pem -out req.pem -signkey key.pem
```

自己署名証明書の場合：

```
openssl x509 -x509toreq -in old_cert.pem -out req.pem -signkey old_cert.pem
```

2. /etc/pki/tls/openssl.cnf 設定ファイルを編集して、[v3_ca] セクションにサーバーの IP アドレスを追加します。

```
subjectAltName = IP:10.0.0.10
```

3. 生成された証明書要求を使用して、指定された IP アドレスで新しい自己署名証明書を生成します。

```
openssl x509 -req -in req.pem -out new_cert.pem -extfile ./openssl.cnf -extensions
v3_ca -signkey old_cert.pem
```

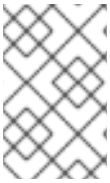
-extensions オプションは、証明書で使用する拡張を設定します。そのためには、適切なセクションを読み込むために v3_ca である必要があります。

4. old_cert.pem ファイルから new_cert.pem ファイルに秘密鍵ブロックをコピーし、関連するすべての情報を 1 つのファイルに保持します。

nss-tools パッケージが提供する certutil ユーティリティーを使用して証明書を作成する場合は、certutil は証明書作成の DNS サブジェクト代替名のみをサポートすることに注意してください。

30.4.3. ドメインを使用した Kerberos 認証の設定

LDAP およびプロキシーアイデンティティープロバイダーはいずれも、別の Kerberos ドメインを使用して認証を提供できます。Kerberos 認証プロバイダーを設定するには、キー配布センター (KDC) と Kerberos ドメインが必要です。すべてのプリンシパル名は指定されたアイデンティティープロバイダーで利用可能でなければなりません。そうでない場合は、SSSD は username@REALM 形式を使用してプリンシパルを構築します。



注記

Kerberos は認証のみを提供でき、ID データベースを提供することはできません。

SSSD は、Kerberos KDC が Kerberos kadmin サーバーでもあることを前提としています。ただし、実稼働環境では、通常 KDC の複数の読み取り専用レプリカと、単一の kadmin サーバーのみがあります。krb5_kpasswd オプションを使用して、パスワードの変更サービスが実行されている場所、またはデフォルト以外のポートで実行している場合は を指定します。krb5_kpasswd オプションが定義されていない場合、SSSD は Kerberos KDC を使用してパスワードを変更しようとします。

基本的な Kerberos 設定オプションは、表30.6「Kerberos 認証設定パラメーター」 に記載されています。sssd-krb5 (5) の man ページには、Kerberos 設定オプションの詳細情報が記載されています。

例30.3 Basic Kerberos 認証

```
# A domain with identities provided by LDAP and authentication by Kerberos
[domain/KRBDOMAIN]
enumerate = false
id_provider = ldap
```

```
chpass_provider = krb5
ldap_uri = ldap://ldap.example.com
ldap_search_base = dc=example,dc=com
ldap-tls_reqcert = demand
ldap_tls_cacert = /etc/pki/tls/certs/ca-bundle.crt

auth_provider = krb5
krb5_server = 192.168.1.1, kerberos.example.com
krb5_realm = EXAMPLE.COM
krb5_kpasswd = kerberos.admin.example.com
krb5_auth_timeout = 15
```

表30.6 Kerberos 認証設定パラメーター

パラメーター	説明
chpass_provider	パスワードの変更操作に使用するサービスを指定します。これは、認証プロバイダーと同じであることを前提とします。Kerberos を使用するには、これを krb5 に設定します。
krb5_server	SSSD が接続する Kerberos サーバーのホスト名(IP アドレスまたはホスト名)のコンマ区切りリストを指定します。この一覧は優先順に指定されるため、リストの最初のサーバーは最初に試行されます。追加のサーバーを一覧表示すると、フェイルオーバー保護が提供されます。 KDC または kpasswd サーバーでサービス検出を使用する場合、SSSD は最初に UDP を接続プロトコルとして指定する DNS エントリーを検索し、TCP にフォールバックします。
krb5_realm	KDC が提供する Kerberos レルムを特定します。
krb5_lifetime	指定されたライフタイム(s)、分(m)、時間(h)、または日(d)で Kerberos チケットを要求します。
krb5_renewable_lifetime	秒単位(s)、分(m)、時間(h)、または日(d)で指定される合計ライフタイムで更新可能な Kerberos チケットを要求します。

パラメーター	説明
krb5_renew_interval	チケットを更新する必要があるかどうかをチェックする SSSD の時間を秒単位で設定します。チケットは、有効期間の半分を超えると自動的に更新されます。このオプションがないか、またはゼロに設定すると、チケットの自動更新が無効になります。
krb5_store_password_if_offline	Kerberos 認証プロバイダーがオフラインの場合にユーザーパスワードを保存し、そのキャッシュを使用してプロバイダーがオンラインに戻るときにチケットを要求するかどうかを設定します。デフォルトは false で、パスワードを保存しません。
krb5_kpasswd	変更パスワードサービスが KDC で実行されていない場合に使用する代替 Kerberos kadmin サーバーを一覧表示します。
krb5_ccname_template	ユーザーのクレデンシャルキャッシュを保存するために使用するディレクトリーを指定します。これは一時的な化が可能であり、以下のトークンがサポートされます。 • %u （ユーザーのログイン名） • %u （ユーザーのログイン UID） • %p （ユーザーのプリンシパル名） • %r （レルム名） • %h （ユーザーのホームディレクトリー） • %d (krb5ccache_dir パラメーターの値)

パラメーター	説明
	• %p (SSSD クライアントのプロセス ID)。 • %% （リテラルパーセント記号 (%)） • XXXXXX: テンプレートの最後にある文字列で、SSSD に一意のファイル名を安全に作成するように指示する。 以下に例を示します。 krb5_ccname_template = FILE:%d/krb5cc_%U_XXXXXX
krb5_ccachedir	認証情報キャッシュを保存するディレクトリーを指定します。これは、%d および %P を除き、krb5_ccname_template と同じトークンを使用してテンプレート化できます。%u、%U、%p、または %h が使用される場合、SSSD はユーザーごとにプライベートディレクトリーを作成します。それ以外の場合は、パブリックディレクトリーを作成します。
krb5_auth_timeout	オンライン認証または変更パスワードリクエストが中断されるまでの秒数を指定します。可能な場合、認証要求はオフラインで継続されます。デフォルトは 15 秒です。

30.4.4. プロキシードメインの設定

SSSD を使用するプロキシーは、中間設定であるリレーです。SSSD はプロキシーサービスに接続し、そのプロキシーは指定されたライブラリーを読み込みます。これにより、SSSD は使用できないリソースの一部を使用できます。たとえば、SSSD は認証プロバイダーとして LDAP および Kerberos のみをサポートしますが、プロキシーを使用すると、SSSD はフィンガープリントスキャナーやスマートカードなどの別の認証方法を使用できます。

表30.7 プロキシードメイン設定パラメーター

パラメーター	説明
proxy_pam_target	PAM が認証プロバイダーとしてプロキシーする必要のあるターゲットを指定します。PAM ターゲットは、デフォルトの PAM ディレクトリー /etc/pam.d/ に PAM スタック情報が含まれるファイルです。 これは、認証プロバイダーのプロキシーに使用されます。  重要 プロキシー PAM スタックに pam_sss.so が再帰的に追加されない ようにします。
proxy_lib_name	アイデンティティー要求をプロキシー処理する既存の NSS ライブラリーを指定します。 これは、アイデンティティープロバイダーをプロキシーするために使用されます。

例30.4 プロキシー ID および Kerberos 認証

プロキシーライブラリーは、proxy_lib_name パラメーターを使用して読み込まれます。このライブラリーは、指定の認証サービスと互換性がある限り使用できます。Kerberos 認証プロバイダーの場合は、NIS などの Kerberos 互換ライブラリーである必要があります。

```
[domain/PROXY_KRB5]
auth_provider = krb5
krb5_server = 192.168.1.1
krb5_realm = EXAMPLE.COM

id_provider = proxy
```

```
proxy_lib_name = nis
enumerate = true
cache_credentials = true
```

例30.5 LDAP Identity および Proxy 認証

プロキシーライブラリーは、proxy_pam_target パラメーターを使用して読み込まれます。このライブラリーは、指定のアイデンティティープロバイダーと互換性のある PAM モジュールである必要があります。たとえば、これは LDAP で PAM フィンガープリントモジュールを使用します。

```
[domain/LDAP_PROXY]
id_provider = ldap
ldap_uri = ldap://example.com
ldap_search_base = dc=example,dc=com

auth_provider = proxy
proxy_pam_target = sssdpamproxy
enumerate = true
cache_credentials = true
```

SSSD ドメインを設定したら、指定した PAM ファイルが設定されていることを確認します。この例では、ターゲットが sssdpamproxy であるため、/etc/pam.d/sssdpamproxy ファイルを作成し、PAM/LDAP モジュールを読み込みます。

```
auth        required      pam_frprint.so
account     required      pam_frprint.so
password    required      pam_frprint.so
session     required      pam_frprint.so
```

例30.6 プロキシー ID および認証

SSSD は、ID プロキシーと認証プロキシーの両方を持つドメインを持つことができます。ここで指定する設定は、認証 PAM モジュールのプロキシー設定 proxy_pam_target と、NIS や LDAP などのサービスの proxy_lib_name のみです。

この例は、可能な設定を示していますが、これは現実的な設定ではありません。LDAP がアイデンティティーおよび認証に使用される場合は、アイデンティティープロバイダーと認証プロバイダーの両方をプロキシーではなく LDAP 設定に設定する必要があります。

```
[domain/PROXY_PROXY]
auth_provider = proxy
id_provider = proxy
proxy_lib_name = ldap
```

```
proxy_pam_target = sssdproxyldap
enumerate = true
cache_credentials = true
```

SSSD ドメインが追加されたら、システム設定を更新してプロキシーサービスを設定します。

1. pam_ldap.so モジュールを必要とする /etc/pam.d/sssdproxyldap ファイルを作成します。

```
auth        required      pam_ldap.so
account     required      pam_ldap.so
password    required      pam_ldap.so
session     required      pam_ldap.so
```

2. nss-pam-ldap パッケージがインストールされていることを確認します。

```
[root@server ~]# yum install nss-pam-ldap
```

3. /etc/nslcd.conf ファイル(LDAP ネームサービスデーモンの設定ファイル)を編集して、LDAP ディレクトリーの情報が含まれるようにします。

```
uid nslcd
gid ldap
uri ldaps://ldap.example.com:636
base dc=example,dc=com
ssl on
tls_cacertdir /etc/openldap/cacerts
```

30.5. SSSD ドメインのアクセス制御の設定

SSSD は、ドメイン設定の基本的なアクセス制御を提供し、単純なユーザーの許可/拒否リスト、または LDAP バックエンド自体の使用を可能にします。

30.5.1. Simple Access プロバイダーの使用

Simple Access Provider は、ユーザー名またはグループのリストに基づいてアクセスを許可または拒否します。

Simple Access Provider は、特定のマシンへのアクセスを制限する方法です。たとえば、ある会

社がラップトップを使用する場合、同じ認証プロバイダーに対して異なるユーザーが正常に認証された場合でも、Simple Access Provider を使用して、特定のユーザーまたは特定のグループのみへのアクセスを制限できます。

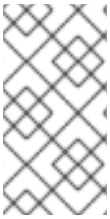
最も一般的なオプションは simple_allow_users および simple_allow_groups です。これは、特定のユーザー（指定のユーザーまたはグループメンバーのいずれか）に明示的にアクセスを付与し、その他のすべてのユーザーへのアクセスを拒否します。拒否リストを作成することも可能です（明示的なユーザーのみへのアクセスを拒否し、他のすべてのユーザーに暗黙的にアクセスを許可）。

Simple Access Provider は、アクセスを付与すべきユーザーかどうかを決定するために、以下の 4 つのルールに従います。

- 許可リストと拒否リストの両方が空の場合、アクセスが許可されます。
- リストを指定すると、許可ルールが最初に評価され、次にルールが拒否されます。実際には、拒否ルールは allow ルールよりも優先されることを意味します。
- 許可されたリストを指定すると、すべてのユーザーが一覧に表示されない限りアクセスが拒否されます。
- 拒否リストのみを指定すると、一覧にない限り、すべてのユーザーがアクセスが許可されます。

この例では、IT グループに所属する 2 つのユーザーおよびすべてのユーザーへのアクセスを付与します。暗黙的に、他のすべてのユーザーは拒否されます。

```
[domain/example.com]
access_provider = simple
simple_allow_users = jsmith,bjensen
simple_allow_groups = itgroup
```



注記

SSSD の LOCAL ドメインは、アクセスプロバイダーとしての simple をサポートしません。

その他のオプションは sssd-simple の man ページに一覧表示されますが、これはほとんど使用さ

れません。

30.5.2. LDAP アクセスフィルターの使用

LDAP サーバー自体はアクセス制御ルールを提供できます。関連するフィルターオプション (ldap_access_filter)は、指定したホストへのアクセスを付与するユーザーを指定します。ユーザーフィルターを使用するか、すべてのユーザーがアクセスを拒否する必要があります。

以下に例を示します。

```
[domain/example.com]
access_provider = ldap
ldap_access_filter = memberOf=cn=allowedusers,ou=Groups,dc=example,dc=com
```

注記

LDAP アクセスプロバイダーのオフラインキャッシュは、ユーザーの最後にオンラインログインの試行が成功したかどうかの判断に限定されます。最後のログイン時にアクセスを付与されたユーザーは、オフライン中にアクセスは引き続き付与されます。

SSSD は、アカウントの有効期限ポリシーおよび authorizedService 属性で結果を確認することもできます。

30.6. ドメインフェイルオーバーの設定

SSSD は、マシンおよびサービスへの接続を個別に試行します。

SSSD がドメインバックエンドのいずれかへの接続を試みると、最初に特定のマシンのホスト名を解決しようとします。この解決を試みると、マシンはオフラインと見なされ、SSSD は他のサービスに対してこのマシンへの接続を試行しなくなります。

解決を試みると、バックエンドはこのマシンのサービスに接続しようとします。サービス接続試行に失敗すると、この特定のサービスのみがオフラインとみなされ、バックエンドが自動的に次のサービスに切り替えます。マシンは引き続きオンラインとみなされ、別のサービスで試行されている可能性があります。

SSSD は、DNS A レコードで指定された最初の IP アドレスのみを試みます。1 つの要求で複数のサーバーを見つけるために、SSSD は SRV レコードに依存します。

SSSD がバックエンドに正常に接続されるまで、接続は 30 秒ごとにオフラインマシンまたはサービスに再試行されます。

30.6.1. フェイルオーバーの設定

フェイルオーバーを設定すると、プライマリーサーバーに障害が発生した場合に SSSD が自動的に別のサーバーに切り替えることができます。これらのサーバーは、/etc/sssd/sssd.conf ファイルの [domain/Name] セクションに、大文字と小文字を区別しないコンマ区切りリストとして入力されます。サーバーは優先順に一覧表示されます。このリストには任意の数のサーバーを含めることができます。

たとえば、ネイティブ LDAP ドメインの場合は以下のようになります。

```
ldap_uri = ldap://ldap0.example.com, ldap://ldap1.example.com, ldap://ldap2.example.com
```

最初のエントリー ldap://ldap0.example.com はプライマリーサーバーです。このサーバーが失敗すると、SSSD は最初に ldap1.example.com への接続を試み、次に ldap2.example.com への接続を試みます。

server パラメーターが指定されていない場合、SSSD はサービス検出を使用してネットワーク上の別のサーバーの検索を試みます。

重要

フェイルオーバーサーバーは、単一のキーの値のコンマ区切りリストとして入力する必要があります。鍵が複数あると、SSSD は最後のエントリーのみを認識します。

30.6.2. フェイルオーバーでの SRV レコードの使用

SSSD は、フェイルオーバー設定で SRV レコードに対応します。SSSD 設定では、SRV 要求を使用して、特定のサーバーの一覧に後で解決されるサーバーを指定できます。

サービス検出を使用するすべてのサービスについて、特別な DNS レコードを DNS サーバーに追加します。

```
_service._protocol._domain TTL priority weight port hostname
```

SRV レコードの 優先順位 と 重み 属性により、プライマリーサーバーに障害が発生した場合に、最初に接続するサーバーを詳細に制御できます。

通常の設定には、このようなレコードが複数含まれ、それぞれにフェイルオーバーに異なる優先順位があり、負荷分散には異なる重みがあります。

SRV レコードの詳細は、RFC 2782 を参照してください。

30.7. ドメインキャッシュファイルの削除

SSSD は、同じタイプおよび異なるタイプのドメインの複数のドメインを定義できます。SSSD は、ドメインごとに個別のデータベースファイルを維持します。つまり、各ドメインには独自のキャッシュがあります。これらのキャッシュファイルは /var/lib/sss/db/ ディレクトリーに保存されます。

ドメインに問題がある場合は、SSSD を停止し、そのドメインのキャッシュファイルを削除して、キャッシュを簡単にパージできます。

すべてのキャッシュファイルは、ドメインに対して名前が付けられます。たとえば、exampleldap という名前のドメインの場合、キャッシュファイルの名前は cache_exampleldap.ldb になります。

キャッシュファイルを削除する場合は注意してください。この操作には大きな影響があります。

- *キャッシュファイルを削除すると、識別およびキャッシュされた認証情報の両方のすべてのユーザーデータが削除されます。したがって、システムがオンラインであり、ドメインのサーバーに対してユーザー名で認証できる場合を除き、キャッシュファイルを削除しないでください。認証情報キャッシュがないと、オフライン認証は失敗します。*

- *別のアイデンティティープロバイダーを参照するように設定が変更された場合、SSSD は元のプロバイダーのキャッシュされたエントリーがタイムアウトするまで、両方のプロバイダーのユーザーを認識します。*

  キャッシュをパージすることで回避できますが、新しいプロバイダーに異なるドメイン名を使用することが推奨されます。SSSD が再起動すると、新しい名前で新しいキャッシュファイルが作成され、古いファイルは無視されます。

30.8. SSSD での NSCD の使用

SSSD は、NSCD デーモンで使用するように設計されていません。SSSD は NSCD と直接競合しませんが、両サービスを使用すると、特にエントリーがキャッシュされる期間において予期しない動作が発生する可能性があります。

問題の最も一般的な証拠は NFS と競合します。Network Manager を使用してネットワーク接続を管理する場合、ネットワークインターフェイスが起動するまでに数分かかる場合があります。この間、さまざまなサービスが起動しようとします。これらのサービスがネットワークが稼働している前に起動し、DNS サーバーが利用できる場合には、これらのサービスは必要な正引きまたは逆引き DNS エントリーを特定できません。これらのサービスは、誤った、または空の resolv.conf ファイルを読み取る可能性があります。このファイルは、通常 1 回だけ読み取るため、このファイルへの変更は自動的に適用されません。これにより、サービスを手動で再起動しない限り、NSCD サービスが実行されているマシンで NFS ロックが失敗する可能性があります。

この問題を回避するには、/etc/nscd.conf ファイルでホストおよびサービスのキャッシュを有効にし、passwd、group、および netgroup エントリーの SSSD キャッシュに依存します。

/etc/nscd.conf ファイルを変更します。

```
enable-cache hosts yes
enable-cache passwd no
enable-cache group no
enable-cache netgroup no
```

ホストの要求に回答する NSCD により、これらのエントリーは NSCD によりキャッシュされ、ブートプロセス中に NSCD によって返されます。その他のエントリーはすべて SSSD により処理されます。

30.9. SSSD のトラブルシューティング

30.9.1. SSSD ログファイルの確認

SSSD は、/var/log/sssd/ ディレクトリーにある操作に関する情報を報告するログファイルを使用します。SSSD は、各ドメインのログファイルと sssd_pam.log および sssd_nss.log ファイルを生成します。

さらに、/var/log/secure ファイルは認証の失敗と、失敗の原因を記録します。

ログレベルを増やすと、SSSD の問題に関する詳細情報が提供されます。ログレベルを変更するには、sssd.conf ファイルの各セクションに debug_level パラメーターを設定します。このファイルは、追加のログを提供します。以下に例を示します。

```
[sssd]
config_file_version = 2
services = nss, pam
domains = LDAP
debug_level = 9
```

表30.8 デバッグログレベル

レベル	説明
0	致命的な障害。SSSD の起動を妨げる、または SSSD の実行を停止させるもの。
1	重大なエラー。SSSD を強制終了しないものの、少なくとも 1 つの主要な機能が適切に機能しないことを示すエラー。
2	深刻なエラー。特定の要求または操作が失敗したことを示すエラー。
3	マイナーな障害。これらのエラーが浸透して、2 の動作不良の原因となるのです。
4	設定設定。
5	関数データ。
6	操作関数のメッセージを追跡します。
7	内部制御関数のメッセージのトレース。
8	対象の関数内部変数の内容。
9	非常に低いレベルのトレース情報。

30.9.2. SSSD 設定に関する問題

SSSD が起動に失敗する

SSSD では、デーモンを起動する前に、必要なすべてのエントリーで設定ファイルを適切に設定する必要があります。

- *SSSD では、サービスが起動する前に、最低でもドメインを適切に設定する必要がありま*

す。ドメインがないと、SSSD を起動すると、ドメインが設定されていないエラーが返されます。

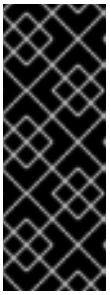
```
# sssd -d4

[sssd] [ldb] (3): server_sort:Unable to register control with rootdse!
[sssd] [confdb_get_domains] (0): No domains configured, fatal error!
[sssd] [get_monitor_config] (0): No domains configured.
```

/etc/sssd/sssd.conf ファイルを編集し、最低でも 1 つのドメインを作成します。

- *SSSD は、開始する前に、少なくとも 1 つ以上の利用可能なサービスプロバイダーも必要です。問題がサービスプロバイダー設定にある場合、エラーメッセージはサービスが設定されていないことを示します。*

```
[sssd] [get_monitor_config] (0): No services configured!
```

/etc/sssd/sssd.conf ファイルを編集し、1 つ以上のサービスプロバイダーを設定します。



重要

SSSD では、サービスプロバイダーを /etc/sssd/sssd.conf ファイルの単一の services エントリーに、コンマ区切りのリストとして設定する必要があります。サービスが複数のエントリーに一覧表示されます。最後のエントリーのみが SSSD によって認識されます。

id を持つグループや、getent group を持つグループメンバーは表示されません。

これは、sssd.conf の [domain/DOMAINNAME] セクションに誤った ldap_schema 設定が原因である可能性があります。

SSSD は RFC 2307 および RFC 2307bis スキーマタイプをサポートします。デフォルトでは、SSSD はより一般的な RFC 2307 スキーマを使用します。

RFC 2307 と RFC 2307bis の相違点は、グループメンバーシップが LDAP サーバーに保存される方法です。RFC 2307 サーバーでは、グループメンバーはメンバーであるユーザーの名前が含まれる、多値 memberuid 属性として保存されます。RFC2307bis サーバーでは、グループメンバーは、このグループのメンバーであるユーザーまたはグループの DN を含む多値 member または uniqueMember 属性として保存されます。RFC2307bis を使用すると、ネストされたグループも保守できます。

グループルックアップが情報が返されない場合は、以下を行います。

1. *ldap_schema を rfc2307bis に設定します。*
2. *Delete /var/lib/sss/db/cache_DOMAINNAME.ldb.*
3. *SSSD を再起動します。*

これが機能しない場合は、以下の行を sssd.conf に追加します。

```
ldap_group_name = uniqueMember
```

次に、キャッシュを削除し、再度 SSSD を再起動します。

認証は LDAP に対して失敗します。

認証を実行するには、SSSD で通信チャネルを暗号化する必要があります。これは、sssd.conf が標準プロトコル (ldap://) 経由で接続するように設定されていると、Start TLS で通信チャネルの暗号化を試みます。sssd.conf がセキュアなプロトコル (ldaps://) に接続するように設定されている場合、SSSD は SSL を使用します。

つまり、LDAP サーバーは SSL または TLS で実行する必要があります。標準の LDAP ポート (389) で TLS を有効にするか、セキュア LDAPS ポート (636) で SSL を有効にする必要があります。SSL または TLS のいずれかを使用する場合、LDAP サーバーも有効な証明書の信頼で設定する必要があります。

無効な証明書の信頼は、LDAP に対する認証に関する最も一般的な問題の 1 つです。クライアントが LDAP サーバー証明書を適切に信頼していない場合、接続を検証できず、SSSD はパスワードの送信を拒否します。LDAP プロトコルでは、パスワードをプレーンテキストで LDAP サーバーに送信する必要があります。暗号化されていない接続でプレーンテキストのパスワードを送信することは、セキュリティーの問題です。

証明書が信頼されていない場合は、TLS 暗号化を開始できなかったことを示す syslog メッセージが書き込まれます。証明書設定は、SSSD とは別に LDAP サーバーにアクセスできるかどうかを確認してテストできます。たとえば、以下は、test.example.com への TLS 接続を介して匿名バインドをテストします。

```
$ ldapsearch -x -ZZ -h test.example.com -b dc=example,dc=com
```

証明書信頼が適切に設定されていない場合、テストは以下のエラーを出して失敗します。

```
ldap_start_tls: Connect error (-11) additional info: TLS error -8179:Unknown code ___f 13
```

証明書を信頼するには、次のコマンドを実行します。

1. LDAP サーバー証明書に署名するために使用される認証局の公開 CA 証明書のコピーを取得してローカルシステムに保存します。

2. ファイルシステムの CA 証明書を参照する sssd.conf ファイルに行を追加します。

```
ldap_tls_cacert = /path/to/cacert
```

3. LDAP サーバーが自己署名証明書を使用する場合は、sssd.conf ファイルから ldap_tls_reqcert 行を削除します。

このパラメーターにより、SSSD が CA 証明書により発行された証明書を信頼するように指示します。これは、自己署名の CA 証明書を使用するセキュリティーリスクになります。

非標準ポートで LDAP サーバーへの接続に失敗します。

SELinux を Enforcing モードで実行する場合は、クライアントの SELinux ポリシーを変更して、標準以外のポートで LDAP サーバーに接続する必要があります。以下に例を示します。

```
# semanage port -a -t ldap_port_t -p tcp 1389
```

NSS がユーザー情報を返すことができません

これは通常、SSSD が NSS サービスに接続できないことを意味します。

- NSS が実行していることを確認します。

```
# service sssd status
```

-

NSS が実行している場合、プロバイダーが /etc/sssd/sssd.conf ファイルの [nss] セクションで適切に設定されていることを確認します。特に、filter_users および filter_groups 属性を確認します。

- *NSS が SSSD が使用するサービスの一覧に含まれていることを確認します。*
- */etc/nsswitch.conf ファイルの設定を確認します。*

NSS が間違ったユーザー情報を返す

検索が正しくないユーザー情報を返す場合は、別のドメインでユーザー名が競合していないことを確認してください。複数のドメインがある場合は、/etc/sssd/sssd.conf ファイルで use_fully_qualified_domains 属性を true に設定します。これは、同じ名前の異なるドメインの異なるユーザーを区別します。

ローカルの SSSD ユーザーのパスワードを設定すると、パスワードが 2 回要求されます。

ローカルの SSSD ユーザーのパスワードを変更しようとすると、パスワードを 2 回求められる場合があります。

```
[root@clientF11 tmp]# passwd user1000
Changing password for user user1000.
New password:
Retype new password:
New Password:
Reenter new Password:
passwd: all authentication tokens updated successfully.
```

これは、PAM 設定が間違っているためです。/etc/pam.d/system-auth ファイルで use_authtok オプションが正しく設定されていることを確認します。

パート IV. システム設定

システム管理者のジョブの一部として、さまざまなタスク、ユーザーの種類、およびハードウェア設定についてシステムを設定します。本セクションでは、Red Hat Enterprise Linux システムを設定する方法を説明します。

第31章 コンソールアクセス

通常の(root 以外の)ユーザーがコンピューターにローカルにログインすると、以下の 2 種類の特別なパーミッションが付与されます。

1. このプログラムは、実行できない特定のプログラムを実行できます。

2. これらは、ディスクへのアクセスに使用される特定のファイル（通常は特殊なデバイスファイル）にアクセスでき、ディスクにはアクセスできなくなります。

1 台のコンピューターに複数のコンソールがあり、複数のユーザーがコンピューターに同時にログインできるため、いずれかのユーザーは基本的にファイルにアクセスするために競合を優先する必要があります。コンソールでログインする最初のユーザーはこれらのファイルを所有します。最初のユーザーがログアウトすると、そのファイルを所有している次のユーザーがログインします。

一方、コンソールにログインする すべて のユーザーは、通常 root ユーザーに制限されるタスクを実行するプログラムを実行できます。X が実行されている場合、グラフィカルユーザーインターフェイスにメニュー項目としてこれらのアクションを追加できます。これらのコンソールで利用できるプログラムには、halt、poweroff、および reboot などがあります。

31.1. CTRL+ALT+DEL でのシャットダウンの無効化

デフォルトでは、/etc/inittab は、コンソールで使用される Ctrl+Alt+Del キーの組み合わせに対して、システムがシャットダウンされるように設定され、再起動します。この機能を完全に無効にするには、/etc/inittab の前にハッシュマーク(#)を配置することで、以下の行をコメントアウトします。

```
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
```

または、root 以外のユーザーに、Ctrl+Alt+Del を使用してコンソールからシステムをシャットダウンしたり、再起動したりすることもできます。以下の手順を実行して、この権限を特定のユーザーに制限できます。

1. 上記の /etc/inittab 行に -a オプションを追加して、以下のようにします。

   ```
   ca::ctrlaltdel:/sbin/shutdown -a -t3 -r now
   ```

   -a フラグはシャットダウンに対し、/etc/ shutdown.allow ファイルを検索するように指示します。

2. /etc に shutdown.allow という名前のファイルを作成します。shutdown.allow ファイルには、Ctrl+Alt+Del を使用してシステムをシャットダウンできるユーザーのユーザー名が一覧表示されます。shutdown.allow ファイルのフォーマットは、以下のように 1 行に 1 つずつユーザー名の一覧です。

```
stephen
jack
sophie
```

この shutdown.allow ファイルの例では 、 ユーザーは Ctrl+Alt+Del を使用して コンソールからシステムをシャットダウンできます。鍵の組み合わせが使用されると、/etc/inittab の shutdown -a コマンドは、/etc/shutdown.allow （または root）のユーザーが仮想コンソールにログインしているかどうかを確認します。ある場合は、システムのシャットダウンが続行されます。シャットダウンされていない場合は、エラーメッセージがシステムコンソールに書き込まれます。

shutdown.allow の詳細は、man ページの shutdown を参照してください。

31.2. コンソールプログラムアクセスの無効化

ユーザーがコンソールプログラムへのアクセスを無効にするには、root で以下のコマンドを実行します。

```
rm -f /etc/security/console.apps/*
```

コンソールが保護される環境では(BIOS およびブートローダーのパスワードが設定され、Ctrl+Alt+Delete が無効になり、電源とリセットスイッチが無効になるなど)、コンソールのユーザーが poweroff を実行したり停止 したり、再起動 したりしないようにできます。これは、デフォルトでコンソールからアクセスできます。

これらの機能を無効にするには、root で以下のコマンドを実行します。

```
rm -f /etc/security/console.apps/poweroff
rm -f /etc/security/console.apps/halt
rm -f /etc/security/console.apps/reboot
```

31.3. コンソールの定義

pam_console.so モジュールは、/etc/security/console.perms ファイルを使用して、システムコンソールのユーザーのパーミッションを決定します。ファイルの構文は非常に柔軟です。これらの命令が

適用されないようにファイルを編集できます。ただし、デフォルトの ファイルには以下のような行があります。

```
<console>=tty[0-9][0-9]* vc/[0-9][0-9]* :[0-9]\.[0-9] :[0-9]
```

ユーザーがログインすると、名前付きターミナルの一部に割り当てられます。これは、名前が : 0 または mymachine.example.com:1. 0、または /dev/ttyS0 や /dev/ pts/2 などのデバイスのいずれかになります。デフォルトでは、ローカルの仮想コンソールとローカル X サーバーがローカルとみなされることを定義しますが、ポート /dev/ttyS1 で隣のシリアルターミナルを local にみなす場合は、その行を変更して読み込むことができます。

```
<console>=tty[0-9][0-9]* vc/[0-9][0-9]* :[0-9]\.[0-9] :[0-9] /dev/ttyS1
```

31.4. コンソールからファイルにアクセスできるようにする

個々のデバイスクラスおよびパーミッション定義のデフォルト設定は、/etc/security/console.perms.d/50-default.perms で定義されます。ファイルおよびデバイスのパーミッションを編集するには、/etc/security/console.perms.d/ に、指定したファイルセットに対する希望の設定を含む新しいデフォルトファイルを作成することが推奨されます。50-default. permsを上書きするには、新しいデフォルトファイルの名前を 50 より大きい数で開始する必要があります（例 : 51 -default.perms）。

これを行うには、/etc/security/console. perms.d/ に 51-default.perms という名前の新しいファイルを作成します。

```
touch /etc/security/console.perms.d/51-default.perms
```

元のデフォルト perms ファイル(50-default.perms)を開きます。最初のセクションでは、以下のような行を持つ デバイスクラス を定義します。

```
<floppy>=/dev/fd[0-1]* \
        /dev/floppy/* /mnt/floppy*
<sound>=/dev/dsp* /dev/audio* /dev/midi* \
  /dev/mixer* /dev/sequencer \
  /dev/sound/* /dev/beep \
  /dev/snd/*
<cdrom>=/dev/cdrom* /dev/cdroms/* /dev/cdwriter* /mnt/cdrom*
```

括弧で囲まれている項目。上記の例では、< cdrom> は CD-ROM ドライブを参照します。新しいデバイスを追加するには、デフォルトの 50-default.perms ファイルで定義しないでください。代わり

に、51-default.perms に定義します。たとえば、スキャナーを定義するには、以下の行を 51-default.perms に追加します。

```
<scanner>=/dev/scanner /dev/usb/scanner*
```

当然ながら、デバイスに適切な名前を使用する必要があります。/dev/scanner が実際にはスキャナーであり、ハードドライブなどの他のデバイスではないことを確認します。

デバイスまたはファイルを適切に定義したら、2 つ目の手順は、パーミッション定義 を指定することです。/etc/security/console.perms.d/50-default.perms の 2 番目のセクションで、以下のような行を定義します。

```
<console> 0660 <floppy> 0660 root.floppy
<console> 0600 <sound>  0640 root
<console> 0600 <cdrom>  0600 root.disk
```

スキャナーのパーミッションを定義するには、51-default.perms で以下のような行を追加します。

```
<console> 0600 <scanner> 0600 root
```

次に、コンソールでログインすると、/dev/scanner デバイスの所有権が 0600 になります（読み取り、書き込みのみ）。ログアウトすると、デバイスは root によって所有され、パーミッションは 0600 のままです(root のみが読み取り可能、書き込み可能)。

WARNING

デフォルトの 50-default.perms ファイルを編集する こと はできません。50-default.perms にすでに定義されているデバイスのパーミッションを編集するには、51 -default.perms にそのデバイスの必要なパーミッション定義を追加します。これにより、50-default.perms で定義されているすべてのパーミッションが上書きされます。

31.5. 他のアプリケーションのコンソールアクセスの有効化

コンソールユーザーが他のアプリケーションにアクセスできるようにするには、さらに作業が必要です。

まず、コンソールアクセスは /sbin/ または /usr /sbin/ にあるアプリケーションに対してのみ 機能するので、実行するアプリケーションが必要です。これを確認した後に、以下の手順を実行します。

1. サンプル foo プログラムなどのアプリケーションの名前から、/usr/bin/consolehelper アプリケーションへのリンクを作成します。

```
cd /usr/bin
ln -s consolehelper foo
```

2. /etc/security/console.apps/foo ファイルを作成します。

```
touch /etc/security/console.apps/foo
```

3. /etc/pam.d/ に、foo サービスの PAM 設定ファイルを作成します。これを行う簡単な方法は、halt サービスの PAM 設定ファイルをコピーしてから、動作を変更する場合はコピーを変更することです。

```
cp /etc/pam.d/halt /etc/pam.d/foo
```

/usr/bin/foo を実行すると、consolehelper が呼び出され、/usr/sbin/userhelper を利用してユーザーを認証します。ユーザーを認証するため、/etc/pam.d/foo が /etc/pam.d/halt のコピーである場合に、consolehelper はユーザーのパスワードを要求します。それ以外の場合は、/etc/pam.d/fooで指定されている内容を正確に実行し、root 権限で /usr/sbin/foo を実行します。

PAM 設定ファイルでは、pam_timestamp モジュールを使用して正常な認証の試行を記憶（またはキャッシュ）するようにアプリケーションを設定できます。アプリケーションが起動し、適切な認証 (root パスワード)が提供されると、タイムスタンプファイルが作成されます。デフォルトでは、正常な認証は 5 分間キャッシュされます。この間、pam_timestamp を使用して同じセッションから実行するように設定された他のアプリケーションは、そのユーザーに対して自動的に認証されます。ユーザーは root パスワードを再度入力する必要はありません。

このモジュールは pam パッケージに含まれています。この機能を有効にするには、etc/pam.d/ の PAM 設定ファイルに以下の行を追加します。

```
auth        include     config-util
account     include     config-util
session     include     config-util
```

これらの行は、/etc/pam.d/system-config-* 設定ファイルからコピーできます。これらの行は、

PAM 設定ファイルの他の auth sufficient session optional 行の 下 に追加する必要があります。

pam_timestamp を使用するよう設定されたアプリケーションが Applications （パネルのメインメニュー）から正常に認証されると、GNOME または KDE デスクトップ環境を実行している場合は、パネルの通知エリアに

アイコンが表示されます。認証の期限が切れると（デフォルトは 5 分）、アイコンは消えます。

ユーザーは、アイコンをクリックし、認証を取得するオプションを選択して、キャッシュされた認証を忘れることができます。

31.6. フロッピー グループ

何らかの理由でコンソールアクセスは適切ではなく、root 以外のユーザーがシステムのディスケットドライブにアクセスする必要がある場合は、フロッピー グループを使用して実行できます。選択したツールを使用して フロッピー グループにユーザーを追加します。たとえば、gpasswd コマンドを使用して、フロッピー グループにユーザー fred を追加できます。

```
gpasswd -a fred floppy
```

これで、ユーザー fred はコンソールからシステムのディスケットドライブにアクセスできるようになりました。

第32章 SYSCONFIG ディレクトリー

/etc/sysconfig/ ディレクトリーには、Red Hat Enterprise Linux のさまざまなシステム設定ファイルが含まれています。

本章では、/etc/sysconfig/ ディレクトリーにあるファイル、それらの機能、およびそれらのコンテンツの概要を説明します。本章の情報を完了することは意図されていません。これらのファイルの多くには、非常に具体的な状況またはまれな状況でのみ使用されるさまざまなオプションがあるためです。

32.1. /ETC/SYSCONFIG/ ディレクトリーのファイル

以下のセクションでは、通常 /etc/sysconfig/ ディレクトリーにある ファイルについて説明します。ここに記載されていないファイルと追加のファイルオプションは、/usr/share/doc/initscripts-<version-number> /sysconfig.txt ファイルにあります(< version-number > を initscripts パッケージのバージョンに置き換えます)。または、/etc/rc.d/ ディレクトリーで initscripts を調べると役に立つことが証明されます。

注記

ここにリストされているファイルの一部が /etc/sysconfig/ ディレクトリーに存在しない場合は、対応するプログラムがインストールされていない可能性があります。

32.1.1. /etc/sysconfig/amd

/etc/sysconfig/amd ファイルには、amd が使用するさまざまなパラメーターが含まれています。これらのパラメーターを使用すると、ファイルシステムの自動マウントとアンマウントが可能になります。

32.1.2. /etc/sysconfig/apmd

/etc/sysconfig/apmd ファイルは、pmd により使用され、中断または再開時に起動/停止/変更する電源設定を行います。このファイルは、ハードウェアが Advanced Power Management (APM)に対応しているかどうか、またはユーザーがシステムを設定してシステムを設定した場合に応じて、システムの起動時に apmd がどのように機能するかを設定します。apm デーモンは、Linux カーネル内の電源管理コードで動作する監視プログラムです。ノートパソコンやその他の電源関連の設定でバッテリーの電力を下げるようユーザーに警告することができます。

32.1.3. /etc/sysconfig/arpwatch

/etc/sysconfig/arpwatch ファイルは、起動時に arpwatch デーモンに引数を渡すために使用されます。arpwatch デーモンは、イーサネット MAC アドレスと IP アドレスのペアのテーブルを維持しま

す。デフォルトでは、このファイルは arpwatch プロセスの所有者をユーザー pcap に設定し、すべてのメッセージを root メールキューに送信します。このファイルで利用可能なパラメーターの詳細は、arpwatch の man ページを参照してください。

32.1.4. /etc/sysconfig/authconfig

/etc/sysconfig/authconfig ファイルは、ホストで使用される認証を設定します。これには、以下の行が 1 つ以上含まれます。

- *USEMD5= <value& gt;。ここで、& lt;value> は以下のいずれかになります。*
  - *はい - MD5 が認証に使用されます。*
  - *認証に MD5 は使用されません。*
- *USEKERBEROS= <value >。ここで、& lt;value& gt; は以下のいずれかになります。*
  - *はい: Kerberos が認証に使用されます。*
  - *いいえ: Kerberos は認証には使用されません。*
- *USELDAPAUTH= <value& gt;。ここで、& lt;value> は以下のいずれかになります。*
  - *はい: LDAP が認証に使用されます。*
  - *いいえ: 認証に LDAP は使用されません。*

32.1.5. /etc/sysconfig/autofs

/etc/sysconfig/autofs ファイルは、デバイスの自動マウントのカスタムオプションを定義します。このファイルは、自動マウントデーモンの動作を制御します。このデーモンは、ファイルシステムを使用する際に自動的にマウントし、非アクティブになってからアンマウントします。ファイルシステムに

は、ネットワークファイルシステム、CD-ROM、ディスケット、およびその他のメディアを含めることができます。

/etc/sysconfig/autofs ファイルには以下が含まれる場合があります。

- LOCALOPTIONS=" <value> ", where & lt;value& gt; はマシン固有の自動マウントルールを定義する文字列です。デフォルト値は空の文字列("")です。
- DAEMONOPTIONS=" <value> "。ここで、< value > はデバイスをアンマウントする前のタイムアウトの長さ（秒単位）です。デフォルト値は 60 秒("--timeout=60")です。
- UNDERSCORETODOT= <value > です。ここで、< value > は、ファイル名のアンダースコアをドットに変換するかどうかを制御するバイナリー値です。たとえば、auto_home を auto.home に、auto_mnt を auto.mnt に、それぞれ設定します。デフォルト値は 1 (true)です。
- DISABLE_DIRECT= <value > です。ここで、< value > は、直接マウントサポートを無効にするかどうかを制御するバイナリー値です。Linux 実装は Sun Microsystems の自動マウント機能に準拠しないためです。デフォルト値は 1 (true)で、Sun 自動マウント機能オプションの仕様構文との互換性を許可します。

32.1.6. /etc/sysconfig/clock

/etc/sysconfig/clock ファイルは、システムハードウェアクロックから読み取られる値の解釈を制御します。

正しい値は以下のとおりです。

- utc = <value>。ここで、<value& gt; は以下のブール値のいずれかになります。
  - true または yes: ハードウェアクロックは Universal Time に設定されます。
  - false または no: ハードウェアクロックはローカルタイムに設定されます。

- ARC= <value>。ここで、& lt;value> は以下になります。
  - false または no - この値は、通常の UNIX エポックが使用されていることを示します。その他の値は、Red Hat Enterprise Linux でサポートされていないシステムで使用されます。

- sm = <value& gt;。ここで、&lt ;value> は以下になります。
  - false または no - この値は、通常の UNIX エポックが使用されていることを示します。その他の値は、Red Hat Enterprise Linux でサポートされていないシステムで使用されます。

- ZONE= <filename > - /etc/localtime がコピーされている /usr/share/zoneinfo の下にあるタイムゾーンファイルです。ファイルには、以下のような情報が含まれています。

```
ZONE="America/New York"
```

  ZONE パラメーターは Time and Date Properties Tool (system-config-date)によって読み取られ、手動で編集してもシステムのタイムゾーンは変更されません。

Red Hat Enterprise Linux の以前のリリースでは、以下の値を使用していました（非推奨）。

- CLOCKMODE= <value >。ここで、& lt;value& gt; は以下のいずれかになります。
  - GMT - クロックは Universal Time （グリニッジ標準時）に設定されます。
  - ARC - ARC コンソールの 42-year 時間オフセットが有効(Alpha ベースのシステムのみ)。

32.1.7. /etc/sysconfig/desktop

/etc/sysconfig/desktop ファイルは、新しいユーザーのデスクトップと、ランレベル 5 に入る際に実行するディスプレイマネージャーを指定します。

正しい値は次のとおりです。

- DESKTOP=" <value> "。" <value>" は以下のいずれかになります。
  - GNOME - GNOME デスクトップ環境を選択します。
  - kde: KDE デスクトップ環境を選択します。
- DISPLAYMANAGER=" <value> "。" <value>" は以下のいずれかになります。
  - GNOME - GNOME Display Manager を選択します。
  - kde: KDE Display Manager を選択します。
  - XDM - X Display Manager を選択します。

詳細は、35章X Window System を参照してください。

32.1.8. /etc/sysconfig/dhcpd

/etc/sysconfig/dhcpd ファイルは、システムの起動時に dhcpd デーモンに引数を渡すために使用されます。dhcpd デーモンは、DHCP (Dynamic Host Configuration Protocol)および Internet Bootstrap Protocol (BOOTP)を実装します。DHCP および BOOTP は、ネットワーク上のマシンにホスト名を割り当てます。このファイルで利用可能なパラメーターの詳細は、dhcpd の man ページを参照してください。

32.1.9. /etc/sysconfig/exim

/etc/sysconfig/exim ファイルを使用すると、メッセージを 1 つ以上のクライアントに送信し、必要

なネットワーク上でメッセージをルーティングできます。ファイルは、exim の実行のデフォルト値を設定します。デフォルト値は、バックグラウンドデーモンとして実行され、何かがバックアップされている場合に 1 時間ごとにキューを確認するように設定されています。

値には以下が含まれます。

- DAEMON= <value>。ここで、<value& gt; は以下のいずれかになります。
  - はい: 受信メール で ポート 25 をリッスンするように設定する必要があります。yes は、Exim's -bd オプションの使用を意味します。
  - no: 受信メールのポート 25 をリッスンするように設定しないでください。
- QUEUE=1h は -q$ QUEUE の ように指定されます。-q オプション は、/etc/sysconfig/exim が存在し、QUEUE が空または定義されているかを示すように指定されません。

32.1.10. /etc/sysconfig/firstboot

システムの初回起動時に、/sbin/init プログラムが etc/rc.d/init.d/firstboot スクリプトを呼び出して、 セットアップエージェント を起動します。このアプリケーションを使用すると、ユーザーは最新の更新や追加のアプリケーションおよびドキュメントをインストールできます。

/etc/sysconfig/firstboot ファイルは、 設定エージェント アプリケーションが後続の再起動では実行されないようにします。次回システムを起動したときに実行するには、/etc/sysconfig/firstboot を削除し、で chkconfig --level 5 firstboot を実行します。

32.1.11. /etc/sysconfig/gpm

/etc/sysconfig/gpm ファイルは、起動時に gpm デーモンに引数を渡すために使用されます。gpm デーモンはマウスサーバーであり、マウスアクセラレーションとマウスの途中クリックの貼り付けを可能にします。このファイルで利用可能なパラメーターの詳細は、glock の man ページを 参照し てください。デフォルトでは、DEVICE ディレクティブは /dev/input/mice に設定されます。

32.1.12. /etc/sysconfig/hwconf

/etc/sysconfig/hwconf ファイルには、kudzu がシステムで検出されたすべてのハードウェアと、使用されているドライバー、ベンダー ID、およびデバイス ID 情報が一覧表示されます。kudzu プログ

ラムは、システム上の新しいハードウェアや変更されたハードウェアを検出し、設定します。/etc/sysconfig/hwconf ファイルは手動で編集することは意図されていません。編集されると、デバイスは追加または削除中と突然表示される可能性があります。

32.1.13. /etc/sysconfig/i18n

/etc/sysconfig/i18n ファイルは、デフォルトの言語、サポートされている言語、およびデフォルトのシステムフォントを設定します。以下に例を示します。

```
LANG="en_US.UTF-8"
SUPPORTED="en_US.UTF-8:en_US:en"
SYSFONT="latarcyrheb-sun16"
```

32.1.14. /etc/sysconfig/init

/etc/sysconfig/init ファイルは、システムの起動プロセス中にシステムがどのように表示され、機能するかを制御します。

以下の値を使用できます。

- BOOTUP= <value >。ここで、& lt;value& gt; は以下のいずれかになります。
  - COL: 標準の色ブート表示で、デバイスおよびサービスの起動の成功または失敗が色分けされます。
  - verbose: 正常または失敗のメッセージよりも、より多くの情報を提供する古いスタイル表示。
  - それ以外のものは新しい表示を意味しますが、ANSI 形式はありません。
- RES_COL= <value >。ここで、< value > はステータスラベルを開始する画面のコラムの数に置き換えます。デフォルトでは 60 に設定されます。
- MOVE_TO_COL= <value > です。ここで、< value > は echo -en コマンドを使用して、カーソルを RES_COL 行の値に移動します。
-

SE --------|-----LOR_SUCCESS= <value > です。ここで、< value > は echo -en コマンドを使用して成功の色を設定します。デフォルトの色は green に設定されています。

- SE ----------|-----LOR_FAILURE= <value > です。ここで、< value > は echo -en コマンドを使用して障害の色を設定します。デフォルトの色は red に設定されます。

- SE --------|-----LOR_WARNING= <value > です。< value > は echo -en コマンドを使用して警告の色を設定します。デフォルトの色は yellow に設定されます。

- SENORMALLOR_NORMAL= <value& gt; です。ここで、< value > は echo -en で色を標準にリセットします。

- LOGLEVEL= <value>。ここで、&lt ;value > はカーネルの初期コンソールロギングレベルを設定します。デフォルトは 3 です。8 はすべての（デバッグを含む）を意味します。1 はカーネルパニックのみを意味します。syslogd デーモンは、この設定が開始されたら上書きされます。

- KeepAlive = <value>。ここで、<value& gt; は以下のブール値のいずれかになります。

  - yes: インタラクティブモードのキーチェックを有効にします。

  - no: インタラクティブモードのキーチェックを無効にします。

32.1.15. /etc/sysconfig/ip6tables-config

/etc/sysconfig/ip6tables-config ファイルは、カーネルが起動時に IPv6 パケットフィルターリングを設定するのに使用する情報、または ip6tables サービスが起動するたびに保存されます。

ip6tables ルールの作成方法に慣れていない限り、このファイルを手動で変更しないでください。ルールは、/sbin/ip6tables コマンドを使用して手動で作成することもできます。作成したら、以下のコマンドを入力して、ルールを /etc/sysconfig/ip6tables ファイルに追加します。

```
service ip6tables save
```

このファイルが存在する場合は、システム再起動またはサービスの再起動時に、保存されているファイアウォールルールが維持されます。

ip6tables の詳細は、「iptables」 を参照してください。

32.1.16. /etc/sysconfig/iptables-config

/etc/sysconfig/iptables-config ファイルは、ブート時またはサービスが起動するたびに、カーネルがパケットフィルターリングサービスを設定するのに使用される情報を保存します。

iptables ルールの構築に精通していない限り、このファイルは手動で変更しないでください。ルールを追加する最も簡単な方法は、Security Level Configuration Tool (system-config-securitylevel)アプリケーションを使用してファイアウォールを作成することです。これらのアプリケーションは、プロセスの最後にこのファイルを自動的に編集します。

ルールは、/sbin/iptables コマンドを使用して手動で作成することもできます。作成したら、以下のコマンドを入力してルールを /etc/sysconfig/iptables ファイルに追加します。

```
service iptables save
```

このファイルが存在する場合は、システム再起動またはサービスの再起動時に、保存されているファイアウォールルールが維持されます。

iptables の詳細は、「iptables」 を参照してください。

32.1.17. /etc/sysconfig/irda

/etc/sysconfig/irda ファイルは、起動時にシステム上のインフラストラクチャーデバイスを設定する方法を制御します。

以下の値を使用できます。

- IRDA= <value >。ここで、& lt;value& gt; は以下のブール値のいずれかになります。
  - はい - 無 線は 実行され、ネットワーク接続を試みる別のノートブックコンピューターなど、インフラストラクチャーポートへの接続を試みているかどうかを確認するため

に定期的にチェックします。インフラストラクチャーデバイスがシステムで機能するには、この行を yes に設定する必要があります。

- no - irattach は実行されず、infrad デバイス通信を防ぎます。

- DEVICE= <value& gt;。ここで、<value > はインフラストラクチャー接続を処理するデバイス（通常はシリアルポート）です。シリアルデバイスエントリーの例は、/dev/ttyS2 です。

- DON GLE= <value> です。ここで、< value > は、infrared 通信に使用されるドバルのタイプを指定します。この設定は、実際のインフラポートではなくシリアルドアグラムを使用するユーザーに存在します。dongle は、従来のシリアルポートに接続され、infrared 経由で通信するデバイスです。実際のインフラポートを持つノートブックは、アドオンのドロングを持つコンピューターよりもはるかに一般的であるため、この行はデフォルトではコメントアウトされています。サンプル dongle エントリーは actisys+ です。

- DISCOVERY= <value>。ここで、<value& gt; は以下のブール値のいずれかになります。

  - Yes: 検出モードで irattach を開始します。つまり、他のインフラされたデバイスをアクティブにチェックします。マシンが infrared 接続をアクティブに検索するには、これを有効にする必要があります（つまり、接続を開始しないピア）。

  - no: 検出モードで irattach を開始しません。

32.1.18. /etc/sysconfig/kernel

/etc/sysconfig/kernel 設定ファイルは、システムの起動時にカーネルの選択を制御します。以下のデフォルト値を持つ 2 つのオプションがあります。

UPDATEDEFAULT=yes

このオプションを使用すると、新規インストールしたカーネルがブートエントリーのデフォルトとして選択されます。

DEFAULTKERNEL=kernel

このオプションは、デフォルトとして使用するパッケージタイプを指定します。

32.1.18.1. 古いカーネルバージョンをデフォルトとして維持する

ブートエントリーの選択で、以前のカーネルバージョンをデフォルトのままにするには、以下を実行します。

- *以下のように、/etc/sysconfig/kernel の UPDATEDEFAULT オプションをコメントアウトします。*

```
# UPDATEDEFAULT=yes
```

32.1.18.2. カーネルデバッガーのデフォルトカーネルとしての設定

ブートエントリーの選択で、カーネルデバッガーをデフォルトのカーネルとして設定するには、以下を実行します。

- *以下のように /etc/sysconfig/kernel 設定ファイルを編集します。*

```
DEFAULTKERNEL=kernel-debug
```

32.1.19. /etc/sysconfig/keyboard

/etc/sysconfig/keyboard ファイルはキーボードの動作を制御します。以下の値を使用できます。

- *KEYBOARDTYPE="sun|pc" の場合、sun は Sun キーボードが /dev/kbd にアタッチされ、pc は PS/2 ポートに接続された PS/2 キーボードを意味します。*

- *KEYTABLE=" <file> "。ここ で、<file > はキー割ファイルの名前です。*

  例： KEYTABLE="us"キーテーブルとして使用できるファイルは、/lib/kbd/keymaps/i386 で始まり、そこから異なるキーボードレイアウトにまとめられるので、< file > .kmap.gz というラベルが付いています。KEYTABLE 設定に一致する /lib/kbd/keymaps/i386 の下にある最初のファイルが使用されます。

32.1.20. /etc/sysconfig/kudzu

/etc/sysconfig/kuzdu ファイルは、システムの起動時に kudzu によってシステムハードウェアの安全なプローブをトリガーします。安全なプローブは、シリアルポートプロービングを無効にする 1 つです。

- SAFE= <value >。ここで、& lt;value& gt; は以下のいずれかになります。
  - はい - kuzdu は安全なプローブを実行します。
  - no: kuzdu は通常のプローブを実行します。

32.1.21. /etc/sysconfig/named

/etc/sysconfig/ named ファイルは、起動時に名前付きデーモンに引数を渡すために使用されます。名前付き デーモンは、BIND バージョン 9 ディストリビューションである Berkeley Internet Name Domain (BIND)を実装する Domain Name System (DNS)サーバーです。このサーバーは、ネットワーク上の IP アドレスに関連付けられたホスト名の表を維持します。

現在、以下の値のみを使用できます。

- ROOTDIR="</some/where>" で、</ some/where > は、named が実行される設定済みの chroot 環境のフルディレクトリーパスを指します。最初に、この chroot 環境を設定する必要があります。詳細は、info chroot と入力します。
- OPTIONS="<value>"。ここで、< value > は named except -t の man ページに記載されているオプションです。-t の代わりに、上記の ROOTDIR 行を使用します。

このファイルで利用可能なパラメーターの詳細は、man ページの named を参照してください。BIND DNS サーバーの設定方法は、19章 を参照してください。デフォルトでは、ファイルにはパラメーターは含まれません。

32.1.22. /etc/sysconfig/network

/etc/sysconfig/network ファイルは、必要なネットワーク設定に関する情報を指定するために使用されます。以下の値を使用できます。

- NETWORKING= <value>。ここで、<value& gt; は以下のブール値のいずれかになります。
  - はい: ネットワークを設定する必要があります。
  - いいえ: ネットワークを設定しないでください。
- HOSTNAME= <value >。ここで、< value > は 完全修飾ドメイン 名(FQDN)である必要があります（例： hostname.expample.com ）。必要なホスト名を指定できます。
- GATEWAY= <value>。ここ で、<value > はネットワークのゲートウェイの IP アドレスになります。
- GATEWAYDEV= <value>。ここ で、<value > は eth0 などのゲートウェイデバイスです。同じサブネットに複数のインターフェイスがあり、それらのインターフェイスの 1 つがデフォルトゲートウェイへの優先ルートに必要な場合に、このオプションを設定します。
- NISDOMAIN= <value>。ここで、<value& gt; は NIS ドメイン名に置き換えます。
- NOZEROCONF= <value> です。< value> を true に設定すると、zeroconf ルートが無効になります。

デフォルトでは、zeroconf ルート(169.254.0.0)はシステムの起動時に有効になります。zeroconf の詳細は、を参照して http://www.zeroconf.org/ ください。

WARNING

カスタム initscripts を使用してネットワーク設定を設定しないでください。起動後のネットワークサービスの再起動時に、ネットワーク init スクリプト外で実行されるネットワーク設定をカスタムの initscripts 設定すると、予測不可能な結果になります。

32.1.23. /etc/sysconfig/nfs

NFS には、RPC サービスにポートを動的に割り当てる portmap が必要です。これにより、ファイアウォールルールの設定で問題が発生します。この問題を解決するには、/etc/sysconfig/nfs ファイルを使用して、必要な RPC サービスを実行するポートを制御します。

/etc/sysconfig/nfs は、デフォルトですべてのシステムに存在するわけではありません。存在しない場合は、これを作成して以下の変数を追加します（または、ファイルが存在する場合は、コメントを解除し、必要に応じてデフォルトのエントリーを変更します）。

MOUNTD_PORT=x

mountd (rpc.mountd)が使用する TCP および UDP ポートを制御します。x を未使用のポート番号に置き換えます。

STATD_PORT=x

ステータス(rpc.statd)が使用する TCP および UDP ポートを制御します。x を未使用のポート番号に置き換えます。

LOCKD_TCPPORT=x

nlockmgr (rpc.lockd)が使用する TCP ポートを制御します。x を未使用のポート番号に置き換えます。

LOCKD_UDPPORT=x

nlockmgr (rpc.lockd)が使用する UDP ポートを制御します。x を未使用のポート番号に置き換えます。

NFS が起動しない場合は、/var/log/messages を確認してください。通常、すでに使用されているポート番号を指定すると、NFS は起動に失敗します。/etc/sysconfig/nfs を編集した後、service nfs restart コマンドを実行して NFS サービスを再起動します。rpcinfo -p コマンドを実行して、変更を確認します。

NFS を許可するようにファイアウォールを設定するには、以下を実行します。

1. NFS 用に TCP および UDP ポート 2049 を許可します。

2. TCP および UDP ポート 111 (portmap/sunrpc)を許可します。

3. MOUNTD_PORT="x"で指定した TCP および UDP ポートを許可します。

4. STATD_PORT="x"で指定した TCP および UDP ポートを許可します。

5. LOCKD_TCPPORT="x"で指定した TCP ポートを許可します。

6. LOCKD_UDPPORT="x"で指定した UDP ポートを許可します。

32.1.24. /etc/sysconfig/ntpd

/etc/sysconfig/ntpd ファイルは、起動時に ntpd デーモンに引数を渡すために使用されます。ntpd デーモンは、システムクロックを設定して維持し、インターネット標準タイムサーバーと同期します。Network Time Protocol (NTP)のバージョン 4 を実装します。このファイルで利用可能なパラメーターの詳細は、Web ブラウザーを使用して /usr/share/doc/ntp- <version> /ntpd.htm （ここでの <version > は ntpdのバージョン番号）を表示します。デフォルトでは、このファイルは ntpd プロセスの所有者をユーザー ntp に設定します。

32.1.25. /etc/sysconfig/radvd

/etc/sysconfig/radvd ファイルは、システムの起動時に radvd デーモンに引数を渡すために使用されます。radvd デーモンは、ルーターの要求をリッスンし、IP バージョン 6 プロトコルのルーター広告を送信します。このサービスは、ネットワーク上のホストがこれらのルーター通知に基づいてデフォルトのルーターを動的に変更できるようにします。このファイルで利用可能なパラメーターの詳細は、radvd の man ページを参照してください。デフォルトでは、このファイルは radvd プロセスの所有者をユーザー radvd に設定します。

32.1.26. /etc/sysconfig/samba

/etc/sysconfig/samba ファイルは、起動時に smbd および nmbd デーモンに引数を渡すために使用されます。smbd デーモンは、ネットワーク上の Windows クライアントのファイル共有接続を提供します。nmbd デーモンは、IP 命名サービス上で NetBIOS を提供します。このファイルで利用可能なパラメーターの詳細は、smbd の man ページを参照してください。デフォルトでは、このファイルは smbd および nmbd をデーモンモードで実行するように設定します。

32.1.27. /etc/sysconfig/selinux

/etc/sysconfig/selinux ファイルには、SELinux の基本的な設定オプションが含まれています。このファイルは、/etc/selinux/config へのシンボリックリンクです。

32.1.28. /etc/sysconfig/sendmail

/etc/sysconfig/sendmail ファイルを使用すると、1 つ以上のクライアントにメッセージを送信し、必要なネットワーク上でメッセージをルーティングできます。ファイルは、Sendmail アプリケーションを実行するデフォルト値を設定します。デフォルト値は、バックグラウンドデーモンとして実行され、何かがバックアップされている場合に 1 時間ごとにキューを確認するように設定されています。

値には以下が含まれます。

- DAEMON= <value>。ここで、<value& gt; は以下のいずれかになります。
  - はい: Sendmail は、受信メールのポート 25 をリッスンするように設定する必要があります。yes は、Sendmail の-bd オプションを使用することを意味します。
  - no: Sendmail は、受信メールのポート 25 をリッスンするように設定しないでください。
- Sendmail に -q$ QUEUE として提供される QUEUE=1h。/etc/sysconfig/sendmail が存在し、QUEUE が空または未定義の場合、-q オプションは Sendmail に指定されません。

32.1.29. /etc/sysconfig/spamassassin

/etc/sysconfig/spamassassin ファイルは、システムの起動時に spamd デーモン（デーモン化されたバージョンの Spamassassin）に引数を渡すために使用されます。SpamAssassin は、電子メールのスパムフィルターアプリケーションです。利用可能なオプションの一覧は、spamd の man ページを参照してください。デフォルトでは、spamd がデーモンモードで実行し、ユーザー設定を作成し、自動作成ホワイトリスト（一括送信者を許可）を設定します。

Spamassassin の詳細は、「spam フィルター」 を参照してください。

32.1.30. /etc/sysconfig/squid

/etc/sysconfig/squid ファイルは、起動時に squid デーモンに引数を渡すために使用されます。squid デーモンは、Web クライアントアプリケーションのプロキシーキャッシュサーバーです。squid プロキシーサーバーの設定に関する詳細は、Web ブラウザーを使用して /usr/share/doc/squid- <version> / ディレクトリーを開きます。<version > は、システムにインストールされている squid バージョン番号に置き換えてください。デフォルトでは、このファイルは squid がデーモンモードで起動するように設定し、それ自体をシャットダウンするまでの時間を設定します。

32.1.31. /etc/sysconfig/system-config-securitylevel

/etc/sysconfig/system-config-securitylevel ファイルには、ユーザーが選択したすべてのオプションが含まれます(system-config-securitylevel)。ユーザーは手動でこのファイルを変更しないでください。Security Level Configuration Tool の詳細は、「ファイアウォールの基本設定」 を参照してください。

32.1.32. /etc/sysconfig/system-config-selinux

/etc/sysconfig/system-config-selinux ファイルには、SELinux Administration Tool (system-config-selinux)が最後に実行されたときにユーザーが選択するすべてのオプションが含まれます。ユーザーは手動でこのファイルを変更しないでください。SELinux 管理ツールと SELinux 全般に関する詳細情報は、「SELinux の概要」 を参照してください。

32.1.33. /etc/sysconfig/system-config-users

/etc/sysconfig/system-config-users ファイルは、グラフィカルアプリケーション User Manager の設定ファイルです。このファイルは、root、デーモン、lp などのシステムユーザーを除外するために使用されます。このファイルは、 User Manager アプリケーションの Preferences > Filter system users and groups プルダウンメニューによって編集されるため、手動で編集することはできません。このアプリケーションの使用に関する詳細は、「」 を参照してください。

32.1.34. /etc/sysconfig/system-logviewer

/etc/sysconfig/system-logviewer ファイルは、グラフィカル、インタラクティブなログ表示アプリケーション Log Viewer の設定ファイルです。このファイルは、ログビューアー アプリケーションの Edit > Preferences プルダウンメニューによって編集されるため、手動で編集しないでください。このアプリケーションの使用に関する詳細は、40章ログファイル を参照してください。

32.1.35. /etc/sysconfig/tux

/etc/sysconfig/tux ファイルは、カーネルベースの Web サーバーである Red Hat Content Accelerator （以前は TUXと呼ばれていました）の設定ファイルです。Red Hat Content Accelerator の設定に関する詳細は、Web ブラウザーを使用して /usr/share/doc/tux- <version> /tux/index.html ファイルを開きます。 < version > は、システムにインストールされている TUX のバージョン番号に置き換えてください。このファイルで利用可能なパラメーターは、/usr/share/doc/tux- <version> /tux/parameters.html に一覧表示されます。

32.1.36. /etc/sysconfig/vncservers

/etc/sysconfig/vncservers ファイルは、仮想ネットワークコンピューティング (VNC)サーバーを起動する方法を設定します。

VNC は、ユーザーが実行中のマシンだけでなく、さまざまなアーキテクチャー上の異なるネットワーク全体でデスクトップ環境を表示できるリモートディスプレイシステムです。

これには、以下が含まれる場合があります。

- VNCSERVERS= <value >。ここで、< value > は "1 :fred" のように設定され、ディスプレイ :1 でユーザー fred に対して VNC サーバーを起動する必要があることを示します。リモート VNC サーバーへの接続を試みる前に、ユーザー fred が vncpasswd コマンドを使用して VNC パスワードを設定する必要があります。

32.1.37. /etc/sysconfig/xinetd

/etc/sysconfig/xinetd ファイルは、起動時に xinetd デーモンに引数を渡すために使用されます。xinetd デーモンは、そのサービスのポートへのリクエストを受け取ると、インターネットサービスを提供するプログラムを起動します。このファイルで利用可能なパラメーターの詳細は、xinetd の man ページを参照してください。xinetd サービスの詳細は、「xinetd」 を参照してください。

32.2. /ETC/SYSCONFIG/ ディレクトリーのディレクトリー

以下のディレクトリーは通常 /etc/sysconfig/ にあります。

apm-scripts/

このディレクトリーには、APM suspend/resume スクリプトが含まれます。ファイルを直接編集しないでください。カスタマイズが必要な場合は、スクリプトの最後に呼び出される /etc/sysconfig/apm-scripts/apmcontinue という名前のファイルを作成します。/etc/sysconfig/apmd を編集してスクリプトを制御することもできます。

cbq/

このディレクトリーには、ネットワークインターフェイスの帯域幅管理にクラス ベースの Queuing を 実行するために必要な設定ファイルが含まれています。CBQ は、IP アドレス、プロトコル、およびアプリケーションタイプの任意の組み合わせに基づいて、ユーザートラフィックをクラスの階層に分割します。

networking/

このディレクトリーは Network Administration Tool (system-config-network)によって使用され、その内容は手動で編集しないでください。Network Administration Tool を使用したネットワークインターフェイスの設定に関する詳細は、17章Network Configuration を参照してください。

network-scripts/

このディレクトリーには、以下のネットワーク関連の設定ファイルが含まれます。

- eth0 イーサネットインターフェイスの ifcfg-eth0 など、設定済みの各ネットワークインターフェイスのネットワーク設定ファイル。
- ifup や ifdown などのネットワークインターフェイスをアップまたはダウンするために使用されるスクリプト。
- ifup-isdn や ifdown-isdn などの ISDN インターフェイスを起動するために使用されるスクリプト。
- 直接編集してはならないさまざまな共有ネットワーク機能スクリプト。

network-scripts ディレクトリーの詳細は、16章Network Interfaces を参照してください。

rhn/

非推奨。このディレクトリーには、RHN Classic コンテンツサービスで使用される設定ファイルおよび GPG キーが含まれます。このディレクトリー内のファイルは手動で編集する必要はありません。

このディレクトリーは、RHN Classic によって管理されているレガシーシステムで利用できます。証明書ベースの Red Hat Network に対して登録されたシステムは、このディレクトリーを使用しません。

32.3. 関連情報

本章では、/etc/sysconfig/ ディレクトリー内のファイルの概要のみを目的としています。以下のソースには、より包括的な情報が含まれます。

32.3.1. インストールされているドキュメント

- /usr/share/doc/initscripts- <version-number> /sysconfig.txt: このファイルには、/etc/sysconfig/ ディレクトリーにあるファイルのより信頼できる一覧と、それらに使用できる設定オプションが含まれています。このファイル のパスにある <version-number > は、インストールされている initscripts パッケージのバージョンに対応します。

第33章 日付と時刻の設定

日時のプロパティーツールを使用すると、ユーザーはシステムの日付と時刻を変更し、システムが使用するタイムゾーンを設定し、システムクロックをタイムサーバーと同期するように Network Time Protocol (NTP)デーモンを設定できます。

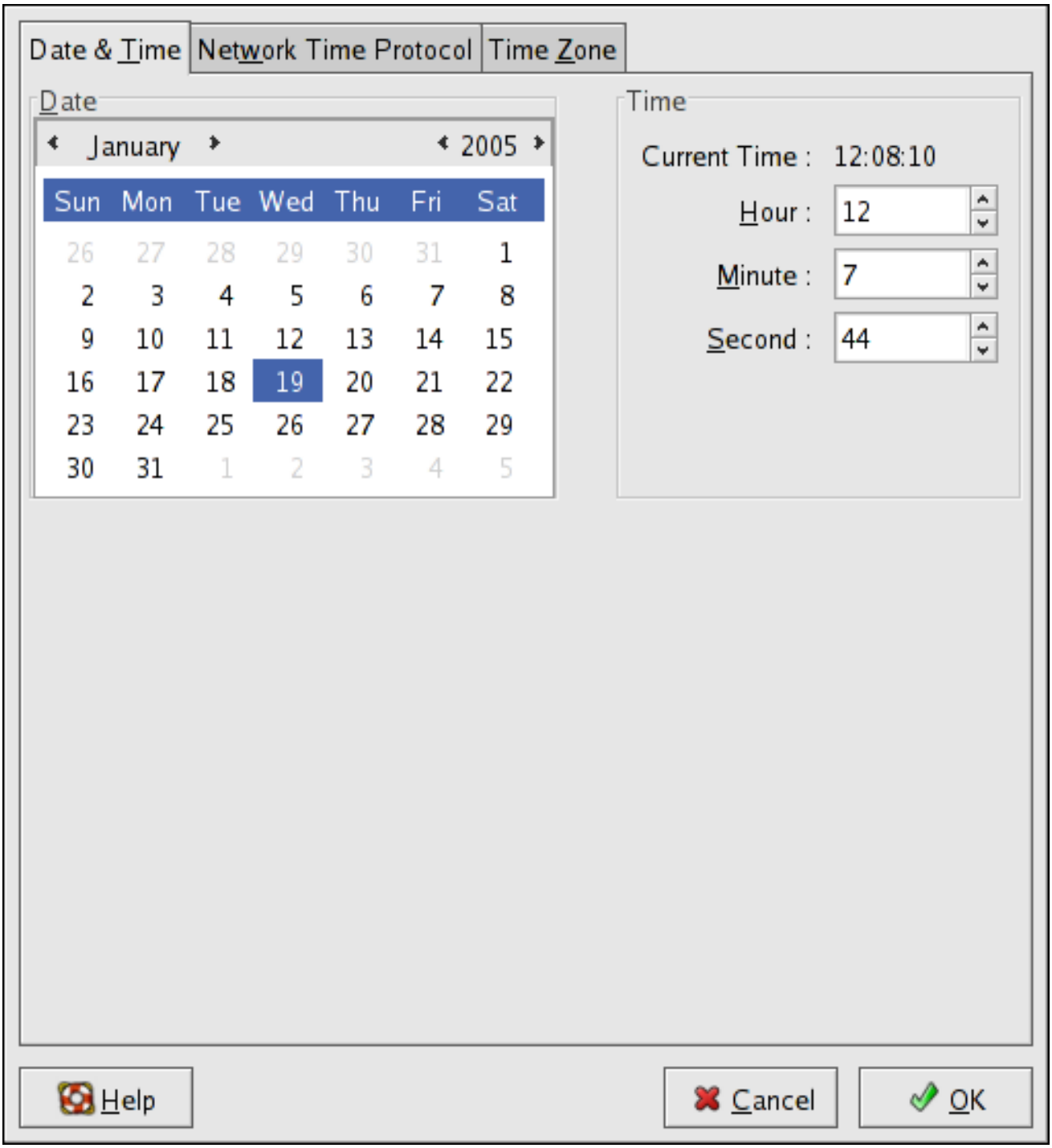
X Window System を実行しており、ツールを使用するには root 権限が必要です。アプリケーションを起動するには、以下の 3 つの方法があります。

- デスクトップから アプリケーション（パネルのメインメニュー）> System Settings > Date & Timeに移動します。
- デスクトップからツールバーで時間を右クリックし、Adjust Date and Time を選択します。
- コマンド system-config-date、system-config-time、または dateconfig をシェルプロンプト(XTerm や GNOME ターミナルなど)に入力します。

33.1. 日時のプロパティー

図33.1「日時のプロパティー」 に示すように、表示される最初のタブはシステムの日付と時刻を設定するために使用します。

図33.1 日時のプロパティー



[D]

日付を変更するには、月の左にある矢印と、月の右にある矢印を使用して年を変更し、曜日をクリックして曜日を変更します。

時間を変更するには、Time セクションの Hour、Minute、および Second の横にある上下の矢印ボタンを使用します。

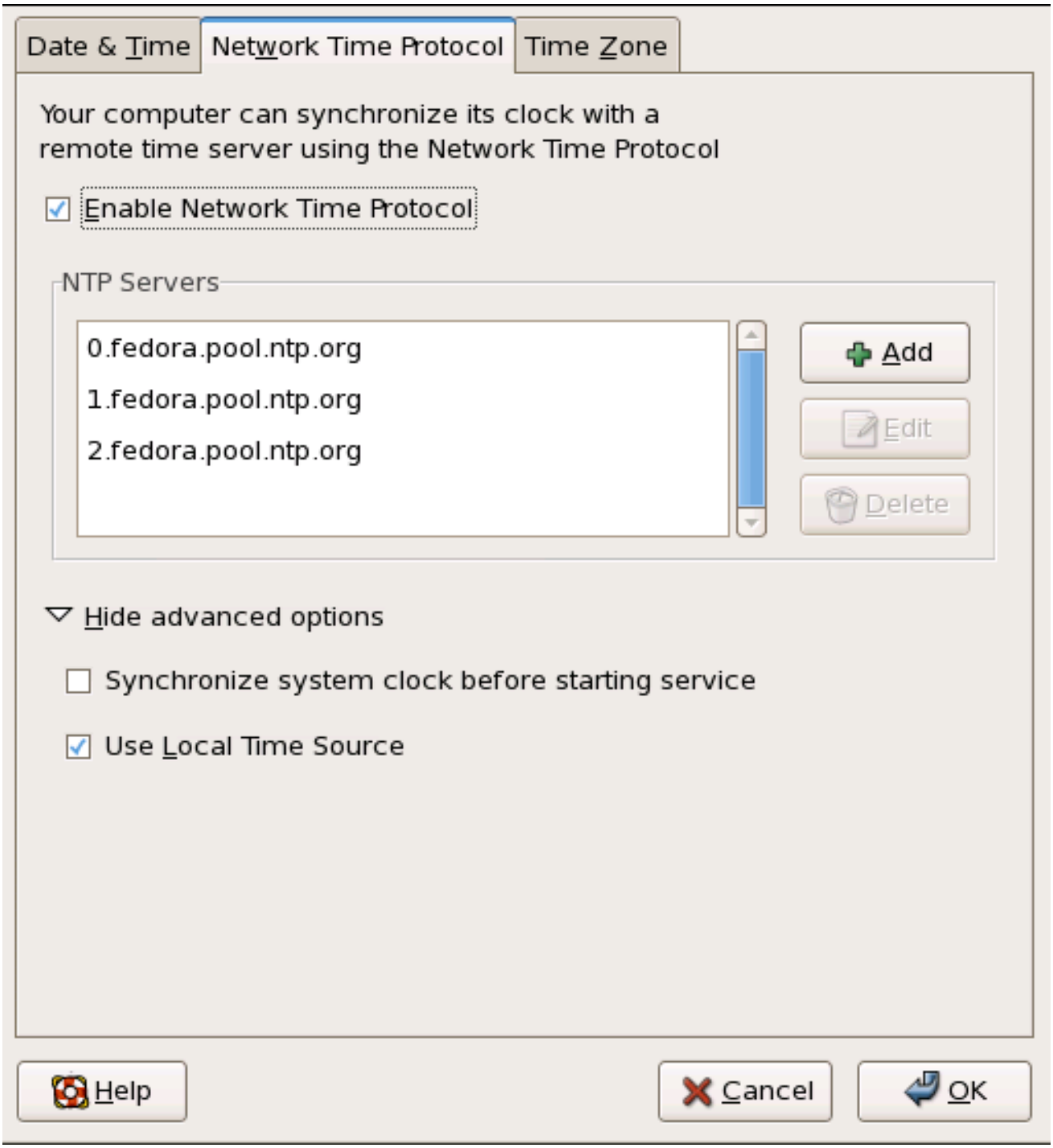
OK ボタンをクリックすると、日付と時刻、NTP デーモン設定、タイムゾーン設定に加えた変更が

適用されます。また、プログラムも終了します。

33.2. ネットワークタイムプロトコル(NTP)プロパティー

図33.2「NTP プロパティー」 に示すように、表示される 2 番目のタブドウィンドウは、NTP を設定するためのものです。

図33.2 NTP プロパティー



[D]

Network Time Protocol (NTP)デーモンは、システムクロックをリモートタイムサーバーまたはタイ

ムソースと同期します。このアプリケーションでは、システムクロックをリモートサーバーと同期するように NTP デーモンを設定できます。この機能を有効にするには、Enable Network Time Protocol を選択します。これにより、NTP サーバー の一覧およびその他のオプションが有効になります。事前定義されたサーバーのいずれかを選択し、Edit をクリックして事前に定義されたサーバーを編集するか、Add をクリックして新しいサーバー名を追加します。OK をクリックするまで、システムは NTP サーバーとの同期を開始しません。OK をクリックすると、設定が保存され、NTP デーモンが起動します（すでに実行中の場合は再起動）。

OK ボタンをクリックすると、日付と時刻、NTP デーモン設定、タイムゾーン設定に加えた変更が適用されます。また、プログラムも終了します。

33.3. タイムゾーンの設定

表示される 3 番目のタブドウィンドウは、システムのタイムゾーンを設定することです。

システムのタイムゾーンを設定するには、タイム ゾーン タブをクリックします。タイムゾーンは、インタラクティブマップを使用するか、マップの下のリストから希望するタイムゾーンを選択して変更できます。マップを使用するには、該当するリージョンをクリックします。このマップは、選択したリージョンにズームされ、その後にタイムゾーンに固有の都市を選択できます。赤い X が表示され、マップの下にある一覧内のタイムゾーンの選択肢が変更します。

または、マップの下にあるリストを使用することもできます。都市を選択する前に地図を選択できるのと同じ方法で、タイムゾーンの一覧はツリーリストになり、特定の大事に分類されています。また、サイエンティックコミュニティーのニーズに対応するために、配列以外のタイムゾーンも追加されました。

OK をクリックし変更を適用して、プログラムを終了します。

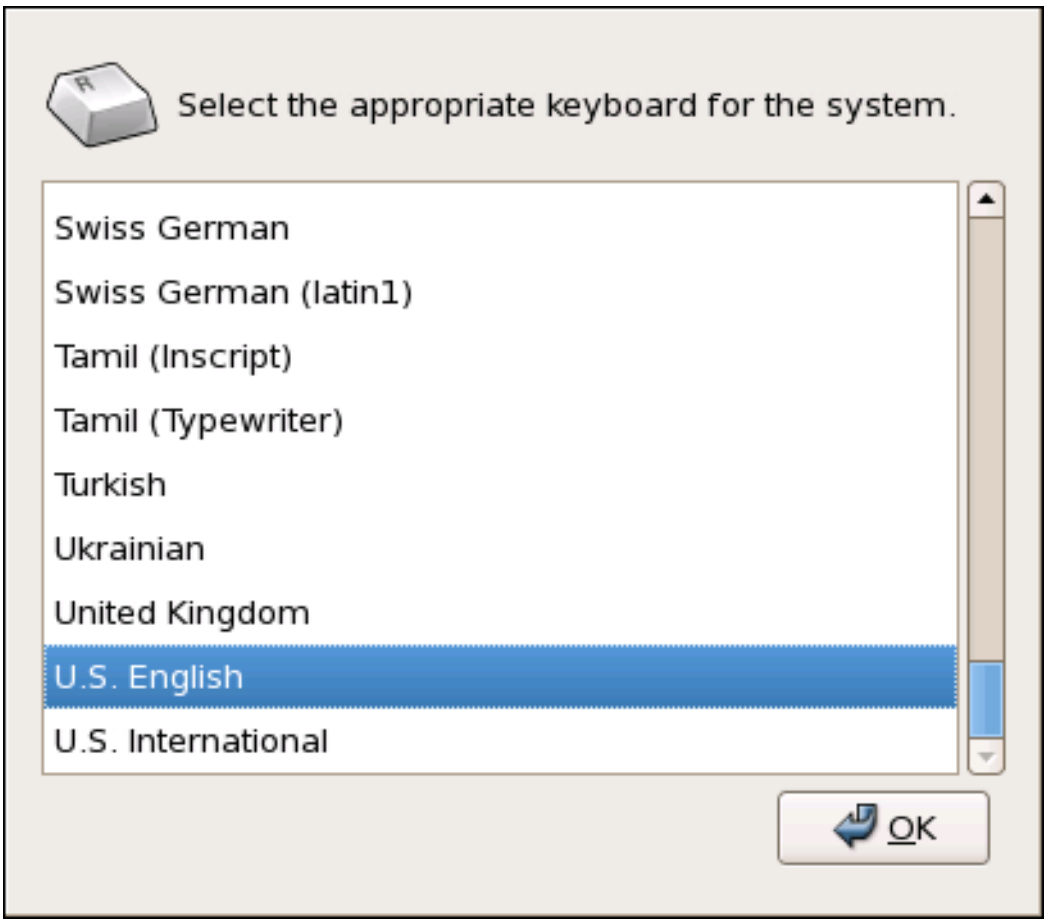
システムクロックが UTC を使用するように設定されている場合は、システム クロックで UTC を使用 するオプションを選択します。UTC は Universal Time, Coordinated の略で、グリニッジ標準時 (GMT)とも呼ばれます。他のタイムゾーンは UTC 時間にプラス/マイナスすることで割り出されます。

第34章 キーボードの設定

インストールプログラムを使用すると、システムのキーボードレイアウトを設定できます。インストール後に別のキーボードレイアウトを設定するには、Keyboard Configuration Tool を使用します。

Keyboard Configuration Tool を起動するには、System （パネルで）> Administration > Keyboard を選択するか、シェルプロンプトでコマンド system-config-keyboard を入力します。

図34.1 キーボード設定ツール



[D]

リストからキーボードレイアウトを選択します（例： U.S）。英語）をクリックし、OK をクリックします。

変更は直ちに反映されます。

第35章 X WINDOW SYSTEM

Red Hat Enterprise Linux の中心はカーネルですが、多くのユーザーにとって、オペレーティングシステムの直面は X Window System が提供するグラフィカル環境です(X とも呼ばれます)。

UNIX 世界には他のウィンドウ環境が存在しており、これには X Window System のリリースを 1984 年 6 月に規定するものも含まれていました。しかし、X は長年にわたり、UNIX 系のほとんどのオペレーティングシステム(Red Hat Enterprise Linux など)のデフォルトのグラフィカル環境になりました。

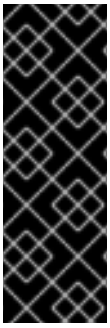
Red Hat Enterprise Linux のグラフィカル環境は、X Window System および関連するテクノロジーの開発およびストラテジーを管理するために作成されたオープンソース組織である X.Org Foundation により提供されます。X.Org は大規模な開発プロジェクトであり、世界中で数百の開発者が含まれるプロジェクトを迅速に開発しています。さまざまなハードウェアデバイスやアーキテクチャーに対する幅広いサポートを特長としており、さまざまなオペレーティングシステムやプラットフォームで実行できます。Red Hat Enterprise Linux の本リリースには、X Window System の X11R7.1 リリースが含まれます。

X Window System はクライアントサーバーアーキテクチャーを使用します。X サーバー (Xorg バイナリー)は、ネットワークまたはローカルループバックインターフェイスを介して X クライアント アプリケーションからの接続をリッスンします。サーバーは、ビデオカード、モニター、キーボード、マウスなどのハードウェアと通信します。X クライアントアプリケーションはユーザー空間に存在し、ユーザーに グラフィカルユーザーインターフェイス (GUI)を作成し、X サーバーにユーザー要求を渡します。

35.1. X11R7.1 リリース

Red Hat Enterprise Linux 5.10 は、X11R7.1 リリースをベース X Window System として使用するようになりました。これには、以前のリリースに対する複数のビデオドライバー、EXA、およびプラットフォームのサポートが含まれます。さらに、本リリースには、X サーバーの自動設定機能が複数含まれています。

X11R7.1 は、X Window System のモジュール化を具体的に活用するための最初のリリースです。X を論理的に個別のモジュールに分割するこのモジュール化により、オープンソース開発者がシステムにコードを簡単に提供できるようになります。



重要な影響

Red Hat Enterprise Linux は、XFree86™ サーバーパッケージを提供しなくなりました。システムを Red Hat Enterprise Linux の最新バージョンにアップグレードする前に、http://hardware.redhat.com/ でオンラインの Red Hat Hardware Compatibility List を確認して、システムのビデオカードが X11R7.1 リリースと互換性があることを確認してください。

X11R7.1 リリースでは、すべてのライブラリー、ヘッダー、およびバイナリーは、/usr/ X11R6 ではなく /usr/ 下に置かれるようになりました。/etc/X11/ ディレクトリーに は、X クライアントおよびサーバーアプリケーションの設定ファイルが含まれます。これには、X サーバー自体、xfs フォントサーバー、X ディスプレイマネージャー、およびその他の多くのベースコンポーネントの設定ファイルが含まれます。

新しい Fontconfig ベースのフォントアーキテクチャーの設定ファイルは引き続き /etc/fonts/fonts.conf です。フォントの設定および追加に関する詳細は、「fonts」 を参照してください。

X サーバーは幅広いハードウェアで高度なタスクを実行するため、機能するハードウェアに関する詳細情報が必要です。X サーバーはこの情報の一部を自動的に検出します。その他の詳細を設定する必要があります。

X11R7.1 リリースパッケージがインストール用に選択されていない場合、インストールプログラムは X を自動的にインストールおよび設定します。ただし、モニター、ビデオカード、または X サーバーによって管理されるその他のデバイスへの変更がある場合は、X を再設定する必要があります。これを行う最適な方法は、特に手動で検出されないデバイスに X Configuration Tool (system-config-display)を使用することです。

Red Hat Enterprise Linux のデフォルトのグラフィカル環境では、X Configuration Tool は System （パネル）> Administration > Display で使用できます。

X Configuration Tool で行った変更は、ログアウトして再度ログインした後に有効になります。

X Configuration Tool の詳細は、36章 を参照してください。

状況によっては、X サーバーを再設定する際に設定ファイル(/etc/X11/xorg.conf)を手動で編集する必要がある場合があります。このファイルの構造に関する詳細は、「X サーバー設定ファイル」 を参照してください。

35.2. デスクトップ環境およびウィンドウマネージャー

X サーバーが実行されると、X クライアントアプリケーションがそれに接続し、そのユーザーの GUI を作成できます。Red Hat Enterprise Linux では、基本的な Tab Window Manager から、ほとんどの Red Hat Enterprise Linux ユーザーが理解している高度に開発およびインタラクティブな GNOME デスクトップ環境まで、さまざまな GUI を使用できます。

後者の包括的な GUI を作成するには、X クライアントアプリケーションの 2 つの主要クラス(デスクトップ環境 と ウィンドウマネージャー)を X サーバーに接続する必要があります。

35.2.1. デスクトップ環境

デスクトップ環境は、さまざまな X クライアントを統合し、一般的なグラフィカルユーザー環境と開発環境を作成します。

デスクトップ環境には高度な機能があり、X クライアントやその他の実行中のプロセスが相互に通信できるだけでなく、その環境で記述されたすべてのアプリケーションがドラッグアンドドロップ操作などの高度なタスクを実行できるようにします。

Red Hat Enterprise Linux は、2 つのデスクトップ環境を提供します。

- GNOME - GTK+ 2 グラフィカルツールキットをベースとする Red Hat Enterprise Linux のデフォルトのデスクトップ環境。
- kde: Qt 3 グラフィカルツールキットに基づく代替デスクトップ環境。

GNOME と KDE の両方には、単語プロセッサー、スプレッドシート、Web ブラウザーなどの高度な生産性アプリケーションがあり、どちらも GUI のルックアンドフィールをカスタマイズするツールも提供します。また、GTK+ 2 と Qt ライブラリーの両方が存在する場合は、KDE アプリケーションを GNOME で実行することも、その逆も同様です。

35.2.2. ウィンドウマネージャー

ウィンドウマネージャー は、デスクトップ環境の一部である X クライアントプログラムであり、場合によってはスタンドアロンです。主な目的は、グラフィカルウィンドウの位置付け、サイズ変更、移動方法を制御することです。ウィンドウマネージャーは、タイトルバー、ウィンドウのフォーカス動作、およびユーザー指定のキーとマウスボタンバインディングも制御します。

Red Hat Enterprise Linux には 4 つのウィンドウマネージャーが含まれています。

kwin

KWin ウィンドウマネージャーは、KDE のデフォルトのウィンドウマネージャーです。これは、カスタムテーマをサポートする効率的なウィンドウマネージャーです。

metacity

Metacity ウィンドウマネージャーは、GNOME のデフォルトのウィンドウマネージャーです。これは、カスタムテーマもサポートするシンプルで効率的なウィンドウマネージャーです。このウィンドウマネージャーを実行するには、metacity パッケージをインストールする必要があります。

mwm

Motif Window Manager (mwm)は、基本的なスタンドアロンウィンドウマネージャーです。これはスタンドアロンのウィンドウマネージャーとなるように設計されているため、GNOME または KDE と併用しないでください。このウィンドウマネージャーを実行するには、openmotif パッケージをインストールする必要があります。

twm

最小 タブウィンドウマネージャー （すべてのウィンドウマネージャーの最も基本的なツールセットを提供するtwm ）は、スタンドアロンまたはデスクトップ環境で使用できます。X11R7.1 リリースの一部としてインストールされます。

上記のウィンドウマネージャーのいずれかを実行するには、最初に Runlevel 3 で起動する必要があります。これを行う方法については、「」 を参照してください。

Runlevel 3 にログインすると、グラフィカル環境ではなくターミナルプロンプトが表示されます。ウィンドウマネージャーを起動するには、プロンプトで xinit -e & lt;path-to-window-manager> と入力します。

<path-to-window-manager& gt; は、ウィンドウマネージャーのバイナリーファイルの場所です。バイナリーファイルは、どの window-manager-name を入力します。window-manager-name は、実行するウィンドウマネージャーの名前です。

以下に例を示します。

```
~]# which twm
/usr/bin/twm
~]# xinit -e /usr/bin/twm
```

上記のコマンドは、twm ウィンドウマネージャーへの絶対パスを返し、2 番目のコマンドは twm を起動します。

ウィンドウマネージャーを終了するには、最後のウィンドウを閉じるか、Ctrl+Alt+Backspace を押します。ウィンドウマネージャーを終了すると、プロンプトで startx と入力して Runlevel 5 に再度ログインできます。

35.3. X サーバー設定ファイル

X サーバーは、単一のバイナリー実行ファイル(/usr/bin/Xorg)です。関連する設定ファイルは /etc/X11/ ディレクトリーに 保存されます（シンボリックリンクである X - /usr/bin/Xorgを参照）。X サーバーの設定ファイルは /etc/X11/xorg.conf です。

/usr/lib/xorg/modules/ ディレクトリーには、実行時に動的にロードできる X サーバーモジュールが含まれます。デフォルトでは、/usr/lib/xorg/modules/ 内の一部のモジュールのみが X サーバーによって自動的に読み込まれます。

オプションのモジュールを読み込むには、X サーバー設定ファイル /etc/X11/xorg.conf で指定する必要があります。モジュールの読み込みの詳細は、「モジュール」 を参照してください。

Red Hat Enterprise Linux 5.10 をインストールすると、インストールプロセス時にシステムハードウェアに関する情報を収集する情報を使用して、X の設定ファイルを作成します。

35.3.1. xorg.conf

/etc/X11/xorg.conf ファイルを手動で編集する必要はほとんどありません が、特にトラブルシューティングを行うときに、さまざまなセクションや任意のパラメーターを理解しておくと便利です。

35.3.1.1. 構造

/etc/X11/xorg.conf ファイルは、システムハードウェアの特定の側面に対応するさまざまなセクションで設定されています。

各セクションは、セクション &lt ;section-name > 行(< section-name > はセクションのタイトル)で始まり、EndSection 行で終わります。各セクションには、オプション名と 1 つ以上のオプション値が含まれる行が含まれます。これらは、二重引用符(")で囲むことがあります。

ハッシュマーク(#)で始まる行は X サーバーで読み取られず、人間が判読できるコメントに使用されます。

/etc/X11/xorg.conf ファイル内の一部のオプションはブール値スイッチを受け入れ、機能をオンまたはオフにします。許可されるブール値は以下のとおりです。

- *1、on、true、または yes - オプションをオンにします。*
- *0、off、false、または no - オプションをオフにします。*

以下は、一般的な /etc/X11/xorg.conf ファイルに表示される順序で、より重要なセクションの一部になります。X サーバー設定ファイルの詳細は、xorg.conf の man ページを参照してください。

35.3.1.2. Serverflags

オプションの ServerFlags セクションには、その他のグローバル X サーバー設定が含まれます。このセクションの設定は、ServerLayout セクションのオプションで上書きできます（詳細は、「ServerLayout」 を参照してください）。

ServerFlags セクション内の各エントリーは独自の行にあり、オプション という用語で始まり、二重引用符（）で囲まれたオプションが続きます。

以下は ServerFlags セクションの例です。

```
Section "ServerFlags"
 Option "DontZap" "true"
EndSection
```

以下は、最も便利なオプションのリストです。

- "DontZap" " <boolean> ": < boolean > の値が true に設定されている場合、この設定は Ctrl+Alt+Backspace キーの組み合わせを使用して X サーバーをすぐに終了しないようにします。

- "DontZoom" " <boolean> ": < boolean > の値が true に設定されている場合、この設定は Ctrl+Alt+Keypad-Plus+Ctrl+Alt+Keypad- キーの組み合わせを使用して設定されたビデオ解像度を循環しないようにします。

35.3.1.3. ServerLayout

ServerLayout セクションでは、X サーバーが制御する入出力デバイスをバインドします。このセクションは、少なくとも 1 つの出力デバイスと入力デバイスを 1 つ指定する必要があります。デフォルトでは、モニター（出力デバイス）およびキーボード（入力デバイス）を指定します。

以下の例は、典型的な ServerLayout セクションを示しています。

```
Section  "ServerLayout"
 Identifier     "Default Layout"
 Screen      0  "Screen0" 0 0
 InputDevice    "Mouse0" "CorePointer"
 InputDevice    "Keyboard0" "CoreKeyboard"
EndSection
```

ServerLayout セクションでは、以下のエントリーが一般的に使用されます。

- identifier - この ServerLayout セクションに一意の名前を指定します。

- screen: X サーバーで使用する スクリーン セクションの名前を指定します。複数の スクリーン オプションが存在する場合があります。

  一般的な Screen エントリーの例を以下に示します。

  ```
  Screen      0  "Screen0" 0 0
  ```

  この例の Screen エントリーの最初の番号(0)は、ビデオカードの最初のモニターコネク

ターまたは ヘッド が、識別子が Screen 0 の Screen セクションで指定されている設定を使用していること を示しています。

Screen 0 の識別子を持つスクリーンセクションの例は、「スクリーン」 を参照してください。

ビデオカードに複数のヘッドがある場合は、別の画面 エントリーと、異なる 画面セクション識別 子が必要です。

Screen0 の右側にある 数字は、画面左上隅の絶対 X および Y コーディネートを指定します（デフォルトでは0 ）。

- InputDevice: X サーバーで使用する InputDevice セクションの名前を指定します。

  少なくとも 2 つの InputDevice エントリーがあることが推奨されます。1 つはデフォルトのマウス用で、もう 1 つはデフォルトのキーボード用です。CorePointer オプションおよび CoreKeyboard オプションは、それらがプライマリーマウスとキーボードであることを示します。

- オプション " <option-name> " - セクションの追加パラメーターを指定するオプションのエントリー。ここで一覧表示されるオプションは ServerFlags セクションに記載されているオプションを上書きします。

  &lt ;option-name> を、xorg.conf の man ページのこのセクションに記載されている有効なオプションに置き換えます。

/etc/X11/xorg.conf ファイルに複数の ServerLayout セクションを配置することができます。ただし、デフォルトでは、サーバーは最初に見つかったもののみを読み取ります。

別の ServerLayout セクションがある場合は、X セッションの開始時にコマンドライン引数として指定できます。

35.3.1.4. ファイル

Files セクションでは、フォントパスなどの X サーバーにとって重要なサービスのパスを設定します。これは任意のセクションであり、これらのパスは通常自動的に検出されます。このセクションを使

用して、自動的に検出されたデフォルトを上書きすることができます。

以下の例は、典型的な ファイル セクションを示しています。

```
Section "Files"
 RgbPath      "/usr/share/X11/rgb.txt"
 FontPath     "unix/:7100"
EndSection
```

以下のエントリーは、ファイル セクションで一般的に使用されます。

- r gbPath: R GB の色データベースの場所を指定します。このデータベースは、X で有効な色名をすべて定義し、それらを特定の RGB 値に関連付けます。

- font Path: X サーバーが xfs フォントサーバーからフォントを取得するために接続する場所を指定します。

  デフォルトでは、FontPath は unix/:7100 です。これは、ポート 7100 のプロセス間通信 (IPC)の UNIX ドメインソケットを使用してフォント情報を取得するように X サーバーに指示します。

  X およびフォントの詳細は、 「fonts」 を参照してください。

- ModulePath: X サーバーモジュールを格納する代替ディレクトリーを指定するオプションのパラメーター。

35.3.1.5. モジュール

デフォルトでは、X サーバーは /usr/lib/xorg/modules/ ディレクトリーから以下のモジュールを自動的に読み込みます。

- extmod

- dbe

- *glx*
- *freetype*
- *type1*
- *record*
- *dri*

これらのモジュールを読み込むデフォルトのディレクトリーは、Files セクションでオプションの ModulePath パラメーターを使用して異なるディレクトリーを指定して変更できます。このセクションの詳細は、「ファイル」 を参照してください。

Module セクションを /etc/X11/xorg.conf に追加すると、X サーバーがデフォルトのモジュールではなく、このセクションに記載されているモジュールを読み込むように指示します。

たとえば、以下の一般的な モジュール セクションは以下のようになります。

```
Section "Module"
 Load  "fbdevhw"
EndSection
```

デフォルトのモジュールの代わりに fbdevhw を読み込むように X サーバーに指示します。

そのため、Module セクションを /etc/X11/xorg.conf に追加する場合は、読み込むデフォルトモジュールと追加のモジュールを指定する必要があります。

35.3.1.6. inputDevice

各 InputDevice セクションは、X サーバーに 1 つの入力デバイスを設定します。システムには通常、キーボードには少なくとも 1 つの InputDevice セクションがあります。ほとんどのマウス設定が自

動的に検出されるため、マウスの入力はまったくありません。

以下の例は、キーボードの典型的な InputDevice セクションを示しています。

```
Section "InputDevice"
        Identifier  "Keyboard0"
        Driver      "kbd"
        Option      "XkbModel" "pc105"
        Option      "XkbLayout" "us"
EndSection
```

以下のエントリーは InputDevice セクションで一般的に使用されます。

- Identifier - この InputDevice セクションの一意の名前を指定します。これは必須のエントリーです。
- driver - デバイスに対して読み込むデバイスドライバー X の名前を指定します。
- option - デバイスに関する必要なオプションを指定します。

  マウスを指定して、デバイスに対して自動検出したデフォルトを上書きすることもできます。通常、xorg.conf にマウスを追加する際に、以下のオプションが含まれます。

  - protocol - IMPS/2 など、マウスで使用されるプロトコルを指定します。
  - device - 物理デバイスの場所を指定します。
  - Emulate3Buttons - 両方のマウスボタンを同時に押したときに、2 ボタンボタンを 3 ボタンのマウスのように動作させるかどうかを指定します。

  このセクションの有効なオプションの一覧については、xorg.conf の man ページを参照してください。

35.3.1.7. 監視

各 Monitor セクションは、システムが使用するモニタータイプを 1 つ設定します。これはオプションのエントリーで、ほとんどのモニターが自動的に検出されるようになりました。

モニターを設定する最も簡単な方法は、インストールプロセス中または X Configuration Tool を使用して X を設定することです。X Configuration Tool の使用に関する詳細は、36章 を参照してください。

以下の例は、モニターの一般的な Monitor セクションを示しています。

```
Section "Monitor"
 Identifier   "Monitor0"
 VendorName   "Monitor Vendor"
 ModelName    "DDC Probed Monitor - ViewSonic G773-2"
 DisplaySize  320 240
 HorizSync    30.0 - 70.0
 VertRefresh  50.0 - 180.0
EndSection
```

WARNING

/etc/X11/xorg.conf の Monitor セクションで値を手動で編集する場合は注意してください。不適切な値は、モニターを破損または破棄する可能性があります。安全な動作パラメーターの一覧は、モニターのドキュメントを参照してください。

以下は、Monitor セクションで使用される一般的なエントリーです。

- identifier - この Monitor セクションの一意の名前を指定します。これは必須のエントリーです。
- vendorName: モニターのベンダーを指定するオプションのパラメーター。
- modelName - モニターのモデル名を指定するオプションのパラメーターです。
- DisplaySize: モニターのイメージ領域の物理サイズ（ミリ秒単位）を指定するオプション

のパラメーター。

- HorizSync: kHz のモニターと互換性のある水平同期頻度の範囲を指定します。これらの値は、X サーバーがモニターの組み込みまたは指定された モードライン エントリーの有効性を判断するのに役立ちます。

- VertRefresh - kHz でモニターによってサポートされる垂直更新頻度の範囲を指定します。これらの値は、X サーバーがモニターの組み込みモードまたは指定された モードライン エントリーの有効性を判断するのに役立ちます。

- モードライン: 特定の解像度でモニターの追加のビデオモードを指定するオプションのパラメーターで、特定の水平同期および垂直更新の解像度を使用します。モードライン エントリーの詳細は、xorg.conf の man ページを参照してください。

- オプション " <option-name> " - セクションの追加パラメーターを指定するオプションのエントリー。&lt ;option-name> を、xorg.conf の man ページのこのセクションに記載されている有効なオプションに置き換えます。

35.3.1.8. Device

各 Device セクションは、システムに 1 つのビデオカードを設定します。1 つの デバイス セクションが最小ですが、マシンにインストールされる各ビデオカードに対して追加のインスタンスが発生する可能性があります。

ビデオカードを設定する最適な方法は、インストールプロセス時または X Configuration Tool を使用して X を設定することです。X Configuration Tool の使用に関する詳細は、36章 を参照してください。

以下の例は、ビデオカードの一般 的なデバイス セクションを示しています。

```
Section "Device"
 Identifier  "Videocard0"
 Driver      "mga"
 VendorName  "Videocard vendor"
 BoardName   "Matrox Millennium G200"
 VideoRam    8192
 Option      "dpms"
EndSection
```

デバイス セクションでは、以下のエントリーが一般的に使用されます。

- *identifier - この デバイス セクションに一意の名前を指定します。これは必須のエントリーです。*

- *driver: ビデオカードを使用するために X サーバーが読み込む必要があるドライバーを指定します。ドライバーの一覧は、hwdata パッケージでインストールされる /usr/share/hwdata/videodrivers を参照してください。*

- *vendorName: ビデオカードのベンダーを指定するオプションのパラメーター。*

- *BoardName - ビデオカードの名前を指定するオプションのパラメーターです。*

- *video ram: ビデオカードで利用可能な RAM の容量をキロバイトで指定する任意のパラメーター。この設定はビデオカードにのみ必要です。X サーバーはビデオ RAM の量を検出するためにプローブできません。*

- *Busid: ビデオカードのバスの場所を指定するエントリー。1 つのビデオカードしかないシステムでは BusID エントリーは任意であり、デフォルトの /etc/X11/xorg.conf ファイルに存在しない場合もあります。ただし、複数のビデオカードが搭載されているシステムでは、BusID エントリーが存在する必要があります。*

- *screen: Device セクションが設定するビデオカード上のモニターコネクターまたはヘッドを指定するオプションのエントリー。このオプションは、複数のヘッドを持つビデオカードにのみ役立ちます。*

  複数のモニターが同じビデオカード上の異なるヘッドに接続されている場合は、個別のデバイス セクションが存在する必要があり、これらのセクションはそれぞれ異なる 画面 値を持っている必要があります。

  Screen エントリーの値は整数である必要があります。ビデオカードの最初のヘッドの値は 0 です。各ヘッドの値はこの値を 1 つ増やします。

- *オプション " <option-name> " - セクションの追加パラメーターを指定するオプションのエントリー。&lt ;option-name> を、xorg.conf の man ページのこのセクションに記載され*

ている有効なオプションに置き換えます。

より一般的なオプションの 1 つが "dpms" (Power Management Signaling の VESA 標準の表示)で、モニターの Service Star エネルギーコンプライアンス設定を有効にします。

35.3.1.9. スクリーン

各 スクリーン のセクションは、Device セクションおよび Monitor セクションを参照して、1 つのビデオカード（またはビデオカードヘッド）を 1 つのモニターにバインドします。1 つの スクリーン セクションが最小ですが、各ビデオカードとマシンに存在するモニターの組み合わせに対して追加のインスタンスが発生する可能性があります。

以下の例は、典型的な スクリーン セクションを示しています。

```
Section "Screen"
 Identifier "Screen0"
 Device     "Videocard0"
 Monitor    "Monitor0"
 DefaultDepth     16
 SubSection "Display"
  Depth     24
  Modes    "1280x1024" "1280x960" "1152x864" "1024x768" "800x600" "640x480"
 EndSubSection
 SubSection "Display"
  Depth     16
  Modes    "1152x864" "1024x768" "800x600" "640x480"
 EndSubSection
EndSection
```

Screen セクションでは、以下のエントリーが一般的に使用されます。

- Identifier - この スクリーン セクションの一意の名前を指定します。これは必須のエントリーです。
- device - デバイス セクションの一意名を指定します。これは必須のエントリーです。
- monitor - Monitor セクションの一意の名前を指定します。これは、xorg.conf ファイルで特定の Monitor セクションが定義されている場合にのみ必要になります。通常、モニターは自動的に検出されます。
-

DefaultDepth: デフォルトの色深度をビットで指定します。上記の例では、16（数千の色が提供される）がデフォルトです。承認できる DefaultDepth は 1 つだけですが、これは Xorg コマンドラインオプション -depth < n > を使用して上書きできます。< n > は追加の深さになります。

- *サブセクション Display: 特定の色深度で利用可能な画面モードを指定します。Screen セクションには複数の Display サブセクションを含めることができます。これは、画面モードが自動的に検出されるため、完全にオプションです。*

  このサブセクションは通常、自動検出モードを上書きするために使用されます。

- *オプション " <option-name> " - セクションの追加パラメーターを指定するオプションのエントリー。&lt ;option-name> を、xorg.conf の man ページのこのセクションに記載されている有効なオプションに置き換えます。*

35.3.1.10. DRI

オプションの DRI セクションは 、Direct Rendering Infrastructure (DRI)のパラメーターを指定します。DRI は、3D ソフトウェアアプリケーションが最新のビデオハードウェアに組み込まれている 3D ハードウェアアクセラレーション機能を利用できるようにするインターフェイスです。さらに、DRI は、ビデオカードドライバーでサポートされている場合、ハードウェアアクセラレーションにより 2D パフォーマンスを改善できます。

DRI Group および Mode は自動的にデフォルト値に初期化されるため、このセクションはほとんど表示されません。別のグループまたはモードが必要な場合は、xorg.conf ファイルにこのセクションを追加すると、これらのデフォルトが上書きされます。

以下の例は、典型的な DRI セクションを示しています。

```
Section "DRI"
 Group       0
 Mode        0666
EndSection
```

ビデオカードはさまざまな方法で DRI を使用しているため、http://dri.sourceforge.net/ を最初に参照せずにこのセクションに追加しないでください。

35.4. FONTS

Red Hat Enterprise Linux は、2 つのサブシステムを使用して X の下でフォントを管理および監視

します(Fontconfig および xfs)。

新しい Fontconfig フォントサブシステムはフォント管理を簡素化し、アンチエイリアスなどの高度な表示機能を提供します。このシステムは、Qt 3 または GTK+ 2 グラフィカルツールキットを使用してプログラムされたアプリケーションに自動的に使用されます。

互換性のために、Red Hat Enterprise Linux にはコア X フォントサブシステムと呼ばれる元のフォントサブシステムが含まれています。このシステムは、15 歳を超えており、X Font Server (xfs)の周りに基づいています。

本セクションでは、両方のシステムを使用して X のフォントを設定する方法を説明します。

35.4.1. fontconfig

Fontconfig font サブシステムを使用すると、アプリケーションはシステムのフォントに直接アクセスでき、Xft またはその他のレンダリングメカニズムを使用して、高度な anti-aliasing で Fontconfig フォントをレンダリングできます。グラフィカルアプリケーションは、Fontconfig で Xft ライブラリーを使用して、画面にテキストを出すことができます。

時間の経過とともに、Fontconfig/Xft font サブシステムがコア X フォントサブシステムに置き換わります。

重要な影響

Fontconfig フォントサブシステムは、独自のフォントレンダリングテクノロジーを使用する OpenOffice.org ではまだ機能しません。

Fontconfig は、手動で編集してはならない /etc/fonts/fonts.conf 設定ファイルを使用することに注意してください。

ヒント

新しいフォントシステムへの移行により、GTK+ 1.2 アプリケーションは、Font Preferences ダイアログを介して行われた変更の影響を受けません（パネルで System（パネル上）> Preferences > Fonts）を選択します。これらのアプリケーションでは、以下の行をファイル ~/.gtkrc.mine に追加してフォントを設定できます。

```
style "user-font" {
 fontset = "<font-specification>"
}

widget_class "*" style "user-font"
```

&lt ;font-specification > を、従来の X アプリケーションで使用されるスタイルのフォント仕様に置き換えます（例： -adobe-helvetica-medium-r-normal---120-*-*-*-*-*-*-* など）。コアフォントの完全な一覧は、xlsfonts を実行するか、xfontsel コマンドを使用して対話的に作成できます。*

35.4.1.1. Fontconfig へのフォントの追加

Fontconfig サブシステムに新しいフォントを追加するのは簡単なプロセスです。

1. *システム全体のフォントを追加するには、新しいフォントを /usr/share/fonts/ ディレクトリーにコピーします。ユーザーインストールとデフォルトのフォントを区別するために、local/ などの新規サブディレクトリーを作成することが推奨されます。*

   個別ユーザーのフォントを追加するには、新規フォントをユーザーのホームディレクトリーの .fonts/ ディレクトリーにコピーします。

2. *以下の例のように、fc-cache コマンドを使用してフォント情報キャッシュを更新します。*

   ```
   fc-cache <path-to-font-directory>
   ```

   このコマンドで、< path-to-font-directory > を新規フォント(/usr/share/fonts/local/ または / home/ <user> /.fonts/)を含むディレクトリーに置き換えます。

ヒント

個々のユーザーは、fonts :/// を Nautilus アドレスバーに入力し、そこに新しいフォントファイルをドラッグすることで、フォントをグラフィカルにインストールすることもできます。

重要な影響

フォントファイル名が .gz 拡張子で終わる場合、これは圧縮され、圧縮解除されるまで使用できません。これを行うには、gunzip コマンドを使用するか、ファイルをダブルクリックしてフォントを Nautilus のディレクトリーにドラッグします。

35.4.2. コア X フォントシステム

互換性のために、Red Hat Enterprise Linux は、X Font Server (xfs)を使用して X クライアントアプリケーションにフォントを提供するコア X フォントサブシステムを提供します。

X サーバーは、/etc/X11/xorg.conf 設定ファイルの Files セクション内の FontPath ディレクティブで指定されたフォントサーバーを検索します。FontPath エントリーの詳細は、「ファイル」 を参照してください。

X サーバーは、指定されたポートで xfs サーバーに接続し、フォント情報を取得します。このため、X を起動するには xfs サービスを実行している必要があります。特定のランレベルのサービス設定の詳細については、18章 を参照してください。

35.4.2.1. XFS 設定

/etc/rc.d/init.d/xfs スクリプトは、xfs サーバーを起動します。設定ファイル内で複数のオプションを設定できます(/etc/X11/fs/config)。

以下は、一般的なオプションの一覧です。

- alternate-servers: このフォントサーバーが利用できない場合に使用する別のフォントサーバーの一覧を指定します。コンマ区切りは、一覧で各フォントサーバーを区切ります。
- catalogue: 使用するフォントパスの順序付きリストを指定します。コンマ区切りは、一覧の各フォントパスを区切ります。

文字列 :unscaled を使用して、フォントパスの直後に :unscaled を使用して、そのパスのスケーリングされていないフォントを最初に読み込みます。次に、パス全体を再度指定して、他のスケーリングされたフォントもロードされるようにします。

- *client-limit: フォントサーバーサービスのクライアントの最大数を指定します。デフォルトは 10 です。*

- *clone-self: クライアント制限 に達すると、フォントサーバーが新規バージョンのクローンを作成できるようにします。デフォルトでは、このオプションは on です。*

- *default-point-size: この値を指定しないフォントのデフォルトポイントサイズを指定します。このオプションの値は decipoints で設定されます。デフォルトの 120 は 12 ポイントフォントに対応します。*

-

-

-

-

ヒント

-

-

35.4.2.2.

1.

```
mkdir /usr/share/fonts/local/
```

```
chkfontpath --add /usr/share/fonts/local/
```

2.

3.

```
ttmkfdir -d /usr/share/fonts/local/ -o /usr/share/fonts/local/fonts.scale
```

4.

```
service xfs reload
```

35.5.

35.5.1.

35.5.2.

-
-
-

/usr/share/doc/initscripts-<version-number>/sysconfig.txt

35.6. 関連情報

35.6.1. インストールされているドキュメント

-
-
-

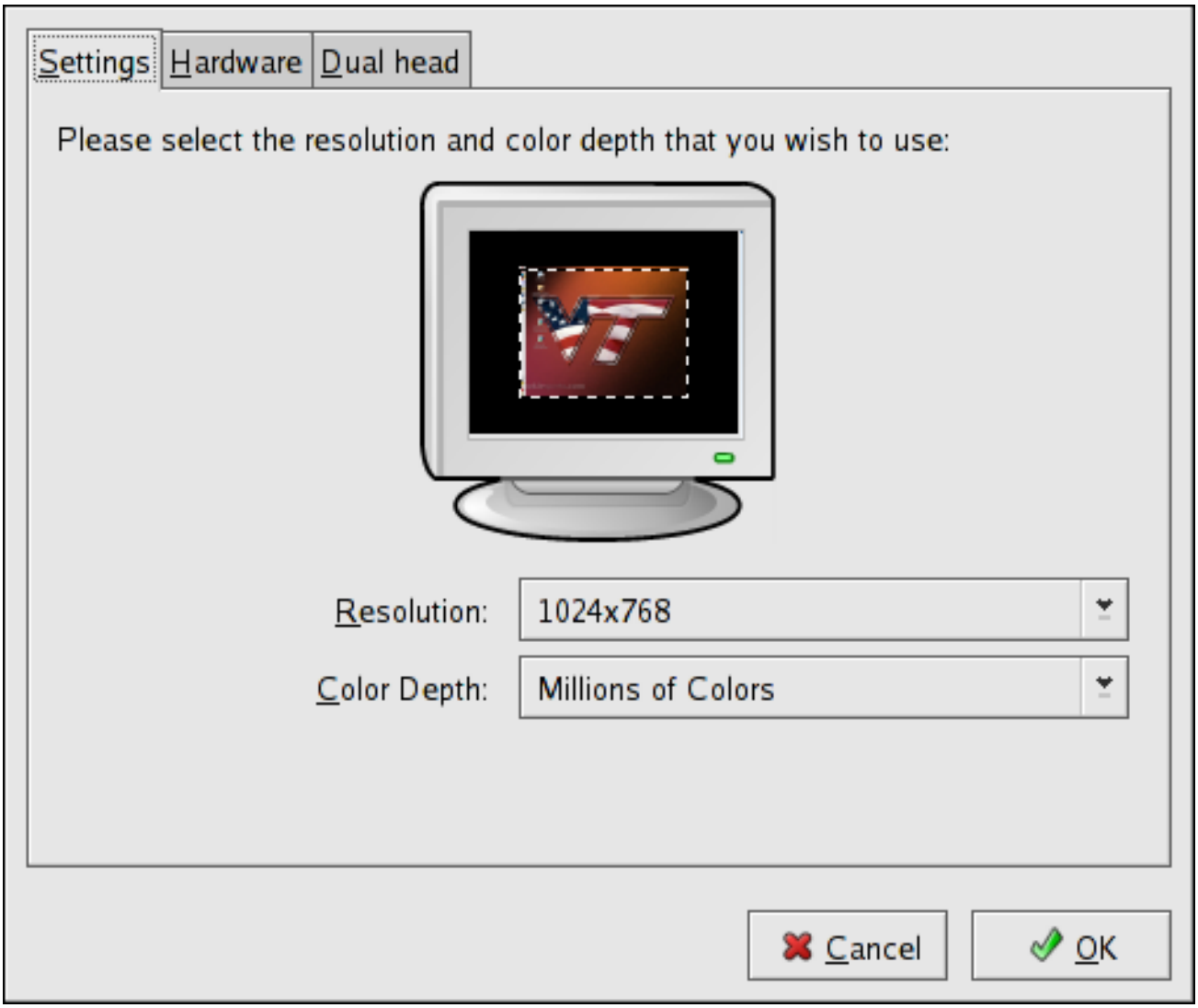
35.6.2. 便利な Web サイト

-
-
-
-

第36章

36.1.

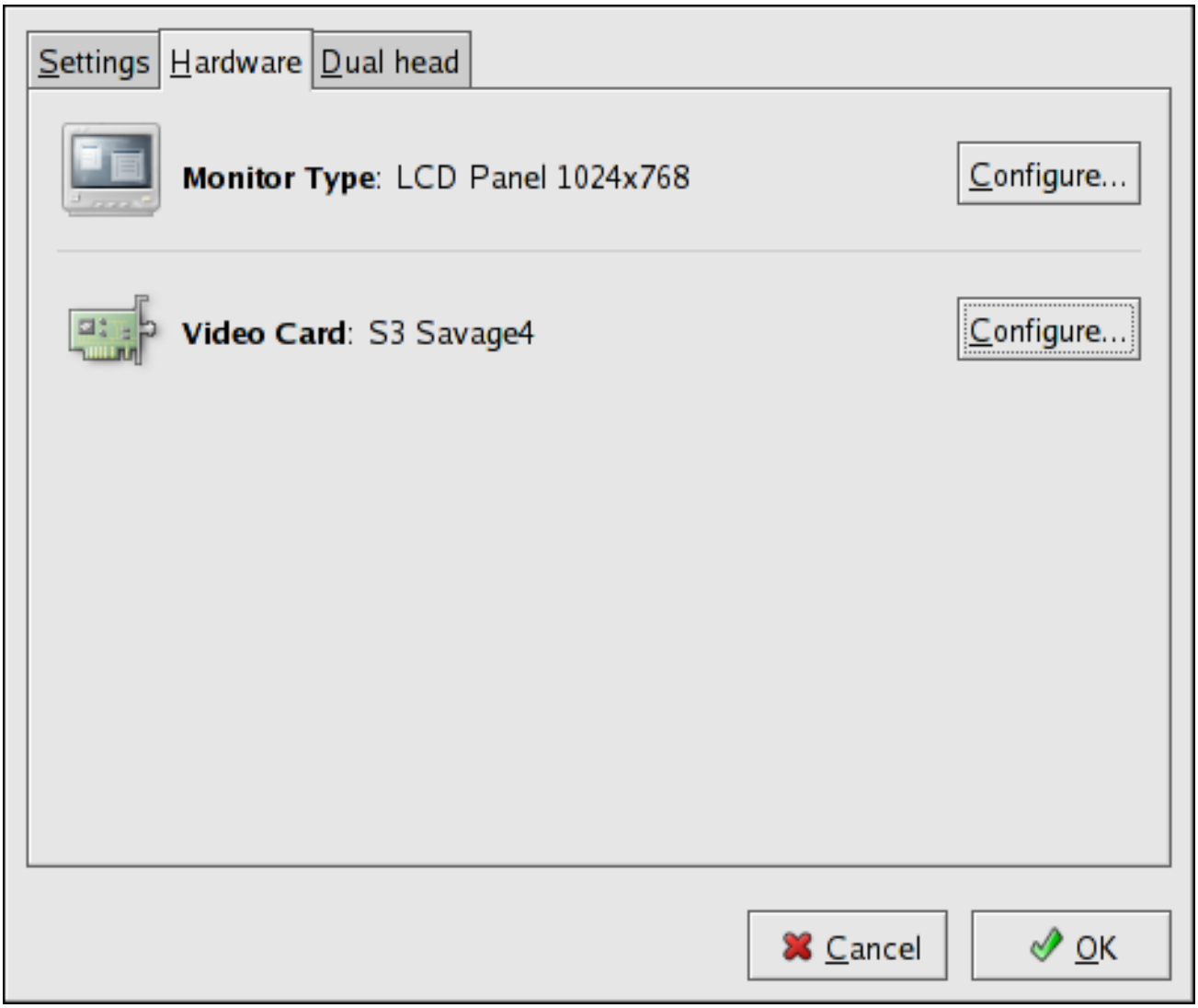
図36.1


Settings
Hardware
Dual head
Please select the resolution and color depth that you wish to use:
Resolution:
1024x768
Color Depth:
Millions of Colors
Cancel
OK

[D]

36.2.

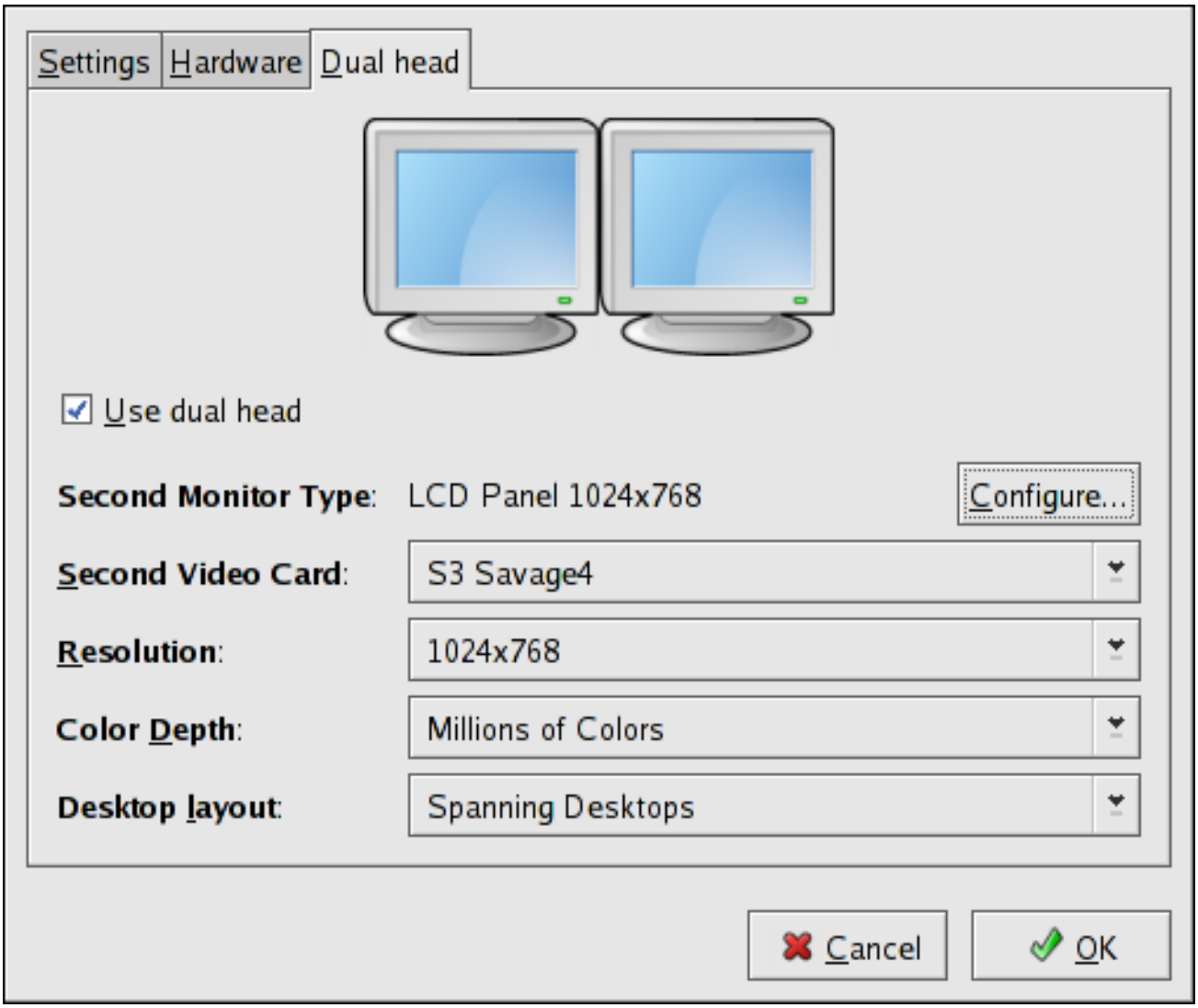
図36.2


Settings
Hardware
Dual head
Monitor Type: LCD Panel 1024x768
Configure...
Video Card: S3 Savage4
Configure...
Cancel
OK

[D]

36.3.

図36.3


Settings
Hardware
Dual head
Use dual head
Second Monitor Type:
LCD Panel 1024x768
Configure...
Second Video Card:
S3 Savage4
Resolution:
1024x768
Color Depth:
Millions of Colors
Desktop layout:
Spanning Desktops
Cancel
OK

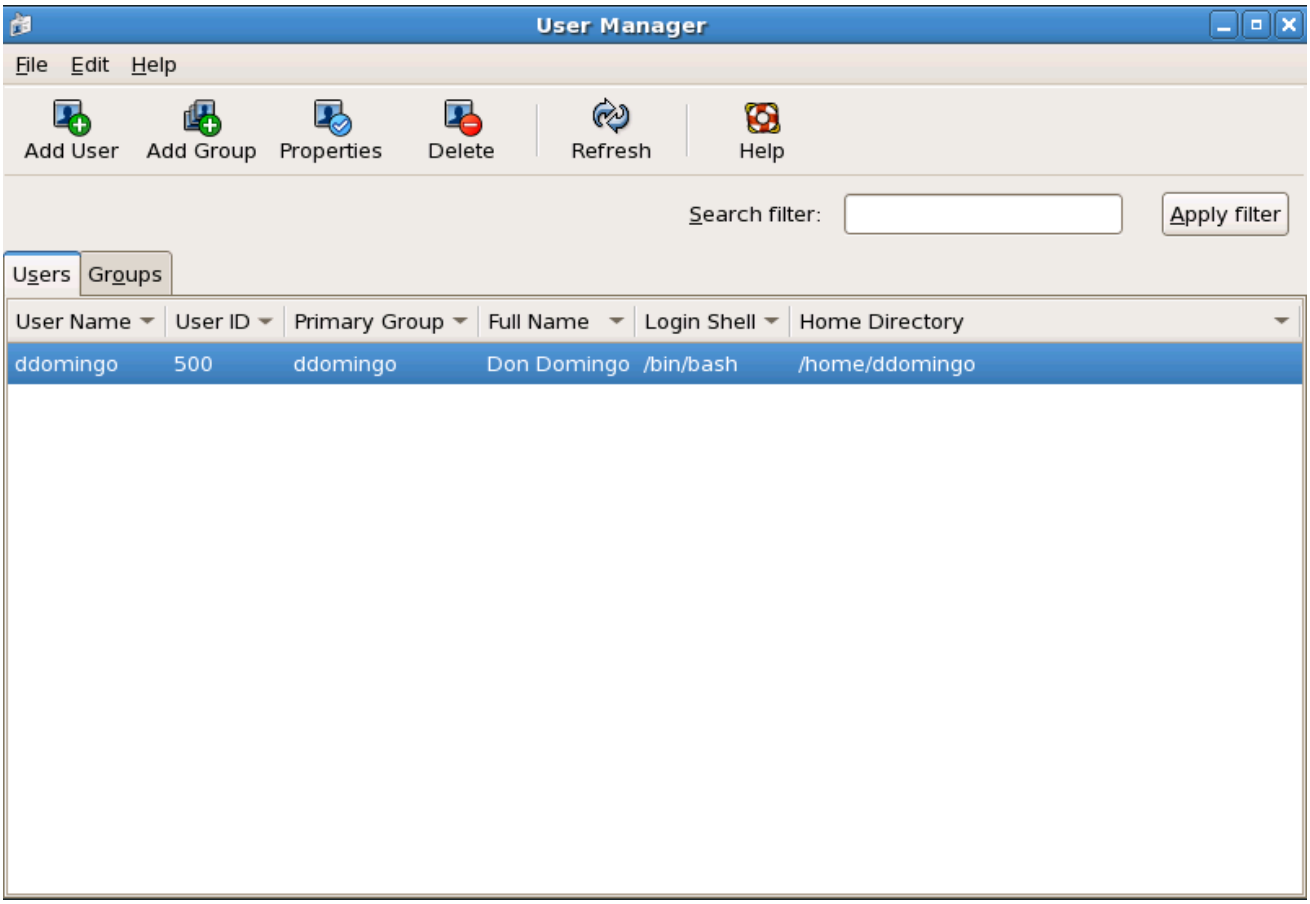
[D]

第37章 ユーザーとグループ

ファイルを作成するユーザーは、そのファイルの所有者であり、グループ所有者でもあります。ファイルには、所有者、グループ、その他に対して読み取り、書き込み、実行のパーミッションが別々に割り当てられます。

37.1.

図37.1 User Manager

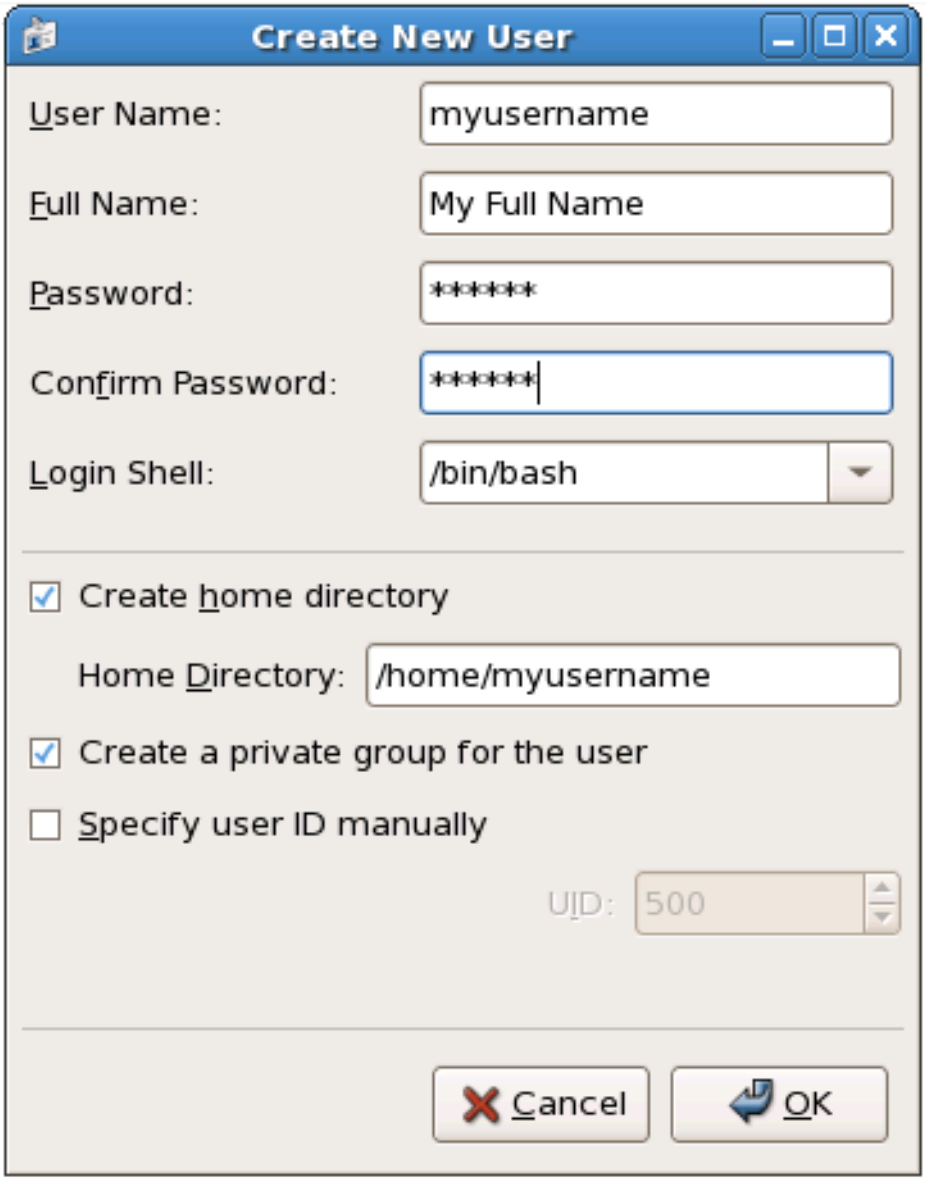


[D]

37.1.1. 新規ユーザーの追加

ヒント

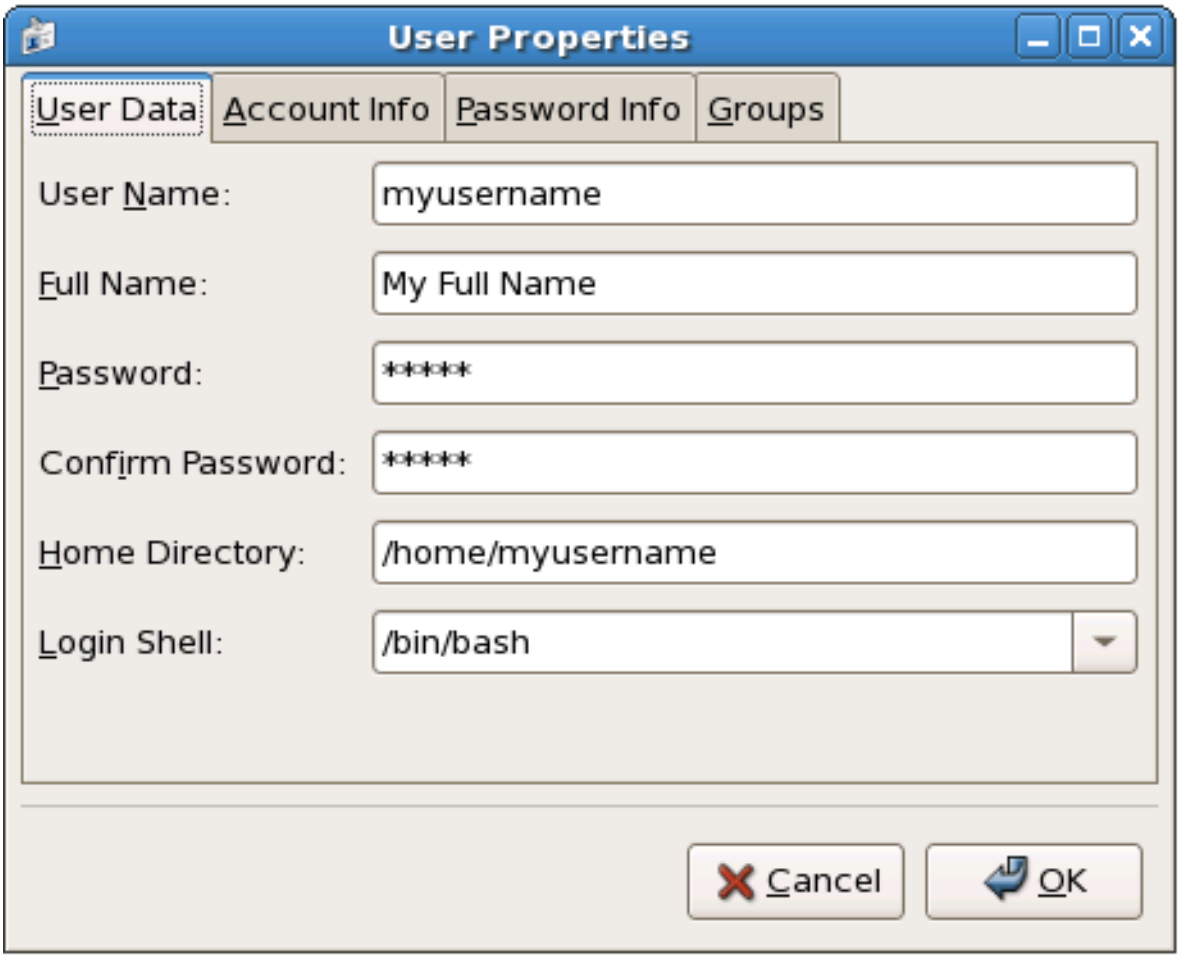
図37.2


Create New User
User Name:
myusername
Full Name:
My Full Name
Password:
Confirm Password:
Login Shell:
/bin/bash
Create home directory
Home Directory:
/home/myusername
Create a private group for the user
Specify user ID manually
UID:
500
Cancel
OK

[D]

37.1.2.

図37.3 ユーザープロパティー

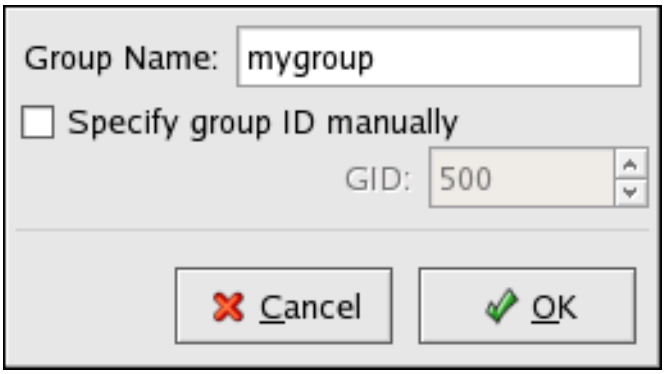


[D]

-
-
-
-

37.1.3. 新規グループの追加

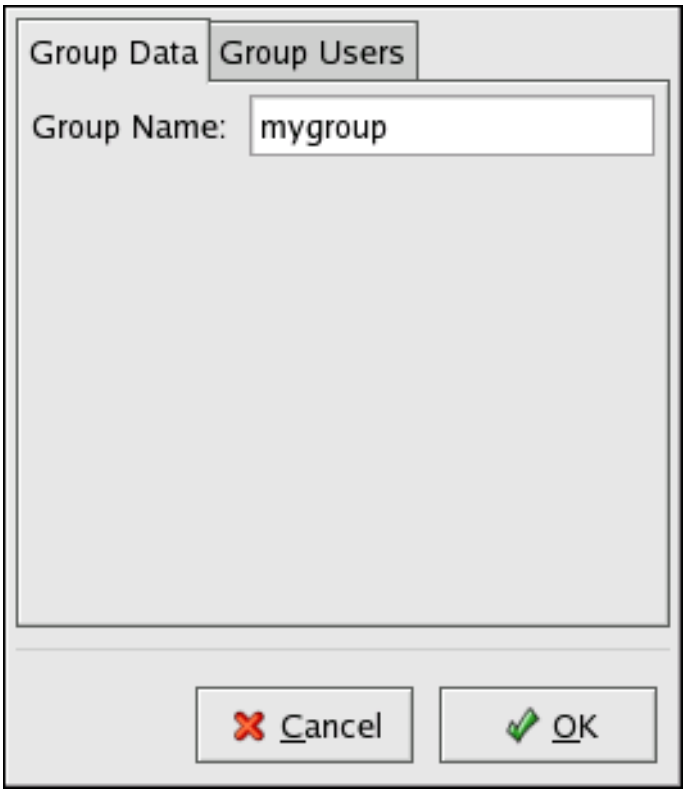
図37.4 新規グループ



[D]

37.1.4.

図37.5



[D]

37.2.

-
-
-
-
-

37.2.1. コマンドラインからの設定

37.2.2. ユーザーの追加

1.

```
useradd <username>
```

2.

```
passwd <username>
```

表37.1

オプション	説明
-c '<comment>'	このオプションは、通常、ユーザーの氏名を指定するのに使用されます。
-d <home-dir>	
-e <date>	
-f <days>	パスワードが失効してからアカウントが無効になるまでの日数です。
-g <group-name>	グループは、ここで指定するよりも前に作成されている必要があります。
-G <group-list>	グループは、ここで指定する前に作成しておく必要があります。
-m	ホームディレクトリーがない場合は、これを作成します。
-M	ホームディレクトリーを作成しません。
-n	ユーザー用のユーザープライベートグループを作成しません。
-r	
-p <password>	
-s	
-u <uid>	

37.2.3.

```
groupadd <group-name>
```

-

表37.2

オプション	説明
-g <gid>	
-r	
-f	

37.2.4.

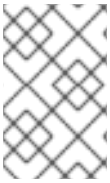
重要な影響

詳細は、「シャドウパスワード」 を参照してください。

表37.3

オプション	説明
-m <days>	
-M <days>	このオプションで指定した日数と、-d オプションで指定した日数を足した日数が、現在の日数より少ない場合、ユーザーはアカウントを使用する前にパスワードを変更する必要があります。
-d <days>	
-I <days>	パスワードの有効期限後、アカウントをロックするまでの非アクティブ日数を指定します。

オプション	説明
-E <date>	アカウントがロックされる日付を YYYY-MM-DD のフォーマットで指定します。日付の代わりに、1970 年 1 月 1 日からの日数を使うこともできます。
-W <days>	パスワードの有効期限の何日前にユーザーに警告を発するかを指定します。
-l	現在のアカウントエージングの設定を一覧表示します。



ヒント

ユーザーが初めてログインしたときにパスワードが失効するように設定することができます。これにより、ユーザーはすぐにパスワードを変更せざるを得なくなります。

1.

-

```
Python 2.4.3 (#1, Jul 21 2006, 08:46:09)
[GCC 4.1.1 20060718 (Red Hat 4.1.1-9)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

-

```
import crypt
print crypt.crypt("<password>","<salt>")
```

-

-

```
usermod -p "<encrypted-password>" <username>
```

これを行うには、以下のコマンドを使用します。

```
usermod -p "" username
```

2.

```
chage -d 0 username
```

このコマンドは、パスワードが最後に変更された日付の値をエポック (1970 年 1 月 1 日) に設定します。この値は、パスワードエージングポリシーがある場合、それに関係なく、パスワードの即時期限切れを強制します。

最初のログイン時に、ユーザーは新しいパスワードの入力を求められるようになりました。

37.2.5. プロセスの説明

1.

この行には以下の特徴があります。

-

-
-
-
-
-
-

2. *この行には以下の特徴があります。*

-
-

注記

- *パスワードは有効期限なしで設定されています。*

3.

-
-
-

4. *juan という名前のグループ用の新しい行が /etc/gshadow に作成されます。この行には以下の特徴があります。*

   - *グループ名 juan で始まります。*
   - *感嘆符(!)が、グループをロックする /etc/gshadow ファイルのパスワードフィールドに表示されます。*
   - *その他のフィールドはすべて空白です。*

5. *ユーザー juan のディレクトリーが /home/ ディレクトリー内に作成されます。このディレクトリーは、ユーザー juan およびグループ juan が所有しています。ただし、ユーザー juan に対してのみ 読み取り、書き込み、実行権限があります。その他のパーミッションは拒否されます。*

6. *（デフォルトユーザー設定を含む）/etc/skel/ ディレクトリー内のファイルは、新しい /home/juan/ ディレクトリーにコピーされます。*

この時点で、juan という名前のロックされたアカウントがシステム上に存在します。このアカウントをアクティブにするには、管理者が passwd コマンドを使用して、アカウントにパスワードを割り当てる必要があります。また、必要に応じて、パスワードのエージングガイドラインを設定します。

37.3. 標準ユーザー

表37.4「標準ユーザー」 /etc/passwd ファイルで設定されている標準ユーザーを、すべて のインストールで一覧表示します。この表の groupid (GID)は、ユーザーの プライマリーグループ です。標準グループの一覧は、「標準グループ」 を参照してください。

表37.4 標準ユーザー

User	UID	GID	ホームディレクトリー	シェル
root	0	0	/root	/bin/bash
bin	1	1	/bin	/sbin/nologin
daemon	2	2	/sbin	/sbin/nologin
adm	3	4	/var/adm	/sbin/nologin
lp	4	7	/var/spool/lpd	/sbin/nologin
sync	5	0	/sbin	/bin/sync
shutdown	6	0	/sbin	/sbin/shutdown
halt	7	0	/sbin	/sbin/halt
mail	8	12	/var/spool/mail	/sbin/nologin
news	9	13	/etc/news	
ucp	10	14	/var/spool/uucp	/sbin/nologin
operator	11	0	/root	/sbin/nologin
games	12	100	/usr/games	/sbin/nologin
Gopher	13	30	/var/gopher	/sbin/nologin
ftp	14	50	/var/ftp	/sbin/nologin
nobody	99	99	/	/sbin/nologin
rpm	37	37	/var/lib/rpm	/sbin/nologin
vcsa	69	69	/dev	/sbin/nologin
dbus	81	81	/	/sbin/nologin
ntp	38	38	/etc/ntp	/sbin/nologin

User	UID	GID	ホームディレクトリー	シェル
カナナ	39	39	/var/lib/canna	/sbin/nologin
nscd	28	28	/	/sbin/nologin
rpc	32	32	/	/sbin/nologin
postfix	89	89	/var/spool/postfix	/sbin/nologin
mailman	41	41	/var/mailman	/sbin/nologin
named	25	25	/var/named	/bin/false
amanda	33	6	var/lib/amanda/	/bin/bash
postgres	26	26	/var/lib/pgsql	/bin/bash
exim	93	93	/var/spool/exim	/sbin/nologin
sshd	74	74	/var/empty/sshd	/sbin/nologin
rpcuser	29	29	/var/lib/nfs	/sbin/nologin
nsfnobody	65534	65534	/var/lib/nfs	/sbin/nologin
pvm	24	24	/usr/share/pvm3	/bin/bash
apache	48	48	/var/www	/sbin/nologin
xfs	43	43	/etc/X11/fs	/sbin/nologin
gdm	42	42	/var/gdm	/sbin/nologin
HTT	100	101	/usr/lib/im	/sbin/nologin
mysql	27	27	/var/lib/mysql	/bin/bash
webalizer	67	67	/var/www/usage	/sbin/nologin
mailnull	47	47	/var/spool/mqueue	/sbin/nologin

User	UID	GID	ホームディレクトリー	シェル
smmsp	51	51	/var/spool/mqueue	/sbin/nologin
squid	23	23	/var/spool/squid	/sbin/nologin
ldap	55	55	/var/lib/ldap	/bin/false
netdump	34	34	/var/crash/	/bin/bash
pcap	77	77	/var/arpwatch	/sbin/nologin
radiusd	95	95	/	/bin/false
radvd	75	75	/	/sbin/nologin
quagga	92	92	/var/run/quagga	/sbin/login
wnn	49	49	/var/lib/wnn	/sbin/nologin
dovecot	97	97	/usr/libexec/dovecot	/sbin/nologin

37.4. 標準グループ

表37.5「標準グループ」 Everything インストールによって設定された標準グループを一覧表示します。グループは /etc/group ファイルに保存されます。

表37.5 標準グループ

Group	GID	Members
root	0	root
bin	1	root、bin、daemon
daemon	2	root、bin、daemon
sys	3	root、bin、adm
adm	4	root, adm, daemon

Group	GID	Members
tty	5	
disk	6	root
lp	7	daemon、lp
mem	8	
kmem	9	
ホイール	10	root
mail	12	mail、postfix、exim
news	13	news
ucp	14	ucp
man	15	
games	20	
Gopher	30	
dip	40	
ftp	50	
lock	54	
nobody	99	
users	100	
rpm	37	
utmp	22	
floppy	19	

Group	GID	Members
vcsa	69	
dbus	81	
ntp	38	
カナナ	39	
nscd	28	
rpc	32	
postdrop	90	
postfix	89	
mailman	41	
exim	93	
named	25	
postgres	26	
sshd	74	
rpcuser	29	
nfsnobody	65534	
pvm	24	
apache	48	
xfs	43	
gdm	42	
HTT	101	

Group	GID	Members
mysql	27	
webalizer	67	
mailnull	47	
smmsp	51	
squid	23	
ldap	55	
netdump	34	
pcap	77	
quaggavt	102	
quagga	92	
radvd	75	
slocate	21	
wnn	49	
dovecot	97	
radiusd	95	

37.5. ユーザープライベートグループ

Red Hat Enterprise Linux は、ユーザープライベートグループ (UPG)スキームを使用するため、UNIX グループの管理が容易になります。

新規ユーザーがシステムに追加されるたびに、UPG が作成されます。UPG は作成したユーザーと同じ名前を持ち、そのユーザーが UPG の唯一のメンバーです。

UPG を使用すると、新しく作成されたファイルまたはディレクトリーにデフォルトのパーミッションを安全に設定でき、そのユーザーのユーザーと、そのユーザーのグループの 両方がファイルまたはディレクトリーを変更できるようになります。

新しく作成されたファイルまたはディレクトリーに適用される権限を決める設定は umask と呼ばれ、/etc/bashrc ファイルで設定します。従来の UNIX システムでは、umask は 022 に設定されています。これにより、ファイルまたはディレクトリーを作成したユーザーのみが変更できるようになりました。このスキームでは、作成者のグループ のメンバーなど、他のすべての ユーザーは変更を加えることはできません。ただし、UPG スキームでは、すべてのユーザーがそれぞれプライベートグループを持つため、このグループ保護は必須ではなくなりました。

37.5.1. グループディレクトリー

多くの IT 組織は、主要なプロジェクトごとにグループを作成し、そのプロジェクトのファイルにアクセスする必要がある場合に、そのグループにユーザーを割り当てます。従来のスキームでは、ファイルの管理は困難でした。誰かがファイルを作成すると、そのファイルが属するプライマリーグループに関連付けられます。1 人のユーザーが複数のプロジェクトで作業する場合、適切なファイルを適切なグループに関連付けるのは困難です。ただし、UPG スキームを使用すると、グループは、setgid ビットが設定されたディレクトリー内に作成されたファイルに自動的に割り当てられます。setgid ビットを使用すると、共通ディレクトリーを共有するグループプロジェクトの管理が非常に簡単です。これは、ユーザーがディレクトリー内で作成するファイルは、ディレクトリーを所有するグループが所有するためです。

たとえば、あるグループが /usr/share/emacs/site-lisp/ ディレクトリーのファイルを作業する必要があるとします。ディレクトリーの変更に信頼されているユーザーもいますが、誰でも信頼されているとは限りません。以下のコマンドのように、emacs グループを作成します。

```
groupadd emacs
```

ディレクトリーの内容を emacs グループに関連付けるには、以下を入力します。

```
chown -R root.emacs /usr/share/emacs/site-lisp
```

gpasswd コマンドを使用して、適切なユーザーをグループに追加できるようになりました。

```
gpasswd -a <username> emacs
```

ユーザーが ディレクトリーにファイルを作成できるようにするには、以下のコマンドを使用します。

```
chmod 775 /usr/share/emacs/site-lisp
```

ユーザーが新しいファイルを作成すると、ユーザーのデフォルトのプライベートグループのグループが割り当てられます。次に、setgid ビットを設定します。これは、ディレクトリー自体と同じグループパーミッションをディレクトリー内で割り当てます(emacs)。以下のコマンドを使用します。

```
chmod 2775 /usr/share/emacs/site-lisp
```

この時点で、各ユーザーのデフォルトの umask は 002 であるため、ユーザーが新しいファイルを書き込むたびに管理者がファイルのパーミッションを変更しなくても、emacs グループのすべてのメンバーが /usr/share/emacs/site-lisp/ ディレクトリーにファイルを作成および編集できます。

37.6. シャドウパスワード

マルチユーザー環境では、(shadow-utils パッケージで提供) シャドウパスワード を使用することが重要です。これにより、システム認証ファイルのセキュリティーが強化されます。このため、インストールプログラムでは、デフォルト設定でシャドウパスワードを有効にしています。

以下は、UNIX ベースのシステムにパスワードを保存する従来の方法よりも pf シャドウパスワードの利点を示しています。

- *暗号化されたパスワードハッシュを、誰でも読み取り可能な /etc/passwd ファイルから、root ユーザーのみが読み取り可能な /etc/shadow に移動することで、システムのセキュリティーを向上させます。*
- *パスワードのエージングに関する情報を保存します。*
- */etc/login.defs ファイルを使用してセキュリティーポリシーを適用できます。*

shadow-utils パッケージが提供するほとんどのユーティリティーは、シャドウパスワードが有効になっているかどうかに関係なく適切に機能します。ただし、パスワードエージングの情報は /etc/shadow ファイルに排他的に保存されるため、パスワードのエージング情報を作成または変更するコマンドは動作しません。

以下は、シャドウパスワードを有効にしないと機能しないコマンドの一覧です。

- *chage*

- *gpasswd*
- */usr/sbin/usermod -e オプションまたは -f オプション*
- */usr/sbin/useradd -e オプションまたは -f オプション*

37.7. 関連情報

ユーザーおよびグループ、およびそれらを管理するツールの詳細は、以下のリソースを参照してください。

37.7.1. インストールされているドキュメント

- *関連する man ページ : ユーザーおよびグループの管理に関連するさまざまなアプリケーションおよび設定ファイルに関する man ページが多数あります。より重要な man ページの一部を以下に示します。*

  ユーザーおよびグループの管理アプリケーション

  - *man chage - パスワードのエージングポリシーとアカウントの有効期限を変更するコマンドです。*
  - *man gpasswd: /etc/group ファイルを管理するコマンドです。*
  - *man groupadd - グループを追加するコマンドです。*
  - *man grpck: /etc/group ファイルを検証するコマンドです。*
  - *man groupdel - グループを削除するコマンドです。*
  - *man groupmod - グループメンバーシップを修正するコマンドです。*
  - *man pwck - /etc/passwd ファイルおよび /etc/shadow ファイルを検証するコマンドです。*

- *man pwconv - 標準パスワードをシャドウパスワードに変換するツールです。*
- *man pwunconv - シャドウパスワードを標準パスワードに変換するツールです。*
- *man useradd: ユーザーを追加するコマンドです。*
- *man userdel: ユーザーを削除するコマンドです。*
- *man usermod - ユーザーを修正するコマンドです。*

設定ファイル

- *man 5 group - システムのグループ情報を含むファイルです。*
- *man 5 passwd - システムのユーザー情報を含むファイルです。*
- *man 5 shadow - システムのパスワードおよびアカウントの有効期限情報を含むファイルです。*

第38章 プリンターの設定

プリンター 設定ツール を使用すると、ユーザーはプリンターを設定できます。このツールは、プリンター設定ファイル、印刷スプールディレクトリー、印刷フィルター、プリンタークラスの維持に役立ちます。

Red Hat Enterprise Linux 5.10 は Common Unix Printing System (CUPS)を使用します。CUPS を使用した以前の Red Hat Enterprise Linux バージョンからシステムがアップグレードされた場合、アップグレードプロセスは設定されたキューを保持します。

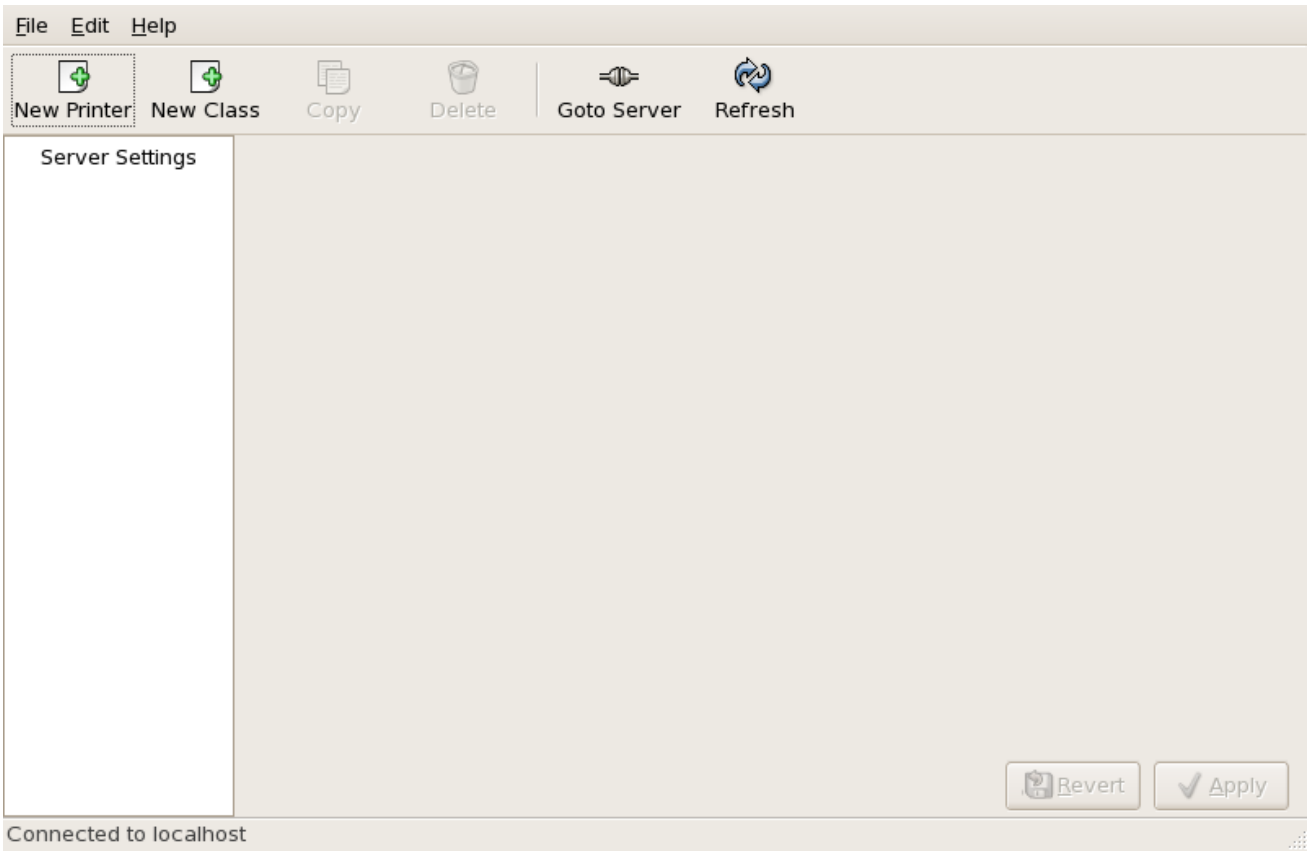
重要

cupsd.conf の man ページには、CUPS サーバーの設定が記載されています。これには、SSL サポートを有効にするためのディレクティブが含まれます。ただし、CUPS では使用されるプロトコルバージョンのコントロールが許可されません。『Resolution for POODLE SSLv3.0 vulnerability (CVE-2014-3566) for components that do not allow SSLv3 to be disabled via configuration settings （設定設定で SSLv3 を無効にできないコンポーネントの POODLE SSLv3』.0 脆弱性(CVE-2014-3566)の解決）で説明されている脆弱性により、Red Hat はセキュリティーのためにこれに依存しないことを推奨します。stunnel を使用してセキュアなトンネルを提供し、SSLv3 を無効にすることが推奨されます。

リモートシステムの 印刷設定 ツールへのアドホックのセキュアな接続は、「X11 転送」 で説明されているように、SSH で X11 転送を使用します。

プリンター設定ツール を使用するには、root 権限が必要です。アプリケーションを起動するには、System （パネルで）> Administration > Printing の順に選択し、シェルプロンプトでコマンド system-config-printer を入力します。

図38.1 プリンター設定ツール



[D]

以下のタイプのプリントキューを設定できます。

- *AppSocket/HP JetDirect: コンピューターではなく HP JetDirect インターフェイスまたは Appsocket インターフェイスを介してネットワークに直接接続されたプリンター。*
- *Internet Printing Protocol (IPP) - Internet Printing Protocol （たとえば、ネットワーク上の CUPS を実行している別の Red Hat Enterprise Linux システムに接続されているプリンター）経由で TCP/IP ネットワーク経由でアクセスできるプリンターです。*
- *LPD/LPR Host または Printer: TCP/IP ネットワーク経由でアクセスできる別の UNIX システムに接続されているプリンター（たとえば、ネットワーク上で LPD を実行している別の Red Hat Enterprise Linux システムに接続されているプリンター）。*
- *ネットワーク接続された Windows (SMB): SMB ネットワーク上でプリンターを共有する別のシステムに接続されているプリンター(Microsoft Windows™ マシンに接続されているプリンターなど)。*
-

Networked JetDirect - コンピューターではなく HP JetDirect を介して直接接続したプリンター。

重要な影響

新しい印刷キューを追加するか、既存のキューを変更する場合は、変更を適用する必要があります。

Apply ボタンをクリックすると、プリンターデーモンが、設定した変更で再起動するように求められます。

Revert ボタンをクリックすると、適用されていない変更が破棄されます。

38.1. ローカルプリンターの追加

ローカルプリンター（コンピューターのパラレルポートまたは USB ポートを介して接続）を追加するには、メインの Printer Configuration Tool ウィンドウの New Printer ボタンをクリックして 図 38.2「プリンターの追加」 にウィンドウを表示します。

図38.2 プリンターの追加

[D]

Forward をクリックして続行します。

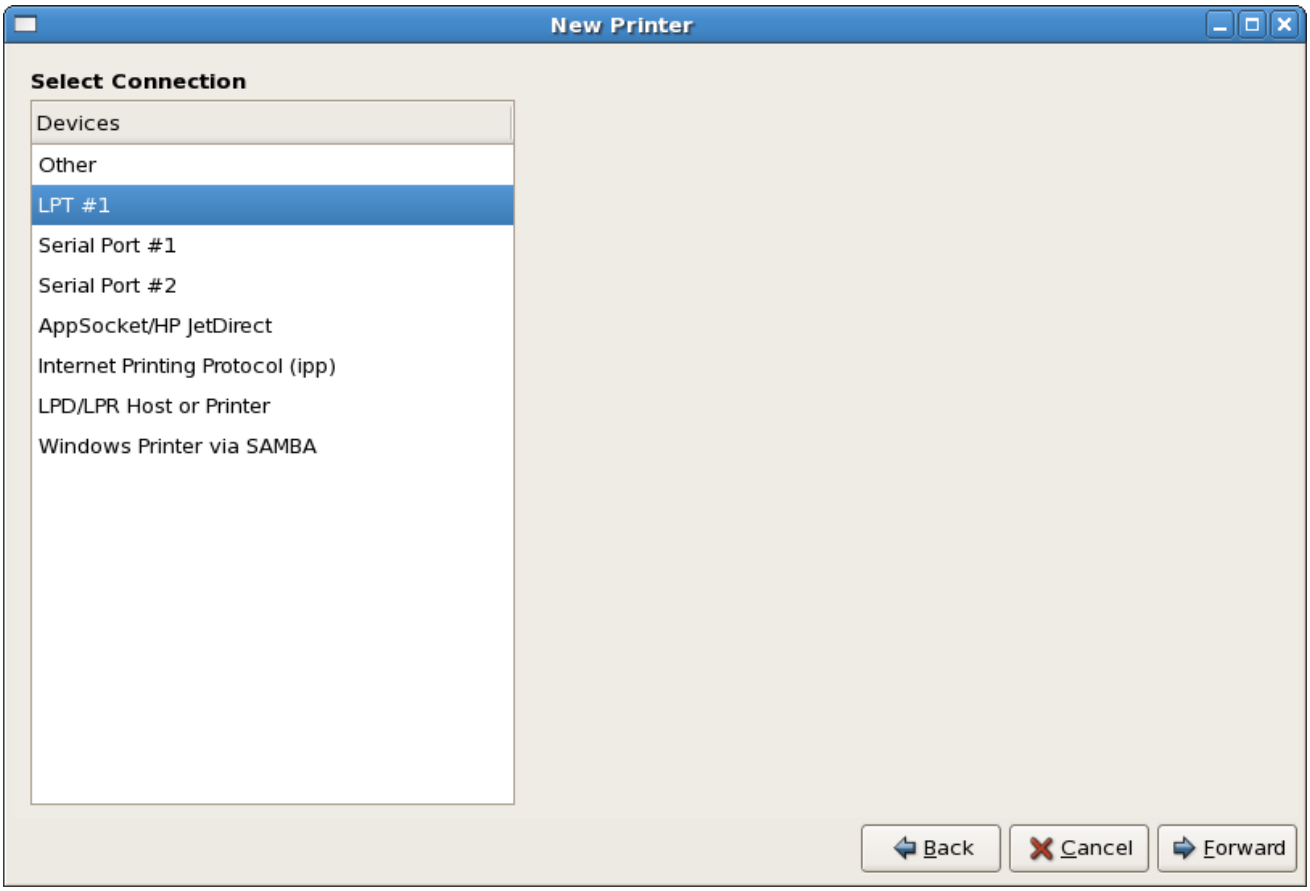
Printer Name フィールドにプリンターの一意の名前を入力します。プリンター名には、文字、数字、ダッシュ(-)、およびアンダースコア(_)を含めることができます。スペースを含めることはできません。

また、説明 および ロケーション フィールドを使用して、このプリンターをシステムに設定できる他の項目とさらに区別することもできます。これらのフィールドはいずれもオプションで、スペースを含めることができます。

Forward をクリックして、New Printer ダイアログを開きます(図38.3「ローカルプリンターの追加」 を参照してください)。プリンターが自動的に検出されると、プリンターモデルは 接続の選択 に表示されます。プリンターモデルを選択し、Forward をクリックして続行します。

デバイスが自動的に表示されない場合は、接続の選択 でプリンターが接続するデバイス（例： LPT #1 または Serial Port #1）を 選択 します。

図38.3 ローカルプリンターの追加



[D]

次に、プリンタータイプを選択します。詳細は、「プリンターモデルの選択と完了」 を参照してください。

38.2. IPP プリンターの追加

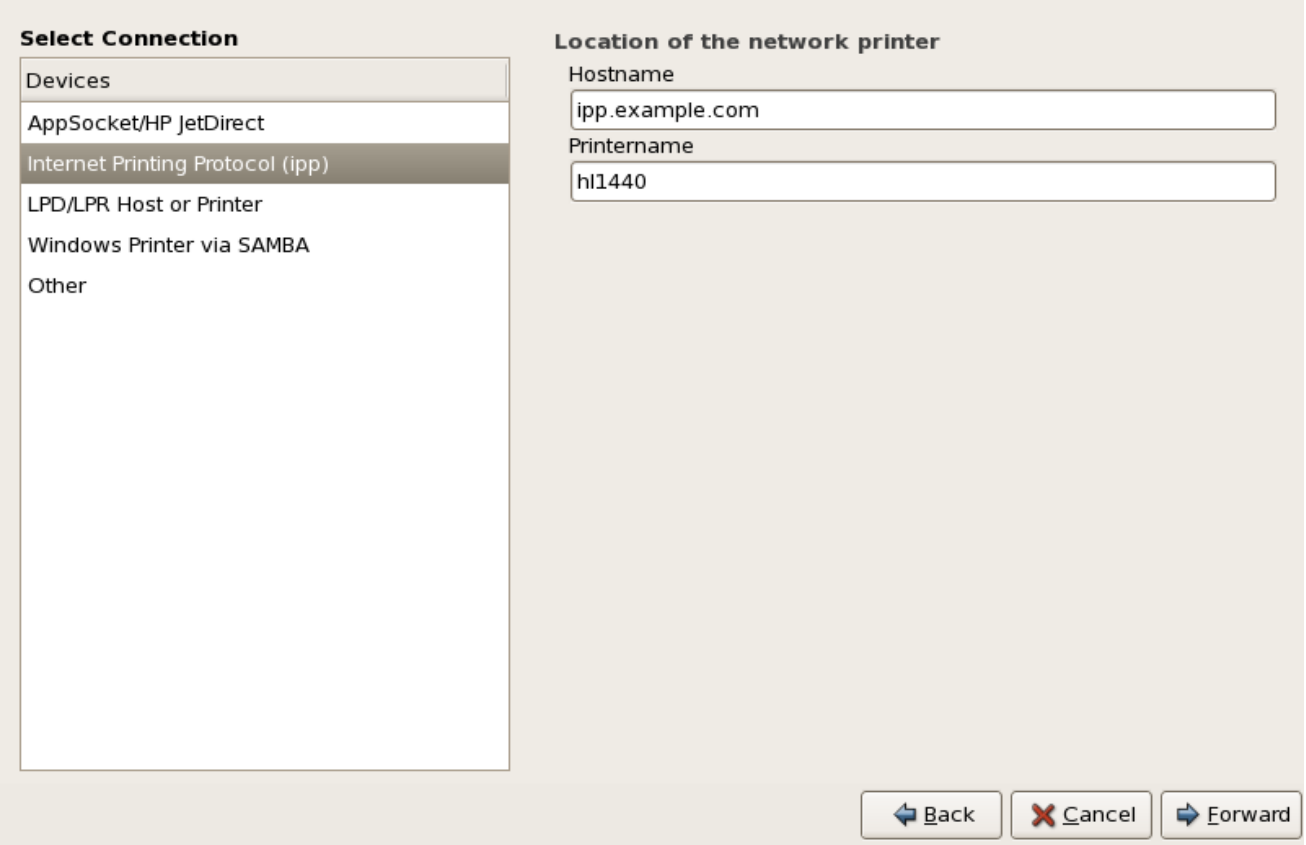
IPP プリンターは、同じ TCP/IP ネットワーク上の別のシステムに接続されているプリンターです。このプリンターが接続されているシステムは CUPS を実行しているか、単に IPP を使用するように設定されているだけです。

プリンターサーバーでファイアウォールが有効になっている場合は、受信 UDP ポート 631 での送受信接続を許可するようにファイアウォールを設定する必要があります。クライアント（印刷要求を送信するシステム）でファイアウォールが有効になっている場合は、ポート 631 を介した接続を受け入れて作成できるようにファイアウォールを設定する必要があります。

メインの Printer Configuration Tool ウィンドウの New Printer ボタンをクリックして、図38.2「プリンターの追加」 にウィンドウを表示することで、ネットワーク IPP プリンターを追加できます。プリンター名(printer 名にはスペースが含まれず、文字、数字、ダッシュ(-)、およびアンダースコア(_))、説明、および ロケーション を入力して、システムに設定できる他のプリンターと区別します。Forward をクリックして続行します。

図38.4「IPP プリンターの追加」 に表示されるウィンドウで、ホスト名 フィールドに IPP プリンターのホスト名と、プリンター名 フィールドにプリンターの一意の 名前 を入力します。

図38.4 IPP プリンターの追加



[D]

進む をクリックして続けます。

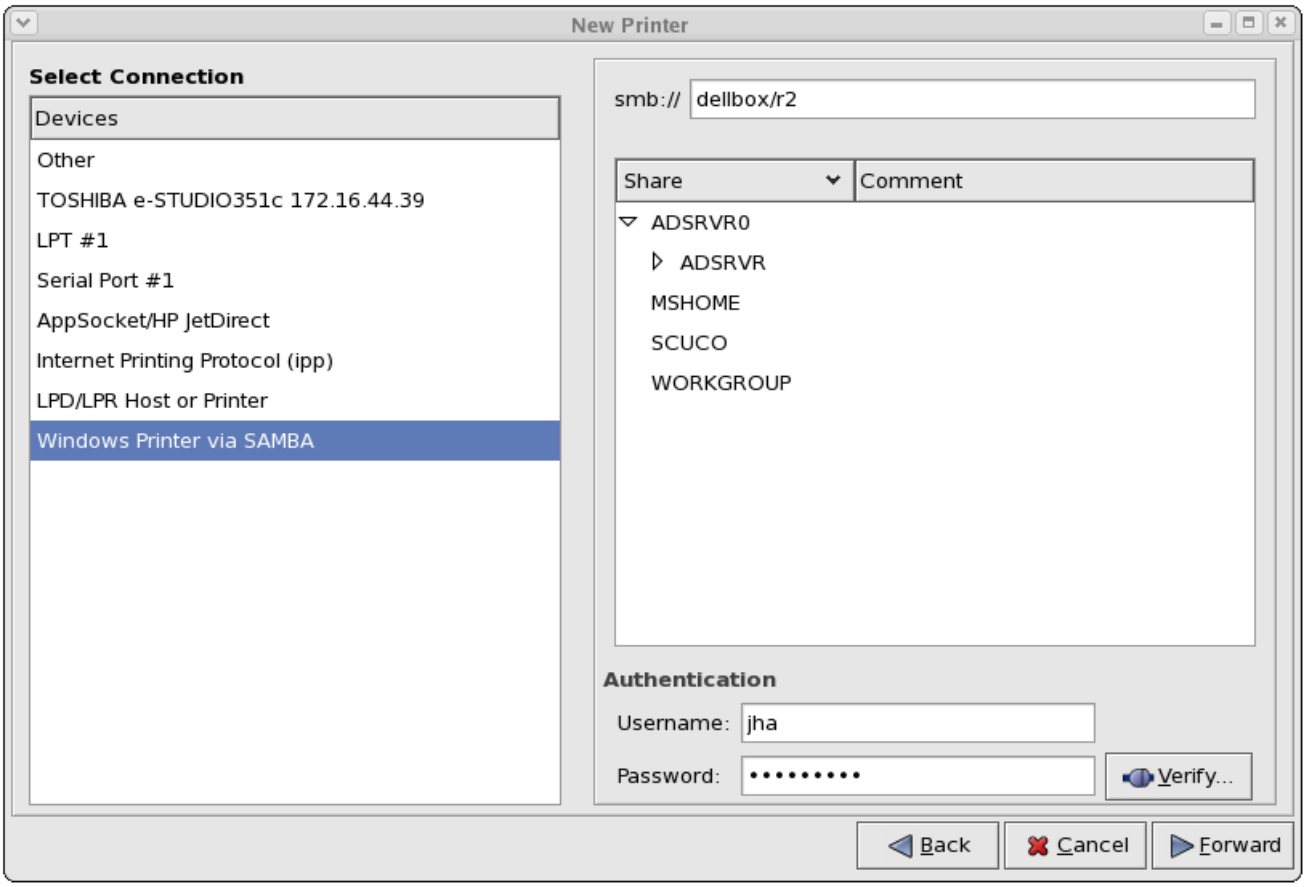
次に、プリンタータイプを選択します。詳細は、「プリンターモデルの選択と完了」 を参照してください。

38.3. SAMBA (SMB)プリンターの追加

Samba (SMB)ベースのプリンター共有を追加するには、メインの Printer Configuration Tool ウィンドウの New Printer ボタンをクリックし、図38.2「プリンターの追加」 にウィンドウを表示します。Printer Name フィールドにプリンターの一意の名前を入力します。プリンター名には、文字、数字、ダッシュ(-)、およびアンダースコア(_)を含めることができます。スペースを含めることはできません。

また、説明 および ロケーション フィールドを使用して、このプリンターをシステムに設定できる他の項目とさらに区別することもできます。これらのフィールドはいずれもオプションで、スペースを含めることができます。

図38.5 SMB プリンターの追加



[D]

図38.5「SMB プリンターの追加」 に示すように、使用可能な SMB 共有が自動的に検出され、Share 列に一覧表示されます。Workgroup の横にある矢印(

)をクリックして展開します。拡張された一覧からプリンターを選択します。

検索しているプリンターが一覧に表示されない場合は、smb:// フィールドに SMB アドレスを入力します。computer name/printer share の形式を使用します。図38.5「SMB プリンターの追加」 では、コンピューター名 は dellbox で、プリンター共有 は r2 です。

ユーザー名 フィールド に、プリンターにアクセスするためのユーザー名を入力します。このユーザーは、SMB システムで存在している必要があり、ユーザーはプリンターへのアクセス権限を持っている必要があります。デフォルトのユーザー名は通常、Windows サーバーの場合は guest で、Samba サーバーの場合は nobody です。

必要に応じて、Username フィールドに指定したユーザーの Password （必要な場合）を入力します。

次に、Verify をクリックして接続をテストできます。確認が成功すると、ダイアログボックスが表

示され、プリンター共有のアクセスを確認します。

次に、プリンタータイプを選択します。詳細は、「プリンターモデルの選択と完了」 を参照してください。

WARNING

Samba プリンターのユーザー名とパスワードは、root および lpd が読み取り可能な暗号化されていないファイルとしてプリンターサーバーに保存されます。したがって、プリンターサーバーに root アクセスを持つ他のユーザーは、Samba プリンターへのアクセスに使用するユーザー名とパスワードを表示できます。

そのため、Samba プリンターにアクセスするためのユーザー名とパスワードを選択する場合は、ローカルの Red Hat Enterprise Linux システムへのアクセスに使用するパスワードとは異なるパスワードを選択することが推奨されます。

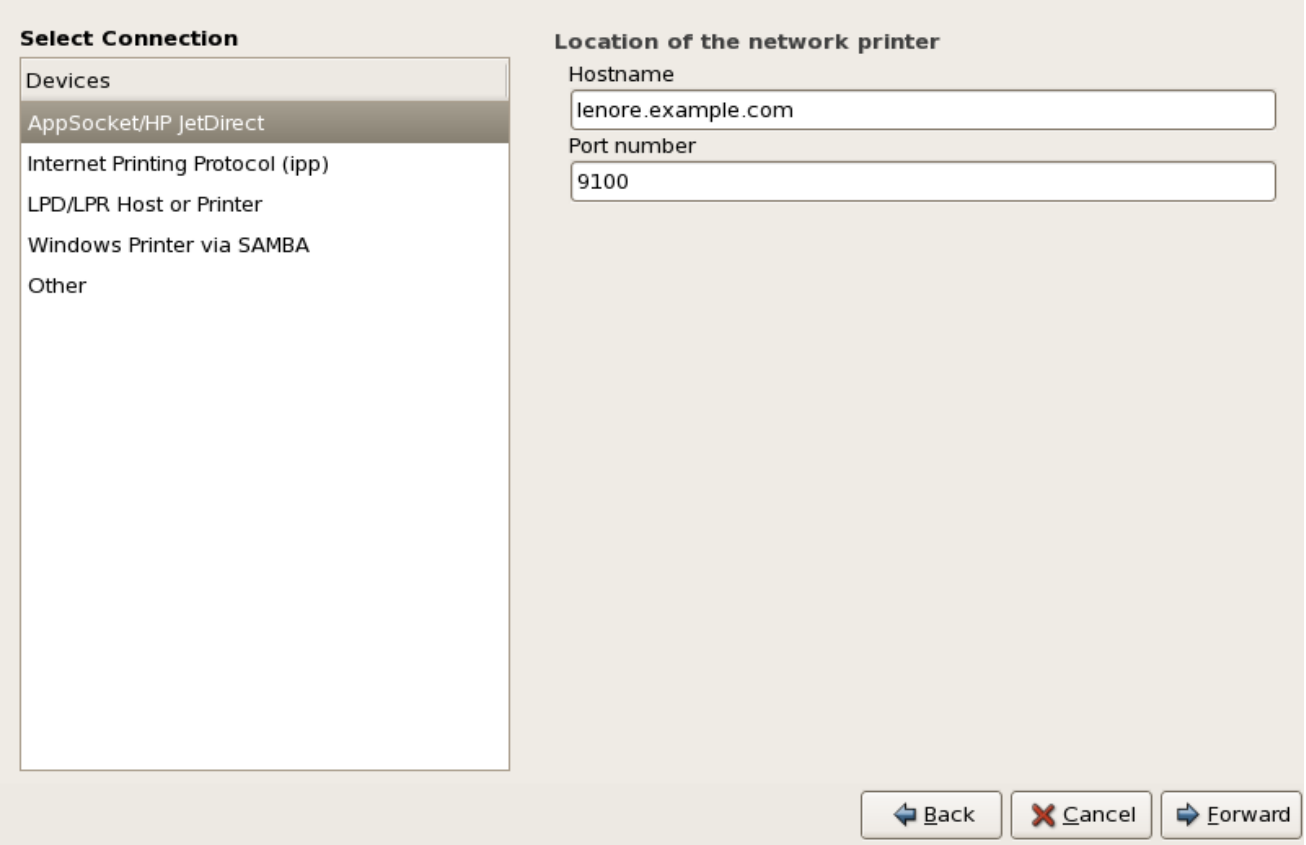
Samba プリンターサーバーで共有するファイルがある場合も、印刷キューで使用されるパスワードとは異なるパスワードを使用することが推奨されます。

38.4. JETDIRECT プリンターの追加

JetDirect または AppSocket の接続されたプリンター共有を追加するには、メインの Printer Configuration Tool ウィンドウの New Printer ボタンをクリックして、図38.2「プリンターの追加」にウィンドウを表示します。Printer Name フィールドにプリンターの一意の名前を入力します。プリンター名には、文字、数字、ダッシュ(-)、およびアンダースコア(_)を含めることができます。スペースを含めること はできません。

また、説明 および ロケーション フィールドを使用して、このプリンターをシステムに設定できる他の項目とさらに区別することもできます。これらのフィールドはいずれもオプションで、スペースを含めることができます。

図38.6 JetDirect プリンターの追加



[D]

進む をクリックして続けます。

以下のオプションのテキストフィールドが表示されます。

- ***hostname - JetDirect プリンターのホスト名または IP アドレス。***
- ***ポート 番号 - 印刷ジョブをリッスンする JetDirect プリンターのポート。デフォルトのポートは 9100 です。***

次に、プリンタータイプを選択します。詳細は、「プリンターモデルの選択と完了」 を参照してください。

38.5. プリンターモデルの選択と完了

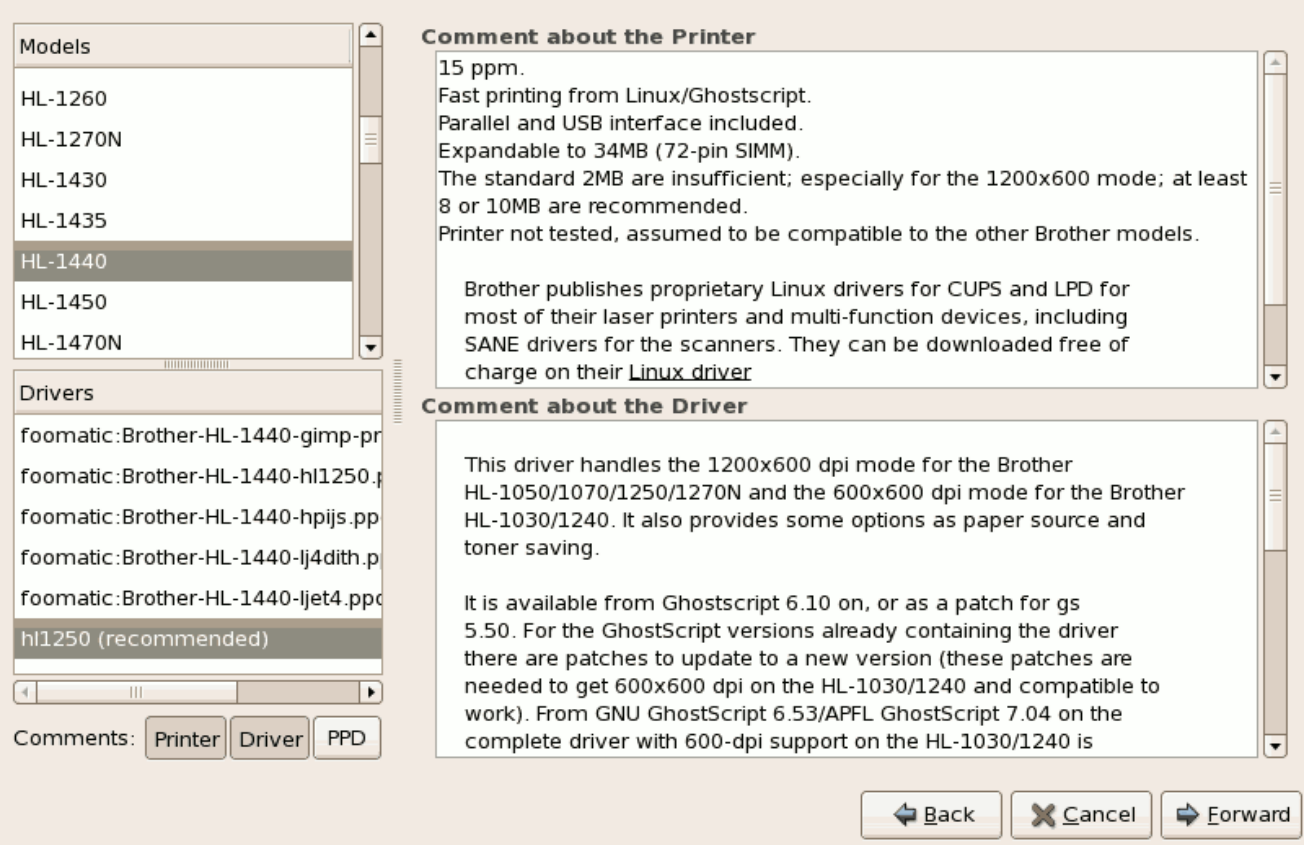
プリンターのキュータイプを適切に選択したら、いずれかのオプションを選択できます。

- *データベースからプリンターを選択 - このオプションを選択した場合は、メーカー の一覧からプリンターの 製造元 を選択します。プリンターの製造元が一覧にない場合は、Generic を選択します。*

- *PPD ファイルの提供 - PPD (PostScript Printer Description)ファイルもプリンターで提供できます。このファイルは通常、製造元によって提供されます。PPD ファイルで提供される場合は、このオプションを選択し、オプションの説明の下にあるブラウザーバーを使用して PPD ファイルを選択できます。*

図38.7「プリンターモデルの選択」 を参照してください。

図38.7 プリンターモデルの選択



[D]

オプションを選択したら、Forward をクリックして続行します。図38.7「プリンターモデルの選択」 が表示されます。ここで、プリンターに対応するモデルとドライバーを選択する必要があります。

推奨される印刷ドライバーは、選択したプリンターモデルに基づいて自動的に選択されます。ただし、別の利用可能なドライバーを選ぶことも可能です。ローカルプリンターはコンピューターに直接接続されているため、プリンターに送信されるデータを処理するにはプリンタードライバーが必要です。

デバイスの PPD ファイル（通常は製造元により提供）がある場合は、Provide PPD file を選択して選択できます。PPD ファイルのファイルシステムを参照するには、Browse をクリックします。

38.5.1. プリンター設定の確認

最後のステップは、プリンターの設定を確認することです。設定が正しい場合は、Apply をクリックして、印刷ジョブを追加します。戻る (Back) をクリックすると、プリンター設定を変更できます。

変更を適用した後、テストページを印刷して、設定が正しいことを確認します。詳細は、「テストページの印刷」 を参照してください。

38.6. テストページの印刷

プリンターを設定したら、テストページを印刷して、プリンターが正しく機能していることを確認する必要があります。テストページを印刷するには、プリンターの一覧から試行するプリンターを選択し、プリンターの Settings タブから テスト ページ の出力 をクリックします。

印刷ドライバーを変更したり、ドライバーオプションを変更した場合は、テストページを印刷して別の設定をテストする必要があります。

38.7. 既存プリンターの修正

既存のプリンターを削除するには、プリンターを選択し、ツールバーの Delete ボタンをクリックします。プリンター設定の削除を確認すると、プリンターの一覧からプリンターが削除されます。

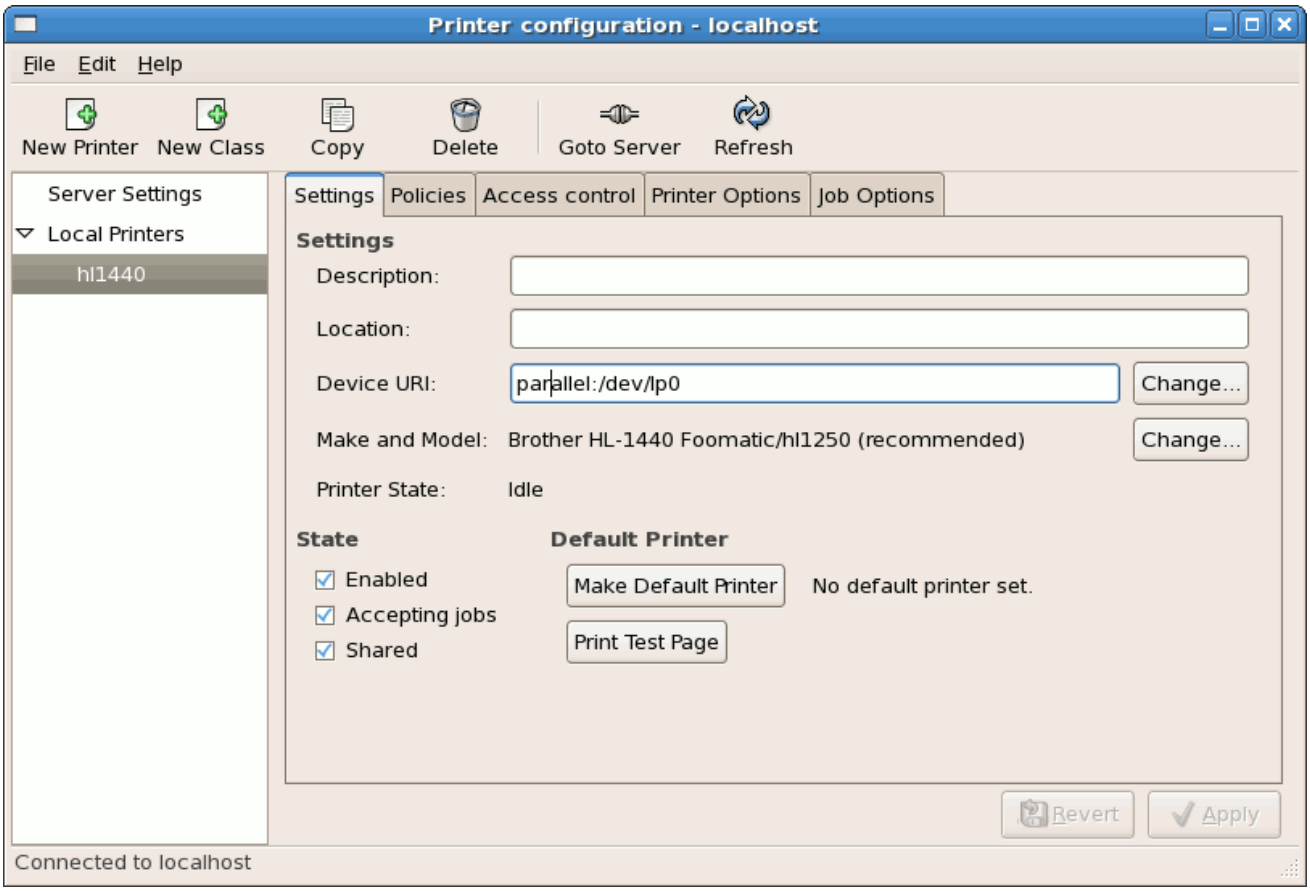
デフォルトのプリンターを設定するには、プリンターの一覧からプリンターを選択し、設定タブのデフォルトプリンターの作成 ボタンをクリックします。

38.7.1. 設定 タブ

プリンターのドライバー設定を変更するには、プリンター 一覧で該当する名前をクリックして、設定 タブを クリック します。

製造元やモデルなどのプリンター設定の変更、デフォルトのプリンターの作成、テストページの印刷、デバイスの場所(URI)の変更などを行うことができます。

図38.8 設定タブ



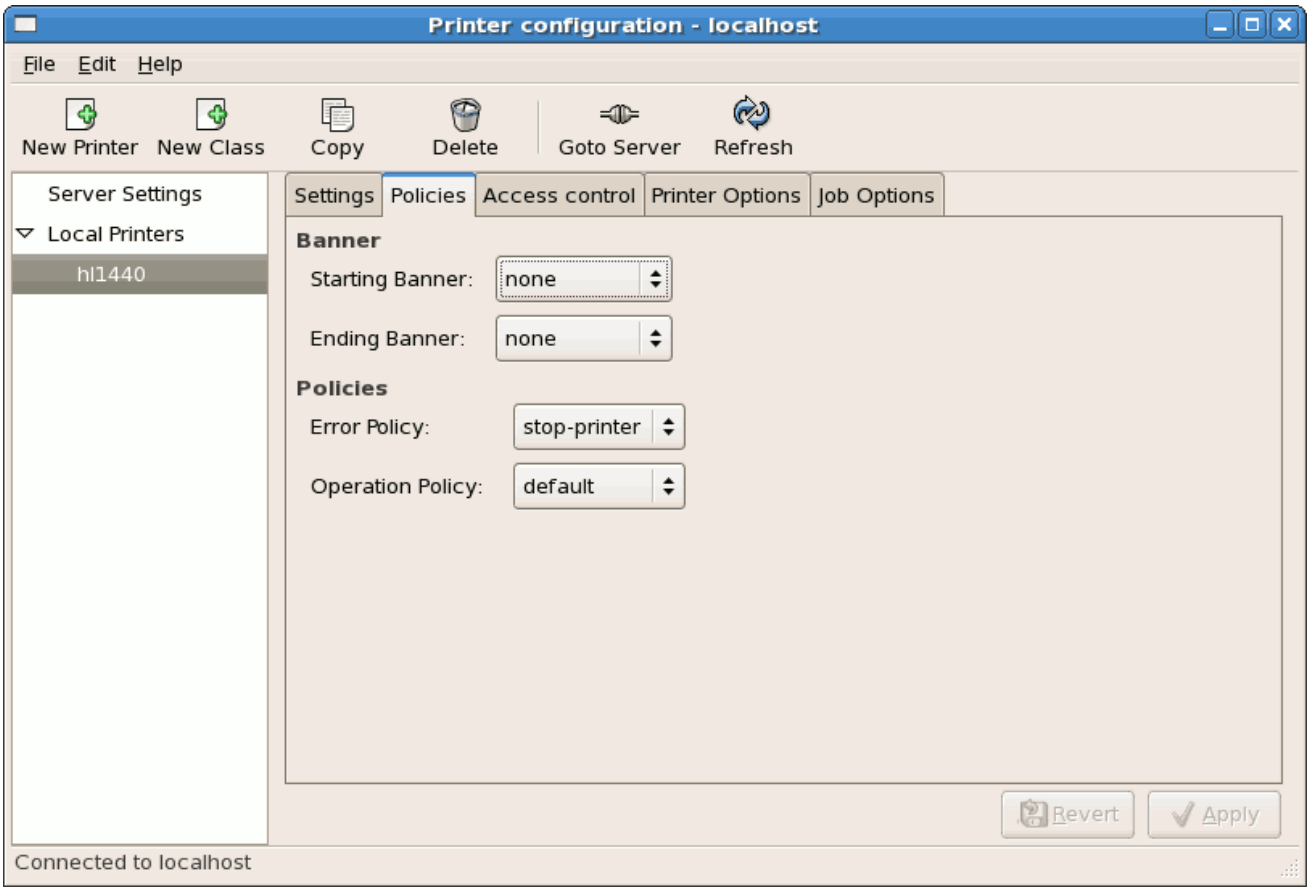
[D]

38.7.2. ポリシー タブ

*印刷出力の設定を変更するには、**Policies** タブをクリックします。*

*たとえば、バナーページ（送信元プリンター、ジョブを開始したユーザー名、印刷中のドキュメントのセキュリティーステータスなど）の印刷ジョブの特徴を説明するページを作成するには、**Starting Banner** または **Ending Banner** ドロップメニューをクリックし、印刷ジョブの性質に最適なオプションを選択します(**topsecret**、**classified**、または **confidential**など)。*

図38.9 ポリシータブ



[D]

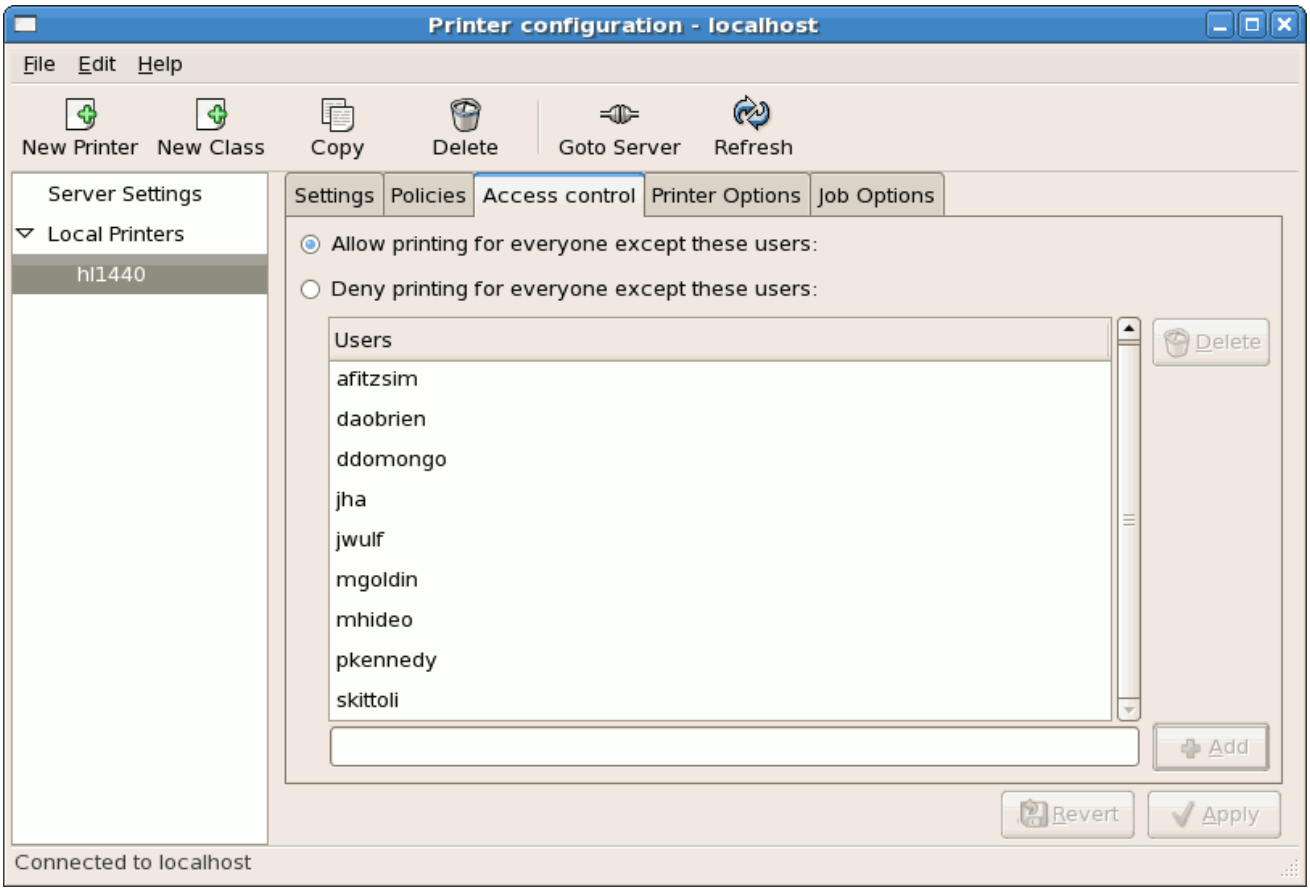
ドロップダウンメニューからオプションを選択して、プリンターの エラーポリシー を設定することもできます。印刷ジョブを中止し、再試行するか、停止するかを選択できます。

38.7.3. アクセス制御 タブ

Access Control タブをクリックして、設定したプリンターへのユーザーレベルのアクセスを変更できます。

テキストボックスを使用してユーザーを追加し、その横にある Add ボタンをクリックします。次に、そのユーザーのサブセットへのプリンターの使用のみを許可するか、それらのユーザーに対する使用を拒否するように選択できます。

図38.10 アクセス制御タブ

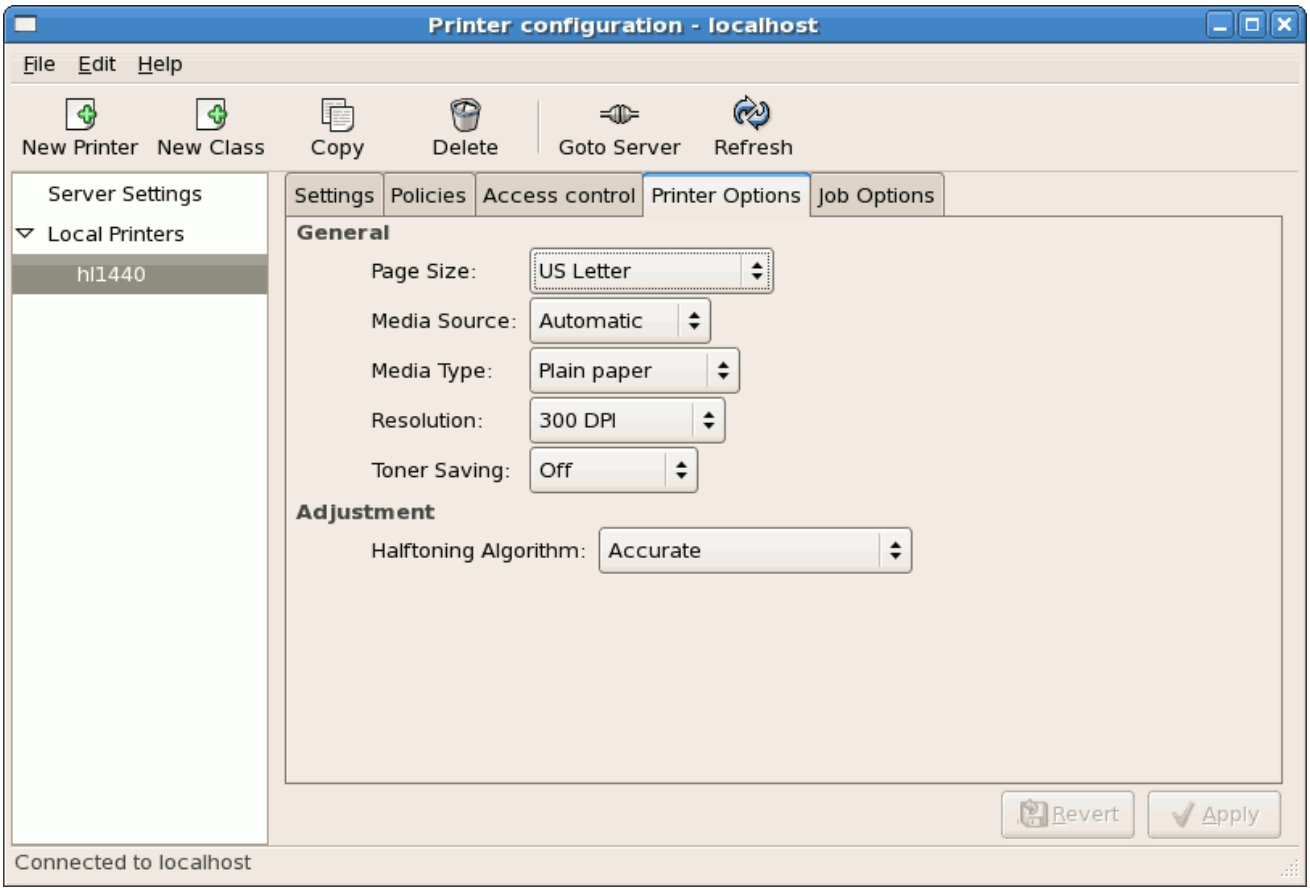


[D]

38.7.4. プリンター および ジョブオプションタブ

プリンターオプション タブに は、 プリンターメディアおよび出力のさまざまな設定オプションが含まれます。

図38.11 プリンターオプションタブ



[D]

- *ページサイズ: ペーパーのサイズを選択できます。オプションには、US Letter、US Legal、A3、および A4 が含まれます。*

- *メディアソース： デフォルトでは Automatic に設定されます。別のトレイからの書籍を使用するように、このオプションを変更します。*

- *メディアタイプ - 文書タイプを変更できます。オプションには、Plain、thick、bond、および transparent が含まれます。*

- *Resolution - 印刷の品質と詳細を設定します。デフォルトは、inch (dpi)ごとに 300 のドットです。*

- *Toner Saving - プリンターがリソースを節約するために少ないトナーを使用してリソースを節約するかどうかを選択します。*

ジョブオプションタブを使用してプリンタージョブオプションを設定することもできます。ドロッ

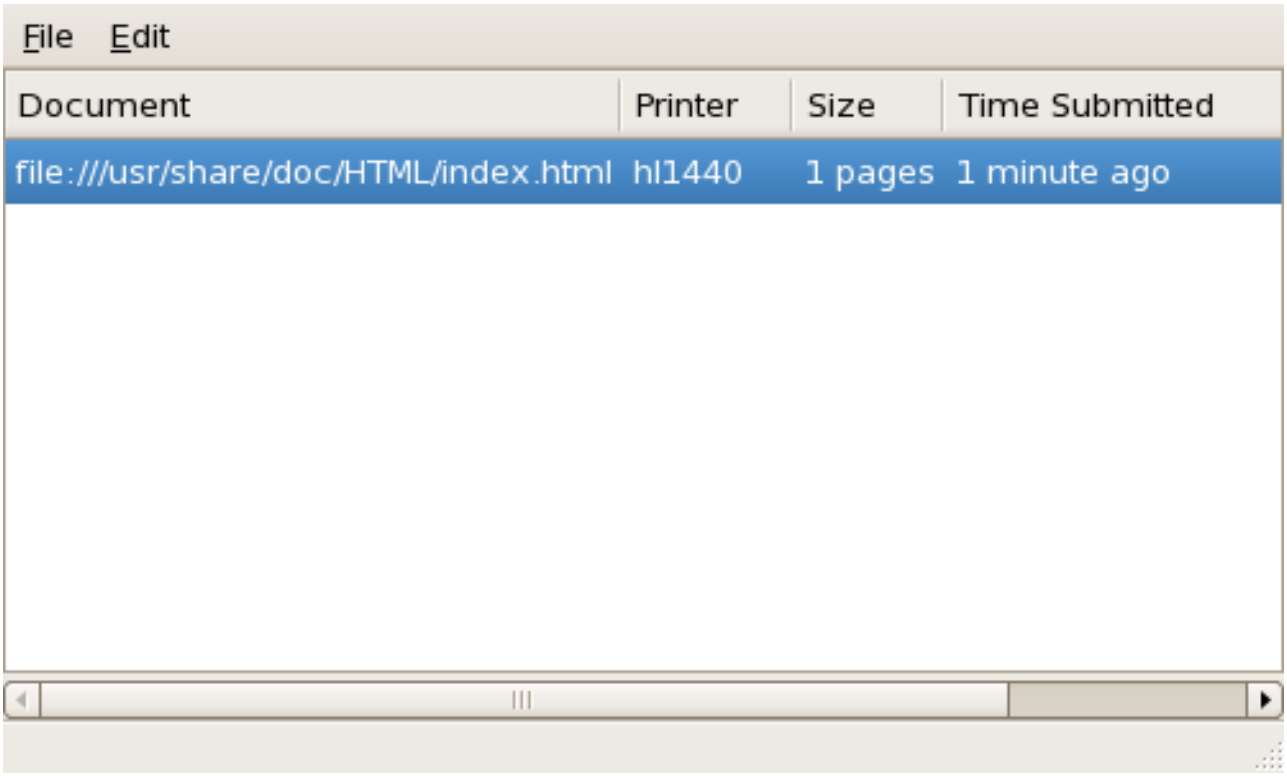
プメニューを使用して、Landscape モード(horizontal モードまたは垂直印刷アウト)、の コピー、スケーリング（印刷可能な領域のサイズを増加または縮小する）など、使用するジョブオプションを選択します。これは、印刷領域の大きさを、より小さな印刷メディアに合わせるのに使用できます。

38.8. 印刷ジョブの管理

Emacs からのテキストファイルの印刷や GIMP からのイメージの出力など、プリンターデーモンに印刷ジョブを送信すると、印刷ジョブが印刷スプールキューに追加されます。印刷のスプールキューはプリンターに送られた印刷ジョブの一覧で、各印刷要求に関する情報 (印刷要求の状態、ジョブ番号など) を表示します。

印刷プロセス中に、パネルの Notification Area に Printer Status アイコンが表示されます。印刷ジョブのステータスを確認するには、図38.12「GNOME 印刷の状態」 のようなウィンドウを表示するプリンターの状態をダブルクリックします。

図38.12 GNOME 印刷の状態



[D]

GNOME Print Status に一覧表示されている特定の印刷ジョブをキャンセルするには、一覧からこれを選択し、プルダウンメニューから Edit > Cancel Documents を選択します。

シェルプロンプトから印刷スプールにある印刷ジョブのリストを表示するには、lpq コマンドを入力します。最後の数行は以下のようになります。

例38.1 lpq 出力の例

```
Rank   Owner/ID              Class  Job Files      Size Time
active user@localhost+902    A      902 sample.txt  2050 01:20:46
```

印刷ジョブをキャンセルするには、lpq コマンドで要求のジョブ番号を見つけ、lprm ジョブ番号 を使用します。たとえば、lprm 902 は、例38.1「lpq 出力の例」 で印刷ジョブをキャンセルします。印刷ジョブをキャンセルするには、適切なパーミッションが必要です。プリンターが接続されているマシンに root としてログインしていない限り、他のユーザーが起動した印刷ジョブをキャンセルできません。

シェルプロンプトから直接ファイルを印刷することもできます。たとえば、コマンド lpr sample.txt はテキストファイル sample.txt を出力します。印刷フィルターはファイルのタイプを決定し、プリンターが理解できる形式に変換します。

38.9. 関連情報

Red Hat Enterprise Linux での印刷の詳細は、以下の資料を参照してください。

38.9.1. インストールされているドキュメント

- マップ lpr: コマンドラインからのファイルの印刷を可能にする lpr コマンドの man ページです。
- man lprm: プリントキューから印刷ジョブを削除するためのコマンドラインユーティリティーの man ページです。
- man mpage - 1 つのドキュメントで複数のページを印刷するためのコマンドラインユーティリティーの man ページです。
- man cupsd: CUPS プリンターデーモンの man ページです。
- man cupsd.conf: CUPS プリンターデーモン設定ファイルの man ページです。
- man classes.conf: CUPS のクラス設定ファイルの man ページです。

38.9.2. 便利な Web サイト

- http://www.linuxprinting.org: 『GNU/Linux Printing』 には、Linux での印刷に関する多くの情報が含まれています。

- http://www.cups.org/ - CUPS のドキュメント、FAQ、および newsgroups。

第39章 自動タスク

Linux では、タスクは指定された期間、指定した日付、またはシステムの負荷平均が指定された数を下回る場合に、自動的に実行されるように設定できます。Red Hat Enterprise Linux は、システムを最新の状態に保つために重要なシステムタスクを実行するように事前設定されています。たとえば、locate コマンドで使用される場所のデータベースは毎日更新されます。システム管理者は、自動化されたタスクを使用して、定期的なバックアップの実行、システムの監視、カスタムスクリプトの実行などを行うことができます。

Red Hat Enterprise Linux には、いくつかの自動タスクユーティリティーが同梱されています。cron、at、および batch です。

39.1. CRON

Cron は、時間、月、曜日、および週の組み合わせに従って繰り返しジョブの実行をスケジュールするために使用できるデーモンです。

cron は、システムが継続的にオンであると想定しています。ジョブのスケジュール時にシステムがオンになっていない場合は、実行されません。1 回限りのジョブをスケジュールするには、「at および Batch」 を参照してください。

cron サービスを使用するには、vixie-cron RPM パッケージがインストールされ、crond サービスが実行されている必要があります。パッケージがインストールされているかを確認するには、rpm -q vixie-cron コマンドを使用します。サービスが実行されているかどうかを確認するには、コマンド /sbin/service crond status を使用します。

39.1.1. Cron ジョブの設定

cron /etc/crontab の主な設定ファイルには、以下の行が含まれます。

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
```

最初の 4 行は、cron ジョブが実行される環境を設定するために使用される変数です。SHELL 変数

は、使用するシェル環境（この例では bash シェル）をシステムに指示しますが、PATH 変数はコマンドの実行に使用するパスを定義します。cron ジョブの出力は、MAILTO 変数で定義されたユーザー名に電子メールで送信されます。MAILTO 変数が空の文字列(MAILTO="")として定義されている場合、電子メールは送信されません。HOME 変数を使用すると、コマンドまたはスクリプトの実行時に使用するホームディレクトリーを設定できます。

/etc/crontab ファイルの各行はジョブを表し、以下の形式になります。

```
minute   hour   day   month   dayofweek   command
```

- *minute - 0 から 59 までの整数*
- *時間 - 0 から 23 までの整数*
- *day: 1 から 31 までの整数（月を指定する場合は有効な日である必要があります）*
- *month - 1 から 12 までの任意の整数（または jan や feb などの月の短縮名）*
- *dayOfWeek - 0 から 7 までの 整数。0 または 7 は日曜日(sun または mon などの曜日の短縮名)を表します。*
- *command - 実行するコマンド（コマンドは ls /proc >> /tmp/proc などのコマンド、またはカスタムスクリプトを実行するコマンドのいずれかになります）*

上記の値のいずれかで、アスタリスク(*)を使用してすべての有効な値を指定できます。たとえば、month 値のアスタリスクは、他の値の制約内の月ごとにコマンドを実行します。

整数間のハイフン(-)は、整数の範囲を指定します。たとえば、1-4 は整数 1、2、3、および 4 を意味します。

コンマで区切られた値のリスト(,)は、リストを指定します。たとえば、3, 4, 6, 8 は、これらの 4 つの特定の整数を示します。

スラッシュ(/)を使用して、ステップの値を指定できます。/<integer> で範囲に従うと、整数値は範

*囲内でスキップできます。たとえば、0-59/2 を使用して、分フィールドで他のすべての分を定義できます。ステップ値はアスタリスクと併用することもできます。たとえば、月のフィールドに */3 の値を使用して、3 カ月ごとにジョブを実行できます。*

ハッシュマーク(#)で始まる行はすべてコメントで、処理されません。

/etc/crontab ファイルにあるように、run-parts スクリプトは、/etc/cron.hourly/、/etc/cron.daily/、/etc/ cron.weekly/、および / etc/cron.monthly/ ディレクトリーのスクリプトを、時、毎日、毎週、毎月ごとに実行します。これらのディレクトリーのファイルはシェルスクリプトである必要があります。

cron ジョブを hourly、daily、weekly、または monthly 以外のスケジュールで実行する必要がある場合は、/etc/cron.d/ ディレクトリーに追加できます。このディレクトリーのすべてのファイルは、/etc/crontab と同じ構文を使用します。例については、例39.1「/etc/crontab の例」 を参照してください。

例39.1 /etc/crontab の例

```
# record the memory usage of the system every monday
# at 3:30AM in the file /tmp/meminfo
30 3 * * mon cat /proc/meminfo >> /tmp/meminfo
# run custom script the first day of every month at 4:10AM
10 4 1 * * /root/scripts/backup.sh
```

root 以外のユーザーは、crontab ユーティリティーを使用して cron ジョブを設定できます。ユーザー定義の crontab はすべて /var/spool/cron/ ディレクトリーに保存され、作成したユーザーのユーザー名を使用して実行されます。crontab をユーザーとして作成するには、そのユーザーとしてログインし、コマンド crontab -e を入力して VISUAL または EDITOR 環境変数で指定されたエディターを使用してユーザーの crontab を編集します。このファイルは、/etc/crontab と同じ形式を使用します。crontab への変更が保存されると、crontab はユーザー名に従って保存され、ファイル /var/spool/cron/ユーザー名 に書き込まれます。

cron デーモンは、変更ごとに /etc/crontab ファイル、/etc/cron.d/ ディレクトリー、および /var/spool/cron/ ディレクトリーをチェックします。変更が見つかった場合は、メモリーに読み込まれます。そのため、crontab ファイルを変更した場合は、デーモンを再起動する必要はありません。

Cron ジョブはランダム間隔で実行できます。これは、ネットワークのオーバーロードを防ぐために高負荷の共有ネットワークに役立ちます。ジョブのランダム化はデフォルトで無効になっていますが、以下のパラメーターを指定して /etc/sysconfig/run-parts ファイルで設定することができます。

- *RANDOMIZE: 1 に設定すると、ランダム化機能が有効になります。0 に設定すると、*

cron ジョブのランダム化は無効になります。

- *RANDOM - 初期ランダムシードを指定します。1 以上の整数値に設定する必要があります。*

- *RANDOMTIME: 1 以上の整数値に設定すると、追加のランダム化レベルが提供されます。*

> *例39.2 /etc/sysconfig/run-parts の例 - Job Randomization の設定*
>
> ```
> RANDOMIZE=1
> RANDOM=4
> RANDOMTIME=8
> ```

39.1.2. Cron へのアクセスの制御

/etc/cron.allow ファイルおよび /etc/cron.deny ファイルは、cron へのアクセスを制限するために使用されます。両方のアクセス制御ファイルの形式は、各行の 1 つのユーザー名です。空白はいずれのファイルでも許可されません。アクセス制御ファイルが変更された場合、cron デーモン(crond)を再起動する必要はありません。アクセス制御ファイルは、ユーザーが cron ジョブの追加または削除を試みるたびに読み込まれます。

root ユーザーは、アクセス制御ファイルにリストされているユーザー名に関係なく、常に cron を使用できます。

cron.allow ファイルが存在する場合は、そこに記載されているユーザーのみが cron を使用でき、cron.deny ファイルは無視されます。

cron.allow が存在しない場合は、cron.deny に記載されているユーザーは cron を使用できません。

39.1.3. サービスの起動と停止

cron サービスを起動するには、コマンド /sbin/service crond start を使用します。サービスを停止するには、コマンド /sbin/service crond stop を使用します。システムの起動時に サービスを開始することが推奨されます。システムの起動時に cron サービスを自動的に開始する方法は、18章 を参照してください。

39.2. AT および BATCH

cron は繰り返しジョブのスケジュールに使用されますが、at コマンドは、特定の時間に 1 回限りのジョブをスケジュールするために使用されます。batch コマンドは、システムの平均負荷が 0.8 未満の場合に実行される 1 回限りのジョブをスケジュールするために使用されます。

at または batch を使用するには、at RPM パッケージがインストールされ、atd サービスが実行されている必要があります。パッケージがインストールされているかを確認するには、rpm -q at コマンドを使用します。サービスが実行されているかどうかを確認するには、コマンド /sbin/service atd status を使用します。

39.2.1. at ジョブの設定

特定の時間に 1 回限りのジョブをスケジュールするには、コマンドを 一 度 に入力します。time は、コマンドを実行する時刻になります。

引数の time は、以下のいずれかになります。

- hh:MM 形式：たとえば、04:00 は 4:00 a.m を指定します。時間がすでに経過している場合は、指定した日に実行されます。
- midnight - 12:00 a.m を指定します。
- noon - 12:00 p.m を指定します。
- teatime - 4:00 p.m を指定します。
- 月名日の形式：たとえば、2002 年 1 月 15 日は 2002 年 1 月 15 日を指定します。年は任意です。
- MMDDYY、MM/DD/YY、または MM.DD.YY 形式：たとえば、2002 年 1 月 15 日の場合は 011502 になります。
- now + time: 時間は分単位、時間、日、または週単位です。たとえば、now + 5 days では、コマンド を 5 日間同時に実行するように指定します。

時間は最初に指定し、その後にオプションの日付を指定する必要があります。時間形式の詳細は、/usr/share/doc/at- <version> /timespec テキストファイルを参照してください。

time 引数を指定して at コマンドを入力すると、at> プロンプトが表示されます。実行するコマンドを入力し、Enter を押して Ctrl+D を入力します。複数のコマンドを指定するには、各コマンドの後に Enter キーを押します。すべてのコマンドを入力したら、Enter を押して空の行に移動し、Ctrl+D と入力します。または、プロンプトでシェルスクリプトを入力し、スクリプトの各行の後に Enter を押してから、空白行で Ctrl+D を入力して終了します。スクリプトを入力すると、使用されるシェルはユーザーの SHELL 環境、ユーザーのログインシェル、または /bin/sh （いずれか最初に見つかった方）で設定されたシェルです。

コマンドまたはスクリプトのセットが標準出力に情報を表示しようとすると、その出力はユーザーに電子メールで送信されます。

コマンド atq を使用して、保留中のジョブを表示します。詳細は、「保留中のジョブの表示」 を参照してください。

at コマンドの使用を制限することができます。詳細は、「at と batch へのアクセスの制御」 を参照してください。

39.2.2. batch ジョブの設定

平均負荷が 0.8 未満の場合に 1 回限りのジョブを実行するには、batch コマンドを使用します。

batch コマンドを入力すると、at> プロンプトが表示されます。実行するコマンドを入力し、Enter を押して Ctrl+D を入力します。複数のコマンドを指定するには、各コマンドの後に Enter キーを押します。すべてのコマンドを入力したら、Enter を押して空の行に移動し、Ctrl+D と入力します。または、プロンプトでシェルスクリプトを入力し、スクリプトの各行の後に Enter を押してから、空白行で Ctrl+D を入力して終了します。スクリプトを入力すると、使用されるシェルはユーザーの SHELL 環境、ユーザーのログインシェル、または /bin/sh （いずれか最初に見つかった方）で設定されたシェルです。負荷平均が 0.8 未満になると、コマンドまたはスクリプトのセットが実行されます。

コマンドまたはスクリプトのセットが標準出力に情報を表示しようとすると、その出力はユーザーに電子メールで送信されます。

コマンド atq を使用して、保留中のジョブを表示します。詳細は、「保留中のジョブの表示」 を参照してください。

batch コマンドの使用を制限することができます。詳細は、「at と batch へのアクセスの制御」 を

参照してください。

39.2.3. 保留中のジョブの表示

保留中の at および batch ジョブを表示するには、atq コマンドを使用します。atq コマンドは、保留中のジョブと、各行にジョブの一覧を表示します。各行は、ジョブ番号、日付、時、ジョブクラス、およびユーザー名形式に従います。ユーザーは、自分のジョブのみを表示できます。root ユーザーが atq コマンドを実行すると、すべてのユーザーのすべてのジョブが表示されます。

39.2.4. その他のコマンドラインオプション

at および batch の追加コマンドラインオプションには以下が含まれます。

表39.1 at および batch のコマンドラインオプション

オプション	説明
-f	プロンプトでコマンドまたはシェルスクリプトを指定する代わりに、ファイルからコマンドまたはシェルスクリプトを読み取ります。
-m	ジョブの完了時にユーザーにメールを送信します。
-v	ジョブが実行される時間を表示します。

39.2.5. at と batch へのアクセスの制御

/etc/at.allow ファイルおよび /etc/at.deny ファイルを使用して、at コマンドおよび batch コマンドへのアクセスを制限できます。両方のアクセス制御ファイルの形式は、各行の 1 つのユーザー名です。空白はいずれのファイルでも許可されません。アクセス制御ファイルが変更された場合、at デーモン (atd)を再起動する必要はありません。アクセス制御ファイルは、ユーザーが at または batch コマンドの実行を試みるたびに読み込まれます。

root ユーザーは、アクセス制御ファイルに関係なく、常に at コマンドおよび batch コマンドを実行できます。

at.allow ファイルが存在する場合、そのファイルにリストされているユーザーのみが at または batch を使用できます。また、at.deny ファイルは無視されます。

at.allow が存在しない場合は、at.deny に記載されているユーザーは at または batch を使用できません。

39.2.6. サービスの起動と停止

サービス で を起動するには、起動時 に /sbin/service コマンドを使用します。サービスを停止するには、停止時に /sbin/service コマンドを使用します。システムの起動時に サービスを開始することが推奨されます。システムの起動時に cron サービスを自動的に開始する方法は、18章 を参照してください。

39.3. 関連情報

自動タスクの設定に関する詳細は、以下のリソースを参照してください。

39.3.1. インストールされているドキュメント

- cron の man ページ - cron の概要
- セクション 1 および 5 の crontab man ページ：セクション 1 の man ページには、crontab ファイルの概要が記載されています。セクション 5 の man ページには、ファイルの形式と、エントリーの例が含まれています。
- /usr/share/doc/at- <version> /timespec には、cron ジョブに指定できる時間の詳細情報が含まれます。
- at の man ページ： at と batch およびそのコマンドラインオプションの説明。

第40章 ログファイル

ログファイル は、システム（カーネル、サービス、実行中のアプリケーションなど）に関するメッセージが含まれるファイルです。各情報にはそれぞれ異なるログファイルがあります。例えば、デフォルトのシステムログファイル、セキュリティーメッセージ専用のログファイル、cron タスク用のログファイルなどです。

ログファイルは、カーネルドライバーのロードを試行するか、システムへの不正なログイン試行を検索する場合など、システムの問題のトラブルシューティングを行う場合に非常に便利です。本章では、ログファイルの場所、ログファイルの閲覧方法、ログファイルの注意すべき項目を説明します。

一部のログファイルは、syslogd と呼ばれるデーモンによって制御されます。syslogd によって維持されるログメッセージの一覧は、/etc/syslog.conf 設定ファイルにあります。

40.1. ログファイルの場所の特定

ほとんどのログファイルは /var/log/ ディレクトリーにあります。httpd や samba などの一部のアプリケーションでは、ログファイル用のディレクトリーが /var/log/ 内にあります。

ログファイルディレクトリー内には、数字の後に番号が付いたファイルが複数あることに気付くかもしれません。これらは、ログファイルがローテーションされる際に作成されます。ログファイルは、ファイルサイズが大きくなり過ぎないようにローテーションが行われます。logrotate パッケージには cron タスクが含まれており、/etc/logrotate.conf 設定ファイルと /etc/logrotate.d/ ディレクトリー内の設定ファイルに従ってログファイルを自動的にローテーションします。デフォルトでは、毎週ローテーションし、以前のログファイルを 4 週間保持するように設定されています。

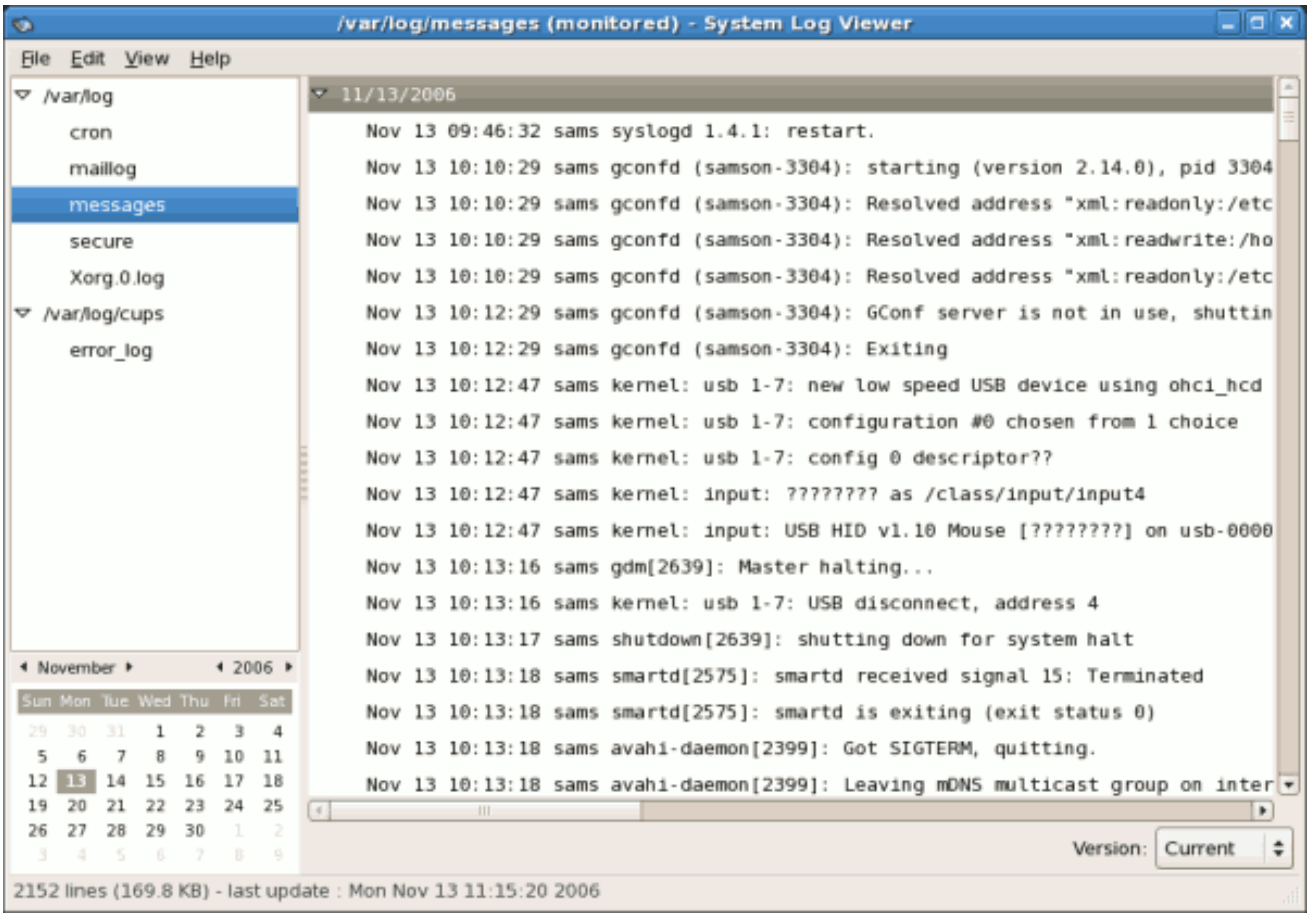
40.2. ログファイルの表示

ほとんどのログファイルはプレーンテキスト形式です。Vi や Emacs などのテキストエディターで表示できます。一部のログファイルは、システム上のすべてのユーザーが読み取り可能ですが、ほとんどのログファイルを読み取るには root 権限が必要になります。

インタラクティブなリアルタイムアプリケーションでシステムログファイルを表示するには、System Log Viewer を使用します。アプリケーションを起動するには、Applications （パネルのメインメニュー）> System > System Logs に移動して、シェルプロンプトでコマンド gnome-system-log を入力します。

このアプリケーションは、存在するログファイルのみを表示します。そのため、図40.1「システムログビューアー」 で表示されている一覧とは異なる場合があります。

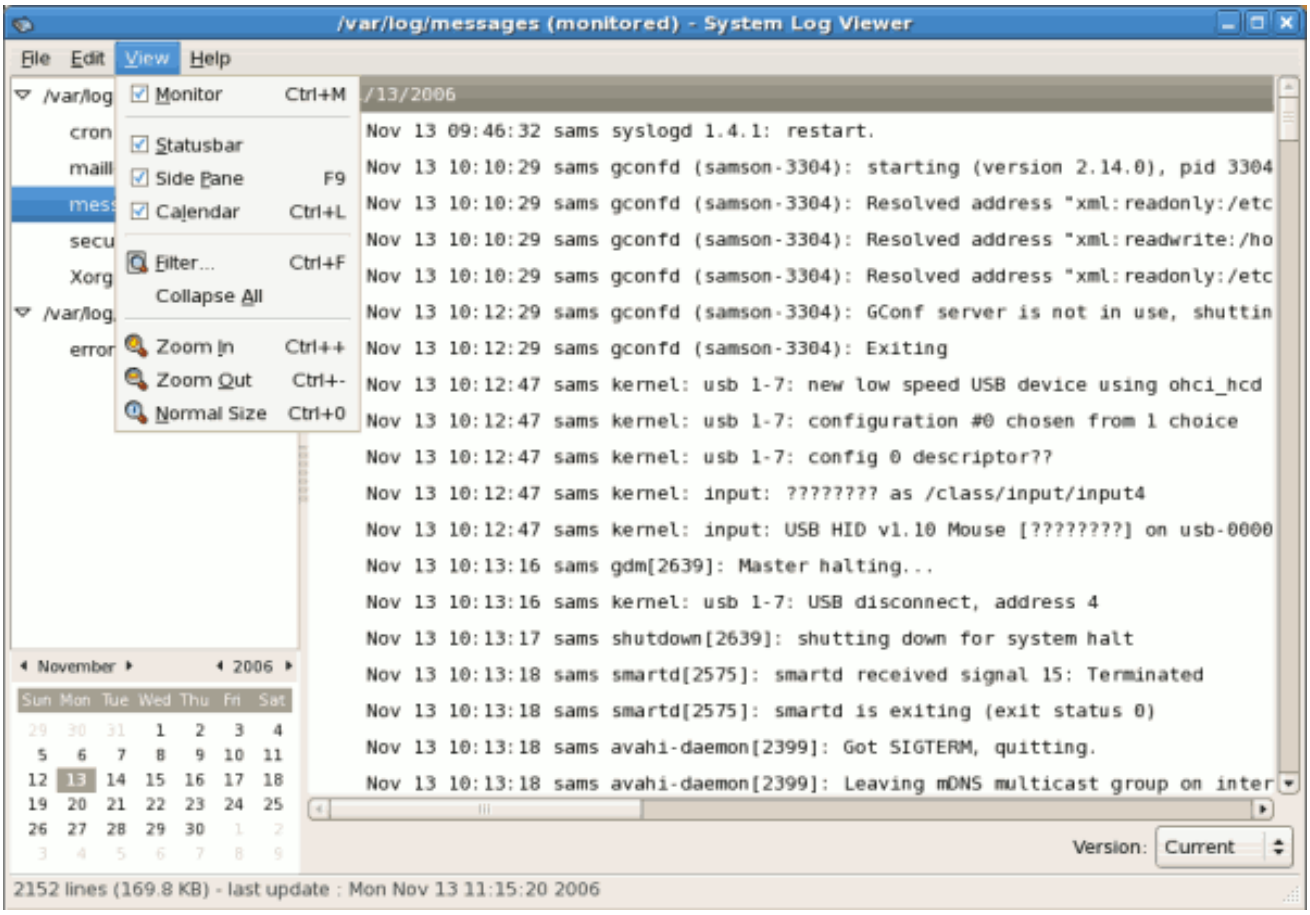
図40.1 システムログビューアー



[D]

*選択したログファイルの内容をフィルターリングするには、メニューから **View** をクリックし、以下に示すように **Filter** を選択します。*

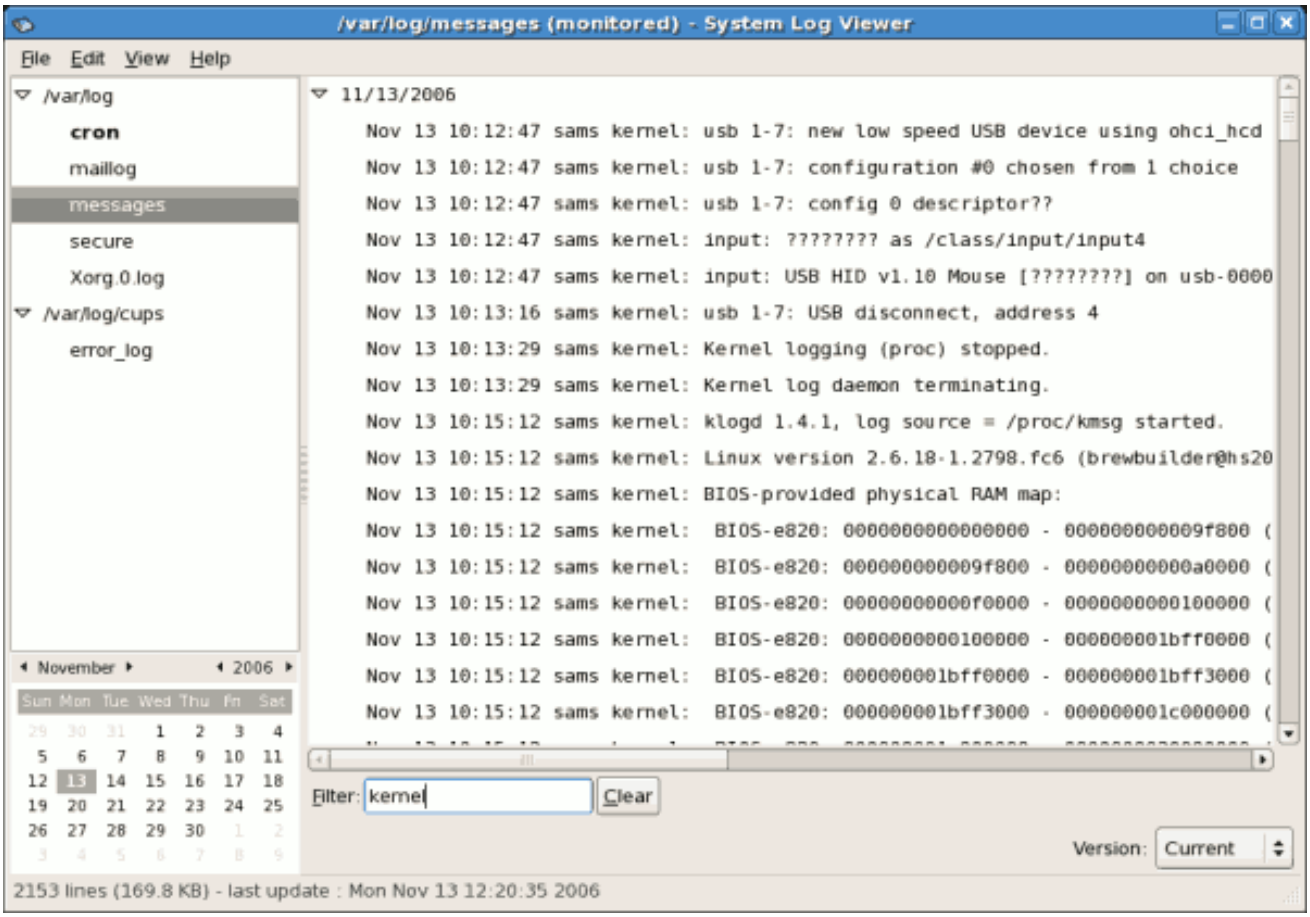
図40.2 システムログビューアー - メニューの表示



[D]

フィルター メニュー 項目 を選択すると、フィルター テキスト フィールドが表示され、フィルターに使用するキーワードを入力できます。フィルターを削除するには、Clear ボタンをクリックします。以下の図はサンプルフィルターを示しています。

図40.3 システムログビューアー - フィルター

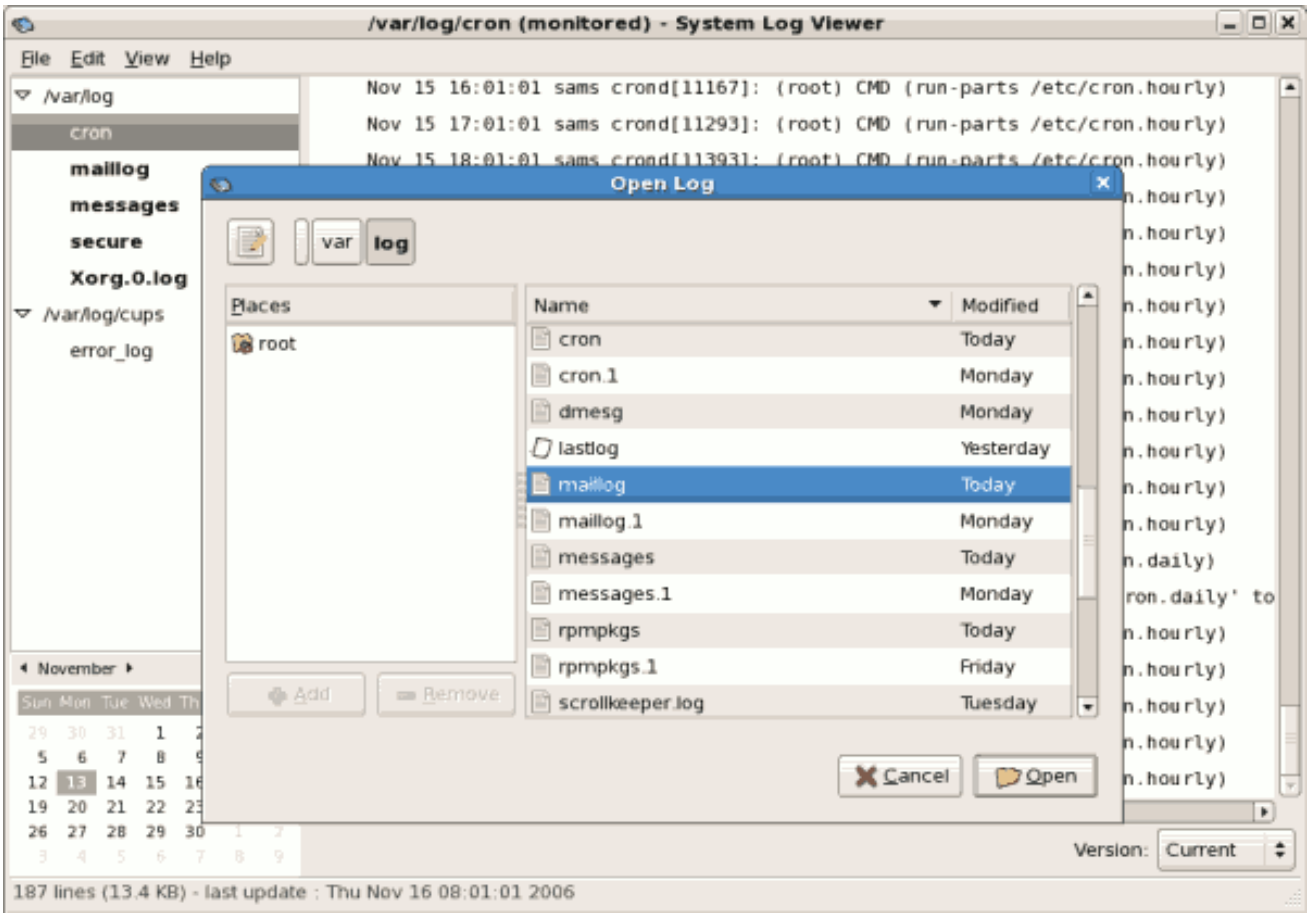


[D]

40.3. ログファイルの追加

*一覧で表示するログファイルを追加するには、**File > Open** の順に選択します。これにより、表示するログファイルのディレクトリーおよびファイル名を選択できる **Open Log** ウィンドウが表示されます。以下の図は **Open Log** ウィンドウを示しています。*

図40.4 ログファイルの追加



[D]

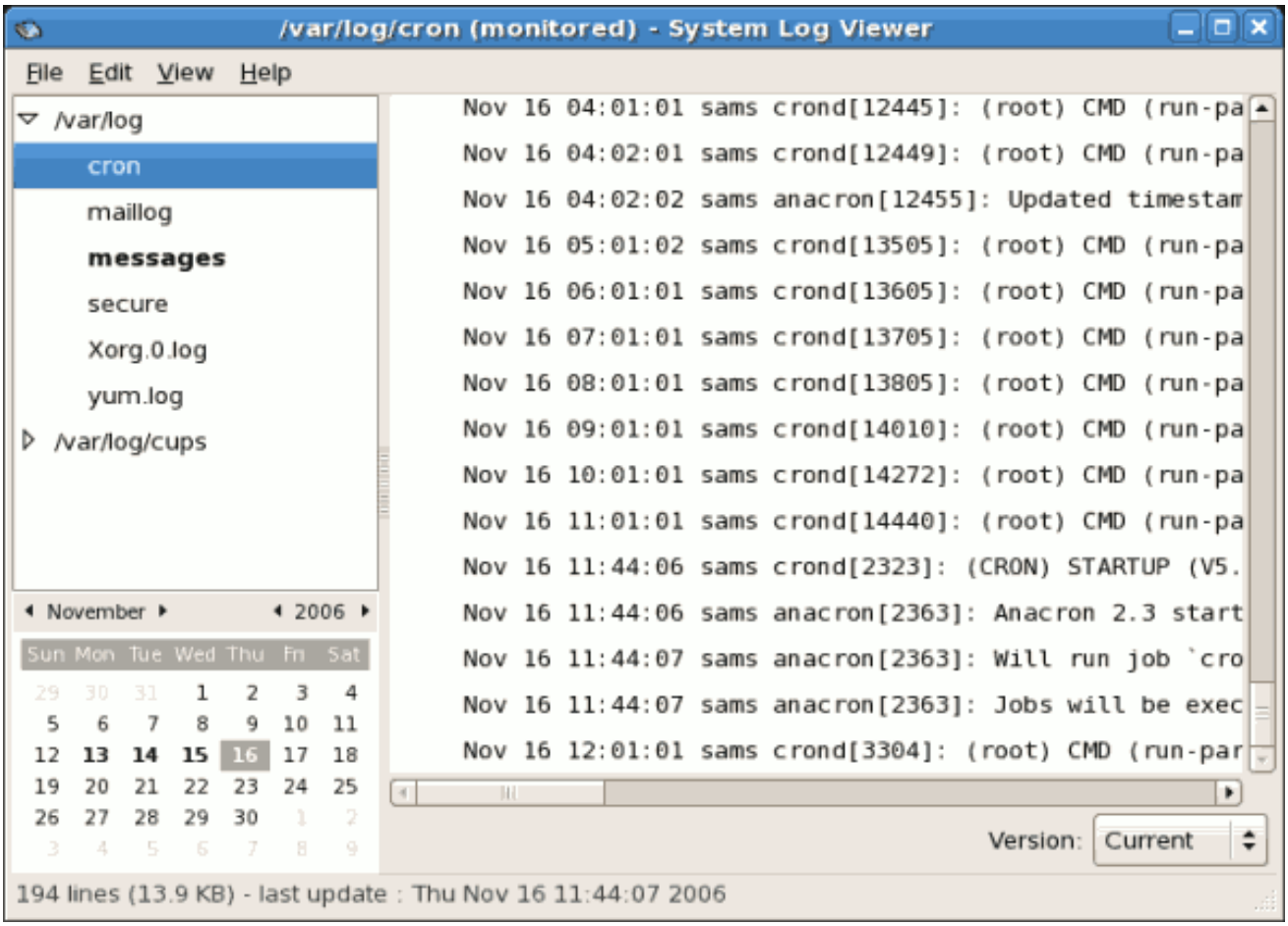
ファイルを開くには、開く ボタンをクリックします。ファイルは表示リストに即座に追加され、ここで選択してコンテンツを表示できます。

また、システムログビューアーでは、ファイル名が.gz で終わる zip 形式のログを開くこともできます。

40.4. ログファイルのモニターリング

システムログビューアー は、デフォルトで開いているすべてのログを監視します。監視されているログファイルに新しい行が追加されると、そのログ名はログの一覧に太字で表示されます。ログファイルが選択または表示されると、新しい行はログファイルの下部に太字で表示され、5 秒後に通常の形式で表示されます。これを以下の図に示します。以下の図は、messages ログファイルの新しいアラートを示しています。ログファイルは太字で表示されます。

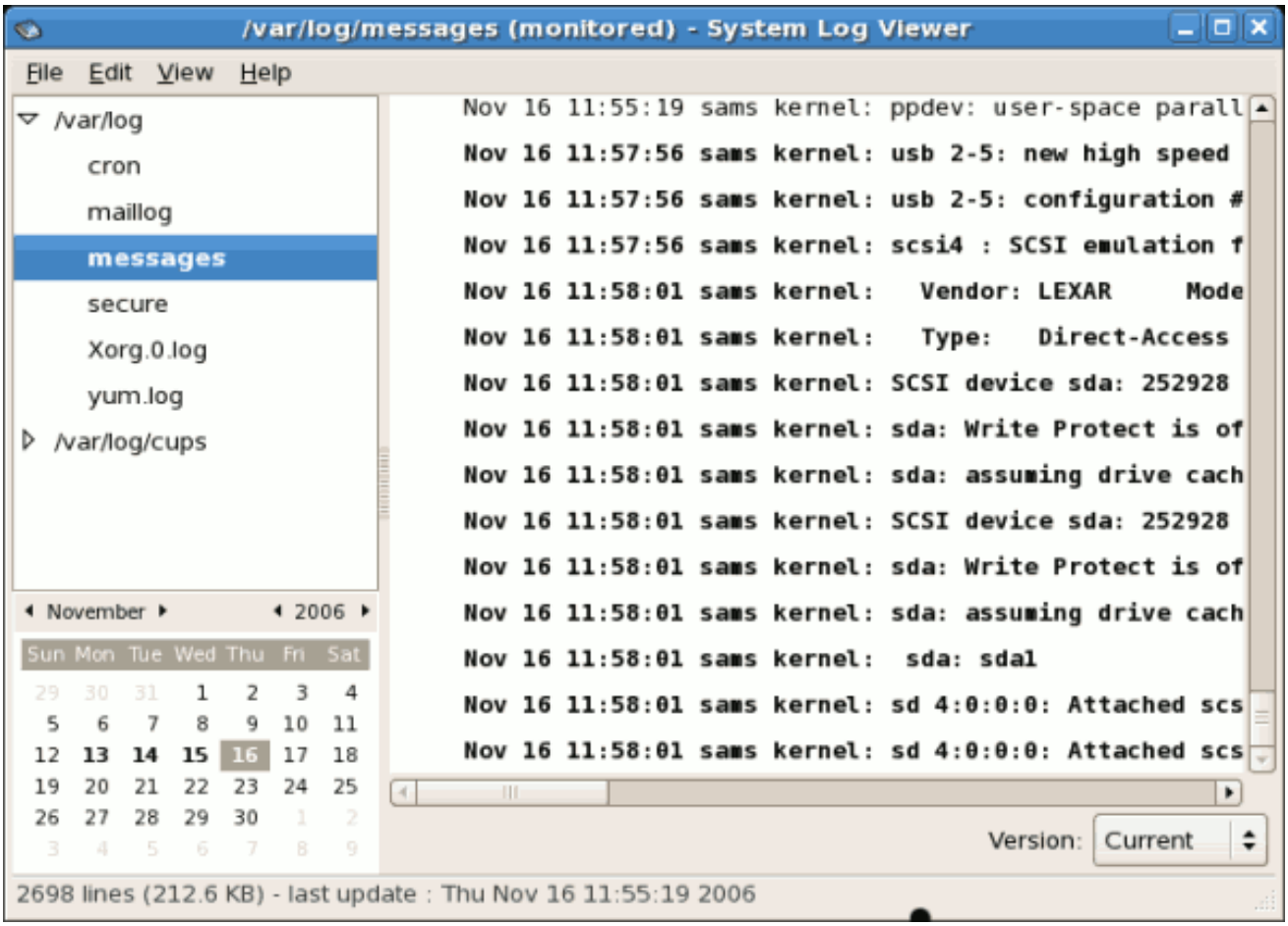
図40.5 ログファイルのアラート



[D]

messages ログファイルをクリックすると、ファイル内のログが表示され、以下のように新しい行が太字で表示されます。

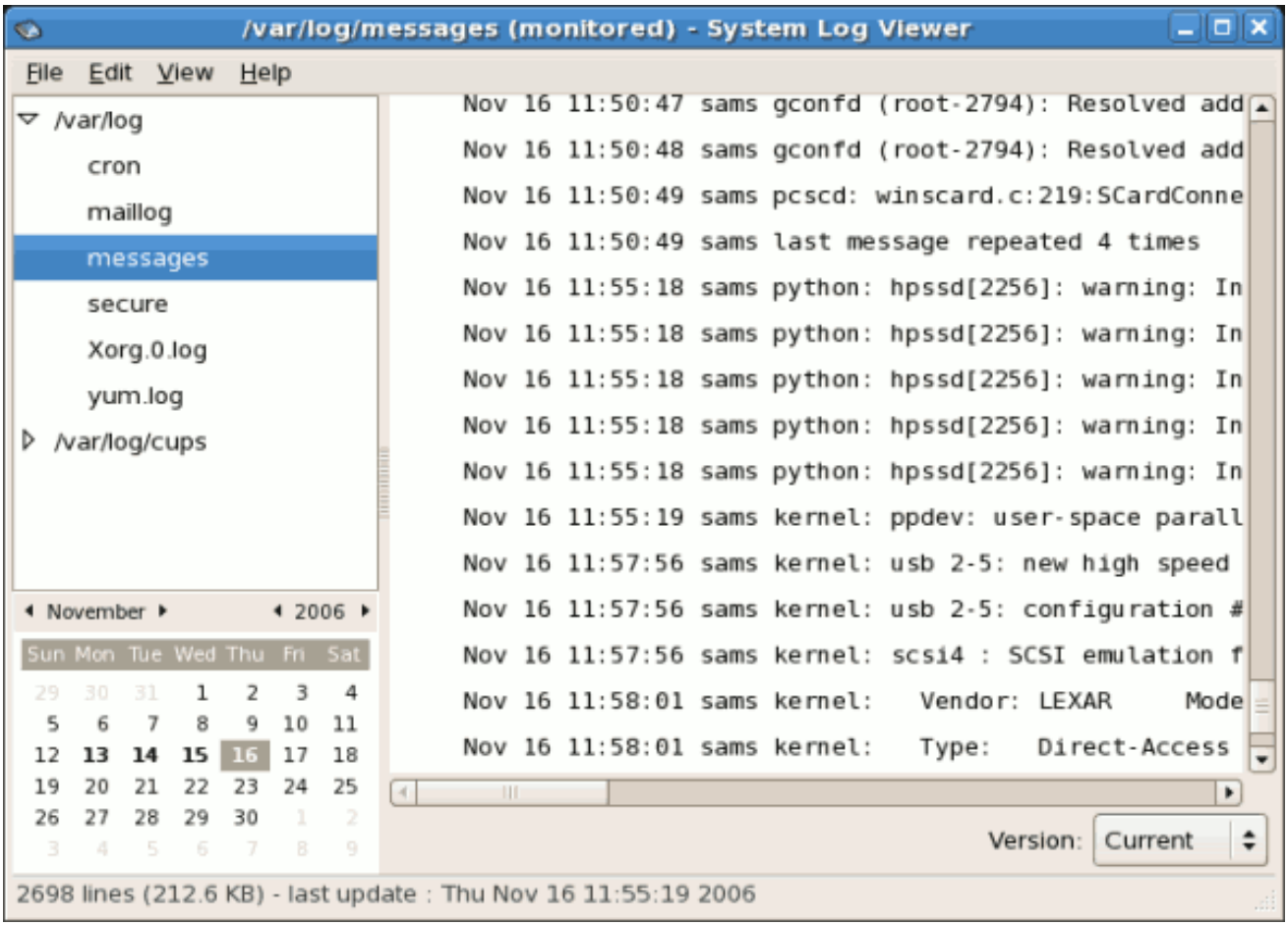
図40.6 ログファイルの内容



[D]

新しい行は、5 秒間太字で表示されますが、その後は通常のフォントで表示されます。

図40.7 5 秒後のログファイルの内容



[D]

パート V. システムモニターリング

システム管理者は、システムパフォーマンスも監視します。Red Hat Enterprise Linux には、管理者がこれらのタスクを支援するツールが含まれています。

第41章 SYSTEMTAP

41.1. はじめに

SystemTap は、実行中の Linux カーネルに関する情報の収集を簡素化するシンプルなコマンドラインインターフェイスとスクリプト言語を提供し、さらに分析できるようにします。複雑なパフォーマンスや機能問題の診断を可能にするために、データを抽出、フィルターリング、および安全に要約できます。

SystemTap を使用すると、スクリプトを SystemTap スクリプト言語で記述できます。これは、C コードカーネルモジュールにコンパイルされ、カーネルに挿入されます。

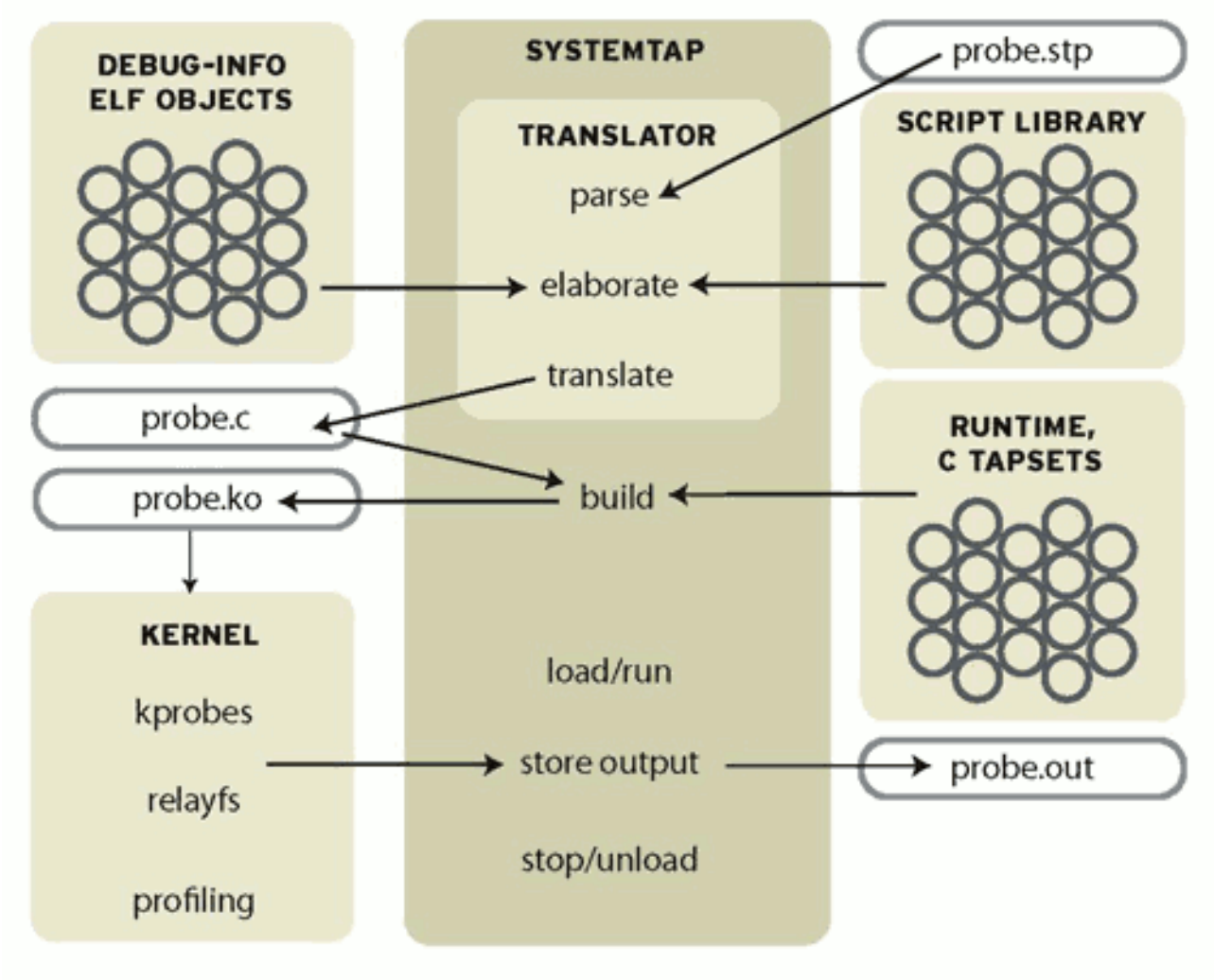
systemtap スクリプトの背後にある重要な概念は、イベントに名前を付け、ハンドラーを提供することです。指定したイベントが発生するたびに、Linux カーネルはハンドラーをクイックサブルーチンであるかのように実行し、再開します。関数の入力または終了、タイマーの有効期限、systemtap セッションの開始や停止など、さまざまな種類のイベントがあります。ハンドラーは一連のスクリプト言語のステートメントで、イベント発生時に実行する作業を指定します。この作業には通常、イベントコンテキストからのデータの抽出、内部変数への保存、または結果の出力が含まれます。

41.2. 実装

SystemTap は、コンパイラー指向のアプローチを使用してインストルメンテーションを生成します。この説明で使用される SystemTap の全体的な図は、図41.1「SystemTap でのデータの流れ」 "Flow of data in SystemTap" を参照してください。ダイアグラムの右上隅にある probe.stp は、開発者が作成したプローブスクリプトです。これは、トランスレーターによって解析され、ツリーを解析します。この間、入力で構文エラーが確認されます。その後、トランスレーターが実行され、スクリプトライブラリーから追加のコードを取得し、デバッグ情報からプローブポイントと変数の場所を決定します。調査が完了すると、トランスレーターは C のカーネルモジュールである probe.c を生成できます。

probe.c ファイルは、GCC コンパイラーを使用して通常のカーネルモジュール probe.ko にコンパイルされます。コンパイルにより、ランタイムライブラリーからサポートコードがプルされる場合があります。GCC が probe.ko を生成すると、SystemTap デーモンが起動してインストルメンテーションモジュールの出力が収集されます。インストルメンテーションモジュールはカーネルに読み込まれ、データ収集が開始されます。インストルメンテーションモジュールからのデータは、relayfs を介してユーザー空間に転送され、デーモンによって表示されます。ユーザーが Control-C に到達すると、デーモンはモジュールをアンロードし、データ収集プロセスもシャットダウンします。

図41.1 SystemTap でのデータの流れ



[D]

41.3. SYSTEMTAP の使用

SystemTap は、SystemTap スクリプトを C に変換することで動作し、システム C コンパイラーを実行してそこからカーネルモジュールを作成します。モジュールが読み込まれると、カーネルにフックしてプローブされたイベントをすべてアクティベートします。その後、イベントが発生すると、コンパイルされたハンドラーが実行されます。最終的にセッションが停止し、フックが切断され、モジュールが削除されます。プロセス全体が 1 つのコマンドラインプログラム stap から実行されます。

41.3.1. トレーシング

プローブの最も単純な種類は、イベントを追跡することです。これは、ストラテジー的に配置された印刷ステートメントをプログラムに挿入する効果です。多くの場合、これは問題の解決の最初のステップです。発生したものの履歴を確認して確認します。

このスタイルのインストルメンテーションは最も単純なものです。systemtap に対して各イベントに何かを印刷するよう依頼するだけです。これをスクリプト言語で表現するには、プローブの場所と印

刷先を指示する必要があります。

41.3.1.1. プローブの場所

SystemTap は、多くの組み込みイベントをサポートします。systemtap が同梱するスクリプトのライブラリー（それぞれをtapset と呼ばれます）は、組み込みファミリーの用語で定義された追加のスクリプトを定義できます。詳細は、stapprobes の man ページを参照してください。これらのイベントはすべて、ドットで区切られたパラメーター化された識別子のような統一された構文を使用して名前が付けられます。

表41.1 SystemTap イベント

イベント	説明
begin	systemtap セッションの開始。
end	systemtap セッションの終了
kernel.function("sys_open")	カーネル内の sys_open という名前の関数へのエントリー。
syscall.close.return	close システムコールから返されます。
module("ext3").statement(0xdeadbeef)	ext3 ファイルシステムドライバーで対処された命令。
timer.ms(200)	200 ミリ秒ごとに実行されるタイマー。

すべての関数エントリーを追跡し、ソースファイル（カーネルの net/socket.c など）で終了するデモケースとして を使用します。kernel.function プローブポイントを使用すると、systemtap がカーネルのデバッグ情報を調べ、オブジェクトコードをソースコードに関連付けるため、これを簡単に表現できます。デバッガーのように機能します。名前や配置が可能な場合は、プローブできます。関数エントリーに kernel.function ("@net/socket.c") を使用し、終了に kernel.function ("*@net/socket.c").return を使用します。関数名の部分と、その後の @FILENAME 部分でのワイルドカードの使用に注意してください。ワイルドカードをファイル名にしたり、検索を正確に制限したい場合はコロン(:)と行番号を追加することもできます。systemtap はプローブポイントに一致する場所に個別のプローブを配置するため、いくつかのワイルドカードを数百または数千のプローブに拡張できるため、質問する内容に注意してください。*

プローブポイントを特定すると、systemtap スクリプトのスケルトンが表示されます。probe キーワードにより、プローブポイントまたはコンマ区切りのリストが導入されました。以下の { および } の中括弧は、一覧表示されたすべてのプローブポイントに対してハンドラーを囲みます。

このスクリプトはそのまま実行できますが、空のハンドラーでは出力はありません。2 つの行を新しいファイルに配置します。stap -v FILE を実行します。^C でいつでも終了します。（ -v オプション

は systemtap に対して、処理中により詳細なメッセージを出力するように指示します。他のオプションを確認するには、-h オプションを試してください。）

41.3.1.2. 印刷する内容

入力および終了された各関数に関心があるため、関数名が含まれる各行をそれぞれに出力する必要があります。この一覧を読みやすくするために、systemtap は、他のトレース関数が呼び出された関数をより深くネストできるように、行を識別する必要があります。各プロセスを、同時に実行できる他のプロセスとは別に指示するには、systemtap もその行にプロセス ID を出力する必要があります。

第42章 システム情報の収集

システムの設定方法を理解する前に、基本的なシステム情報を収集する方法を理解する必要があります。たとえば、空きメモリーの量、使用可能なハードドライブの領域、ハードドライブのパーティション設定方法、および実行中のプロセスを見つける方法を理解する必要があります。本章では、簡単なコマンドと簡単なプログラムを使用して、Red Hat Enterprise Linux システムからこのタイプの情報を取得する方法を説明します。

42.1. システムプロセス

ps ax コマンドは、他のユーザーが所有するプロセスなど、現在のシステムプロセスの一覧を表示します。各プロセスとともに所有者を表示するには、ps aux コマンドを使用します。このリストは静的リストです。つまり、コマンドの呼び出し時に実行した内容のスナップショットです。実行中のプロセスを定期的に更新した一覧が必要な場合は、以下のように top を使用します。

ps 出力は長くなる可能性があります。画面をスクロールしないようにするには、less でパイプできます。

```
ps aux | less
```

ps コマンドを grep コマンドと組み合わせて使用すると、プロセスが実行されているかどうかを確認できます。たとえば、Emacs が実行されているかどうかを確認するには、次のコマンドを使用します。

```
ps ax | grep emacs
```

top コマンドは、現在実行中のプロセスと、メモリーや CPU 使用率などの重要な情報を表示します。この一覧はどちらもリアルタイムおよびインタラクティブです。top コマンドの出力例を以下に示します。

```
top - 15:02:46 up 35 min,  4 users,  load average: 0.17, 0.65, 1.00
Tasks: 110 total,   1 running, 107 sleeping,   0 stopped,   2 zombie
Cpu(s): 41.1% us,  2.0% sy,  0.0% ni, 56.6% id,  0.0% wa,  0.3% hi,  0.0% si
Mem:    775024k total,   772028k used,     2996k free,    68468k buffers
Swap:  1048568k total,      176k used,  1048392k free,   441172k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 4624 root      15   0 40192  18m 7228 S 28.4  2.4   1:23.21 X
 4926 mhideo    15   0 55564  33m 9784 S 13.5  4.4   0:25.96 gnome-terminal
 6475 mhideo    16   0  3612  968  760 R  0.7  0.1   0:00.11 top
 4920 mhideo    15   0 20872  10m 7808 S  0.3  1.4   0:01.61 wnck-applet
    1 root      16   0  1732  548  472 S  0.0  0.1   0:00.23 init
    2 root      34  19     0    0    0 S  0.0  0.0   0:00.00 ksoftirqd/0
    3 root       5 -10     0    0    0 S  0.0  0.0   0:00.03 events/0
```

```
 4 root      6 -10     0    0    0 S  0.0  0.0   0:00.02 khelper
 5 root      5 -10     0    0    0 S  0.0  0.0   0:00.00 kacpid
29 root      5 -10     0    0    0 S  0.0  0.0   0:00.00 kblockd/0
47 root      16   0    0    0    0 S  0.0  0.0   0:01.74 pdflush
50 root      11 -10     0    0    0 S  0.0  0.0   0:00.00 aio/0
30 root      15   0    0    0    0 S  0.0  0.0   0:00.05 khubd
49 root      16   0    0    0    0 S  0.0  0.0   0:01.44 kswapd0
```

top を終了するには、q キーを押します。

表42.1「インタラクティブな top コマンド」 には、top で使用できる便利な対話型のコマンドが含まれています。詳細は、top(1)の man ページを参照してください。

表42.1 インタラクティブな top コマンド

コマンド	説明
スペース	ディスプレイを直ちに更新します。
h	ヘルプ画面の表示
k	プロセスを強制終了します。プロセス ID およびプロセスに送信するシグナルがプロンプトされます。
n	表示されるプロセス数を変更します。番号を入力するようプロンプトされます。
u	ユーザー別に並べ替えます。
M	メモリー使用量で並べ替えます。
P	CPU 使用率で並べ替えます。

top 用のグラフィカルインターフェイスを使用する場合は、GNOME System Monitor を使用できます。デスクトップから起動するには、システム > 管理 > システム モニター を選択するか、シェルプロンプト(XTerm など)で gnome-system-monitor と入力します。Process Listing タブを選択します。

GNOME System Monitor を使用すると、実行中のプロセスの一覧でプロセスを検索できます。Gnome System Monitor を使用すると、すべてのプロセス、プロセス、またはアクティブなプロセスを表示することもできます。

Edit メニュー項目を使用すると、以下が可能になります。

- *プロセスを停止します。*
- *プロセスを続行または開始します。*
- *プロセスを終了します。*
- *プロセスを強制終了します。*
- *選択したプロセスの優先度を変更します。*
- *システムモニター設定を編集します。これには、リストを更新し、System Monitor ウィンドウに表示されるプロセスフィールドを選択する間隔の秒数を変更することが含まれます。*

View メニュー項目では、以下を実行できます。

- *アクティブなプロセスのみを表示します。*
- *すべてのプロセスを表示します。*
- *プロセスを表示します。*
- *プロセスの依存関係を表示します。*
- *プロセスを非表示にします。*
- *非表示のプロセスを表示します。*

- *メモリーマップを表示します。*

- *選択したプロセスで開いているファイルを表示します。*

プロセスを停止するには、プロセスを選択し、プロセスの 終了 をクリックし ます。または、プロセスを選択して、メニューの Edit をクリックして、 Stop Process を選択します。

特定の列で情報を並べ替えるには、列の名前をクリックします。これにより、選択した列で情報を昇順で並べ替えます。コラムの名前を再度クリックして、昇順と降順のソートを切り替えます。

図42.1 GNOME システムモニター

[D]

42.2. MEMORY USAGE

free コマンドは、システムの物理メモリーとスワップ領域の合計量と、カーネルバッファーで使用済み、共有済み、およびキャッシュされているメモリー量を表示します。

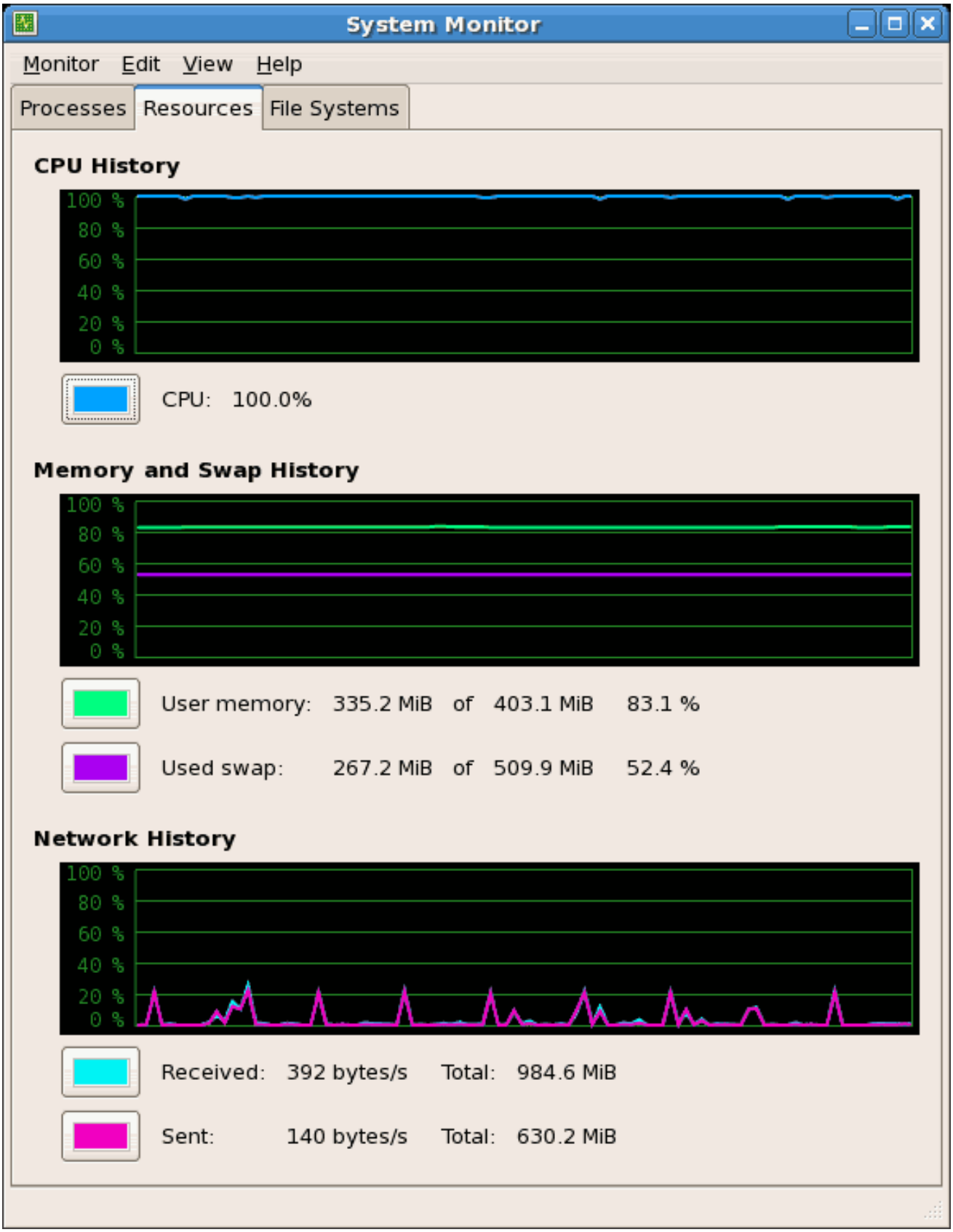
```
             total       used       free     shared    buffers     cached
Mem:        645712     549720      95992          0     176248     224452
-/+ buffers/cache:     149020     496692
Swap:      1310712          0    1310712
```

free -m コマンドは、読み取りが簡単な同じ情報をメガバイト単位で表示します。

```
             total       used       free     shared    buffers     cached
Mem:           630        536         93          0        172        219
-/+ buffers/cache:        145        485
Swap:         1279          0       1279
```

フリー の のグラフィカルインターフェイスを好む場合は、GNOME System Monitor を使用できます。デスクトップから起動するには、システム > 管理 > システム モニター に移動するか、シェルプロンプト(XTerm など)で gnome-system-monitor と入力します。Resources タブをクリックします。

図42.2 GNOME システムモニター - リソースタブ



[D]

42.3. ファイルシステム

df コマンドは、システムのディスク領域の使用量を報告します。シェルプロンプトでコマンド df を

入力すると、出力は以下のようになります。

```
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/mapper/VolGroup00-LogVol00
                      11675568   6272120   4810348  57% / /dev/sda1
                        100691      9281     86211  10% /boot
none                    322856         0    322856   0% /dev/shm
```

デフォルトでは、このユーティリティーは、1 キロバイトブロック単位でパーティションのサイズと、使用中および利用可能なディスク領域の容量をキロバイト単位で表示します。メガバイトおよびギガバイトで情報を表示するには、df -h コマンドを使用します。-h 引数は、人間が判読できる形式を表します。出力は以下の例のようになります。

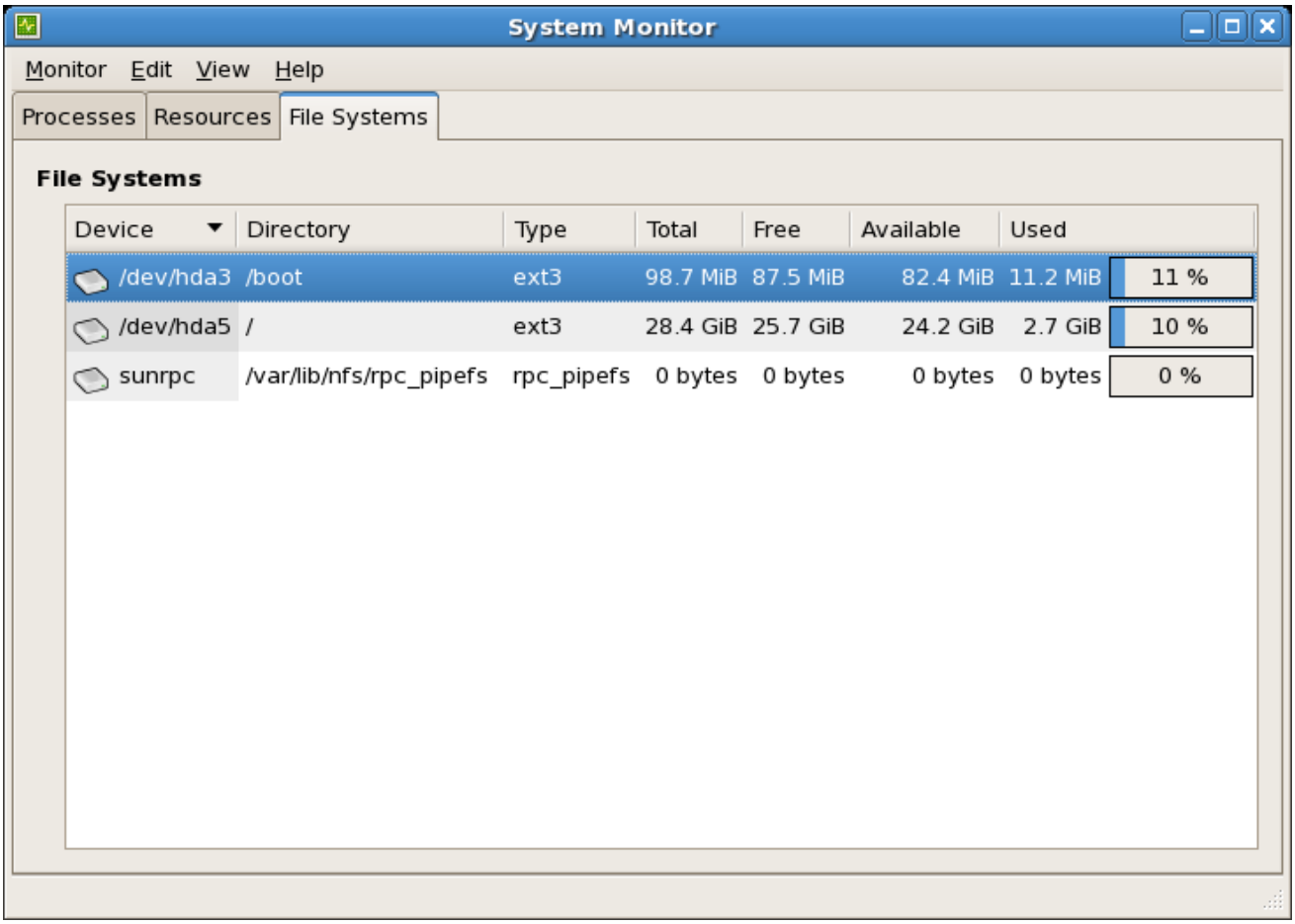
```
Filesystem           Size  Used Avail Use% Mounted on
/dev/mapper/VolGroup00-LogVol00
                      12G  6.0G  4.6G  57% / /dev/sda1
  99M  9.1M   85M  10% /boot
none    316M     0  316M   0% /dev/shm
```

マウントされたパーティションの一覧には、/dev/shm のエントリーがあります。このエントリーは、システムの仮想メモリーファイルシステムを表します。

du コマンドは、ディレクトリー内のファイルが使用している推定領域を表示します。シェルプロンプトで du と入力すると、各サブディレクトリーのディスク使用量が一覧に表示されます。現在のディレクトリーおよびサブディレクトリーの合計は、一覧の最後の行として表示されます。すべてのサブディレクトリーの合計を表示したくない場合は、du -hs コマンドを使用して、人間が判読できる形式でディレクトリーの合計のみを表示します。du --help コマンドを使用して、他のオプションを表示します。

グラフィカル形式でシステムのパーティションとディスク領域の使用状況を表示するには、システム > 管理 > システムモニター をクリックして Gnome System Monitor を使用するか、シェルプロンプトで gnome-system-monitor を入力します(XTerm など)。ファイルシステム タブを選択して、システムのパーティションを表示します。以下の図は、ファイルシステム タブを示しています。

図42.3 GNOME システムモニター - ファイルシステム

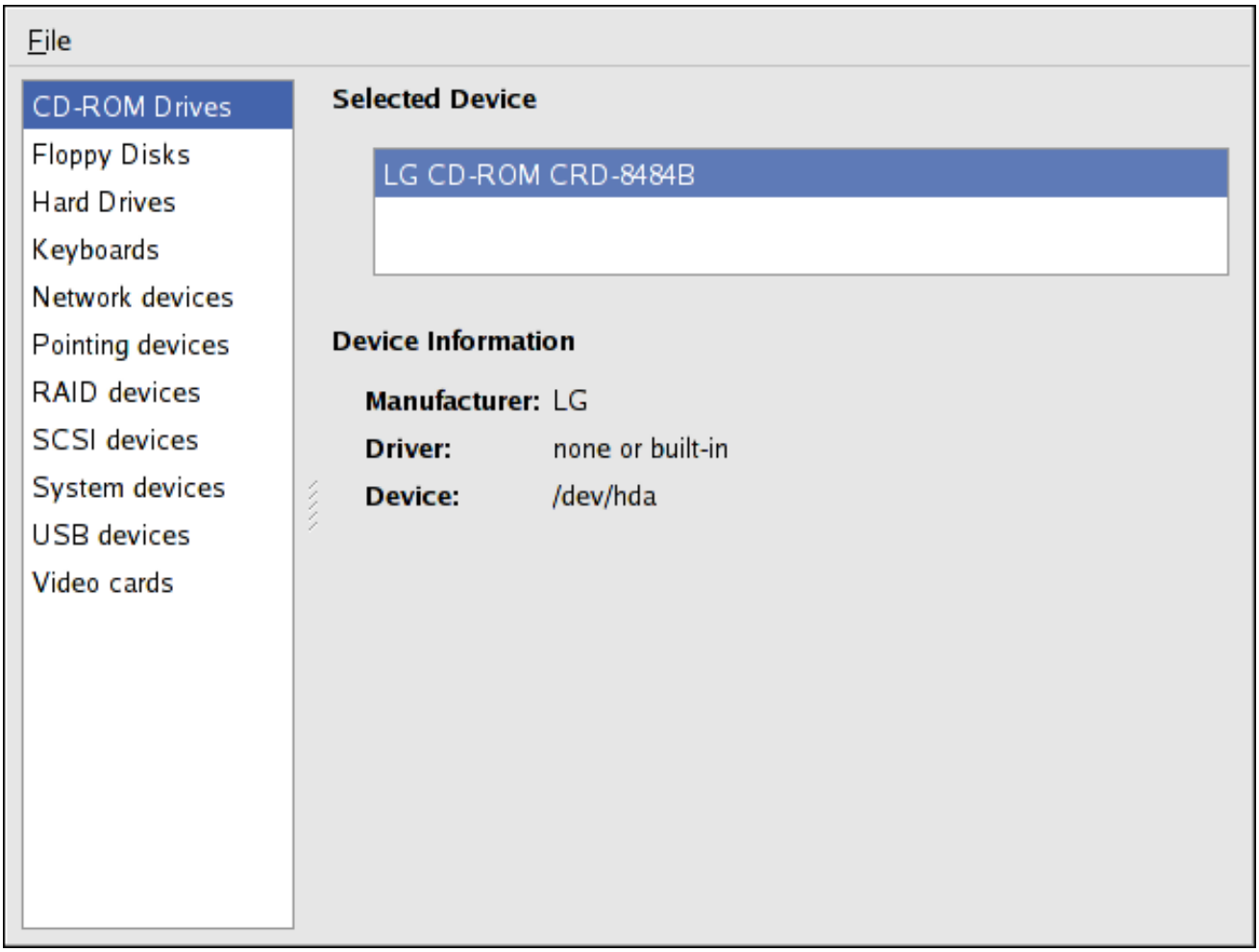


[D]

42.4. ハードウェア

*ハードウェアの設定に問題がある場合や、システムにあるハードウェアを把握する必要がある場合は、ハードウェア ブラウザー アプリケーションを使用してプローブできるハードウェアを表示できます。デスクトップからプログラムを起動するには、**System**（パネルのメインメニュー）> **Administration** > **Hardware** を選択するか、シェルプロンプトで **hwbrowser** と入力します。図 42.4「ハードウェアブラウザー」 に示すように、**CD-ROM** デバイス、ディスケットドライブ、ハードドライブ、パーティション、ネットワークデバイス、システムデバイス、ビデオカードが表示されます。左側のメニューのカテゴリー名をクリックすると、情報が表示されます。*

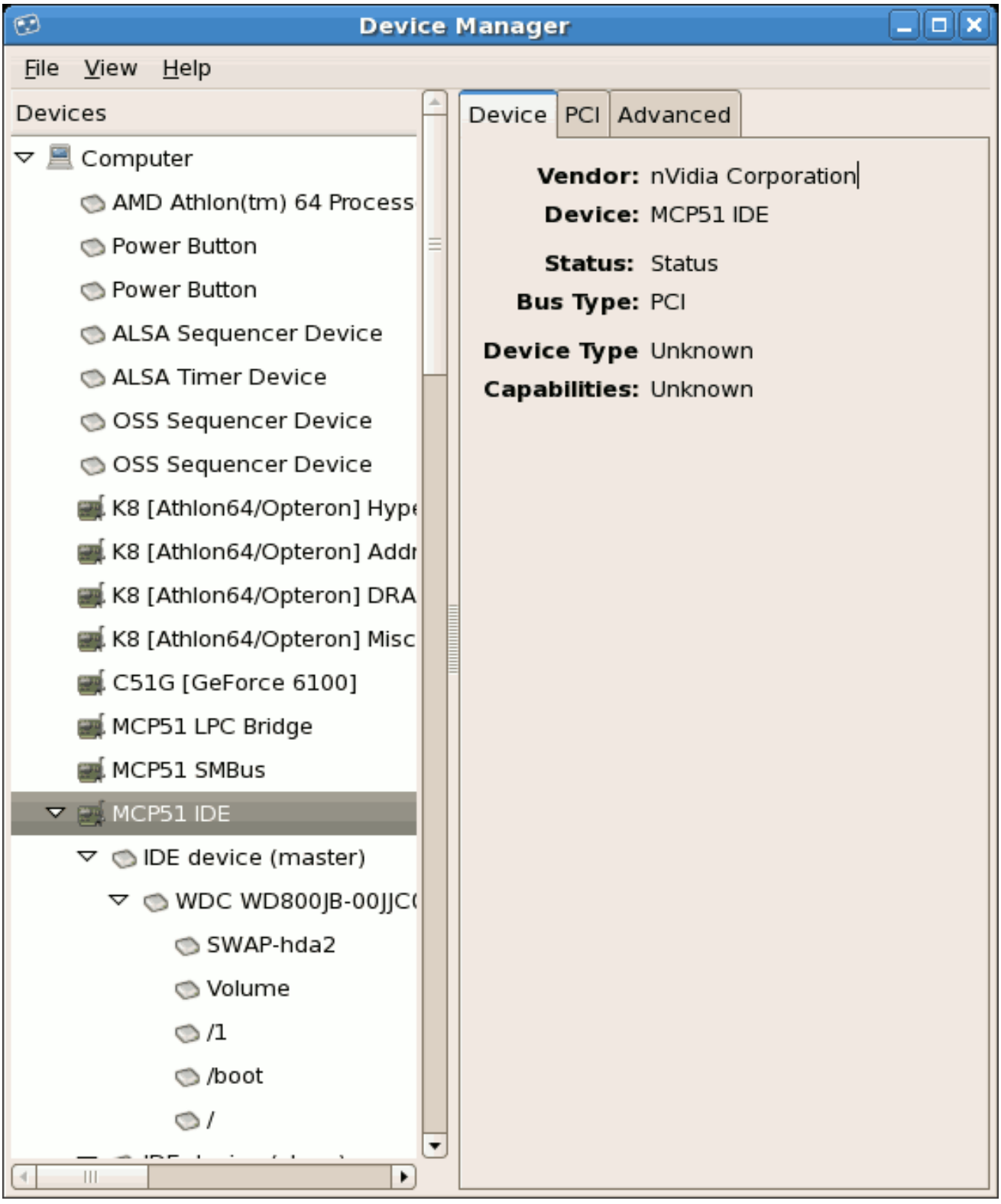
図42.4 ハードウェアブラウザー



[D]

*Device Manager アプリケーションを使用して、システムハードウェアを表示することもできます。このアプリケーションを起動するには、**System**（パネルのメインメニュー）> **Administration** > **Hardware like the Hardware Browser** を選択します。ターミナルからアプリケーションを起動するには、**hal-device-manager** と入力します。インストール設定によっては、上記のグラフィカルメニューがこのアプリケーションを起動するか、またはクリックすると ハードウェアブラウザーを 開始できます。以下の図は、**Device Manager** ウィンドウを示しています。*

図42.5 デバイスマネージャー



[D]

lspci コマンドを使用して、すべての PCI デバイスを一覧表示することもできます。詳細情報については lspci -v コマンドを使用するか、詳細の出力には lspci -vv コマンドを使用します。

たとえば、lspci を使用して、システムのビデオカードの製造元、モデル、およびメモリーサイズを確認できます。

```
00:00.0 Host bridge: ServerWorks CNB20LE Host Bridge (rev 06)
00:00.1 Host bridge: ServerWorks CNB20LE Host Bridge (rev 06)
00:01.0 VGA compatible controller: S3 Inc. Savage 4 (rev 04)
00:02.0 Ethernet controller: Intel Corp. 82557/8/9 [Ethernet Pro 100] (rev 08)
00:0f.0 ISA bridge: ServerWorks OSB4 South Bridge (rev 50)
00:0f.1 IDE interface: ServerWorks OSB4 IDE Controller
00:0f.2 USB Controller: ServerWorks OSB4/CSB5 OHCI USB Controller (rev 04)
01:03.0 SCSI storage controller: Adaptec AIC-7892P U160/m (rev 02)
01:05.0 RAID bus controller: IBM ServeRAID Controller
```

また、lspci は、製造元またはモデル番号が分からない場合に、システムのネットワークカードを判断する際に役立ちます。

42.5. 関連情報

システム情報の収集方法は、以下のリソースを参照してください。

42.5.1. インストールされているドキュメント

- ps --help - ps で使用できるオプションの一覧を表示します。
- トップ man ページ： top とそのオプションの詳細を確認するには、man top と入力します。
- 無料 の man ページ： man free と入力し、多くのオプションを確認できます。
- df man ページ： man df と入力して、df コマンドと、その多くのオプションの詳細を確認します。
- DU man ページ： man du と入力して、du コマンドとそのオプションの詳細を確認します。
- lspci man ページ：lspci コマンドと、その多くのオプションの詳細を確認するには、man lspci と入力します。
- /proc/ ディレクトリー - /proc/ ディレクトリーの内容を使用して、より詳細なシステム情報を収集することもできます。

第43章 OPROFILE

OProfile はオーバーヘッドが低く、システム全体のパフォーマンス監視ツールです。プロセッサー上のパフォーマンス監視ハードウェアを使用して、メモリーの参照時、L2 キャッシュ要求の数、受信したハードウェア割り込みの回数など、システム上のカーネルおよび実行ファイルに関する情報を取得します。Red Hat Enterprise Linux システムでは、このツールを使用するには oprofile RPM パッケージをインストールする必要があります。

多くのプロセッサーには、専用のパフォーマンス監視ハードウェアが含まれます。このハードウェアを使用すると、特定のイベントが発生したタイミング（要求されたデータがキャッシュにないなど）を検出できます。ハードウェアは通常、イベントが発生するたびにインクリメントされる 1 つ以上の カウンター の形式を取ります。カウンター値が基本的にロールオーバーすると割り込みが生成され、パフォーマンス監視が生成する詳細（つまりオーバーヘッド）を制御できます。

OProfile はこのハードウェア（またはパフォーマンス監視ハードウェアが存在しない場合のタイマーベースの置換）を使用して、カウンターが割り込みを生成するたびにパフォーマンス関連のデータの サンプル を収集します。これらのサンプルは定期的にディスクに書き込まれます。後で、これらのサンプルに含まれるデータを使用して、システムレベルとアプリケーションレベルのパフォーマンスのレポートを生成できます。

OProfile は便利なツールですが、使用時にはいくつかの制限に注意してください。

- 共有ライブラリー の使用：共有ライブラリー 内のコード用のサンプルは、--separate=library オプションが使用されない限り、特定のアプリケーションには含まれません。
- パフォーマンスモニターリングのサンプル：パフォーマンス監視レジスターがサンプルをトリガーすると、割り込み処理はゼロ例外で除算さ れるのと正確ではありません。プロセッサーによる命令の順序外実行により、サンプルは近くの命令に記録される場合があります。
- opreport はインライン関数のサンプルを適切に関連付けません。opreport は単純なアドレス範囲メカニズムを使用してアドレスがどの機能にあるかを判断します。インライン関数サンプルは inline 関数には属性されず、inline 関数が挿入された関数には含まれません。
- OProfile は複数の実行からのデータの蓄積 を行います。OProfile はシステム全体のプロファイラーで、プロセスが複数回起動およびシャットダウンすることを期待します。そのため、複数の実行からのサンプルは累積されます。コマンド opcontrol --reset を使用して、以前の実行からのサンプルを消去します。
-

CPU の制限のないパフォーマンスの問題: *OProfile は、CPU の制限プロセスの問題を検索することを目的としています。OProfile は、ロックを待機しているため、または他のイベントが発生する(I/O デバイスが操作を完了するなど)、sleep のプロセスを特定しません。*

43.1. ツールの概要

表43.1「OProfile コマンド」 では、oprofile パッケージで提供されるツールの概要を説明します。

表43.1 OProfile コマンド

コマンド	説明
ophelp	システムのプロセッサーで利用可能なイベントと、それぞれの簡単な説明を表示します。
opimport	サンプルデータベースファイルをシステム用に外部のバイナリー形式からネイティブの形式に変換します。異なるアーキテクチャーからのサンプルデータベースを解析する場合にのみこのオプションを使用してください。
opannotate	アプリケーションがデバッグシンボルでコンパイルされている場合は、実行可能ファイルのアノテーション付きのソースを作成します。詳細は、「opannotateの使用」 を参照してください。
opcontrol	収集されるデータを設定します。詳細は、「OProfile の設定」 を参照してください。
opreport	プロファイリングデータを取得します。詳細は、「opreportの使用」 を参照してください。

コマンド	説明
oprofiled	デーモンとして実行して定期的にサンプルデータをディスクに書き込みます。

43.2. OPROFILE の設定

OProfile を実行する前に、これを設定する必要があります。少なくとも、カーネルの監視（またはカーネルを監視しない選択）を選択することが必要です。以下のセクションでは、opcontrol ユーティリティーを使用して OProfile を設定する方法を説明します。opcontrol コマンドを実行すると、設定オプションは /root/.oprofile/daemonrc ファイルに保存されます。

43.2.1. カーネルの指定

まず、OProfile がカーネルを監視するかどうかを設定します。これは、OProfile を起動する前に必要な唯一の設定オプションです。その他はすべてオプションです。

カーネルを監視するには、root で以下のコマンドを実行します。

```
opcontrol --setup --vmlinux=/usr/lib/debug/lib/modules/`uname -r`/vmlinux
```

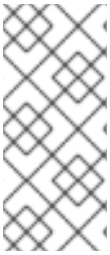
注記

カーネルを監視するには、（圧縮されていないカーネルを含む） debuginfo パッケージがインストールされている必要があります。

OProfile がカーネルを監視しないように設定するには、root で以下のコマンドを実行します。

```
opcontrol --setup --no-vmlinux
```

このコマンドは、oprofile カーネルモジュールもロードされていない場合は読み込み、/dev/oprofile/ ディレクトリーを作成します。このディレクトリーの詳細は、「/dev/oprofile/について」 を参照してください。



注記

OProfile がカーネルのプロファイリング用に設定されていない場合でも、oprofile モジュールがそこからロードされるように、SMP カーネルがまだ実行されている必要があります。

収集したデータの保存方法や場所ではなく、カーネル内でサンプルを収集するかどうかを設定します。カーネルライブラリーとアプリケーションライブラリー用にさまざまなサンプルファイルを生成するには、「カーネルおよびユーザー空間プロファイルの分離」 を参照してください。

43.2.2. イベントのモニターへの設定

ほとんどのプロセッサーには カウンター が含まれます。これは、OProfile が特定のイベントを監視するために使用します。表43.2「OProfile プロセッサーおよびカウンター」 で示されているように、利用可能なカウンターの数はプロセッサーによって異なります。

表43.2 OProfile プロセッサーおよびカウンター

プロセッサー	cpu_type	カウンターの数
Pentium Pro	i386/ppro	2
Pentium II	i386/pii	2
Pentium-24	i386/piii	2
Pentium 4 （非スレッド）	i386/p4	8
Pentium 4 （ハイパースレッディング）	i386/p4-ht	4
Athlon	i386/athlon	4
AMD64	x86-64/hammer	4
Itanium	ia64/itanium	4
Itanium 2	ia64/itanium2	4
TIMER_INT	timer	1
IBM eServer iSeries および pSeries	timer	1

プロセッサー	cpu_type	カウンターの数
	ppc64/power4	8
	ppc64/power5	6
	ppc64/970	8
IBM eServer S/390 および S/390x	timer	1
IBM eServer zSeries	timer	1

表43.2「OProfile プロセッサーおよびカウンター」 を使用して、正しいプロセッサータイプが検出され、同時に監視できるイベントの数を判断します。プロセッサーがパフォーマンス監視ハードウェアに対応していない場合は、タイマー がプロセッサータイプとして使用されます。

timer を使用すると、ハードウェアがハードウェアパフォーマンスカウンターをサポートしないため、イベントをプロセッサーに設定できません。代わりに、タイマー割り込みがプロファイリングに使用されます。

タイマー がプロセッサータイプとして使用されていない場合は、監視されるイベントを変更できます。また、プロセッサーのカウンター 0 はデフォルトで時間ベースのイベントに設定されます。プロセッサーに複数のカウンターが存在する場合は、カウンター 0 以外のカウンターはデフォルトでイベントに設定されていません。監視されるデフォルトイベントは 表43.3「デフォルトのイベント」 に表示されます。

表43.3 デフォルトのイベント

プロセッサー	カウンターのデフォルトイベント	説明
Pentium Pro、Pentium II、Pentium II、Athlon、AMD64	CPU_CLK_UNHALTED	プロセッサーのクロックは停止しません。
Pentium 4 (HT および非 HT)	GLOBAL_POWER_EVENTS	プロセッサーが停止されない時間
Itanium 2	CPU_CYCLES	CPU サイクル
TIMER_INT	（なし）	各タイマー割り込みの例

プロセッサー	カウンターのデフォルトイベント	説明
ppc64/power4	サイクル	プロセッサーサイクル
ppc64/power5	サイクル	プロセッサーサイクル
ppc64/970	サイクル	プロセッサーサイクル

一度に監視できるイベントの数は、プロセッサーのカウンターの数によって決まります。ただし、1 対 1 の相関ではありません。一部のプロセッサーでは、特定のイベントを特定のカウンターにマッピングする必要があります。利用可能なカウンターの数を確認するには、以下のコマンドを実行します。

```
ls -d /dev/oprofile/[0-9]*
```

利用可能なイベントは、プロセッサーのタイプによって異なります。プロファイリングに使用できるイベントを確認するには、root で以下のコマンドを実行します（一覧はシステムのプロセッサータイプに固有のものです）。

```
ophelp
```

各カウンターのイベントは、コマンドラインまたはグラフィカルインターフェイスで設定できます。グラフィカルインターフェイスの詳細は、「グラフィカルインターフェイス」 を参照してください。カウンターを特定のイベントに設定できない場合は、エラーメッセージが表示されます。

コマンドラインで設定可能な各カウンターのイベントを設定するには、opcontrol を使用します。

```
opcontrol --event=<event-name>:<sample-rate>
```

& lt;event-name& gt; を ophelp からの正確なイベント名に置き換え、< sample-rate > をサンプル間のイベント数に置き換えます。

43.2.2.1. サンプリングレート

デフォルトでは、時間ベースのイベントセットが選択されます。プロセッサーごとに 100,000 個のクロックサイクルごとにサンプルを作成します。タイマー割り込みが使用される場合、タイマーは jiffy レートに設定され、ユーザーが設定できません。cpu_type が timer でない場合、各イベントには サンプリングレート を設定できます。サンプリングレートは、各サンプルスナップショット間のイベント数です。

カウンターにイベントを設定する場合は、サンプルレートを指定することもできます。

```
opcontrol --event=<event-name>:<sample-rate>
```

&lt ;sample-rate> を、再度サンプリングするまで待機するイベント数に置き換えます。カウントが小さいほど、サンプルの頻度が高くなります。頻繁に発生しないイベントの場合は、イベントインスタンスを取得するのに少ない数が必要になる場合があります。

注意

サンプリングレートを設定する場合は細心の注意を払ってください。サンプリングが頻繁になりすぎると、システムの負荷が過剰になり、システムがフリーズしているかのように表示されるか、システムが実際にフリーズします。

43.2.2.2. ユニットマスク

ユーザーパフォーマンスの監視イベントによっては、イベントをさらに定義するためにユニットマスクが必要になる場合があります。

各イベントのユニットマスクは、ophelp コマンドで一覧表示されています。各ユニットマスクの値は 16 進数形式で一覧表示されます。複数のユニットマスクを指定するには、ビット単位の または 操作を使用して 16 進数の値を組み合わせる必要があります。

```
opcontrol --event=<event-name>:<sample-rate>:<unit-mask>
```

43.2.3. カーネルおよびユーザー空間プロファイルの分離

デフォルトでは、カーネルモードとユーザーモード情報が各イベントについて収集されます。特定のカウンターのカーネルモードのイベントを無視するように OProfile を設定するには、以下のコマンドを実行します。

```
opcontrol --event=<event-name>:<sample-rate>:<unit-mask>:0
```

以下のコマンドを実行して、カウンターのプロファイリングカーネルモードを再度開始します。

```
opcontrol --event=<event-name>:<sample-rate>:<unit-mask>:1
```

特定のカウンターのユーザーモードのイベントを無視するように **OProfile** を設定するには、以下のコマンドを実行します。

```
opcontrol --event=<event-name>:<sample-rate>:<unit-mask>:<kernel>:0
```

以下のコマンドを実行して、カウンターのユーザーモードを再度開始します。

```
opcontrol --event=<event-name>:<sample-rate>:<unit-mask>:<kernel>:1
```

OProfile デーモンがプロファイルデータをサンプルファイルに書き込むと、カーネルおよびライブラリープロファイルデータを個別のサンプルファイルに分割できます。デーモンがサンプルファイルに書き込みする方法を設定するには、**root** で以下のコマンドを実行します。

```
opcontrol --separate=<choice>
```

<choice& gt; は、以下のいずれかになります。

- **none -** プロファイルを分離しません（デフォルト）。
- ライブラリー **-** ライブラリー のアプリケーションごとのプロファイルを生成します
- **kernel -** カーネルおよびカーネルモジュールのアプリケーションごとのプロファイルを生成します。
- **all -** ライブラリーのアプリケーションごとのプロファイルと、カーネルおよびカーネルモジュールのアプリケーションごとのプロファイルを生成します

--separate=library を使用すると、サンプルファイル名には、実行可能ファイルの名前とライブラリーの名前が含まれます。

注記

これらの設定変更は、oprofile を再起動すると有効になります。

43.3. OPROFILE の開始および停止

OProfile を使用したシステムの監視を開始するには、root で以下のコマンドを実行します。

```
opcontrol --start
```

以下のような出力が表示されます。

```
Using log file /var/lib/oprofile/oprofiled.log Daemon started. Profiler running.
```

/root/.oprofile/daemonrc の設定が使用されます。

OProfile デーモン oprofiled が起動し、サンプルデータを /var/lib/oprofile/samples/ ディレクトリーに定期的に書き込みます。デーモンのログファイルは /var/lib/oprofile/oprofiled.log にあります。

プロファイラーを停止するには、root で以下のコマンドを実行します。

```
opcontrol --shutdown
```

43.4. データの保存

サンプルを特定の時間に保存すると便利な場合があります。たとえば、実行可能ファイルをプロファイリングする場合、さまざまな入力データセットに基づいて異なるサンプルを収集すると便利です。監視されるイベントの数がプロセッサーで利用可能なカウンターの数を超えると、OProfile の複数の実行を使用してデータを収集し、毎回サンプルデータを異なるファイルに保存します。

サンプルファイルの現在のセットを保存するには、< name > を現行セッションで一意の説明的な名前に置き換えて以下のコマンドを実行します。

```
opcontrol --save=<name>
```

ディレクトリー /var/lib/oprofile/samples/ が作成され、現在のサンプルファイルがコピーされます。

43.5. データの分析

OProfile デーモン oprofiled は、定期的にサンプルを収集し、それらを /var/lib/oprofile/samples/ ディレクトリーに書き込みます。データを読み取る前に、root で以下のコマンドを実行して、すべてのデータがこのディレクトリーに書き込まれることを確認します。

```
opcontrol --dump
```

各サンプルファイル名は、実行可能ファイルの名前に基づいています。たとえば、/bin/bash の Pentium-24 プロセッサーにおけるデフォルトイベントのサンプルは以下のようになります。

```
\{root\}/bin/bash/\{dep\}/\{root\}/bin/bash/CPU_CLK_UNHALTED.100000
```

以下のツールは、サンプルデータが収集されるとプロファイルできます。

- opreport
- opannotate

これらのツール、およびバイナリープロファイルを使用して、さらに分析できるレポートを生成します。

WARNING

プロファイル化される実行ファイルは、データを分析するためにこれらのツールと共に使用する必要があります。データの収集後に変更する必要がある場合は、サンプルの作成に使用される実行ファイルとサンプルファイルをバックアップします。サンプルファイルとバイナリーは合意する必要がある点に注意してください。バックアップを作成しないと、バックアップは機能しません。oparchive を使用すると、この問題に対処できます。

各実行可能ファイルのサンプルは、1 つのサンプルファイルに書き込まれます。動的にリンクされた各ライブラリーからのサンプルも、単一のサンプルファイルに書き込まれます。OProfile の実行中に、

監視対象の実行ファイルに変更があり、実行ファイルのサンプルファイルが存在する場合は、既存のサンプルファイルが自動的に削除されます。したがって、既存のサンプルファイルが必要な場合は、実行ファイルを新しいバージョンに置き換える前に、作成した実行ファイルと共にバックアップする必要があります。oprofile 分析ツールは、分析中にサンプルを作成した実行可能ファイルを使用します。実行ファイルが変更されると、分析ツールは関連するサンプルを分析できません。サンプルファイルをバックアップする方法は、「データの保存」 を参照してください。

43.5.1. opreportの使用

opreport ツールは、プロファイリングされるすべての実行ファイルの概要を提供します。

以下は、サンプル出力の一部です。

```
Profiling through timer interrupt
TIMER:0|
samples|      %|
------------------
25926 97.5212 no-vmlinux
359  1.3504 pi
65  0.2445 Xorg
62  0.2332 libvte.so.4.4.0
56  0.2106 libc-2.3.4.so
34  0.1279 libglib-2.0.so.0.400.7
19  0.0715 libXft.so.2.1.2
17  0.0639 bash
8  0.0301 ld-2.3.4.so
8  0.0301 libgdk-x11-2.0.so.0.400.13
6  0.0226 libgobject-2.0.so.0.400.7
5  0.0188 oprofiled
4  0.0150 libpthread-2.3.4.so
4  0.0150 libgtk-x11-2.0.so.0.400.13
3  0.0113 libXrender.so.1.2.2
3  0.0113 du
1  0.0038 libcrypto.so.0.9.7a
1  0.0038 libpam.so.0.77
1  0.0038 libtermcap.so.2.0.8
1  0.0038 libX11.so.6.2
1  0.0038 libgthread-2.0.so.0.400.7
1  0.0038 libwnck-1.so.4.9.0
```

各実行可能ファイルは、それぞれの行に一覧表示されます。最初の列は、実行ファイルに対して記録されたサンプル数です。2 番目のコラムは、サンプルの合計数に対するサンプルの割合です。3 列目は、実行ファイルの名前です。

利用可能なコマンドラインオプションのリストについては opreport の man ページを参照してください。たとえば、サンプルの数が最も少ない実行可能ファイルから、サンプル数が最も多いものに並べ替えるために使用される -r オプションなどです。

43.5.2. 単一実行可能ファイルでの opreport の使用

特定の実行可能ファイルに関する詳細なプロファイル情報を取得するには、opreport を使用します。

```
opreport <mode> <executable>
```

分析するには、実行可能ファイルへの完全パスである <executable> 必要があります。<mode> は以下のいずれかである必要があります。

-l

シンボルでサンプルデータを一覧表示します。たとえば、以下は、コマンド opreport -l /lib/tls/libc- <version > .so の実行からの出力の一部になります。

```
samples  %        symbol name
12       21.4286  __gconv_transform_utf8_internal
5        8.9286   _int_malloc
4        7.1429   malloc
3        5.3571   __i686.get_pc_thunk.bx
3        5.3571   _dl_mcount_wrapper_check
3        5.3571   mbrtowc
3        5.3571   memcpy
2        3.5714   _int_realloc
2        3.5714   _nl_intern_locale_data
2        3.5714   free
2        3.5714   strcmp
1        1.7857   __ctype_get_mb_cur_max
1        1.7857   __unregister_atfork
1        1.7857   __write_nocancel
1        1.7857   _dl_addr
1        1.7857   _int_free
1        1.7857   _itoa_word
1        1.7857   calc_eclosure_iter
1        1.7857   fopen@@GLIBC_2.1
1        1.7857   getpid
1        1.7857   memmove
1        1.7857   msort_with_tmp
1        1.7857   strcpy
1        1.7857   strlen
1        1.7857   vfprintf
1        1.7857   write
```

最初の列はシンボルのサンプル数で、2 番目のコラムは、実行ファイルのサンプル全体に対するこのシンボルのサンプルの割合であり、3 番目のコラムはシンボル名です。

サンプルの最大数から最小（逆引き順序）に出力を並べ替えるには、-l オプションとともに -r

を使用します。

-i <symbol-name>

シンボル名に固有のサンプルデータを一覧表示します。たとえば、以下の出力は、opreport -l -i __gconv_transform_utf8_internal /lib/tls/libc- <version> .so コマンドからの ものです。

```
samples  %        symbol name
12       100.000  __gconv_transform_utf8_internal
```

最初の行は、シンボル/実行可能な組み合わせの概要です。

最初の列は、メモリーシンボルのサンプル数です。2 番目のコラムは、シンボルのサンプルの合計数に対するメモリーアドレスのサンプルの割合です。3 列目はシンボル名です。

-d

-l よりも詳細でシンボルでサンプルデータを一覧表示します。たとえば、以下の出力は、コマンド opreport -l -d __gconv_transform_utf8_internal /lib/tls/libc- <version> .so からの ものです。

```
vma      samples  %        symbol name
00a98640 12       100.000  __gconv_transform_utf8_internal
00a98640 1        8.3333
00a9868c 2        16.6667
00a9869a 1        8.3333
00a986c1 1        8.3333
00a98720 1        8.3333
00a98749 1        8.3333
00a98753 1        8.3333
00a98789 1        8.3333
00a98864 1        8.3333
00a98869 1        8.3333
00a98b08 1        8.3333
```

データは -l オプションと同じですが、シンボルごとに、使用される各仮想メモリーアドレスが表示されます。仮想メモリーアドレスごとに、シンボルのサンプル数に対するサンプル数およびパーセンテージが表示されます。

-x<symbol-name>

出力からシンボルのコンマ区切りリストを除外します。

session:<name>

/var/lib/oprofile/samples/ ディレクトリーに対するセッションまたはディレクトリーへの完全パスを指定します。

43.5.3. モジュールの詳細な出力の取得

OProfile は、マシンで実行しているカーネルおよびユーザー空間コードに関するシステム全体のデータを収集します。ただし、モジュールがカーネルに読み込まれると、カーネルモジュールの起点に関する情報が失われます。モジュールは、起動時に initrd ファイル、さまざまなカーネルモジュールを持つ ディレクトリー、またはローカルに作成されたカーネルモジュールから取得できます。その結果、OProfile がモジュールのサンプルを記録すると、root ディレクトリー内の実行ファイルのモジュールのサンプルが一覧表示されますが、モジュールの実際のコードが含まれる場所である訳ではありません。分析ツールが実行ファイルを取得できるようにするには、いくつかの手順を実行する必要があります。

たとえば、AMD64 マシンでは、サンプリングは Data cache access および Data cache misses を記録するよう設定され、ext3 モジュールのデータを表示することを前提とします。

```
~]$ opreport /ext3
CPU: AMD64 processors, speed 797.948 MHz (estimated)
Counted DATA_CACHE_ACCESSES events (Data cache accesses) with a unit mask of 0x00
(No unit mask) count 500000
Counted DATA_CACHE_MISSES events (Data cache misses) with a unit mask of 0x00 (No unit
mask) count 500000
DATA_CACHE_ACC...|DATA_CACHE_MIS...|
samples|      %|  samples|      %|
------------------------------------
148721 100.000      1493 100.000 ext3
```

モジュールのアクションの詳細なビューを取得するには、モジュールをトリップ解除（例：カスタムビルドからインストール）するか、カーネルに debuginfo RPM をインストールする必要があります。

実行しているカーネル "uname -a" を見つけ、適切な debuginfo rpm を取得し、マシンにインストールします。

次に、oprofile が正しい場所でモジュールのコードを見つけるようにシンボリックリンクを作成します。

```
~]# ln -s /lib/modules/`uname -r`/kernel/fs/ext3/ext3.ko /ext3
```

次に、以下の方法で詳細情報を取得できます。

```
~]# opreport image:/ext3 -l|more
warning: could not check that the binary file /ext3 has not been modified since the profile was taken. Results may be inaccurate.
CPU: AMD64 processors, speed 797.948 MHz (estimated)
Counted DATA_CACHE_ACCESSES events (Data cache accesses) with a unit mask of 0x00 (No unit mask) count 500000
Counted DATA_CACHE_MISSES events (Data cache misses) with a unit mask of 0x00 (No unit mask) count 500000
samples  %        samples  %        symbol name
16728    11.2479  7         0.4689  ext3_group_sparse
16454    11.0637  4         0.2679  ext3_count_free_blocks
14583     9.8056  51        3.4159  ext3_fill_super
8281      5.5681  129       8.6403  ext3_ioctl
7810      5.2514  62        4.1527  ext3_write_info
7286      4.8991  67        4.4876  ext3_ordered_writepage
6509      4.3767  130       8.7073  ext3_new_inode
6378      4.2886  156      10.4488  ext3_new_block
5932      3.9887  87        5.8272  ext3_xattr_block_list
...
```

43.5.4. opannotateの使用

opannotate ツールは、特定の命令のサンプルとソースコードの対応する行との一致を試みます。生成されるファイルには、左側の行のサンプルがあります。また、各関数の最初のコメントにも、関数の合計数のサンプルが一覧表示されます。

このユーティリティーが機能するには、実行可能ファイルを GCC の -g オプションでコンパイルする必要があります。デフォルトでは、Red Hat Enterprise Linux パッケージはこのオプションでコンパイルされていません。

opannotate の一般的な構文は、以下のとおりです。

```
opannotate --search-dirs <src-dir> --source <executable>
```

ソースコードと、分析する実行ファイルを含むディレクトリーを指定する必要があります。追加のコマンドラインオプションの一覧は、opannotate の man ページを参照してください。

43.6. /DEV/OPROFILE/について

/dev/oprofile/ ディレクトリーには、OProfile のファイルシステムが含まれます。cat コマンドを使用して、このファイルシステムの仮想ファイルの値を表示します。たとえば、以下のコマンドは検出されたプロセッサー OProfile のタイプを表示します。

```
cat /dev/oprofile/cpu_type
```

各カウンターの /dev/oprofile/ にディレクトリーが存在する。たとえば、2 つのカウンターがある場合は、/dev/oprofile/0/ ディレクトリーと dev/oprofile /1/ ディレクトリーが存在 します。

カウンターの各ディレクトリーには以下のファイルが含まれます。

- count: サンプルの間隔。
- enabled - 0 の場合、カウンターはオフになり、サンプルは収集されません。1 の場合は、カウンターがオンになり、サンプルが収集されます。
- event: 監視するイベント。
- kernel - 0 の場合、プロセッサーがカーネルスペースにある場合、このカウンターイベントのサンプルは収集されません。1 の場合、プロセッサーがカーネルスペースにある場合でもサンプルが収集されます。
- unit_mask: カウンターに対して有効なユニットマスクを定義します。
- user: 0 の場合、プロセッサーが ユーザー 空間にある場合、カウンターイベントのサンプルは収集されません。1 の場合、プロセッサーがユーザー空間にある場合でもサンプルが収集されます。

これらのファイルは、cat コマンドで取得できます。以下に例を示します。

```
cat /dev/oprofile/0/count
```

43.7. 使用例

開発者は OProfile を使用してアプリケーションのパフォーマンスを分析することができますが、システム管理者がシステム分析を実行することもできます。以下に例を示します。

- システムで最も多く使用されているアプリケーションやサービスの特定 - opreport を使用して、アプリケーションやサービスが使用するプロセッサー時間を判断できます。システムが複数のサービスに使用されるが、実行中である場合、最も多くのプロセッサー時間を消費するサービスは専用システムに移動できます。

- プロセッサーの使用状況を決定 する - CPU_CLK_UNHALTED イベントを監視して、特定の期間におけるプロセッサー負荷を判断できます。その後、このデータを使用して、追加のプロセッサーまたは高速なプロセッサーがシステムパフォーマンスを向上させる可能性があるかどうかを判断できます。

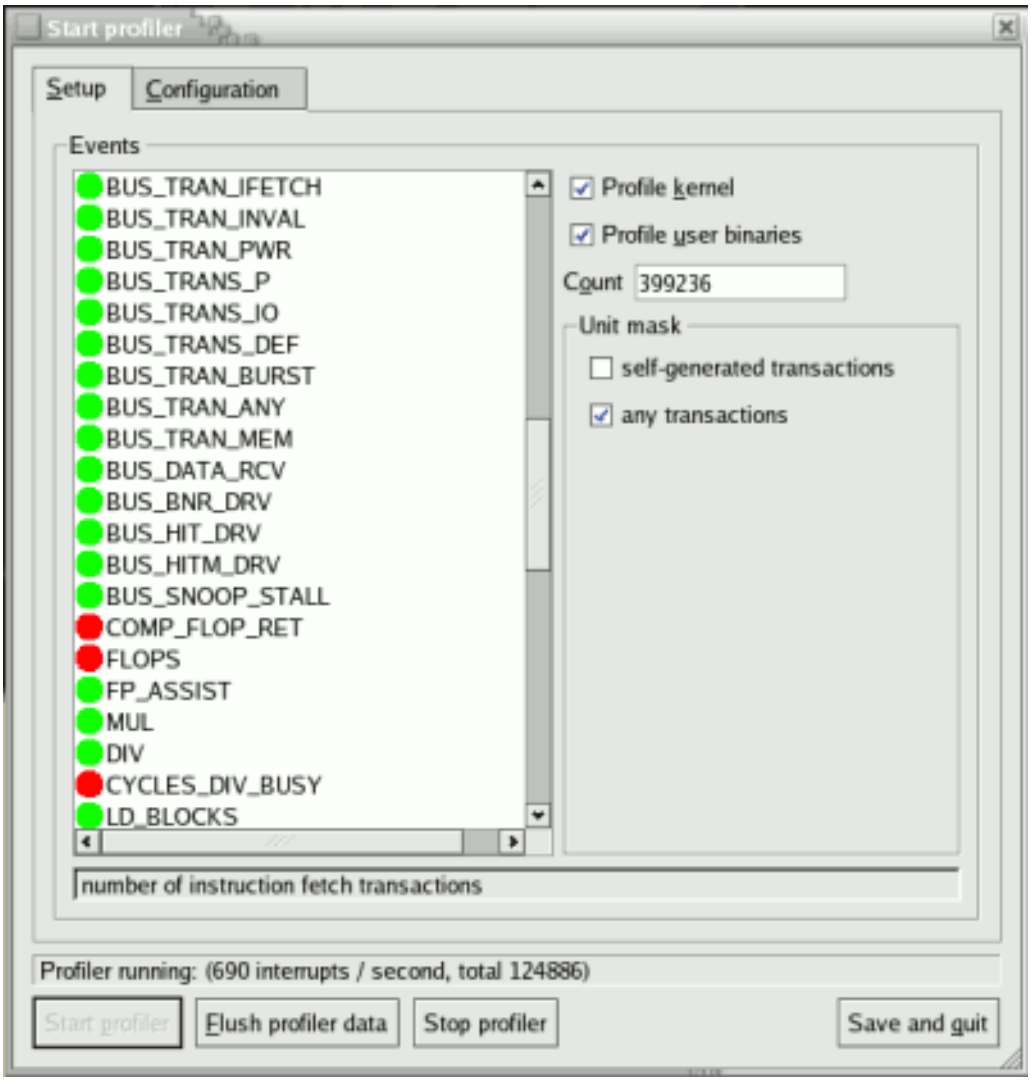
43.8. グラフィカルインターフェイス

一部の OProfile 設定は、グラフィカルインターフェイスで設定できます。これを起動するには、シェルプロンプトで root として oprof_start コマンドを実行します。グラフィカルインターフェイスを使用するには、oprofile-gui パッケージをインストールする必要があります。

オプションを変更したら、Save and quit ボタンをクリックして保存します。設定は /root/.oprofile/daemonrc に書き込まれ、アプリケーションは終了します。アプリケーションを終了しても、OProfile のサンプリングは停止しません。

Setup タブで、「イベントのモニターへの設定」 で説明されているようにプロセッサーカウンターにイベントを設定するには、プルダウンメニューからカウンターを選択し、一覧からイベントを選択します。イベントの簡単な説明が、リストの下にあるテキストボックスに表示されます。特定のカウンターと特定のアーキテクチャーで利用可能なイベントのみが表示されます。このインターフェイスは、プロファイラーが実行されているかどうか、およびプロファイラーに関する簡単な統計も表示します。

図43.1 OProfile の設定



[D]

このタブの右側で、「カーネルおよびユーザー空間プロファイルの分離」で説明されているように、Profile kernel オプションを選択し、現在選択されているイベントのカーネルモードでイベントをカウントします。このオプションが選択されていない場合、カーネルのサンプルは収集されません。

「カーネルおよびユーザー空間プロファイルの分離」で説明されている ように、Profile user binary オプションを選択して、現在選択されているイベントのユーザーモードでイベントをカウントします。このオプションが選択されていない場合、ユーザーアプリケーションのサンプルは収集されません。

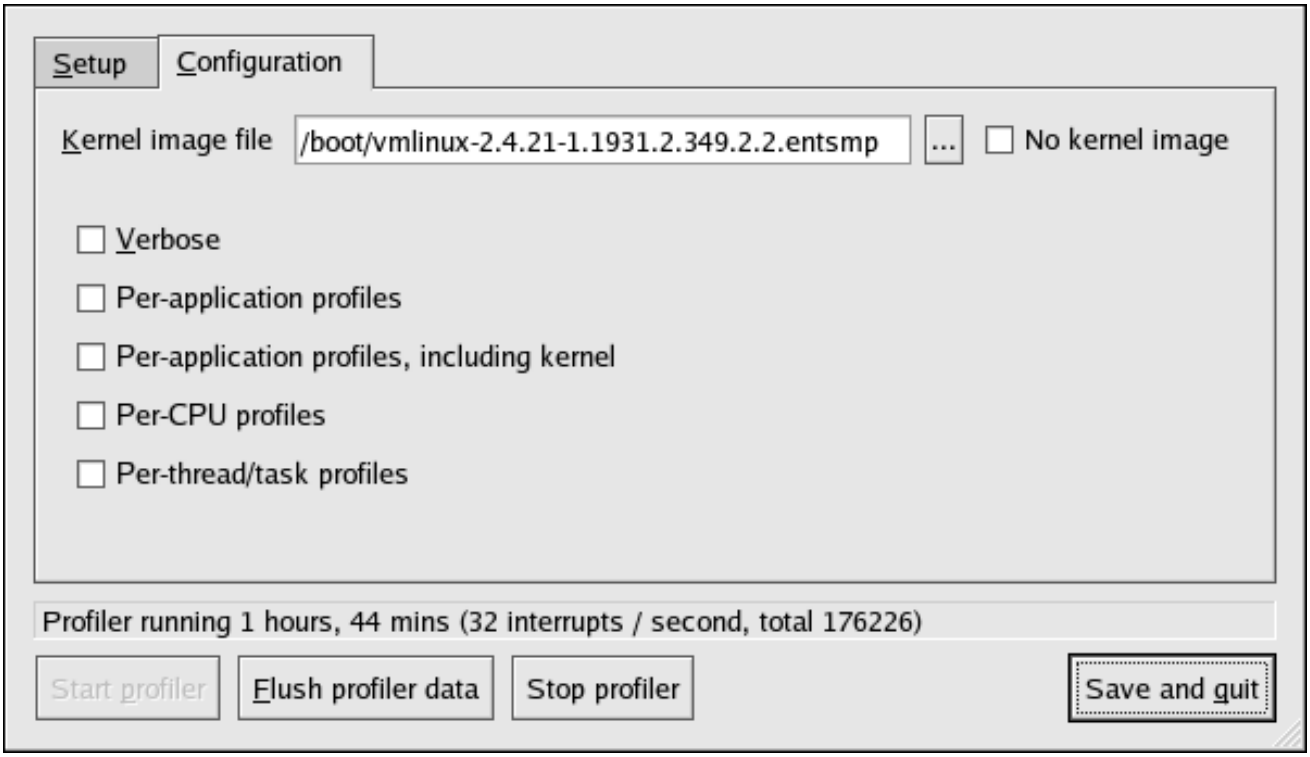
Count テキストフィールドを使用して、「サンプリングレート」で説明されているように、現在選択されているイベントのサンプリングレートを設定します。

「ユニットマスク」で説明されているように、現在選択されているイベントにユニットマスクが利用可能な場合は、セットアップ タブの右側の Unit Masks エリアに表示されます。ユニットマスクの横

にあるチェックボックスを選択して、イベントに対して有効にします。

Configuration タブで、カーネルのプロファイルを設定するには、カーネル イメージ ファイル のテキストフィールドに、監視するカーネルの vmlinux ファイルの名前と場所を入力します。OProfile がカーネルを監視しないように設定するには、No kernel image を選択します。

図43.2 OProfile の設定



[D]

Verbose オプションが選択されている場合、oprofiled デーモンログに詳細情報が含まれます。

アプリケーションごとのカーネルサンプルファイル を選択すると、「カーネルおよびユーザー空間プロファイルの分離」 で説明されているように、OProfile はカーネルおよびカーネルモジュールのアプリケーションごとのプロファイルを生成します。これは、opcontrol --separate=kernel コマンドと同等です。アプリケーション別の共有 libs サンプルファイル を選択すると、OProfile はライブラリーのアプリケーションごとのプロファイルを生成します。これは opcontrol --separate=library コマンドと同等です。

「データの分析」 で説明されているように、データを強制的にサンプルファイルに書き込むには、Flush プロファイラーデータ ボタンをクリックします。これは opcontrol --dump コマンドと同じです。

グラフィカルインターフェイスから OProfile を起動するには、Start profiler をクリックします。プロファイラーを停止するには、Stop profiler をクリックします。アプリケーションを終了しても、

OProfile のサンプリングは停止しません。

43.9. 関連情報

本章では、OProfile と、その設定および使用方法を説明します。詳細は、以下のリソースを参照してください。

43.9.1. インストールされている Docs

- /usr/share/doc/oprofile-<version>/oprofile.html — 『OProfile Manual』

- OProfile の man ページ： opcontrol、opreport、opannotate、および ophelpについて説明します。

43.9.2. 便利な Web サイト

- http://oprofile.sourceforge.net/ - 最新のドキュメント、メーリングリスト、IRC チャンネルなどが含まれています。

パート VI. カーネルおよびドライバーの設定

システム管理者は、カーネルについて確認し、カスタマイズすることができます。Red Hat Enterprise Linux には、管理者がカスタマイズを支援するカーネルツールが含まれています。

第44章 カーネルの手動アップグレード

Red Hat Enterprise Linux カーネルは、サポート対象のハードウェアとの整合性と互換性を確保するために、Red Hat Enterprise Linux カーネルチームがカスタムを構築します。リリースする前に、カーネルは Red Hat が定める厳格な品質保証テストセットをパスしなければなりません。

Red Hat Enterprise Linux カーネルは RPM 形式でパッケージ化されるため、Package Management Tool または yum コマンドを使用してアップグレードおよび検証が容易になります。Package Management Tool は、Red Hat Enterprise Linux サーバーに自動的にクエリーを実行し、カーネルを含むマシンで更新する必要のあるパッケージを決定します。本章では、yum コマンドを使用せずに、カーネルパッケージの手動更新を必要とする個人に のみ 役立ちます。

WARNING

カスタムカーネルの構築は、Red Hat Global Services Support チームがサポートしていないため、このマニュアルでは説明しません。

ヒント

Red Hat では、アップグレードされたカーネルをインストールするために、yum の使用を 強く 推奨しています。

Red Hat Network、Package Management Tool、および yum の詳細は、15章システムの登録およびサブスクリプション管理 を参照してください。

44.1. カーネルパッケージの概要

Red Hat Enterprise Linux には、以下のカーネルパッケージが含まれています（アーキテクチャーには適用されない場合もあります）。

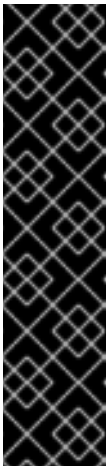
- kernel: マルチプロセッサーシステムのカーネルが含まれます。x86 システムの場合は、最初の 4GB の RAM のみが使用されます。そのため、RAM が 4GB を超える x86 システムは、kernel-PAE を使用する必要があります。

- kernel-devel - kernel パッケージに対してモジュールを構築するのに十分な カーネル ヘッ

ダーと makefiles が含まれます。

- *kernel-PAE (i686 システム専用)- このパッケージは、(カーネル パッケージに対してすでに有効になっているオプションに加えて)以下の主要な設定オプションを提供します。*

  - *RAM が 4GB を超えるシステムに対する PAE （物理アドレス拡張）のサポート、および最大 16GB までの信頼性が高いシステムに対する PAE (Physical Address Extension) のサポート。*



重要な影響

物理アドレス拡張により、x86 プロセッサーは最大 64GB の物理 RAM に対応しますが、Red Hat Enterprise Linux 4 と 5 カーネルの違いにより、Red Hat Enterprise Linux 4 (kernel-hugemem パッケージを含む)のみが 64GB のメモリーをすべて確実に対処できます。また、Red Hat Enterprise Linux 5 の PAE バリアントでは、Red Hat Enterprise Linux 4 kernel-hugemem バリアントなど、プロセスごとに 4GB のメモリーは許可されません。ただし、x86_64 カーネルはこれらの制限を受けず、大規模なメモリーシステムで使用する推奨 Red Hat Enterprise Linux 5 アーキテクチャーです。

- *kernel-PAE -devel: kernel-PAE パッケージに対してモジュールを構築するのに必要なカーネルヘッダーと makefiles が含まれます。*

- *kernel-doc: カーネルソースからのドキュメントファイルが含まれます。これらのファイルには、同梱で配布される Linux カーネルとデバイスドライバーのさまざまな部分が文書化されています。このパッケージをインストールすると、オプションへの参照が提供され、読み込み時に Linux カーネルモジュールに渡すことができます。*

  デフォルトでは、これらのファイルは /usr/share/doc/kernel-doc- <version> / ディレクトリーに配置されます。

- *kernel-headers: Linux カーネルとユーザー空間ライブラリーおよびプログラムとの間のインターフェイスを指定する C ヘッダーファイルが含まれます。ヘッダーファイルは、ほとんどの標準プログラムを構築するのに必要な構造と定数を定義します。*

- *kernel-xen: 仮想化の実行に必要な Linux カーネルのバージョンが含まれます。*

-

kernel-xen-devel - kernel-xen パッケージに対してモジュールを構築するのに必要なカーネルヘッダーと makefiles が含まれます。

注記

*kernel-source パッケージが削除され、Red Hat Network からのみ取得できる RPM に置き換えられました。その後、この *.src.rpm パッケージは、rpmbuild コマンドを使用してローカルに再構築する必要があります。カーネルソースパッケージの取得およびインストールの詳細については、最新の更新リリースノート（すべての更新を含む）を参照してください。 http://www.redhat.com/docs/manuals/enterprise/*

44.2. アップグレードの準備

カーネルをアップグレードする前に、予防的な前準備手順の実行をお薦めします。最初のステップでは、問題が発生した場合にシステムに起動メディアが存在することを確認します。ブートローダーが新しいカーネルを起動するように適切に設定されていない場合、システムは起動用メディアがなくても Red Hat Enterprise Linux で起動できません。

ブートディスクを作成するには、root でログインし、シェルプロンプトでコマンド /sbin/mkbootdisk 'uname -r' を実行します。

ヒント

その他のオプションについては、mkbootdisk の man ページを参照してください。システム BIOS もサポートしていれば、CD-R、CD-RW、および USB フラッシュドライブを介して起動可能なメディアを作成できます。

ブートメディアでマシンを再起動し、それが機能することを確認してから続行します。

インストールされているカーネルパッケージを確認するには、シェルプロンプトで rpm -qa | grep kernel コマンドを実行します。

出力には、システムのアーキテクチャーに応じて、以下のパッケージの一部またはすべてが含まれます（バージョン番号とパッケージは異なる場合があります）。

```
kernel-2.6.9-5.EL
kernel-devel-2.6.9-5.EL
kernel-utils-2.6.9-5.EL
kernel-doc-2.6.9-5.EL
```

```
kernel-smp-2.6.9-5.EL
kernel-smp-devel-2.6.9-5.EL
kernel-hugemem-devel-2.6.9-5.EL
```

出力から、カーネルアップグレードにダウンロードする必要があるパッケージを判断します。1 つのプロセッサーシステムでは、必要なパッケージは kernel パッケージのみです。異なるパッケージの説明は、「カーネルパッケージの概要」 を参照してください。

ファイル名で、各カーネルパッケージには、パッケージが構築されたアーキテクチャーが含まれます。形式は kernel- <variant> - <version > . <arch > .rpm です。ここで、< variant > は PAE、xen などのいずれかになります。&lt ;arch> は以下のいずれかになります。

- AMD64 および Intel EM64T アーキテクチャー用 x86_ 64
- ia64 (Intel® Itanium™ アーキテクチャーの場合)
- IBM® eServer™ pSeries™ アーキテクチャーの場合は ppc64
- IBM® S/390® アーキテクチャーの s390
- IBM® eServer™ System z® アーキテクチャーの s390x
- i686 (Intel® Pentium® II、Intel® Pentium® II、Intel® Pentium® 4、AMD Athlon®、および AMD Duron® システム用)

44.3. アップグレードしたカーネルのダウンロード

システム用に更新されたカーネルが利用可能かを判定する手段は数種類あります。

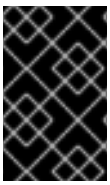
- セキュリティーエラータ：セキュリティーの問題を修正するカーネルアップグレードを含むセキュリティーエラータに http://www.redhat.com/security/updates/ ついての詳細は、を参照してください。
- Red Hat Network 経由 - カーネル RPM パッケージをダウンロードしてインストールします。Red Hat Network は、最新のカーネルをダウンロードし、システムでカーネルをアップグ

レードして、必要に応じて初期 RAM ディスクイメージを作成し、ブートローダーが新しいカーネルをブートするように設定できます。詳細は、http://www.redhat.com/docs/manuals/RHNetwork/ を参照してください。

更新されたカーネルをダウンロードしてインストールするために Red Hat Network を使用している場合は、「初期 RAM ディスクイメージの確認」 および 「ブートローダーの確認」 の手順に従ってください。デフォルトでは、ブートするカーネルは変更し ません。Red Hat Network は、デフォルトのカーネルを自動的に最新バージョンに更新します。カーネルを手動でインストールするには、「アップグレードの実行」 に進みます。

44.4. アップグレードの実行

必要なパッケージをすべて取り込んだ後は、既存カーネルをアップグレードします。



重要な影響

新しいカーネルに問題がある場合を考え、古いカーネルの維持を強く推奨します。

シェルプロンプトで、カーネル RPM パッケージを格納しているディレクトリーに移動します。rpm コマンドで -i 引数を使用して、古いカーネルを保持します。-U オプションは、現在インストールされているカーネルを上書きするため、使用し ない でください。これにより、ブートローダーの問題が生じます。以下に例を示します。

```
rpm -ivh kernel-<kernel version>.<arch>.rpm
```

次の手順では、初期 RAM ディスクイメージが作成されたことを確認します。詳細は、「初期 RAM ディスクイメージの確認」 を参照してください。

44.5. 初期 RAM ディスクイメージの確認

システムが ext3 ファイルシステム、SCSI コントローラー、またはラベルを使用して /etc/fstab 内のパーティションを参照する場合は、初期 RAM ディスクが必要です。初期 RAM ディスクにより、モジュラーカーネルは、通常モジュールが存在するデバイスにアクセスする前に起動する必要がある可能性のあるモジュールにアクセスできます。

IBM eServer iSeries 以外のアーキテクチャーでは、mkinitrd コマンドで初期 RAM ディスクを作成できます。ただし、このステップは、カーネルとその関連パッケージが Red Hat が配布する RPM パッケージからインストールまたはアップグレードされると自動的に実行されます。この場合、初期 RAM ディスクを手動で作成する必要があります。初期 RAM ディスクが存在することを確認するには、ls -l

/boot コマンドを使用して initrd- <version > .img ファイルが作成されていることを確認します（バージョンは、インストールしたカーネルのバージョンと一致する必要があります）。

iSeries システムでは、初期 RAM ディスクファイルと vmlinux ファイルが 1 つのファイルに統合されます。このファイルは addRamDisk コマンドで作成されます。この手順は、カーネルとその関連パッケージが Red Hat, Inc. によって配布される RPM パッケージからインストールまたはアップグレードされると自動的に実行されるため、手動で実行する必要はありません。作成したことを確認するには、ls -l /boot コマンドを使用して、/ boot/vmlinitrd- <kernel-version > ファイルが既に存在することを確認します(< kernel-version > は、インストールしたカーネルのバージョンと一致する必要があります)。

次の手順では、ブートローダーが新しいカーネルをブートするように設定されていることを確認します。詳細は、「ブートローダーの確認」 を参照してください。

44.6. ブートローダーの確認

カーネル RPM パッケージは、新たにインストールしたカーネルを起動するようにブートローダーを設定します(IBM eServer iSeries システムを除く)。ただし、デフォルトで新しいカーネルをブートするようにブートローダーを設定しません。

ブートローダーが正しく設定されていることを確認することをお勧めします。これは重要なステップです。ブートローダーが正しく設定されていない場合、システムは Red Hat Enterprise Linux で正しく起動しません。この場合は、以前に作成したブートメディアでシステムを起動し、ブートローダーを再度設定してみてください。

44.6.1. x86 システム

すべての x86 システム（すべての AMD64 システムを含む）はブートローダーとして GRUB を使用します。

44.6.1.1. GRUB

/boot/grub/grub.conf ファイルに、以前インストールした カーネル パッケージと同じバージョンの title セクションが含まれていることを確認します。

```
# Note that you do not have to rerun grub after making changes to this file
# NOTICE:  You have a /boot partition.  This means that
#          all kernel and initrd paths are relative to /boot/, eg.
#          root (hd0,0)
#          kernel /vmlinuz-version ro root=/dev/hda2
#          initrd /initrd-version.img
#boot=/dev/hda
```

```
default=1 timeout=10
splashimage=(hd0,0)/grub/splash.xpm.gz
title Red Hat Enterprise Linux (2.6.9-5.EL)
        root (hd0,0)
  kernel /vmlinuz-2.6.9-5.EL ro root=LABEL=/
  initrd /initrd-2.6.9-5.EL.img
title Red Hat Enterprise Linux (2.6.9-1.906_EL)
        root (hd0,0)
  kernel /vmlinuz-2.6.9-1.906_EL ro root=LABEL=/
  initrd /initrd-2.6.9-1.906_EL.img
```

別の /boot/ パーティションが作成された場合、カーネルと initrd イメージへのパスは /boot/ と相対的になります。

デフォルトが新しいカーネルに設定されていないことに注意してください。デフォルトで新しいカーネルを起動するように GRUB を設定するには、デフォルト 変数の値を、新しいカーネルを含む title セクションの title セクション番号に変更します。数は 0 で始まります。たとえば、新しいカーネルが最初の title セクションである場合は、default を 0 に設定します。

コンピューターを再起動してメッセージを監視し、ハードウェアが正しく検出されていることを確認することで、新しいカーネルのテストを開始します。

44.6.2. Itanium システム

Itanium システムは、設定ファイルとして /boot/efi/EFI/redhat/elilo.conf を使用する ELILO を使用します。このファイルには、インストールした カーネル パッケージと同じバージョンの イメージ セクションが含まれていることを確認します。

```
prompt timeout=50 default=old  image=vmlinuz-2.6.9-5.EL
        label=linux
  initrd=initrd-2.6.9-5.EL.img         read-only
  append="root=LABEL=/" image=vmlinuz-2.6.9-1.906_EL
  label=old
  initrd=initrd-2.6.9-1.906.img         read-only
  append="root=LABEL=/"
```

デフォルトが新しいカーネルに設定されていないことに注意してください。新しいカーネルを起動するように ELILO を設定するには、default 変数の値を、新しいカーネルを含む image セクションの ラベル の値に変更します。

コンピューターを再起動してメッセージを監視し、ハードウェアが正しく検出されていることを確認することで、新しいカーネルのテストを開始します。

44.6.3. IBM S/390 および IBM System z Systems

IBM S/390 および IBM System z システムは、設定ファイルとして /etc/zipl.conf を使用する z/IPL を使用します。以前にインストールした kernel パッケージと同じバージョンの セクションが ファイルに含まれていることを確認します。

```
[defaultboot] default=old target=/boot/
[linux]
        image=/boot/vmlinuz-2.6.9-5.EL
  ramdisk=/boot/initrd-2.6.9-5.EL.img
  parameters="root=LABEL=/"
[old]
        image=/boot/vmlinuz-2.6.9-1.906_EL
  ramdisk=/boot/initrd-2.6.9-1.906_EL.img
  parameters="root=LABEL=/"
```

デフォルトが新しいカーネルに設定されていないことに注意してください。デフォルトで新規カーネルを起動するように z/IPL を設定するには、default 変数の値を新規カーネルを含むセクションの名前に変更します。各セクションの最初の行には、括弧内の名前が含まれます。

設定ファイルを変更したら、root で /sbin/zipl を実行して変更を適用します。

コンピューターを再起動してメッセージを監視し、ハードウェアが正しく検出されていることを確認することで、新しいカーネルのテストを開始します。

44.6.4. IBM eServer iSeries Systems

カーネルをアップグレードすると、/boot/vmlinitrd- <kernel-version > ファイルがインストールされます。ただし、dd コマンドを使用して、新しいカーネルを起動するようにシステムを設定する必要があります。

1. root で cat /proc/iSeries/mf/side コマンドを実行し、デフォルトのサイド(A、B、または C)を決定します。

2. root で次のコマンドを発行します。ここで、< kernel-version > は新しいカーネルのバージョンで、< side > は直前のコマンドのサイドになります。

```
dd if=/boot/vmlinitrd-<kernel-version> of=/proc/iSeries/mf/<side>/vmlinux bs=8k
```

コンピューターを再起動してメッセージを監視し、ハードウェアが正しく検出されていることを確認することで、新しいカーネルのテストを開始します。

44.6.5. IBM eServer pSeries Systems

IBM eServer pSeries システムは、設定ファイルとして /etc/aboot.conf を使用する YABOOT を使用します。以前にインストールした kernel パッケージと同じバージョンの イメージ セクションが ファイルに含まれていることを確認します。

```
boot=/dev/sda1 init-message=Welcome to Red Hat Enterprise Linux! Hit <TAB> for boot
options
partition=2 timeout=30 install=/usr/lib/yaboot/yaboot delay=10 nonvram
image=/vmlinux--2.6.9-5.EL
        label=old
  read-only
  initrd=/initrd--2.6.9-5.EL.img
  append="root=LABEL=/"
image=/vmlinux-2.6.9-5.EL
  label=linux
  read-only
  initrd=/initrd-2.6.9-5.EL.img
  append="root=LABEL=/"
```

デフォルトが新しいカーネルに設定されていないことに注意してください。最初のイメージのカーネルはデフォルトで起動します。デフォルトのカーネルをブートするには、イメージスタンザを一覧の最初のものになるように移動するか、ディレクティブ default を追加して、新しいカーネルを含むイメージスタンザの ラベル に設定します。

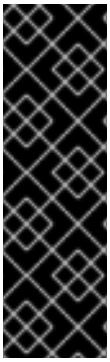
コンピューターを再起動してメッセージを監視し、ハードウェアが正しく検出されていることを確認することで、新しいカーネルのテストを開始します。

第45章 一般的なパラメーターおよびモジュール

本章では、一般的なハードウェアデバイス ドライバーで利用可能な いくつか の可能なパラメーターについて説明します。[9]Red Hat Enterprise Linux では、カーネル モジュール と呼ばれます。ほとんどの場合、デフォルトのパラメーターは機能します。ただし、デバイスが正しく機能するか、デバイスのデフォルトパラメーターを上書きするために追加のモジュールパラメーターが必要になる場合があります。

インストール時に、Red Hat Enterprise Linux は、安定したインストール環境を作成するために、限定されたデバイスドライバーのサブセットを使用します。インストールプログラムは多くの異なるタイプのハードウェアでのインストールをサポートしますが、一部のドライバー(SCSI アダプターおよびネットワークアダプター用など)はインストールカーネルに含まれていないものもあります。代わりに、起動時にユーザーがモジュールとして読み込む必要があります。

インストールが完了すると、カーネルモジュールを介して多数のデバイスのサポートが存在します。



重要な影響

Red Hat は、kernel -smp-unsupported- <kernel-version> および kernel-hugemem-unsupported- <kernel-version > と呼ばれるパッケージのグループに、多数の対応対応デバイスドライバーを提供しています。&lt ;kernel-version> を、システムにインストールされているカーネルのバージョンに置き換えます。これらのパッケージは Red Hat Enterprise Linux インストールプログラムによってインストールされず、提供されるモジュールは Red Hat, Inc ではサポートされません。

45.1. カーネルモジュールユーティリティー

module-init-tools パッケージがインストールされている場合は、カーネルモジュールを管理するコマンドのグループが利用できます。これらのコマンドを使用して、モジュールが正常に読み込まれているかどうか、または新しいハードウェアで異なるモジュールを試行するときに確認します。

コマンド /sbin/lsmod は、現在読み込まれているモジュールの一覧を表示します。以下に例を示します。

```
Module                  Size  Used by
tun                    11585  1
autofs4                21573  1
hidp                   16193  2
rfcomm                 37849  0
l2cap                  23873  10 hidp,rfcomm
bluetooth              50085  5 hidp,rfcomm,l2cap
sunrpc                153725  1
dm_mirror              29073  0
```

```
dm_mod              57433  1 dm_mirror
video              17221  0
sbs                16257  0
i2c_ec              5569  1 sbs
container           4801  0
button              7249  0
battery            10565  0
asus_acpi          16857  0
ac                  5701  0
ipv6              246113  12
lp                 13065  0
parport_pc         27493  1
parport            37001  2 lp,parport_pc
uhci_hcd           23885  0
floppy             57317  1
sg                 34653  0
snd_ens1371        26721  1
gameport           16073  1 snd_ens1371
snd_rawmidi        24897  1 snd_ens1371
snd_ac97_codec     91360  1 snd_ens1371
snd_ac97_bus        2753  1 snd_ac97_codec
snd_seq_dummy       4293  0
snd_seq_oss        32705  0
serio_raw           7493  0
snd_seq_midi_event  8001  1 snd_seq_oss
snd_seq            51633  5 snd_seq_dummy,snd_seq_oss,snd_seq_midi_event
snd_seq_device      8781  4 snd_rawmidi,snd_seq_dummy,snd_seq_oss,snd_seq
snd_pcm_oss        42849  0
snd_mixer_oss      16833  1 snd_pcm_oss
snd_pcm            76485  3 snd_ens1371,snd_ac97_codec,snd_pcm_oss
snd_timer          23237  2 snd_seq,snd_pcm
snd                52933  12
snd_ens1371,snd_rawmidi,snd_ac97_codec,snd_seq_oss,snd_seq,snd_seq_device,snd_pcm
_oss,snd_mixer_oss,snd_pcm,snd_timer
soundcore          10145  1 snd
i2c_piix4           8909  0
ide_cd             38625  3
snd_page_alloc     10569  1 snd_pcm
i2c_core           21697  2 i2c_ec,i2c_piix4
pcnet32            34117  0
cdrom              34913  1 ide_cd
mii                 5825  1 pcnet32
pcspkr              3521  0
ext3              129737  2
jbd                58473  1 ext3
mptspi             17353  3
scsi_transport_spi  25025  1 mptspi
mptscsih           23361  1 mptspi
sd_mod             20929  16
scsi_mod          134121  5 sg,mptspi,scsi_transport_spi,mptscsih,sd_mod
mptbase            52193  2 mptspi,mptscsih
```

各行について、1 番目の列はモジュール名、2 番目の列はモジュールのサイズ、3 番目の列は使用数です。

/sbin/lsmod の出力は、/proc/modules の出力よりも詳細で読みやすいものになります。

カーネルモジュールを読み込むには、/sbin/modprobe コマンドの後にカーネルモジュール名を指定します。デフォルトでは、modprobe は /lib/modules/ <kernel-version> /kernel/drivers/ サブディレクトリーからモジュールの読み込みを試みます。ネットワークインターフェイスドライバーの net/ サブディレクトリーなど、各タイプのモジュールにはサブディレクトリーがあります。カーネルモジュールの中にはモジュールの依存関係があるものがあります。つまり、読み込むには他のモジュールを最初に読み込む必要があります。/sbin/modprobe コマンドは、これらの依存関係をチェックし、指定されたモジュールを読み込む前にモジュールの依存関係を読み込みます。

たとえば、コマンドは以下のようになります。

```
modprobe e100
```

モジュールの依存関係を読み込み、e100 モジュールをロードします。

/sbin/modprobe としてすべてのコマンドに出力するには、-v オプションを使用します。以下に例を示します。

```
modprobe -v e100
```

以下のような出力が表示されます。

```
insmod /lib/modules/2.6.9-5.EL/kernel/drivers/net/e100.ko
Using /lib/modules/2.6.9-5.EL/kernel/drivers/net/e100.ko
Symbol version prefix 'smp_'
```

/sbin/insmod コマンドは、カーネルモジュールを読み込むためにも存在しますが、依存関係は解決しません。したがって、/sbin/modprobe コマンドを使用することが推奨されます。

カーネルモジュールをアンロードするには、/sbin/rmmod コマンドの後にモジュール名を使用します。rmmod ユーティリティーは、使用されていないモジュールのみをアンロードします。これは、使用中の他のモジュールの依存関係ではありません。

たとえば、コマンドは以下のようになります。

```
rmmod e100
```

e100 カーネルモジュールをアンロードします。

もう 1 つの便利なカーネルモジュールユーティリティーは modinfo です。/sbin/modinfo コマンドを使用して、カーネルモジュールに関する情報を表示します。一般的な構文は以下のとおりです。

```
modinfo [options] <module>
```

オプションには、モジュールの簡単な説明を表示する -d と、モジュールがサポートするパラメーターを一覧表示する -p が含まれます。オプションの完全なリストは、modinfo の man ページ(man modinfo)を参照してください。

45.2. 永続モジュールの読み込み

カーネルモジュールは通常、/etc/modprobe.conf ファイルの正しい設定が指定されているファシリティーにより直接読み込まれます。ただし、システムの起動時にモジュールの読み込みを明示的に強制する必要がある場合があります。

Red Hat Enterprise Linux は、システムの起動時に /etc/rc.modules ファイルの存在をチェックします。このファイルには、モジュールを読み込むさまざまなコマンドが含まれています。rc.modules はブートプロセスで先に実行されるため、rc.local ではなく rc.modules を使用する必要があります。

たとえば、以下のコマンドは、システムの起動時に foo モジュールの読み込みを設定します(root として)。

```
echo modprobe foo >> /etc/rc.modules
chmod +x /etc/rc.modules
```

ヒント

ネットワークおよび SCSI インターフェイスに独自のメカニズムがあるため、このアプローチは必要ありません。

45.3. モジュールパラメーターの指定

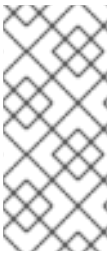
場合によっては、適切に機能させるには、モジュールがパラメーターを読み込む必要があるため、パラメーターをモジュールに提供する必要がある場合があります。

たとえば、Intel Ether Express/100 カードの接続速度を 100Mbps で全 duplex を有効にするに

は、e100_speed_duplex=4 オプションで e100 ドライバーを読み込みます。

注意

パラメーターにコンマがある場合は、必ずコンマの後にスペースを入れ ない で ください。



ヒント

modinfo コマンドは、バージョン、依存関係、パラメーターオプション、エイリアスなど、カーネルモジュールに関するさまざまな情報を一覧表示する場合にも便利です。

45.4. ストレージパラメーター

表45.1 ストレージモジュールパラメーター

ハードウェア	*モジュール*	*パラメーター*
3ware Storage Controller および 9000 シリーズ	*3w-xxxx.ko、3w-9xxx.ko*	
Adaptec Advanced Raid 製品、Dell PERC2、2/Si、3/Si、3/Di、HP NetRAID-4M、IBM ServeRAID、および ICP SCSI ドライバー	*aacraid.ko*	*nondasd - 非dasd デバイスに対する hba のスキャンを制御します。0=off, 1=on* *dacmode - dma アドレス指定が 64 ビット DAC を使用しているかどうかを制御します。0=off, 1=on* *commit: 外部アレイのアダプターに COMMIT_CONFIG を発行するかどうかを制御し*

ハードウェア	モジュール	ます。これは通常、BIOS を持たないシステムに必要です。0=off, 1=on パラメーター
		startup_timeout - アダプターがカーネルを起動して実行しているのを待機する時間（秒単位）。通常、これは BIOS を持たない大規模なシステムで調整されます。
		aif_timeout - アプリケーションが AIF を取得するのを待機する時間（秒単位）。通常、これは負荷の高いシステム向けに調整されます。
		numacb: 割り当てたアダプター制御ブロック(FIB)の数に制限を要求します。有効な値は 512 および down です。デフォルトでは、ファームウェアからの提案を使用します。
		acbsize - 特定のアダプター制御ブロック(FIB)サイズを要求します。有効な値は 512、2048、4096、および 8192 です。デフォルトでは、ファームウェアからの提案を使用します。
Adaptec 28xx、R9xx、39xx AHA-284x、AHA-29xx、	aic7xxx.ko	

ハードウェア	モジュール	パラメーター
AHA-394x、AHA-398x、AHA-274x、AHA-2842、AHA-2910B、AHA-2920C、AHA-2930/U/U2 aHA-2940/w/u/uW/AU/、u2W/u2/U2B/、u2BOEM、aHA-2944d/wd/UD/uWD、aHA-2950u2/W/b、aHA-3940/u/w/uW/、auW/u2W/u2B、 AHA-3950U2D, AHA-3985/U/W/UW, AIC-777x, AIC-785x, AIC-786x, AIC-787x, AIC-788x , AIC-789x, AIC-3860		詳細: 詳細/診断ロギングの有効化 allow_memio - デバイスレジスターをメモリーマップを許可します。 debug - 有効にするデバッグ値のビットマスク no_probe: EISA/VLB コントローラーのプロービングを切り替え probe_eisa_vl - EISA/VLB コントローラープロービングを切り替え no_reset - 初期バスのリセットをスプレッシャル extended - すべてのコントローラーで拡張されたジオメトリーの有効化 periodic_otag: タグ不足を防ぐために、順序付けされたタグ付けされたトランザクションを定期的に送信しま

ハードウェア	モジュール	す。これは、古いディスクドライブまたは RAID アレイで必要になる場合があります。パラメーター
		tag_info:<tag_str >: ターゲットごとのタグの深さを設定します。 global_tag_depth:<int> - すべてのバス上のすべてのターゲットに対するグローバルタグの深さ seltime:<int& gt; - Selection Timeout (0/256ms,1/128ms,2/64ms,3/32ms)
IBM ServeRAID	ips.ko	

ハードウェア	モジュール	パラメーター
LSI Logic MegaRAID Mailbox Driver	megaraid_mbox.ko	unconf_disks - 未設定のディスクをカーネルに公開するように設定します (default=0)。 busy_wait - ビジー時にメールボックスの最大待機時間（デフォルトは 100） max_sectors: IO コマンドごとのセクターの最大数（デフォルトは 128） cmd_per_lun: 論理ユニットごとのコマンドの最大数（デフォルトは 64） fast_load - ドライバーの高速読み込み、物理デバイスをスキップします。（デフォルト=0） debug_level - ドライバーのデバッグレベル（デフォルトは 0）
Emulex LightPulse Fibre	lpfc.ko	

ハードウェア	モジュール	パラメーター
Emulex Light-pulse Fibre Channel SCSI ドライバー	lpfc.ko	lpfc_poll - FCP リングポーリングモードの制御：0 - none、1 - 割り込みが有効なポーリング 3 - FCP リング割り込みのポーリングと無効化 lpfc_log_verbose — Verbose logging bit-mask lpfc_lun_queue_depth - 特定の LUN にキューに入れることができる FCP コマンドの最大数 lpfc_hba_queue_depth: lpfc HBA にキューに入れることができる FCP コマンドの最大数 lpfc_scan_down - デバイスのスキャンを最も高い ALPA から最小にスキャンを開始します。 lpfc_nodev_tmo - Seconds ドライバーは、デバイスが戻るのを待つ I/O を保持します lpfc_topology - ファイバーチャネルトポロジーの選

ハードウェア	モジュール	択 パラメーター
		lpfc_link_speed - リンク速度の選択 lpfc_fcp_class - FCP シーケンスに対してファイバーチャネルクラスを選択します。 lpfc_use_adisc - 再検出時に ADISC を使用して FCP デバイスを認証します。 lpfc_ack0 - ACK0 サポートの有効化 lpfc_cr_delay: 割り込み応答が生成されるまでのミリ秒数 lpfc_cr_count - 割り込み応答が生成されるまでの I/O 完了数 lpfc_multi_ring_support - IOCB エントリーを分散させる

ハードウェア	モジュール	プライマリー SLI リングの数パラメーター。
		lpfc_fdmi_on - FDMI サポートの有効化 lpfc_discovery_threads: 検出時の ELS コマンドの最大数 lpfc_max_luns: 許可される最大 LUN lpfc_poll_tmo - Milliseconds ドライバーが FCP リングをポーリングするまで待機する
HP Smart Array	cciss.ko	

ハードウェア	モジュール	パラメーター
LSI Logic MPT Fusion	mptbase.ko mptctl.ko mptfc.ko mptlan.ko mptsas.ko mptscsih.ko mptspi.ko	mpt_msi_enable - MSI サポートの有効化 mptfc_dev_loss_tmo - ドライバープログラムが、デバイス損失イベントに従って rport が返されるのを待つ初期時間。 mpt_pt_clear - Clear persistency table mpt_saf_te: SEP プロセッサーを強制的に有効にする
QLogic Fibre Channel Driver	qla2xxx.ko	ql2xlogintimeout - ログインのタイムアウト値（秒単位） qlport_down_retry: PORT-DOWN ステータスを返すポートへのコマンドの最大再試行回数

ハードウェア	モジュール	パラメーター
		ql2xplogiabsentdevice - PLOGI にスキャン後に存在しないデバイスに PLOGI を有効にするオプション。 ql2xloginretrycount - NVRAM ログイン再試行回数の代替値を指定します。 ql2xallocfwdump - HBA の初期化中にファームウェアダンプのメモリー割り当てを有効にするオプション。デフォルトは 1 で、メモリーを割り当てます。 extended_error_logging - 拡張エラーログを有効にするオプション。 ql2xfdmienable - FDMI 登録を有効にします。
NCR、Symbios、LSI 8xx、および 1010	sym53c8xx	cmd_per_lun - デフォルトで使用するタグの最大数 tag_ctrl: LUN ごとのタグの詳細な制御

ハードウェア	モジュール	パラメーター
		Burst: 最大バースト0: 無効にするには 255、レジスターから読み取る場合は 255 led - 1 に設定して、LED サポートを有効にします。 diff - 0 （差分モードなし）、BIOS の場合は 1、常に 2、GPIO3 ではなく 3 irqm - open drain の場合は 0、totem pole をそのまま残す 1 buschk - 確認しない場合は 0、エラーの割り当て解除は 1、エラーの警告は 2 になります。 hostid - ホストアダプターに使用する SCSI ID verb - 最小の詳細度は 0、通常の場合は 1、過剰の場合は 2

ハードウェア	モジュール	パラメーター
		debug - デバッグを有効にするビットを設定します。 settle - 秒単位の遅延を設定します。デフォルト 3 NVRAM - 現在使用されていないオプション excl: コントローラーが接続されないように、ここに ioport アドレスを一覧表示します。 safe - その他の設定をセーフモードに設定します。

45.5. イーサネットパラメーター

重要な影響

最新のイーサネットベースのネットワークインターフェイスカード(NIC)のほとんどは、設定を変更するためにモジュールパラメーターを必要としません。代わりに、ethtool または mii-tool を使用して設定できます。これらのツールが機能しなかった後にのみ、モジュールパラメーターを調整する必要があります。モジュールパラメーターは、modinfo コマンドを使用して表示できます。

注記

これらのツールの使用方法は、ethtool、mii-tool、および modinfo の man ページを参照してください。

表45.2 Ethernet モジュールパラメーター

ハードウェア	モジュール	パラメーター
3com EtherLink PCIvideo/XL Vortex (3c590, 3c592, 3c597) Boomerang (3c900, 3c905, 3c595)	3c59x.ko	debug - 3c59x デバッグレベル(0-6) オプション - 3c59x: Bits 0-3: メディアタイプ、ビット 4: バスマスター、ビット 9: 完全なデュプレックス global_options - 3c59x: オプションと同じですが、オプションが設定されていないとすべての NIC に適用されます。 full_duplex - 3c59x full duplex setting (s) (1) global_full_duplex - 3c59x: full_duplex と同じですが、full_duplex が設定されていない場合にはすべての NIC に適用されます。

ハードウェア	モジュール	パラメーター
		hw_checksums - アダプターによる 3c59x ハードウェアチェックサムチェック(0-1) flow_ctrl - 3c59x 802.3x フロー制御の使用(PAUSE のみ) (0-1) enable_wol - 3c59x: アダプターの Wake-on-LAN をオンにする(0-1) global_enable_wol - 3c59x: enable_wol と同じですが、enable_wol が設定されていない場合にはすべての NIC に適用されます。 rx_copybreak - 3c59x copy breakpoint for copy-only-tiny-frames max_interrupt_work - 割り込みごとに処理される 3c59x の最大イベント compaq_ioaddr - 3c59x PCI I/O ベースアドレス (Compaq BIOS の問題回避策)

ハードウェア	モジュール	パラメーター
		compaq_irq - 3c59x PCI IRQ 番号(Compaq BIOS の問題回避策) *compaq_device_id - 3c59x PCI デバイス ID (Compaq BIOS の問題回避策)* *watchdog - 3c59x 送信タイムアウト（ミリ秒単位）* *global_use_mmio - 3c59x: use_mmio と同じですが、オプションが設定されていないとすべての NIC に適用されます。* *use_mmio - 3c59x: メモリーマッピング PCI I/O リソース(0-1)を使用します。*
RTL8139、SMC EZ Card Fast Ethernet、RTL8129 を使用する RealTek カード、または RTL8139 Fast Ethernet チップセット	*8139too.ko*	

ハードウェア	モジュール	パラメーター
Broadcom 4400 10/100 PCI ethernet ドライバー	b44.ko	b44_debug - B44 ビットマッピングのデバッグメッセージの有効化値
Broadcom NetXtreme II BCM5706/5708 Driver	bnx2.ko	disable_msi - メッセージシグナル割り込み(MSI)の無効化
Intel Ether Express/100 ドライバー	e100.ko	debug - デバッグレベル (0=none,...,16=all) eeprom_bad_csum_allow - 不正な eeprom チェックサムを許可します。
Intel EtherExpress/1000 ギガビット	e1000.ko	TxDescriptors - 送信記述子の数 RxDescriptors - 受信記述子の数

ハードウェア	モジュール	パラメーター
		Speed - Speed 設定 duplex - Duplex 設定 AutoNeg - アドバタイズされたオートネゴシエーション設定 flowcontrol: フロー制御の設定 XsumRX - Receive Checksum オフロードを無効または有効にします。 TxIntDelay - 送信割り込みの遅延 TxAbsIntDelay - mit Absolute Interrupt Delay RxIntDelay: Receive Interrupt Delay

ハードウェア	モジュール	パラメーター
		RxAbsIntDelay - Receive Absolute Interrupt Delay *InterruptThrottleRate - Interrupt Throttling Rate* *SmartPowerDownEnable - Enable PHY smart power down* *KumeranLockLoss - Kumeran ロック損失回避策を有効にする*
Myricom 10G driver (10GbE)	*myri10ge.ko*	*myri10ge_fw_name - ファームウェアイメージ名* *myri10ge_ecrc_enable - PCI-E で Extended CRC を有効にする* *myri10ge_max_intr_slots - 割り込みキュースロット*

ハードウェア	モジュール	パラメーター
		myri10ge_small_bytes - スモールパケットのしきい値 myri10ge_msi - メッセージ署名割り込みの有効化 myri10ge_intr_coal_delay - Interrupt coalescing delay myri10ge_flow_control - Pause パラメーター myri10ge_deassert_wait - レガシー割り込みを破棄する際に待機する myri10ge_force_firmware: ファームウェアがアライメントされた完了を想定するように強制する myri10ge_skb_cross_4k - 4KB の境界を超える小規模な skb 可能性

ハードウェア	モジュール	パラメーター
		myri10ge_initial_mtu - 初期 MTU *myri10ge_napi_weight — Set NAPI weight* *myri10ge_watchdog_timeout - ウォッチドッグのタイムアウトを設定します。* *myri10ge_max_irq_loops - スタックしたレガシー IRQ 検出しきい値を設定します*

ハードウェア	モジュール	パラメーター
NatSemi DP83815 Fast Ethernet	natsemi.ko	MTU: DP8381x MTU（すべてのボード） debug - DP8381x のデフォルトデバッグレベル rx_copybreak - DP8381x copy breakpoint for copy-only-tiny-frames オプション - DP8381x: Bits 0-3: メディアタイプ、ビット 17: full duplex full_duplex - DP8381x full duplex setting (s) (1)
AMD PCnet32 and AMD PCnetPCI	pcnet32.ko	
PCnet32 および PCnetPCI	pcnet32.ko	debug - pcnet32 デバッグレベル max_interrupt_work - pcnet32 割り込みごとに処理

ハードウェア	モジュール	される最大イベント パラメーター
		rx_copybreak - pcnet32 copy breakpoint for copy-only-tiny-frames *tx_start_pt - pcnet32 送信開始ポイント(0-3)* *pcnet32vlb - pcnet32 Vesa local bus (VLB)サポート(0/1)* *オプション - pcnet32 初期オプション設定(0-15)* *full_duplex - pcnet32 full duplex setting (s) (1)* *homepna - 79C978 カードの pcnet32 モード (HomePNA の場合は 1、イーサネットの場合は 0、デフォルトイーサネット)*

ハードウェア	モジュール	パラメーター
Realtek RTL-8169 Gigabit Ethernet driver	r8169.ko	メディア: phy 操作を強制します。ethtool (8)で非推奨になりました。 rx_copybreak - copy-only-tiny-frames のブレークポイントをコピー use_dac - PCI DAC を有効にします。32 ビットの PCI スロットでは安全ではありません。 debug - デバッグの詳細レベル(0=none, ..., 16=all)
Neterion Xframe 10GbE サーバーアダプター	s2io.ko	

ハードウェア	モジュール	パラメーター
SIS 900/701G PCI Fast Ethernet	sis900.ko	multicast_filter_limit - SiS 900/7016 フィルターリングされたマルチキャストアドレスの最大数 max_interrupt_work - SiS 900/7016 割り込みごとに処理される最大イベント sis900_debug - SiS 900/7016 ビットマッピングのデバッグメッセージレベル
Adaptec Starfire イーサネットドライバー	starfire.ko	max_interrupt_work: 割り込みごとに処理される最大イベント MTU: MTU （全ボード） debug - デバッグレベル (0-6) rx_copybreak - copy-only-tiny-frames のブレーク

ハードウェア	モジュール	ポイントをコピー パラメーター
		intr_latency: 最大割り込みレイテンシー（マイクロ秒単位） small_frames: 割り込みレイテンシーをバイパスする受信フレームの最大サイズ (0,64,128,256,512) オプション - 非推奨 : Bits 0-3: メディアタイプ、ビット 17: 完全なデュプレックス full_duplex - 非推奨 : 完全なデュプレックス設定を強制(0/1) enable_hw_cksum - ハードウェアcksum サポートを有効/無効にします(0/1)
Broadcom Tigon3	tg3.ko	tg3_debug - Tigon3 ビットマッピングのデバッグメッセージの有効化値

ハードウェア	モジュール	パラメーター
ThunderLAN PCI	tlan.ko	aui - ThunderLAN は AUI ポート(0-1)を使用します。 duplex - ThunderLAN デュプレックス設定(0-default、1-half、2-full) 速度 - ThunderLAN ポートスペンデン設定(0,10,100) debug - ThunderLAN デバッグマスク bbuf - ThunderLAN は big buffer (0-1)を使用します。

ハードウェア	モジュール	パラメーター
digital 21x4x Tulip PCI Ethernet cards SMC EtherPower 10 PCI (8432T/8432BT) SMC EtherPower 10/100 PCI (9332DST) DEC EtherWorks 100/10 PCI (DE500-XA) DEC EtherWorks 10 PCI (DE450) DEC QSILVER's, Znyx 312 etherarray Allied Telesis LA100PCI-T Danpex EN-9400, Cogent EM110	tulip.ko	io io_port
VIA VT86c100A Rhine-II PCI または 3043 Rhine-I D-Link DFE-930-TX PCI 10/100 のいずれかを使用した VIA Rhine PCI Fast Ethernet カード	via-rhine.ko	max_interrupt_work - VIA Rhine 割り込みごとに処理される最大イベント debug - VIA Rhine デバッグレベル(0-7) rx_copybreak - copy-only-tiny-frames の VIA Rhine copy breakpoints avoid_D3 - 電源状態 D3 を回避（破損した BIOS の回避策）

45.5.1. Channel Bonding モジュール

Red Hat Enterprise Linux では、管理者は ボンディングカーネルモジュールと、チャンネルボンディング インターフェイス と呼ばれる特別なネットワークインターフェイスを使用して、NIC を 1 つ

の チャンネル にバインドできます。このチャンネルボンディングにより、複数のネットワークインターフェイスが 1 つとして機能できるようになり、また同時に帯域幅が増加し、冗長性を提供します。

複数のネットワークインターフェイスをチャンネル化するには、管理者は以下の手順を実行する必要があります。

1. 以下の行を /etc/modprobe.conf に追加します。

```
alias bond<N> bonding
```

&lt ;N& gt; を 0 などのインターフェイス番号に置き換えます。設定したチャンネルボンディングインターフェイスごとに、/etc/modprobe.conf に対応するエントリーが必要です。

2. 「チャンネルボンディングインターフェイス」 で説明されているように、チャンネルボンディングインターフェイスを設定します。

3. パフォーマンスを強化するには、利用可能なモジュールオプションを調節して、最適な組み合わせを確認します。特に miimon、arp_interval、arp_ip_target パラメーターに注意してください。利用可能なオプション一覧と、ボンディングされたインターフェイスに最適なオプションを迅速に判別する方法については、「ボンディングモジュールのディレクティブ」 を参照してください。

45.5.1.1. ボンディングモジュールのディレクティブ

ボンディングインターフェイスの設定ファイル(ifcfg-bond0 など)の BONDING_OPTS="<bonding parameters> " ディレクティブに追加する前に、ボンディングされたインターフェイスにどのチャンネルボンディングモジュールパラメーター が最適かをテストすることが推奨されます。ボンディングされたインターフェイスのパラメーターは、sysfs ファイルシステム内のファイルを操作することで、ボンディングモジュールをアンロード (およびリロード) することなく設定できます。

sysfs は、カーネルオブジェクトをディレクトリー、ファイル、シンボリックリンクとして表す仮想ファイルシステムです。sysfs を使用すると、カーネルオブジェクトに関する情報のクエリーや、通常のファイルシステムコマンドを使用することでこれらのオブジェクトを操作することもできます。sysfs 仮想ファイルシステムには /etc/fstab の行があり、/sys の下にマウントされます。ボンディングされたインターフェイスはすべて、/sys/class/net/ ディレクトリー配下のファイルと対話して動的に設定できます。

ifcfg-bond0 などのチャンネルボンディングインターフェイスファイルを作成し、「チャンネルボンディングインターフェイス」 の手順に従ってボンディングされたインターフェイスに SLAVE=yes

ディレクティブおよび MASTER=bond0 ディレクティブを挿入した後に、ボンディングされたインターフェイスに最適なパラメーターのテストと判断に進むことができます。

まず、ifconfig bond < N> を root として実行して、作成した ボンディング を起動 します。

```
ifconfig bond0 up
```

ifcfg-bond0 ボンディングインターフェイスファイルを正しく作成した場合は、ifconfig の実行の出力に bond0 が一覧表示されます（オプションなし）。

```
~]# ifconfig
bond0     Link encap:Ethernet  HWaddr 00:00:00:00:00:00
          UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
eth0      Link encap:Ethernet  HWaddr 52:54:00:26:9E:F1
          inet addr:192.168.122.251  Bcast:192.168.122.255  Mask:255.255.255.0
          inet6 addr: fe80::5054:ff:fe26:9ef1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:207 errors:0 dropped:0 overruns:0 frame:0
          TX packets:205 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:70374 (68.7 KiB)  TX bytes:25298 (24.7 KiB)
[output truncated]
```

アップになっていないボンドも含めてすべての既存のボンドを表示するには、以下を実行します。

```
~]# cat /sys/class/net/bonding_masters
bond0
```

/sys/class/net/bond <N> /bonding/ ディレクトリーにあるファイルを操作することで、各ボンドを個別に設定できます。まず、設定するボンドをダウンにします。

```
ifconfig bond0 down
```

たとえば、bond0 の MII 監視を 1 秒間隔で有効にするには、(root として)を実行できます。

```
echo 1000 > /sys/class/net/bond0/bonding/miimon
```

bond0 を balance-alb モードに設定するには、以下のいずれかを実行できます。

```
echo 6 > /sys/class/net/bond0/bonding/mode
```

またはモード名を使用します。

```
echo balance-alb > /sys/class/net/bond0/bonding/mode
```

問題のボンドにいくつかのオプションを設定した後に、ifconfig bond <N > をup にすると、それを起動してテストできます。オプションを変更する場合はインターフェイスを停止して、sysfs を使用してそのパラメーターを修正後、有効に戻して再確認します。

ボンディングに最適なパラメーターのセットを決定したら、設定しているボンディングされたインターフェイスの /etc/sysconfig/network-scripts/ifcfg-bond <N > ファイルの BONDING_OPTS= ディレクティブにそれらのパラメーターをスペース区切りリストとして追加します。ボンディングが起動すると(ONBOOT=yes ディレクティブが設定されている場合はブートシーケンス中にシステムがシステムによって)、BONDING_OPTS で指定されたボンディングオプションがそのボンディングに対して有効になります。ボンディングされたインターフェイス（および BONDING_OPTS）の設定に関する詳細は、「チャンネルボンディングインターフェイス」 を参照してください。

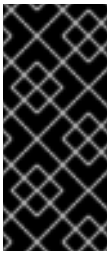
ボンディングモジュールで利用可能なチャンネルボンディングモジュールパラメーターの一覧を以下に示します。チャンネルボンディングの設定に関する詳細とボンディングモジュールパラメーターの完全リストの詳細は、kernel-doc パッケージをインストールし、含まれている bonding.txt ファイルを見つけて開きます。

```
yum -y install kernel-doc
nano -w $(rpm -ql kernel-doc | grep bonding.txt)
```

ボンディングインターフェイスパラメーター

arp_interval=<time_in_milliseconds>

ARP 監視が発生する頻度を指定します（ミリ秒単位）。



重要な影響

arp_interval および arp_ip_target の両パラメーター、あるいは miimon パラメーターの指定は不可欠です。指定されないと、リンクが失敗した場合にネットワークパフォーマンスが低下する恐れがあります。

mode= 0 または mode =1 (2 つの負荷分散モード)でこの設定を使用する場合は、NIC 全体に均等にパケットを分散するようにネットワークスイッチを設定する必要があります。これを実行する方法の詳細については、/usr/share/doc/kernel-doc- <kernel_version> /Documentation/networking/bonding.txt を参照してください。

デフォルトでは値は 0 に設定されており、ARP 監視を無効にします。

arp_ip_target=<ip_address> [,<ip_address_2>,...<ip_address_16>]

arp_interval パラメーターが有効な場合に ARP 要求のターゲット IP アドレスを指定します。コンマ区切りのリストで、最大 16 個の IP アドレスを指定できます。

arp_validate=<value>

ARP プローブのソース/ディストリビューションを検証します。デフォルトは none です。他の有効な値は、active、backup、および all です。

debug=<number>

デバッグメッセージを有効にします。以下の値が使用できます。

- *0 : デバッグメッセージが無効になります。これがデフォルトです。*
- *1 : デバッグメッセージが有効になっています。*

downdelay=<time_in_milliseconds>

リンクを無効にする前に、リンクの失敗後に待機する時間を指定します (ミリ秒単位)。値は、miimon パラメーターで指定される値の倍数でなければなりません。デフォルトでは値は 0 に設定されており、ARP 監視を無効にします。

lacp_rate=<value>

リンクパートナーが 802.3ad モードで LACPDU パケットを送信するレートを指定します。以下の値が使用できます。

- *slow または 0: デフォルト設定。パートナーが 30 秒ごとに LACPDU を送信するよう指定します。*

- *fast または 1: パートナーが LACPDU を 1 秒ごとに送信するように指定します。*

miimon=<time_in_milliseconds>

MII リンク監視が発生する頻度を指定します (ミリ秒単位)。MII は NIC がアクティブであることを検証するために使用されるため、これは高可用性が必要な場合に役立ちます。特定の NIC のドライバーが MII ツールに対応していることを確認するには、root で以下のコマンドを入力します。

```
ethtool <interface_name> | grep "Link detected:"
```

このコマンドで、&lt ;interface_name> をボンディングインターフェイスではなく、eth0 などのデバイスインターフェイスの名前に置き換えます。MII が対応している場合は、コマンドは以下を返します。

```
Link detected: yes
```

高可用性のためにボンディングされたインターフェイスを使用する場合、各 NIC のモジュールは MII に対応していなければなりません。値を 0 (デフォルト) に設定すると、この機能はオフになります。この設定を設定する際に、このパラメーターのスタート地点は 100 になります。

重要な影響

arp_interval および arp_ip_target の両パラメーター、あるいは miimon パラメーターの指定は不可欠です。指定されないと、リンクが失敗した場合にネットワークパフォーマンスが低下する恐れがあります。

mode=<value>

ここでの &lt ;value> は以下のいずれかになります。

- *balance-rr または 0: 耐障害性とロードバランシングにラウンドロビンポリシーを設定します。利用可能な最初のインターフェイスからそれぞれのボンディングされたスレーブインターフェイスで送受信が順次行われます。*

- *active-backup または 1: 耐障害性のためアクティブなバックアップポリシーを設定します。利用可能な最初のボンディングされたスレーブインターフェイスを介して送受信が行われます。別のボンディングされたスレーブインターフェイスは、アクティブなボンディングされたスレーブインターフェイスが失敗した場合にのみ使用されます。*

- *balance-xor または 2: 耐障害性とロードバランシングに XOR （排他的または）ポリシーを設定します。この方法では、インターフェイスは、スレーブ NIC のいずれかに対して、受信要求の MAC アドレスと MAC アドレスが照合されます。このリンクが確立されると、最初に利用可能なインターフェイスから順番に送信が送信されます。*

- *broadcast または 3: 耐障害性にブロードキャストポリシーを設定します。すべての送信は、すべてのスレーブインターフェイスで行われます。*

- *802.3ad または 4: IEEE 802.3ad 動的リンクアグリゲーションのポリシーを設定します。同一の速度とデュプレックス設定を共有するアグリゲーショングループを作成します。アクティブなアグリゲーターのすべてのスレーブで送受信を行います。802.3ad に対応するスイッチが必要です。*

- *balance-tlb または 5: 耐障害性とロードバランシングのための送信ロードバランシング (TLB) ポリシーを設定します。発信トラフィックは、各スレーブインターフェイスの現在の負荷に従って分散されます。受信トラフィックは、現在のスレーブにより受信されます。受信しているスレーブが失敗すると、別のスレーブが失敗したスレーブの MAC アドレスを引き継ぎます。*

- *balance-alb または 6: 耐障害性とロードバランシングに Active Load Balancing (ALB)ポリシーを設定します。IPV4 トラフィック用の送受信負荷分散が含まれます。ARP ネゴシエーションにより、受信負荷分散が実現されます。*

num_unsol_na=<number>

フェイルオーバーイベント後に発行される未設定の IPv6 Neighbor Advertisement の数を指定します。フェイルオーバー直後に、未承認の NA が発行されます。

有効な範囲は 0 - 255 で、デフォルト値は 1 です。このオプションは、active-backup モードにのみ影響します。

primary=<interface_name>

プライマリーデバイスのインターフェイス名（例： eth0 ）を指定します。primary デバイスは、使用される最初のボンディングインターフェイスであり、失敗しない限りは破棄されません。この設定が特に役立つのは、ボンディングインターフェイスの NIC の 1 つが高速なため、大規模な負荷に対応できる場合です。

この設定は、ボンディングインターフェイスが active-backup モードの場合にのみ有効です。詳細は、 /usr/share/doc/kernel-doc- <kernel-version> /Documentation/networking/bonding.txt を参照してください。

primary_reselect=<value>

プライマリースレーブに対して再選択ポリシーを指定します。これは、アクティブなスレーブの失敗やプライマリースレーブの回復が発生した場合に、どのようにプライマリースレーブが選択されてアクティブなスレーブになるかという点に影響します。このオプションは、プライマリースレーブと他のスレーブ間のフラップを回避するように設計されています。以下の値が使用できます。

- always または 0: プライマリースレーブは有効になるといつでもアクティブなスレーブになります。
- better または 1: プライマリースレーブの速度とデュプレックスが、現在のアクティブなスレーブの速度とデュプレックスと比べて良い場合は、プライマリースレーブは有効になるとアクティブなスレーブになります。
- failure または 2: 現在のアクティブなスレーブが失敗してプライマリースレーブが有効になる場合のみ、プライマリースレーブはアクティブなスレーブになります。

primary_reselect の設定は、以下の 2 つの場合では無視されます。

- アクティブなスレーブがない場合は、回復する最初のスレーブがアクティブなスレーブになります。
- 初めにプライマリースレーブがスレーブにされた場合は、それは常にアクティブなスレーブになります。

sysfs で primary_reselect ポリシーを変更すると、新しいポリシーに従って、最適なアクティブなスレーブが即座に選択されます。これにより、状況によってはアクティブなスレーブに変

更が生じる場合があります。

updelay=<time_in_milliseconds>

リンクを有効にする前の待機時間を指定します (ミリ秒単位)。値は、miimon パラメーターで指定される値の倍数でなければなりません。デフォルトでは値は 0 に設定されており、ARP 監視を無効にします。

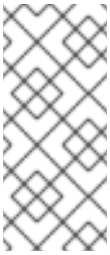
use_carrier=<number>

リンク状態を決定するために miimon が MII/ETHTOOL ioctls または netif_carrier_ok() を使用するかどうか指定します。netif_carrier_ok() 機能は、デバイスドライバーを使用して netif_carrier_on/off で状態を維持します。ほとんどのデバイスドライバーはこの機能をサポートします。

MII/ETHROOL ioctls ツールは、カーネル内の非推奨の呼び出しシーケンスを使用します。ただし、デバイスドライバーが netif_carrier_on/off に対応していない場合でも、これは設定可能です。

有効な値は以下のとおりです。

- *1: デフォルト設定。netif_carrier_ok() の使用を有効にします。*
- *0: MII/ETHTOOL ioctls の使用を有効にします。*



ヒント

リンクがアップになっているべきでないときにボンディングインターフェイスがアップになっている場合は、ネットワークデバイスドライバーが netif_carrier_on/off に対応していない可能性があります。

xmit_hash_policy=<value>

balance-xor および 802.3ad モードで、スレーブを選択する時に使用する送信ハッシュポリシーを選択します。以下の値が使用できます。

-

0 または layer2: デフォルト設定。このオプションは、ハードウェア MAC アドレスの XOR を使用してハッシュを生成します。使用する式は以下のとおりです。

```
(<source_MAC_address> XOR <destination_MAC>) MODULO <slave_count>
```

このアルゴリズムは、すべてのトラフィックを同じスレーブの特定のネットワークピアに割り振り、802.3ad に対応します。

- *1 または layer3+4: 上位レイヤープロトコルの情報を (利用可能な場合は) 使用して、ハッシュを生成します。これにより、特定のネットワークピアへのトラフィックが複数のスレーブに及ぶようにできますが、単一の接続では複数のスレーブに及びません。*

  断片化された TCP および UDP パケットに使用される公式は、以下のとおりです:

  ```
  ((<source_port> XOR <dest_port>) XOR
    ((<source_IP> XOR <dest_IP>) AND 0xffff)
      MODULO <slave_count>
  ```

  断片化された TCP または UDP パケットおよび他のすべての IP プロトコルトラフィックの場合、送信元ポートおよび宛先ポート情報は省略されます。非 IP トラフィックの場合、式は layer2 送信ハッシュポリシーと同じです。

  このポリシーの目的は、特に PFC2 付きの Cisco スイッチや Foundry および IBM 製品など一部のスイッチの動作を真似ることです。

  このポリシーで使用されるアルゴリズムは、802.3ad に対応していません。

- *2 または layer2+3: layer2 および layer3 プロトコル情報の組み合わせを使用して、ハッシュを生成します。*

  ハードウェア MAC アドレスと IP アドレスの XOR を使用してハッシュを生成します。式は以下のとおりです。

  ```
  (((<source_IP> XOR <dest_IP>) AND 0xffff) XOR
    ( <source_MAC> XOR <destination_MAC> ))
      MODULO <slave_count>
  ```

このアルゴリズムは、すべてのトラフィックを同じスレーブの特定のネットワークピアに割り振ります。非 IP トラフィックの場合、式は layer2 送信ハッシュポリシーと同じです。

このポリシーの目的は、特に layer3 ゲートウェイデバイスが大半の宛先に到達する必要がある環境において、layer2 単独の場合より分散されたトラフィックを提供することです。

このアルゴリズムは、802.3ad に対応しています。

45.6. 関連情報

カーネルモジュールとそのユーティリティーの詳細は、以下のリソースを参照してください。

45.6.1. インストールされているドキュメント

- lsmod の man ページ - 出力の説明と説明。
- insmod の man ページ - コマンドラインオプションの説明とリスト。
- modprobe の man ページ - コマンドラインオプションの説明と一覧
- rmmod の man ページ - コマンドラインオプションの説明とリスト。
- modinfo の man ページ - コマンドラインオプションの説明とリスト。
- /usr/share/doc/kernel-doc- <version> /Documentation/kbuild/modules.txt - カーネルモジュールをコンパイルし、使用する方法。このファイルを読み取るには、kernel-doc パッケージがインストールされている必要があります。

45.6.2. 便利な Web サイト

- http://tldp.org/HOWTO/Module-HOWTO/: 『Linux ドキュメントプロジェクトの Linux Loadable カーネルモジュール HOWTO』

[9]
ドライバーは、Linux が特定のハードウェアデバイスを使用できるようにするソフトウェアです。ドライバーがないと、カーネルは接続されたデバイスと通信できません。

第46章 KDUMP クラッシュリカバリーサービス

kdump は高度なクラッシュダンプメカニズムです。有効にすると、システムは別のカーネルのコンテキストから起動します。この 2 番目のカーネルは少量のメモリーを予約し、その唯一の目的は、システムがクラッシュした場合にコアダンプイメージを取得することです。コアダンプを分析する機能は、システム障害の正確な原因を特定するのに役立ちます。そのため、この機能を有効にすることが強く推奨されます。

本章では、Red Hat Enterprise Linux で kdump サービスの設定、テスト、および使用方法を説明します。また、crash デバッグユーティリティーを使用して、作成されるコアダンプを分析する方法の概要を説明します。

46.1. KDUMP サービスのインストール

システムで kdump サービスを使用するには、kexec-tools パッケージがインストールされていることを確認します。これを行うには、root で次のコマンドを実行します。

```
~]# yum install kexec-tools
```

Red Hat Enterprise Linux に新しいパッケージをインストールする方法の詳細は、パートII「パッケージ管理」 を参照してください。

46.2. KDUMP サービスの設定

kdump サービスの設定には、一般的な 3 つの方法があります。初回起動時に有効にして設定するか、グラフィカルユーザーインターフェイスに Kernel Dump Configuration ユーティリティーを使用するか、コマンドラインで手動で行います。

重要

Intel IOMMU ドライバーの現在の実装の制限により、kdump サービスがコアダンプイメージをキャプチャーできないことがあります。Intel アーキテクチャーで kdump を確実に使用するには、IOMMU サポートを無効にすることが推奨されます。

警告

同じベンダーの HP Smart Array デバイスとシステムボードの特定の組み合わせで kdump サービスが確実に機能しないことが知られています。このため、ユーザーは本番環境で使用する前に設定をテストすることを強く推奨します。必要に応じて、ネットワーク経由でカーネルクラッシュダンプをリモートマシンに保存するように kdump を設定することを強くお勧めします。kdump 設定のテスト方法は、「設定のテスト」 を参照してください。

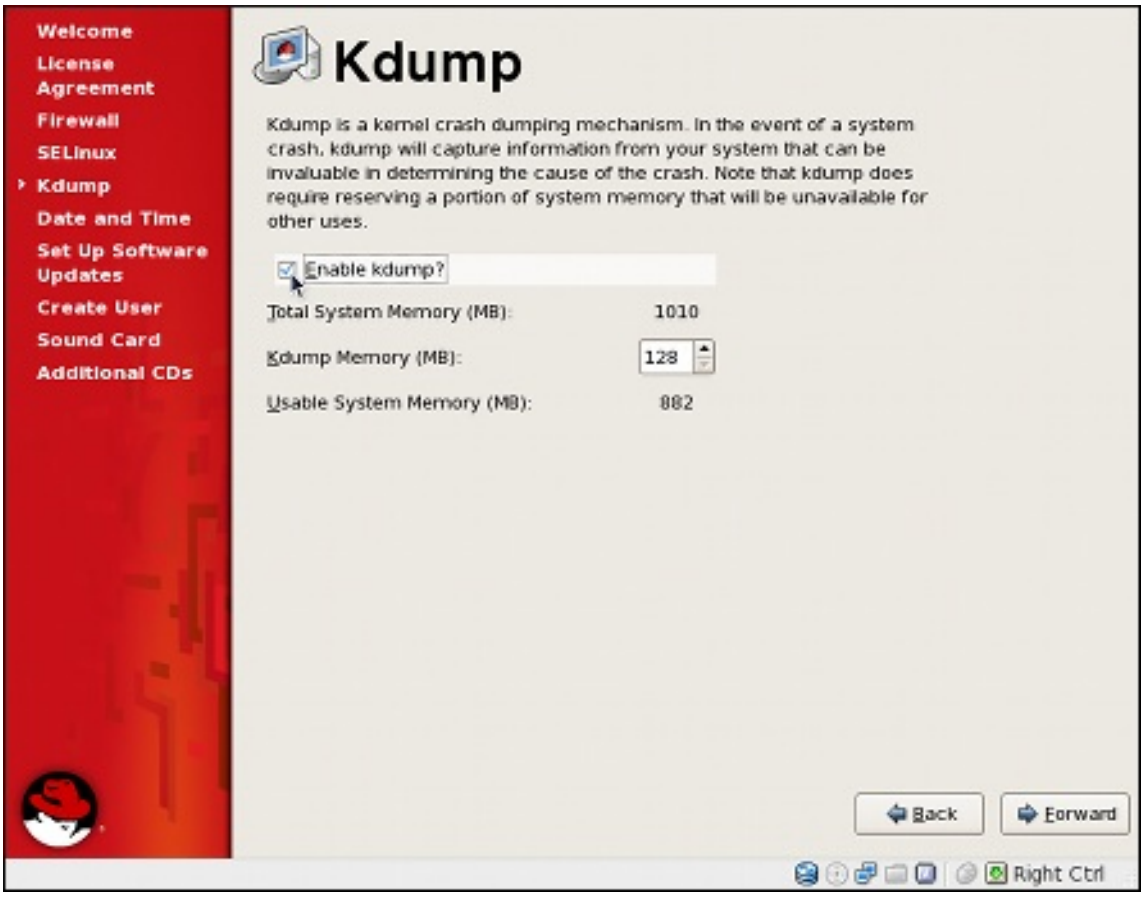
46.2.1. 初回起動時の kdump の設定

システムの初回起動時には、firstboot アプリケーションが起動し、新規インストールしたシステムの初期設定を行います。kdump を設定するには、Kdump ページに移動し、以下の手順に従います。

重要

システムに十分なメモリーがない場合は、Kdump ページは利用できません。最小メモリー要件については、Red Hat Enterprise Linux の比較チャート を参照してください。kdump クラッシュリカバリーを有効にすると、最小メモリー要件が予約されたメモリー量によって増加します。この値は、ユーザーおよび x86 アーキテクチャー、AMD64 アーキテクチャー、および Intel 64 アーキテクチャーでは、物理メモリーの TB ごとに 128 MB と 64 MB （つまり、物理メモリーが 1 TB のシステムの合計 192 MB）にデフォルト設定されます。

図46.1 kdump 設定画面



[D]

46.2.1.1. サービスの有効化

システムの起動時に kdump デーモンを起動するには、kdump を有効 にする チェックボックスを選択します。これにより、ランレベル 2、3、4、および 5 の サービスが有効になり、現行セッションで起動されます。同様に、チェックボックスを消去すると、すべてのランレベルでそのチェックボックスが無効になり、サービスがすぐに停止されます。

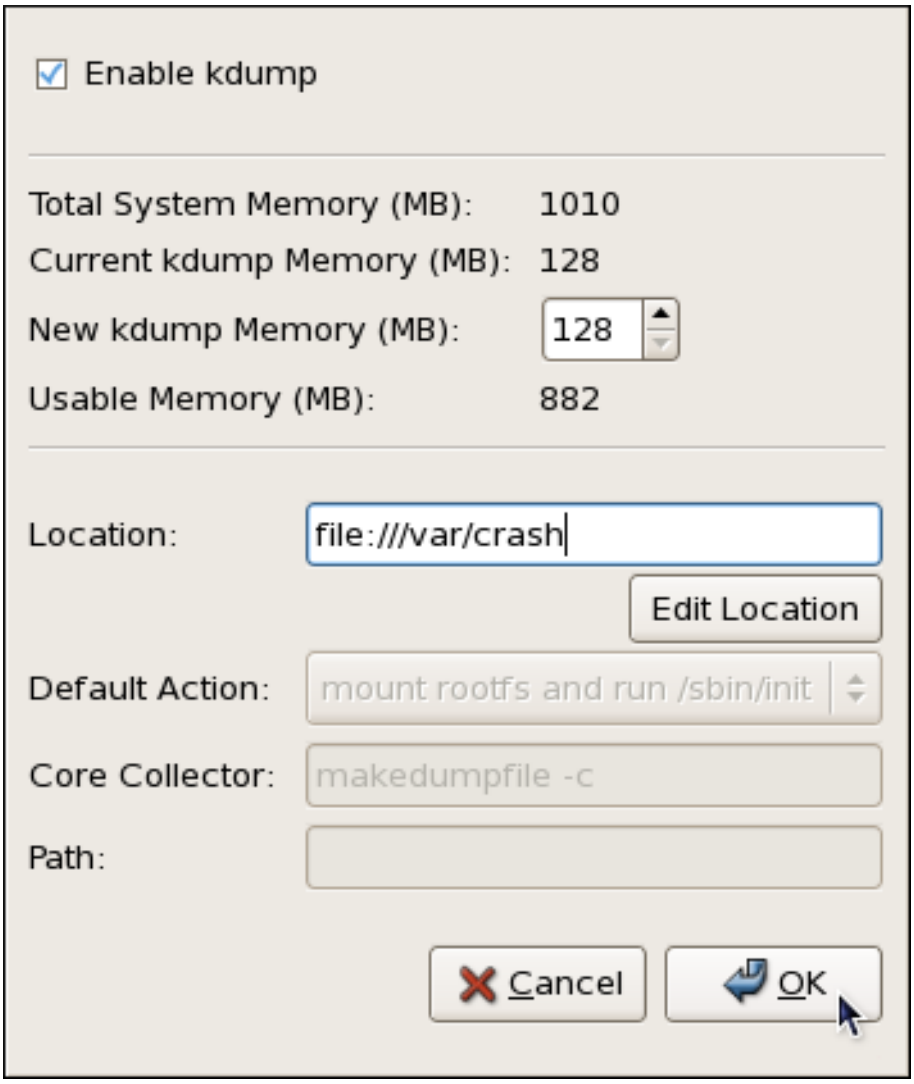
46.2.1.2. メモリー使用量の設定

kdump カーネル用に予約されているメモリー量を設定するには、Kdump Memory フィールドの横にある上下の矢印ボタンをクリックして値を増減します。システムメモリー フィールドはそれに応じて変更され、システムで 使用 できる残りのメモリーが表示されます。

46.2.2. カーネルダンプ設定ユーティリティーの使用

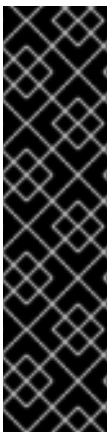
Kernel Dump Configuration ユーティリティーを起動するには、パネルから Applications → System Tools → Kdump を選択するか、シェルプロンプトで system-config-kdump と入力します。すでに認証されていない限り、root パスワードを入力するように求められます。

図46.2 Kernel Dump 設定ユーティリティー



[D]

このユーティリティーを使用すると、kdump を設定し、システムの起動時にサービスを有効または無効にすることができます。完了したら、OK をクリックして変更を保存します。システムの再起動が要求されます。



重要

システムに十分なメモリーがない場合、Kernel Dump Configuration ユーティリティーが起動しず、エラーメッセージが表示されます。最小メモリー要件については、Red Hat Enterprise Linux の比較チャート を参照してください。kdump クラッシュリカバリーを有効にすると、最小メモリー要件が予約されたメモリー量によって増加します。この値は、ユーザーおよび x86 アーキテクチャー、AMD64 アーキテクチャー、および Intel 64 アーキテクチャーでは、物理メモリーの TB ごとに 128 MB と 64 MB （つまり、物理メモリーが 1 TB のシステムの合計 192 MB）にデフォルト設定されます。

46.2.2.1. サービスの有効化

システムの起動時に kdump デーモンを起動するには、kdump を有効 にする チェックボックスを選択します。これにより、ランレベル 2、3、4、および 5 の サービスが有効になり、現行セッションで起動されます。同様に、チェックボックスを消去すると、すべてのランレベルでそのチェックボックスが無効になり、サービスがすぐに停止されます。

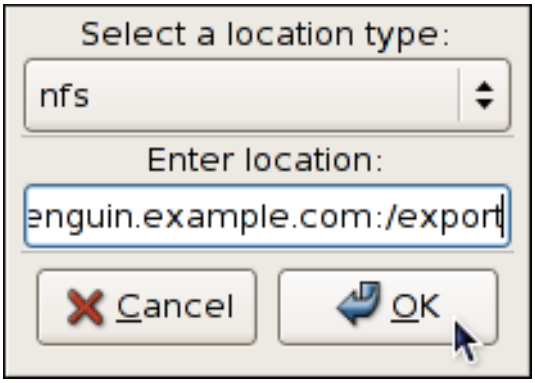
46.2.2.2. メモリー使用量の設定

kdump カーネル用に予約されているメモリー容量を設定するには、New kdump Memory フィールドの横にある上下の矢印ボタンをクリックして値を増減します。システムで使用できるメモリーの残量に応じて Usable Memory フィールドが変わることに注意してください。

46.2.2.3. ターゲットタイプの設定

カーネルクラッシュがキャプチャーされると、コアダンプはローカルファイルシステムのファイルとして保存するか、デバイスに直接書き込むか、NFS (Network File System)プロトコルまたは SSH (Secure Shell)プロトコルを使用してネットワーク経由で送信することができます。これを変更するには、Edit Location ボタンをクリックし、以下で説明されているようにロケーションタイプを選択します。

図46.3 Edit Location ダイアログ



[D]

ローカルファイルシステムにダンプを保存するには、プルダウン一覧から ファイル を選択します。必要に応じて、ファイルを別のパーティションに書き込む場合は、使用しているファイルシステムに応じてプルダウン一覧から ext3 または ext2 を選択し、Enter location フィールドに有効なデバイス名を入力します。OK をクリックした後、下部の Path フィールドの値を変更することで、宛先ディレクトリーをカスタマイズできます。

ダンプをデバイスに直接書き込むには、プルダウン一覧から raw を選択し、有効なデバイス名（例： /dev/sdb1）を入力します。完了したら、OK をクリックして選択を確定します。

NFS プロトコルを使用してダンプをリモートマシンに保存するには、プルダウン一覧から nfs を選択し、hostname:directory フォームに有効なターゲットを入力します（例：

penguin.example.com:/export）。OK ボタンをクリックすると、変更が確定します。最後に、Path フィールドの値を編集して、宛先ディレクトリー（例： cores）をカスタマイズします。

SSH プロトコルを使用してリモートのマシンにダンプを保存するには、プルダウン一覧から ssh を選択し、username@hostname フォームに有効なユーザー名およびホスト名を入力します（例： OK ボタンをクリックすると、変更が確定します。最後に、Path フィールドの値を編集して、宛先ディレクトリー（例： /export/cores）をカスタマイズします。

SSH サーバーの設定方法やキーベースの認証の設定方法は、20章OpenSSH を参照してください。

46.2.2.4. コアコレクターの設定

vmcore ダンプファイルのサイズを縮小するために、kdump では外部アプリケーション（つまりコアコレクター）を指定してデータを圧縮し、必要に応じて関連性のない情報をすべて除外できます。現在、完全にサポートされている唯一のコアコレクターは makedumpfile です。

ダンプファイルの圧縮を有効にするには、Core Collector フィールドの makedumpfile コマンドの後に -c パラメーターがリストされていることを確認します（例： makedumpfile -c）。

ダンプから特定のページを削除するには、Core Collector フィールドの makedumpfile コマンドの後に -d value パラメーターを追加します。この 値 は、表46.1「サポートされるフィルターレベル」で説明されているように、省略するページの値の合計です。たとえば、ゼロページと空きページの両方を削除するには、makedumpfile -d 17 を使用します。

利用可能なオプションの完全なリストは、makedumpfile の man ページを参照してください。

46.2.2.5. デフォルトの動作の変更

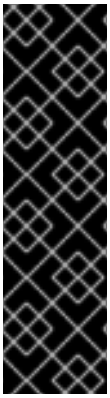
kdump がコアダンプの作成に失敗した場合に実行するアクションを選択するには、デフォルトのアクション プルダウン リストから適切なオプションを選択します。使用可能なオプションは、mount rootfs と /sbin/init （デフォルトの動作）、reboot （システムの再起動）、shell （対話式シェルプロンプトを持つユーザーの表示）、および halt （システムを停止するため）です。

46.2.3. コマンドラインで kdump の設定

46.2.3.1. メモリー使用量の設定

x86、AMD64、および Intel 64 アーキテクチャーで kdump カーネル用に予約されるメモリー容量を設定するには、root で /boot/grub/grub.conf ファイルを開き、crashkernel= <size> M@16M パラ

メーターをカーネルオプションの一覧に追加します(例46.1「/boot/grub/grub.conf ファイルのサンプル」 を参照)。



重要

システムに十分なメモリーがない場合、kdump クラッシュリカバリーサービスは動作しません。最小メモリー要件については、Red Hat Enterprise Linux の比較チャートを参照してください。kdump を有効にすると、最小メモリー要件は、予約されているメモリー量を増やします。この値は、ユーザーおよび x86 アーキテクチャー、AMD64 アーキテクチャー、および Intel 64 アーキテクチャーでは、物理メモリーの TB ごとに 128 MB と 64 MB （つまり、物理メモリーが 1 TB のシステムの合計 192 MB）にデフォルト設定されます。

例46.1 /boot/grub/grub.conf ファイルのサンプル

```
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this file
# NOTICE:  You have a /boot partition.  This means that
#          all kernel and initrd paths are relative to /boot/, eg.
#          root (hd0,0)
#          kernel /vmlinuz-version ro root=/dev/sda3
#          initrd /initrd-version.img
#boot=/dev/sda
default=0
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Red Hat Enterprise Linux Server (2.6.18-274.3.1.el5)
        root (hd0,0)
        kernel /vmlinuz-2.6.18-274.3.1.el5 ro root=/dev/sda3 crashkernel=128M@16M
        initrd /initrd-2.6.18-274.3.1.el5.img
```

46.2.3.2. ターゲットタイプの設定

カーネルクラッシュがキャプチャーされると、コアダンプはローカルファイルシステムのファイルとして保存するか、デバイスに直接書き込むか、NFS (Network File System)プロトコルまたは SSH (Secure Shell)プロトコルを使用してネットワーク経由で送信することができます。現時点で設定できるのは、これらのオプションの 1 つのみであることに注意してください。デフォルトのオプションでは、vmcore ファイルをローカルファイルシステムの /var/crash/ ディレクトリーに保存します。これを変更するには、root で /etc/kdump.conf 設定ファイルを開き、以下のようにオプションを編集します。

コアダンプを保存するローカルディレクトリーを変更するには、「#」path /var/crash の行頭にあるハッシュ記号(#)を取り除き、値を希望のディレクトリーパスに置き換えます。必要に応じて、ファイルを別のパーティションに書き込む場合は、#ext3 /dev/sda3 行でも同じ手順に従い、ファイルシステ

ムタイプとデバイス（デバイス名、ファイルシステムのラベル、UUID がすべてサポートされる）を変更します。以下に例を示します。

```
ext3 /dev/sda4
path /usr/local/cores
```

ダンプをデバイスに直接書き込むには、「#」raw /dev/sda5 の行頭にあるハッシュ記号(#)を取り除き、値を任意のデバイス名に置き換えます。以下に例を示します。

```
raw /dev/sdb1
```

NFS プロトコルを使用してリモートのマシンにダンプを保存するには、「#」net my.server.com:/export/tmp の行頭にあるハッシュ記号(#)を取り除き、値を有効なホスト名とディレクトリーパスに置き換えます。以下に例を示します。

```
net penguin.example.com:/export/cores
```

SSH プロトコルを使用してリモートのマシンにダンプを保存するには、「#」net user@my.server.com の行頭にあるハッシュ記号(#)を取り除き、値を有効なユーザー名およびホスト名に置き換えてください。以下に例を示します。

```
net john@penguin.example.com
```

SSH サーバーの設定方法やキーベースの認証の設定方法は、20章OpenSSH を参照してください。

46.2.3.3. コアコレクターの設定

vmcore ダンプファイルのサイズを縮小するために、kdump では外部アプリケーション（つまりコアコレクター）を指定してデータを圧縮し、必要に応じて関連性のない情報をすべて除外できます。現在、完全にサポートされている唯一のコアコレクターは makedumpfile です。

コアコレクターを有効にするには、/etc/kdump.conf 設定ファイルを root として開き、「#」core_collector makedumpfile -c --message-level 1 の行頭にあるハッシュ記号(#)を取り除き、以下のようにコマンドラインオプションを編集します。

ダンプファイルの圧縮を有効にするには、-c パラメーターを追加します。以下に例を示します。

```
core_collector makedumpfile -c
```

ダンプから特定のページを削除するには、-d valueパラメーターを追加します。value は、表46.1「サポートされるフィルターレベル」 で説明されているように、省略するページの値の合計になります。ゼロと未使用ページを除外する場合は次のようになります。

```
core_collector makedumpfile -d 17 -c
```

利用可能なオプションの完全なリストは、makedumpfile の man ページを参照してください。

表46.1 サポートされるフィルターレベル

オプション	説明
1	ゼロページ
2	キャッシュページ
4	キャッシュプライベート
8	ユーザーページ
16	フリーページ

46.2.3.4. デフォルトの動作の変更

デフォルトでは、kdump がコアダンプの作成に失敗すると、root ファイルシステムがマウントされ、/sbin/init が実行されます。この動作を変更するには、root として /etc/kdump.conf 設定ファイルを開き、#default shell の行頭にあるハッシュ記号(「#」)を取り除き、表46.2「サポートされるアクション」 で説明されているように、値を目的のアクションに置き換えます。以下に例を示します。

```
default halt
```

表46.2 サポートされるアクション

オプション	アクション
reboot	システムを再起動します。プロセスのコアが失われます。
halt	コアの取得に失敗した後に、システムを停止します。
shell	initramfs 内から msh セッションを実行し、ユーザーが手動でコアを記録できるようにします。

46.2.3.5. サービスの有効化

システムの起動時に kdump デーモンを起動するには、シェルプロンプトで root として以下を入力します。

```
~]# chkconfig kdump on
```

これにより、ランレベル 2、3、4、および 5 の サービスが有効になります。同様に、chkconfig kdump off と入力すると、すべてのランレベルで無効になります。現行セッションでサービスを起動するには、root で以下のコマンドを使用します。

```
~]# service kdump start
No kdump initial ramdisk found.                    [WARNING]
Rebuilding /boot/initrd-2.6.18-194.8.1.el5kdump.img
Starting kdump:                                    [  OK  ]
```

ランレベルとサービスの一般的な設定に関する詳細は、18章 を参照してください。

46.2.4. 設定のテスト

警告

以下のコマンドにより、カーネルがクラッシュします。これらの手順に従う場合は注意してください。実稼働マシンで 使用しない は使用しないでください。

設定をテストするには、kdump を有効にしてシステムを再起動し、root でサービスが実行中であることを確認します。

```
~]# service kdump status
Kdump is operational
```

次に、root で次のコマンドを実行します。

```
~]# echo 1 > /proc/sys/kernel/sysrq
~]# echo c > /proc/sysrq-trigger
```

これにより、Linux カーネルを強制的にクラッシュさせ、YYYY-MM-DD-HH:MM/vmcore ファイルが設定で選択した場所（デフォルトでは /var/crash/ ）にコピーされます。

46.3. コアダンプの分析

注記

vmcore ダンプファイルを分析するには、crash パッケージおよび kernel-debuginfo パッケージがインストールされている必要があります。これを行うには、root で次のコマンドを実行します。

```
~]# yum install --enablerepo=rhel-debuginfo crash kernel-debuginfo
```

Red Hat Enterprise Linux に新しいパッケージをインストールする方法の詳細は、パートII「パッケージ管理」 を参照してください。

システムクラッシュの原因を確認するには、crash ユーティリティーを使用できます。このユーティリティーを使用すると、実行中の Linux システム、および netdump、diskdump、xendump、または kdump によって作成されたコアダンプを対話的に分析できます。開始すると、GNU Debugger (GDB) と非常に似ている対話式のプロンプトが表示されます。

シェルプロンプトで次の形式のコマンドを入力してユーティリティーを起動します。

```
crash /var/crash/timestamp/vmcore /usr/lib/debug/lib/modules/kernel/vmlinux
```

カーネル のバージョンは、kdump が取得したバージョンと同じである必要があることに注意してください。現在実行中のカーネルを確認するには、uname -r コマンドを使用します。

例46.2 crash ユーティリティーの実行

```
~]# crash /var/crash/2010-08-04-17\:55/vmcore \
/usr/lib/debug/lib/modules/2.6.18-194.8.1.el5/vmlinux

crash 4.1.2-4.el5_5.1
Copyright (C) 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009  Red Hat, Inc.
Copyright (C) 2004, 2005, 2006  IBM Corporation
Copyright (C) 1999-2006  Hewlett-Packard Co
Copyright (C) 2005, 2006  Fujitsu Limited
Copyright (C) 2006, 2007  VA Linux Systems Japan K.K.
Copyright (C) 2005  NEC Corporation
Copyright (C) 1999, 2002, 2007  Silicon Graphics, Inc.
Copyright (C) 1999, 2000, 2001, 2002  Mission Critical Linux, Inc.
```

```
This program is free software, covered by the GNU General Public License,
and you are welcome to change it and/or distribute copies of it under
certain conditions.  Enter "help copying" to see the conditions.
This program has absolutely no warranty.  Enter "help warranty" for details.

GNU gdb 6.1
Copyright 2004 Free Software Foundation, Inc.
GDB is free software, covered by the GNU General Public License, and you are
welcome to change it and/or distribute copies of it under certain conditions.
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB.  Type "show warranty" for details.
This GDB was configured as "i686-pc-linux-gnu"...

      KERNEL: /usr/lib/debug/lib/modules/2.6.18-194.8.1.el5/vmlinux
    DUMPFILE: /var/crash/2010-08-04-17:55/vmcore
        CPUS: 1
        DATE: Wed Aug  4 17:50:41 2010
      UPTIME: 00:56:53
LOAD AVERAGE: 0.47, 0.47, 0.55
       TASKS: 128
    NODENAME: localhost.localdomain
     RELEASE: 2.6.18-194.el5
     VERSION: #1 SMP Tue Mar 16 21:52:43 EDT 2010
     MACHINE: i686  (2702 Mhz)
      MEMORY: 1 GB
       PANIC: "SysRq : Trigger a crashdump"
         PID: 6042
     COMMAND: "bash"
        TASK: f09c7000  [THREAD_INFO: e1ba9000]
         CPU: 0
       STATE: TASK_RUNNING (SYSRQ)

crash>
```

対話式プロンプトを終了して crash を終了するには、exit と入力します。

46.3.1. メッセージバッファーの表示

カーネルメッセージバッファーを表示するには、対話式プロンプトで log コマンドを入力します。

例46.3 カーネルメッセージバッファーの表示

```
crash> log
Linux version 2.6.18-194.el5 (mockbuild@x86-007.build.bos.redhat.com) (gcc version 4.1.2
20080704 (Red Hat 4.1.2-48)) #1 SMP Tue Mar 16 21:52:43 EDT 2010
BIOS-provided physical RAM map:
 BIOS-e820: 0000000000010000 - 000000000009fc00 (usable)
 BIOS-e820: 000000000009fc00 - 00000000000a0000 (reserved)
 BIOS-e820: 00000000000f0000 - 0000000000100000 (reserved)
 BIOS-e820: 0000000000100000 - 000000003fff0000 (usable)
```

```
 BIOS-e820: 000000003fff0000 - 0000000040000000 (ACPI data)
 BIOS-e820: 00000000fffc0000 - 0000000100000000 (reserved)
127MB HIGHMEM available.
896MB LOWMEM available.
Using x86 segment limits to approximate NX protection
On node 0 totalpages: 262128
  DMA zone: 4096 pages, LIFO batch:0
  Normal zone: 225280 pages, LIFO batch:31
  HighMem zone: 32752 pages, LIFO batch:7
DMI 2.5 present.
Using APIC driver default
... several lines omitted ...
SysRq : Trigger a crashdump
```

このコマンドの使用方法についての詳しい情報を参照するには、help log を入力します。

46.3.2. バックトレースの表示

カーネルスタックトレースを表示するには、対話式プロンプトで bt コマンドを入力します。bt pid を使用すると、選択したプロセスのバックトレースを表示できます。

例46.4 カーネルスタックトレースの表示

```
crash> bt
PID: 6042   TASK: f09c7000  CPU: 0   COMMAND: "bash"
 #0 [e1ba9d10] schedule at c061c738
 #1 [e1ba9d28] netlink_getsockopt at c05d50bb
 #2 [e1ba9d34] netlink_queue_skip at c05d40d5
 #3 [e1ba9d40] netlink_sock_destruct at c05d506d
 #4 [e1ba9d84] sock_recvmsg at c05b6cc8
 #5 [e1ba9dd4] enqueue_task at c041eed5
 #6 [e1ba9dec] try_to_wake_up at c041f798
 #7 [e1ba9e10] vsnprintf at c04efef2
 #8 [e1ba9ec0] machine_kexec at c0419bf0
 #9 [e1ba9f04] sys_kexec_load at c04448a1
#10 [e1ba9f4c] tty_audit_exit at c0549f06
#11 [e1ba9f50] tty_audit_add_data at c0549d5d
#12 [e1ba9f84] do_readv_writev at c0476055
#13 [e1ba9fb8] system_call at c0404f10
    EAX: fffffffda  EBX: 00000001  ECX: b7f7f000  EDX: 00000002
    DS:  007b       ESI: 00000002  ES:  007b      EDI: b7f7f000
    SS:  007b       ESP: bf83f478  EBP: bf83f498
    CS:  0073       EIP: 009ac402  ERR: 00000004  EFLAGS: 00000246
```

このコマンドの使用方法についての詳しい情報を表示するには、help bt と入力します。

46.3.3. プロセスステータスの表示

対話式プロンプトで **ps** コマンドを入力してシステム内のプロセスの状態を表示します。**ps pid** を使用すると、選択したプロセスのステータスを表示できます。

例46.5 システム内のプロセスの状態の表示

```
crash> ps
  PID    PPID  CPU   TASK    ST  %MEM     VSZ    RSS  COMM
    0      0   0  c068a3c0  RU   0.0       0      0  [swapper]
    1      0   0  f7c81aa0  IN   0.1    2152    616  init
... several lines omitted ...
  6017      1   0  e39f6550  IN   1.2   40200  13000  gnome-terminal
  6019   6017   0  e39f6000  IN   0.1    2568    708  gnome-pty-helpe
  6020   6017   0  f0421550  IN   0.1    4620   1480  bash
  6021      1   0  f7f69aa0  ??   1.2   40200  13000  gnome-terminal
  6039   6020   0  e7e84aa0  IN   0.1    5004   1300  su
> 6042   6039   0  f09c7000  RU   0.1    4620   1464  bash
```

このコマンドの使用方法についての詳しい情報を参照するには、**help ps** を入力します。

46.3.4. 仮想メモリー情報の表示

基本的な仮想メモリー情報を表示するには、対話式プロンプトで **vm** コマンドを入力します。**vm pid** を使用すると、選択したプロセスの情報を表示できます。

例46.6 現在のコンテキストの仮想メモリー情報の表示

```
crash> vm
PID: 6042   TASK: f09c7000  CPU: 0   COMMAND: "bash"
  MM       PGD      RSS    TOTAL_VM
e275ee40  e2b08000  1464k    4620k
  VMA       START      END    FLAGS  FILE
e315d764    1fe000    201000     75  /lib/libtermcap.so.2.0.8
e315de9c    201000    202000 100073  /lib/libtermcap.so.2.0.8
c9b040d4    318000    46a000     75  /lib/libc-2.5.so
e315da04    46a000    46c000 100071  /lib/libc-2.5.so
e315d7b8    46c000    46d000 100073  /lib/libc-2.5.so
e315de48    46d000    470000 100073
e315dba8    9ac000    9ad000 8040075
c9b04a04    a2f000    a4a000    875  /lib/ld-2.5.so
c9b04374    a4a000    a4b000 100871  /lib/ld-2.5.so
e315d6bc    a4b000    a4c000 100873  /lib/ld-2.5.so
e315d908    fa1000    fa4000     75  /lib/libdl-2.5.so
e315db00    fa4000    fa5000 100071  /lib/libdl-2.5.so
e315df44    fa5000    fa6000 100073  /lib/libdl-2.5.so
e315d320    ff0000    ffa000     75  /lib/libnss_files-2.5.so
e315d668    ffa000    ffb000 100071  /lib/libnss_files-2.5.so
```

```
e315def0    ffb000    ffc000 100073  /lib/libnss_files-2.5.so
e315d374   8048000   80f5000   1875  /bin/bash
c9b045c0   80f5000   80fa000 101873  /bin/bash
... several lines omitted ...
```

このコマンドの使用方法についての詳しい情報を参照するには、help vm と入力してください。

46.3.5. 開いているファイルの表示

対話式プロンプトで files コマンドを入力してオープン ファイル に関する情報を表示します。files pid を使用して、選択したプロセスで開いているファイルを表示できます。

例46.7 現在のコンテキストのオープンファイルについての情報の表示

```
crash> files
PID: 6042   TASK: f09c7000  CPU: 0   COMMAND: "bash"
ROOT: /    CWD: /root
 FD    FILE     DENTRY    INODE    TYPE  PATH
  0  e33be480  e609bf70  f0e1d880  CHR   /dev/pts/1
  1  e424d8c0  d637add8  f7809b78  REG   /proc/sysrq-trigger
  2  e33be480  e609bf70  f0e1d880  CHR   /dev/pts/1
 10  e33be480  e609bf70  f0e1d880  CHR   /dev/pts/1
255  e33be480  e609bf70  f0e1d880  CHR   /dev/pts/1
```

このコマンドの使用方法についての詳しい情報を参照するには、help files と入力します。

46.4. 関連情報

46.4.1. インストールされているドキュメント

man kdump.conf

利用可能なオプションの詳細なドキュメントを含む /etc/kdump.conf 設定ファイルの man ページです。

man kexec

kexec の man ページには、その使用方法に関する詳細なドキュメントが含まれています。

man クラッシュ

crash ユーティリティーの man ページには、その使用方法に関する完全なドキュメントが含まれています。

/usr/share/doc/kexec-tools-version/kexec-kdump-howto.txt

kdump および kexec のインストールと使用の概要。

46.4.2. 便利な Web サイト

https://access.redhat.com/kb/docs/DOC-6039

kexec および kdump 設定に関する Red Hat ナレッジベースアーティクルです。

http://people.redhat.com/anderson/

crash ユーティリティーのホームページです。

パート VII. セキュリティーおよび認証

システム管理者がミッションクリティカルなシステム、サービス、またはデータを保護する必要があるかどうかに関係なく、Red Hat Enterprise Linux は包括的なセキュリティー戦略の一部として機能するさまざまなツールおよび方法を提供します。

本章では、セキュリティーの概要と、特に Red Hat Enterprise Linux の観点から説明します。セキュリティー評価、一般的な不正使用、侵入およびインシデント対応領域における概念的な情報を提供します。また、SELinux を使用してワークステーション、サーバー、VPN、ファイアウォールなどの実装を強化する方法についての概念および具体的な設定情報も提供します。

本章では、IT セキュリティーに関する基本的な知識を想定しています。したがって、物理アクセスの制御、サウンドアカウント管理ポリシーおよび手順、監査など、一般的なセキュリティープラクティスの最小範囲のみを説明します。必要に応じて、この情報および関連情報の外部リソースへの参照が行われます。

第47章 セキュリティーの概要

ビジネスの実行や個人情報の追跡に役立つ強力なネットワークコンピューターへの依存度が高まるため、業界はネットワークおよびコンピューターのセキュリティーの実践に重点を置いています。企業では、システムを適切に監査し、組織の運用要件を満たすようにソリューションを調整するために、セキュリティーのエキスパートの知識とスキルを確保しています。ほとんどの組織は本質的に動的であるため、企業の IT リソースにローカルおよびリモートからアクセスするワーカーでは、セキュアなコンピューティング環境の必要性がわかりやすくなりました。

残念ながら、ほとんどの組織（個別ユーザーも含む）は、費用、生産性、および予算面の懸念が高まるため、セキュリティーを後で検討するプロセスについて検討しています。多くの場合、適切なセキュリティー実装は、不正な侵入が発生した後、延期された後です。セキュリティーのエキスパートは、インターネットなどの信頼できないネットワークにサイトを接続する前に実施した適切な措置が、侵入でほとんどの試みを阻害する効果的な手段であることに同意します。

47.1. セキュリティーの概要

47.1.1. コンピューターセキュリティーとは

コンピューターセキュリティーは、コンピューティングと情報処理の幅広い分野で使用される一般的な用語です。コンピューターシステムとネットワークに依存して日々のビジネストランザクションを実施し、重要な情報にアクセスする業界は、データをアセット全体の重要な部分と見なします。期間やメトリクスには、毎日のビジネス用語に入りました。たとえば、所有者コスト(contact)や QoS (Quality of Service)など、数々のビジネス用語に入りました。これらのメトリクスでは、計画およびプロセス管理コストの一部として、データの整合性や高可用性などの側面を計算します。電子商取引などの業界では、データの可用性と信頼性は、成功と失敗の違いです。

47.1.1.1. How did Computer Security Come about?

情報セキュリティーは、個人、銀行、およびその他の制限された情報を開示しないパブリックネットワークへの依存度が高まるため、長年にわたって進化しています。Mitnick や Vladimir Levin case など、多くのインスタンスがあり、あらゆる業界で組織を要求して、情報の送信や公開の処理方法を再考します。インターネットの人気は、データセキュリティーの意図的な作業を求める最も重要な開発の 1 つでした。

インターネットが提供するリソースにアクセスするために、個人のコンピューターを使用しているユーザーの数が増えています。調査や情報の取得から電子メールやコマース取引まで、インターネットは 20 位のうちの最も重要な開発の 1 つとして考慮されました。

ただし、インターネットとそれ以前のプロトコルは、信頼ベース のシステムとして開発されました。つまり、インターネットプロトコル自体は保護されるように設計されていません。TCP/IP 通信スタックに組み込まれた承認済みのセキュリティー標準はなく、ネットワーク全体で悪意のあるユーザー

やプロセスに開くことができます。現代の開発によりインターネット通信がより安全になりましたが、国内的な注意を集め、完全に安全ではないという事実を警告するインシデントがいくつかあります。

47.1.1.2. Security Today

2000 年 2 月に、DDoS (Distributed Denial of Service)攻撃は、インターネット上で最も負荷の高い複数のサイトに無駄になりました。攻撃は yahoo.com、cnn.com、amazon.com、fbi.gov、およびその他のサイトは完全に到達不能な状態になりました。これは、通常ユーザーに完全に到達不能になります。これは、大きなバイトの ICMP パケット転送でルーターを 1 時間関連付けています(ping フラッド とも呼ばれます)。この攻撃は、脆弱なネットワークサーバーをスキャンし、サーバーに Trojans と呼ばれるクライアントアプリケーションをスキャンして、特殊に作成された未知のプログラムを使用して、未知の攻撃を受けました。また、検出したすべてのサーバーで攻撃に時間をかけ、その攻撃を使用できなくなっていました。パケットの送信先や送信目的に関係なく、ルーターとプロトコルがすべての受信データを受け入れるように構造化されている方法の基本的な欠陥に対する多くの攻撃を緩和します。

現在、約 9.45億人のユーザーがインターネットを使用または使用しているか(Computer Industry Almanac、2004 年)同時に、以下を行います。

- いずれの日も、Carnegie Mellonvideo の CERT Coordination Center にレポートされたセキュリティー侵害の約 225 大まかな欠如があります。[10]
- 2003 年、CERT の数が、2002 年の 82,094 から、2001 年に 52,658 から 137,529 にジャンプした CERT の数。[11]
- 過去 3 年間で危険な 3 つのインターネットベクトルによる世界的な影響は、US$13.2 Billion で推定されました。[12]

コンピューターセキュリティーは、すべての IT 予算の数量で正当な費用になりました。データの整合性と高可用性を必要とする組織は、システム管理者、開発者、エンジニアがシステム、サービス、および情報の 24 時間の信頼性を確保するためのスキルの恩恵を受けています。悪意のあるユーザー、プロセス、またはコーディネート攻撃の違反は、組織の成功に対する直接の脅威です。

残念ながら、システムおよびネットワークのセキュリティーは、組織がどのように情報を使用、使用、操作、送信するかについての詳しい知識を必要とするため、システムおよびネットワークのセキュリティーが困難になる場合があります。組織（および組織を設定するユーザー）がビジネスを行う方法を理解することは、適切なセキュリティー計画を実装するのに最も適しています。

47.1.1.3. セキュリティーの標準化

すべての業界で、米国政府の Association (AMA)、Electrical and Electronics Engineers (IEEE) などの標準によって設定された規制やルールに依存します。情報セキュリティーにも同じことが言えます。多くのセキュリティーコンサルタントやベンダーが 機密性 (Confidentiality)、保全性 (Integrity)、可用性 (Availability) の頭文字をとった CIA として知られる標準セキュリティーモデルを採用しています。この 3 階層モデルは、機密情報のリスク評価やセキュリティー方針の確立において、一般的に採用されているモデルです。以下でこの CIA モデルを説明します。

- 機密性 - 機密情報は、事前に定義された個人だけが利用できるようにする必要があります。許可されていない情報の送信や使用は、制限する必要があります。たとえば、情報に機密性があれば、権限のない個人が顧客情報や財務情報を悪意のある目的 (ID 盗難やクレジット詐欺など) で入手できません。

- 保全性 - 情報は、改ざんして不完全または不正確なものにすべきではありません。承認されていないユーザーが、機密情報を変更したり破壊したりする機能を使用できないように制限する必要があります。

- 可用性 - 情報は、認証されたユーザーが必要な時にいつでもアクセスできるようにする必要があります。可用性は、合意した頻度とタイミングで情報を入手できることを保証します。これは、パーセンテージで表されることが多く、ネットワークサービスプロバイダーやその企業顧客が使用するサービスレベルアグリーメント (SLA) で正式に合意となります。

47.1.2. セキュリティーコントロール

多くの場合、コンピューターセキュリティーは、一般に コントロール と呼ばれる以下の 3 つのマスターカテゴリーに分類されます。

- 物理的

- 技術的

- 管理的

この 3 つのカテゴリーは、セキュリティーの適切な実施における主な目的を定義するものです。このコントロールには、コントロールと、その実装方法を詳細化するサブカテゴリーがあります。

47.1.2.1. 物理的コントロール

物理的コントロールは、機密資料への非認証アクセスの抑止または防止のために、明確な構造でセキュリティー対策を実施します。物理的コントロールの例は以下のとおりです。

- 有線監視カメラ
- 動作または温度の感知アラームシステム
- 警備員
- 写真付き身分証明書
- 施錠された、デッドボルト付きのスチールドア
- バイオメトリクス (本人確認を行うための指紋、声、顔、虹彩、筆跡などの自動認識方法が含まれます)

47.1.2.2. 技術的コントロール

技術的コントロールでは、物理的な構造物やネットワークにおける機密データのアクセスや使用を制御する基盤となる技術を使用します。技術的コントロールは広範囲に及び、以下のような技術も含まれます。

- 暗号化
- スマートカード
- ネットワーク認証
- アクセス制御リスト (ACL)
- ファイルの完全性監査ソフトウェア

47.1.2.3. 管理的コントロール

管理的コントロールは、セキュリティーの人的要素を定義します。これには、組織内のすべてのパーソナルレベルが含まれ、以下のようにどのユーザーがどのリソースや情報にアクセスできるかを決定します。

- トレーニングおよび認識の向上
- 災害準備および復旧計画
- 人員採用と分離の戦略
- 人員登録とアカウンティング

47.1.3. まとめ

セキュリティーの出所、理由、側面について学んだので、Red Hat Enterprise Linux に関する適切なアクションを決めることができます。適切なストラテジーを計画および実装するために、セキュリティーを設定する要素と条件を理解することが重要です。この情報を念頭に置いて、このプロセスを形成でき、セキュリティープロセスの詳細を把握しておくと、パスが明確になります。

47.2. 脆弱性のアセスメント

時間、リソース、動機があると、クラッカーはほぼすべてのシステムに侵入できます。結局のところ、現在利用可能なすべてのセキュリティー手順と技術は、すべてのシステムが侵入から安全であることを保証することはできません。ルーターは、インターネットへのセキュアなゲートウェイを提供します。ファイアウォールは、ネットワークの境界を保護します。仮想プライベートネットワーク (VPN) では、データが、暗号化されているストリームで安全に通過できます。侵入検知システムは、悪意のある活動を警告します。しかし、これらの技術が成功するかどうかは、以下のような数多くの要因によって決まります。

- 技術の設定、監視、および保守を行うスタッフの専門知識
- サービスとカーネルのパッチ、および更新を迅速かつ効率的に行う能力
- ネットワーク上での警戒を常に怠らない担当者の能力

データシステムと各種技術が動的であることを考えると、企業リソースを保護するタスクは極めて複雑になる可能性もあります。この複雑さゆえに、使用するすべてのシステムの専門家を見つけることは、多くの場合困難になります。情報セキュリティーの多くの分野によく精通している人材を確保することはできても、多くの分野を専門とするスタッフを確保することは容易ではありません。これは、情報セキュリティーの各専門分野で、継続的な注意と重点が必要となるためです。情報セキュリティーは、常に変化しています。

47.2.1. 不利な点を考える

エンタープライズネットワークを管理すると仮定します。このようなネットワークは、一般的にオペレーティングシステム、アプリケーション、サーバー、ネットワークモニター、ファイアウォール、侵入検知システムなどで設定されています。次に、これらの各項目を最新に維持してみてください。現在のソフトウェアおよびネットワーク環境が複雑であるため、悪用とバグはある程度のものです。ネットワーク全体でパッチや更新を最新の状態に維持すると、異種システムを使用する大規模な組織では難易な作業になります。

専門知識の要件と現在の状態を維持するタスクを組み合わせて、インシデントの発生、システムの侵害、データが破損し、サービスが中断される可能性は軽減されます。

セキュリティーテクノロジーとシステム、ネットワーク、およびデータの保護を支援するために、弱点を確認してシステムのセキュリティーを判断する必要があります。独自のシステムおよびネットワークリソースに対する予防的な脆弱性アセスメントにより、クラッカーが悪用される前に対処できる潜在的な問題を特定することができます。

脆弱性アセスメントは、ネットワークおよびシステムセキュリティーの内部監査です。ここで、ネットワークの機密性、整合性、および可用性を示す結果(「セキュリティーの標準化」で説明)。通常、脆弱性アセスメントは、対象システムとリソースに関する重要なデータを収集する調査フェーズから開始します。その後システム準備フェーズとなります。基本的にこのフェーズでは、対象を絞り、すべての既知の脆弱性を調べます。readiness フェーズでは、レポートフェーズで評価され、その結果は高、中、低リスクのカテゴリーに分類され、ターゲットのセキュリティーを強化する（または脆弱性のリスクを軽減する）方法が説明されます。

たとえば、自宅の脆弱性アセスメントを実施することを想定してみましょう。まずは自宅のドアを点検し、各ドアが閉まっていて、かつ施錠されていることを確認します。また、すべての窓が完全に閉まっていて鍵が閉まっていることも確認します。これと同じ概念が、システム、ネットワーク、および電子データにも適用されます。悪意のあるユーザーはデータを盗んで、破壊します。悪意のあるユーザーが使用するツール、思考、動機に注目すると、彼らの行動にすばやく反応することが可能になります。

47.2.2. アセスメントとテストの定義

脆弱性アセスメントは、外観と内部に の 2 つのタイプ に 分割できます。

脆弱性アセスメントの外観をご利用になれば、外部からシステムに攻撃を試みます。会社を外から見ることで、クラッカーの視点に立つことができます。一般にルーティング可能な IP アドレス、DMZ にあるシステム、ファイアウォールの外部インターフェイスなど、クラッカーが目を付けるものに着目します。DMZ は非武装地帯 (demilitarized zone) を表し、企業のプライベート LAN などの信頼できる内部ネットワークと、公的なインターネットなどの信頼できない外部ネットワークの間にあるコンピューターまたは小さなサブネットワークに相当します。通常、DMZ には、Web (HTTP)サーバー、FTP サーバー、SMTP (e-mail)サーバー、DNS サーバーなどのインターネットトラフィックにアクセスできるデバイスが含まれます。

脆弱性アセスメントの内部を見ると、内部にあり、ステータスが信頼できるため、メリットがあります。内部からの視点は、実行者やその同僚がシステムにログオンした時点で得られるものです。プリントサーバー、ファイルサーバー、データベースなどのリソースを見ることができます。

これら 2 種類の脆弱性アセスメントには大きな違いがあります。会社の内部により、外部の権限よりも多くの権限が昇格されます。現在、ほとんどの組織では、侵入者に欠けるようにセキュリティーが設定されています。組織の内部（部門ファイアウォール、ユーザーレベルのアクセス制御、内部リソースに対する認証手順など）を保護するために、ほとんど行われていません。また、一般的にほとんどのシステムは社内にあるため、内部からの方がより多くのリソースを確認できます。会社外で自分で設定すると、すぐに信頼できない状態になります。通常、外部から利用できるシステムやリソースは、非常に限られたものになります。

脆弱性アセスメントと 侵入テスト の違いを考えてみましょう。脆弱性アセスメントを、侵入テストの第一歩と捉えてください。このアセスメントで得られる情報は、その後のテストで使用します。アセスメントはホールと潜在的な脆弱性をチェックしていますが、侵入テストでは実際に発見を悪用しようとします。

ネットワークインフラストラクチャーのアセスメントは動的なプロセスです。セキュリティー (情報セキュリティーおよび物理的なセキュリティー) は動的なものです。アセスメントを実施することで概要が明らかになり、誤検出 (False positives) および検出漏れ (False negatives) が示される場合があります。

セキュリティー管理者の力量は、使用するツールとその管理者が有する知識で決まります。現在使用できるアセスメントツールのいずれかを選び、それらをシステムに対して実行すると、ほぼ間違いなく誤検出がいくつか見つかります。プログラム障害でもユーザーエラーでも、結果は同じです。このツールは、実際に存在しない脆弱性を見つける可能性があります（誤検出）。または、ツールが実際に存在する脆弱性が検出されない可能性があります（負の値）。

脆弱性アセスメントと侵入テストの違いが定義されたところで、新たなベストプラクティスの一環として侵入テストを実施する前に、アセスメントの結果を注意深く確認し、検討してみましょう。

WARNING

実稼働リソースで脆弱性を悪用しようとすると、システムおよびネットワークの生産性および効率に悪影響を与える可能性があります。

脆弱性アセスメントの実施には、以下のような利点があります。

- *情報セキュリティーにプロアクティブなフォーカスを作成*
- *クラッカーが発見する前に潜在的な不正使用を見つける*
- *システムを最新の状態に保ち、パッチを当てる*
- *スタッフの専門知識開発における成長と支援*
- *Abates Financial loss and negative publicity*

47.2.2.1. メソッドの確立

脆弱性アセスメントの方法論が確立されれば、脆弱性アセスメント用のツール選択に役立ちます。現時点では、事前定義の方法論や業界で承認された方法論はありませんが、一般常識やベストプラクティスを適切なガイドとして活用できます。

**ターゲットとは何を指していますか ?1 台のサーバー、またはネットワーク全体およびネットワーク内にあるすべてのサーバーを確認しますか ?会社外ですか ? それとも内部ですか ? *この質問に対する回答は、選択したツールだけでなく、そのツールの使用方法を決定する際に重要です。*

方法論の確立に関する詳細は、以下の Web サイトを参照してください。

- *http://www.isecom.org/projects/osstmm.htm* **『Open Source Security Testing Methodology Manual』 *(OSSTMM)***

- http://www.owasp.org/ 『Open Web Application Security プロジェクト』

47.2.3. ツールの評価

アセスメントは、情報収集ツールを使用することで開始できます。ネットワーク全体を評価する際は、最初にレイアウトを描いて、稼働しているホストを把握します。ホストの場所を確認したら、それぞれのホストを個別に検査します。各ホストにフォーカスするには別のツールセットが必要になります。どのツールを使用すべきかを知っておくことは、脆弱性の発見において最も重要なステップになる可能性があります。

日常生活のあらゆる状況と同様に、同じジョブを実行できる異なるツールは数多くあります。この概念は脆弱性アセスメントの実施にも当てはまります。ツールには、オペレーティングシステムやアプリケーションに固有のものや、(使用されるプロトコルに基づいて) ネットワークに固有のツールもあります。無料のツールも、有料のツールもあります。直感的で使いやすいツールもあれば、不可解で文書化が不十分で、かつ他のツールにはない機能を備えているツールもあります。

適切なツールを見つけることは、難易度の高いタスクであり、最終的には経験が多くなる場合があります。可能であれば、テストラボを立ち上げて、できるだけ多くのツールを試し、それぞれの長所と短所に注意してみてください。ツールの README ファイルまたは man ページを確認してください。さらに、インターネットで、記事、ステップバイステップのガイド、ツールに固有のメーリングリストなどの詳細について検討してください。

以下に説明するツールは、利用可能なツールのほんの一部です。

47.2.3.1. Nmap を使用したホストのスキャン

Nmap は、Red Hat Enterprise Linux に含まれる一般的なツールで、ネットワークのレイアウトを決定するために使用できます。Nmap は長年にわたって利用でき、情報の収集時におそらく最もよく使われるツールです。オプションと使用方法の詳細な説明を示す優れた man ページが含まれています。管理者は、ネットワーク上で Nmap を使用してホストシステムを見つけ、それらのシステムでポートを開くことができます。

Nmap は、脆弱性アセスメントにおける最初のステップです。ネットワーク内のすべてのホストをマップし、Nmap が特定のホストで実行しているオペレーティングシステムの識別を試みるオプションを渡すこともできます。Nmap は、セキュアなサービスを使用してポリシーを確立し、未使用のサービスを停止するのに適した基盤です。

47.2.3.1.1. Nmap の使用

Nmap はシェルプロンプトから実行できます。これには、nmap コマンドの後にスキャンするマシンのホスト名または IP アドレスを入力します。

```
nmap foo.example.com
```

スキャンの結果（ホストの場所によっては数分かかる場合があります）は、以下のようになります。

```
Starting nmap V. 3.50 ( www.insecure.org/nmap/ )
Interesting ports on localhost.localdomain (127.0.0.1):
(The 1591 ports scanned but not shown below are in state: closed)
Port       State       Service
22/tcp     open        ssh
25/tcp     open        smtp
111/tcp    open        sunrpc
443/tcp    open        https
515/tcp    open        printer
950/tcp    open        oftep-rpc
6000/tcp   open        X11

Nmap run completed -- 1 IP address (1 host up) scanned in 71.825 seconds
```

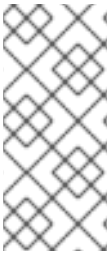
Nmap は、リスニングまたは待機中のサービスの最も一般的なネットワーク通信ポートをテストします。この知識は、不要なサービスや未使用のサービスを閉じる管理者に役立ちます。

Nmap の使用に関する詳細は、以下の URL の公式ホームページを参照してください。

http://www.insecure.org/

47.2.3.2. Nessus

Nessus はフルサービスのセキュリティースキャナーです。Nessus のプラグインアーキテクチャーにより、ユーザーはシステムおよびネットワーク用にカスタマイズできます。スキャナーと同様に、Nessus は依存する署名データベースと同じくらい優れています。幸いなことに、Nessus は頻繁に更新され、完全なレポート作成、ホストスキャン、およびリアルタイムの脆弱性検索が含まれます。Nessus として頻繁に更新される強力なツールであっても、誤検出や誤検出が発生する可能性があることに注意してください。



注記

Nessus は Red Hat Enterprise Linux に含まれていないため、サポートされていません。これは、この人気のあるアプリケーションの使用に興味があるかもしれないユーザーへの参照として、このドキュメントに含まれています。

Nessus の詳細は、以下の URL にある公式 Web サイトを参照してください。

http://www.nessus.org/

47.2.3.3. Nikto

Nikto は、優れた Common Gateway Interface (CGI)スクリプトスキャナーです。Nikto は、CGI の脆弱性を確認するだけでなく、侵入検知システムを廃止するための確率的な方法でチェックを行います。これには詳細なドキュメントが含まれており、プログラムを実行する前に入念に確認する必要があります。CGI スクリプトを提供する Web サーバーがある場合、Nikto はこれらのサーバーのセキュリティーを確認する優れたリソースになります。

注記

Nikto は Red Hat Enterprise Linux に含まれていないため、サポートされていません。これは、この人気のあるアプリケーションの使用に興味があるかもしれないユーザーへの参照として、このドキュメントに含まれています。

Nikto の詳細については、以下の URL を参照してください。

http://www.cirt.net/code/nikto.shtml

47.2.3.4. VLAD the Scanner

VLAD は、Bindview, Inc. の RAZOR チームによって開発された脆弱性スキャナーで、一般的なセキュリティー問題(SNMP の問題、ファイル共有の問題など)の SANS Top Ten リストを確認します。フル機能の Nessus ではありませんが、VLAD は調査する価値があります。

注記

VLAD は Red Hat Enterprise Linux に含まれていないため、サポートされていません。これは、この人気のあるアプリケーションの使用に興味があるかもしれないユーザーへの参照として、このドキュメントに含まれています。

VLAD の詳細は、以下の URL の RAZOR チーム Web サイトを参照してください。

http://www.bindview.com/Support/Razor/Utilities/

47.2.3.5. 将来のニーズの予測

ターゲットとリソースに応じて、利用できるツールが多数あります。ワイヤレスネットワーク、Novell ネットワーク、Windows システム、Linux システムなどのツールがあります。評価実行のもう 1 つの重要な部分には、物理セキュリティー、人事スクリーニング、または音声/PBX ネットワーク評価の確認などが含まれます。ワイヤレスネットワークの脆弱性に対する企業の物理構造のスキャンを行うウォーターのウォーターを ウォークスティング するなど、新しい概念は、調査可能な概念であり、必要に応じて評価に組み込まれます。脆弱性アセスメントの計画と実施の限度が限ってきます。

47.3. 攻撃者および脆弱性

適切なセキュリティーストラテジーを計画して実装するには、まず判断された問題の一部を認識し、攻撃を悪用してシステムの不正使用に悪用します。ただし、これらの問題の詳細を説明する前に、攻撃者が特定する際に使用される用語を定義する必要があります。

47.3.1. ハッカーのクイック履歴

ハッカー の現代の意味は、1960 秒、Massachusetts Institute of Technology (MIT) Tech Model Railroad Club (Massachusetts Institute of Technology) Tech Model Railroad Club (Massachusetts Institute of Technology)および Massachusetts Institute of Technology (MIT) Tech Model Railroad Club (Massachusetts Institute of Technology) Tech Model Railroad ClubHacker は、問題の明確さまたは回避策を発見した club メンバーに使用される名前でした。

ハッカーという用語は、コンピューターの buffs からギフト化されたプログラマーまで、すべてを記述したものでした。ほとんどのハッカーに共通する特性は、コンピューターのシステムとネットワークがどのように外っていないかを調べることです。オープンソースソフトウェア開発者は、多くの場合、自身とその同社がハッカーと見なされ、その単語をその用語として使用します。

通常、ハッカーは、情報と専門知識が不可欠であることを指示する ハッカーの ethic 形式に従います。この知識を共有することは、コミュニティーのハッカーに主張していることを指示します。このよ

うな知識を感じている間、コンピューターシステムのセキュリティー制御を回避するという難易度的な課題を楽しんでいます。このため、プレッカーという用語は多くの場合、ハッカーという用語を使用して、不必要な、悪意のある、または criminal intent のシステムとネットワークにアクセスする人を記述します。このタイプのコンピューターハッカーのより正確な用語は クラッカー です。この用語は、2 つのコミュニティーを区別するための中間者によって作られた用語です。

47.3.1.1. Gray の shades

システムおよびネットワークの脆弱性を発見し、悪用する個人のコミュニティーにはいくつかのグループがあります。これらのグループは、多くの場合、セキュリティー調査を行うときに破棄され、この成熟が意図していることを示すため、Red Hat のシェアによって記述されます。

ホワイト リストには、ネットワークおよびシステムをテストしてパフォーマンスを確認し、侵入する脆弱性を判断する人です。通常、ホワイトリストは、独自のシステム、またはセキュリティー監査の目的で特別に採用したクライアントのシステムをクラッキングします。アカデミック研究者および専門的なセキュリティー対策は、Red Hat のハッカーの 2 つの例です。

ブラックアハッカー は、クラッカーとの同義語です。一般的に、クラッカーは、プログラミングやシステムに分割するアカデミック側に重点が置かれていません。多くの場合、システムで利用可能なクラッキングプログラムや、システムの既知の脆弱性を利用して、ターゲットシステムまたはネットワークで個人による損傷や損害の損傷など、機密情報を発見します。

一方、グレーのRed Hatハッカー は、ほとんどの状況でホワイトRed Hatハッカーのスキルとインテンシャルを特長としていますが、自分の知識を短時間で利用することができます。グレーの Red Hat ハッカーは、自分のアジェンダを達成するためにブラックアを望んで、ブラックスウォーライザーと見なすことができます。

グレーのハッカーは、通常、ハッカーのイーサネットックの別の形式をサブスクライブします。これは、ハッカーが盗難や侵害の機密性をコミットしない限り、システムに侵入できることを意味します。ただし、システムに分割する動作は、それ自体ではいっぱいではないものもあります。

侵入の意図に関係なく、クラッカーが悪用しようとする可能性のある弱点を把握しておくことが重要です。本章の残りの部分では、これらの問題に重点を置いています。

47.3.2. ネットワークセキュリティーへの脅威

ネットワークの以下の側面を設定する際に不適切なプラクティスを使用すると、攻撃のリスクが増大する可能性があります。

47.3.2.1. セキュリティーが十分ではないアーキテクチャー

間違った設定のネットワークは、未承認ユーザーの主要なエントリーポイントになります。信頼ベースのオープンローカルネットワークを非常に安全ではないインターネットに脆弱にしておくことは、crime-ridden neighborhood にドアアを残するのとよく似ています。しかし、最終的 には、誰かがこの機会を悪用する可能性があります。

47.3.2.1.1. ブロードキャストネットワーク

システム管理者は、セキュリティー計画においてネットワーキングハードウェアの重要性を見落としがちです。ハブやルーターなどの単純なハードウェアは、ブロードキャストまたはスイッチ化されていない原則に依存します。つまり、ノードがネットワークを介して受信者ノードへデータを送信するたびに、ハブまたはルーターは受信者ノードがデータを受信して処理するまで、データパケットのブロードキャストを送信します。この方法は、外部侵入者およびローカルホストの未承認ユーザーによるアドレス解決プロトコル(arp)またはメディアアクセス制御(MAC)アドレスのスプーフィングに最も脆弱です。

47.3.2.1.2. 集中化サーバー

ネットワーキングのもうひとつの落とし穴は、集中化されたコンピューティングの使用にあります。多くの企業では、一般的なコスト削減手段として、すべてのサービスを 1 台の強力なマシンに統合しています。集中化は、複数サーバーを設定するよりも管理が簡単で、コストを大幅に削減できるので便利です。ただし、集中化されたサーバーはネットワークにおける単一障害点となります。中央のサーバーが攻撃されると、ネットワークが完全に使用できなくなるか、データの不正操作や盗難が起きやすくなる可能性があります。このような状況では、中央サーバーがネットワーク全体へのアクセスを許可するドアになります。

47.3.3. サーバーセキュリティーへの脅威

サーバーには組織の重要情報が数多く含まれることが多いため、サーバーのセキュリティーは、ネットワークのセキュリティーと同様に重要です。サーバーが攻撃されると、クラッカーが意のままにすべてのコンテンツを盗んだり、不正に操作したりできるようになる可能性があります。以下のセクションでは、主要な問題の一部を詳述します。

47.3.3.1. 未使用のサービスと開かれたポート

Red Hat Enterprise Linux のフルインストールには、1000 以上のアプリケーションとライブラリーパッケージが含まれています。ただし、サーバー管理者が、ディストリビューションに含まれるすべての個別パッケージをインストールすることはほとんどありません。代わりに、複数のサーバーアプリケーションを含むパッケージのベースインストールを行います。

システム管理者は、インストールに含まれるプログラムに注意を向けずにオペレーティングシステムをインストールしてしまうことがよくあります。これにより、不要なサービスがインストールされ、デフォルト設定でオンになっていることで、問題が発生する場合があります。これにより、管理者が認識せずに、Telnet、DHCP、または DNS などの不要なサービスがサーバーまたはワークステーションで実行される可能性があります。これにより、サーバーへの不要なトラフィックが発生したり、クラッ

カーのシステムに送られる可能性があります。ポートを閉じたり、未使用のサービスを無効にする方法については、「サーバーセキュリティー」 を参照してください。

47.3.3.2. パッチが適用されないサービス

デフォルトのインストールに含まれるほとんどのサーバーアプリケーションは、ソフトウェアの細部まで徹底的にテストされており、堅牢な作りになっています。何年も実稼働環境で使用される中で、そのコードは入念に改良され、数多くのバグが発見されて修正されてきました。

しかし、完璧なソフトウェアというものはなく、改良の余地は常にあります。または、比較的新しいソフトウェアは、実稼働環境に導入されてから日が浅く、他のサーバーソフトウェアほど普及していないこともあるため、厳密なテストが期待通りに行われていない状況も少なくありません。

開発者やシステム管理者が、サーバーアプリケーションで悪用される可能性のあるバグを発見することも多々あり、Bugtraq メーリングリスト (http://www.securityfocus.com)、Computer Emergency Response Team (CERT) Web サイト (http://www.cert.org) などで、バグ追跡やセキュリティー関連の Web サイトに関連する情報が公開されています。このような情報発信は、コミュニティーにセキュリティーの脆弱性を警告する効果的な方法ではありますが、システムに速やかにパッチを当てるかどうかは個々のシステム管理者が決定します。クラッカーも、パッチが適用されていないシステムがあればクラッキングできるように、脆弱性トラッキングサービスにアクセスし、関連情報を利用できることを考慮すると、速やかな対応がとりわけ重要になります。優れたシステム管理を行うには、警戒を怠らず、バグ追跡を絶えず行い、適切なシステム保守を実行して、よりセキュアなコンピューティング環境を維持することが求められます。

システムを最新状態に維持する方法についての詳細は、「セキュリティー更新」 を参照してください。

47.3.3.3. 管理における不注意

管理者がシステムにパッチを当てないことが、サーバーのセキュリティーに対する最大の脅威の 1 つになります。System Administration Network and Security Institute (SANS)によると、コンピューターのセキュリティーの脆弱性の主な原因は、コンピューターのセキュリティー脆弱性の主な原因として、セキュリティーの維持と、トレーニングや、この作業を行うことができない時間を提供することです。[13] これは、管理者の経験の少なさだけでなく、管理者の過信やモチベーションの低さなども原因となります。

管理者が、サーバーやワークステーションにパッチを当てることを忘れたり、システムのカーネルやネットワーク通信のログメッセージを見落とす場合もあります。その他にも、よく起こるケースとして、サービスのデフォルトパスワードや鍵を変更しないまま放置しておくことが挙げられます。たとえば、データベースにはデフォルトの管理パスワードが設定されているものがありますが、ここでは、システム管理者がインストール後すぐにデフォルトパスワードを変更することを、データベース開発者は想定しています。しかし、データベース管理者がパスワードを変更することを忘れると、クラッカーの

経験が浅くても、周知のデフォルトパスワードを使用してデータベースの管理者権限を得ることができます。この他に、管理者の不注意によりサーバーが危険にさらされる場合もあります。

47.3.3.4. 本質的に安全ではないサービス

どんなに注意深い組織であっても、選択するネットワークサービスが本質的に安全でない限り、攻撃を受けやすくなります。たとえば、多くのサービスは、信頼できるネットワークでの使用を想定して開発されますが、このサービスが (本質的に信頼できない) インターネットで利用可能になる時点で、この仮定は成立しなくなります。

安全ではないネットワークサービスの例として、暗号化されていないユーザー名とパスワードを認証時に要求するサービスが挙げられます。具体例としては、Telnet や FTP の 2 つがあげられます。パケット盗聴ソフトウェアがリモートユーザーとこのようなサービスの間のトラフィックを監視していれば、ユーザー名とパスワードは簡単に傍受される可能性があります。

また、基本的にこのようなサービスはセキュリティー業界で 中間者 攻撃と呼ばれる攻撃の被害者になりやすくなります。この種の攻撃では、クラッカーが、ネットワーク上でクラッキングしたネームサーバーを操って、目標のサーバーではなくクラッカーのマシンを指定して、ネットワークトラフィックをリダイレクトします。誰かがサーバーへのリモートセッションを開くと、攻撃者のマシンがリモートサービスと無防備なユーザーとの間に存在する目に見えないパイプとして機能し、この間を流れる情報を取り込みます。このようにして、クラッカーはサーバーやユーザーに気付かれることなく、管理パスワードや生データを収集できるようになります。

安全ではないサービスの例としては、他にも NFS、NIS などのネットワークファイルシステムおよび情報サービスが挙げられます。このサービスは、LAN 利用を目的として開発されましたが、(リモートユーザー用の) WAN も対象に含まれるように拡張されました。NFS では、クラッカーによる NFS 共有のマウントやそこに格納されているものへのアクセスを防ぐ認証やセキュリティーの仕組みがデフォルトで設定されていません。NIS も、プレーンテキストの ASCII または DBM (ASCII から派生) データベースに、パスワードやファイルパーミッションなど、ネットワーク上の全コンピューターへの周知が必要となる重要な情報を保持しています。クラッカーがこのデータベースのアクセス権を取得すると、管理者のアカウントを含む、ネットワークのすべてのユーザーアカウントにアクセスできるようになります。

デフォルトでは、Red Hat Enterprise Linux は、このようなすべてのサービスをオフにしてリリースされています。ただし、管理者は、このようなサービスを使用しないといけない場合があるため、注意して設定することが重要となります。安全な方法でサービスを設定する方法は、「サーバーセキュリティー」 を参照してください。

47.3.4. ワークステーションおよび家庭用 PC のセキュリティーに対する脅威

ワークステーションや家庭用 PC はネットワークやサーバーほど攻撃にさらされることはないかもしれませんが、クレジットカード情報のような機密データが含まれるため、システムクラッカーの標的になります。ワークステーションは知らぬ間に攻撃者によって選択され、一連の攻撃でスレーブマシンとして使用される可能性もあります。このため、ユーザーはワークステーションの脆弱性を理解してお

くと、オペレーティングシステムの再インストールや、深刻な場合はデータ盗難からの回復といった問題から免れることができます。

47.3.4.1. 不適切なパスワード

攻撃者が最も簡単にシステムへのアクセスを得る方法の 1 つとして、パスワードが適切でないことが挙げられます。パスワードの作成時に一般的なミスを回避する方法は、「パスワードセキュリティー」 を参照してください。

47.3.4.2. 脆弱なクライアントアプリケーション

管理者がサーバーに十分な安全対策を施し、パッチを当てている場合でも、リモートユーザーによるアクセスが安全であるわけではありません。たとえば、サーバーが公開ネットワーク上で Telnet や FTP のサービスを提供している場合、攻撃者はネットワークを通過するプレーンテキストのユーザー名とパスワードを取り込み、アカウント情報を使用してリモートユーザーのワークステーションにアクセスすることが可能です。

SSH などのセキュアなプロトコルを使用している場合であっても、クライアントアプリケーションを定期的に更新していないと、リモートユーザーは特定の攻撃を受けやすくなる可能性があります。たとえば、v.1 SSH クライアントは悪意のある SSH サーバーからの X 転送攻撃に対して脆弱です。クライアントがサーバーに接続すると、攻撃者はネットワーク上でクライアントによるキー入力やマウス操作をひそかに収集できます。この問題は v.2 SSH プロトコルで修正されましたが、ユーザーはどのアプリケーションにこのような脆弱性があるかを追跡し、必要に応じてアプリケーションを更新する必要があります。

「ワークステーションのセキュリティー」 では、管理者およびホームユーザーがコンピューターワークステーションの脆弱性を制限するために必要な手順について詳しく説明します。

47.4. 一般的な不正使用と攻撃

表47.1「一般的な不正使用」 では、侵入者が組織のネットワークリソースにアクセスするために使用する最も一般的な不正使用とエントリーポイントの一部について詳しく説明します。この一般的な不正使用では、それがどのように実行され、管理者がその攻撃からネットワークをどのように適切に保護できるかを理解していることが重要になります。

表47.1 一般的な不正使用

不正使用	説明	注記

不正使用	説明	注記
空またはデフォルトのパスワード	管理パスワードを空白のままにしたり、製品ベンダーが設定したデフォルトのパスワードをそのまま使用します。これは、ルーターやファイアウォールなどのハードウェアで最もよく見られますが、Linux で実行されるサービスにはデフォルトの管理者パスワードを含めることができます（ただし、Red Hat Enterprise Linux 5 には同梱されません）。	一般的に、ルーター、ファイアウォール、VPN、ネットワーク接続ストレージ (NAS) の機器など、ネットワークハードウェアに関連するものです。 多くのレガシーオペレーティングシステム、特にサービスをバンドルする OS (UNIX や Windows など)で一般的。 管理者が rush で特権ユーザーアカウントを作成し、そのアカウントを検出した悪意のあるユーザーに最適なエントリーポイントであるパスワードを null のままにする場合があります。
デフォルトの共有鍵	セキュアなサービスでは、開発や評価テスト向けにデフォルトのセキュリティー鍵がパッケージ化されていることがあります。この鍵を変更せずにインターネットの実稼働環境に置いた場合は、同じデフォルトの鍵を持つすべての ユーザーがその共有鍵のリソースや、そこにあるすべての機密情報にアクセスできるようになります。	無線アクセスポイントや、事前設定済みでセキュアなサーバー機器に最も多く見られます。

不正使用	説明	注記
IP スプーフィング	リモートマシンがローカルネットワークのノードのように動作し、サーバーに脆弱性を見つけるとバックドアプログラムまたはトロイの木馬をインストールして、ネットワークリソース全体へのコントロールを得ようとします。	スプーフィングは、攻撃者が TCP/IP SYN-ACK 番号を予測してターゲットシステムへの接続を調整する必要があるため、非常に困難になりますが、クラッカーによるこのような脆弱性の実行を支援するために利用できるツールがいくつかあります。 標的となるシステムで実行している source-based 認証技術を使用するサービス (rsh、telnet、FTP など) により異なりますが、このようなサービスは、ssh、または SSL/TLS で使用される PKI などの形式の暗号化認証と比較すると推奨されません。
盗聴	2 つのノード間の接続を盗聴することにより、ネットワーク上のアクティブなノード間を行き交うデータを収集します。	この種類の攻撃には大抵、Telnet、FTP、HTTP 転送などのプレーンテキストの転送プロトコルが使用されます。 リモートの攻撃者がこのような攻撃を仕掛けるには、LAN で、攻撃するシステムへのアクセス権が必要になります。通常、クラッカーは、LAN 上にあるシステムを危険にさらすためにアクティブ攻撃 (IP スプーフィングや中間者攻撃など) を仕掛けます。 パスワードのなりすましに対する防護策としては、暗号化鍵交換、ワンタイムパスワード、または暗号化された認証によるサービス使用が挙げられます。通信中は強力な暗号化を実施することをお勧めします。

不正使用	説明	注記
共有の脆弱性	攻撃者はインターネットで実行しているサービスの欠陥や抜け穴を見つけます。攻撃者がこの脆弱性を利用する場合は、システム全体と格納されているデータを攻撃するだけでなく、ネットワーク上の他のシステムも攻撃する可能性があります。	CGI などの HTTP ベースのサービスは、リモートのコマンド実行やインタラクティブなシェルアクセスに対しても脆弱です。HTTP サービスが nobody などの権限のないユーザーとして実行している場合でも、設定ファイルやネットワークマップなどの情報が読み取られる可能性があります。または、攻撃者がサービス拒否攻撃を開始して、システムのリソースを浪費させたり、他のユーザーが利用できないようにする可能性もあります。 開発時およびテスト時には気が付かない脆弱性がサービスに含まれることがあります。(アプリケーションのメモリーバッファー領域をあふれさせ、任意のコマンドを実行できるようなインタラクティブなコマンドプロンプトを攻撃者に提供するように、攻撃者が任意の値を使用してサービスをクラッシュさせる バッファーオーバーフローなどの) 脆弱性は、完全な管理コントロールを攻撃者に与えるものとなる可能性があります。 管理者は、root 権限でサービスが実行されないようにし、ベンダー、または CERT、CVE などのセキュリティー組織がアプリケーション用のパッチやエラータ更新を提供していないかを常に注意する必要があります。

不正使用	説明	注記
アプリケーションの脆弱性	攻撃者はデスクトップおよびワークステーションのアプリケーション（メールクライアントなど）で障害を見つけ、任意コードを実行し、今後の不正使用のために Trojan Horses を引き継ぎ、システムをクラッシュさせます。攻撃を受けたワークステーションがネットワークの残りの部分に対して管理特権を持っている場合は、さらなる不正使用が起こる可能性があります。	ワークステーションとデスクトップは、ユーザーが侵害を防いだり検知するための専門知識や経験を持たないため、不正使用の対象になりやすくなります。認証されていないソフトウェアをインストールしたり、要求していないメールの添付ファイルを開く際には、それに伴うリスクについて個々に通知することが必須です。 電子メールクライアントソフトウェアが添付ファイルを自動的に開いたり、実行したりしないようにするといった、予防手段を取ることが可能です。さらに、Red Hat Network またはその他のシステム管理サービスを介したワークステーションソフトウェアの自動更新により、マルチシートのセキュリティーデプロイメントの負担を軽減することができます。

不正使用	説明	注記
サービス拒否攻撃 (DoS: Denial of Service)	単独の攻撃者または攻撃者のグループは、目標のホスト (サーバー、ルーター、ワークステーションのいずれか) に認証されていないパケットを送り、組織のネットワークまたはサーバーのリソースに対して攻撃を仕掛けます。これにより、正当なユーザーがリソースを使用できなくなります。	米国で最も多く報告された DOS の問題は、2000 年に発生しました。この時、通信量が非常に多い民間および政府のサイトが一部が利用できなくなりました。ゾンビ (zombie) や、リダイレクトされたブロードキャストノードとして動作する高帯域幅接続を有し、セキュリティー侵害された複数のシステムを使用して、調整された ping フラッド攻撃が行われたためです。 ソースパケットは、通常（再ブロードキャスト）され、実際の攻撃元を調査するのが困難になります。 iptables を使用したイングレスフィルターリング(IETF rfc2267) および snort などのネットワーク ID の進捗は、管理者が分散 DoS 攻撃を追跡し、防止するのに役立ちます。

47.5. セキュリティー更新

セキュリティー上の脆弱性が検出されると、潜在的なセキュリティーリスクを制限するために、影響を受けるソフトウェアを更新する必要があります。ソフトウェアが現在サポートされている Red Hat Enterprise Linux ディストリビューション内のパッケージの一部である場合、Red Hat, Inc. は、脆弱性をできるだけ早く修正する更新パッケージのリリースに取り組んでいます。多くの場合、特定のセキュリティーエクスプロイトに関するアナウンスはパッチ（または問題を修正するソースコード）に含まれています。このパッチは Red Hat Enterprise Linux パッケージに適用され、Red Hat の品質保証チームによってテストされ、エラータ更新としてリリースされます。ただし、発表にパッチが含まれて

いない場合、Red Hat 開発者はソフトウェアのメンテナーと連携して問題を修正します。問題が修正されると、パッケージはエラータ更新としてテストされ、リリースされます。

システムで使用しているソフトウェアでエラータの更新がリリースされると、影響を受けるパッケージをできるだけ早く更新して、システムが潜在的に脆弱になる時間を最小限に抑えることを強く推奨します。

47.5.1. パッケージの更新

システムでソフトウェアを更新する場合は、信頼できるソースから更新をダウンロードすることが重要です。攻撃者は、問題を解決するはずのパッケージと同じバージョン番号で、異なるセキュリティーエクスプロイトを施したパッケージを簡単に作り直し、インターネット上で公開することができます。この場合、元の RPM に対するファイルの検証など、セキュリティー対策を使用しても不正使用が検出されません。そのため、RPM は Red Hat, Inc. などの信頼できるソースからのみダウンロードし、その整合性を検証するためにパッケージの署名を確認することが非常に重要です。

Red Hat では、エラータ更新に関する情報を見つける方法を 2 つ提供しています。

1. 一覧表示され、Red Hat Network でダウンロードできます。

2. Red Hat エラータ Web サイトで一覧表示およびリンク解除

注記

Red Hat Enterprise Linux の製品ライン以降、更新されたパッケージは Red Hat Network からのみダウンロードできます。Red Hat エラータ Web サイトには更新された情報が含まれていますが、ダウンロードする実際のパッケージは含まれていません。

47.5.1.1. RHN Classic での自動更新の使用

警告：非推奨機能

システムの自動更新は、RHN Classic を使用してのみ利用できます。これは、コンテンツリポジトリーチャネルへのアクセス時にサブスクリプション消費に基づきます。RHN Classic は、証明書ベースの Red Hat Network に更新されていないレガシーシステムを使用するお客様の環境の利便性として利用できます。

更新とコンテンツストリームは、証明書ベースの Red Hat Network で異なるため、自動更新は使用されません。

新しい証明書ベースの Red Hat Network と、証明書ベースの Red Hat Network と RHN Classic の相違点は、15章システムの登録およびサブスクリプション管理 で説明されています。

RHN Classic を使用すると、更新プロセスの大半が自動化されます。これは、システムに必要な RPM パッケージを判断し、安全なリポジトリーからダウンロードし、RPM 署名を検証し、改ざんされていないことを確認します。パッケージのインストールはすぐに行われるか、一定期間にスケジュールされる可能性があります。

RHN Classic には、システムに関するハードウェアおよびソフトウェア情報が含まれる各マシンのシステムプロファイル が必要です。この情報は機密で保持され、他の人には提供されません。これは、各システムに適用可能なエラータ更新のみを決定するためにのみ使用されます。また、その更新がないと、RHN Classic は特定のシステムの更新が必要であるかどうかを判断できません。セキュリティーエラータ（または任意のタイプのエラータ）がリリースされると、RHN Classic はエラータの説明と影響を受けるシステムの一覧と共に電子メールを送信します。更新を適用するには、Red Hat Update Agent を使用するか、カスタマーポータルの RHN Classic サブスクリプション管理 エリアでパッケージを更新するようにスケジュールします。

重要な影響

セキュリティーエラータをインストールする前に、エラータレポートに含まれる特別な指示を必ず読み、それに応じて実行してください。エラータ更新による変更の適用に関する一般的な手順は、「変更の適用」 を参照してください。

47.5.1.2. Red Hat エラータ Web サイトの使用

セキュリティーエラータレポートがリリースされると、http://www.redhat.com/security/ から入手できる Red Hat エラータ Web サイトに公開されます。このページから、システムの製品とバージョ

ンを選択し、ページ上部の セキュリティー を選択して、Red Hat Enterprise Linux Security Advisories のみを表示します。アドバイザリーのいずれかの概要がシステムで使用されるパッケージを記述している場合は、概要をクリックして詳細を確認してください。

詳細ページでは、セキュリティーエクスプロイトと、セキュリティーホールを修正するためにパッケージの更新に加えて実行する必要がある特別な命令について説明します。

更新したパッケージをダウンロードするには、リンクをクリックして Red Hat Network にログインし、パッケージ名をクリックしてハードドライブに保存します。/tmp/updates などの新しいディレクトリーを作成し、ダウンロードしたパッケージをすべて保存することを強く推奨します。

47.5.1.3. 署名パッケージの検証

すべての Red Hat Enterprise Linux パッケージは、Red Hat, Inc で署名されています。GPG キー。GPG は GNU Privacy Guard (GnuPG)の略で、分散ファイルの信頼性を確保するために使用される無料ソフトウェアパッケージです。たとえば、Red Hat が保持する秘密鍵（シークレットキー）は、公開鍵のロックを解除して、パッケージを検証する間に、パッケージをロックします。Red Hat が配信する公開鍵が RPM 検証中に秘密鍵と一致しない場合は、パッケージが変更されているため、信頼できない可能性があります。

Red Hat Enterprise Linux 内の RPM ユーティリティーは、インストール前に RPM パッケージの GPG 署名を自動的に検証しようとします。Red Hat GPG キーがインストールされていない場合は、Red Hat Enterprise Linux インストール CD-ROM などの安全な静的場所からインストールします。

CD-ROM が /mnt/cdrom にマウントされている場合は、以下のコマンドを使用してキー リング（システム上の信頼できるキーのデータベース）にインポートします。

```
rpm --import /mnt/cdrom/RPM-GPG-KEY-redhat-release
```

RPM 検証用にインストールされた鍵の一覧を表示するには、以下のコマンドを実行します。

```
rpm -qa gpg-pubkey*
```

Red Hat キーの場合、出力には以下が含まれます。

```
gpg-pubkey-37017186-45761324
```

特定のキーの詳細を表示するには、以下の例のように rpm -qi コマンドの後に直前のコマンドの出力を使用します。

```
rpm -qi gpg-pubkey-37017186-45761324
```

RPM ファイルの署名をインストールする前に、RPM ファイルの署名を検証することが非常に重要です。これにより、パッケージの Red Hat, Inc. リリースから変更されていないことを確認する必要があります。ダウンロードしたすべてのパッケージを一度に確認するには、以下のコマンドを実行します。

```
rpm -K /tmp/updates/*.rpm
```

各パッケージで GPG キーが正常に検証された場合、コマンドは gpg OK を返します。そうでない場合は、正しい Red Hat 公開鍵を使用していることと、コンテンツのソースを確認してください。GPG 検証に合格しないパッケージは、サードパーティーによって変更されている可能性があるため、インストールしないでください。

GPG キーを確認し、エラータレポートに関連付けられたすべてのパッケージをダウンロードしたら、シェルプロンプトで root としてパッケージをインストールします。

47.5.1.4. 署名パッケージのインストール

ほとんどのパッケージのインストールは、以下のコマンドを実行して安全に実行できます（カーネルパッケージを除く）。

```
rpm -Uvh /tmp/updates/*.rpm
```

カーネルパッケージの場合は、以下のコマンドを使用します。

```
rpm -ivh /tmp/updates/<kernel-package>
```

前の例の < kernel-package > を、カーネル RPM の名前に置き換えます。

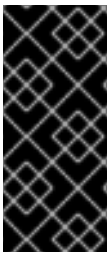
新しいカーネルを使用してマシンを安全に再起動したら、以下のコマンドを使用して古いカーネルを削除できます。

```
rpm -e <old-kernel-package>
```

前 の例の <old-kernel-package > を、古いカーネル RPM の名前に置き換えます。

注記

古いカーネルを削除する必要はありません。デフォルトのブートローダー GRUB では、複数のカーネルをインストールでき、ブート時にメニューから選択できます。



重要な影響

セキュリティーエラータをインストールする前に、エラータレポートに含まれる特別な指示を必ず読み、それに応じて実行してください。エラータ更新による変更の適用に関する一般的な手順は、「変更の適用」 を参照してください。

47.5.1.5. 変更の適用

Red Hat Network または Red Hat エラータ Web サイト経由でセキュリティーエラータをダウンロードしてインストールした後に、古いソフトウェアの使用を停止し、新しいソフトウェアの使用を開始することが重要です。これがどのように行われるかは、更新されたソフトウェアのタイプによって異なります。以下の一覧は、ソフトウェアの一般的なカテゴリーを項目化し、パッケージのアップグレード後に更新されたバージョンを使用する手順を説明します。

注記

一般的には、システムを再起動することが、ソフトウェアパッケージの最新バージョンが使用されていることを確認する最も確実な方法です。ただし、このオプションは常にシステム管理者で利用できるとは限りません。

アプリケーション

ユーザー空間アプリケーションは、システムユーザーが開始できるプログラムです。通常、このようなアプリケーションは、ユーザー、スクリプト、または自動タスクユーティリティーがそれらを起動し、長期間維持されない場合にのみ使用されます。

このようなユーザー空間アプリケーションが更新されたら、システム上のアプリケーションのインスタンスをすべて停止し、プログラムを再度起動し、更新されたバージョンを使用します。

カーネル

カーネルは、Red Hat Enterprise Linux オペレーティングシステムのコアソフトウェアコンポーネントです。メモリー、プロセッサー、および周辺機器へのアクセスを管理し、すべてのタスクをスケジュールします。

その中心的なロールがあるため、コンピューターを停止せずにカーネルを再起動することはできません。そのため、システムを再起動するまで、更新されたバージョンのカーネルを使用することはできません。

共有ライブラリー

共有ライブラリーは、多くのアプリケーションやサービスによって使用される glibc などのコードの単位です。共有ライブラリーを使用するアプリケーションは、通常、アプリケーションの初期化時に共有コードを読み込むため、更新されたライブラリーを使用するアプリケーションはすべて停止および再起動する必要があります。

実行中のどのアプリケーションが特定のライブラリーに対してリンクしているかを確認するには、以下の例のように lsof コマンドを使用します。

```
lsof /usr/lib/libwrap.so*
```

このコマンドは、ホストアクセス制御に TCP ラッパーを使用する実行中のすべてのプログラム一覧を返します。したがって、tcp_wrappers パッケージが更新されると、リストされているプログラムは停止して再起動する必要があります。

SysV Services

SysV サービスは、システムの起動プロセス中に起動する永続的なサーバープログラムです。SysV サービスの例としては、sshd、vsftpd、xinetd などがあります。

これらのプログラムは通常マシンが起動されている限りメモリーに保持されるため、更新された各 SysV サービスは、パッケージのアップグレード後に停止して再起動する必要があります。これは、Services Configuration Tool を使用するか、root シェルプロンプトにログインして、/sbin/service コマンドを以下の例のように実行して実行できます。

```
service <service-name> restart
```

前の例で、&lt ;service-name& gt; を sshd などのサービスの名前に置き換えます。

Services Configuration Tool の詳細は、17章Network Configuration を参照してください。

xinetd サービス

xinetd スーパーサービスが制御するサービスは、アクティブな接続がある場合にのみ実行されます。xinetd が制御するサービスの例には、Telnet、IMAP、および POP3 が含まれます。

これらのサービスの新しいインスタンスは新しいリクエストを受け取るたびに xinetd によって起動されるため、アップグレード後に発生する接続は更新されたソフトウェアによって処理されます。ただし、xinetd が制御するサービスのアップグレード時にアクティブな接続がある場合は、ソフトウェアの古いバージョンによって提供されます。

特定の xinetd 制御サービスの古いインスタンスを強制終了するには、そのサービスの パッケージをアップグレードしてから、現在実行中のプロセスをすべて停止します。プロセスが実行されているかどうかを確認するには、ps コマンドを使用して kill または killall コマンドを使用して、サービスの現在のインスタンスを停止します。

たとえば、セキュリティーエラータ imap パッケージがリリースされている場合は、パッケージをアップグレードし、root でシェルプロンプトに以下のコマンドを入力します。

```
ps -aux | grep imap
```

このコマンドは、アクティブな IMAP セッションをすべて返します。以下のコマンドを実行すると、個々のセッションを終了できます。

```
kill <PID>
```

セッションの終了に失敗した場合は、代わりに以下のコマンドを使用します。

```
kill -9 <PID>
```

前の例では、& lt;PID > を IMAP セッションのプロセス ID 番号(ps コマンドの 2 列目にある) に置き換えます。

アクティブな IMAP セッションをすべて強制終了するには、以下のコマンドを実行します。

```
killall imapd
```

[10]
Source: http://www.cert.org

[11]
Source: http://www.cert.org/stats/

[12]
出所： http://www.newsfactor.com/perl/story/16407.html

[13]
Source: http://www.sans.org/security-resources/mistakes.php

第48章 ネットワークのセキュリティー保護

48.1. ワークステーションのセキュリティー

Linux 環境のセキュリティー保護は、ワークステーションから開始します。個人マシンをロックするか、エンタープライズシステムのセキュリティーを保護する場合でも、サウンドセキュリティーポリシーは個々のコンピューターから開始します。コンピューターネットワークは、最も弱いノードほど安全です。

48.1.1. ワークステーションのセキュリティーの評価

Red Hat Enterprise Linux ワークステーションのセキュリティーを評価する際には、以下を考慮してください。

- BIOS およびブートローダーセキュリティー：承認されていないユーザーがマシンに物理的にアクセスし、パスワードなしで単一ユーザーまたはレスキューモードで起動 できますか？
- パスワードセキュリティー - マシン上のユーザーアカウントのパスワードのセキュリティーはどのように安全ですか？
- 管理コントロール - システム上にアカウントがあり、管理者の制御はどの程度ありますか？
- 利用可能なネットワーク サービス - ネットワークからの要求をリッスンしているサービスや、それらを全く実行すべきサービス
- パーソナル ファイアウォール - 必要なファイアウォールのタイプ。
- Security Enhanced Communication Tools - ワークステーションと、避けるべきなワークステーション間の通信に使用する必要があるツールは何ですか？

48.1.2. BIOS およびブートローダーのセキュリティー

BIOS (もしくは BIOS に相当するもの) およびブートローダーをパスワードで保護することで、システムに物理的にアクセス可能な未承認ユーザーがリムーバブルメディアを使用して起動したり、シングルユーザーモードで root 権限を取得することを防ぐことができます。このような攻撃に対するセキュリティー対策は、ワークステーションの情報の機密性とマシンの場所によって異なります。

たとえば、見本市で使用されていて機密情報を含んでいないマシンでは、このような攻撃を防ぐことが重要ではないかもしれません。しかし、同じ見本市で、企業ネットワークに対して暗号化されていない SSH 秘密鍵のある従業員のノートパソコンが、誰の監視下にもなく置かれていた場合は、重大なセキュリティー侵害につながり、その影響は企業全体に及ぶ可能性があります。

一方で、ワークステーションが権限のあるユーザーもしくは信頼できるユーザーのみがアクセスできる場所に置かれてるのであれば、BIOS もしくはブートローダーの安全確保は必要ない可能性もあります。

48.1.2.1. BIOS パスワード

コンピューターの BIOS をパスワードで保護する主な 2 つの理由を以下に示します。[14]:

1. BIOS 設定の変更を防止する - 侵入者が BIOS にアクセスできる場合は、ディスケットまたは CD-ROM から起動するように設定できます。このようにすると、侵入者がレスキューモードやシングルユーザーモードに入ることが可能になり、システムで任意のプロセスを開始したり、機密性の高いデータをコピーできるようになってしまいます。

2. システムの起動を防止する - BIOS の中には起動プロセスをパスワードで保護できるものもあります。これを有効にすると、攻撃者は BIOS がブートローダーを開始する前にパスワード入力を求められます。

BIOS パスワードの設定方法はコンピューターメーカーで異なるため、具体的な方法はコンピューターのマニュアルを参照してください。

BIOS パスワードを忘れた場合は、マザーボードのジャンパーでリセットするか、CMOS バッテリーを外します。このため、可能な場合はコンピューターのケースをロックすることが推奨されます。ただし、CMOS バッテリーを外す前にコンピューターもしくはマザーボードのマニュアルを参照してください。

48.1.2.1.1. x86 以外のプラットフォームのセキュリティー保護

他のアーキテクチャーは、異なるプログラムを使用して、x86 システムの BIOS とほぼ同等の低レベルのタスクを実行します。たとえば、Intel® Itanium™ コンピューターは Extensible Firmware Interface (EFI)シェルを使用します。

他のアーキテクチャーで BIOS のようなプログラムをパスワード保護する方法は、メーカーにお問い合わせください。

48.1.2.2. ブートローダーのパスワード

Linux ブートローダーをパスワードで保護する主な理由は、以下のとおりです。

1. シングルユーザーモードへのアクセスの防止 - 攻撃者がシステムをシングルユーザーモードで起動できる場合、root パスワードを求められることなく、root として自動的にログインします。

2. GRUB コンソールへのアクセスの防止 - マシンがブートローダーとして GRUB を使用している場合、攻撃者は GRUB エディターインターフェイスを使用して設定を変更したり、cat コマンドを使用して情報を収集したりできます。

3. 安全でないオペレーティングシステムへのアクセスの防止 - デュアルブートシステムの場合、攻撃者はアクセス制御やファイルパーミッションを無視するオペレーティングシステムを起動時に選択することができます(DOS など)。

Red Hat Enterprise Linux には、x86 プラットフォームの GRUB ブートローダーが同梱されています。GRUB の詳細は、Red Hat Installation Guide を参照してください。

48.1.2.2.1. GRUB が保護するパスワード

設定ファイルに password ディレクティブを追加すると、「ブートローダーのパスワード」 の最初の 2 つの問題に対処するように GRUB を設定できます。これを行うには、最初に強力なパスワードを選択し、シェルを開き、root でログインし、以下のコマンドを入力します。

```
grub-md5-crypt
```

プロンプトが表示されたら、GRUB パスワードを入力し、Enter を押します。これにより、パスワードの MD5 ハッシュが返されます。

次に、GRUB 設定ファイル /boot/grub/grub.conf を編集します。ファイルを開き、ドキュメントのメインセクションにある timeout 行の下に、以下の行を追加します。

```
password --md5 <password-hash>
```

&lt ;password-hash> を /sbin/grub-md5-cryptが返す値に置き換えます。[15]をクリックします。

次回システムを起動すると、GRUB メニューは、p を押すと、その後に GRUB パスワードを指定しなくても、エディターまたはコマンドラインインターフェイスにアクセスできなくなります。

ただし、このソリューションでは、攻撃者がデュアルブート環境でセキュアでないオペレーティングシステムで起動できないわけではありません。そのためには、/boot/grub/grub.conf ファイルの別の部分を編集する必要があります。

セキュリティーを保護するオペレーティングシステムの title 行を探し、その下の lock ディレクティブの行を追加します。

DOS システムの場合には、スタンザは以下のように開始する必要があります。

```
title DOS lock
```

WARNING

この方法が正しく機能するには、/boot/grub/grub.conf ファイルのメインセクションに password 行が存在する必要があります。そうしないと、攻撃者は GRUB エディターインターフェイスにアクセスし、ロックラインを削除できます。

特定のカーネルまたはオペレーティングシステムに異なるパスワードを作成するには、スタンザに lock 行を追加し、続いて password 行を追加します。

一意のパスワードで保護される各スタンザは、以下の例のような行で始まる必要があります。

```
title DOS lock password --md5 <password-hash>
```

48.1.3. パスワードセキュリティー

パスワードは、Red Hat Enterprise Linux がユーザーの ID を検証するために使用する主要な方法です。このため、パスワードのセキュリティーは、ユーザー、ワークステーション、ネットワークを保護するために非常に重要です。

セキュリティー上の理由から、インストールプログラムは、Message-Digest Algorithm (MD5)およびシャドウパスワードを使用するようにシステムを設定します。これらの設定を変更しないことを強く推奨します。

インストール時に MD5 パスワードの選択を解除すると、古い Data Encryption Standard (DES)形式が使用されます。この形式では、パスワードを 8 桁の英数字（句読者などの特殊文字を許可）に制限し、中程度の 56 ビット暗号化レベルを提供します。

インストール時にシャドーパスワードの選択を解除すると、すべてのパスワードが一方向ハッシュとして誰でも読み取り可能な /etc/passwd ファイルに保存されるため、システムはオフラインでのパスワードクラッキング攻撃に対して脆弱になります。侵入者が通常のユーザーとしてマシンにアクセスできる場合は、/etc/passwd ファイルを自分のマシンにコピーし、それに対して任意の数のパスワードクラッキングプログラムを実行できます。ファイル内に安全でないパスワードがあれば、パスワードクラッカーに発見されるのは時間の問題です。

シャドーパスワードは、パスワードのハッシュを root ユーザーのみが読み取り可能なファイル /etc/shadow に保存することで、このタイプの攻撃を排除します。

これにより、潜在的な攻撃者は、SSH や FTP などのマシン上のネットワークサービスにログインして、リモートでパスワードクラッキングを試みるようになります。この種のブルートフォース攻撃ははるかに遅く、何百回ものログイン試行の失敗がシステムファイルに書き込まれるため、明らかな痕跡が残ります。当然ながら、クラッカーがパスワードの弱いシステムで深夜に攻撃を開始した場合、クラッカーはダイヤーの前にアクセスを取得し、ログファイルを編集してその追跡をカバーすることがあります。

形式およびストレージの考慮事項に加えて、コンテンツの問題があります。パスワードクラッキング攻撃からアカウントを保護できる最も重要なことは、強力なパスワードを作成することです。

48.1.3.1. 強固なパスワードの作成

安全なパスワードを作成する場合は、以下のガイドラインに従うことが推奨されます。

- Words または Numbers のみは使用しないでください。パスワードには数字または単語のみを使用しないでください。

  非セキュアな例には、以下が含まれます。

  - *8675309*

- *juan*
- *hackme*

- **Recogniz able Words は使用しないでください。適切な名前、辞書の単語、またはテレビター *の用語や、数字で予約されている場合でも、回避すべきです。***

  非セキュアな例には、以下が含まれます。

  - *john1*
  - *DS-9*
  - *mentat123*

- **Foreign Languages で Words を使用しないでください。パスワードクラッキングプログラムは、多くの言語のディクショナリーを含む単語リストに対してチェックを行う *ことがよくあります。安全なパスワードに外部言語を使用することは安全ではありません。***

  非セキュアな例には、以下が含まれます。

  - *cheguevara*
  - *bienvenido1*
  - *1dumbKopf*

- **do not use Hacker Terminology*: ハッカーの用語(l337 (LEET)マクターとも呼ばれる)を使用する場合には、パスワードについて再度考えてください。多くの単語リストには、LEET マクターが含まれます。***

非セキュアな例には、以下が含まれます。

- *H4X0R*
- *1337*

- **do not use Personal Information - パスワードに個人情報を使用 *しないでください。攻撃者がアイデンティティーを知らせると、パスワードを減らすタスクが簡単になります。以下は、パスワードの作成時に回避する情報の種類の一覧です。***

非セキュアな例には、以下が含まれます。

- *自分の名前*
- *ペットの名前*
- *ファミリーメンバーの名前*
- *生年月日*
- *電話番号または zip コード*

- **Not Invert Recognizable Words - *Good password checkers always reverse common words, so inverting a bad password does not any more secure.***

非セキュアな例には、以下が含まれます。

- *R0X4H*

- *nauj*
- *9-DS*

- **Do not Write Down Your Password***: 文書にパスワードを保存しないでください。覚えておく方がはるかに安全です。*

- *すべてのマシンに同じパスワードを使用しないでください。各マシンに個別のパスワードを作成することが重要です。これにより、1 つのシステムが危険にさらされると、すべてのマシンが危険にさらされることはありません。*

以下のガイドラインは、強力なパスワードを作成するのに役立ちます。

- *パスワードを* **Least Eight Characters Long** *にする - パスワードが より長くなります。MD5 パスワードを使用する場合は、15 文字以上である必要があります。DES パスワードでは、最大長(8 文字)を使用します。*

- **大文字の組み合わせと小文字の組み合わせ** *- Red Hat Enterprise Linux は大文字と小文字を区別するため、組み合わせてパスワードの強度を強化します。*

- **英数字と番号の組み合わせ** *- 特に（最初または末尾だけでなく）中間者に追加する場合はパスワードの強度を強化できます。*

- **Include Non-Alphanumeric Characters** *- &, $, > などの特殊文字を使用すると、パスワードの強度が大幅に向上します(DES パスワードを使用する場合は不可能です)。*

- *パスワードを覚えておくことができない場合、パスワードの選択 -* **世界で最善な** *パスワードはほとんど役に立ちません。パスワードを記憶するには、頭文字などのデバイスを使用してください。*

これらのルールはすべて、不正なパスワードの特性を回避しながら、適切なパスワードのすべての基準を満たすパスワードを作成することが難しい場合があります。幸いなことに、覚えやすい安全なパスワードを生成する手順がいくつかあります。

48.1.3.1.1. 安全なパスワード作成方法

セキュアなパスワードの作成に使用する方法は多数あります。より一般的な方法の 1 つに、頭字語が必要です。以下に例を示します。

- 以下のような簡単に覚えているフレーズについて考えてみましょう。

  "Randm through the river and through the woods, to grandmother's house we go."

- 次に、頭字語（句読点を含む）に変換します。

  otrattw,tghwg.

- 略語の文字の番号と記号を置き換えて、複雑さを追加します。たとえば、7 を t に、at 記号(@)を a に置き換えます。

  o7r@77w,7ghwg.

- H など、少なくとも 1 文字を大文字にして、複雑さをさらに追加します。

  o7r@77w,7gHwg.

- 最後に、システムには上記のサンプルパスワードを使用しないでください。

安全なパスワードの作成は必須ですが、特に大規模な組織内のシステム管理者にとっては、パスワードを適切に管理することも重要です。以下のセクションでは、組織内でユーザーパスワードを作成し、管理するためのグッドプラクティスについて詳しく説明します。

48.1.3.2. 組織内でのユーザーパスワードの作成

組織がユーザー数が多い場合、システム管理者は、適切なパスワードを強制的に使用するために 2 つの基本的なオプションを利用できます。パスワードが許容可能な品質であることを確認しつつ、ユー

ザーのパスワードを作成したり、ユーザーが自分のパスワードを作成できるようにすることもできます。

ユーザーのパスワードを作成することで、パスワードの安全性は確保されますが、組織が大きくなるにつれ、大変な作業となります。また、ユーザーがパスワードを書き込むリスクが高まります。

このような理由から、ほとんどのシステム管理者は、ユーザーが自分のパスワードを作成することを好みますが、パスワードが適切であること、場合によっては、ユーザーがパスワードのエージングを通じて定期的にパスワードを変更するように強制します。

48.1.3.2.1. 強固なパスワードの強制

侵入からネットワークを保護するには、システム管理者が組織内で使用するパスワードが強力なものであることを確認することをお勧めします。ユーザーがパスワードの作成や変更を求められたら、Pluggable Authentication Manager (PAM)を認識するコマンドラインアプリケーション passwd を使用できます。したがって、パスワードが短すぎるか、または解読しやすいかどうかを確認してください。このチェックは、pam_NORMAL.so PAM モジュールを使用して実行されます。PAM はカスタマイズ可能であるため、pam_passwdqc (http://www.openwall.com/passwdqc/から利用可能)や新しいモジュールを書き込むなど、パスワード整合性チェッカーをさらに追加することができます。利用可能な PAM モジュールの一覧は、 http://www.kernel.org/pub/linux/libs/pam/modules.html を参照してください。PAM の詳細は、 「PAM (プラグ可能な認証モジュール)」 を参照してください。

パスワードの作成時に実行されるパスワードチェックでは、パスワードに対してパスワードクラッキングプログラムを実行するのと同様に、不正なパスワードが検出されません。

Red Hat Enterprise Linux では、パスワードクラッキングプログラムが多数用意されていますが、オペレーティングシステムには同梱されていません。以下は、一般的なパスワードクラッキングプログラムの一部です。

注記

これらのツールはいずれも Red Hat Enterprise Linux には提供されないため、Red Hat, Inc. では対応していません。

- John The Ripper - 高速で柔軟なパスワードクラッキングプログラム。複数の単語リストを使用でき、ブルートフォースのパスワードクラッキングが可能です。 http://www.openwall.com/john/ からオンラインで使用できます。

- crack - 最もよく知られているパスワードクラッキングソフトウェアも非常 に 高速ですが、John The Ripper として簡単に使用できません。オンラインについては

http://www.openwall.com/john/ を参照してください。

- Slurpie - Slurpie は John The Ripper and Crack と似ていますが、複数のコンピューターで同時に実行するように設計されており、分散パスワードクラッキング攻撃を作成します。これは、http://www.ussrback.com/distributed.htm でオンラインの他の多くの分散攻撃セキュリティー評価ツールと共に確認できます。

WARNING

組織内のクラッキングを試みる前に、常に書き込みで承認を受けます。

48.1.3.2.2. パスワードのエージング

パスワードエージングは、組織内の不正なパスワードから保護するためにシステム管理者が使用するもう一つの技術です。パスワードエージングとは、指定された期間 (通常は 90 日間) が経過すると、ユーザーは新しいパスワードを作成するように求められることを意味します。この背景の理論は、ユーザーが定期的にパスワードを変更することを強制した場合、クラッキングされたパスワードは侵入者にとって限られた時間しか有効でないという理論があります。しかし、パスワードエージングには、ユーザーがパスワードを書き留める可能性が高くなるというデメリットがあります。

Red Hat Enterprise Linux では、chage コマンドまたはグラフィカルユーザー マネージャー (system-config-users)アプリケーションという、2 つの Red Hat Enterprise Linux でのパスワードの変更を指定するために使用される主なプログラムがあります。

chage コマンドの -M オプションは、パスワードの最大有効日数を指定します。たとえば、ユーザーのパスワードを 90 日間で期限切れになるように設定するには、以下のコマンドを使用します。

```
chage -M 90 <username>
```

上記のコマンドで、< username> をユーザーの名前に置き換えます。パスワードの有効期限を無効にするには、-M オプションの後に 99999 の値を使用することは従来の 273 年以上に相当します。

また、インタラクティブモードで chage コマンドを使用して、複数のパスワードエージングおよびアカウントの詳細を変更することができます。次のコマンドを使用して、インタラクティブモードに入ります。

```
chage <username>
```

以下は、このコマンドを使用した対話セッションの例です。

```
~]# chage davido
Changing the aging information for davido
Enter the new value, or press ENTER for the default

        Minimum Password Age [0]: 10
        Maximum Password Age [99999]: 90
        Last Password Change (YYYY-MM-DD) [2006-08-18]:
        Password Expiration Warning [7]:
        Password Inactive [-1]:
        Account Expiration Date (YYYY-MM-DD) [1969-12-31]:
~]#
```

利用可能なオプションの詳細は、**chage** の **man** ページを参照してください。

また、以下のように、グラフィカル **User Manager** アプリケーションを使用して、パスワードのエージングポリシーを作成することもできます。この手順を実行するには、管理者権限が必要なことに注意してください。

1. パネル上の システム メニューをクリックして 管理 から ユーザーとグループ をクリックして **User Manager** を表示させます。または、シェルプロンプトでコマンド **system-config-users** を入力します。

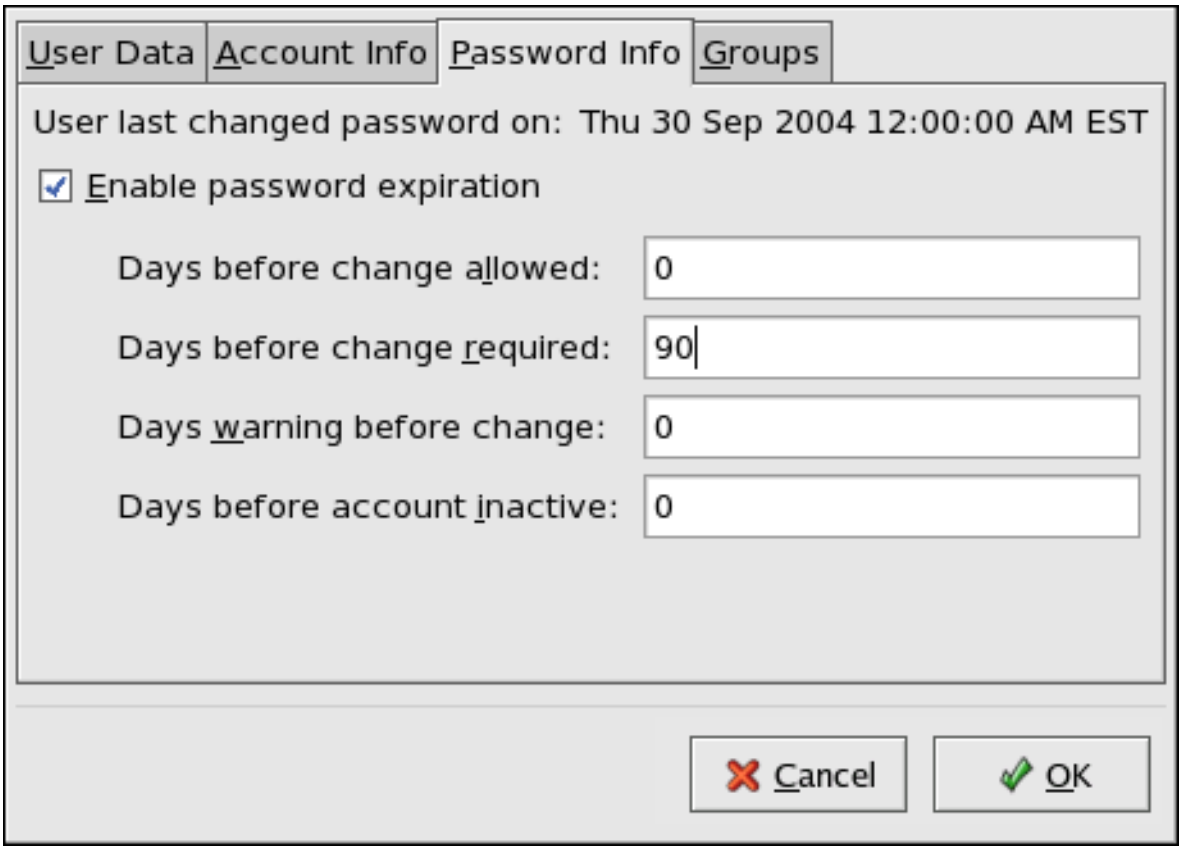
2. ユーザー タブをクリックして、ユーザーリストの中から必要なユーザーを選択します。

3. ツールバーの 設定 をクリックして、ユーザー設定のダイアログボックスを表示させます (または ファイル メニューで 設定 を選択します)。

4. **Password Info** タブをクリックし、**Enable password expiration** のチェックボックスを選択します。

5. **Days before change required** フィールドに必要な値を入力して、**OK** をクリックします。

図48.1 パスワードのエージングオプションの指定



[D]

ユーザーおよびグループの設定（初回パスワードを強制するための手順を含む）の詳細は、37章ユーザーとグループ を参照してください。

48.1.4. 管理的コントロール

ホームマシンを管理する場合、ユーザーはいくつかのタスクを root ユーザーとして実行するか、sudo や su などの setuid プログラムを介して効果的な root 権限を取得して実行する必要があります。setuid プログラムとは、プログラムを操作するユーザーではなく、プログラムの所有者のユーザーID(UID) で動作するプログラムのことです。このようなプログラムは、以下の例のように、長いフォーマットリストの所有者セクションでsで示されます。

```
-rwsr-xr-x    1 root     root       47324 May  1 08:09 /bin/su
```

注記

s は大文字または小文字の場合があります。大文字で表示されている場合は、基になる許可ビットが設定されていないことを意味します。

ただし、組織の管理者では、組織内のユーザーが自分のマシンに必要な管理アクセス量に応じて選択する必要があります。pam_console.so と呼ばれる PAM モジュールを介して、リムーバブルメディ

アの再起動やマウントなど、通常は root ユーザー専用に予約されている一部のアクティビティーは、物理コンソールでログインする最初のユーザーに許可されます(pam_console.so モジュールの詳細は、「PAM (プラグ可能な認証モジュール)」 を参照してください)。 ただし、ネットワーク設定の変更、新しいマウスの設定、ネットワークデバイスのマウントなど、その他の重要なシステム管理タスクは、管理者権限なしではできません。その結果、システム管理者は、ネットワーク上のユーザーがどの程度のアクセスを受けるべきかを決定する必要があります。

48.1.4.1. Root アクセスの許可

組織内のユーザーが信頼され、computer-literate である場合は、root アクセスを許可することは問題ではない場合があります。ユーザーによる root アクセスを可能にすると、デバイスの追加やネットワークインターフェイスの設定などのマイナーなアクティビティーが個々のユーザーが処理でき、システム管理者はネットワークセキュリティーやその他の重要な問題に対処することができます。

一方、個々のユーザーに root アクセス権限を付与すると、以下の問題が発生する可能性があります。

- マシンの設定ミス - root アクセス権を持つユーザーは、マシンの設定を誤設定し、問題の解決にサポートが必要になる場合があります。さらに悪いことに、知らずにセキュリティーホールを発生させてしまう可能性があります。
- 安全でないサービス の実行 - root アクセスを持つユーザーは、マシン上で FTP や Telnet などのセキュアでないサーバーを実行し、ユーザー名とパスワードを危険にさらす可能性があります。これらのサービスは、この情報をプレーンテキストでネットワーク経由で送信します。
- 電子メールの添付ファイルを root で実行 — まれにですが、Linux に影響を与える電子メールウィルスが存在します。ただし、脅威となるのは、root ユーザーが実行する時だけです。

48.1.4.2. Root アクセスの拒否

管理者がこれらの理由またはその他の理由で root としてログインできない場合、root パスワードは秘密にして、ブートローダーのパスワード保護によりランレベル 1 またはシングルユーザーモードへのアクセスを禁止する必要があります（このトピックの詳細については、「ブートローダーのパスワード」 を参照してください）。

また、管理者が root ログインを許可しない 4 つの方法を以下に示します。

root シェルの変更

ユーザーが root として直接ログインできないように、システム管理者は、/etc/passwd ファイルで root アカウントのシェルを /sbin/nologin に設定します。

表48.1 Root シェルの無効化

影響あり	影響なし
root シェルへのアクセスを阻止し、そのような試行をログに記録します。以下のプログラムは root アカウントにアクセスできません。 • login • gdm • kdm • xdm • su • ssh • scp • sftp	FTP クライアント、メールクライアント、多くの setuid プログラムなど、シェルを必要としないプログラム。以下のプログラムは root アカウントにアクセスできません。 • sudo • FTP クライアント • Email クライアント

任意のコンソールデバイス(tty)を介した root アクセスの無効化

root アカウントへのアクセスをさらに制限するために、管理者は /etc/securetty ファイルを編集してコンソールで root ログインを無効にすることができます。このファイルは、root ユーザーがログインできるすべてのデバイスを一覧表示します。ファイルが存在しない場合は、コンソールまたは raw ネットワークインターフェイスを介して、root ユーザーはシステム上の任意の通信デバイス経由でログインできます。これは危険です。これは、ユーザーが Telnet を介して root としてマシンにログインできるためです。これにより、パスワードがプレーンテキストでネットワークを介して送信されます。

デフォルトでは、Red Hat Enterprise Linux の /etc/securetty ファイルでは、マシンに物理的に接続されているコンソールで root ユーザーしかログインできません。root ユーザーがログインできないようにするには、root でシェルプロンプトで以下のコマンドを入力して、このファイルの内容を削除します。

```
echo > /etc/securetty
```

KDM、GDM、XDM のログインマネージャーでsecuretty サポートを有効にするには、次の行を追加します。

```
auth [user_unknown=ignore success=ok ignore=ignore default=bad] pam_securetty.so
```

追加対象のファイルは以下のとおりです。

- /etc/pam.d/gdm
- /etc/pam.d/gdm-autologin
- /etc/pam.d/gdm-fingerprint
- /etc/pam.d/gdm-password
- /etc/pam.d/gdm-smartcard
- /etc/pam.d/kdm

- */etc/pam.d/kdm-np*
- */etc/pam.d/xdm*

WARNING

空の /etc/securetty ファイルは、root ユーザーが OpenSSH スイートを使用してリモートでログインすることを阻止しません。これは、認証後までコンソールが開かないためです。

表48.2 Root ログインの無効化

影響あり	影響なし
コンソールまたはネットワークを介して root アカウントにアクセスできないようにします。以下のプログラムは root アカウントにアクセスできません。 • login • gdm • kdm • xdm • tty を開くその他のネットワークサービス	root としてログインしないが、setuid またはその他のメカニズムを使用して管理タスクを実行するプログラム。以下のプログラムは root アカウントにアクセスできません。 • su • sudo • ssh • scp • sftp

Root SSH ログインの無効化

SSH プロトコルを介した root ログインを防ぐには、SSH デーモンの設定ファイル /etc/ssh/sshd_config を編集し、以下の行を変更します。

```
#PermitRootLogin yes
```

の行を以下のように変更します。

```
PermitRootLogin no
```

表48.3 Root SSH ログインの無効化

影響あり	影響なし
ツールの OpenSSH スイートを介した root アクセスを防ぎます。以下のプログラムは root アカウントにアクセスできません。 • ssh • scp • sftp	OpenSSH のツール群に含まれないプログラム。

PAM を使用して、サービスへの root アクセスを制限する

PAM は、/lib/security/pam_listfile.soモジュールを通じて、特定のアカウントを非常に柔軟に拒否することができます。管理者は、このモジュールを使用して、ログインを許可されていないユーザーのリストを参照できます。システムサービスへの root アクセスを制限するには、/etc/pam.d/ ディレクトリーのターゲットサービスの ファイルを編集し、pam_listfile.so モジュールが認証に必要であることを確認します。

以下は、/etc/pam.d/vsftpd PAM 設定ファイルの vsftpd FTP サーバーに対するモジュールの使用例です (ディレクティブが 1 行の場合、最初の行の最後の \ 文字は 不要 です)。

```
auth    required    /lib/security/pam_listfile.so    item=user \
 sense=deny file=/etc/vsftpd.ftpusers onerr=succeed
```

これにより、PAM は /etc/vsftpd.ftpusers ファイルを参照し、記載されているユーザーのサービスへのアクセスを拒否するように指示されます。管理者はこのファイルの名前を変更することができ、各サービスごとに個別のリストを保持することも、1 つの中央リストを使用して複数のサービスへのアクセスを拒否することもできます。

管理者が複数のサービスへのアクセスを拒否したい場合、同様の行を PAM 設定ファイル (メールクライアントの場合は /etc/pam.d/pop と /etc/pam.d/imap、SSH クライアントの場合は /etc/pam.d/ssh) に追加することができます。

PAM の詳細は、「PAM (プラグ可能な認証モジュール)」 を参照してください。

表48.4 PAM を使用した root の無効化

影響あり	影響なし

影響あり	影響なし
PAM が認識するネットワークサービスへの root アクセスを防ぎます。以下のサービスは、root アカウントにアクセスできません。 • login • gdm • kdm • xdm • ssh • scp • sftp • FTP クライアント • Email クライアント • すべての PAM 対応サービス	PAM を意識していないプログラム、サービス。

影響あり	影響なし

48.1.4.3. Root アクセスの制限

管理者は、root ユーザーへのアクセスを完全に拒否するのではなく、su や sudo などの setuid プログラムを介してのみアクセスを許可したい場合があります。

48.1.4.3.1. su コマンド

ユーザーが su コマンドを実行すると、root パスワードが求められ、認証後に root シェルプロンプトが表示されます。

su コマンドでログインすると、そのユーザーは root ユーザーに なり、システムへの絶対管理アクセスを持つことになります。[16]をクリックします。さらに、ユーザーが root になったら、パスワードを求められることなく、su コマンドを使用してシステム上の他のユーザーに変更を加えることができます。

このプログラムは非常に強力であるため、組織内の管理者はコマンドにアクセスできるユーザーを制限したい場合があります。

これを行う最も簡単な方法は、wheel と呼ばれる特別な管理グループにユーザーを追加することです。これを実行するには、以下のコマンドを root で入力します。

```
usermod -G wheel <username>
```

このコマンドで、< username> を wheel グループに追加するユーザー名に置き換えます。

また、以下のように User Manager を使用してグループメンバーシップを変更することもできます。この手順を実行するには、管理者権限が必要なことに注意してください。

1. パネル上の システム メニューをクリックして 管理 から ユーザーとグループ をクリックして User Manager を表示させます。または、シェルプロンプトでコマンド system-config-users を入力します。

2. ユーザー タブをクリックして、ユーザーリストの中から必要なユーザーを選択します。

3.
ツールバーの 設定 をクリックして、ユーザー設定のダイアログボックスを表示させます (または ファイル メニューで 設定 を選択します)。

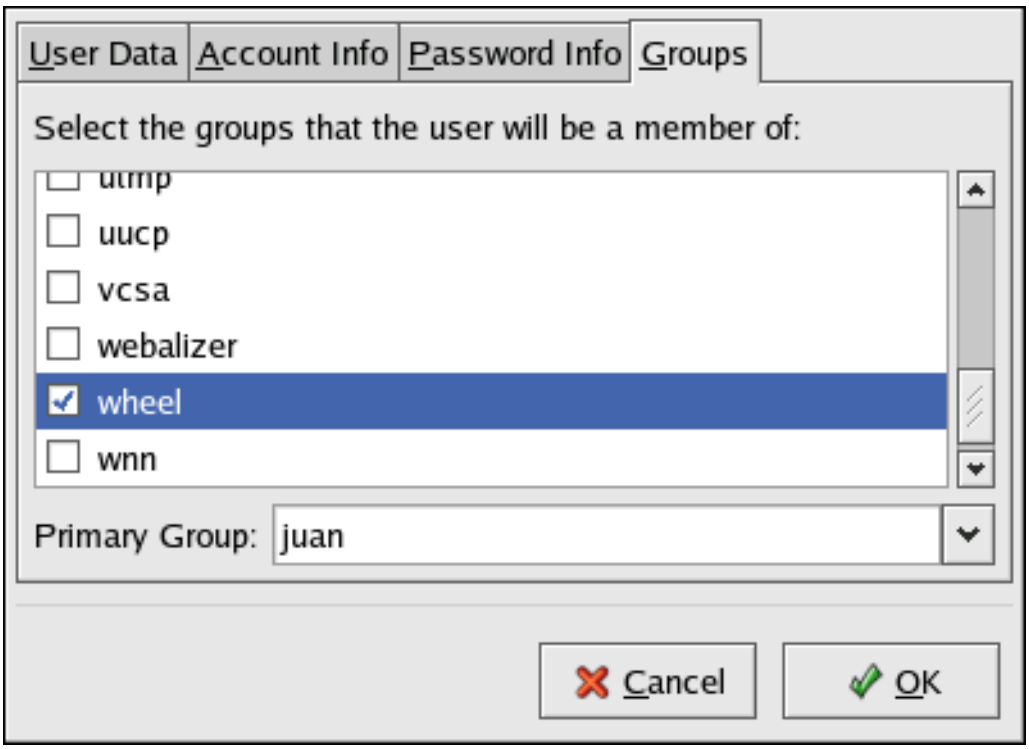
4.
グループ タブをクリックし、wheel グループのチェックボックスにチェックマークを付けて OK をクリックします。図48.2「wheel グループにユーザーを追加します。」 を参照してください。

5.
テキストエディターで su (/etc/pam.d/su)の PAM 設定ファイルを開き、以下の行からコメント # を削除します。

```
auth        required      pam_wheel.so use_uid
```

この変更により、wheel の管理グループメンバーのみが、su コマンドを使用して別のユーザーに切り替えることができます。

図48.2 wheel グループにユーザーを追加します。



[D]

注記

root ユーザーはデフォルトで wheel グループに含まれます。

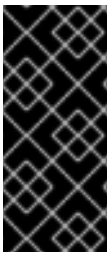
48.1.4.3.2. sudo コマンド

sudo コマンドは、ユーザーに管理アクセスを付与する別のアプローチを提供します。信頼されるユーザーが、管理コマンドの前に sudo を付けると、このユーザー自身 のパスワードが要求されます。ユーザーが認証され、コマンドが許可されると、管理コマンドは root 権限で実行されているかのように実行されます。

sudo コマンドの基本的な形式は次のとおりです。

```
sudo <command>
```

上記の例では、& lt;command& gt; は、通常は root ユーザー用に予約されたコマンド(mount など)に置き換えます。



重要な影響

sudoers は 5 分以内にパスワードを要求せずにコマンドを再度使用できるため、sudo コマンドのユーザーはマシンから接続する前にログアウトする細心の注意を払う必要があります。この設定は、設定ファイル /etc/sudoers を使用して変更できます。

sudo コマンドを使用すると、ハイレベルの柔軟性が可能になります。たとえば、/etc/sudoers 設定ファイルに記載されているユーザーのみが sudo コマンドの使用が許可され、root シェルではなく、ユーザーの シェルでコマンドが実行されます。これは、「Root アクセスの拒否」 に示されるように、root シェルを完全に無効にできることを意味します。

sudo コマンドは、包括的な監査証跡も提供します。認証が成功したたびに /var/log/messages ファイルに記録され、発行者のユーザー名とともに発行されたコマンドは /var/log/secure ファイルに記録されます。

sudo コマンドのもう 1 つの利点は、管理者がそれぞれのユーザーにニーズに応じて特定のコマンドへのアクセスを許可することができることです。

管理者が sudo 設定ファイル /etc/sudoers を編集する場合は、visudo コマンドを使用する必要があります。

誰かに完全な管理権限を付与するには、visudo と入力して、ユーザー権限の指定セクションに以下のような行を追加します。

```
juan ALL=(ALL) ALL
```

この例では、ユーザー juan が任意のホストから sudo を使用し、任意のコマンドを実行できます。

以下の例は、sudo を設定する際に可能な粒度を示しています。

```
%users  localhost=/sbin/shutdown -h now
```

この例が示しているのは、コンソールからであれば、どのユーザーが /sbin/shutdown -h コマンドを実行できることを示しています。

sudoers の man ページには、このファイルのオプションの詳細なリストがあります。

48.1.5. 利用可能なネットワークサービス

組織内のシステム管理者にとって、管理コントロールへのユーザーアクセスは重要な問題ですが、Linux システムを管理および運用するすべての人にとって、どのネットワークサービスがアクティブであるかを監視することは最も重要なことです。

Red Hat Enterprise Linux のサービスの多くは、ネットワークサーバーとして機能します。ネットワークサービスがマシン上で実行されている場合、サーバーアプリケーション (デーモン と呼ばれる) は、1 つ以上のネットワークポートで接続をリッスンしています。これらの各サーバーは、潜在的な攻撃経路として扱われる必要があります。

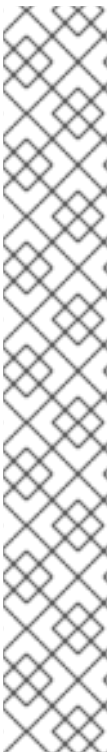
48.1.5.1. サービスへのリスク

ネットワークサービスは、Linux システムに多くのリスクをもたらす可能性があります。以下は、一部の主要な問題の一覧になります。

- サービス拒否攻撃 (DoS) — サービスにリクエストをフラッディングさせると、サービスは各リクエストをログに記録して応答しようとするため、サービス拒否攻撃はシステムを使用不能にすることができます。

- Script Vulnerability Attacks - サーバーが、Web サーバーが一般的に行うようにサーバー側のアクションを実行するスクリプトを使用している場合、クラッカーが不適切に記述されたスクリプトを攻撃する可能性があります。これらのスクリプトの脆弱性攻撃により、バッファーオーバーフロー状態が発生したり、攻撃者がシステム上のファイルを変更したりできます。

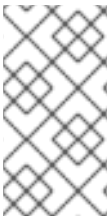
- *バッファーオーバーフロー攻撃 - 0 から 1023 までのポート番号が付いたポートに接続するサービスは、管理ユーザーとして実行する必要があります。アプリケーションに悪用可能なバッファーオーバーフローがある場合、攻撃者はデーモンを実行しているユーザーとしてシステムにアクセスできます。悪用可能なバッファーオーバーフローが存在するため、クラッカーは自動化ツールを使って脆弱性のあるシステムを特定し、アクセス権を獲得した後は、自動化ルートキットを使ってシステムへのアクセス権を維持します。*



注記

バッファーオーバーフローの脆弱性の脅威は、ExecShield によって Red Hat Enterprise Linux で軽減されます。これは、x86 互換のユニプロセッサーおよびマルチプロセッサーカーネルでサポートされる実行可能なメモリーのセグメンテーションおよび保護テクノロジーです。ExecShield は、仮想メモリーを実行可能セグメントと非実行セグメントに分離することで、バッファーオーバーフローのリスクを低減します。実行可能セグメントの外で実行しようとするプログラムコード (バッファーオーバーフローの悪用から注入された悪意のあるコードなど) は、セグメンテーションフォールトを引き起こし、終了します。

ExecShield には、AMD64 プラットフォームでの No eXecute (NX)テクノロジーのサポートと、Itanium システムおよび Intel® 64 システムの eXecute Disable (XD)テクノロジーのサポートも含まれています。これらのテクノロジーは ExecShield と連携して機能し、4KB の実行可能コードの粒度で仮想メモリーの実行可能な部分で悪意のあるコードが実行されるのを防ぐため、スパルドオーバーフローの悪用による攻撃のリスクが軽減されます。



ヒント

ネットワークに対する攻撃に対する公開を制限するには、未使用のすべてのサービスをオフにする必要があります。

48.1.5.2. サービスの識別と設定

セキュリティーを強化するために、Red Hat Enterprise Linux でインストールされるほとんどのネットワークサービスは、デフォルトでオフになっています。ただし、いくつかの注目すべき例外があります。

- *cupsd: Red Hat Enterprise Linux のデフォルトのプリントサーバー。*
- *L PD: 代替プリントサーバー。*
- *xinetd — gssftp や telnet などのさまざまな下位サーバーへの接続を制御するスーパー*

サーバー。

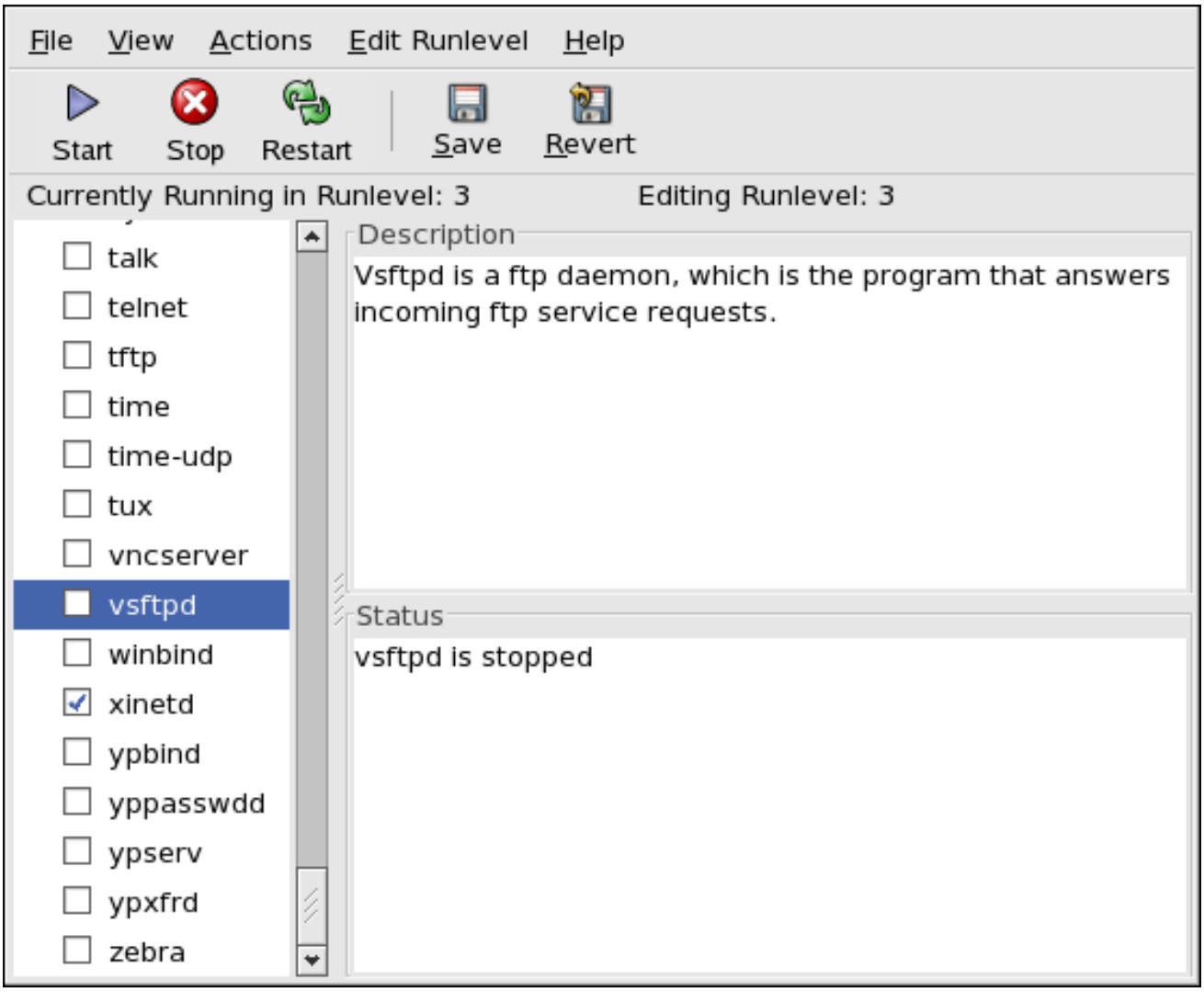
- Sendmail - Sendmail Mail Transport Agent (MTA)はデフォルトで有効になっていますが、からの接続はリッスンしません。 localhostをクリックします。

- sshd — OpenSSH サーバー。Telnet に代わる安全なサーバー。

これらのサービスを実行したままにするかどうかを決定する際には、慎重に一般的な意味とエラーを使用することが推奨されます。たとえば、プリンターが利用できない場合は、cupsd を実行したままにします。portmap の場合も同様です。NFSv3 ボリュームをマウントしないか、NIS (ypbind サービス)を使用しない場合は、portmap を無効にする必要があります。

Red Hat Enterprise Linux には、サービスのオン/オフを切り替えるように設計された 3 つのプログラムが同梱されています。これらは、サービス設定ツール (system-config-services)、ntsysv、および chkconfig です。これらのツールの使用方法は、18章 を参照してください。

図48.3 サービス設定ツール



[D]

特定のサービスの目的が不明な場合は、Services Configuration Tool には 図48.3「サービス設定ツール」 の説明フィールドがあり、追加情報が提供されます。

起動時に起動できるネットワークサービスの確認は、ストーリーの一部のみです。また、どのポートが開いていてリッスンしているかも確認する必要があります。詳細は、「リッスンしているポートの確認」 を参照してください。

48.1.5.3. 安全でないサービス

潜在的に、どのようなネットワークサービスも安全ではありません。そのため、使っていないサービスをオフにすることはとても重要です。サービスの不正使用は定期的に確認され、パッチが適用されるため、ネットワークサービスに関連するパッケージを定期的に更新することが非常に重要です。詳細は、「セキュリティー更新」 を参照してください。

一部のネットワークプロトコルは、本質的に他のプロトコルよりも安全ではありません。以下の動

作を実行するサービスがそれに当たります。

- 暗号化されていないネットワーク上でのユーザー名とパスワードの送信 — Telnet や FTP などの多くの古いプロトコルは、認証セッションを暗号化しないため、可能な限り避ける必要があります。

- 暗号化されていないネットワーク上での機密データの送信 — 多くのプロトコルは、暗号化されていないネットワーク上でデータを送信します。これらのプロトコルには、Telnet、FTP、HTTP、および SMTP が含まれます。NFS や SMB などの多くのネットワークファイルシステムも、暗号化されていないネットワークを介して情報を送信します。これらのプロトコルを使用する場合、ユーザーの責任において、送信されるデータの種類を制限する必要があります。

  netdump などのリモートメモリーダンプサービスは、暗号化されていないネットワーク上でメモリーの内容を送信します。メモリーダンプにはパスワードや、さらに悪いデータベースエントリーやその他の機密情報を含めることができます。

  finger や rwhod などの他のサービスは、システムのユーザーに関する情報を表示します。

本質的に安全でないサービスの例として、rlogin、rsh、telnet、vsftpd があります。

SSH を優先して、すべてのリモートログインおよびシェルプログラム(rlogin、rsh、telnet)を回避する必要があります。sshd の詳細は、「セキュリティー強化通信ツール」 を参照してください。

FTP は、リモートシェルとしてのシステムのセキュリティーにとって本質的に危険ではありませんが、問題を回避するために FTP サーバーは慎重に設定し監視する必要があります。FTP サーバーのセキュリティー保護に関する詳細は、「FTP のセキュア化」 を参照してください。

慎重に実装し、ファイアウォールの内側に置くべきサービスは以下の通りです。

- finger

- authd （これは以前の Red Hat Enterprise Linux リリースで identd と呼ばれていました）

- *netdump*
- *netdump-server*
- *nfs*
- *rwhod*
- *sendmail*
- *SMB (Samba)*
- *yppasswdd*
- *ypserv*
- *ypxfrd*

ネットワークサービスのセキュリティー保護に関する詳細は、「サーバーセキュリティー」 を参照してください。

次のセクションでは、簡単なファイアウォールを設定するために使用できるツールについて説明します。

48.1.6. 個人ファイアウォール

必要な ネットワークサービスを設定したら、ファイアウォールを実装することが重要です。

重要な影響

必要なサービスを設定し、ファイアウォールを実装するしてから、インターネットまたは信頼しない他のネットワークに接続する必要があります。

ファイアウォールにより、ネットワークパケットがシステムのネットワークインターフェイスにアクセスできなくなります。ファイアウォールによってブロックされたポートにリクエストが行われた場合、要求は無視されます。ブロックされたポートの 1 つをサービスがリッスンしている場合、パケットは受信されず、事実上無効になります。このため、設定されたサービスによって使用されるポートへのアクセスをブロックせずに、ファイアウォールを設定して使用されていないポートへのアクセスをブロックする場合は注意が必要です。

ほとんどのユーザーは、Red Hat Enterprise Linux に同梱されるグラフィカルなファイアウォール設定ツールである Security Level Configuration Tool (system-config-securitylevel)に最適なツールになります。このツールは、コントロールパネルインターフェイスを使用して汎用ファイアウォール用に幅広い iptables ルールを作成します。

このアプリケーションの使用方法と利用可能なオプションは、「ファイアウォールの基本設定」 を参照してください。

高度なユーザーおよびサーバー管理者は、iptables でファイアウォールを手動で設定することが適切なオプションです。詳細は、「ファイアウォール」 を参照してください。iptables コマンドの包括的なガイドは、「iptables」 を参照してください。

48.1.7. セキュリティー強化通信ツール

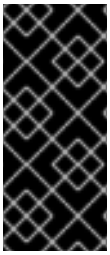
インターネットのサイズと人気が大きくなるにつれ、通信インターセプションの脅威があります。長年にわたり、ネットワーク上で転送される通信を暗号化するツールを開発しました。

Red Hat Enterprise Linux には、ネットワークを通過する情報を保護するために、高レベルの公開鍵暗号暗号アルゴリズムを使用する 2 つの基本ツールが同梱されています。

- OpenSSH: ネットワーク通信を暗号化する SSH プロトコルのフリー実装。
- GNU Privacy Guard (GPG): データを暗号化する PGP (Pretty Good Privacy)暗号化アプリケーションの無料実装。

OpenSSH は、リモートマシンにアクセスするより安全な方法で、telnet や rsh などの以前の暗号化されていないサービスを置き換えます。OpenSSH には、sshd と呼ばれるネットワークサービスと、コマンドラインクライアントアプリケーション 3 つが含まれています。

- ssh: セキュアなリモートコンソールアクセスクライアント。
- scp: セキュアなリモートコピーコマンド。
- SFTP: インタラクティブなファイル転送セッションを可能にするセキュアな擬似ftpクライアント。



重要な影響

sshd サービスは本質的に安全ですが、セキュリティーの脅威を防ぐために、このサービスを最新の状態に維持する 必要 があります。詳細は、「セキュリティー更新」を参照してください。

GPG は、プライベートメール通信を保証する 1 つの方法です。パブリックネットワークを介して機密データを電子メールで送信したり、ハードドライブの機密データを保護するためにも使用できます。

48.2. サーバーセキュリティー

システムがパブリックネットワークのサーバーとして使用されると、攻撃のターゲットになります。システムを強化し、サービスをロックダウンすることは、システム管理者にとって最も重要なことです。

特定の問題を解決する前に、サーバーセキュリティーの強化に関する以下の一般的なヒントを確認してください。

- 最新の脅威から保護するために、すべてのサービスを最新の状態に保ちます。
- 可能な限りセキュアなプロトコルを使用します。
- 可能な場合は、マシンごとに 1 つのタイプのネットワークサービスのみを提供します。

- 疑わしいアクティビティーのために、すべてのサーバーを慎重に監視します。

48.2.1. TCP Wrapper と xinetd を使用したサービスの保護

TCP Wrapper は、さまざまなサービスにアクセス制御を提供します。SSH、Telnet、FTP などの最新のネットワークサービスのほとんどは、受信要求と要求されたサービスの間の保護を表す TCP Wrappers を利用します。

TCP Wrapper が提供する利点は、xinetd と併用すると、追加のアクセス、ロギング、バインディング、リダイレクト、およびリソース使用状況の制御を提供するスーパーサーバーである xinetd と併用すると強化されます。

ヒント

iptables ファイアウォールルールを TCP Wrapper と xinetd と併用して、サービスアクセス制御内に冗長性を作成することが推奨されます。iptables コマンドでのファイアウォールの実装に関する詳細は、「ファイアウォール」 を参照してください。

TCP Wrapper および xinetd の設定に関する詳細は、「」 を参照してください。

以下のサブセクションでは、各トピックに関する基本的な知識と、特定のセキュリティーオプションに焦点を当てていることを前提としています。

48.2.1.1. TCP Wrapper を使用したセキュリティーの強化

TCP Wrapper の機能は、サービスへのアクセスを拒否するだけではありません。このセクションでは、これらを使用して接続バナーを送信し、特定のホストからの攻撃を警告し、ロギング機能を強化する方法を説明します。TCP Wrapper 機能と制御言語の詳細は、hosts_options の man ページを参照してください。

48.2.1.1.1. TCP Wrapper と接続バナー

ユーザーがサービスに接続する際に適切なバナーを表示することは、潜在的な攻撃者に対して、システム管理者が警戒していることを知らせる良い方法です。システムに関するどの情報をユーザーに表示するかを制御することもできます。サービスに TCP Wrapper バナーを実装するには、banner オプションを使用します。

以下の例では、vsftpdにバナーを導入します。最初にバナーファイルを作成します。これは、シス

テム上のどこにあってもかまいませんが、デーモンと同じ名前である必要があります。この例では、ファイルは /etc/banners/vsftpd と呼ばれ、以下の行が含まれます。

```
220-Hello, %c
220-All activity on ftp.example.com is logged.
220-Inappropriate use will result in your access privileges being removed.
```

%c トークンは、ユーザー名、ホスト名、ユーザー名および IP アドレスなどのさまざまなクライアント情報を提供し、接続をより意図した状態にします。

このバナーを受信接続に表示するには、/etc/hosts.allow ファイルに次の行を追加します。

```
vsftpd : ALL : banners /etc/banners/
```

48.2.1.1.2. TCP Wrapper と攻撃警告

特定のホストまたはネットワークがサーバーを攻撃していることが検出された場合、TCP Wrapper を使用して、spawn ディレクティブを使用したそのホストまたはネットワークからの後続の攻撃について管理者に警告できます。

この例では、206.182.68.0/24 ネットワークからサーバーを攻撃しようとするクラッカーが検出されたと仮定します。/etc/hosts.deny ファイルに次の行を配置して、そのネットワークからの接続試行を拒否し、その試行を特別なファイルに記録します。

```
ALL : 206.182.68.0 : spawn /bin/ 'date' %c %d >> /var/log/intruder_alert
```

%d トークンは、攻撃者がアクセスしようとしたサービスの名前を提供します。

接続を許可してログに記録するには、spawn ディレクティブを /etc/hosts.allow ファイルに配置します。

注記

spawn ディレクティブはシェルコマンドを実行するため、特定のクライアントがサーバーに接続しようとすると、管理者に通知するか、コマンドチェーンを実行する特別なスクリプトを作成します。

48.2.1.1.3. TCP Wrapper とロギングの強化

特定のタイプの接続が他のタイプよりも懸念される場合は、重大度 オプションを使用して、そのサービスのログレベルを上げることができます。

この例では、FTP サーバーのポート 23 (Telnet ポート) に接続しようとしている人はクラッカーであると想定します。これを示すには、ログファイルにデフォルトのフラグ info の代わりに emerg フラグを配置し、接続を拒否します。

これを行うには、次の行を /etc/hosts.deny に配置します。

```
in.telnetd : ALL : severity emerg
```

これは、デフォルトの authpriv ログ機能を使用しますが、優先度をデフォルト値の info から emerg に上げます。これにより、ログメッセージがコンソールに直接送信されます。

48.2.1.2. xinetd でのセキュリティーの強化

本セクションでは、xinetd を使用してトラップサービスを設定し、それを使用して任意の xinetd サービスで利用可能なリソースレベルを制御することに重点を置いています。サービスのリソース制限を設定すると、DoS(Denial of Service)攻撃に役立ちます。利用可能なオプションの一覧は、xinetd および xinetd.conf の man ページを参照してください。

48.2.1.2.1. トレイトの設定

xinetd の重要な機能の 1 つは、ホストをグローバル no_access リストに追加する機能です。このリストのホストは、指定された期間、または xinetd が再起動するまで、xinetd が管理するサービスへの後続の接続を拒否します。これは、SENSOR 属性を使用して実行できます。これは、サーバー上のポートをスキャンしようとするホストを簡単にブロックする方法です。

SENSOR の設定の最初の手順は、使用を計画しないサービスを選択することです。この例では、Telnet が使用されます。

/etc/xinetd.d/telnet ファイルを編集し、フラグ の行を次のように変更します。

```
flags          = SENSOR
```

以下の行を追加します。

```
deny_time      = 30
```

これにより、そのホストによる 30 分間接続試行が拒否されます。deny_time 属性の他の許容値は FOREVER です。これは、xinetd が再起動するまで ban を有効になり、NEVER は接続を許可し、ログに記録します。

最後に、最後の行が表示されるはずです。

```
disable       = no
```

これにより、トラップ自体が有効になります。

SENSOR の使用は、望ましくないホストからの接続を検出および停止するのに適した方法として、2 つの欠点があります。

- Stealth スキャンでは機能しません。
- SENSOR が実行されていることがわかっている攻撃者は、IP アドレスを改ざんし、禁止されたポートに接続することで、特定のホストに対してサービス拒否攻撃をマウントすることができます。

48.2.1.2.2. サーバーリソースの制御

xinetd のもう 1 つの重要な機能は、その制御下でサービスのリソース制限を設定する機能です。

これは、以下のディレクティブを使用して行います。

- CPs = <number_of_connections> <wait_period> - 受信接続のレートを制限します。このディレクティブは 2 つの引数を取ります。
  - <number_of_connections >: 処理する 1 秒あたりの接続数。受信接続のレートがこれよりも大きい場合、サービスは一時的に無効になります。デフォルト値は 15 (50)です。
  - <wait_period >: 無効にした後にサービスを再度有効にするまで待機する秒数。デ

フォルトの間隔は 10 秒です。

- instances = <number_of_connections> - サービスに対して許可される接続の総数を指定します。このディレクティブは、整数値または UNLIMITED を受け入れます。

- per_source = <number_of_connections>: 各ホストがサービスに対して許可される接続の数を指定します。このディレクティブは、整数値または UNLIMITED を受け入れます。

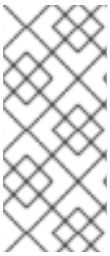
- rlimit_as = <number[K|M]>: サービスがキロバイトまたはメガバイトで占有できるメモリーアドレス空間の量を指定します。このディレクティブは、整数値または UNLIMITED を受け入れます。

- rlimit_cpu = <number_of_seconds& gt;: サービスが CPU を占有する時間を秒単位で指定します。このディレクティブは、整数値または UNLIMITED を受け入れます。

これらのディレクティブを使用すると、1 つの xinetd サービスがシステムに圧倒されなくなり、サービス拒否が発生する可能性があります。

48.2.2. ポートマップのセキュリティー保護

portmap サービスは、NIS や NFS などの RPC サービス用の動的ポート割り当てデーモンです。認証メカニズムが弱く、制御するサービスに幅広いポート範囲を割り当てることができます。これらの理由から、セキュリティー保護することは困難です。



注記

NFSv4 では必要がなくなったため、ポートマップのセキュリティー保護は NFSv2 および NFSv3 の実装にのみ影響します。NFSv2 または NFSv3 サーバーを実装する予定の場合は、portmap が必要であり、以下のセクションが適用されます。

RPC サービスを実行している場合は、以下の基本的なルールにしたがってください。

48.2.2.1. TCP Wrapper によるポートマップの保護

TCP Wrapper を使用して portmap サービスにアクセスできるネットワークやホストを制限することが重要です。認証形式が組み込まれていないためです。

また、サービスへのアクセスを制限する場合は、IP アドレス のみ を使用してください。ホスト名は、DNS ポイズニングやその他の方法で偽装できるため、使用しないようにしてください。

48.2.2.2. iptables によるポートマップの保護

portmap サービスへのアクセスをさらに制限するには、サーバーに iptables ルールを追加し、特定のネットワークへのアクセスを制限することが推奨されます。

以下は、iptables コマンドの 2 つの例です。1 つ目は、192.168.0.0/24 ネットワークからポート 111 (portmap サービスにより使用)への TCP 接続を許可します。2 つ目は、localhost から同じポートへの TCP 接続を許可します。これは、Nautilus が使用する sgi_fam サービスに必要です。他のパケットはすべて遮断されます。

```
iptables -A INPUT -p tcp -s! 192.168.0.0/24 --dport 111 -j DROP
iptables -A INPUT -p tcp -s 127.0.0.1 --dport 111 -j ACCEPT
```

同様に UDP トラフィックを制限するには、次のコマンドを使用します。

```
iptables -A INPUT -p udp -s! 192.168.0.0/24 --dport 111 -j DROP
```

ヒント

iptables コマンドでのファイアウォールの実装に関する詳細は、「ファイアウォール」 を参照してください。

48.2.3. NIS のセキュア化

ネットワーク情報 サービス(NIS)は、ypserv と呼ばれる RPC サービスで、portmap およびその他の関連サービスとともに使用して、ユーザー名、パスワード、およびその他の機密情報のマップを、そのドメイン内にあるように要求するコンピューターに配布します。

NIS サーバーは、複数のアプリケーションで設定されています。内容は以下の通りです。

- /usr/sbin/rpc.yppasswdd — yppasswdd サービスとも呼ばれるこのデーモンを使用すると、ユーザーは NIS パスワードを変更できます。
- /usr/sbin/rpc.ypxfrd — ypxfrd サービスとも呼ばれるこのデーモンは、ネットワークを介した NIS マップ転送を担当します。

- */usr/sbin/yppush:* このアプリケーションは、変更した *NIS* データベースを複数の *NIS* サーバーに伝播します。

- */usr/sbin/ypserv* — これは、*NIS* サーバーデーモンです。

NIS は現在の基準からすると、やや安全性に欠けています。ホスト認証メカニズムがなく、パスワードハッシュを含むすべての情報を暗号化せずにネットワーク経由で送信します。そのため、*NIS* を使用するネットワークを構築する際には、細心の注意が必要です。さらに、*NIS* のデフォルト設定が本質的に安全でないという事実が、この問題をさらに複雑にしています。

NIS サーバーの実装を計画している方は、「ポートマップのセキュリティー保護」 で説明されているように、最初に *portmap* サービスをセキュリティー保護し、ネットワーク計画などの次の問題に対処することが推奨されます。

48.2.3.1. ネットワークの注意深いプランニング

NIS はネットワーク上で機密情報を暗号化せずに送信するため、サービスをファイアウォールの内側で、そしてセグメント化された安全なネットワーク上で実行することが重要となります。*NIS* 情報が安全でないネットワークを介して送信される場合は常に、傍受されるリスクがあります。慎重なネットワーク設計は、深刻なセキュリティー侵害を防ぐ上で役立ちます。

48.2.3.2. パスワードのような NIS ドメイン名とホスト名の使用

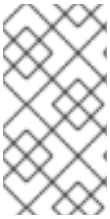
ユーザーが *NIS* サーバーの *DNS* ホスト名と *NIS* ドメイン名を知っている限り、*NIS* ドメイン内のすべてのマシンは、コマンドを使用して認証なしでサーバーから情報を抽出できます。

たとえば、誰かがノートパソコンをネットワークに接続するか、外部からネットワークに侵入した場合 (そして、内部 *IP* アドレスをスプーフィングした場合)、以下のコマンドで */etc/passwd* マップが表示されます。

```
ypcat -d <NIS_domain> -h <DNS_hostname> passwd
```

この攻撃者が *root* ユーザーの場合、以下のコマンドを入力することで */etc/shadow* ファイルを取得することができます。

```
ypcat -d <NIS_domain> -h <DNS_hostname> shadow
```



注記

Kerberos を使用する場合、/etc/shadow ファイルは NIS マップ内に保存されません。

攻撃者が NIS マップへのアクセスを困難にするには、DNS ホスト名に o7hfawtgmhwg.domain.com などのランダムな文字列を作成します。同様に、異なるランダムな NIS ドメイン名を作成します。これにより、攻撃者が NIS サーバーにアクセスすることがより困難になります。

48.2.3.3. /var/yp/securenets ファイルを編集する

/var/yp/securenetsファイルが空白または存在しない場合 (デフォルトのインストール後の場合)、NIS はすべてのネットワークをリッスンします。まず最初にすべきことは、ネットマスクとネットワークのペアをファイルに記述し、ypservが適切なネットワークからのリクエストにのみ反応するようにすることです。

以下は、/var/yp/securenets ファイルからのエントリーの例です。

```
255.255.255.0     192.168.0.0
```

WARNING

/var/yp/securenets ファイルを作成せずに、NIS サーバーを初めて起動しないでください。

この手法では、IP スプーフィング攻撃からの保護はできませんが、少なくとも NIS サーバーがサービスを提供するネットワークに制限を設けることができます。

48.2.3.4. 静的ポートの割り当てと iptables ルールの使用

NIS に関連するすべてのサーバーは、ユーザーがログインパスワードを変更するためのデーモンであるrpc.yppasswdd を除いて、特定のポートを割り当てることができます。他の 2 つの NIS サーバーデーモンであるrpc.ypsfrdとypservにポートを割り当てることで、NIS サーバーデーモンを侵入者からさらに保護するためのファイアウォールルールを作成することができます。

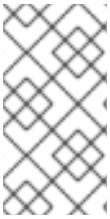
これを行うには、/etc/sysconfig/network に以下の行を追加します。

```
YPSERV_ARGS="-p 834" YPXFRD_ARGS="-p 835"
```

次に、以下の iptables ルールを使用して、サーバーがこれらのポートをリッスンするネットワークを強制できます。

```
iptables -A INPUT -p tcp -s! 192.168.0.0/24 --dport 834 -j DROP
iptables -A INPUT -p tcp -s! 192.168.0.0/24 --dport 835 -j DROP
iptables -A INPUT -p udp -s! 192.168.0.0/24 --dport 834 -j DROP
iptables -A INPUT -p udp -s! 192.168.0.0/24 --dport 835 -j DROP
```

これは、192.168.0.0/24 ネットワークから要求が送信された場合にのみ、サーバーはポート 834 および 835 への接続を許可することを意味します。



ヒント

iptables コマンドでのファイアウォールの実装に関する詳細は、「ファイアウォール」 を参照してください。

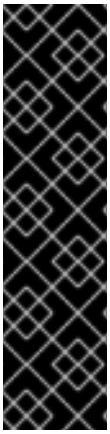
48.2.3.5. Kerberos 認証の使用

NIS を認証に使用する際に考慮すべき問題の 1 つは、ユーザーがマシンにログインするたびに、/etc/shadow マップからのパスワードハッシュがネットワーク経由で送信されることです。侵入者が NIS ドメインにアクセスしてネットワークトラフィックを盗んだ場合は、ユーザー名とパスワードのハッシュを収集できます。十分な時間があれば、パスワードクラッキングプログラムは弱いパスワードを推測することができ、攻撃者はネットワーク上の有効なアカウントにアクセスすることができます。

Kerberos は秘密鍵暗号を使用しているので、パスワードハッシュがネットワーク経由で送信されることはなく、システムの安全性が大幅に向上します。Kerberos の詳細は、「Kerberos」 を参照してください。

48.2.4. NFS のセキュア化

ネットワーク ファイルシステム(NFS)は、クライアントマシンにネットワークにアクセスできるファイルシステムを提供するサービスです。NFS の詳細は、21章Network File System (NFS) を参照してください。以下のサブセクションでは、NFS に関する基本的な知識を想定しています。



重要な影響

Red Hat Enterprise Linux (NFSv4)に含まれる NFS のバージョンでは、「ポートマップのセキュリティー保護」 で概説されているように portmap サービスが不要になりました。NFS トラフィックは、UDP ではなくすべてのバージョンで TCP を使用し、NFSv4 を使用する際に要求するようになりました。NFSv4 には、RPCSEC_GSS カーネルモジュールの一部として、Kerberos ユーザーおよびグループ認証が含まれるようになりました。Red Hat Enterprise Linux は NFSv2 および NFSv3 をサポートしているため、portmap に関する情報は引き続き含まれています。いずれも portmap を使用します。

48.2.4.1. ネットワークの注意深いプランニング

NFSv4 は、Kerberos を使用して暗号化されたすべての情報をネットワーク経由で渡すことができるので、ファイアウォールの背後にある場合やセグメント化されたネットワーク上にある場合は、サービスを正しく設定することが重要です。NFSv2 と NFSv3 は引き続きデータを安全に渡さないため、この点を考慮する必要があります。これらの点はすべて、ネットワーク設計に注意を払うことで、セキュリティー違反を防ぐことができます。

48.2.4.2. 構文エラーに注意

NFS サーバーは、/etc/exports ファイルを参照して、エクスポートするファイルシステムとこれらのディレクトリーをエクスポートするホストを決定します。このファイルを編集する際には、余計なスペースを加えないように注意してください。

たとえば、/etc/exports ファイルの次の行は、ディレクトリー /tmp/nfs/ を読み取り/書き込みパーミッションを持つホスト bob.example.com と共有しています。

```
/tmp/nfs/     bob.example.com(rw)
```

一方、/etc/exports ファイルの以下の行は、同じディレクトリーを読み取り専用パーミッションでホスト bob.example.com と共有し、ホスト名の後の 1 つのスペース文字が原因で、読み取り/書き込み権限で すべてのユーザー と共有します。

```
/tmp/nfs/     bob.example.com (rw)
```

showmount コマンドを使用して、設定されている NFS 共有をチェックし、共有されているものを確認することをお勧めします。

```
showmount -e <hostname>
```

48.2.4.3. no_root_squash オプションは使用しないでください。

デフォルトでは、NFS 共有は、root ユーザーを非特権ユーザーアカウントである nfsnobody ユーザーに変更します。これにより、root で作成されたすべてのファイルの所有者が nfsnobody に変更され、setuid ビットが設定されたプログラムのアップロードができなくなります。

no_root_squash を使用すると、リモート root ユーザーは、共有ファイルシステム上の任意のファイルを変更し、他のユーザーが誤って実行するようにアプリケーションをトロイの木馬に感染したままにすることができます。

48.2.5. Apache HTTP Server のセキュリティー保護

Apache HTTP Server は、Red Hat Enterprise Linux に同梱される最も安定したセキュアなサービスの 1 つです。Apache HTTP Server を保護するための多数のオプションと手法を利用することができます。数が多いため、ここでは詳細な説明はしません。

Apache HTTP Server を設定する場合は、アプリケーションで利用可能なドキュメントを読むことが重要です。これには、25章Apache HTTP サーバー と、http://www.redhat.com/docs/manuals/stronghold/ で利用可能な Stronghold マニュアルが含まれます。

システム管理者は、次の設定オプションを使用する際に注意する必要があります。

48.2.5.1. FollowSymLinks

このディレクティブはデフォルトで有効になっていますので、Web サーバーのドキュメント root へのシンボリックリンクを作成する場合は注意が必要です。たとえば、/ へシンボリックリンクを提供することはお勧めできません。

48.2.5.2. インデックス のディレクティブ

このディレクティブはデフォルトで有効になっていますが、望ましくない場合もあります。訪問者がサーバー上のファイルを閲覧できないようにするには、このディレクティブを削除してください。

48.2.5.3. UserDir ディレクティブ

UserDir ディレクティブは、システムにユーザーアカウントが存在することを確認できるため、デフォルトでは無効になっています。サーバーでユーザーディレクトリーの閲覧を可能にするには、以下のディレクティブを使用します。

```
UserDir enabled
UserDir disabled root
```

これらのディレクティブは、/root/ 以外のすべてのユーザーディレクトリーの閲覧を有効にします。無効化されたアカウントの一覧にユーザーを追加するには、UserDir disabled 行にスペースで区切られたユーザーの一覧を追加します。

48.2.5.4. IncludesNoExec ディレクティブを削除しないでください。

デフォルトでは、Server-Side Includes (SSI) モジュールは、コマンドを実行できません。この設定は、攻撃者がシステム上でコマンドを実行できるようにする可能性があるため、絶対に必要な場合を除き変更しないことが推奨されます。

48.2.5.5. 実行可能なディレクトリーのパーミッションの制限

root ユーザーのみが、スクリプトまたは CGI を含むディレクトリーへの書き込み権限を持っていることを確認してください。これを行うには、以下のコマンドを入力します。

```
chown root <directory_name>
chmod 755 <directory_name>
```

重要な影響

システム上で実行されているスクリプトは、実稼働させる 前に 必ず意図したとおりに動作することを確認してください。

48.2.6. FTP のセキュア化

ファイル転送プロトコル (FTP) は、ネットワーク上でファイルを転送するために設計された古い TCP プロトコルです。ユーザー認証を含むサーバーとのすべてのトランザクションが暗号化されていないため、安全でないプロトコルとみなされ、慎重に設定される必要があります。

Red Hat Enterprise Linux は 3 つの FTP サーバーを提供します。

- gssftpd - Kerberos 対応の xinetd- ネットワーク経由で認証情報を送信しない ベースの FTP デーモン。
- Red Hat Content Accelerator (tux) — FTP 機能を持つカーネルスペースの Web サーバー。

- **vsftpd** — スタンドアロンの、セキュリティー指向の FTP サービスの実装。

vsftpd FTP サービスをセットアップするためのセキュリティーガイドラインを以下に示します。

48.2.6.1. FTP グリーティングバナー

ユーザー名とパスワードを送信する前に、すべてのユーザーにグリーティングバナーが表示されます。デフォルトでは、このバナーには、システムの弱点を特定しようとするクラッカーに有用なバージョン情報が含まれています。

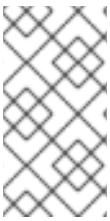
vsftpd のグリーティングバナーを変更するには、/etc/vsftpd/vsftpd.conf ファイルに次のディレクティブを追加します。

```
ftpd_banner=<insert_greeting_here>
```

上記のディレクティブの **<insert_greeting_here>** をグリーティングメッセージのテキストで置き換えます。

複数行バナーの場合は、バナーファイルを使用することが推奨されます。複数のバナーの管理を簡単にするために、すべてのバナーを /etc/banners/ という新しいディレクトリーに配置します。この例の FTP 接続のバナーファイルは /etc/banners/ftp.msg です。以下は、このようなファイルの例です。

```
########## # Hello, all activity on ftp.example.com is logged. ##########
```



注記

「TCP Wrapper と接続バナー」 で指定されているように、ファイルの各行を 220 で始める必要はありません。

vsftpd のこのグリーティングバナーファイルを参照するには、以下のディレクティブを /etc/vsftpd/vsftpd.conf ファイルに追加します。

```
banner_file=/etc/banners/ftp.msg
```

重要な影響

/etc/vsftpd/vsftpd.conf でバナーファイルへのパスを正しく指定しないと、vsftpd に接続しようとすると接続が即座に閉じられ、500 OOPS: cannot open banner < path_to_banner_file > エラーメッセージが表示されます。

/etc/vsftpd/vfsftpd.conf の banner_file ディレクティブは、設定ファイルの ftpd_banner ディレクティブよりも優先されることに注意してください。banner_file が指定されている場合は、ftpd_banner は無視されます。

また、「TCP Wrapper と接続バナー」 で説明されているように、TCP Wrappers を使用して着信接続に追加のバナーを送信することもできます。

48.2.6.2. Anonymous Access

/var/ftp/ ディレクトリーが存在すると、匿名アカウントが有効になります。

このディレクトリーを作成する最も簡単な方法は、vsftpd パッケージをインストールすることです。本パッケージは、匿名ユーザーのためのディレクトリーツリーを構築し、匿名ユーザーのためにディレクトリーのパーミッションを読み取り専用に設定します。

デフォルトでは、匿名ユーザーはどのディレクトリーにも書き込むことができません。

注意

FTP サーバーへの匿名アクセスを可能にする場合、機密データが保存される場所に注意してください。

48.2.6.2.1. 匿名のアップロード

匿名ユーザーがファイルをアップロードできるようにするため、/var/ftp/pub/ 内に書き込み専用のディレクトリーを作成することを推奨します。

これを行うには、以下のコマンドを入力します。

```
mkdir /var/ftp/pub/upload
```

次に、匿名ユーザーがディレクトリーの内容を閲覧できないように、パーミッションを変更します。

```
chmod 730 /var/ftp/pub/upload
```

ディレクトリーの長い形式のリストは、次のようになります。

```
drwx-wx---    2 root     ftp          4096 Feb 13 20:05 upload
```

WARNING

匿名ユーザーにディレクトリーの読み取り/書き込みを許可すると、サーバーが盗まれたソフトウェアのリポジトリーになる可能性があります。

また、**vsftpd** の下に、以下の行を **/etc/vsftpd/vsftpd.conf** ファイルに追加します。

```
anon_upload_enable=YES
```

48.2.6.3. ユーザーアカウント

FTP は、認証用の安全でないネットワーク上で暗号化されていないユーザー名とパスワードを送信するため、ユーザーアカウントからサーバーへのシステムユーザーアクセスを拒否することが推奨されます。

vsftpd のすべてのユーザーアカウントを無効にするには、次のディレクティブを **/etc/vsftpd/vsftpd.conf** に追加します。

```
local_enable=NO
```

48.2.6.3.1. ユーザーアカウントの制限

root ユーザーや sudo 権限を持つアカウントなど、特定のアカウントまたは特定のアカウントグループに対する FTP アクセスを無効にするには、「Root アクセスの拒否」 で説明されているように PAM リストファイルを使用するのが最も簡単な方法です。vsftpd の PAM 設定ファイルは /etc/pam.d/vsftpd です。

また、各サービスでユーザーアカウントを直接無効にすることもできます。

vsftpd で特定のユーザーアカウントを無効にするには、ユーザー名を /etc/vsftpd.ftpusersに追加します。

48.2.6.4. TCP Wrapper を使用してアクセスを制御する

TCP Wrapper を使用して、「TCP Wrapper を使用したセキュリティーの強化」 で概説されているように、いずれかの FTP デーモンへのアクセスを制御します。

48.2.7. Sendmail のセキュア化

Sendmail は、Simple Mail Transport Protocol (SMTP)を使用して他の MTA 間で電子メッセージを配信し、クライアントまたは配信エージェントに電子メッセージを配信する Mail Transport Agent (MTA)です。多くの MTA は相互にトラフィックを暗号化することが可能ですが、ほとんどの場合は暗号化しません。そのため、パブリックネットワークを介して電子メールを送信することは、本質的に安全でない通信形式と見なされます。

メールの仕組みおよび一般的な設定の概要については、27章メール を参照してください。本セクションでは、/etc/mail/sendmail.mc を編集し、m4 コマンドを使用して、有効な /etc/mail/sendmail.cf を生成する基本的な知識を想定しています。

Sendmail サーバーの実装を計画しているユーザーは、以下の問題に対処することが推奨されます。

48.2.7.1. サービス拒否攻撃を制限する

電子メールの性質上、断固とした攻撃者は、サーバーを非常に簡単にメールで溢れさせ、サービス拒否を引き起こすことができます。/etc/mail/sendmail.mc で以下のディレクティブに制限を設定することで、このような攻撃の効果は制限されます。

- confCONNECTION_RATE_THROTTLE: サーバーが 1 秒あたり受信できる接続の数。デフォルトでは、Sendmail は接続の数を制限しません。制限が設定され、到達すると、追加の

接続が遅延します。

- confMAX_DAEMON_CHILDREN: サーバーが生成できる子プロセスの最大数。デフォルトでは、Sendmail は子プロセスの数に制限を割り当てません。制限が設定され、到達すると、追加の接続が遅延します。

- confMIN_FREE_BLOCKS: サーバーがメールを受け入れるために使用できる空きブロックの最小数。デフォルトは 100 ブロックです。

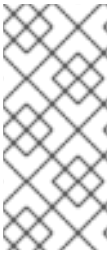
- confMAX_HEADERS_LENGTH: メッセージヘッダーの最大許容サイズ（バイト単位）。

- confMAX_MESSAGE_SIZE: 1 つのメッセージの最大許容サイズ（バイト単位）。

48.2.7.2. NFS および Sendmail

メールスプールディレクトリー /var/spool/mail/ を NFS 共有ボリュームに置かないでください。

NFSv2 および NFSv3 では、ユーザー ID およびグループ ID の管理を行わないため、2 人以上のユーザーが同じ UID を持ち、互いのメールを受信および読み取ることができます。



注記

Kerberos を使用する NFSv4 では、SECRPC_GSS カーネルモジュールは UID ベースの認証を利用しないため、これは当てはまりません。ただし、メールスプールディレクトリーを NFS 共有ボリュームに配置し ない ことが推奨されます。

48.2.7.3. メール専用ユーザー

Sendmail サーバーでローカルユーザーによる不正使用を防ぐためには、メールユーザーが電子メールプログラムを使用して Sendmail サーバーにのみアクセスすることをお勧めします。メールサーバー上のシェルアカウントは許可されるべきではなく、/etc/passwd ファイル内のすべてのユーザーシェルは /sbin/nologin に設定されている必要があります (root ユーザーを除く可能性があります)。

48.2.8. リッスンしているポートの確認

ネットワークサービスを設定したら、システムのネットワークインターフェイスで実際にリッスンしているポートに注意することが重要です。開いているポートはすべて、侵入によるものです。

ネットワーク上でリッスンしているポートを一覧表示する基本的な方法は 2 つあります。信頼性が低いアプローチは、netstat -an や lsof -i などのコマンドを使用してネットワークスタックをクエリーすることです。この方法は、これらのプログラムはネットワークからマシンに接続されないため、信頼性は低くなりますが、システムで実行されているものを確認してください。このため、これらのアプリケーションは攻撃者が交換するためのターゲットを頻繁に使用します。クラッカーは、netstat と lsof を独自の修正バージョンに置き換えて、承認されていないネットワークポートをオープンした場合に、その追跡をカバーしようとします。

ネットワーク上でリッスンしているポートを確認するためのより信頼性の高い方法は、nmap などのポートスキャナーを使用することです。

以下のコマンドは、ネットワークから TCP 接続をリッスンするポートを決定します。

```
nmap -sT -O localhost
```

このコマンドの出力は、以下のようになります。

```
Starting nmap 3.55 ( http://www.insecure.org/nmap/ ) at 2004-09-24 13:49 EDT
Interesting ports on localhost.localdomain (127.0.0.1):
(The 1653 ports scanned but not shown below are in state: closed)
PORT      STATE SERVICE
22/tcp    open  ssh
25/tcp    open  smtp
111/tcp   open  rpcbind
113/tcp   open  auth
631/tcp   open  ipp
834/tcp   open  unknown
2601/tcp  open  zebra
32774/tcp open  sometimes-rpc11
Device type: general purpose
Running: Linux 2.4.X|2.5.X|2.6.X OS details: Linux 2.5.25 - 2.6.3 or Gentoo 1.2 Linux 2.4.19 rc1-rc7)
Uptime 12.857 days (since Sat Sep 11 17:16:20 2004)
Nmap run completed -- 1 IP address (1 host up) scanned in 5.190 seconds
```

この出力は、sunrpc サービスが存在するため、システムが portmap を実行していることを示しています。ただし、ポート 834 には mystery サービスもあります。ポートが既知のサービスの公式リストに関連付けられているかどうかを確認するには、以下を入力します。

```
cat /etc/services | grep 834
```

このコマンドは出力を返しません。これは、ポートが予約済み範囲(0 から 1023)にあり、root アクセスが開いているのに対し、既知のサービスに関連付けられていないことを示しています。

次に、netstat または lsof を使用してポートに関する情報を確認します。netstat を使用してポート 834 を確認するには、次のコマンドを使用します。

```
netstat -anp | grep 834
```

このコマンドは、以下の出力を返します。

```
tcp   0    0 0.0.0.0:834    0.0.0.0:*   LISTEN   653/ypbind
```

netstat に開いているポートが存在するのは、クラッカーがハッキングシステムで誤ってポートを開くことは、このコマンドで認識できない可能性が高いためです。また、[p] オプションは、ポートを開いたサービスのプロセス ID (PID)を表示します。この場合、オープンポートは ypbind (NIS)に属します。これは、portmap サービスとともに処理される RPC サービスです。

lsof コマンドは、開いているポートをサービスにリンクできるため、netstat と同様の情報を表示します。

```
lsof -i | grep 834
```

このコマンドからの出力の関連部分は以下のとおりです。

```
ypbind    653     0   7u  IPv4    1319          TCP *:834 (LISTEN)
ypbind    655     0   7u  IPv4    1319          TCP *:834 (LISTEN)
ypbind    656     0   7u  IPv4    1319          TCP *:834 (LISTEN)
ypbind    657     0   7u  IPv4    1319          TCP *:834 (LISTEN)
```

これらのツールは、マシンで実行しているサービスのステータスについて非常に多くのことを示しています。これらのツールは柔軟で、ネットワークサービスおよび設定に関する情報を提供します。詳細は、lsof、netstat、nmap、および services の man ページを参照してください。

48.3. シングルサインオン(SSO)

48.3.1. はじめに

Red Hat Enterprise Linux SSO 機能により、Red Hat Enterprise Linux デスクトップユーザーがパスワードを入力する必要がある回数が減ります。複数の主要なアプリケーションは、同じ基礎となる認証および承認メカニズムを利用して、ユーザーがログイン画面から Red Hat Enterprise Linux にログインし、パスワードを再入力する必要はありません。これらのアプリケーションの詳細を以下に示します。

さらに、ユーザーはネットワークがない場合に(オフラインモード)、またはワイヤレスアクセスなどのネットワーク接続が信頼できない場合でも、マシンにログインできます。後者の場合、サービスは正常に低下します。

48.3.1.1. サポート対象のアプリケーション

現在、以下のアプリケーションは、Red Hat Enterprise Linux の統合ログインスキームでサポートされています。

- Login
- screensaver
- Firefox および Thunderbird

48.3.1.2. サポートされる認証メカニズム

現在、Red Hat Enterprise Linux は以下の認証メカニズムをサポートしています。

- Kerberos 名/パスワードログイン
- スマートカード/PIN ログイン

48.3.1.3. 対応するスマートカード

Red Hat Enterprise Linux は Cyberflex e-gate カードとリーダーでテストされていますが、PCSC-lite でサポートされるリーダーであれば、Java カード 2.1.1 と Global Platform 2.0.1 仕様の両方に準拠するカードが正しく動作するはずです。

Red Hat Enterprise Linux は、Common Access Cards (CAC)でもテストされています。CAC でサポートされているリーダーは SCM SCR 331 USB Reader です。

Red Hat Enterprise Linux 5.2 の時点で、Gemalto スマートカード(PKCSI v2.1 で DER SHA1 値が設定された DER SHA1 値を持つ標準) (Cyberflex Access 64k v2)がサポートされるようになりまし

た。これらのスマートカードは、Chip/Smart Card Interface Devices (CCID)に準拠するリーダーを使用するようになりました。

48.3.1.4. Red Hat Enterprise Linux Single Sign-on の利点

多数のプロトコルやクレデンシャルストアを利用するセキュリティーメカニズムが多数存在します。たとえば、SSL、SSH、IPsec、および Kerberos などがあります。Red Hat Enterprise Linux SSO は、上記の要件をサポートするためにこれらのスキームを統一することを目的としています。これは、Kerberos を X.509v3 証明書に置き換えることを意味するのではなく、システムユーザーとそれらを管理する管理者の両方に負荷を軽減するようにユニットします。

この目的を達成するために、Red Hat Enterprise Linux は以下のようになります。

- 各オペレーティングシステム上の NSS 暗号ライブラリーの共有インスタンス 1 つを提供します。
- Certificate System の Enterprise Security Client (ESC)をベースオペレーティングシステムで出荷します。ESC アプリケーションは、スマートカード挿入イベントを監視します。ユーザーが Red Hat Enterprise Linux Certificate System サーバー製品で使用するために設計されたスマートカードを挿入したことを検知すると、そのスマートカードの登録方法をユーザーに指示するユーザーインターフェイスが表示されます。
- スマートカードを使用してオペレーティングシステムにログインするユーザーが Kerberos 認証情報（ファイルサーバーなどにログインできる）も取得できるように Kerberos と NSS を統合する。

48.3.2. 新しいスマートカードの使用

スマートカードを使用してシステムにログインし、この技術が提供するセキュリティーオプションを増やす前に、基本的なインストールと設定手順を実行する必要があります。これについては、以下で説明します。

注記

本セクションでは、スマートカードを開始するための概要ビューを提供します。詳細は、Red Hat Certificate System Enterprise Security Client Guide を参照してください。

1. *Kerberos 名とパスワードでログインします。*

2. *nss-tools パッケージが読み込まれていることを確認します。*

3. *企業固有のルート証明書をダウンロードしてインストールします。以下のコマンドを使用して、ルート CA 証明書をインストールします。*

```
certutil -A -d /etc/pki/nssdb -n "root ca cert" -t "CT,C,C" \
 -i ./ca_cert_in_base64_format.crt
```

4. *次の RPM がシステムにインストールされていることを確認します。esc、pam_pkcs11、coolkey、ifd-egate、ccid、gdm、authconfig、authconfig-gtk。*

5. *スマートカードログインサポートの有効化*

   a. *Gnome Title Bar で、System->Administration->Authentication を選択します。*

   b. *必要に応じて、マシンの root パスワードを入力します。*

   c. *Authentication Configuration ダイアログで、Authentication タブをクリックします。*

   d. *Enable Smart Card Support チェックボックスを選択します。*

   e. *Configure Smart Card... ボタンをクリックして、Smartcard Settings ダイアログを表示し、必要な設定を指定します。*

      - *ログインにはスマートカードが 必要 - このチェックボックスをオフにします。スマートカードで正常にログインしたら、このオプションを選択して、ユーザーがスマートカードなしでログインできないようにします。*

      -

Card Removal Action - ログイン後にスマートカードを削除するときに何が起こるかを制御します。利用可能なオプションは以下のとおりです。

- *lock - スマートカードを削除すると、X 画面がロックされます。*
- *ignore - スマートカードを削除しても効果はありません。*

6. *OCSP(Online Certificate Status Protocol)を有効にする必要がある場合は、/etc/pam_pkcs11/pam_pkcs11.conf ファイルを開き、以下の行を見つけます。*

   enable_ocsp = false;

   以下のようにこの値を true に変更します。

   enable_ocsp = true;

7. *スマートカードの登録*

8. *CAC カードを使用している場合は、以下の手順も実行する必要があります。*

   a. *root アカウントに切り替え、/etc/pam_pkcs11/cn_map という名前のファイルを作成します。*

   b. *以下のエントリーを cn_map ファイルに追加します。*

MY.CAC_CN.123454 -> myloginid

ここで、MY.CAC_CN.123454 は CAC のコモンネームで、myloginid は UNIX ログイン ID です。

9. *ログアウト*

48.3.2.1. トラブルシューティング

スマートカードが機能しなくなった場合は、以下のコマンドを使用して問題の原因を見つけてください。

```
pklogin_finder debug
```

登録したスマートカードがプラグインされている間に pklogin_finder ツールをデバッグモードで実行すると、証明書の有効性に関する情報の出力を試みます。また、正常にカードにある証明書からログイン ID をマップしようとすると、これからログイン ID をマップしようとします。

48.3.3. スマートカードの登録の仕組み

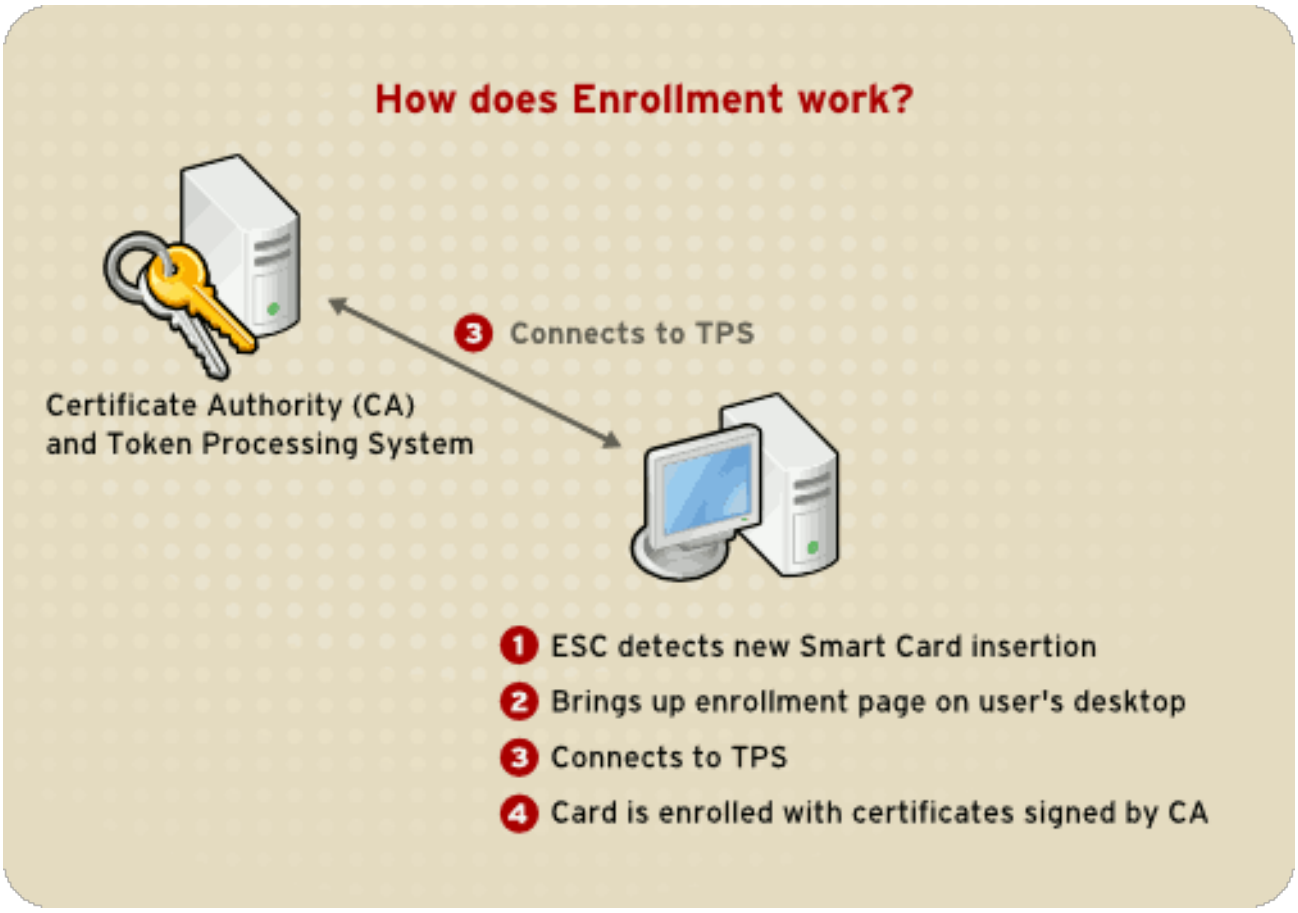
スマートカードは、有効な認証局(CA)が署名した適切な証明書を受け取ったときに 登録 されると見なされます。これには、以下で説明するいくつかの手順が含まれます。

1. *ユーザーは、スマートカードをワークステーションのスマートカードリーダーに挿入します。このイベントは、Enterprise Security Client (ESC)によって認識されます。*

2. *登録ページがユーザーのデスクトップに表示されます。ユーザーは必要な情報を完了し、ユーザーのシステムは Token Processing System (TPS)および CA に接続します。*

3. *TPS は、CA によって署名された証明書を使用してスマートカードを登録します。*

図48.4 スマートカードの登録の仕組み



[D]

48.3.4. スマートカードログインの仕組み

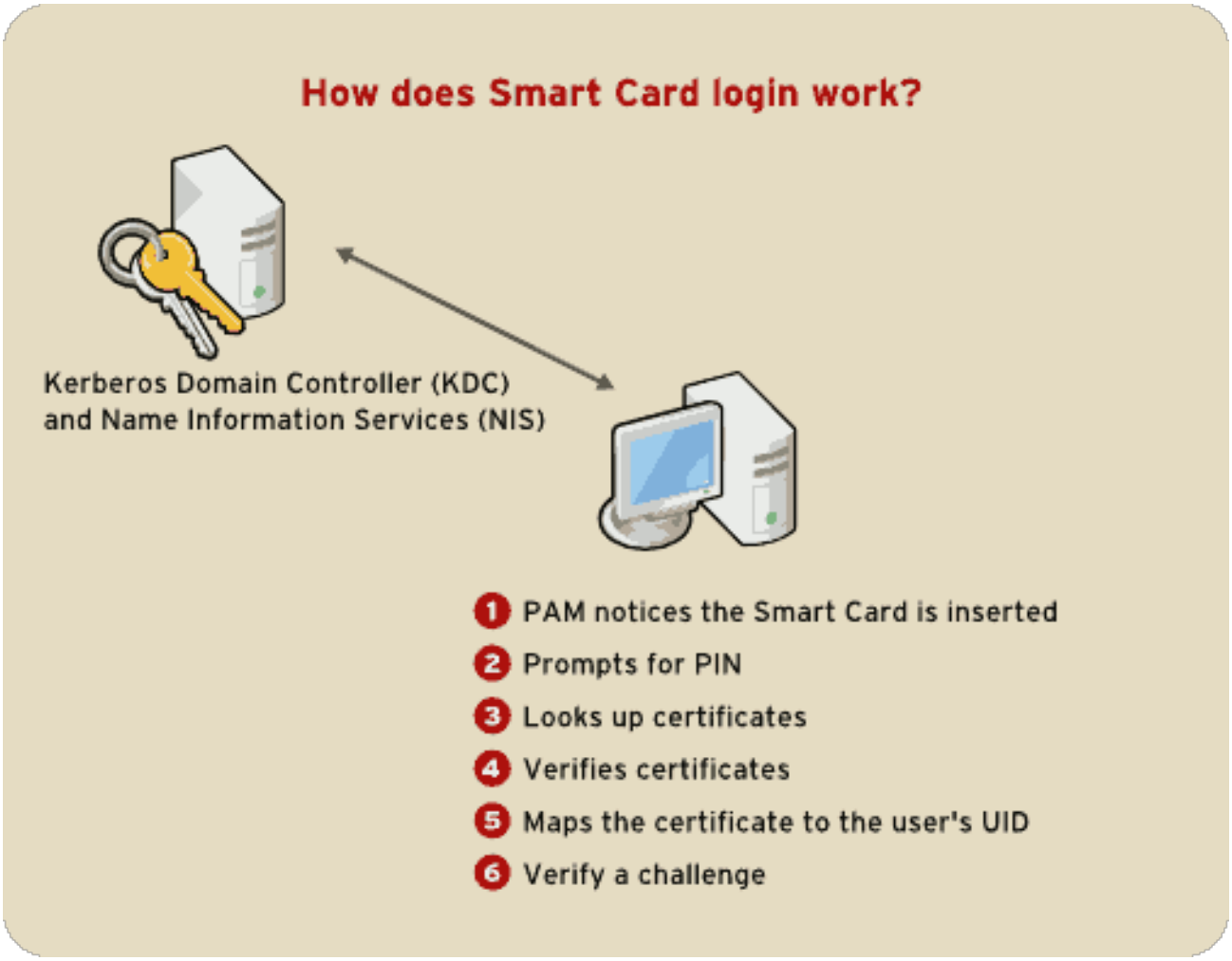
本セクションでは、スマートカードを使用してログインするプロセスの概要を説明します。

1. *ユーザーがスマートカードリーダーにスマートカードを挿入すると、このイベントは PAM 機能によって認識され、ユーザーの PIN の入力が求められます。*

2. *次に、システムはユーザーの現在の証明書を検索し、その有効性を検証します。証明書はユーザーの UID にマッピングされます。*

3. *これは KDC に対して検証され、ログインに付与されます。*

図48.5 スマートカードログインの仕組み



[D]

注記

フォーマットされている場合でも、登録されていないカードでログインすることはできません。新しいカードを登録する前に、フォーマットされた、登録されたカードでログインするか、またはスマートカードを使用してログインする必要があります。

Kerberos および PAM の詳細は、「Kerberos」 および 「PAM (プラグ可能な認証モジュール)」を参照してください。

48.3.5. Firefox で SSO に Kerberos を使用する設定

シングルサインオンに Kerberos を使用するように Firefox を設定できます。この機能が正しく機能するには、Kerberos 認証情報を適切な KDC に送信するように Web ブラウザーを設定する必要があります。以下のセクションでは、設定の変更およびその他の要件について説明します。

1.

Firefox のアドレスバーに、about:config と入力して現在の設定オプションの一覧を表示します。

2. *Filter フィールドに、オプションのリストを制限するために negotiate と入力します。*

3. **network.negotiate-auth.trusted-uris** *エントリーをダブルクリックして、***Enter string value** *ダイアログボックスを表示します。*

4. *認証に使用するドメイン名(.example.com など)を入力します。*

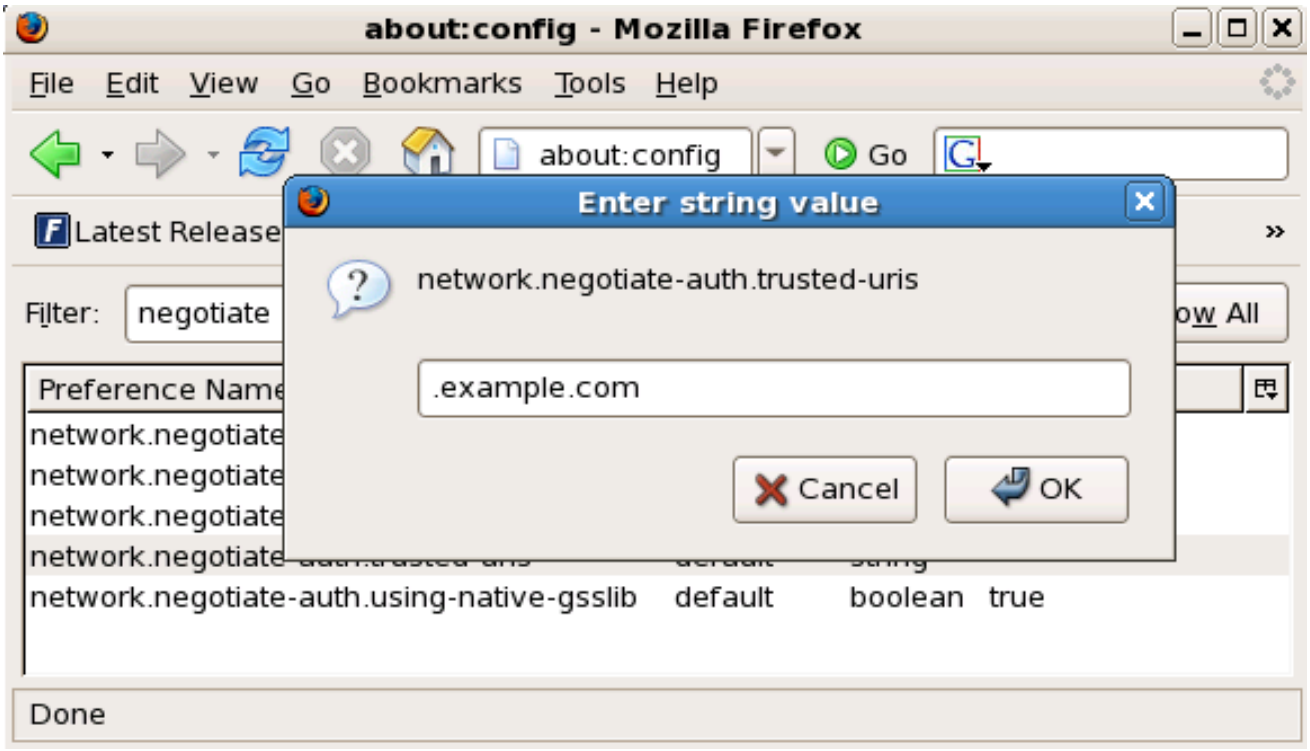
5. *同じドメインを使用して、***network.negotiate-auth.delegation-uris** *エントリーに対して上記の手順を繰り返します。*

注記

この値は空白のままにすることができます。これは、Kerberos チケットのパススルーを可能にするためですが、必須ではありません。

これら 2 つの設定オプションが一覧にない場合は、お使いの Firefox のバージョンが Negotiate 認証をサポートするのに古くなる可能性があるため、アップグレードを検討してください。

図48.6 Kerberos での SSO 用の Firefox の設定



[D]

次に、Kerberos チケットがあることを確認する必要があります。コマンドシェルで kinit と入力して Kerberos チケットを取得します。利用可能なチケットの一覧を表示するには、klist と入力します。以下は、これらのコマンドの出力例を示しています。

```
~]$ kinit
Password for user@EXAMPLE.COM:

~]$ klist
Ticket cache: FILE:/tmp/krb5cc_10920
Default principal: user@EXAMPLE.COM

Valid starting     Expires            Service principal
10/26/06 23:47:54  10/27/06 09:47:54  krbtgt/USER.COM@USER.COM
        renew until 10/26/06 23:47:54

Kerberos 4 ticket cache: /tmp/tkt10920
klist: You have no tickets cached
```

48.3.5.1. トラブルシューティング

上記の設定手順を実行し、Negotiate 認証が機能しない場合は、認証プロセスの詳細ロギングを有効にすることができます。これは、問題の原因を見つけるのに役立ちます。詳細なロギングを有効にするには、以下の手順に従います。

1. Firefox のすべてのインスタンスを閉じます。

2. コマンドシェルを開き、以下のコマンドを入力します。

```
export NSPR_LOG_MODULES=negotiateauth:5
export NSPR_LOG_FILE=/tmp/moz.log
```

3. そのシェルから Firefox を再起動し、以前に認証できなかった Web サイトにアクセスします。情報は /tmp/moz.log に記録され、問題が解消される可能性があります。以下に例を示します。

```
-1208550944[90039d0]: entering nsNegotiateAuth::GetNextToken()
-1208550944[90039d0]: gss_init_sec_context() failed: Miscellaneous failure
No credentials cache found
```

これは、Kerberos チケットがなく、kinit を実行する必要があることを示しています。

kinit をマシンから正常に実行できるが認証できない場合は、以下のようなメッセージがログファイルに表示される場合があります。

```
-1208994096[8d683d8]: entering nsAuthGSSAPI::GetNextToken()
-1208994096[8d683d8]: gss_init_sec_context() failed: Miscellaneous failure
Server not found in Kerberos database
```

これは通常、Kerberos 設定の問題を示しています。/etc/krb5.conf ファイルの [domain_realm] セクションに正しいエントリーがあることを確認します。以下に例を示します。

```
.example.com = EXAMPLE.COM
example.com = EXAMPLE.COM
```

ログに何も表示されない場合は、プロキシーの背後にある可能性があり、そのプロキシーが Negotiate 認証に必要な HTTP ヘッダーを削除している可能性があります。回避策として、代わりに HTTPS を使用してサーバーへの接続を試みることができます。これにより、要求を変更せずに渡すことができます。次に、上記のように ログファイルを使用したデバッグに進みます。

48.4. PAM (プラグ可能な認証モジュール)

システムにユーザーアクセスを付与するプログラムは、認証 を使用して相互のアイデンティティーを確認します (つまり、ユーザーがそのユーザーであると確立するためのプログラム)。

これまでは、各プログラムはユーザー認証の方法を使用していました。Red Hat Enterprise Linux では、多くのプログラムが プラグ可能な認証モジュール (PAM) と呼ばれる集中型認証メカニズムを使用するように設定されています。

PAM はプラグ可能なモジュラーアーキテクチャーを使用しており、システム管理者はシステム用の認証ポリシーを柔軟に設定することができます。

ほとんどの場合、PAM 対応アプリケーションのデフォルトの PAM 設定ファイルで十分です。ただし、場合によっては PAM 設定ファイルを編集する必要がある場合があります。PAM の設定が間違っているとシステムセキュリティーが危険にさらされるため、変更を行う前にこれらのファイルの構造を理解することが重要です。詳細は、「PAM 設定ファイル形式」 を参照してください。

48.4.1. PAM の利点

PAM には、以下の利点があります。

- さまざまなアプリケーションで使用できる一般的な認証スキーム。
- システム管理者とアプリケーション開発者の両方に対する優れた柔軟性と制御性。
- 単一の完全に文書化されたライブラリー。開発者は独自の認証スキームを作成せずにプログラムを作成できます。

48.4.2. PAM 設定ファイル

/etc/pam.d/ ディレクトリーには、PAM 対応アプリケーションごとに PAM 設定ファイルが含まれます。以前のバージョンの PAM では、/etc/pam.conf ファイルが使用されていましたが、このファイルは非推奨となり、/etc/pam.d/ ディレクトリーが存在しない場合にのみ使用されます。

48.4.2.1. PAM サービスファイル

PAM 対応の アプリケーションまたは サービス には、/etc/pam.d/ ディレクトリーにファイルがあります。このディレクトリーの各ファイルは、アクセスを制御するサービスと同じ名前を持ちます。

PAM 対応プログラムは、サービス名を定義し、独自の PAM 設定ファイルを /etc/pam.d/ ディレク

トリーにインストールします。たとえば、login プログラムは、login としてサービス名を定義し、/etc/pam.d/login の PAM 設定ファイルをインストールします。

48.4.3. PAM 設定ファイル形式

各 PAM 設定ファイルには、以下のようにフォーマットされたディレクティブのグループが含まれています。

```
<module interface>  <control flag>   <module name>   <module arguments>
```

これらの各要素については、以下のセクションで説明します。

48.4.3.1. モジュールインターフェイス

現在、4 種類の PAM モジュールインターフェイスが利用できます。それぞれは、承認プロセスのさまざまな要素に対応します。

- auth: このモジュールインターフェイスは使用を認証します。たとえば、パスワードの有効性を要求し、検証します。このインターフェイスのあるモジュールは、グループメンバーシップや Kerberos チケットなどの認証情報を設定することもできます。

- account: このモジュールインターフェイスは、アクセスが許可されることを確認します。たとえば、ユーザーアカウントの有効期限が切れたか、または特定の時刻にユーザーがログインできるかどうかを確認できます。

- password: このモジュールインターフェイスは、ユーザーパスワードの変更に使用されます。

- session: このモジュールインターフェイスは、ユーザーセッションを設定および管理します。このインターフェイスのあるモジュールは、ユーザーのホームディレクトリーをマウントしたり、ユーザーのメールボックスを利用可能にするなど、アクセスを許可するために必要な追加のタスクも実行できます。

注記

個別のモジュールは、いずれかのインターフェイスまたはすべてのモジュールインターフェイスを提供できます。たとえば、pam_unix.so は 4 つのモジュールインターフェイスをすべて提供します。

PAM 設定ファイルでは、モジュールインターフェイスは以下のように最初のフィールドで定義されます。たとえば、設定の典型的な行は以下のようになります。

```
auth required pam_unix.so
```

これにより、PAM が pam_unix.so モジュールの認証インターフェイスを使用するように指示します。

48.4.3.1.1. モジュールインターフェイスのスタッキング

モジュールインターフェイスのディレクティブは、重ねて配置することでスタック化 が可能なので、複数のモジュールをまとめて 1 つの目的に使用することができます。モジュールの制御フラグが sufficient または required の値を使用する場合（これらのフラグの詳細については 「制御フラグ」 を参照）、モジュールを一覧表示する順番は認証プロセスにとって重要です。

スタック化により、管理者はユーザーが認証を行う前に、特定の条件を必須とすることが容易になります。たとえば、reboot コマンドは通常、PAM 設定ファイルにあるように、いくつかのスタックされたモジュールを使用します。

```
~]# cat /etc/pam.d/reboot
#%PAM-1.0
auth sufficient pam_rootok.so
auth required pam_console.so
#auth include system-auth
account required pam_permit.so
```

- 最初の行はコメントで、処理されません。
- auth sufficient pam_rootok.so - この行は、UID が 0 であることを確認して、pam_rootok.so モジュールを使用して、現在のユーザーが root かどうかを確認します。このテストに成功すると、他のモジュールは参照されず、コマンドが実行されます。このテストが失敗すると、次のモジュールが参照されます。
- auth required pam_console.so - この行は、pam_console.so モジュールを使用してユーザーを認証しようとします。このユーザーがすでにコンソールにログインしている場合、pam_console.so はサービス名(reboot)と同じ名前を持つ /etc/security/console.apps/ ディレクトリーにファイルが存在するかどうかを確認します。そのようなファイルが存在する場合は、認証が成功し、制御が次のモジュールに渡されます。

- *#auth include system-auth - この行はコメント化され、処理されません。*

- *account required pam_permit.so - この行は、pam_permit.so モジュールを使用して、root ユーザーまたはコンソールにログインしているすべてのユーザーがシステムを再起動します。*

48.4.3.2. 制御フラグ

すべての PAM モジュールは、呼び出されると成功または失敗の結果を生成します。コントロールフラグは結果に基づいて PAM に指示します。モジュールは特定の順序でスタックでき、制御フラグは特定モジュールの成功または失敗の重要性を判断し、ユーザーをサービスに対して認証する際の全体的な目標を決定します。

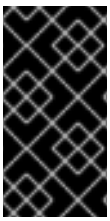
事前定義された制御フラグは 4 つあります。

- *required: 認証を継続するには、モジュール結果が成功する必要があります。この時点でテストが失敗すると、そのインターフェイスを参照するすべてのモジュールテストの結果が完了するまでユーザーには通知されません。*

- *requisite: 認証を継続するには、モジュール結果が成功する必要があります。ただし、この時点でテストが失敗すると、最初に失敗した required または requisite モジュールテストを反映したメッセージとともに、すぐにユーザーに通知されます。*

- *sufficient: モジュール結果は、失敗すると無視されます。ただし、モジュールフラグ付きの sufficient の結果が成功で、かつ、以前のモジュールフラグ付きの required が失敗していない場合、その他の結果は不要で、ユーザーはサービスに認証されます。*

- *optional: モジュール結果は無視されます。その他のモジュールがインターフェイスを参照しない場合に認証成功に必要となるのは、optional としてフラグが付いたモジュールです。*



重要な影響

必要 なモジュールが呼び出される順序は重要ではありません。十分 な制御フラグと必須 制御フラグのみにより、順序が重要になります。

より正確な制御を可能にする新しい制御フラグ構文が PAM で利用可能になりました。

/usr/share/doc/pam- <version-number> / ディレクトリーにある pam.d の man ページと PAM ドキュメントです。ここで、< version-number > はシステム上の PAM のバージョン番号で、この新しい構文の詳細を説明します。

48.4.3.3. モジュール名

モジュール名は、指定されたモジュールインターフェイスを含むプラグ可能なモジュールの名前で PAM を提供します。古いバージョンの Red Hat Enterprise Linux では、モジュールへの完全パスは PAM 設定ファイルで提供されていました。ただし、64 ビットの PAM モジュールを /lib64/ security/ ディレクトリーに保存する multilib システムについては、アプリケーションが適切なバージョンの lib pam にリンクされているため、ディレクトリー名は省略されます。これにより、モジュールの正しいバージョンを見つけることができます。

48.4.3.4. モジュール引数

PAM はいくつかのモジュール向けの認証中に 引数 を使って情報をプラグ可能なモジュールに渡します。

たとえば、pam_userdb.so モジュールは Berkeley DB ファイルに保存されている情報を使用してユーザーを認証します。Berkeley DB は、多くのアプリケーションに埋め込まれたオープンソースデータベースシステムです。モジュールは db 引数を取り、リクエストされたサービスに使用するデータベースを認識できるようにします。

以下は、PAM 設定の典型的な pam_userdb.so 行です。& lt;path-to-file> は、Berkeley DB データベースファイルへの完全パスです。

> ***auth required pam_userdb.so db=***<path-to-file>

無効な引数は 通常 無視され、PAM モジュールの成功や失敗には影響しません。ただし、一部のモジュールは無効な引数で失敗する可能性があります。ほとんどのモジュールはエラーを /var/log/secure ファイルに報告します。

48.4.4. PAM 設定ファイルのサンプルについて

以下は、PAM アプリケーション設定ファイルのサンプルです。

```
#%PAM-1.0
auth required  pam_securetty.so
auth required  pam_unix.so nullok
auth required  pam_nologin.so
account required  pam_unix.so
```

```
password required  pam_cracklib.so retry=3
password required  pam_unix.so shadow nullok use_authtok
session required  pam_unix.so
```

- 最初の行は、行頭のハッシュ記号 (#) が示すように、コメントになります。

- 2 行目から 4 行目は、ログイン認証用に 3 つのモジュールをスタックしています。

  auth required pam_securetty.so - このモジュールは、ユーザーが root としてログインしようとすると、そのファイルが存在する 場合 は、ユーザーがログインする tty が /etc/securetty ファイルに一覧表示されている ことを確認します。

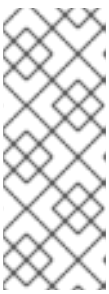
  tty がファイルに記載されていない場合は、root でログインしようとすると Login incorrect メッセージで失敗します。

  auth required pam_unix.so nullok — このモジュールはユーザーにパスワードを要求し、/etc/passwd に保存された情報を使用してパスワードをチェックしします。存在する場合は /etc/shadow。

  認証フェーズでは、pam_unix.so モジュールは、ユーザーのパスワードが passwd ファイルか シャドウ ファイルにあるかを自動的に検出します。詳細は、「シャドウパスワード」を参照してください。

  - 引数は pam_unix.so モジュールに対し、空のパスワードを許可するように nullok に指示します。

- auth required pam_nologin.so — これは、認証の最終ステップです。これは、/etc/nologin ファイルが存在するかどうかを確認します。ユーザーが存在して root でない場合は、認証に失敗します。

  注記

  この例では、最初の auth モジュールが失敗しても、3 つの auth モジュールがすべてチェックされます。これにより、ユーザーは認証に失敗したステージを把握できません。攻撃者のこのような知識により、システムのクラッキング方法がより簡単に推測される可能性があります。

-

account required pam_unix.so — このモジュールは、必要なアカウントの検証を実行します。たとえば、シャドウパスワードが有効になっていると、pam_unix.so モジュールのアカウントインターフェイスが、アカウントの有効期限が切れたかどうか、または許可された猶予期間内にユーザーがパスワードを変更していないかどうかを確認します。

- *password required pam_NORMAL.so retry=3 - パスワードの有効期限が切れると、pam_NORMAL.so モジュールのパスワードコンポーネントは新しいパスワードを要求します。その後、新たに作成されたパスワードをテストして、辞書ベースのパスワードクラッキングプログラムで簡単に判別できるかどうかを確認します。*
  - *引数 retry=3 は、テストに 1 回失敗しても、ユーザーは強固なパスワードを作成する機会があと 2 回あることを示しています。*
- *password required pam_unix.so shadow nullok use_authtok - この行は、プログラムがユーザーのパスワードを変更した場合、pam_unix.so モジュールの パスワード インターフェイスを使用してそれを行う必要があることを指定します。*
  - *引数 shadow は、ユーザーのパスワード更新の際にシャドウパスワードを作成するようモジュールに指示します。*
  - *この引数 nullok は、ユーザーが空のパスワードから パスワードを変更できるようにするようにモジュールに指示します。それ以外の場合は、null パスワードはアカウントロックとして扱われます。*
  - *この行の最後の引数 use_authtok は、PAM モジュールをスタックする際に順序の重要性を示す優れた引数を提供します。この引数は、ユーザーに新しいパスワードを要求しないようにモジュールに指示します。代わりに、以前のパスワードモジュールで記録されたパスワードを受け入れます。これにより、新しいパスワードはすべて、受け入れられる前に安全なパスワードの pam_NORMAL.so テストを渡す必要があります。*
- *session required pam_unix.so — 最後の行は、pam_unix.so モジュールのセッションインターフェイスにセッションを管理するよう指示します。このモジュールは、各セッションの開始と最後で、ユーザー名とサービスタイプを /var/log/secure に記録します。このモジュールは、追加機能のために他のセッションモジュールとスタックすることで補足できます。*

48.4.5. PAM モジュールの作成

PAM 対応アプリケーションで使用するために、いつでも新しい PAM モジュールを作成または追加できます。

たとえば、開発者はワンタイムパスワードの作成方法を作成し、それをサポートする PAM モジュールを作成することができます。PAM 対応プログラムは、再コンパイルや変更を行わずに、新しいモジュールおよびパスワードメソッドをすぐに使用できます。

これにより、開発者およびシステム管理者は、再コンパイルせずに異なるプログラムの認証方法をテストするだけでなく、組み合わせることができるようになります。

モジュール作成に関するドキュメントは、/usr/share/doc/pam- <version-number> / ディレクトリーに含まれています。< version-number > は、システムの PAM のバージョン番号になります。

48.4.6. PAM と管理認証情報のキャッシング

Red Hat Enterprise Linux の多くのグラフィカル管理ツールは、pam_timestamp.so モジュールを使用して最大 5 分間、ユーザーに昇格した特権を提供します。このメカニズムの仕組みを理解することが重要です。これは、pam_timestamp.so が有効なときににターミナルから出るユーザーが、コンソールに物理的にアクセスできるユーザーすべてがマシンを変更できる状態のままにするためです。

PAM タイムスタンプスキームでは、グラフィカル管理アプリケーションにより、起動時に root パスワードの入力が求められます。ユーザーが認証されたとき、pam_timestamp.so モジュールはタイムスタンプファイルを作成します。デフォルトでは、これは /var/run/sudo/ ディレクトリーに作成されます。タイムスタンプファイルがすでに存在する場合は、グラフィカル管理プログラムではパスワードの入力が求められません。代わりに、pam_timestamp.so モジュールはタイムスタンプファイルを最新の状態にし、ユーザーの不完全な管理アクセスを 5 分追加で保持します。

/var/run/sudo/<user> ファイルを確認して、タイムスタンプファイルの実際の状態を確認 できます。デスクトップでは、関連するファイルは unknown:root です。これが存在し、タイムスタンプが 5 分未満の場合は、認証情報が有効です。

タイムスタンプファイルが存在すると、パネルの通知スペースに認証アイコンが表示されます。

図48.7 認証アイコン

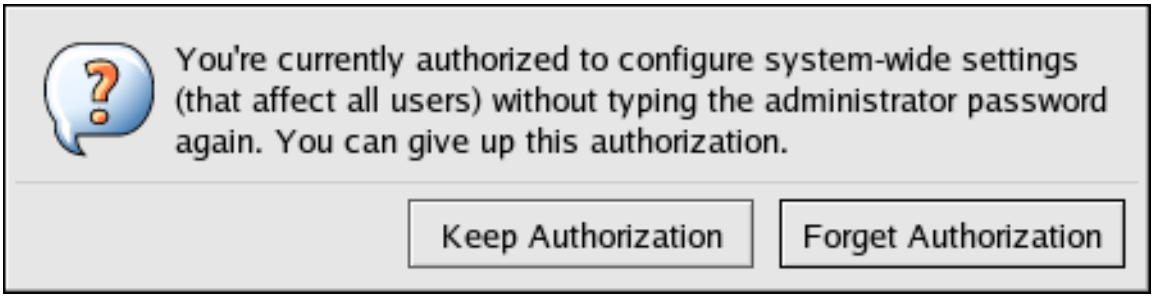
[D]

48.4.6.1. タイムスタンプファイルの削除

PAM タイムスタンプがアクティブなコンソールを利用する前に、タイムスタンプファイルを破棄することが推奨されます。グラフィカル環境でこれを行うには、パネルの認証アイコンをクリックしま

す。これにより、ダイアログボックスが開きます。Forget Authorization ボタンをクリックして、アクティブなタイムスタンプファイルを破棄します。

図48.8 認証ダイアログを閉じる



[D]

PAM タイムスタンプファイルに関して、以下に注意してください。

- *ssh を使用してシステムにリモートでログインしている場合は、/sbin/pam_timestamp_check -k root コマンドを使用してタイムスタンプファイルを破棄します。*

- *特権アプリケーションを起動したのと同じターミナルウィンドウから /sbin/pam_timestamp_check -k root コマンドを実行する必要があります。*

- */sbin/pam_timestamp_check -k コマンドを使用するには、pam_timestamp.so モジュールを起動したユーザーとしてログインしている必要があります。このコマンドを使用するには、root でログインしないでください。*

- *デスクトップで認証情報を強制終了する場合は（アイコンの Forget Authorization アクションを使用せずに）、以下のコマンドを使用します。*

```
pam_timestamp_check -k root </dev/null >/dev/null 2>/dev/null
```

このコマンドを使用しないと、コマンドを実行する pty から認証情報（存在する場合）のみが削除されます。

pam_timestamp_check を使用したタイムスタンプファイルの破棄の詳細は、pam_timestamp_check の man ページを参照してください。

48.4.6.2. 一般的な pam_timestamp ディレクティブ

pam_timestamp.so モジュールは複数のディレクティブを受け入れます。最も一般的に使用される 2 つのオプションを以下に示します。

- timestamp_timeout - タイムスタンプファイルが有効な期間（秒単位）を指定します。デフォルト値は 300 (5 分) です。
- timestampdir - タイムスタンプファイルを保存するディレクトリーを指定します。デフォルト値は /var/run/sudo/ です。

pam_timestamp.so モジュールの制御に関する詳細は、「インストールされているドキュメント」を参照してください。

48.4.7. PAM とデバイスの所有者

Red Hat Enterprise Linux では、マシンの物理コンソールにログインする最初のユーザーは、特定のデバイスを操作して、通常は root ユーザー用に予約された特定のタスクを実行することができます。これは、pam_console.so と呼ばれる PAM モジュールによって制御されます。

48.4.7.1. デバイスの所有者

ユーザーが Red Hat Enterprise Linux システムにログインすると、pam_console.so モジュールはログ イン またはグラフィカルログインプログラムである gdm、kdm、および xdm によって呼び出されます。このユーザーが、コンソールユーザー と呼ばれる物理コンソールにログインする最初のユーザー である場合、モジュールは通常 root が所有するさまざまなデバイスのユーザー所有権を付与します。コンソールユーザーは、そのユーザーの最後のローカルセッションが終了するまでこれらのデバイスを所有します。このユーザーがログアウトした後、デバイスの所有権は root ユーザーに戻ります。

影響を受けるデバイスには、サウンドカード、ディスケットドライブ、および CD-ROM ドライブが含まれますが、これらに限定されません。

この機能により、ローカルユーザーは root アクセスを取得しなくてもこれらのデバイスを操作できるため、コンソールユーザーの一般的なタスクを簡素化できます。

pam_console.so が制御するデバイスの一覧を変更するには、以下のファイルを編集します。

- */etc/security/console.perms*
- */etc/security/console.perms.d/50-default.perms*

上記のファイルに記載されているものとは異なるデバイスのパーミッションを変更したり、指定したデフォルトをオーバーライドしたりできます。50-default.perms ファイルを変更するのではなく、新しいファイル(xx-name.permsなど)を作成し、必要な変更を入力する必要があります。新しいデフォルトファイルの名前は、50 を超える数字で開始する必要があります（例： 51-default.perms）。これにより、50-default.perms ファイルでデフォルト値が上書きされます。

WARNING

リモートユーザーがランレベル 5 で実行するように gdm、kdm、または xdm ディスプレイマネージャー設定ファイルを変更 し た場合は、/etc/security/console.perms の < console > と {} ディレクティブを次の値に変更することを推奨します。

```
<console>=tty[0-9][0-9]* vc/[0-9][0-9]* :0\.[0-9] :0
<xconsole>=:0\.[0-9] :0
```

これにより、リモートユーザーがマシン上のデバイスおよび制限されたアプリケーションにアクセスできなくなります。

gdm、kdm、または xdm ディスプレイマネージャー設定ファイルが変更され、リモートユーザーが 5 以外の複数のユーザーランレベルで実行するように設定されていて、ディレクティブを完全に削除 {} して < console > ディレクティブを変更することが推奨されます。

```
<console>=tty[0-9][0-9]* vc/[0-9][0-9]*
```

48.4.7.2. アプリケーションアクセス

コンソールユーザーは、/etc/security/console.apps/ ディレクトリーで使用するために設定された

特定のプログラムにもアクセスできます。

このディレクトリーには、コンソールユーザーが /sbin および /usr/sbin で特定のアプリケーションを実行できるようにする設定ファイルが含まれています。

これらの設定ファイルの名前は、設定したアプリケーションと同じです。

コンソールユーザーがアクセスできるアプリケーションの重要なグループは、システムをシャットダウンまたは再起動する 3 つのプログラムです。

- /sbin/halt
- /sbin/reboot
- /sbin/poweroff

これは PAM 対応のアプリケーションであるため、pam_console.so モジュールを使用要件として呼び出します。

詳細は、「インストールされているドキュメント」 を参照してください。

48.4.8. 関連情報

以下のリソースでは、PAM の使用および設定方法について詳しく説明します。これらのリソースに加えて、システムの PAM 設定ファイルを読み取って、設定方法をよりよく理解します。

48.4.8.1. インストールされているドキュメント

- PAM 関連の man ページ：PAM に関連するさまざまなアプリケーションや設定ファイルには、いくつかの man ページが存在します。以下は、より重要な man ページの一部の一覧です。

  設定ファイル

- *PAM - PAM 設定ファイルの構造や目的など、PAM に関する理解情報。*

  この man ページでは、/etc/pam.conf と個々の設定ファイルの両方が、/etc/pam.d/ ディレクトリーにある説明にご留意ください。デフォルトでは、Red Hat Enterprise Linux は /etc/pam.d/ ディレクトリーの個々の設定ファイルを使用し、/etc/pam.conf が存在しても無視します。

- *pam_console - pam_console.so モジュールの目的を説明しています。また、PAM 設定ファイル内のエントリーに適した構文も説明します。*

- *console.apps: /etc/security/console.apps 設定ファイルで利用可能な形式およびオプションを説明します。これは、PAM が割り当てたコンソールユーザーがアクセス可能なアプリケーションを定義します。*

- *console.perms - /etc/security/console.perms 設定ファイルで利用可能な形式およびオプションを説明します。これは、PAM によって割り当てられるコンソールユーザーパーミッションを指定します。*

- *pam_timestamp: pam_timestamp.so モジュールを説明しています。*

- */usr/share/doc/pam- <version-number >:* 『System Administrators' Guide』、『Module Writers' Manual』、および 『Application Developers' Manual』、*および PAM 標準 DCE-RFC 86.0 のコピーが含まれます。< version-number > は PAM のバージョン番号です。*

- */usr/share/doc/pam- <version- <version-number> /txts/README.pam_timestamp - pam_timestamp.so PAM モジュールに関する情報が含まれます。< version-number > は PAM のバージョン番号になります。*

48.4.8.2. 便利な Web サイト

- *http://www.kernel.org/pub/linux/libs/pam/ - Linux-PAM プロジェクトの主要なディストリビューション Web サイト。さまざまな PAM モジュール、FAQ、および追加の PAM ドキュメントに関する情報が含まれます。*

注記

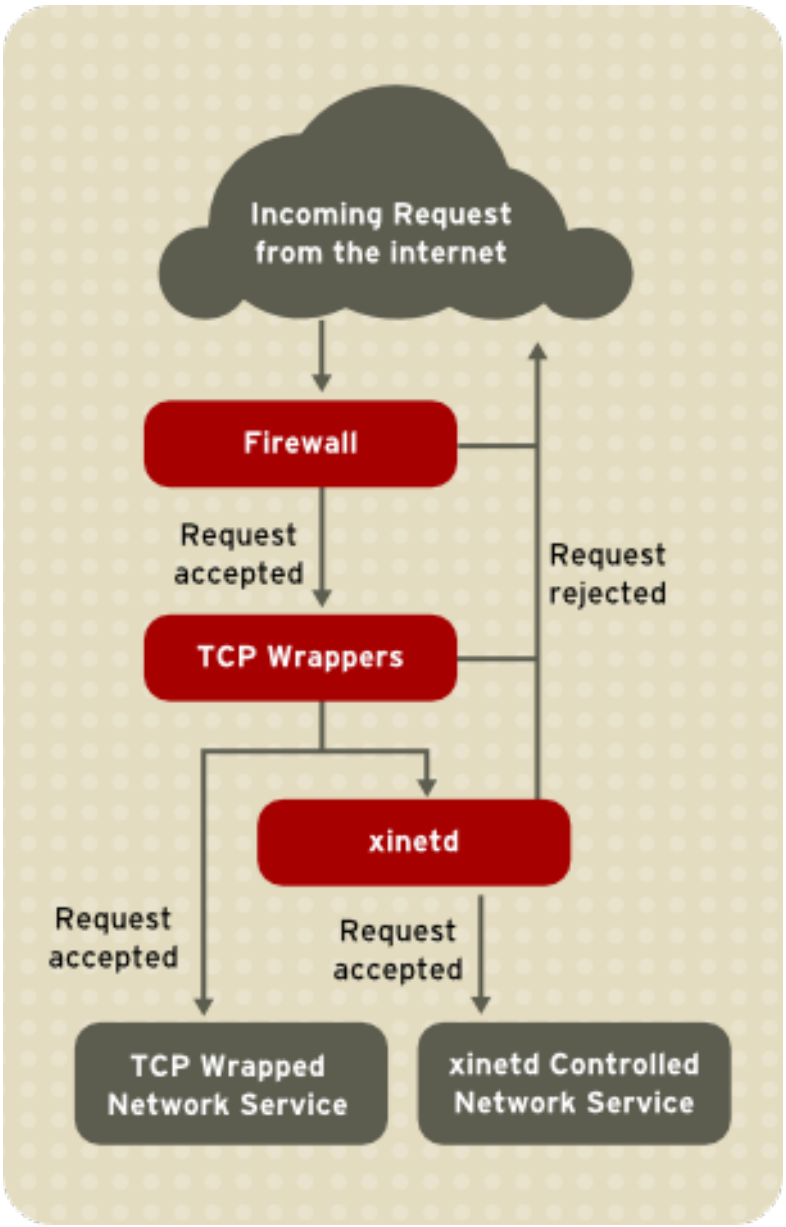
上記の Web サイトのドキュメントは、最後にリリースされた PAM のアップストリームバージョン用で、Red Hat Enterprise Linux に含まれる PAM バージョンについては 100% 正確ではない可能性があります。

48.5. TCP WRAPPER および XINETD

ネットワークサービスへのアクセスの制御は、サーバー管理者向けの最も重要なセキュリティータスクの 1 つです。Red Hat Enterprise Linux は、この目的のためのツールを複数提供します。たとえば、iptablesベースのファイアウォールは、カーネルのネットワークスタック内の非welcomeネットワークパケットを除外します。これを使用するネットワークサービスの場合、TCP Wrapper は、どのホストがラップされたネットワークサービスに接続されているかを定義することで、保護層を追加します。このようなラップされたネットワークサービスの 1 つが xinetd スーパーサーバー です。このサービスは、ネットワークサービスのサブセットへの接続を制御し、アクセス制御をさらに絞り込むため、スーパーサーバーと呼ばれます。

図48.9「ネットワークサービスへのアクセス制御」 は、これらのツールがどのように連携してネットワークサービスを保護するかに関する基本的な図です。

図48.9 ネットワークサービスへのアクセス制御



[D]

本章では、ネットワークサービスへのアクセスを制御する TCP Wrapper と xinetd のロールと、ロギングと使用状況管理の両方を強化するためにこれらのツールがどのように使用されるかを確認することに重点を置いています。iptables でファイアウォールを使用する方法は、「iptables」 を参照してください。

48.5.1. TCP Wrapper

TCP Wrappers パッケージ(tcp_wrappers)はデフォルトでインストールされ、ホストベースのアクセス制御にホストベースのアクセス制御が提供されます。パッケージ内の最も重要なコンポーネントは、/usr/lib/libwrap.a ライブラリーです。一般的に、TCP-wrapped サービスは、libwrap.a ライブラリーに対してコンパイルされているサービスです。

TCP ラップされたサービスに接続を試みると、サービスはまずホストのアクセスファイル (/etc/hosts.allow および /etc/hosts.deny)を参照して、クライアントが接続できるかどうかを判断します。ほとんどの場合、syslog デーモン(syslogd)を使用して要求元のクライアントの名前と要求されたサービスを /var/log/secure または /var/log/messages に書き込みます。

クライアントが接続できる場合、TCP Wrapper は要求されたサービスへの接続をリリースし、クライアントとサーバー間の通信の一部を取りません。

TCP Wrapper は、アクセス制御およびロギングの他に、要求されたネットワークサービスへの接続の制御を拒否または解放する前に、コマンドを実行してクライアントと対話できます。

TCP Wrapper はすべてのサーバー管理者のセキュリティーツールに追加される価値があるため、Red Hat Enterprise Linux 内のほとんどのネットワークサービスは libwrap.a ライブラリーにリンクされています。このようなアプリケーションには、/usr/sbin/sshd、/usr/sbin/sendmail、および /usr/sbin/xinetd が含まれます。

注記

ネットワークサービスバイナリーが libwrap.a にリンクされているかどうかを確認するには、root で以下のコマンドを入力します。

```
ldd <binary-name> | grep libwrap
```

&lt ;binary-name> を、ネットワークサービスバイナリーの名前に置き換えます。

コマンドが出力のないプロンプトに直接返されると、ネットワークサービスは libwrap.a にリンクされていません。

以下の例は、/usr/sbin/sshd が libwrap.a にリンクされていることを示しています。

```
~]# ldd /usr/sbin/sshd | grep libwrap
        libwrap.so.0 => /usr/lib/libwrap.so.0 (0x00655000)
~]#
```

48.5.1.1. TCP Wrapper の利点

TCP Wrapper は、他のネットワークサービス制御手法と比較して、以下の利点があります。

- クライアントとラップされたネットワークサービスの両方に対する透過性：接続クライアントとラップ されたネットワークサービスの両方が、*TCP Wrapper* が使用されていることを認識しません。正当なユーザーは、禁止されたクライアントからの接続中にログに記録され、要求されたサービスに接続されます。

- 複数のプロトコルの一元管理*: TCP Wrapper* は、保護するネットワークサービスとは別に動作し、多くのサーバーアプリケーションが共通のアクセス制御設定ファイルのセットを共有できるため、管理が容易になります。

48.5.2. TCP Wrapper 設定ファイル

クライアントがサービスに接続できるかどうかを判断するには、TCP Wrappers は、一般的に ホストアクセス ファイルと呼ばれる以下の 2 つのファイルを参照します。

- */etc/hosts.allow*

- */etc/hosts.deny*

TCP ラップされたサービスがクライアント要求を受信すると、次の手順を実行します。

1. これは /etc/hosts.allow を参照します。*: TCP ラップされたサービスは /etc/hosts.allow ファイルを順番に解析し、そのサービスに指定された最初のルールを適用します。マッチングルールを見つけると、接続が許可されます。そうでない場合は、次のステップに移動します。*

2. これは、/etc/hosts.deny を参照します。*: TCP ラップされたサービスは、/etc/hosts.deny ファイルを順番に解析します。マッチングルールを見つけると、接続を拒否します。そうでない場合は、サービスへのアクセスを付与します。*

TCP Wrapper を使用してネットワークサービスを保護する際に考慮すべき重要な点を以下に示します。

- *hosts.allow のアクセスルールは最初に適用されるため、hosts.deny で指定されたルールよりも優先されます。したがって、hosts.allow でサービスへのアクセスが許可される*

と、**hosts.deny** 内の同じサービスへのアクセスを拒否するルールは無視されます。

- 各ファイルのルールは上から読み取られ、指定されたサービスの最初のマッチングルールのみが適用されます。ルールの順序は極めて重要です。

- いずれかのファイルにサービスのルールがない場合、またはファイルが存在しない場合には、サービスへのアクセスが許可されます。

- **TCP** ラップされたサービスは、ホストのアクセスファイルのルールをキャッシュしないため、ネットワークサービスを再起動しなくても **hosts.allow** または **hosts.deny** への変更は即座に有効になります。

WARNING

ホストアクセスファイルの最後の行が改行文字ではない場合(**Enter** キーを押して作成)、ファイルの最後のルールが失敗し、エラーが **/var/log/messages** または **/var/log/secure** のいずれかに記録されます。これは、バックスラッシュ文字を使用せずに複数の行にまたがるルールにも当てはまります。以下の例は、これらの状況のいずれかによるルール失敗に関するログメッセージの関連部分を示しています。

```
warning: /etc/hosts.allow, line 20: missing newline or line too long
```

48.5.2.1. アクセスルールのフォーマット

/etc/hosts.allow と **/etc/hosts.deny** の両方の形式は同じです。各ルールはそれぞれの行に指定する必要があります。ハッシュ(#)で始まる空白行または行は無視されます。

各ルールは、以下の基本形式を使用して、ネットワークサービスへのアクセスを制御します。

```
<daemon list>: <client list> [: <option>: <option>: ...]
```

- **<daemon list>**: プロセス名（サービス名ではない ）またはすべてのワイルドカードのコンマ区切りリスト。デーモンリストは、柔軟性を高めるために演算子(「Operator」 を参照)も受け入れます。

- *<client list >: ルールの影響を受けるホストを識別するホスト名、ホスト IP アドレス、特別なパターン、またはワイルドカードのコンマ区切りリスト。クライアントリストは、「Operator」 に記載されている演算子も受け入れ、柔軟性を高めることができます。*

- *<option>: ルールがトリガーされたときに実行されるアクションのオプション のアクションまたはコロン区切りの一覧。オプションフィールドは、拡張、シェルコマンドの起動、アクセスを許可または拒否、およびロギング動作の変更をサポートします。*

注記

上記の目標条件の詳細については、本ガイドの他の場所で参照できます。

- *「ワイルドカード」*
- *「パターン」*
- *「拡張」*
- *「オプションフィールド」*

以下は、基本的なホストのアクセスルールの例です。

```
vsftpd : .example.com
```

このルールは、example.com ドメインのホストから FTP デーモン(vsftpd)への接続を監視するように TCP Wrapper に指示します。このルールが hosts.allow に表示されると、接続は受け入れられます。このルールが hosts.deny に表示されると、接続は拒否されます。

次のホストアクセスルールの例はより複雑で、2 つのオプションフィールドを使用します。

```
sshd : .example.com  \ : spawn /bin/echo `/bin/date` access denied>>/var/log/sshd.log \ : deny
```

各オプションフィールドの前にバックスラッシュ(\)が付くことに注意してください。バックスラッシュを使用すると、長さが原因でルールが失敗することを防ぎます。

このサンプルルールは、example.com ドメインのホストから SSH デーモン(sshd)への接続を試行する場合は、echo コマンドを実行して特別なログファイルに試行を追加し、接続を拒否します。オプションの deny ディレクティブが使用されるため、この行は hosts.allow ファイルに表示された場合でもアクセスを拒否します。利用可能なオプションの詳細は、「オプションフィールド」 を参照してください。

48.5.2.1.1. ワイルドカード

ワイルドカードを使用すると、TCP Wrapper がデーモンまたはホストのグループとより簡単に一致できるようになります。これらは、アクセスルールのクライアントリストフィールドで最も頻繁に使用されます。

以下のワイルドカードを使用できます。

- all: すべてに一致します。これは、デーモンリストとクライアント一覧の両方に使用できます。
- LOCAL - localhost などのピリオド(.)を含まないホストに一致します。
- KNOWN - ホスト名およびホストアドレスが分かっているホスト、またはユーザーが認識されているホストと一致します。
- UNKNOWN: ホスト名またはホストアドレスが不明なホスト、またはユーザーが不明なホストと一致します。
- PARANOID: ホスト名がホストアドレスに一致しないホストと一致します。

注意

KNOW N、 UNKNOWN、および PARANOID ワイルドカードは、正しい操作のために機能する DNS サーバーに依存するため、注意して使用する必要があります。名前解決が中断されると、正当なユーザーがサービスへのアクセスを取得できなくなる可能性があります。

48.5.2.1.2. パターン

アクセスルールのクライアントフィールドでパターンを使用して、クライアントホストのグループをより正確に指定できます。

以下は、クライアントフィールドのエントリーの一般的なパターンの一覧です。

- ピリオド(.)で始まるホスト名 - ホスト名の先頭にピリオドを配置すると、その名のリストされた *コンポーネントを共有するすべてのホストと一致します。以下の例は、example.com ドメイン内のホストに適用されます。*

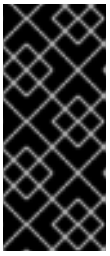
```
ALL : .example.com
```

- ピリオド(.)で終わる IP アドレス *は、IP アドレスの最後にピリオドを配置すると、IP アドレスの最初の数値グループを共有するすべてのホストと一致します。以下の例では、192.168.x.x ネットワーク内の任意のホストに適用されます。*

```
ALL : 192.168.
```

- IP アドレス/ネットマスクのペア - *ネットマスク式は、特定の IP アドレスのグループへのアクセスを制御するパターンとして使用することもできます。以下の例は、アドレス範囲が 192.168.0. 0 から 192.168.1.255 までのホストに適用されます。*

```
ALL : 192.168.0.0/255.255.254.0
```



重要な影響

IPv4 アドレス空間で作業する場合、アドレス/接頭辞長(prefixlen)ペア宣言(CIDR 表記)はサポートされません。この形式を使用できるのは、IPv6 ルールのみです。

- [IPv6 address]/prefixlen ペア - *[net]/prefixlen ペアを、特定の IPv6 アドレスのグループへのアクセスを制御するパターンとして使用することもできます。以下の例では、3ffe:505:2:1: through 3ffe:505:2:1: through 3ffe:505: 2:1:ffff:ffff:ffff : のアドレス範囲を持つホストに適用されます。*

  ```
  ALL : [3ffe:505:2:1::]/64
  ```

- アスタリスク(*): Asterisks を使用して、他のタイプのパターンを含むクライアントリストで混在しない限り、ホスト名または *IP アドレスのグループ全体を照合できます。以下の例では、example.com ドメイン内の任意のホストに適用されます。*

  ```
  ALL : *.example.com
  ```

- スラッシュ(/): クライアントリストがスラッシュで始まる場合は、*ファイル名として処理されます。これは、多数のホストを指定するルールが必要な場合に役立ちます。以下の例では、すべての Telnet 接続の /etc/telnet.hosts ファイルに TCP Wrapper を参照します。*

  ```
  in.telnetd : /etc/telnet.hosts
  ```

他の少ないパターンも TCP Wrappers によって受け入れられます。詳細は、hosts_access の man 5 ページを参照してください。

WARNING

ホスト名およびドメイン名を使用する場合は十分に注意してください。攻撃者は、さまざまなコツを使用して、正確な名前解決を回避できます。さらに、DNS サービスが中断されることにより、許可されたユーザーであってもネットワークサービスが使用できなくなります。したがって、可能な限り IP アドレスを使用することが推奨されます。

48.5.2.1.3. portmap および TCP Wrapper

Portmap の TCP Wrapper の実装は、ホストルックアップをサポートしません。つまり、portmap はホスト名を使用してホストを識別できません。したがって、hosts.allow または hosts.deny の portmap のアクセス制御ルールは、ホストの指定に IP アドレスまたはキーワード ALL を使用する必要があります。

portmap アクセス制御ルールへの変更は、すぐに有効にならない可能性があります。portmap サービスを再起動する必要がある場合があります。

NIS や NFS など、広く使用されているサービスは、動作に portmap に依存するため、これらの制限事項に注意してください。

48.5.2.1.4. Operator

現在、アクセス制御ルールは 1 つの Operator (EXCEPT)を受け入れます。これは、デーモンリストとルールのクライアントリストの両方で使用できます。

EXCEPT 演算子を使用すると、同じルール内のより幅広い一致に対して特定の例外を許可します。

hosts.allow ファイルからの例では、example.com ホストはすべて cracker.example.com 以外のすべてのサービスに接続できます。

```
ALL: .example.com EXCEPT cracker.example.com
```

hosts.allow ファイルの別の例では、192.168.0. X ネットワークのクライアントは FTP 以外のすべてのサービスを使用できます。

```
ALL EXCEPT vsftpd: 192.168.0.
```

注記

多くの場合、EXCEPT 演算子の使用は避けやすくなります。これにより、EXCEPT オペレーターでソートせずに、適切なファイルをすばやくスキャンして、サービスへのアクセスが許可または拒否されるホストを確認できます。

48.5.2.2. オプションフィールド

TCP Wrapper の Red Hat Enterprise Linux 実装は、アクセスを許可および拒否する基本的なルー

ルの他に、オプションフィールド でアクセス制御言語の拡張をサポートします。ホストアクセスルールのオプションフィールドを使用すると、管理者はログ動作の変更、アクセス制御の統合、シェルコマンドの起動など、さまざまなタスクを実行できます。

48.5.2.2.1. ロギング

オプションフィールドを使用すると、管理者は severity ディレクティブを使用してルールのログファシリティと優先度レベルを簡単に変更できます。

以下の例では、example.com ドメイン内のホストから SSH デーモンへの接続が、優先度が emerg のデフォルトの authpriv syslog ファシリティーに記録されます（ファシリティー値が指定されていないため）。

```
sshd : .example.com : severity emerg
```

severity オプションを使用してファシリティーを指定することもできます。以下の例では、ホストによる SSH 接続の試行を example.com ドメインの local0 機能に記録し、優先度が alert にします。

```
sshd : .example.com : severity local0.alert
```

注記

実際には、この例では syslog デーモン(syslogd)が local0 ファシリティーにログを記録するように設定されるまで動作しません。カスタムログ機能の設定に関する詳細は、syslog.conf の man ページを参照してください。

48.5.2.2.2. アクセス制御

オプションフィールドを使用すると、allow ディレクティブまたは deny ディレクティブを最終オプションとして追加して、1 つのルール内のホストを明示的に 許可 または 拒否 することもできます。

たとえば、以下の 2 つのルールは client-1.example.com からの SSH 接続を許可しますが、client-2.example.com からの接続を拒否します。

```
sshd : client-1.example.com : allow
sshd : client-2.example.com : deny
```

ルールごとにアクセス制御を許可することで、オプションフィールドを使用すると、管理者は hosts.allow または hosts.deny のいずれかのすべてのアクセス制御ルールを単一のファイルに統合で

きます。一部の管理者は、これをより簡単にアクセスルールを整理する方法を検討します。

48.5.2.2.3. シェルコマンド

オプションフィールドを使用すると、次の 2 つのディレクティブを使用してシェルコマンドを起動できるようになります。

- **Launchs:** シェルコマンドを子プロセスとして起動します。このディレクティブは、**/usr/sbin/safe_finger** を使用して、要求しているクライアントに関する詳細情報を取得したり、**echo** コマンドを使用して特別なログファイルを作成したりできます。

  以下の例では、**example.com** ドメインから **Telnet** サービスにアクセスしようとするクライアントは、特別なファイルに記録されます。

  ```
  in.telnetd : .example.com \
   : spawn /bin/echo `/bin/date` from %h>>/var/log/telnet.log \
   : allow
  ```

- **twist:** 要求されたサービスを、指定したコマンドに置き換えます。このディレクティブは、多くの場合、侵入者(**honey pots** とも呼ばれる)のトラップを設定するために使用されます。また、接続しているクライアントへのメッセージ送信にも使用できます。**twist** ディレクティブは、ルール行の最後に行われる必要があります。

  以下の例では、**example.com** ドメインから **FTP** サービスにアクセスしようとするクライアントには、**echo** コマンドを使用してメッセージを送信します。

  ```
  vsftpd : .example.com \
   : twist /bin/echo "421 This domain has been black-listed. Access denied!"
  ```

シェルコマンドオプションの詳細は、**man** ページの **hosts_options** を参照してください。

48.5.2.2.4. 拡張

展開は、**spawned** ディレクティブおよび **twist** ディレクティブと併用すると、関係するクライアント、サーバー、およびプロセスに関する情報を提供します。

以下は、サポートされている拡張の一覧です。

- *%a - クライアントの IP アドレスを返します。*
- *% a - サーバーの IP アドレスを返します。*
- *%c: ユーザー名、ホスト名、ユーザー名および IP アドレスなどのさまざまなクライアント情報を返します。*
- *%d: デーモンプロセス名を返します。*
- *%h - クライアントのホスト名（またはホスト名が利用できない場合は IP アドレス）を返します。*
- *%h - サーバーのホスト名（またはホスト名が利用できない場合は IP アドレス）を返します。*
- *%n - クライアントのホスト名を返します。利用できない場合は、unknown が出力されます。クライアントのホスト名とホストアドレスが一致しない場合は、paranoid が出力されます。*
- *%n - サーバーのホスト名を返します。利用できない場合は、unknown が出力されます。サーバーのホスト名とホストアドレスが一致しない場合は、paranoid が出力されます。*
- *%p - デーモンのプロセス ID を返します。*
- *%s - デーモンプロセス、サーバーのホストまたは IP アドレスなどのさまざまな種類のサーバー情報を返します。*
- *%u - クライアントのユーザー名を返します。利用できない場合は、unknown が出力されます。*

以下のルール例では、spawn コマンドとともに拡張を使用して、カスタマイズされたログファイルでクライアントホストを特定します。

example.com ドメインのホストから SSH デーモン(sshd)への接続を試行する場合は、echo コマンドを実行して、クライアントのホスト名(%h 拡張を使用)を含む試行を特別なファイルに記録します。

```
sshd : .example.com  \
 : spawn /bin/echo `/bin/date` access denied to %h>>/var/log/sshd.log \
 : deny
```

同様に、拡張を使用してメッセージをクライアントにパーソナライズすることができます。以下の例では、example.com ドメインから FTP サービスにアクセスしようとすると、サーバーから禁止されていることが通知されます。

```
vsftpd : .example.com \
: twist /bin/echo "421 %h has been banned from this server!"
```

利用可能な拡張と追加のアクセス制御オプションの詳細については、hosts_access の man ページの 5 セクション5 および hosts_ optionsの man ページを参照してください。

TCP Wrapper の詳細は、「関連情報」 を参照してください。

48.5.3. xinetd

xinetd デーモンは TCP ラップされた スーパーサービス で、FTP、IMAP、Telnet などの一般的なネットワークサービスのサブセットへのアクセスを制御します。また、アクセス制御、強化されたロギング、バインディング、リダイレクト、およびリソース使用状況制御のためのサービス固有の設定オプションも提供します。

クライアントが xinetd によって制御されるネットワークサービスへの接続を試みると、スーパーサービスは要求を受け取り、TCP Wrappers アクセス制御ルールを確認します。

アクセスが許可されると、xinetd は、そのサービスの独自のアクセスルールで接続が許可されることを確認します。また、サービスがより多くのリソースを割り当てることができるかどうかや、定義されたルールに違反するかどうかをチェックします。

これらの条件すべてが満たされている場合（つまり、サービスへのアクセスが許可される場合、サービスはリソース制限に到達せず、サービスが定義されたルールに違反することはありません）、xinetd は要求されたサービスのインスタンスを起動し、接続の制御を渡します。接続が確立されると、xinetd はクライアントとサーバー間の通信の一部を取りません。

48.5.4. xinetd 設定ファイル

xinetd の設定ファイルは、以下のとおりです。

- /etc/xinetd.conf: グローバル xinetd 設定ファイル。
- /etc/xinetd.d/: サービス固有のファイルをすべて含むディレクトリー。

48.5.4.1. /etc/xinetd.conf ファイル

/etc/xinetd.conf ファイルには、xinetd のコントロール下にあるすべてのサービスに影響する一般的な設定が含まれています。xinetd サービスが最初に起動されると読み取られるため、設定の変更を有効にするには、xinetd サービスを再起動する必要があります。以下は、/etc/xinetd.conf ファイルの例です。

```
defaults
{
        instances               = 60
  log_type               = SYSLOG authpriv
  log_on_success         = HOST PID
  log_on_failure         = HOST
  cps                    = 25 30
}
includedir /etc/xinetd.d
```

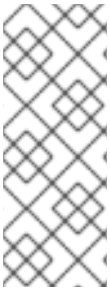
これらの行は、xinetd の以下の側面を制御します。

- instances - xinetd が処理できる同時要求の最大数を指定します。
- log_type: ログエントリーを /var/log/secure ファイルに書き込む authpriv ログファシリティーを使用するように xinetd を設定します。FILE /var/log/xinetdlog などのディレクティブを追加すると、/var/log/ ディレクトリーに xinetdlog という名前のカスタムログファイルが作成されます。
- log_on_success - 正常な接続試行をログに記録するように xinetd を設定します。デフォルトでは、リモートホストの IP アドレスと、要求を処理するサーバーのプロセス ID が記録されます。
- log_on_failure: xinetd が、失敗した接続試行をログに記録するか、接続が拒否されたか

どうかを設定します。

- CP: xinetd を、特定の サービスに対する 1 秒あたり 25 を超える接続を許可するように設定します。この制限を超えると、サービスは 30 秒間廃止されます。

- includedir /etc/xinetd.d/ - /etc/xinetd.d/ ディレクトリーにあるサービス固有の設定ファイルに宣言されたオプションを含めます。詳細は、「/etc/xinetd.d/ ディレクトリー」 を参照してください。



注記

多くの場合、/etc/xinetd.conf の log_on_success 設定および log_on_failure 設定は、サービス固有の設定ファイルでさらに変更されます。したがって、/etc/xinetd.conf ファイルよりも、指定のサービスのログファイルにより多くの情報が表示される可能性があります。詳細は、「ロギングのオプション」 を参照してください。

48.5.4.2. /etc/xinetd.d/ ディレクトリー

/etc/xinetd.d/ ディレクトリーには、xinetd が管理する各サービスの設定ファイルと、ファイルの名前がサービスに関連付けられます。xinetd.conf と同様に、このディレクトリーは xinetd サービスが起動したときにのみ読み取られます。変更を有効にするには、管理者は xinetd サービスを再起動する必要があります。

/etc/xinetd.d/ ディレクトリーのファイルの形式は、/etc/xinetd.conf と同じ規則を使用します。各サービスの設定が個別のファイルに保存される主な理由は、カスタマイズを容易にし、他のサービスに影響を与える可能性を減らすことです。

これらのファイルの構造化方法を理解するには、/etc/xinetd.d/krb5-telnet ファイルを検討してください。

```
service telnet
{
        flags           = REUSE
  socket_type     = stream
  wait            = no
  user            = root
  server          = /usr/kerberos/sbin/telnetd
  log_on_failure  += USERID
  disable         = yes
}
```

これらの行は、telnet サービスのさまざまな側面を制御します。

- service: サービス 名を指定します（通常は /etc/services ファイルに記載されているもののいずれか）。

- flags: 接続の属性を複数設定します。REUSE は、Telnet 接続のソケットを再利用するように xinetd に指示します。

注記

REUSE フラグは非推奨になりました。すべてのサービスは、暗黙的に REUSE フラグを使用するようになりました。

- socket_type: ネットワークソケットの種別を ストリーム に設定します。

- wait: サービスがシングルスレッド(yes)またはマルチスレッド(no)であるかを指定します。

- user: プロセスを実行するユーザー ID を指定します。

- server - 起動するバイナリー実行ファイルを指定します。

- log_on_failure: xinetd.conf にすでに定義されているものに加えて、log_on_failure のログパラメーターを指定します。

- disable: サービスを無効にする（はい）か有効にするかを指定します(は無効)。

これらのオプションとその使用方法の詳細については、xinetd.conf の man ページを参照してください。

48.5.4.3. xinetd 設定ファイルの変更

xinetd が保護するサービスには、さまざまなディレクティブを使用できます。本セクションでは、

一般的に使用されるオプションの一部を説明します。

48.5.4.3.1. ロギングのオプション

以下のロギングオプションは、/etc/xinetd.conf と /etc/xinetd.d/ ディレクトリー内のサービス固有の設定ファイルの両方で利用できます。

以下は、一般的に使用されるロギングオプションの一部です。

- ATTEMPT - 試行に失敗したファクトをログに記録します(log_on_failure)。
- DURATION - サービスがリモートシステムで使用される期間をログに記録します(log_on_success)。
- EXIT - サービスの終了ステータスまたは終了シグナルをログに記録します(log_on_success)。
- HOST: リモートホストの IP アドレスをログに記録します(log_on_failure および log_on_success)。
- PID - 要求を受信するサーバーのプロセス ID をログに記録します(log_on_success)。
- USERID: 全マルチスレッドストリームサービスに対して RFC 1413 で定義された方法を使用してリモートユーザーをログに記録します(log_on_failure および log_on_success)。

ロギングオプションの一覧は、xinetd.conf の man ページを参照してください。

48.5.4.3.2. アクセス制御オプション

xinetd サービスのユーザーは、TCP Wrapper のホストアクセスルールの使用、xinetd 設定ファイルを介したアクセス制御の提供、またはその両方の組み合わせを選択できます。TCP Wrappers ホストアクセス制御ファイルの詳細は、「TCP Wrapper 設定ファイル」 を参照してください。

本セクションでは、xinetd を使用してサービスへのアクセスを制御する方法を説明します。

注記

TCP Wrapper とは異なり、アクセス制御への変更は xinetd 管理者が xinetd サービスを再起動する場合にのみ有効になります。

また、TCP Wrapper とは異なり、xinetd を介したアクセス制御は xinetd によって制御されるサービスにのみ影響します。

xinetd のホストアクセス制御は、TCP Wrapper で使用される方法とは異なります。TCP Wrapper は、すべてのアクセス設定を /etc/hosts.allow と /etc/hosts.deny の 2 つのファイルに配置しますが、xinetd のアクセス制御は /etc/xinetd.d/ ディレクトリーの各サービスの設定ファイルにあります。

以下のホストアクセスオプションは xinetd でサポートされています。

- only_from: 指定されたホストのみがサービスを使用できるようにします。
- no_access: サービスの使用から一覧表示されるホストをブロックします。
- access_times: 特定のサービスを使用できる時間の範囲を指定します。時間の範囲は、24 時間形式の表記(HH:MM-HH:MM)で記述する必要があります。

only_from オプションおよび no_access オプションは、IP アドレスまたはホスト名の一覧を使用するか、ネットワーク全体を指定できます。TCP Wrapper と同様に、xinetd のアクセス制御と強化されたロギング設定を組み合わせると、禁止ホストからの要求をブロックし、各接続試行を詳細に記録することでセキュリティーを強化できます。

たとえば、以下の /etc/xinetd.d/telnet ファイルを使用して、特定のネットワークグループからの Telnet アクセスをブロックし、ユーザーがログインできる全体的な時間範囲を制限できます。

```
service telnet
{
        disable         = no
  flags           = REUSE
```

```
  socket_type     = stream
  wait            = no
  user            = root
  server          = /usr/kerberos/sbin/telnetd
  log_on_failure  += USERID
  no_access       = 172.16.45.0/24
  log_on_success  += PID HOST EXIT
  access_times    = 09:45-16:15
}
```

この例では、10.0.1 .2 などの 10.0.1.0/24 ネットワークのクライアントシステムが Telnet サービスにアクセスしようとすると、以下のメッセージが表示されます。

```
Connection closed by foreign host.
```

さらに、ログインの試行は以下のように /var/log/messages に記録されます。

```
Sep  7 14:58:33 localhost xinetd[5285]: FAIL: telnet address from=172.16.45.107
Sep  7 14:58:33 localhost xinetd[5283]: START: telnet pid=5285 from=172.16.45.107
Sep  7 14:58:33 localhost xinetd[5283]: EXIT: telnet status=0 pid=5285 duration=0(sec)
```

TCP Wrapper を xinetd アクセス制御と併用する場合は、2 つのアクセス制御メカニズム間の関係を理解することが重要です。

以下は、クライアントが接続を要求する際に xinetd が続くイベントシーケンスになります。

1. xinetd デーモンは、libwrap.a ライブラリー呼び出しを使用して、ホストアクセスルールにアクセスします。deny ルールがクライアントと一致する場合、接続は破棄されます。allow ルールがクライアントと一致する場合、接続は xinetd に渡されます。

2. xinetd デーモンは、xinetd サービスと要求されたサービスの両方について、独自のアクセス制御ルールをチェックします。deny ルールがクライアントと一致する場合、接続は破棄されます。それ以外の場合は、xinetd は要求されたサービスのインスタンスを開始し、そのサービスへの接続の制御を渡します。

重要な影響

xinetd アクセス制御とともに TCP Wrapper のアクセス制御を使用する場合は注意が必要です。設定が間違っていると、望ましくない結果が生じる可能性があります。

48.5.4.3.3. バインディングおよびリダイレクトオプション

xinetd のサービス設定ファイルは、サービスを IP アドレスにバインドし、そのサービスの受信要求を別の IP アドレス、ホスト名、またはポートにリダイレクトすることをサポートしています。

バインディングは、サービス固有の設定ファイルの bind オプションで制御され、そのサービスをシステム上の 1 つの IP アドレスにリンクします。これを設定すると、bind オプションは、正しい IP アドレスへの要求のみがサービスにアクセスできるようにします。この方法を使用して、要件に基づいて異なるサービスを異なるネットワークインターフェイスにバインドできます。

これは、複数のネットワークアダプターや IP アドレスが複数あるシステムで特に便利です。このようなシステムでは、セキュアでないサービス(Telnet など)は、プライベートネットワークに接続されたインターフェイスでのみ、インターネットに接続されたインターフェイスではリッスンするように設定できます。

redirect オプションは、IP アドレスまたはホスト名の後にポート番号を受け入れます。このサービスの要求を指定されたホストおよびポート番号にリダイレクトするようにサービスを設定します。この機能は、同じシステム上の別のポート番号を参照したり、要求を同じマシン上の別の IP アドレスにリダイレクトしたり、完全に異なるシステムとポート番号に要求したり、これらのオプションの組み合わせを移行したりできます。したがって、システムで特定のサービスに接続するユーザーは、中断することなく別のシステムに再ルーティングされる可能性があります。

xinetd デーモンは、要求元のクライアントマシンと実際には 2 つのシステム間でデータを転送して、接続期間中に有効なプロセスを生成することで、このリダイレクトを実行できます。

bind および redirect オプションの利点は、一緒に使用されると最も明確になります。サービスをシステム上の特定の IP アドレスにバインドし、このサービスの要求を最初のマシンのみが確認できる 2 つ目のマシンにリダイレクトすることで、内部システムを使用して、完全に異なるネットワークのサービスを提供できます。または、これらのオプションを使用して、マルチホームマシン上の特定のサービスの公開を既知の IP アドレスに制限したり、そのために特に設定された別のマシンにサービスの要求をリダイレクトしたりできます。

たとえば、Telnet サービス向けにこの設定でファイアウォールとして使用されるシステムについて考えてみましょう。

```
service telnet
{
        socket_type  = stream
  wait   = no
  server   = /usr/kerberos/sbin/telnetd
  log_on_success  += DURATION USERID
  log_on_failure  += USERID
```

```
  bind                = 123.123.123.123
  redirect            = 10.0.1.13 23
}
```

このファイルの bind および redirect オプションは、マシンの Telnet サービスが、インターネットに接続されている外部 IP アドレス(123.123.123.123)にバインドされていることを確認します。さらに、123.123. 123.123 に送信された Telnet サービスの要求は、2 番目のネットワークアダプターを介して、ファイアウォールおよび内部システムのみがアクセスできる内部 IP アドレス(10.0.1.13)にリダイレクトされます。次に、ファイアウォールは 2 つのシステム間の通信を送信し、接続システムは、実際に別のマシンに接続されているときに 123.123.123.123 に接続していると見なします。

この機能は、ブロードバンド接続と固定 IP アドレスを 1 つだけ持つユーザーに特に便利です。NAT (Network Address Translation)を使用する場合、ゲートウェイマシンの背後にあるシステムは、ゲートウェイシステム外からは内部のみの IP アドレスを使用することができません。ただし、xinetd が制御する特定のサービスが bind および redirect オプションで設定されている場合、ゲートウェイマシンは外部システムと、そのサービスを提供するように設定された特定の内部マシンとの間のプロキシーとして機能します。さらに、追加の保護には、xinetd のさまざまなアクセス制御とロギングオプションも利用できます。

48.5.4.3.4. リソース管理オプション

xinetd デーモンは、DoS (Denial of Service)攻撃からの基本的な保護レベルを追加できます。以下は、このような攻撃の効果を制限するのに役立つディレクティブの一覧です。

- *per_source: ソース IP アドレスあたりのサービスの最大インスタンス数を定義します。これは整数を引数としてのみ使用でき、xinetd.conf と、xinetd.d/ ディレクトリーのサービス固有の設定ファイルの両方で使用できます。*

- *CP: 1 秒あたりの接続の最大数を定義します。このディレクティブは、空白で区切られた 2 つの整数引数を取ります。最初の引数は、1 秒あたりのサービスに対して許可される接続の最大数です。2 つ目の引数は、サービスを再度有効にする前に xinetd が待機する必要のある秒数です。これは整数としてのみ使用でき、xinetd.conf ファイルまたは xinetd.d/ ディレクトリー内のサービス固有の設定ファイルで使用できます。*

- *max_load: サービスの CPU 使用率または負荷平均しきい値を定義します。浮動小数点数の引数を受け入れます。*

  平均負荷は、特定の時点でアクティブなプロセス数に関する大まかな測定値です。平均負荷の詳細は、uptime コマンド、who コマンド、および procinfo コマンドを参照してください。

xinetd では、より多くのリソース管理オプションを利用できます。詳細は、xinetd.conf の man ページを参照してください。

48.5.5. 関連情報

TCP Wrapper および xinetd の詳細は、システムドキュメントおよびインターネットを参照してください。

48.5.5.1. インストールされているドキュメント

システムに関するドキュメントは、TCP Wrapper、xinetd、および アクセス制御の追加設定オプションの検索を開始するのに適した場所です。

- /usr/share/doc/tcp_wrappers- <version> / - このディレクトリーには、TCP Wrapper の仕組みと、存在するさまざまなホスト名およびホストアドレスのスプーフィングリスクを説明する README ファイルが含まれています。

- /usr/share/doc/xinetd- <version> / - このディレクトリーには、アクセス制御の側面を説明する README ファイルと、/etc/xinetd.d/ ディレクトリーのサービス固有の設定ファイルを変更するさまざまな概念を持つ sample.conf ファイルが含まれています。

- TCP Wrapper と xinetd関連の man ページ : TCP Wrapper と xinetd に関連するさまざまなアプリケーションおよび設定ファイル用の man ページが多数あります。以下は、より重要な man ページの一部です。

  サーバーアプリケーション

  - man xinetd: xinetd の man ページ

  設定ファイル

  - man 5 hosts_access: TCP Wrapper の man ページは、アクセス制御ファイルをホストします。

  - man hosts_options: TCP Wrappers オプションフィールドの man ページです。

- man xinetd.conf: xinetd 設定オプションを一覧表示する man ページです。

48.5.5.2. 便利な Web サイト

- http://www.xinetd.org/ - サンプル設定ファイル、機能の完全なリスト、および情報 FAQ を含む xinetd のホーム。

- http://www.macsecurity.org/resources/xinetd/tutorial.shtml: 特定のセキュリティー目標を達成するためにデフォルトの xinetd 設定ファイルを最適化するさまざまな方法を説明する詳細なチュートリアルです。

48.5.5.3. 関連書籍

- 『Hacking Linux Exposed』 by 2007 Hatch, James Lee, and George Kurtz; Osbourne/McGraw-Hill - TCP Wrappers および xinetd に関する情報を含む優れたセキュリティーリソース。

48.6. KERBEROS

ネットワーク内のシステムのセキュリティーと整合性は望ましくない可能性があります。複数の管理者がネットワークで実行されているサービスや、これらのサービスが使用される方法を追跡するためにだけの時間を占有することができます。

さらに、ネットワークサービスへのユーザーの認証は、従来の FTP プロトコルおよび Telnet プロトコルを使用したネットワーク上で暗号化されていないパスワードの転送によって明らかであるため、プロトコルが使用する方法が本質的に安全でないと、危険を証明することができます。

Kerberos は、安全でない認証メソッドを許可するプロトコルの必要性をなくすことです。これにより、ネットワークセキュリティー全体が強化されます。

48.6.1. Kerberos とは

Kerberos は MIT によって作成されたネットワーク認証プロトコルで、対称キー暗号を使用します。[17] ネットワークサービスに対してユーザーを認証します。つまり、パスワードがネットワーク上で送信されることはありません。

そのため、ユーザーが Kerberos を使用してネットワークサービスに対して認証を行う際に、ネットワークトラフィックを監視してパスワードの収集を図っている不正なユーザーを効果的に阻止することができます。

48.6.1.1. Kerberos の利点

従来のネットワークサービスの多くは、パスワードベースの認証スキームを使用します。このようなスキームでは、ユーザー名とパスワードを指定して、特定のネットワークサーバーに対してユーザーが認証する必要があります。ただし、多くのサービスに対する認証情報の送信は暗号化されません。このようなスキームをセキュアにするには、ネットワークを外部からアクセスできないようにする必要があり、ネットワーク上のすべてのコンピューターおよびユーザーが信頼でき、信頼できるものでなければなりません。

その場合でも、インターネットに接続されたネットワークは安全であるとは想定されなくなります。ネットワークへのアクセスを取得する攻撃者は、パケットスニファーとも呼ばれる単純なパケットアナライザーを使用してユーザー名とパスワードを傍受し、ユーザーアカウントやセキュリティーインフラストラクチャー全体の整合性を損なうことができます。

Kerberos の主な設計の目的は、ネットワーク全体で暗号化されていないパスワードの送信を排除することです。適切に使用されていると、Kerberos は、パケットスニッファーがネットワーク上でもたらす脅威を効果的に排除します。

48.6.1.2. Kerberos の欠点

Kerberos は一般的で深刻なセキュリティーの脅威を排除しますが、さまざまな理由で実装が困難な場合があります。

- このタスクを実行する自動メカニズムがないため、/etc/passwd や /etc/shadow などの標準の UNIX パスワードデータベースから Kerberos パスワードデータベースにユーザーパスワードを移行することはできます。オンラインの Kerberos FAQ で質問 2.23 を参照してください。

  http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html

- Kerberos は、ほとんどの Red Hat Enterprise Linux サーバーが使用する PAM (Pluggable Authentication Modules)システムとの部分的な互換性しかありません。この問題の詳細は、「Kerberos および PAM」 を参照してください。

- Kerberos は、各ユーザーが信頼されていて、信頼できないネットワーク上で信頼できないホストを使用していることを前提としています。その主な目的は、暗号化されていないパス

ワードがそのネットワーク上で送信されないようにすることです。ただし、適切なユーザー以外のユーザーが、認証に使用されるチケットを発行するホスト(キー配布センター (KDC)と呼ばれる)にアクセスできる場合、Kerberos 認証システム全体が危険にさらされます。

- アプリケーションが Kerberos を使用するには、そのソースを変更して Kerberos ライブラリーに適切な呼び出しを行う必要があります。この方法で変更したアプリケーションは、Kerberos 対応 または Kerberized として考慮されます。アプリケーションによっては、アプリケーションのサイズや設計により、非常に問題になる場合があります。その他の互換性のないアプリケーションでは、サーバーとクライアントが通信する方法に変更を加える必要があります。ここでも、詳細なプログラミングが必要になる場合があります。デフォルトでは Kerberos サポートのないクローズソースアプリケーションは、多くの場合最も問題となります。

- Kerberos は、すべてまたはなしのソリューションです。ネットワークで Kerberos を使用する場合は、Kerberos 以外の対応サービスに転送される暗号化されていないパスワードはすべて危険にさらされます。したがって、このネットワークは Kerberos を使用する利点はありません。Kerberos でネットワークを保護するには、暗号化されていないパスワードを送信するすべて のクライアント/サーバーアプリケーションの Kerberos 対応バージョンを使用するか、そのようなクライアント/サーバーアプリケーションをまったく使用しない必要があります。

48.6.2. Kerberos の用語

Kerberos には、サービスのさまざまな側面を定義する独自の用語があります。Kerberos の仕組みを理解する前に、以下の用語を理解することが重要です。

認証サーバー(AS)

ユーザーがサービスにアクセスできるようになる目的のサービスのチケットを発行するサーバー。AS は、要求でクレデンシャルを送信しない、または送信していないクライアントから要求に応答します。通常、TGT (Ticket-granting Ticket)を発行することで、TGS (Ticket-granting Server)サービスへのアクセスを取得するために使用されます。AS は通常、キー配布センター (KDC)と同じホストで実行されます。

ciphertext

暗号化されたデータ。

クライアント

Kerberos からチケットを取得できるネットワーク上のエンティティー（ユーザー、ホスト、またはアプリケーション）。

credentials

特定のサービスのクライアントの ID を確認する電子クレデンシャルの一時的なセット。チケットとも呼ばれます。

認証情報キャッシュまたはチケットファイル

ユーザーとさまざまなネットワークサービス間の通信を暗号化する鍵を含むファイル。Kerberos 5 は、共有メモリーなどの他のキャッシュタイプを使用するためのフレームワークをサポートしますが、ファイルはより詳細にサポートされます。

暗号化ハッシュ

ユーザーの認証に使用される一方向ハッシュ。暗号化されていないデータを使用するよりも安全性が高くなりますが、経験のあるクラッカーの暗号は比較的簡単です。

GSS-API

Generic Security Service Application Program Interface (RFC-2743 で定義)は、The Internet Engineering Task Force (Internet Engineering Task Force)によって公開される一連の関数です。この API は、基盤となるメカニズムに関する特定の知識がなくても、クライアントおよびサービスによって相互の認証に使用されます。ネットワークサービス(cyrus-IMAP など)が GSS-API を使用する場合は、Kerberos を使用して認証できます。

ハッシュ

ハッシュ値 とも呼ばれます。ハッシュ関数 で文字列を渡すことで生成された値。これらの値は、通常、送信データが改ざんされないようにするために使用されます。

ハッシュ関数

入力データからデジタルフィンガープリントを生成する方法。これらの関数は、ハッシュ値 を生成するためにデータを再編成、変換、または変更します。

key

他のデータの暗号化または復号化時に使用されるデータ。暗号化されたデータは、適切なキーなしで復号できず、クラッカーの一部で調整が非常に良くなります。

キー配布センター(KDC)

Kerberos チケットを発行し、通常は ticket-granting サーバー(TGS)と同じホストで実行されるサービス。

keytab （またはキーテーブル）

プリンシパルとそのキーの暗号化されていないリストが含まれるファイル。サーバーは、kinit を使用する代わりにキータブファイルから必要なキーを取得します。デフォルトのキータブファイルは /etc/krb5.keytab です。KDC 管理サーバー /usr/kerberos/sbin/kadmind は、他のファイルを使用する唯一のサービスです(/var/kerberos/krb5kdc/kadm5.keytabを使用します)。

kinit

kinit コマンドを使用すると、すでにログインしているプリンシパルが、最初の TGT (Ticket-granting Ticket)を取得してキャッシュできます。詳細は、kinit の man ページを参照してください。

プリンシパル（またはプリンシパル名）

プリンシパルは、Kerberos を使用した認証が許可されるユーザーまたはサービスの一意の名前です。プリンシパルの形式は root[/instance]@REALM に従います。一般的なユーザーの場合、root はログイン ID と同じです。インスタンス はオプションです。プリンシパルにインスタンスがある場合、これはスラッシュ("/")でルートから分離されます。空の文字列("")は有効なインスタンス（デフォルトの NULL インスタンスとは異なる）とみなされますが、これを使用すると混乱が生じる可能性があります。レルムのすべてのプリンシパルには独自のキーがあり、ユーザーはパスワードから派生するか、またはサービスにランダムに設定されます。

realm

KDC と呼ばれる 1 つ以上のサーバーと、潜在的に多数のクライアントで設定される Kerberos を使用するネットワーク。

サービス

ネットワーク経由でアクセスするプログラム。

ticket

特定のサービスのクライアントの ID を確認する電子クレデンシャルの一時的なセット。credentials とも呼ばれます。

Ticket-Granting サーバー(TGS)

ユーザーがサービスにアクセスできるようになる目的のサービスのチケットを発行するサーバー。通常、TGS は KDC と同じホストで実行されます。

TGT (Ticket-Granting Ticket)

KDC から適用せずにクライアントが追加のチケットを取得できるようにする特別なチケット。

暗号化されていないパスワード

プレーンテキストの人間が判読できるパスワード。

48.6.3. Kerberos の仕組み

Kerberos は、ユーザー名/パスワードの認証方法とは異なります。Kerberos は、各ユーザーを各ネットワークサービスに対して認証する代わりに、対称暗号化と信頼できるサードパーティー(KDC)を使用して、ネットワークサービススイートに対してユーザーを認証します。ユーザーが KDC に対して認証を行うと、KDC はそのセッションに固有のチケットをユーザーのマシンに送信し、Kerberos 対応のサービスは、ユーザーがパスワードを使用して認証を要求するのではなく、ユーザーのマシンでチケットを検索します。

Kerberos 対応のネットワーク上のユーザーがワークステーションにログインすると、認証サーバーからの TGT の要求の一部としてプリンシパルが KDC に送信されます。この要求は、ログインプログラムによりユーザーに透過的となるように送信したり、ユーザーのログイン後に kinit プログラムから送信したりできます。

次に KDC はデータベース内でプリンシパルをチェックします。プリンシパルが見つかると、KDC は TGT を作成します。TGT は、ユーザーのキーを使用して暗号化され、そのユーザーに戻ります。

クライアント上のログインまたは kinit プログラムはユーザーの鍵を使用して TGT を復号化します。これは、ユーザーのパスワードから計算します。ユーザーのキーはクライアントマシン上でのみ使用され、ネットワーク上では送信されません。

TGT は、一定期間 (通常は 10 から 15 時間) の後に期限切れに設定され、クライアントマシンの認証情報キャッシュに保存されます。セキュリティーの破られた TGT が攻撃者に利用される時間を短くするために、有効期限が設定されています。TGT の発行後、TGT の有効期限が切れるまで、もしくはログアウトして再度ログインするまで、ユーザーはパスワードを再入力する必要はありません。

ユーザーがネットワークサービスにアクセスする必要がある場合は、クライアントソフトウェアは TGT を使用して TGS からその特定のサービスの新しいチケットを要求します。サービスチケットはその後、そのサービスに対して透過的にユーザーを認証するために使用されます。

WARNING

ネットワーク上のユーザーが Kerberos 以外の対応サービスに対してパスワードをプレーンテキストで送信して認証すると、Kerberos システムが危険にさらされる可能性があります。Kerberos 以外の対応サービスの使用は推奨されません。このようなサービスには、Telnet や FTP が含まれます。SSH や SSL で保護されたサービスなどの他の暗号化プロトコルの使用が推奨されますが、理想的ではありません。

これは、Kerberos 認証の仕組みの幅広い概要です。詳細は、「関連情報」 を参照してください。

注記

Kerberos は、正しく機能するために以下のネットワークサービスに依存します。

- ネットワーク上のマシン間のクロック同期の概算。

  クロック同期プログラムは、ntpd などのネットワークに設定する必要があります。ネットワークタイムプロトコルサーバーのセットアップに関する詳細は、/usr/share/doc/ntp- <version-number> /index.html を参照してください。< version-number > は、システムにインストールされている ntp パッケージのバージョン番号に置き換えます。

- DNS (Domain Name Service)

  ネットワーク上の DNS エントリーとホストがすべて適切に設定されていることを確認する必要があります。詳細は、/usr/share/doc/krb5-server-<version-number > の『Kerberos V5 System Administrator's Guide』 を参照してください。< version-number > は、システムにインストールされている krb5-server パッケージのバージョン番号に置き換えてください。

48.6.4. Kerberos および PAM

Kerberos 対応のサービスは現在、PAM (Pluggable Authentication Modules)を利用していません。これらのサービスは、PAM を完全に迂回します。ただし、pam_krb5 モジュール(pam_krb5 パッケージで提供)がインストールされていると、PAM を使用するアプリケーションは認証に Kerberos を使用できます。pam_krb5 パッケージには、ログイン や gdm などのサービスがユーザーを認証し、パスワードを使用して初期認証情報を取得できるようにする設定ファイルのサンプルが含まれています。ネットワークサーバーへのアクセスが常に、IMAP などの GSS-API を使用する Kerberos 対応のサービスまたはサービスを使用して実行される場合、ネットワークは適度に安全であると見なされます。

ヒント

管理者は、Kerberos パスワードを使用してほとんどのネットワークサービスをユーザーが認証できないように注意してください。これらのサービスが使用するプロトコルの多くは、ネットワーク経由で送信する前にパスワードを暗号化せず、Kerberos システムの利点を破棄します。たとえば、ユーザーは Kerberos 認証に使用するパスワードと同じパスワードで Telnet サービスに対して認証することはできません。

48.6.5. Kerberos 5 サーバーの設定

Kerberos を設定するときは、KDC を最初にインストールします。スレーブサーバーをセットアップする必要がある場合は、最初にマスターをインストールします。

最初の Kerberos KDC を設定するには、以下の手順に従います。

1. Kerberos を設定する前に、時刻同期と DNS がすべてのクライアントおよびサーバーマシンで正しく機能していることを確認します。Kerberos サーバーとそのクライアント間の時刻同期に特に注意してください。サーバーとクライアント間の時間差が 5 分を超える場合（これは Kerberos 5 で設定可能）、Kerberos クライアントはサーバーに認証できません。この時間同期は、攻撃者が古い Kerberos チケットを使用して有効なユーザーとしてマスカレードしないようにするために必要です。

   Kerberos が使用されていない場合でも、NTP (Network Time Protocol)と互換性のあるクライアント/サーバーネットワークを設定することが推奨されます。Red Hat Enterprise Linux には、この目的のために ntp パッケージが含まれています。ネットワークタイムプロトコルサーバーの設定方法は /usr/share/doc/ntp-< version-number > /index.html (<version-number> はシステムにインストールされている ntp パッケージのバージョン番号)、および NTP の詳細は http://www.ntp.org を参照してください。

2. KDC を実行する専用マシンにkrb5- libs、krb5-server、および krb5-workstation パッケージをインストールします。このマシンは非常に安全である必要があります。可能な場合は、KDC 以外のサービスを実行しないでください。

3. レルム名と、ドメインからレルムへのマッピングを反映するように /etc/krb5.conf と /var/kerberos/krb5kdc/kdc.conf 設定ファイルを編集します。シンプルなレルムは、EXAMPLE.COM と example.com のインスタンスを正しいドメイン名で置き換えることで設定できます。これは、正しい形式で大文字と小文字の名前を維持することが確実にでき、KDC を kerberos.example.com から Kerberos サーバーの名前に変更することで設定できます。通常、レルム名はすべて大文字で、DNS ホスト名およびドメイン名はすべて小文字になります。これらの設定ファイルの形式に関する詳細は、それぞれの man ページを参照してください。

4. シェルプロンプトから kdb5_util ユーティリティーを使用してデータベースを作成します。

   ```
   /usr/kerberos/sbin/kdb5_util create -s
   ```

   この create コマンドは、Kerberos レルムのキーを保存するデータベースを作成します。-s スイッチは、マスターサーバーキーが保存される stash ファイルの作成を強制します。キー

の読み取り元となる stash ファイルがない場合、Kerberos サーバー (krb5kdc) は起動時に毎回マスターサーバーのパスワード (このパスワードを使って鍵を再生成できる) を要求します。

5.

/var/kerberos/krb5kdc/kadm5.acl ファイルを編集します。このファイルは、Kerberos データベースへの管理アクセス権限およびそのアクセスレベルを決定するために kadmind によって使用されます。ほとんどの組織は、次の 1 行で取得できます。

```
*/admin@EXAMPLE.COM  *
```

多くのユーザーは、データベース内で単一のプリンシパルで表されます (joe@EXAMPLE.COM などの NULL または空のインスタンス)。この設定では、admin （例：joe/admin@EXAMPLE.COM）のインスタンスを持つ 2 番目のプリンシパルを持つユーザーは、レルムの Kerberos データベース全体のフルパワーをワイドできます。

kadmind がサーバーで起動した後、ユーザーはレルム内のいずれかのクライアントまたはサーバーで kadmin を実行することで、そのサービスにアクセスできます。ただし、kadm5.acl ファイルにリストされているユーザーのみが、自身のパスワードを変更することを除いて、データベースを編集できます。

注記

この kadmin ユーティリティーはネットワーク経由で kadmind サーバーと通信し、Kerberos を使用して認証を処理します。したがって、ネットワーク経由でサーバーに接続してサーバーを管理するには、最初のプリンシパルがすでに存在している必要があります。kadmin.local コマンドを使用して最初のプリンシパルを作成します。これは、KDC と同じホストで使用するように特別に設計されており、認証に Kerberos を使用しません。

KDC ターミナルに以下の kadmin.local コマンドを入力して、最初のプリンシパルを作成します。

```
/usr/kerberos/sbin/kadmin.local -q "addprinc username/admin"
```

6.

以下のコマンドを使用して Kerberos を起動します。

```
service krb5kdc start
service kadmin start
service krb524 start
```

7.

kadmin 内で addprinc コマンドを使用してユーザーのプリンシパルを追加しま

す。kadmin および kadmin.local は、KDC へのコマンドラインインターフェイスです。そのため、addprinc などのコマンドの多くは、kadmin プログラムの起動後に利用できます。詳細については kadmin の man ページを参照してください。

8. *KDC がチケットを発行していることを確認します。まず、kinit を実行してチケットを取得し、認証情報キャッシュファイルに保存します。次に、klist を使用してキャッシュ内の認証情報の一覧を表示し、kdestroy を使用して、キャッシュと含まれる認証情報を破棄します。*

注記

デフォルトでは、kinit は、(Kerberos サーバーではなく)同じシステムログインユーザー名を使用して認証を試みます。ユーザー名が Kerberos データベースのプリンシパルに対応しない場合は、kinit がエラーメッセージを発行します。その場合は、コマンドライン(kinit <principal>)の引数として、正しいプリンシパルの名前とともにkinit を提供します。

これらの手順が完了したら、Kerberos サーバーが稼働しているはずです。

48.6.6. Kerberos 5 クライアントの設定

Kerberos 5 クライアントの設定は、サーバーの設定とは関係ありません。少なくとも、クライアントパッケージをインストールし、各クライアントに有効な krb5.conf 設定ファイルを提供します。クライアントシステムにリモートでログインする方法として、ssh と slogin が推奨されていますが、Kerberos 化されたバージョンの rsh と rlogin は引き続き利用できますが、それらをデプロイするにはさらに多くの設定変更を行う必要があります。

1. *Kerberos クライアントと KDC の間で時刻同期が行われていることを確認します。詳細は、「Kerberos 5 サーバーの設定」 を参照してください。さらに、Kerberos クライアントプログラムを設定する前に、DNS が Kerberos クライアントで適切に機能していることを確認します。*

2. *すべてのクライアントマシンにkrb5-libsおよびkrb5-workstationパッケージをインストールします。各クライアントに有効な /etc/krb5.conf ファイルを指定します（通常は、KDC で使用される同じ krb5.conf ファイルになります）。*

3. *レルムのワークステーションが Kerberos を使用して、ssh または Kerberized rsh または rlogin を使用して接続するユーザーを認証する前に、Kerberos データベースに独自のホストプリンシパルが必要になります。sshd、kshd、および klogind サーバープログラムはすべて、ホスト サービスのプリンシパルのキーへのアクセスが必要になります。さらに、kerberized rsh および rlogin サービスを使用するには、そのワークステーションに xinetd パッケージがインストールされている必要があります。*

kadmin を使用して、KDC 上のワークステーション用のホストプリンシパルを追加します。この場合のインスタンスはワークステーションのホスト名です。kadmin の addprinc コマンドに -randkey オプションを指定してプリンシパルを作成し、それをランダムな鍵に割り当てます。

```
addprinc -randkey host/blah.example.com
```

プリンシパルが作成されたので、ワークステーション自体で **kadmin** *を実行し、* **kadmin** *内で* **ktadd** *コマンドを使用すると、ワークステーション 用の鍵を抽出できるようになりました。*

```
ktadd -k /etc/krb5.keytab host/blah.example.com
```

4.

その他の Kerberos ネットワークサービスを使用するには、まずそれらを起動する必要があります。以下は、一般的な Kerberos サービスの一覧と、そのサービスを有効にする手順です。

- *SSH - クライアントとサーバーの設定で GSSAPIAuthentication が有効になっている場合、OpenSSH は GSS-API を使用してサーバーにユーザーを認証します。クライアントでも GSSAPIDelegateCredentials が有効な場合は、ユーザーの認証情報がリモートシステムで利用可能になります。*

- *rsh および rlogin: kerberized バージョンの rsh および rlogin を使用するには、klogin、eklogin、および kshell を有効にします。*

- *telnet - kerberized Telnet を使用するには、krb5-telnet を 有効にする必要があります。*

- *FTP: FTP アクセスを提供するには、ftp のルートでプリンシパルのキーを作成して展開します。インスタンスを FTP サーバーの完全修飾ホスト名に設定し、gssftp を有効にするようにしてください。*

- *IMAP - kerberized IMAP サーバーを使用するには、cyrus-sasl-gssapi パッケージがインストールされている場合は、cyrus-imap パッケージも Kerberos 5 を使用します。cyrus-sasl-gssapi パッケージには、GSS-API 認証をサポートする Cyrus SASL プラグインが含まれます。Cyrus IMAP は、cyrus ユーザーが /etc/krb5.keytab で適切な鍵を見つけ、プリンシパルのルートが imap (kadminで作成された)に設定されている限り Kerberos で適切に機能します。*

cyrus-imap の代替は、dovecot パッケージにあります。これは、Red Hat Enterprise Linux にも含まれています。このパッケージには IMAP サーバーが含まれていますが、現在のところ GSS-API および Kerberos には対応していません。

- CVS - kerberized CVS サーバーを使用するには、gserver は root が cvs のプリンシパルを使用し、それ以外の場合は CVS pserver と同じです。

サービスを有効にする方法は、18章 を参照してください。

48.6.7. ドメインからレルムへのマッピング

クライアントが特定のサーバーで実行されているサービスにアクセスしようとすると、サービス名(ホスト)とサーバー名(foo.example.com)を認識しますが、ネットワーク上に複数のレルムをデプロイする可能性があるため、サービスが存在するレルムの名前を推測する必要があります。

デフォルトでは、レルム名はサーバーの DNS ドメイン名になり、大文字になります。

foo.example.org → EXAMPLE.ORG

foo.example.com → EXAMPLE.COM

foo.hq.example.com → HQ.EXAMPLE.COM

設定によっては、これで十分ですが、派生したレルム名は存在しないレルムの名前になります。このような場合は、サーバーの DNS ドメイン名からレルムの名前へのマッピングをクライアントシステムの krb5.conf の domain_realm セクションで指定する必要があります。以下に例を示します。

```
[domain_realm]
.example.com = EXAMPLE.COM
example.com = EXAMPLE.COM
```

上記の設定では、2 つのマッピングを指定します。最初のマッピングは、example.com DNS ドメイン内のシステムが EXAMPLE.COM レルムに属することを指定します。2 つ目は、正確な名前 "example.com" を持つシステムもレルムにあることを指定します。（ドメインと特定のホストの違いは、最初の "." の有無によってマークされます。） マッピングは DNS に直接保存することもできます。

48.6.8. セカンダリー KDC の設定

いくつかの理由により、特定のレルムに対して複数の KDC を実行することを選択できます。このシナリオでは、1 つの KDC (マスター KDC)が書き込み可能なレルムデータベースの書き込み可能なコ

ピーを保持し、**kadmind** （レルムの管理サーバーでも）を実行し、**1** つ以上の **KDC (**スレーブ**KDC)**はデータベースの読み取り専用コピーを保持し、**kpropd** を実行します。

マスタースレーブを伝達するステップでは、マスター **KDC** がデータベースを一時ダンプファイルにダンプして、そのファイルを各スレーブに送信する必要があります。このファイルは、そのダンプファイルのコンテンツでこれ以前に受信したデータベースの読み取り専用コピーを上書きします。

スレーブ **KDC** を設定するには、マスター **KDC** の **krb5.conf** および **kdc.conf** ファイルがスレーブ **KDC** にコピーされていることを確認します。

マスター **KDC** の **root** シェルから **kadmin.local** を起動し、その **add_principal** コマンドを使用してマスター **KDC** の ホスト サービスの新規エントリーを作成し、**ktadd** コマンドを使用してサービス用にランダムキーを同時に設定し、マスターのデフォルト **keytab** ファイルに保存します。このキーは、**kprop** コマンドがスレーブサーバーに対して認証するために使用されます。インストールするスレーブサーバーの数に関係なく、これは一度だけ実行する必要があります。

```
~]# kadmin.local -r EXAMPLE.COM
Authenticating as principal root/admin@EXAMPLE.COM with password.
kadmin: add_principal -randkey host/masterkdc.example.com
Principal "host/host/masterkdc.example.com@EXAMPLE.COM" created.
kadmin: ktadd host/masterkdc.example.com
Entry for principal host/masterkdc.example.com with kvno 3, encryption type Triple DES cbc mode
with \
 HMAC/sha1 added to keytab WRFILE:/etc/krb5.keytab.
Entry for principal host/masterkdc.example.com with kvno 3, encryption type ArcFour with
HMAC/md5 \
 added to keytab WRFILE:/etc/krb5.keytab.
Entry for principal host/masterkdc.example.com with kvno 3, encryption type DES with HMAC/sha1
added \
 to keytab WRFILE:/etc/krb5.keytab.
Entry for principal host/masterkdc.example.com with kvno 3, encryption type DES cbc mode with
RSA-MD5 \
 added to keytab WRFILE:/etc/krb5.keytab.
kadmin: quit
```

スレーブ **KDC** で **root** シェルから **kadmin** を起動し、その **add_principal** コマンドを使用してスレーブ **KDC** の ホスト サービスの新規エントリーを作成し、**kadmin** の**ktadd** コマンドを使用して、サービスのランダムキーを同時に設定し、スレーブのデフォルト **keytab** ファイルに保存します。このキーは、クライアントの認証時に **kpropd** サービスによって使用されます。

```
~]# kadmin -p jimbo/admin@EXAMPLE.COM -r EXAMPLE.COM
Authenticating as principal jimbo/admin@EXAMPLE.COM with password.
Password for jimbo/admin@EXAMPLE.COM:
kadmin: add_principal -randkey host/slavekdc.example.com
Principal "host/slavekdc.example.com@EXAMPLE.COM" created.
kadmin: ktadd host/slavekdc.example.com@EXAMPLE.COM
Entry for principal host/slavekdc.example.com with kvno 3, encryption type Triple DES cbc mode
```

```
with \
 HMAC/sha1 added to keytab WRFILE:/etc/krb5.keytab.
Entry for principal host/slavekdc.example.com with kvno 3, encryption type ArcFour with HMAC/md5
added \
 to keytab WRFILE:/etc/krb5.keytab.
Entry for principal host/slavekdc.example.com with kvno 3, encryption type DES with HMAC/sha1
added \
 to keytab WRFILE:/etc/krb5.keytab.
Entry for principal host/slavekdc.example.com with kvno 3, encryption type DES cbc mode with RSA-
MD5 added \
 to keytab WRFILE:/etc/krb5.keytab.
kadmin: quit
```

サービスキーを使用すると、スレーブ KDC は接続するクライアントをすべて認証できます。当然ながら、それらすべてがスレーブの kprop サービスを新しいレルムデータベースで提供できる訳ではありません。アクセスを制限するため、スレーブ KDC の kprop サービスは、プリンシパル名が /var/kerberos/krb5kdc/kpropd.acl に記載されているクライアントからの更新のみを受け入れます。マスター KDC のホストサービス名をこのファイルに追加します。

```
~]# echo host/masterkdc.example.com@EXAMPLE.COM > /var/kerberos/krb5kdc/kpropd.acl
```

スレーブ KDC がデータベースのコピーを取得したら、暗号化に使用されたマスターキーも必要です。KDC データベースのマスターキーがマスター KDC の古いファイルに保存されている場合（通常は /var/kerberos/krb5kdc/.k5.REALM という名前）、利用可能なセキュアな方法を使用してこれをスレーブ KDC にコピーするか、kdb5_util create -s を実行して、スレーブ KDC で同一の stash ファイルを作成します。

スレーブ KDC のファイアウォールにより、マスター KDC がポート 754 (krb5_prop)で TCP を使用して接続し、kprop サービスを起動できることを確認します。次に、kadmin サービスが無効になっていることを再確認します。

次に、マスター KDC でレルムデータベースをダンプし、kprop コマンドが読み取るデフォルトのデータファイルに、手動でデータベース伝搬テストを実行し (/var/kerberos/krb5kdc/slave_datatrans)、kprop コマンドを使用してその内容をスレーブ KDC に送信します。

```
~]# /usr/kerberos/sbin/kdb5_util dump /var/kerberos/krb5kdc/slave_datatrans
~]# kprop slavekdc.example.com
```

kinit を使用して、krb5.conf がレルムの KDC 一覧にあるスレーブ KDC のみを一覧表示することを確認します。これにより、スレーブ KDC から初期認証情報を正しく取得できるようになりました。

これにより、レルムデータベースをダンプし、kprop コマンドを実行してデータベースを各スレーブ KDC に送信するスクリプトを作成し、cron サービスがスクリプトを定期的に実行するように設定し

ます。

48.6.9. レルム間の認証の設定

レルム間の認証 は、1 つのレルムのクライアント（通常はユーザー）が **Kerberos** を使用してサービス（通常は特定のサーバーシステムで実行しているサーバープロセス）に対して認証を行う状況を説明するために使用される用語です。

最も単純なケースでは、**A.EXAMPLE.COM** という名前のレルムのクライアントが **B.EXAMPLE.COM** レルムのサービスにアクセスするには、両方のレルムが **krbtgt/B.EXAMPLE.COM@A.EXAMPLE.COM** という名前のプリンシパルの鍵を共有し、両方のキーに同じキーバージョン番号が関連付けられている必要があります。

これを行うには、非常に強力なパスワードまたはパスフレーズを選択し、**kadmin** を使用して両方のレルムにプリンシパルのエントリーを作成します。

```
~]# kadmin -r A.EXAMPLE.COM
kadmin: add_principal krbtgt/B.EXAMPLE.COM@A.EXAMPLE.COM
Enter password for principal "krbtgt/B.EXAMPLE.COM@A.EXAMPLE.COM":
Re-enter password for principal "krbtgt/B.EXAMPLE.COM@A.EXAMPLE.COM":
Principal "krbtgt/B.EXAMPLE.COM@A.EXAMPLE.COM" created.
kadmin: quit
~]# kadmin -r B.EXAMPLE.COM
kadmin: add_principal krbtgt/B.EXAMPLE.COM@A.EXAMPLE.COM
Enter password for principal "krbtgt/B.EXAMPLE.COM@A.EXAMPLE.COM":
Re-enter password for principal "krbtgt/B.EXAMPLE.COM@A.EXAMPLE.COM":
Principal "krbtgt/B.EXAMPLE.COM@A.EXAMPLE.COM" created.
kadmin: quit
```

get_principal コマンドを使用して、鍵バージョン番号 (**kvno** の値) と暗号化タイプの両方が一致することを確認します。

データベースのダンプが実行されない

セキュリティー意識のある管理者は、パスワードの代わりにランダムキーの割り当てに **add_principal** コマンドの **-randkey** オプションを使用し、最初のレルムのデータベースから新しいエントリーをダンプして、2 番目のレルムにインポートすることができます。データベースダンプに含まれる鍵自体がマスターキーを使用して暗号化されるため、レルムデータベースのマスターキーが同一でない限り動作しません。

A.EXAMPLE.COM レルムのクライアントは、B.EXAMPLE.COM レルムのサービスに対して認証できるようになりました。別の方法では、B.EXAMPLE.COM レルムが A.EXAMPLE.COM レルムを 信頼 するか、またはより簡単にフレーズされた B.EXAMPLE.COM が A.EXAMPLE.COM を 信頼 するようになりました。

これにより、クロスレルム信頼はデフォルトで一方向です。B.EXAMPLE.COM レルムの KDC は、B.EXAMPLE.COM レルムのサービスに対して認証するために A.EXAMPLE.COM からのクライアントを信頼する可能性がありますが、B.EXAMPLE.COM レルムのクライアントが A.EXAMPLE.COM レルムのサービスに対して認証するために信頼されているかどうかには影響しません。反対方向の信頼を確立するには、両方のレルムが krbtgt/A.EXAMPLE.COM@B.EXAMPLE.COM サービスの鍵を共有する必要があります（上記の例と比較した 2 つのレルムの順番で逆順で注意します）。

直接信頼関係がレルム間の信頼を提供する唯一の方法である場合、複数のレルムを含むネットワークはセットアップが非常に困難になります。幸い、クロスレルムの信頼は推移的です。A.EXAMPLE.COM のクライアントが B.EXAMPLE.COM のサービスに対して認証でき、B.EXAMPLE.COM のクライアントが C.EXAMPLE.COM のサービスに対して認証できる場合、C.EXAMPLE.COM のクライアントは、C.EXAMPLE.COM で直接 A.EXAMPLE.COMの サービスに対して認証することもできます。つまり、相互に信頼する必要がある複数のレルムがあるネットワークでは、セットアップする信頼関係に適切な選択を行うことで、必要な作業量を大幅に削減できる可能性があることを意味します。

従来の問題が発生しました。クライアントのシステムは、特定のサービスが属するレルムを適切に推測できるように設定する必要があり、そのレルム内のサービスの認証情報を取得する方法を判断できる必要があります。

最初：特定のレルムの特定サーバーシステムから提供されるサービスのプリンシパル名は通常、以下のようになります。

```
service/server.example.com@EXAMPLE.COM
```

この例では、サービス は通常、使用中のプロトコルの名前(**ldap imap、cvs、および HTTP***)または* **ホスト、server.example.com** *はサービスを実行するシステムの完全修飾ドメイン名で、EXAMPLE.COM はレルムの名前です。*

*サービスが属するレルムを推測するために、クライアントはほとんどの場合 /etc/krb5.conf の DNS または domain_realm セクションを参照して、ホスト名(***server.example.com***)または DNS ドメイン名(***.example.com***)をレルム名(***EXAMPLE.COM***)にマッピングします。*

サービスが属するレルムを決定するには、クライアントは問い合わせる必要のあるレルムのセットと、サービスへの認証に使用する認証情報を取得するためにそれらと通信する必要のある順番を決定す

る必要があります。

これは 2 つの方法の 1 つで実行できます。

明示的な設定を必要としないデフォルトの方法は、共有階層内でレルム名を提供することです。たとえば、A.EXAMPLE.COM、B.EXAMPLE.COM、および EXAMPLE.COM という名前のレルムを想定します。A.EXAMPLE.COM レルムのクライアントが B.EXAMPLE.COM のサービスに対して認証を試みると、デフォルトでは、最初に EXAMPLE.COM レルムの認証情報を取得しようとし、これらの認証情報を使用して B.EXAMPLE.COM レルムで使用する認証情報を取得します。

このシナリオのクライアントは、レルム名を DNS 名を処理する可能性があるものとして扱います。これは、自身のレルム名のコンポーネントを繰り返し取り除き、階層内で上のレリムの名前を生成します。これは、サーバーレリムの上でもある位置に達するまで行われます。この時点で、サービスのレルムに到達するまで、サービスのレルム名のコンポーネントを先頭に追加し始めます。プロセスに関与する各レルムは別のホップです。

たとえば、A.EXAMPLE.COM で認証情報を使用し、B.EXAMPLE.COM のサービスに対して認証します。

A.EXAMPLE.COM → EXAMPLE.COM → B.EXAMPLE.COM

- *A.EXAMPLE.COM および EXAMPLE.COM が krbtgt/EXAMPLE.COM@A.EXAMPLE.COMの鍵を共有*
- *EXAMPLE.COM および B.EXAMPLE.COM が krbtgt/B.EXAMPLE.COM@EXAMPLE.COMの鍵を共有*

もう 1 つの例は、SITE1.SALES.EXAMPLE.COM で認証情報を使用して EVERYWHERE.EXAMPLE.COM のサービスに対して認証を行います。

SITE1.SALES.EXAMPLE.COM → SALES.EXAMPLE.COM → EXAMPLE.COM → EVERYWHERE.EXAMPLE.COM

- *SITE1.SALES.EXAMPLE.COM と SALES.EXAMPLE.COM が krbtgt/SALES.EXAMPLE.COM@SITE1.SALES.EXAMPLE.COMの鍵を共有*
- *SALES.EXAMPLE.COM と EXAMPLE.COM が krbtgt/EXAMPLE.COM@SALES.EXAMPLE.COMの鍵を共有*
- *EXAMPLE.COM と EVERYWHERE.EXAMPLE.COM が krbtgt/EVERYWHERE.EXAMPLE.COM@EXAMPLE.COMの鍵を共有*

別の例として、その名前が共通の接尾辞(DEVEL.EXAMPLE.COM および PROD.EXAMPLE.ORG)を共有しているレルム名を使用します。

DEVEL.EXAMPLE.COM → EXAMPLE.COM → COM → ORG → EXAMPLE.ORG → PROD.EXAMPLE.ORG

- *DEVEL.EXAMPLE.COM と EXAMPLE.COM が krbtgt/EXAMPLE.COM@DEVEL.EXAMPLE.COMの鍵を共有*
- *EXAMPLE.COM と COM が krbtgt/COM@EXAMPLE.COMの鍵を共有*
- *COM および ORG が krbtgt/ORG@COMの鍵を共有*
- *ORG と EXAMPLE.ORG が krbtgt/EXAMPLE.ORG@ORGを共有*
- *EXAMPLE.ORG および PROD.EXAMPLE.ORG が krbtgt/PROD.EXAMPLE.ORG@EXAMPLE.ORGの鍵を共有*

より複雑で柔軟な方法は、/etc/krb5.conf の capaths セクションを設定することです。これにより、1 つのレルムのクレデンシャルを持つクライアントは、チェーン内で次にあるレルムを検索し、最終的にサーバーに認証できるようになります。

capaths セクションの形式は比較的簡単です。セクションの各エントリーは、クライアントが存在する可能性があるレルムの後に名前が付けられます。そのサブセクション内では、クライアントが認証情報を取得する必要のある中間レルムのセットが、サービスが置かれるレルムに対応するキーの値として一覧表示されます。中間レルムがない場合は、.が使用されます。

以下は例になります。

```
[capaths]
A.EXAMPLE.COM = {
 B.EXAMPLE.COM = .
 C.EXAMPLE.COM = B.EXAMPLE.COM
 D.EXAMPLE.COM = B.EXAMPLE.COM
 D.EXAMPLE.COM = C.EXAMPLE.COM
}
```

この例では、 A.EXAMPLE.COM レルムのクライアントは、A.EXAMPLE.COM KDC から直接 B.EXAMPLE.COM のレルム間の認証情報を取得できます。

これらのクライアントがC.EXAMPLE.COM レルムでサービスに問い合わせたい場合は、最初に B.EXAMPLE.COM レルムから必要な認証情報を取得する必要があります（これには krbtgt /B.EXAMPLE.COM@A.EXAMPLE.COM が必要です）。次に、これら の認証情報を使用して C.EXAMPLE.COM レルムで使用する認証情報を取得する必要があります(krbtgt/C.EXAMPLE.COM@B.EXAMPLE.COMを使用)。

これらのクライアントが D.EXAMPLE.COM レルムのサービスに問い合わせたい場合は、最初に B.EXAMPLE.COM レルムから必要な認証情報を取得し、次に C.EXAMPLE.COM レルムから認証情報を取得して D.EXAMPLE.COM レルムで使用する認証情報を取得する必要があります。

注記

特に示される capath エントリーがない場合、Kerberos はレルム間の信頼関係が階層を形成していると仮定します。

A.EXAMPLE.COM レルムのクライアントは、B.EXAMPLE.COM レルムから直接レルム間の認証情報を取得できます。.を指定しないと、クライアントは代わりに階層パスを使用しようとします。この場合は以下のようになります。

A.EXAMPLE.COM → EXAMPLE.COM → B.EXAMPLE.COM

48.6.10. 関連情報

Kerberos の詳細は、以下のリソースを参照してください。

48.6.10.1. インストールされているドキュメント

- *『Kerberos V5 インストールガイド』 および PostScript および HTML 形式 『の Kerberos V5 システム管理者のガイド』。これらは、/usr/share/doc/krb5-server- <version-number> / ディレクトリー(< version-number > は、システムにインストールされている krb5-server パッケージのバージョン番号)にあります。*

- *PostScript および HTML 形式の 『Kerberos V5 UNIX ユーザーガイド』これらは、/usr/share/doc/krb5-workstation- <version-number> / ディレクトリー(< version-number > は、システムにインストールされている krb5-workstation パッケージのバージョン番号)にあります。*

- *Kerberos の man ページ : Kerberos 実装に関連するさまざまなアプリケーションおよび設定ファイルに関する man ページが多数あります。以下は、より重要な man ページの一部の一覧です。*

  クライアントアプリケーション

  - *man kerberos - Kerberos システムの紹介で、認証情報の仕組みを説明し、Kerberos チケットの取得および破棄に関する推奨事項を提供します。man ページの下部では、関連する man ページが多数参照されています。*

  - *man kinit: このコマンドを使用してチケット保証チケットを取得し、キャッシュする方法が説明されています。*

- *man kdestroy: このコマンドを使用して Kerberos 認証情報を破棄する方法が説明されています。*

- *man klist - このコマンドを使用して、キャッシュされた Kerberos 認証情報を一覧表示する方法が説明されています。*

管理アプリケーション

- *man kadmin - このコマンドを使用して Kerberos V5 データベースを管理する方法が説明されています。*

- *man kdb5_util - このコマンドを使用して Kerberos V5 データベース上で低レベルの管理機能を作成して実行する方法を説明します。*

サーバーアプリケーション

- *man krb5kdc: Kerberos V5 KDC で利用可能なコマンドラインオプションを説明しています。*

- *man kadmind: Kerberos V5 管理サーバーで利用可能なコマンドラインオプションを説明しています。*

設定ファイル

- *man krb5.conf - Kerberos V5 ライブラリーの設定ファイル内で使用できる形式とオプションを説明しています。*

- *man kdc.conf: Kerberos V5 AS および KDC の設定ファイル内で利用可能な形式およびオプションを説明しています。*

48.6.10.2. 便利な Web サイト

-

http://web.mit.edu/kerberos/www/ - 『Kerberos: MIT のネットワーク認証プロトコル (Network Authentication Protocol』)の Web ページ

- http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html - Kerberos Frequently Asked Questions (FAQ)

- ftp://athena-dist.mit.edu/pub/kerberos/doc/usenix.PS - PostScript バージョンの『Kerberos: Jennifer G. Steiner による Open Network Systems の認証サービス』。Steiner、Clifford Neuman、Jeffrey I. Schiller。この文書は、Kerberos を説明するオリジナルの資料です。

- http://web.mit.edu/kerberos/www/dialogue.html - 『Designing an Authentication System: a Dialogue in Four Scenes』 from Bill Bryant in 1988, modified by Theodore Ts'o in 1997.本書は、Kerberos スタイルの認証システムの作成を考慮している 2 人の開発者の会話を表しています。会話スタイルは、Kerberos に精通していないユーザーにとって適切な開始場所となります。

- http://www.ornl.gov/~jar/HowToKerb.html - 『Kerberize your site』 is a good reference for kerberizing a network.

- http://www.networkcomputing.com/netdesign/kerb1.html: 『Kerberos ネットワーク設計マニュアル』 は、Kerberos システムの詳細な概要です。

48.7. 仮想プライベートネットワーク(VPN)

多くの場合、Satellite オフィスが複数ある組織は、転送中の機密データの効率や保護のために、専用ラインで相互に接続することがよくあります。たとえば、多くのビジネスでは、フレームリレーまたは 非同期転送モード (ATM)行をエンドツーエンドネットワークソリューションとして使用し、1 つのオフィスを他のオフィスにリンクします。これは特に、エンタープライズレベルの専用デジタルサーキットに関連する高費用を支払うことなく拡張したい小規模中規模の企業(SMB)の場合は、高価な価値があります。

このニーズに対応するために、仮想プライベートネットワーク (VPNs)が開発されました。専用のサーキットと同じ機能原則に従い、VPNは 2 人の当事者（またはネットワーク）間の安全なデジタル通信を可能にし、既存の ローカルエリアネットワーク(LAN)からWAN(Wide Area Network)を作成できます。フレームリレーまたは ATM の相違点はトランスポートメディアにあります。VPNは、データグラムをトランスポート層として使用して IP 経由で送信し、インターネットを介したセキュアなコンジットを目的の宛先に送信します。ほとんどのフリーソフトウェア VPN 実装には、転送中のデータをさらにマスクするためのオープン標準暗号化方法が組み込まれています。

一部の組織では、ハードウェア VPN ソリューションを使用してセキュリティーを強化し、ソフトウェアやプロトコルベースの実装を使用するものもあります。いくつかのベンダーは、Cisco、Nortel、IBM、Checkpoint などのハードウェア VPN ソリューションを提供します。FreeS/Wan と呼ばれる Linux 用の無料ソフトウェアベースの VPN ソリューションがあり、標準化された Internet Protocol Security (IPsec)実装を利用します。これらの VPN ソリューションは、ハードウェアまたはソフトウェアベースであるかに関係なく、あるオフィスから別のオフィスへの IP 接続間に存在する特殊なルーターとして機能します。

48.7.1. VPN の仕組み

クライアントからパケットが送信されると、VPN ルーターまたはゲートウェイを介して送信され、ルーティングおよび認証に 認証ヘッダー (AH)が追加されます。その後、データは暗号化され、最後に Encapsulating Security Payload (ESP)で囲まれます。後者は、復号化および処理の命令を設定します。

受信側の VPN ルーターはヘッダー情報を取り除き、データの復号化を行い、それを目的の宛先（ネットワーク上のワークステーションまたは他のノード）にルーティングします。ネットワーク対ネットワークの接続を使用すると、ローカルネットワーク上の受信側ノードはすでに暗号化解読されいてすぐに処理ができる状態のパケットを受信します。ネットワーク/ネットワーク間 VPN 接続の暗号化/復号化プロセスは、ローカルノードに透過的です。

このような高さのセキュリティーレベルにより、攻撃者はパケットを傍受するだけでなく、パケットも復号化する必要があります。サーバーとクライアント間で中間者攻撃を使用する侵入者は、セッションを認証するために少なくとも 1 つの秘密鍵にアクセスできる必要があります。VPNは、認証と暗号化の複数のレイヤーを使用するため、複数のリモートノードを接続して統一されたイントラネットとして機能する安全かつ効果的な方法です。

48.7.2. VPN および Red Hat Enterprise Linux

Red Hat Enterprise Linux は、 WAN に安全に接続するためのソフトウェアソリューションの実装に関して、さまざまなオプションを提供します。インターネットプロトコルセキュリティー (IPsec) は、Red Hat Enterprise Linux でサポートされている VPN 実装であり、ブランチオフィスやリモートユーザーを使用する組織のユーザービリティーニーズを十分に対応しています。

48.7.3. IPsec

Red Hat Enterprise Linux は IPsec をサポートし、インターネットなどの共通の通信ネットワーク上のセキュアなトンネルを使用してリモートホストとネットワークを相互に接続します。IPSec は、ホスト間（コンピューター間ワークステーションに 1 台）またはネットワーク間(LAN/WAN を 1 つ)設定を使用して実装できます。

Red Hat Enterprise Linux の IPsec 実装は、Internet Engineering Task Force (IETF)によって実装されたプロトコルである Internet Key Exchange (IKE)を使用します。これは、相互認証および接続

システム間の安全な関連付けに使用されます。

48.7.4. IPsec 接続の作成

IPsec 接続が 2 つの論理フェーズに分割されます。フェーズ 1 では、IPsec ノードはリモートノードまたはネットワークとの接続を初期化します。リモートノードまたはネットワークは、要求しているノードの認証情報をチェックし、両者が接続の認証方法をネゴシエートします。

Red Hat Enterprise Linux システムでは、IPsec 接続は IPsec ノード認証の 事前共有鍵 メソッドを使用します。共有前のキー IPsec 接続では、IPsec 接続の フェーズ 2 に移行するために、両方のホストが同じ鍵を使用する必要があります。

IPsec 接続のフェーズ 2 では、IPsec ノード間で セキュリティーアソシエ ーション(SA)が作成されます。このフェーズは、暗号化メソッド、シークレットセッションキー交換パラメーターなどの設定情報を使用して SA データベースを確立します。このフェーズは、リモートノードとネットワーク間の実際の IPsec 接続を管理します。

IPsec の Red Hat Enterprise Linux の実装では、IKE を使用してインターネット内のホスト間で鍵を共有します。racoon のキー設定デーモンは、IKE 鍵の分散と交換を処理します。このデーモンの詳細は、racoon の man ページを参照してください。

48.7.5. IPsec のインストール

IPsec を実装するには、ipsec-tools RPM パッケージがすべての IPsec ホスト（ホスト間設定を使用している場合）またはルーター（ネットワーク間設定を使用する場合）にインストールする必要があります。RPM パッケージには、IPsec 接続を設定するための必須ライブラリー、デーモン、および設定ファイルが含まれます。

- /sbin/setkey - カーネルの IPsec のキー管理およびセキュリティー属性を操作します。この実行可能ファイルは、racoon キー管理デーモンによって制御されます。詳細は、setkey(8) man ページを参照してください。

- /usr/sbin/racoon - IPsec に接続されたシステム間のセキュリティー関連付けと鍵共有を管理および制御するために使用される IKE 鍵管理デーモン。

- /etc/racoon/racoon.conf: 接続で使用される認証方法や暗号化アルゴリズムなど、IPsec 接続のさまざまな側面の設定に使用される racoon デーモン設定ファイル。利用可能なディレクティブの完全なリストは、racoon.conf(5) man ページを参照してください。

Red Hat Enterprise Linux で IPsec を設定するには、Network Administration Tool を使用するか、ネットワークおよび IPsec 設定ファイルを手動で編集します。

- IPsec 経由でネットワーク接続された 2 つのホストを接続するには、「IPsec Host-to-Host の設定」 を参照してください。

- IPsec 経由で 1 つの LAN/WAN を接続するには、「IPsec Network-to-Network の設定」を参照してください。

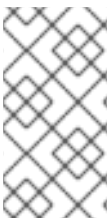
48.7.6. IPsec Host-to-Host の設定

IPSec は、ホスト間接続を使用して 1 つのデスクトップまたはワークステーション（ホスト）を別の接続するように設定できます。このタイプの接続は、各ホストが接続されているネットワークを使用して、各ホストにセキュアなトンネルを作成します。ホスト間の接続の要件は、各ホストでの IPsec の設定であるため、最小限に抑えられます。ホストには、通信ネットワーク（インターネットなど）および Red Hat Enterprise Linux が IPsec 接続を作成するために専用の接続のみが必要です。

48.7.6.1. ホスト間接続

ホスト間の IPsec 接続は、2 つのシステム間で暗号化された接続で、いずれも同じ認証キーで IPsec を実行します。IPsec 接続を有効にすると、2 つのホスト間のネットワークトラフィックはすべて暗号化されます。

ホスト間の IPsec 接続を設定するには、各ホストに次の手順を使用します。



注記

設定する実際のマシンで以下の手順を実行します。IPsec 接続をリモートで設定および確立しないようにします。

1. コマンドシェルで、system-config-network と入力して Network Administration Tool を起動します。

2. IPsec タブで、New をクリックして IPsec 設定ウィザードを起動します。

3.
Forward をクリックして、ホスト間の IPsec 接続の設定を開始します。

4.
接続の一意の名前を入力します（例： ipsec0必要に応じて、チェックボックスを選択して、コンピューターの起動時に接続を自動的にアクティブにします。進む をクリックして続けます。

5.
接続タイプとして Host to Host encryption を選択し、Forward をクリックします。

6.
使用する暗号化のタイプ(manual または automatic)を選択します。

手動暗号化を選択する場合は、暗号鍵をプロセスの後で提供する必要があります。自動暗号化を選択すると、racoon デーモンが暗号化キーを管理します。自動暗号化を使用する場合は、ipsec-tools パッケージをインストールする必要があります。

進む をクリックして続けます。

7.
リモートホストの IP アドレスを入力します。

リモートホストの IP アドレスを確認するには、リモートホスト で以下のコマンドを 使用します。

```
ifconfig <device>
```

& lt;device > は、VPN 接続に使用するイーサネットデバイスに置き換えます。

システムにイーサネットカードが 1 つしか存在しない場合、デバイス名は通常 eth0 になります。以下の例は、このコマンドに関連する情報を示しています（出力のみであることに注意してください）。

```
eth0      Link encap:Ethernet  HWaddr 00:0C:6E:E8:98:1D
          inet addr:172.16.44.192  Bcast:172.16.45.255  Mask:255.255.254.0
```

IP アドレスは、inet addr: ラベルの後の番号です。

注記

ホスト間接続の場合、両方のホストにパブリックかつルーティング可能なアドレスが必要です。また、両方のホストに、sam LAN 上にいる限り、プライベートでルーティング不可能なアドレス(10.x.x.x または 192.168.x.x 範囲から)を指定できます。

ホストが異なる LAN 上にある場合、または別のホストにプライベートアドレスがある間にパブリックアドレスがある場合は、「IPsec Network-to-Network の設定」 を参照してください。

進む をクリックして続けます。

8. *手順 6 で手動暗号化を選択した場合は、使用する暗号化キーを指定するか、Generate をクリックして作成します。*

   a. *認証キーを指定するか、Generate をクリックして生成します。数字と文字の組み合わせを任意に指定できます。*

   b. *進む をクリックして続けます。*

9. *IPsec - Summary ページの情報を確認してから Apply をクリックします。*

10. *File > Save をクリックして設定を保存します。*

変更を有効にするには、ネットワークを再起動する必要がある場合があります。ネットワークを再起動するには、以下のコマンドを使用します。

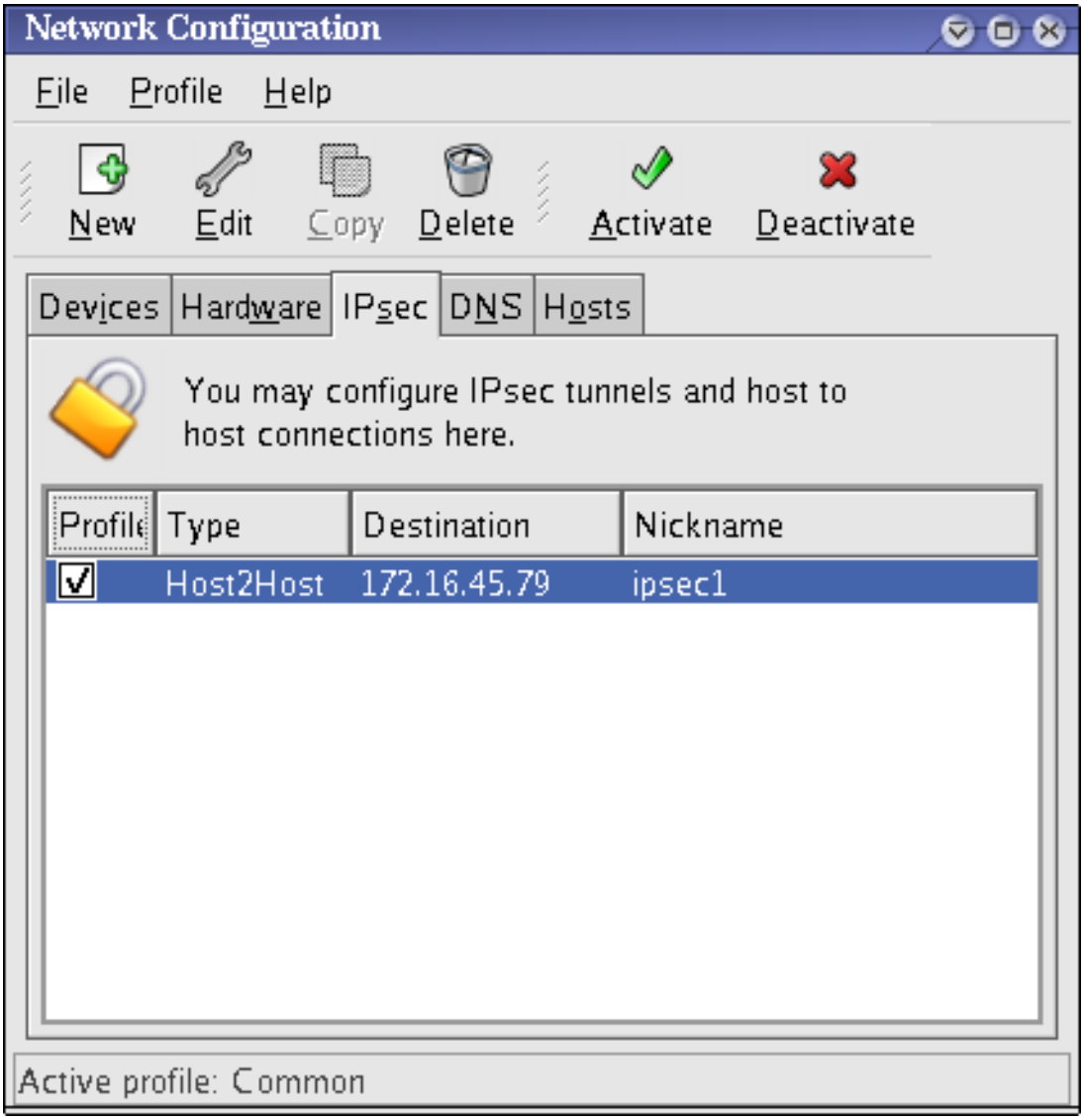
```
service network restart
```

11. *一覧から IPsec 接続を選択し、Activate ボタンをクリックします。*

12. *他のホストに対して、手順全体を繰り返します。手順 8 の同じキーを他のホストで使用することが重要です。そうしないと、IPsec は機能しません。*

IPsec 接続の設定後、図48.10「IPsec 接続」 に示されるように、IPsec リストに表示されます。

図48.10 IPsec 接続



[D]

以下のファイルは、IPsec 接続の設定時に作成されます。

- */etc/sysconfig/network-scripts/ifcfg-<nickname>*
- */etc/sysconfig/network-scripts/keys-<nickname>*
- */etc/racoon/<remote-ip>.conf*

- */etc/racoon/psk.txt*

自動暗号化を選択すると、/etc/racoon/racoon.conf も作成されます。

インターフェイスが起動すると、/etc/racoon/racoon.conf が変更され、< remote-ip>.conf が含まれるようになります。

48.7.6.2. 手動 IPsec Host-to-Host 設定

接続作成の最初の手順は、各ワークステーションからシステムおよびネットワーク情報を収集することです。ホスト間の接続には、以下が必要です。

- *各ホストの IP アドレス*
- *一意の名前（例： ipsec1 ）。これは、IPsec 接続を特定し、他のデバイスまたは接続と区別するために使用されます。*
- *固定暗号化キーまたは racoon によって自動的に生成される暗号鍵。*
- *接続の初期段階で使用され、セッション中に暗号鍵を交換するために使用される事前共有認証キー。*

たとえば、Workstation A および Workstation B が IPsec トンネルを介して相互に接続するとします。ユーザーは Key_Value01 の値で共有前のキーを使用して接続し、ユーザーは各ホスト間で認証キーを自動的に生成して共有することに同意します。両方のホストユーザーは、接続 ipsec1 に名前を付けます。

注記

大文字、小文字、数字、および句読点の組み合わせを使用する PSK を選択する必要があります。簡単に保証できる PSK はセキュリティーリスクを設定します。

各ホストに同じ接続名を使用する必要はありません。インストールに便利で意味のある名前を選択する必要があります。

以下は、Workstation B を使用したホスト間の IPsec 接続の Workstation A の IPsec 設定ファイルです。この例では接続を識別する一意の名前は ipsec1 であるため、作成されるファイルは /etc/sysconfig/network-scripts/ifcfg-ipsec1 と呼ばれます。

```
DST=X.X.X.X
TYPE=IPSEC
ONBOOT=no
IKE_METHOD=PSK
```

Workstation A の場合、X.X.X.X.X は Workstation B の IP アドレスで、X. X.X.X.X は Workstation A の IP アドレスです。この接続は起動時に開始するように設定されず(ONBOOT=no)、事前共有鍵メソッド(IKE_METHOD=PSK)を使用します。

以下は、両方のワークステーションが相互に認証するために必要な、事前共有キーファイル(/etc/sysconfig/network-scripts/keys-ipsec1と呼ばれる)の内容です。このファイルのコンテンツは両方のワークステーションで同一で、root ユーザーのみがこのファイルの読み取りや書き込みが可能でなければなりません。

```
IKE_PSK=Key_Value01
```

重要な影響

root ユーザーのみがファイルの読み取りまたは編集ができるように keys-ipsec1 ファイルを変更するには、ファイルの作成後に以下のコマンドを使用します。

```
chmod 600 /etc/sysconfig/network-scripts/keys-ipsec1
```

いつでも認証キーを変更するには、両方のワークステーションの keys-ipsec1 ファイルを編集します。適切な接続の場合には、両方の認証キーが同一でなければなりません。

以下の例は、リモートホストへのフェーズ 1 接続の特定の設定を示しています。このファイルはX.X.X.X.conf と呼ばれます。X.X.X.X は、リモート IPsec ホストの IP アドレスです。このファイルは、IPsec トンネルがアクティベートされると自動的に生成されるため、直接編集しないでください。

```
remote X.X.X.X
{
        exchange_mode aggressive, main;
  my_identifier address;
  proposal {
   encryption_algorithm 3des;
  hash_algorithm sha1;
  authentication_method pre_shared_key;
```

```
  dh_group 2 ;
 }
}
```

IPsec 接続の初期化時に作成されるデフォルトのフェーズ 1 設定ファイルには、IPsec の Red Hat Enterprise Linux 実装で使用される以下のステートメントが含まれています。

remote X.X.X.X

この設定ファイルの後続のスタンザは、X.X.X.X IP アドレスで識別されるリモートノードにのみ適用されることを指定します。

exchange_mode aggressive

Red Hat Enterprise Linux における IPsec のデフォルト設定は、アグレッシブ認証モードを使用します。これにより、複数のホストとの複数の IPsec 接続の設定を許可し、接続のオーバーヘッドが低減します。

my_identifier アドレス

ノードの認証時に使用する識別方法を指定します。Red Hat Enterprise Linux は IP アドレスを使用してノードを識別します。

encryption_algorithm 3des

認証中に使用される暗号化暗号を指定します。デフォルトでは、Triple Data Encryption Standard (3DES)が使用されます。

hash_algorithm sha1;

ノード間のフェーズ 1 ネゴシエーション中に使用されるハッシュアルゴリズムを指定します。デフォルトでは、Secure Hash Algorithm バージョン 1 が使用されます。

authentication_method pre_shared_key

ノードのネゴシエーション中に使用される認証方法を指定します。デフォルトでは、Red Hat Enterprise Linux は認証に事前共有キーを使用します。

dh_group 2

動的に生成されるセッションキーを確立するための Diffie-Hellman グループ番号を指定します。デフォルトでは、modp1024 （グループ 2）が使用されます。

48.7.6.2.1. Racoon 設定ファイル

/etc/racoon/racoon.conf ファイルは、/ etc/racoon/X.X.X.X.conf ステートメントを除き、すべての IPsec ノードで同一でなければ なりません。このステートメント（および参照するファイル）は、IPsec トンネルがアクティブになると生成されます。Workstation A の場合、include ステートメントの X.X.X.X は Workstation B の IP アドレスです。ワークステーション B とは反対になります。以下は、IPsec 接続がアクティベートされた場合の一般的な racoon.conf ファイルを示しています。

```
# Racoon IKE daemon configuration file.
# See 'man racoon.conf' for a description of the format and entries.

path include "/etc/racoon";
path pre_shared_key "/etc/racoon/psk.txt";
path certificate "/etc/racoon/certs";

sainfo anonymous
{
        pfs_group 2;
        lifetime time 1 hour ;
        encryption_algorithm 3des, blowfish 448, rijndael ;
        authentication_algorithm hmac_sha1, hmac_md5 ;
        compression_algorithm deflate ;
}
include "/etc/racoon/X.X.X.X.conf";
```

このデフォルトの racoon.conf ファイルには、IPsec 設定、事前共有キーファイル、および証明書の定義されたパスが含まれます。sainfo anonymous のフィールドは、IPsec 接続の性質（使用される暗号化アルゴリズムを含む）と鍵交換方法である IPsec ノード間のフェーズ 2 SA を記述します。以下のリストは、フェーズ 2 のフィールドを定義します。

sainfo anonymous

IPsec 認証情報が一致する場合、SA が任意のピアで匿名で初期化できることを示します。

pfs_group 2

Diffie-Hellman 鍵交換プロトコルを定義します。これは、IPsec ノードが IPsec 接続の 2 番目のフェーズで相互一時セッションキーを確立する方法を決定します。デフォルトでは、IPsec の Red Hat Enterprise Linux 実装は、Diffie-Hellman 暗号化鍵交換グループのグループ 2 （または

modp1024）を使用します。Group 2 は 1024 ビットのモジュール指数を使用し、秘密鍵が侵害された場合でも攻撃者が以前の IPsec 送信を復号化できないようにします。

ライフタイム 1 時間

このパラメーターは SA の有効期間を指定し、データの時間またはバイト単位で定量化できます。IPsec のデフォルトの Red Hat Enterprise Linux 実装は、1 時間の長さを指定します。

encryption_algorithm 3des、blowfish 448、rijndael

フェーズ 2 でサポートされる暗号を指定します。Red Hat Enterprise Linux は、3DES、448 ビット Blowfish、および Rijndael (Advanced Encryption Standard または AESで使用される暗号)をサポートします。

authentication_algorithm hmac_sha1、hmac_md5

は、認証でサポートされるハッシュアルゴリズムを一覧表示します。サポートされるモードは sha1 および md5 ハッシュされたメッセージ認証コード(HMAC)です。

compression_algorithm deflate

IP Payload Compression (IPCOMP)サポート用の Deflate 圧縮アルゴリズムを定義します。これにより、低速な接続で IP データグラムの送信を高速化できます。

接続を開始するには、各ホストで以下のコマンドを使用します。

```
ifup <nickname>
```

<nickname> は、IPsec 接続に指定した名前に置き換えます。

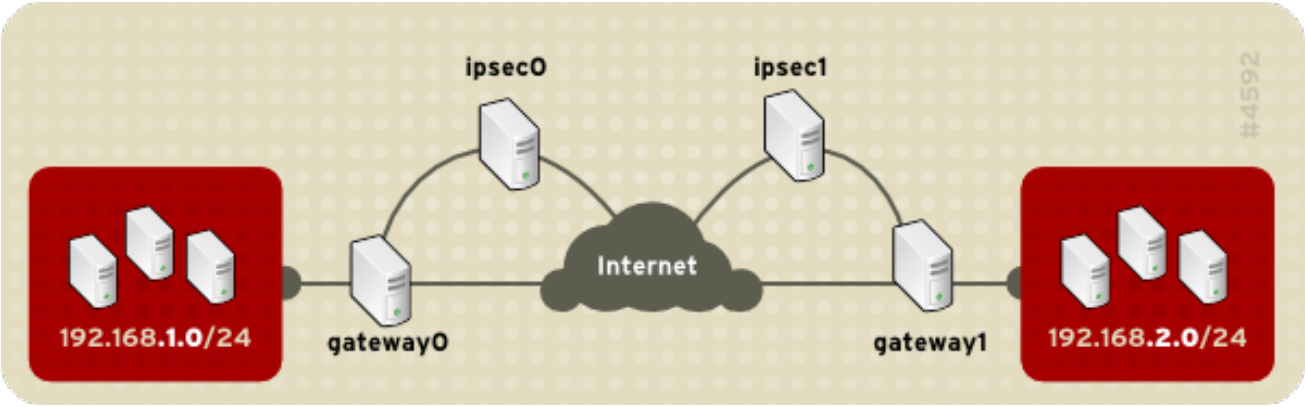
IPsec 接続をテストするには、tcpdump ユーティリティーを実行し、ホスト間で転送されているネットワークパケットを表示し、IPsec で暗号化されていることを確認します。パケットには AH ヘッダーが含まれる必要があり、ESP パケットとして表示されるはずです。ESP は暗号化されることを意味します。以下に例を示します。

```
~]# tcpdump -n -i eth0 host <targetSystem>

IP 172.16.45.107 > 172.16.44.192: AH(spi=0x0954ccb6,seq=0xbb): ESP(spi=0x0c9f2164,seq=0xbb)
```

48.7.7. IPsec Network-to-Network の設定

また、IPSec は、ネットワーク接続を使用してネットワーク全体(LAN、WANなど)をリモートネットワークに接続するように設定することもできます。ネットワーク間接続では、LAN 上の 1 つのノードからリモート LAN 上のノードに情報を透過的に処理し、ルーティングするために、接続ネットワークの各ノードで IPsec ルーターを設定する必要があります。図48.11「ネットワーク間 IPsec トンネル接続」 ネットワーク間 IPsec トンネリング接続を表示します。

図48.11 ネットワーク間 IPsec トンネル接続



[D]

この図では、インターネットで区切られた 2 つの別個の LANを示しています。これらの LANは IPsec ルーターを使用してインターネット経由でセキュアなトンネルを使用して接続を認証および開始します。転送で傍受されたパケットは、これらの LAN間のパケットを保護するためにブルートフォースの復号化が必要になります。IP 範囲 192.168.1.0/24 の 1 つから IP 範囲内の別のノードとの通信のプロセスは、IP 範囲の処理、暗号化/復号化、および IPsec パケットのルーティングが IPsec ルーターによって完全に処理されるので、ノードに対して完全に透過的です。

ネットワーク間接続に必要な情報は次のとおりです。

- 専用 IPsec ルーターの外部からアクセスできる IP アドレス
- IPsec ルーターが提供する LAN/WAN のネットワークアドレス範囲(192.168.1.0/24 または 10.0.1.0/24 など)
- ネットワークノードからインターネットにデータをルーティングするゲートウェイデバイスの IP アドレス

- 一意の名前（例： ipsec1 ）。これは、IPsec 接続を特定し、他のデバイスまたは接続と区別するために使用されます。

- 固定暗号化キーまたは racoonによって自動的に生成される暗号鍵

- 接続の初期段階で使用され、セッション中に暗号鍵を交換するために使用される事前共有認証キー。

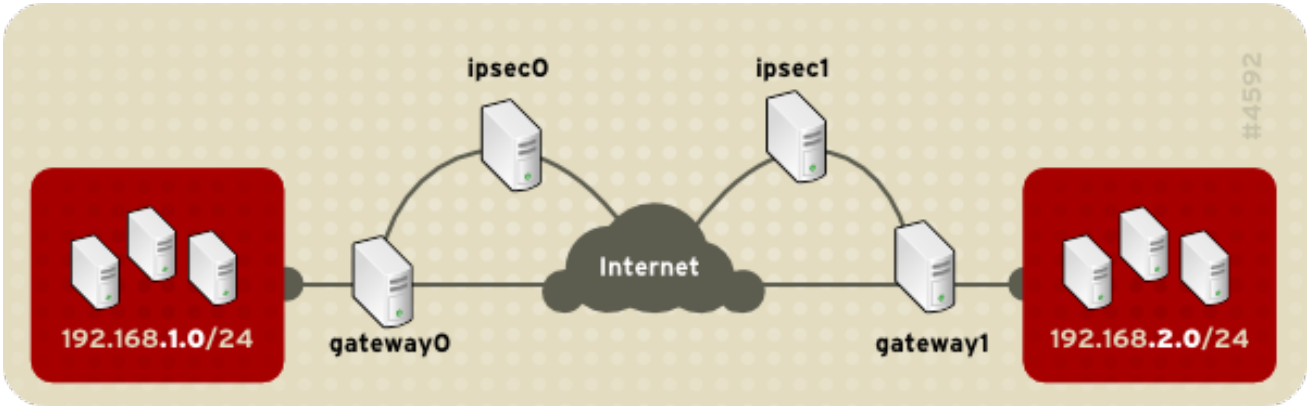
48.7.7.1. ネットワーク/ネットワーク(VPN)接続

ネットワーク/ネットワーク IPsec 接続は、各ネットワークに 1 つずつ、プライベートサブネットのネットワークトラフィックがルーティングされる 2 つの IPsec ルーターを使用します。

たとえば、図48.12「Network-to-Network IPsec」 で示すように、192.168.1.0/24 プライベートネットワークがネットワークのトラフィックを private ネットワークに送信すると、パケットは gateway0 を、インターネットを介して ipsec0 に、ipsec1 から gateway1 に、およびこのサブネットに ipsec1 を通過します。

IPSec ルーターには、一般にアドレス可能な IP アドレスと、対応するプライベートネットワークに接続された 2 つ目のイーサネットデバイスが必要です。トラフィックは、暗号化された接続がある別の IPsec ルーターを対象としている場合にのみ IPsec ルーターを通過します。

図48.12 Network-to-Network IPsec



[D]

別のネットワーク設定オプションには、各 IP ルーターとインターネット間のファイアウォール、各 IPsec ルーターとサブネットゲートウェイ間のイントラネットファイアウォールが含まれます。IPsec ルーターとサブネットのゲートウェイは、2 つのイーサネットデバイスを持つ 1 つのシステムにすることができます。1 つは IPsec ルーターとして動作するパブリック IP アドレスを持つものと、プライベートサブネットのゲートウェイとして機能するプライベート IP アドレスを持つシステム

です。各 IPsec ルーターは、プライベートネットワークまたはパブリックゲートウェイのゲートウェイを使用して、パケットを他の IPsec ルーターに送信できます。

ネットワーク/ネットワーク間の IPsec 接続を設定するには、以下の手順を使用します。

1. *コマンドシェルで、system-config-network と入力して Network Administration Tool を起動します。*

2. *IPsec タブで、New をクリックして IPsec 設定ウィザードを起動します。*

3. *Forward をクリックして、ネットワーク間 IPsec 接続の設定を開始します。*

4. *接続の一意のニックネームを入力します（例： ipsec0必要に応じて、チェックボックスを選択して、コンピューターの起動時に接続を自動的にアクティブにします。進む をクリックして続けます。*

5. *接続タイプとして Network to Network encryption (VPN) を選択し、Forward をクリックします。*

6. *使用する暗号化のタイプ(manual または automatic)を選択します。*

   手動暗号化を選択する場合は、暗号鍵をプロセスの後で提供する必要があります。自動暗号化を選択すると、racoon デーモンが暗号化キーを管理します。自動暗号化を使用する場合は、ipsec-tools パッケージをインストールする必要があります。

   進む をクリックして続けます。

7. *Local Network ページで、以下の情報を入力します。*

   - *ローカルネットワークアドレス: プライベートネットワークに接続された IPsec ルーター上のデバイスの IP アドレス。*

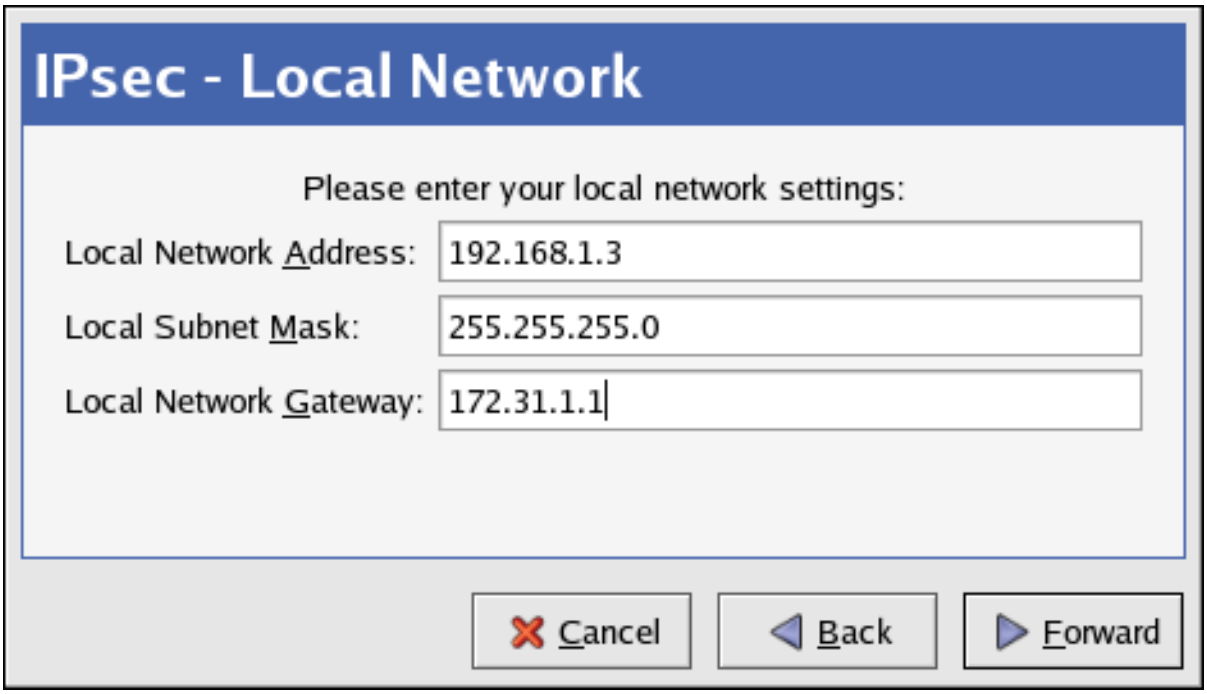
   - *ローカルサブネットマスク - ローカルネットワーク IP アドレスのサブネットマス*

ク。

- *Local Network Gateway - プライベートサブネットのゲートウェイ。*

進む をクリックして続けます。

図48.13 ローカルネットワーク情報



[D]

8. *Remote Network ページで、以下の情報を入力します。*

- *リモート IP アドレス：他 のプライベートネットワーク用に IPsec ルーターの一般にアドレス指定可能な IP アドレスです。この例では、ipsec0 の場合は ipsec1 の公開されている IP アドレスを入力します。その逆も同様です。*

- *リモートネットワークアドレス: 他 の IPsec ルーターの背後にあるプライベートサブネットのネットワークアドレス。この例では、ipsec1 を設定する場合は 192.168.1.0 を入力し、ipsec0 を設定する場合は 192.168.2.0 と入力します。*

- *リモートサブネットマスク - リモート IP アドレスのサブネットマスク。*

- *Remote Network Gateway - リモートネットワークアドレスのゲートウェイの IP ア*

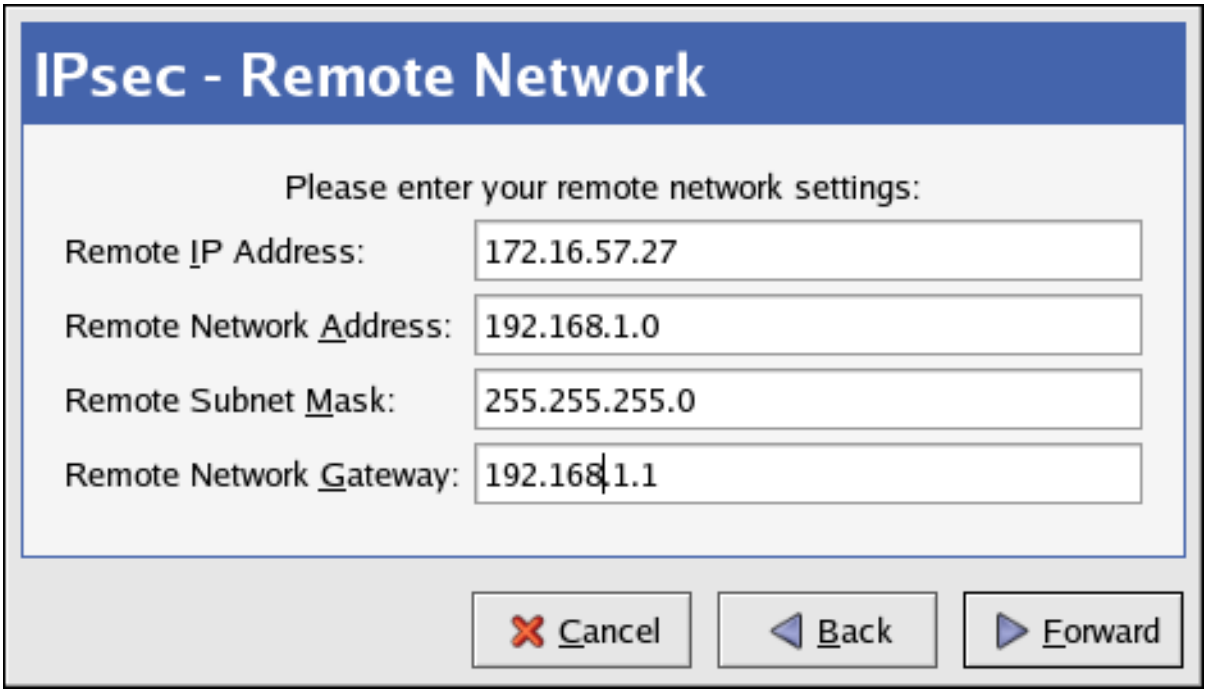
ドレス。

- 手順 6 で手動暗号化を選択した場合は、使用する暗号化キーを指定するか、Generate をクリックして作成します。

  認証キーを指定するか、Generate をクリックして生成します。このキーは、数字と文字の組み合わせを任意に指定できます。

進む をクリックして続けます。

図48.14 リモートネットワーク情報



[D]

9. IPsec - Summary ページの情報を確認してから Apply をクリックします。

10. File > Save を選択して設定を保存します。

11. 一覧から IPsec 接続を選択し、Activate をクリックして接続をアクティブにします。

12. IP 転送を有効にします。

a.
/etc/sysctl.conf を編集し、net.ipv4.ip_forward を 1 に設定します。

b.
以下のコマンドを使用して変更を適用します。

```
sysctl -p /etc/sysctl.conf
```

IPsec 接続をアクティブにするネットワークスクリプトは、必要に応じて IPsec ルーターを介してパケットを送信するためのネットワークルートを自動的に作成します。

48.7.7.2. 手動 IPsec Network-to-Network 設定

LAN A (lana.example.com)と LAN B (lanb.example.com)が IPsec トンネルを介して相互に接続したいとします。LAN A のネットワークアドレスは 192.168.1.0/24 の範囲にあり、LAN B は range です。ゲートウェイ IP アドレスは、LAN A の場合は 192.168.1.254、LAN B の場合は 192.168.1.254 です。IPsec ルーターは各 LAN ゲートウェイから分離され、2 つのネットワークデバイスを使用します。eth0 はインターネットにアクセスする外部からアクセスできる静的 IP アドレスに割り当てられますが、eth1 は 1 つのネットワークノードからリモートネットワークノードへの LAN パケットを処理し、送信するためのルーティングポイントとして機能します。

各ネットワーク間の IPsec 接続は、共有前の鍵と r3dh4tl1nux の値を使用し、A と B の管理者は各 IPsec ルーター間で認証キーを自動的に生成して共有することに同意します。LAN A の管理者は IPsec connection ipsec0 という名前を決定し、LAN B の管理者は IPsec 接続 ipsec1 という名前を付けます。

以下の例は、LAN A の network-to-network IPsec 接続の ifcfg ファイルの内容を示しています。この例の接続を識別する一意の名前は ipsec0 であるため、作成されるファイルは /etc/sysconfig/network-scripts/ifcfg-ipsec0 と呼ばれます。

```
TYPE=IPSEC
ONBOOT=yes
IKE_METHOD=PSK
SRCGW=192.168.1.254
DSTGW=192.168.2.254
SRCNET=192.168.1.0/24
DSTNET=192.168.2.0/24
DST=X.X.X.X
```

以下の一覧では、このファイルの内容を説明します。

TYPE=IPSEC

接続のタイプを指定します。

ONBOOT=yes

起動時に接続を開始するように指定します。

IKE_METHOD=PSK

接続が認証の事前共有鍵メソッドを使用することを指定します。

SRCGW=192.168.1.254

ソースゲートウェイの IP アドレス。LAN A の場合、これは LAN A ゲートウェイで、LAN B の場合は LAN B ゲートウェイです。

DSTGW=192.168.2.254

宛先ゲートウェイの IP アドレス。LAN A の場合、これは LAN B ゲートウェイで、LAN B の場合は LAN A ゲートウェイです。

SRCNET=192.168.1.0/24

IPsec 接続の移行元ネットワークを指定します。この例では、LAN A のネットワーク範囲です。

DSTNET=192.168.2.0/24

IPsec 接続の宛先ネットワークを指定します。この例では、LAN B のネットワーク範囲です。

DST=X.X.X.X

LAN B の外部からアクセスできる IP アドレス。

以下の例は、両方のネットワークが相互に認証するために使用する /etc/sysconfig/network-scripts/keys-ipsecX(X は LAN A の場合は 0、LAN B の場合は 1)と呼ばれる事前共有キーファイルの内容です。このファイルのコンテンツは同一で、root ユーザーのみがこのファイルの読み取りや書き込みが可能でなければなりません。

```
IKE_PSK=r3dh4tl1nux
```

重要な影響

root ユーザーのみがファイルの読み取りまたは編集ができるように keys-ipsecX ファイルを変更するには、ファイルの作成後に以下のコマンドを使用します。

```
chmod 600 /etc/sysconfig/network-scripts/keys-ipsec1
```

認証キーを変更するには、両方の IPsec ルーターの keys-ipsecX ファイルを編集します。適切な接続の場合には、両方の鍵が同じでなければなりません。

以下の例は、IPsec 接続の /etc/racoon/racoon.conf 設定ファイルの内容です。ファイル下部の include 行が自動的に生成され、IPsec トンネルが実行されている場合のみ表示されることに注意してください。

```
# Racoon IKE daemon configuration file.
# See 'man racoon.conf' for a description of the format and entries.
path include "/etc/racoon";
path pre_shared_key "/etc/racoon/psk.txt";
path certificate "/etc/racoon/certs";

sainfo anonymous
{
 pfs_group 2;
 lifetime time 1 hour ;
 encryption_algorithm 3des, blowfish 448, rijndael ;
 authentication_algorithm hmac_sha1, hmac_md5 ;
 compression_algorithm deflate ;
}
include "/etc/racoon/X.X.X.X.conf"
```

以下は、リモートネットワークに接続するための特定の設定です。ファイルはX. X.X.X.conf と呼ばれます(X.X.X.X はリモート IPsec ルーターの IP アドレスです)。このファイルは、IPsec トンネルがアクティベートされると自動的に生成されるため、直接編集しないでください。

```
remote X.X.X.X
{
        exchange_mode aggressive, main;
```

```
 my_identifier address;
 proposal {
  encryption_algorithm 3des;
  hash_algorithm sha1;
  authentication_method pre_shared_key;
  dh_group 2 ;
 }
}
```

IPsec 接続を開始する前に、カーネルで IP 転送を有効にする必要があります。IP 転送を有効にするには、以下を実行します。

1. **/etc/sysctl.conf を編集し、net.ipv4.ip_forward を 1 に設定します。**

2. **以下のコマンドを使用して変更を適用します。**

```
sysctl -p /etc/sysctl.conf
```

IPsec 接続を開始するには、各ルーターで以下のコマンドを使用します。

```
ifup ipsec0
```

接続が有効になり、LAN A と LAN B の両方が相互に通信できます。ルートは、IPsec 接続で ifup を実行して、呼び出される初期化スクリプトを介して自動的に作成されます。ネットワークのルートの一覧を表示するには、以下のコマンドを使用します。

```
ip route list
```

IPsec 接続をテストするには、外部ルーティング可能なデバイス（この例では eth0）で tcpdump ユーティリティーを実行し、ホスト（またはネットワーク）間で転送されるネットワークパケットを表示し、IPsec で暗号化されていることを確認します。たとえば、LAN A の IPsec 接続を確認するには、以下のコマンドを使用します。

```
tcpdump -n -i eth0 host lana.example.com
```

パケットには AH ヘッダーが含まれる必要があり、ESP パケットとして表示されるはずです。ESP は暗号化されることを意味します。以下に例を示します（バックスラッシュは 1 行継続を表します）。

```
12:24:26.155529 lanb.example.com > lana.example.com: AH(spi=0x021c9834,seq=0x358): \
lanb.example.com > lana.example.com: ESP(spi=0x00c887ad,seq=0x358) (DF) \
(ipip-proto-4)
```

48.7.8. IPsec 接続の開始および停止

IPsec 接続が起動時にアクティベートするように設定されていない場合は、コマンドラインからこれを制御できます。

接続を開始するには、ホスト間 IPsec の各ホストで、またはネットワーク間 IPsec の場合は各 IPsec ルーターで以下のコマンドを使用します。

```
ifup <nickname>
```

ここで、**<nickname >** は以前に設定されたニックネームです（例： **ipsec0** ）。

接続を停止するには、以下のコマンドを使用します。

```
ifdown <nickname>
```

48.8. ファイアウォール

情報セキュリティーは通常、製品ではなくプロセスと考えられています。ただし、標準のセキュリティー実装は通常、アクセス権限を制御し、ネットワークリソースを認証、識別でき、追跡可能なユーザーに制限する専用のメカニズムを使用します。Red Hat Enterprise Linux には、ネットワークレベルのアクセス制御の問題に関する管理者およびセキュリティーエンジニアを支援するツールがいくつか含まれています。

ファイアウォールは、ネットワークセキュリティー実装のコアコンポーネントの 1 つです。いくつかのベンダー市場のファイアウォールソリューションは、市場のすべてのレベルに対応します。ホームユーザーは、重要なエンタープライズ情報を保護するため、1 つのPCをデータセンターソリューションから保護します。ファイアウォールは、Cisco、Nokia、Sonnicwall によるファイアウォールアプライアンスなどのスタンドアロンハードウェアソリューションです。また、Checkpoint、McAfee、Symantec などのベンダーは、ホームおよびビジネス市場向けのプロプライエタリソフトウェアファイアウォールソリューションも開発しました。

ハードウェアファイアウォールとソフトウェアのファイアウォールの違いに加え、ファイアウォールが別のソリューションから分離する方法にも違いがあります。表48.5「ファイアウォールのタイプ」では、3 つの一般的なファイアウォールタイプとその機能について説明します。

表48.5 ファイアウォールのタイプ

メソッド	説明	メリット	デメリット
NAT	NAT (Network Address Translation)は、プライベート IP サブネットワークを 1 つまたはパブリック IP アドレスの小規模なプールの背後に配置し、複数のソースへのすべての要求をマスカレードします。Linux カーネルには、JBDS カーネルサブシステムを介して NAT 機能が組み込まれています。	LAN 上のマシンに対して透過的に設定可能 1 つ以上の外部 IP アドレスの背後にある多くのマシンとサービスの保護により、管理者の作業が簡素化されます。 LAN へのユーザーアクセスと、LAN からのユーザーアクセスの制限を設定するには、NAT ファイアウォール/ゲートウェイでポートを開いて閉じることで設定できます。	ユーザーがファイアウォール外のサービスに接続した後、悪意のあるアクティビティーを防ぐことができない

メソッド	説明	メリット	デメリット
パケットフィルター	パケットフィルターリングファイアウォールは、LAN を通過する各データパケットを読み取ります。ヘッダー情報でパケットを読み取り、処理し、ファイアウォール管理者が実装するプログラム可能なルールセットに基づいてパケットをフィルターリングします。Linux カーネルには、JBS カーネルサブシステムを介してパケットフィルターリング機能が組み込まれています。	iptables フロントエンドユーティリティーでカスタマイズ可能 すべてのネットワークアクティビティーはアプリケーションレベルではなくルーターレベルでフィルターリングされるため、クライアント側でカスタマイズする必要はありません。 packets はプロキシー経由で送信されないため、クライアントからリモートホストへ直接接続するため、ネットワークのパフォーマンスは高速になります。	NORMAL は、プロキシーファイアウォールなどのコンテンツのパケットをフィルターできない NORMAL はプロトコル層でパケットを処理しますが、アプリケーション層でパケットをフィルターできません。 complex 複雑なネットワークアーキテクチャーは、特に IP マスカレード またはローカルサブネットおよび DMZ ネットワークと組み合わせた場合、パケットフィルターリングルールの確立を困難にすることができます。

メソッド	説明	メリット	デメリット
Proxy	プロキシーファイアウォールは、LAN クライアントからプロキシーマシンへの特定のプロトコルまたはタイプのすべての要求をフィルターリングします。これにより、ローカルクライアントの代わりにインターネットに対してこれらの要求が行われます。プロキシーマシンは、悪意のあるリモートユーザーと内部ネットワーククライアントマシン間のバッファーとして機能します。	管理者は、LAN の外部で機能するアプリケーションやプロトコルを制御できる 一部のプロキシーサーバーでは、インターネット接続を使用して要求するのではなく、頻繁にアクセスされるデータをローカルにキャッシュできます。これにより、帯域幅の消費を削減できます。 NORMAL プロキシーサービスは、厳密にログに記録および監視できるため、ネットワーク上のリソース使用率をより厳密に制御できます。	NORMAL プロキシーは、多くの場合、アプリケーション固有 (HTTP、Telnet など)、または protocol-restricted （ほとんどのプロキシーは TCP に接続されたサービスでのみ機能します）です。 アプリケーションサービスはプロキシーの背後で実行できないため、アプリケーションサーバーは別の形式のネットワークセキュリティーを使用する必要があります。 すべての要求および送信がクライアントからリモートサービスに直接渡されるのではなく 1 つのソースを通過するため、すべての要求と送信が 1 つのソースを通過するため、すべてのリクエストと送信がネットワークのボトルネックになることができます。

48.8.1. ubuntu および IPTables

Linux カーネルは、HynetQ と呼ばれる強力なネットワークサブシステムを特長としています。ubuntu サブシステムは、ステートフルまたはステートレスパケットフィルターリングと NAT および IP マスカレードサービスを提供します。また、高度なルーティングおよび接続状態管理のために IP ヘッダー情報をマレッディングする機能もあります。ubuntu は、iptables ツールを使用して制御します。

48.8.1.1. iptables の概要

電源と柔軟性は、iptables 管理ツールである ipchains sor の構文と同様のコマンドラインツールである iptables 管理ツールを使用して実装されます。

ただし、同様の構文は同様の実装を意味するわけではありません。ipchains では、ソースパスのフィルター、宛先パスのフィルターリング、送信元接続ポートと宛先接続ポートの両方のフィルターなど、特別なルールセットが必要です。

一方、iptables は、iptables を使用してネットワーク接続、検査、および処理を強化します。iptables は、高度なロギング、ルーティング前およびポストルーティングアクション、ネットワークアドレス変換、ポート転送をすべて 1 つのコマンドラインインターフェイスで提供します。

このセクションでは、iptables の概要について説明します。詳細は、「iptables」 を参照してください。

48.8.2. ファイアウォールの基本設定

ビルディング中のファイアウォールと同様に、コンピューターのファイアウォールは、悪意のあるソフトウェアがコンピューターに広がらないように試みます。また、承認されていないユーザーがコンピューターにアクセスできないのを防ぐのに役立ちます。

デフォルトの Red Hat Enterprise Linux インストールでは、コンピューターまたはネットワークと、信頼できないネットワーク（インターネットなど）の間にファイアウォールが存在します。これは、コンピューターのリモートユーザーがアクセスできるサービスを決定します。ファイアウォールを適切に設定すると、システムのセキュリティーを大幅に向上させることができます。インターネット接続のある Red Hat Enterprise Linux システムにファイアウォールを設定することが推奨されます。

48.8.2.1. Security Level Configuration Tool

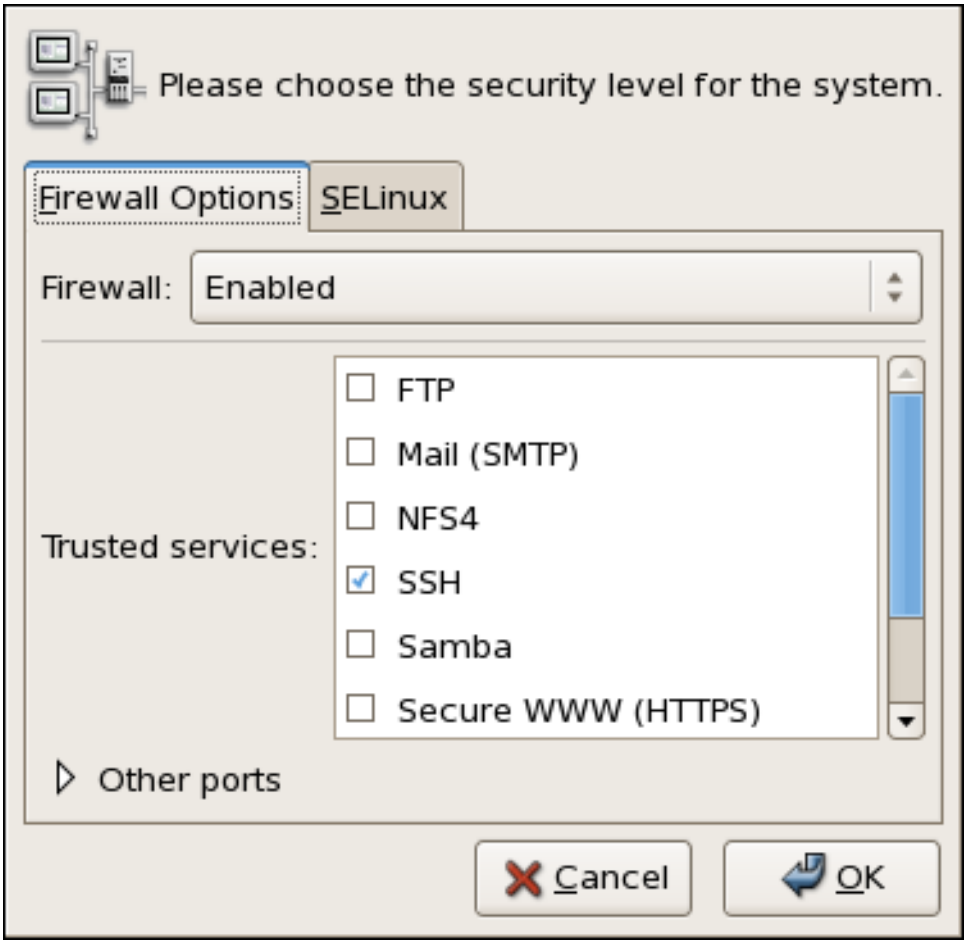
Red Hat Enterprise Linux インストールの Firewall Configuration 画面では、基本的なファイアウォールを有効にし、特定のデバイス、受信サービス、およびポートを許可するオプションが提供されました。

インストール後に、Security Level Configuration Tool を使用してこの設定を変更できます。

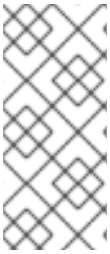
このアプリケーションを起動するには、以下のコマンドを使用します。

system-config-securitylevel

図48.15 Security Level Configuration Tool



[D]



注記

***Security Level Configuration Tool** は基本的なファイアウォールのみを設定します。システムにより複雑なルールが必要な場合は、「iptables」 で特定の **iptables** ルールの設定に関する詳細を確認してください。*

48.8.2.2. ファイアウォールの有効化および無効化

ファイアウォールには、以下のいずれかのオプションを選択します。

- ***disabled** - ファイアウォールを無効にすると、システムに完全にアクセスでき、セキュリティーチェックは行われません。これは、（インターネットではなく）信頼できるネットワークで実行している場合や、**iptables** コマンドラインツールを使用してカスタムファイアウォールを設定する必要がある場合にのみ選択する必要があります。*

WARNING

ファイアウォール設定とカスタマイズされたファイアウォールルールは、/etc/sysconfig/iptables ファイルに保存されます。Disabled を選択して OK をクリックすると、これらの設定とファイアウォールルールが失われます。

- *enabled: このオプションは、DNS 応答や DHCP 要求など、アウトバウンド要求に応答しない着信接続を拒否するように設定します。このマシンで実行中のサービスへのアクセスが必要な場合は、特定サービスに対してファイアウォールの通過許可を選択できます。*

  システムをインターネットに接続しているが、サーバーを実行する予定がない場合は、最も安全な選択肢になります。

48.8.2.3. 信頼できるサービス

Trusted services 一覧でオプションを有効にすると、指定したサービスがファイアウォールを通過できます。

WWW (HTTP)

HTTP プロトコルは Apache （およびその他の Web サーバー）によって Web ページを提供するために使用されます。Web サーバーを公開しようとする場合は、このチェックボックスを選択します。このオプションは、ローカルページの表示や Web ページの開発には必要ありません。このサービスは、httpd パッケージをインストールすることを要求します。

WWW (HTTP) を有効にしても、HTTPS (SSL バージョンの HTTP)のポートが開かれません。このサービスが必要な場合は、Secure WWW (HTTPS) チェックボックスを選択します。

FTP

FTP プロトコルは、ネットワーク上のマシン間でファイルを転送するために使用されます。FTP サーバーを公開しようとする場合は、このチェックボックスを選択します。このサービスは、vsftpd パッケージをインストールする必要があります。

SSH

Secure Shell (SSH)は、リモートマシンでコマンドにログインして実行するツールセットです。ssh 経由でマシンへのリモートアクセスを許可するには、このチェックボックスを選択します。このサービスは、openssh-server パッケージをインストールする必要があります。

Telnet

telnet は、リモートマシンにログインするためのプロトコルです。telnet 通信は暗号化されず、ネットワークスヌーピングからのセキュリティーは提供されません。受信 Telnet アクセスを許可することは推奨されません。telnet 経由でマシンへのリモートアクセスを許可するには、このチェックボックスを選択します。このサービスは、telnet-server パッケージをインストールする必要があります。

メール(SMTP)

SMTP は、リモートホストをマシンに直接接続してメールを配信できるようにするプロトコルです。POP3 または IMAP を使用して ISP サーバーからメールを収集する場合、または fetchmail などのツールを使用する場合は、このサービスを有効にする必要はありません。マシンにメールを配信できるようにするには、このチェックボックスを選択します。SMTP サーバーが適切に設定されていないと、リモートマシンがサーバーを使用してスパムを送信することができることに注意してください。

NFS4

*NFS (Network File System)は、*NIX システムで一般的に使用されるファイル共有プロトコルです。このプロトコルのバージョン 4 は、先行プロトコルよりも安全です。システムのファイルまたはディレクトリーを他のネットワークユーザーと共有する場合は、このチェックボックスを選択します。*

Samba

Samba は、Microsoft のプロプライエタリー SMB ネットワークプロトコルの実装です。ファイル、ディレクトリー、またはローカルで接続されたプリンターを Microsoft Windows マシンと共有する必要がある場合は、このチェックボックスを選択します。

48.8.2.4. その他のポート

Security Level Configuration Tool には、iptables によって信頼されるカスタム IP ポートを指定するためのその他 のポートセクションが含まれています。たとえば、IRC およびインターネット印刷プ

ロトコル(IPP)がファイアウォールを通過できるようにするには、以下を その他のポート セクションに追加します。

194:tcp,631:tcp

48.8.2.5. 設定の保存

OK をクリックして変更を保存し、ファイアウォールを有効または無効にします。Enable firewall が選択されている場合、選択されたオプションは iptables コマンドに変換され、/etc/sysconfig/iptables ファイルに書き込まれます。また、iptables サービスも起動して、選択したオプションを保存した後すぐにファイアウォールがアクティブになるようにします。Disable firewall が選択されている場合、/etc/sysconfig/iptables ファイルは削除され、iptables サービスはすぐに停止します。

選択したオプションは、/etc/sysconfig/system-config-securitylevel ファイルにも書き込まれ、次回アプリケーションを起動したときに設定を復元できるようにします。このファイルは手動で編集しないでください。

ファイアウォールはすぐにアクティブ化されますが、iptables サービスは起動時に自動的に起動するように設定されていません。詳細は、「IPTables サービスのアクティブ化」 を参照してください。

48.8.2.6. IPTables サービスのアクティブ化

ファイアウォールルールは、iptables サービスが実行している場合にのみアクティブになります。サービスを手動で起動するには、以下のコマンドを使用します。

```
service iptables restart
```

システムの起動時に iptables が起動するようにするには、以下のコマンドを使用します。

```
chkconfig --level 345 iptables on
```

ipchains サービスは、Red Hat Enterprise Linux には含まれていません。ただし、ipchains がインストールされている場合（たとえば、アップグレードが実行され、システムに ipchains が以前にインストールされていた場合など）、ipchains サービスと iptables サービスを同時にアクティブにしないでください。ipchains サービスが無効になり、起動時に起動しないように設定されていることを確認するには、以下の 2 つのコマンドを使用します。

```
service ipchains stop
chkconfig --level 345 ipchains off
```

48.8.3. IPTables の使用

iptables を使用する最初の手順は、iptables サービスを起動することです。以下のコマンドを使用して iptables サービスを起動します。

```
service iptables start
```

注記

iptables サービスのみを使用する場合は、ip6tables サービスをオフにできます。ip6tables サービスを無効にする場合は、必ず IPv6 ネットワークを無効にするようにしてください。一致するファイアウォールがないと、ネットワークデバイスをアクティブのままにしないでください。

システムの起動時に iptables を強制的に起動するには、以下のコマンドを使用します。

```
chkconfig --level 345 iptables on
```

これにより、システムがランレベル 3、4、または 5 で起動するたびに iptables を強制的に起動します。

48.8.3.1. iptables コマンドの構文

以下の iptables コマンドの例は、基本的なコマンド構文を示しています。

```
iptables -A <chain> -j <target>
```

-A オプションは、< chain> にルールを追加するように指定し ます。各チェーンは 1 つ以上の ルール で設定されるため、ルール セット とも呼ばれます。

3 つの組み込みチェーンは INPUT、OUTPUT、および FORWARD です。これらのチェーンは永続的であるため、削除できません。チェーンは、パケットが処理されるポイントを指定します。

-j & lt;target > オプションは、ルールのターゲットを指定します。つまり、パケットがルールと一致する場合のアクション。組み込みターゲットの例は ACCEPT、DROP、および REJECT です。

利用可能なチェーン、オプション、およびターゲットの詳細は、iptables の man ページを参照し

てください。

48.8.3.2. 基本的なファイアウォールポリシー

基本的なファイアウォールポリシーを確立すると、より詳細なユーザー定義のルールを構築するための基盤が作成されます。

各 iptables チェーンはデフォルトのポリシーで設定され、デフォルトポリシーと連携する 0 個以上のルールを使用して、ファイアウォールの全体的なルールセットを定義します。

チェーンのデフォルトポリシーは DROP または ACCEPT のいずれかです。通常、セキュリティー関連の管理者は DROP のデフォルトポリシーを実装し、ケースバイケースで特定のパケットのみを許可します。たとえば、以下のポリシーは、ネットワークゲートウェイ上の着信パケットおよび送信パケットをすべてブロックします。

```
iptables -P INPUT DROP
iptables -P OUTPUT DROP
```

また、転送されたパケット （ファイアウォールからその宛先ノードにルーティングされるネットワークトラフィック）も拒否され、内部クライアントがインターネットへの不正公開を制限することが推奨されます。これを行うには、以下のルールを使用します。

```
iptables -P FORWARD DROP
```

各チェーンのデフォルトポリシーを確立したら、特定のネットワークおよびセキュリティー要件に対してさらにルールを作成および保存できます。

以下のセクションでは、iptables ルールを保存する方法と、iptables ファイアウォールを構築する際に実装するルールの一部の概要を説明します。

48.8.3.3. IPTables ルールの保存および復元

iptables への変更は推移的です。システムを再起動するか、iptables サービスが再起動すると、ルールは自動的にフラッシュされ、リセットされます。iptables サービスの起動時に読み込まれるようにルールを保存するには、以下のコマンドを使用します。

```
service iptables save
```

ルールは /etc/sysconfig/iptables ファイルに保存され、サービスの起動時またはマシンが再起動さ

れるたびに適用されます。

48.8.4. 一般的な IPTables フィルターリング

リモート攻撃者が LAN にアクセスできないことは、ネットワークセキュリティーの最も重要な側面の 1 つです。LAN の整合性は、厳格なファイアウォールルールを使用することで、悪意のあるリモートユーザーから保護する必要があります。

ただし、すべての着信、送信、および転送パケットをブロックするデフォルトのポリシーを設定すると、ファイアウォール/ゲートウェイおよび内部 LAN ユーザーが相互に、または外部リソースと通信することはできません。

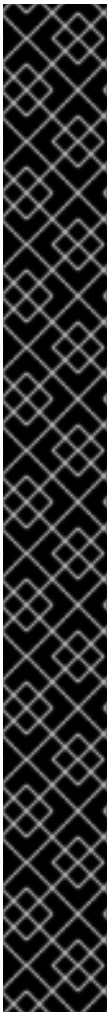
ユーザーがネットワーク関連の機能を実行し、ネットワークアプリケーションを使用できるようにするために、管理者は通信用に特定のポートを開く必要があります。

たとえば、ファイアウォール でポート 80 へのアクセスを許可するに は、以下のルールを追加します。

```
iptables -A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
```

これにより、標準のポート 80 を使用して通信する Web サイトを参照できます。セキュアな Web サイト（例：https://www.example.com/）へのアクセスを許可するには、以下のようにポート 443 へのアクセスも指定する必要があります。

```
iptables -A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
```



重要な影響

iptables ルールセットを作成する場合、順序は重要です。

ルールで 192.168.100.0/24 サブネットからのパケットがドロップされ、その後に 192.168.100.13 （ドロップされたサブネット内）からのパケットを許可するルールが続きます。

192.168.100.13 からのパケットを許可するルールは、残りのサブネットを破棄するルールの前に指定する必要があります。

既存のチェーンの特定の場所にルールを挿入するには、-I オプションを使用します。以下に例を示します。

```
iptables -I INPUT 1 -i lo -p all -j ACCEPT
```

このルールは、ローカルループバックデバイスのトラフィックを許可する INPUT チェーンの最初のルールとして挿入されます。

LAN へのリモートアクセスが必要になる場合があります。SSH などのセキュアなサービスは、LAN サービスへの暗号化されたリモート接続に使用できます。

PPP ベースのリソース（モデムバンクや一括 ISP アカウントなど）を使用する管理者は、ダイヤルアップアクセスを使用して安全なファイアウォールバリアを回避することができます。これらは直接接続であるため、モデム接続は通常ファイアウォール/ゲートウェイの背後にあります。

ただし、ブロードバンド接続を持つリモートユーザーの場合は、特別なケースを行うことができます。iptables を設定して、リモート SSH クライアントからの接続を受け入れることができます。たとえば、以下のルールはリモート SSH アクセスを許可します。

```
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 22 -j ACCEPT
```

これらのルールにより、インターネットに直接接続された単一の PC やファイアウォール/ゲートウェイなど、個別システムの送受信アクセスが可能になります。ただし、ファイアウォール/ゲートウェイの背後にあるノードがこれらのサービスにアクセスできるようにすることはできません。これらの

サービスへの LAN アクセスを許可するには、iptables でNAT(Network Address Translation)を使用します。

48.8.5. FORWARD および NAT ルール

ほとんどの ISP は、公開されたルーティング可能な IP アドレスの数だけを、それらが提供する組織に限られています。

したがって、管理者は LAN 上の全ノードにパブリック IP アドレスを割り当てずに、インターネットサービスへのアクセスを共有する代替方法を見つける必要があります。プライベート IP アドレスの使用は、LAN 上の全ノードが内部および外部ネットワークサービスに適切にアクセスできるようにする最も一般的な方法です。

エッジルーター（ファイアウォールなど）は、インターネットから受信送信を受信し、パケットを目的の LAN ノードにルーティングすることができます。同時に、ファイアウォール/ゲートウェイは、LAN ノードからリモートインターネットサービスに発信要求をルーティングすることもできます。

ネットワークトラフィックの転送は、特に内部 IP アドレスを偽装し、リモートの攻撃者のマシンが LAN 上のノードとして機能する最新のクラッキングツールを利用できる場合に危険になる可能性があります。

これを防ぐために、iptables はネットワークリソースの異常な使用を防ぐために実装可能なルーティングおよび転送ポリシーを提供します。

FORWARD チェーンを使用すると、管理者は LAN 内でパケットをルーティングできる場所を制御できます。たとえば、LAN 全体の転送を許可するには(eth1 のファイアウォール/ゲートウェイに内部 IP アドレスが割り当てられていると仮定)、以下のルールを使用します。

```
iptables -A FORWARD -i eth1 -j ACCEPT
iptables -A FORWARD -o eth1 -j ACCEPT
```

このルールにより、ファイアウォール/ゲートウェイの背後にあるシステムに内部ネットワークへのアクセスが付与されます。ゲートウェイは、1 つの LAN ノードから目的の宛先ノードにパケットをルーティングし、すべてのパケットを eth1 デバイスを介して渡します。

注記

デフォルトでは、Red Hat Enterprise Linux カーネルの IPv4 ポリシーは IP 転送のサポートを無効にします。これにより、Red Hat Enterprise Linux を実行するマシンが専用のエッジルーターとして機能できなくなります。IP 転送を有効にするには、次のコマンドを使用します。

```
sysctl -w net.ipv4.ip_forward=1
```

この設定の変更は現行セッションでのみ有効です。再起動やネットワークサービスの再起動後は維持されません。IP 転送を永続的に設定するには、以下のように /etc/sysctl.conf ファイルを編集します。

以下の行を見つけます。

```
net.ipv4.ip_forward = 0
```

次のように編集します。

```
net.ipv4.ip_forward = 1
```

以下のコマンドを使用して、sysctl.conf ファイルへの変更を有効にします。

```
sysctl -p /etc/sysctl.conf
```

48.8.5.1. POSTROUTING および IP マスカレード

ファイアウォールの内部 IP デバイス経由で転送されたパケットを受け入れると、LAN ノードが相互に通信することができますが、インターネットに外部から通信することはできません。

プライベート IP アドレスを持つ LAN ノードが外部のパブリックネットワークと通信できるようにするには、IP マスカレード の のファイアウォールを設定します。これにより、LAN ノードからの要求をファイアウォールの外部デバイスの IP アドレスでマスクします（この場合は eth0）。

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

このルールは NAT パケットマッチングテーブル(-t nat)を使用し、ファイアウォールの外部ネットワークデバイス(-o eth0)上の NAT 用の組み込みの POSTROUTING (-A POSTROUTING)を指定しま

す。

POSTROUTING を使用すると、パケットがファイアウォールの外部デバイスから出るので変更できます。

-j MASQUERADE ターゲットは、ファイアウォール/ゲートウェイの外部 IP アドレスを持つノードのプライベート IP アドレスをマスクするために指定されます。

48.8.5.2. PREROUTING

外部で利用可能な内部ネットワークにサーバーがある場合は、NAT の PREROUTING チェーンの -j DNAT ターゲットを使用して、内部サービスへの接続を要求する着信パケットを転送できる宛先 IP アドレスとポートを指定できます。

たとえば、受信する HTTP リクエストを 172.31.0.23 で専用の Apache HTTP サーバーに転送する場合は、以下のコマンドを使用します。

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j DNAT --to 172.31.0.23:80
```

このルールは、nat テーブルが組み込みの PREROUTING チェーンを使用して、受信 HTTP リクエストを 172.31.0.23 に一覧表示されている宛先 IP アドレスのみに転送することを指定します。

注記

FORWARD チェーンにデフォルトの DROP ポリシーがある場合は、宛先 NAT ルーティングが可能になるように、すべての着信 HTTP 要求を転送するルールを追加する必要があります。これを行うには、以下のコマンドを使用します。

```
iptables -A FORWARD -i eth0 -p tcp --dport 80 -d 172.31.0.23 -j ACCEPT
```

このルールは、ファイアウォールからの着信 HTTP 要求（ファイアウォールの背後にある Apache HTTP Server）をすべて転送します。

48.8.5.3. DMZs および IPTables

iptables ルールを作成して、専用 HTTP サーバーや FTP サーバーなどの特定のマシンにトラフィックをルーティングする ことが できます。DMZ は、インターネットなどのパブリック伝送者にサービスを提供すること専用の特別なローカルサブネットワークです。

たとえば、受信 HTTP 要求を 10.0.4.2 (LAN の 192.168.1.0/24 範囲外)で専用の HTTP サーバーにルーティングするルールを設定するには、NAT は PREROUTING テーブルを使用してパケットを適切な宛先に転送します。

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j DNAT --to-destination 10.0.4.2:80
```

このコマンドでは、LAN の外部からポート 80 へのすべての HTTP 接続は、他の内部ネットワークとは別のネットワーク上の HTTP サーバーにルーティングされます。この形式のネットワークセグメンテーションは、ネットワーク上のマシンへの HTTP 接続を許可するよりも安全であることが証明されます。

HTTP サーバーがセキュアな接続を受け入れるように設定されている場合は、ポート 443 も転送する必要があります。

48.8.6. 悪意のあるソフトウェアおよびスポンクション IP アドレス

LAN 内の特定のサブネットや特定のノードへのアクセスを制御する、より詳細なルールを作成できます。また、Trojans、worms、およびその他のクライアント/サーバーウェアークスティクスなど、異常な特定のアプリケーションやプログラムを制限したりすることもできます。

たとえば、Trojans によっては、ポート 31337 から 31340 （クラッキング用語で elite ポートと呼ばれる）のサービスに対してネットワークをスキャンします。

これらの標準以外のポートを介して通信する正当なサービスがないため、そのサービスをブロックすると、ネットワーク上の潜在的なノードが独立してリモートマスターサーバーと通信する可能性を効果的に損なう可能性があります。

以下のルールは、ポート 31337 の使用を試みるすべての TCP トラフィックをドロップします。

```
iptables -A OUTPUT -o eth0 -p tcp --dport 31337 --sport 31337 -j DROP
iptables -A FORWARD -o eth0 -p tcp --dport 31337 --sport 31337 -j DROP
```

プライベート IP アドレス範囲のスプーフィングを試みて LAN を推測しようとする外部接続をブロックすることもできます。

たとえば、LAN が 192.168.1.0/24 の範囲を使用する場合、インターネット向けネットワークデバイス（例：eth0）に LAN IP 範囲のアドレスを持つパケットをドロップするように指示するルールを設計

できます。

転送されたパケットをデフォルトポリシーとして拒否することが推奨されます。そのため、外部向けデバイス(eth0)へのスプーフィングされた IP アドレスは自動的に拒否されます。

```
iptables -A FORWARD -s 192.168.1.0/24 -i eth0 -j DROP
```

注記

追加 されたルールを扱う場合は DROP ターゲットと REJECT ターゲットの間に区別があります。

REJECT ターゲットはアクセスを拒否し、サービスへの接続を試みるユーザーに connection refused エラーを返します。名前が示すように DROP ターゲットは、警告なしでパケットをドロップします。

管理者は、これらのターゲットを使用する際に独自の判断を使用することができます。ただし、ユーザーの混乱を回避し、接続を継続しようとするには、REJECT ターゲットが推奨されます。

48.8.7. iptables および接続トラッキング

接続の 状態に基づいて、サービスへの接続を検査および制限できます。iptables 内のモジュールは、接続 追跡 と呼ばれるメソッドを使用して受信接続に関する情報を保存します。以下の接続状態に基づいてアクセスを許可または拒否できます。

- NEW - HTTP リクエストなどの新しい接続を要求するパケット。
- ESTABLISHED: 既存の接続の一部であるパケットです。
- RELATED: 新しい接続を要求しているが、既存の接続の一部であるパケットです。たとえば、FTP はポート 21 を使用して接続を確立しますが、データは別のポート（通常はポート 20）で転送されます。
- INVALID: コネクション追跡テーブルの接続の一部ではないパケット。

プロトコル自体がステートレス(UDP など)の場合でも、iptables 接続追跡のステートフル機能をネットワークプロトコルで使用できます。以下の例は、接続追跡を使用して、確立された接続に関連付けられたパケットのみを転送するルールを示しています。

```
iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
```

48.8.8. IPv6

IPv6 と呼ばれる次世代インターネットプロトコルの導入は、IPv4 （または IP）の 32 ビットのアドレス制限を超えて拡張されます。IPv6 は 128 ビットアドレスに対応しているため、IPv6 対応の通信ネットワークは、IPv4 よりも大量のルーティング可能なアドレスに対応できます。

Red Hat Enterprise Linux は、Netfilter 6 サブシステムと ip6tables コマンドを使用して IPv6 ファイアウォールルールをサポートします。Red Hat Enterprise Linux 5 では、IPv4 サービスと IPv6 サービスの両方がデフォルトで有効になっています。

ip6tables コマンド構文は、128 ビットアドレスをサポートする場合を除き、iptables と同じです。たとえば、以下のコマンドを使用して、IPv6 対応のネットワークサーバーで SSH 接続を有効にします。

```
ip6tables -A INPUT -i eth0 -p tcp -s 3ffe:ffff:100::1/128 --dport 22 -j ACCEPT
```

IPv6 ネットワークの詳細は、http://www.ipv6.org/ の IPv6 Information Page を参照してください。

48.8.9. 関連情報

ファイアウォールにはいくつかの側面があり、この章では説明できない Linux ubuntu サブシステムがあります。詳細は、以下のリソースを参照してください。

48.8.9.1. インストールされているドキュメント

- 多くのコマンドオプションの定義など、iptables コマンドの詳細については、「iptables」 を参照してください。
- iptables の man ページには、さまざまなオプションの概要が記載されています。

48.8.9.2. 便利な Web サイト

- *http://www.netfilter.org/ - SQS および iptables プロジェクトの公式ホームページです。*

- *http://www.tldp.org/: Linux ドキュメントプロジェクトには、ファイアウォールの作成と管理に関連する便利なガイドが複数含まれています。*

- *http://www.iana.org/assignments/port-numbers - Internet Assigned Numbers Authority によって割り当てられた登録済みおよび一般的なサービスポートの公式リストです。*

48.8.9.3. 関連ドキュメント

- *『Red Hat Linux』 ファイアウォール(Bill McCarty)、Red Hat のプレス。Red Hat Press: オープンソースのパケットフィルターリングテクノロジーを使用してネットワークおよびサーバーファイアウォールを構築するための包括的なリファレンスです（例：Bill McCarty）。これには、ファイアウォールログの分析、ファイアウォールルールの開発、およびさまざまなグラフィカルツールを使用したファイアウォールのカスタマイズに関するトピックが含まれます。*

- *『Linux Firewalls』 (Robert Ziegler)、New Riders Press には、2.2 カーネル ipchains と iptables の両方を使用してファイアウォールを構築する際のさまざまな情報が含まれています。リモートアクセスの問題や侵入検知システムなどの追加のセキュリティートピックについても扱います。*

48.9. IPTABLES

Red Hat Enterprise Linux には、ネットワーク パケットフィルターリング の高度なツールが含まれています。これは、カーネル内のネットワークスタックの開始、移動、終了時にネットワークパケットを制御するプロセスです。2.4 より前のバージョンは、パケットのフィルターリングと、フィルターリングプロセスの各ステップでパケットに適用されるルールの一覧を ipchains に依存していました。2.4 カーネルには、ipchains と似た iptables (netfilterとも呼ばれる)が導入されましたが、ネットワークパケットのフィルターリングに使用できる範囲と制御が大幅に拡張されました。

本章では、パケットフィルターリングの基本について重点を置いて、ipchains と iptables の違いを定義します。また、iptables コマンドで使用できるさまざまなオプションについて説明し、システムの再起動後もフィルターリングルールを保持する方法を説明します。

iptables ルールを作成し、これらのルールに基づいてファイアウォールを設定する方法については、「関連情報」 を参照してください。

WARNING

2.4 以降のカーネルのデフォルトのファイアウォールメカニズムは iptables ですが、ipchains がすでに実行されている場合は iptables を使用することはできません。起動時に ipchains が存在する場合は、カーネルがエラーを発行し、iptables の起動に失敗します。

ipchains の機能は、これらのエラーによる影響を受けません。

48.9.1. パケットフィルターリング

Linux カーネルは kcontext 機能 を 使用してパケットをフィルターリングし、システムで受信または通過できるようにします。この機能は Linux カーネルに組み込まれており、以下のように 3 つの組み込み テーブル または ルールリスト があります。

- *filter: ネットワークパケットを処理するデフォルトのテーブル。*
- *NAT - 新しい接続を作成し、ネットワークアドレス変換 (NAT)に使用されるパケットを変更するために使用されます。*
- *mangle - 特定のタイプのパケット変更に使用されます。*

各テーブルには、netfilter によってパケット で 実行されるアクションに対応する組み込みチェーンのグループがあります。

filter テーブルの組み込みチェーンは次のとおりです。

- *INPUT: ホストの対象となるネットワークパケットに適用されます。*
- *OUTPUT: ローカルで生成されたネットワークパケットに適用されます。*

- *FORWARD: ホスト経由でルーティングされるネットワークパケットに適用されます。*

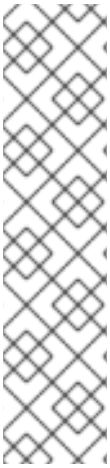
nat テーブルの組み込みチェーンは以下のとおりです。

- *PREROUTING - ネットワークパケットが到達するとそのパケットを変更します。*
- *OUTPUT: ローカルで生成されたネットワークパケットが送信される前に変更します。*
- *POSTROUTING - ネットワークパケットが送信される前に変更します。*

mangle テーブルの組み込みチェーンは次のとおりです。

- *INPUT: ホストの対象となるネットワークパケットを変更します。*
- *OUTPUT: ローカルで生成されたネットワークパケットが送信される前に変更します。*
- *FORWARD - ホストを介してルーティングされるネットワークパケットを変更します。*
- *PREROUTING - 着信ネットワークパケットがルーティングされる前に、それらを変更します。*
- *POSTROUTING - ネットワークパケットが送信される前に変更します。*

Linux システムによって受信または送信されたすべてのネットワークパケットは、少なくとも 1 つのテーブルの対象となります。ただし、パケットはチェーンの最後に検出される前に、各テーブル内の複数のルールの対象となる場合があります。これらのルールの構造や目的は異なる場合がありますが、特定のプロトコルやネットワークサービスを使用する場合、通常は特定の IP アドレスまたは特定の IP アドレス、または一連のアドレスから送信されるパケットを特定したいと思われます。



注記

デフォルトでは、ファイアウォールルールは /etc/sysconfig/iptables ファイルまたは /etc/sysconfig/ip6tables ファイルに保存されます。

iptables サービスは、Linux システムの起動時に DNS 関連のサービスの前に起動します。つまり、ファイアウォールルールは数値の IP アドレス（例：192.168.0.1）のみを参照できます。このようなルールのドメイン名（例：host.example.com）はエラーを生成します。

宛先に関係なく、パケットがテーブルの 1 つにある特定のルールに一致すると、ターゲット またはアクションが適用されます。ルールが一致するパケットに ACCEPT ターゲットを指定する場合、パケットは残りのルールチェックをスキップし、宛先に進むことができます。ルールが DROP ターゲットを指定する場合、そのパケットはシステムへのアクセスを拒否し、パケットを送信したホストに返送されません。ルールで QUEUE ターゲットが指定されている場合、パケットはユーザー空間に渡されます。ルールでオプションの REJECT ターゲットが指定されている場合、パケットは破棄されますが、エラーパケットがパケットの送信元に送信されます。

すべてのチェーンには、ACCEPT、DROP、REJECT、または QUEUE のデフォルトポリシーがあります。チェーン内のどのルールもパケットに適用されない場合、パケットはデフォルトのポリシーに従って処理されます。

iptables コマンドはこれらのテーブルを設定し、必要に応じて新しいテーブルを設定します。

48.9.2. IPTables と IPChains の相違点

ipchains と iptables はいずれも、Linux カーネル内で動作するルールのチェーンを使用して、指定されたルールまたはルールセットとの一致に基づいてパケットをフィルターリングします。ただし、iptables は、パケットのフィルターリングをより拡張可能な方法で提供しており、管理者はシステムにとって複雑なものを構築することなく、制御を強化できます。

ipchains と iptables の主な相違点に注意してください。

iptables を使用すると、フィルターされた各パケットは、複数のチェーンではなく 1 つのチェーンのルールを使用して処理されます。

たとえば、ipchains を使用するシステムに送信される FORWARD パケットは、INPUT、FORWARD、および OUTPUT チェーンを通過して、宛先を続行する必要があります。ただし、iptables はローカルシステム向けの宛先である場合のみ、iptables はパケットを INPUT チェーンに送信し、ローカルシステムがパケットを生成した場合にのみ OUTPUT チェーンに送信します。

したがって、実際にはパケットを処理するチェーン内に特定のパケットを取得するように設計されたルールを配置することが重要です。

DENY ターゲットが DROP に変更になりました。

ipchains では、チェーン内のルールに一致するパケットは DENY ターゲットに転送することができます。このターゲットは、iptables で DROP に変更する必要があります。

ルールにオプションを配置する際の順序は重要です。

ipchains では、ルールオプションの順序は重要ではありません。

iptables コマンドの構文はより厳格です。iptables コマンドでは、送信元ポートまたは宛先ポートの前にプロトコル(ICMP、TCP、または UDP)を指定する必要があります。

ネットワークインターフェイスは、ファイアウォールルールの正しいチェーンに関連付けられている必要があります。

たとえば、受信インターフェイス(-i オプション)は INPUT または FORWARD チェーンでのみ使用できます。同様に、出力インターフェイス(-o オプション)は、FORWARD チェーンまたは OUTPUT チェーンでのみ使用できます。

つまり、INPUT チェーンと受信インターフェイスは連携し、OUTPUT チェーンと発信インターフェイスは連携します。FORWARD チェーンは、着信インターフェイスと発信インターフェイスの両方で機能します。

OUTPUT チェーンは着信インターフェイスで使用されなくなり、送信インターフェイスを通過するパケットは INPUT チェーンを認識しません。

これは、変更の包括的なリストではありません。詳細は、「関連情報」 を参照してください。

48.9.3. IPTables のコマンドオプション

パケットのフィルターリングに関するルールは、iptables コマンドを使用して作成されます。パケットの以下の側面は、基準として最もよく使用されます。

- パケットタイプ: コマンドがフィルターするパケットのタイプを指定します。

- パケットソース/宛先： コマンドがパケットの送信元または宛先に基づいてフィルターするパケットを指定します。

- target - 上記の基準に一致するパケットに対して実行するアクションを指定します。

パケットのこれらの側面に対応する特定のオプションの詳細は、「iptables の一致オプション」 および 「ターゲットオプション」 を参照してください。

ルールを有効にするには、特定の iptables ルールで使用されるオプションは、ルール全体の目的および条件に基づいて論理的にグループ化する必要があります。このセクションの残りの部分では、iptables コマンドで一般的に使用されるオプションを説明します。

48.9.3.1. IPTables コマンドオプションの構造

多くの iptables コマンドの構造は次のとおりです。

```
iptables [-t <table-name>] <command> <chain-name> \
  <parameter-1> <option-1> \
  <parameter-n> <option-n>
```

<table-name& gt;: ルールが適用されるテーブルを指定します。省略した場合は、filter テーブルが使用されます。

<command >: ルールの追加や削除など、実行するアクションを指定します。

<chain-name& gt; - 編集、作成、または削除を行うチェーンを指定します。

<parameter>-<option> ペア - ルールに一致するパケットの処理方法を指定するパラメーター および関連オプション。

iptables コマンドの長さと複雑さは、その目的に基づいて大幅に変更される可能性があります。

たとえば、チェーンからルールを削除するコマンドは非常に短くなります。

iptables -D <chain-name> <line-number>

一方、さまざまな特定のパラメーターやオプションを使用して特定のサブネットからパケットをフィルターするルールを追加するコマンドは、かなり長くなる可能性があります。iptables コマンドを作成する際には、有効なルールを構築するために追加のパラメーターとオプションが必要なパラメーターやオプションがあることに注意してください。これにより、より多くのパラメーターを必要とする追加のパラメーターを使用してカスケード効果が発生する可能性があります。別のオプションのセットを必要とするすべてのパラメーターおよびオプションが満たされるまで、ルールは有効ではありません。

iptables コマンド構造の包括的なリストを表示するには、iptables -h と入力します。

48.9.3.2. コマンドオプション

コマンドオプションは、特定のアクションを実行するように iptables に指示します。iptables コマンドごとに 1 つのコマンドオプションのみが許可されます。help コマンドを除き、すべてのコマンドは大文字で記述されます。

iptables コマンドは次のとおりです。

- *- a - 指定されたチェーンの最後にルールを追加します。以下の -I オプションとは異なり、整数の引数は指定しません。常に指定されたチェーンの最後にルールを追加します。*
- *-c - ユーザー指定のチェーンに追加する前に、特定のルールを確認します。このコマンドは、追加のパラメーターおよびオプションの入力を求めて、複雑な iptables ルールを構築するのに役立ちます。*
- *-d <integer> | <rule >: 特定のチェーンのルールを番号（チェーンの 5 番目のルールの 5 など）、またはルール指定で削除します。ルールの指定は、既存のルールに完全に一致する必要があります。*
- *-e - ユーザー定義のチェーンの名前を変更します。ユーザー定義のチェーンは、デフォルトの既存のチェーン以外のチェーンです。（ユーザー定義のチェーンの作成に関する詳細は、以下の -N オプションを参照してください）。 これは表の構造には影響しません。*

注記

デフォルトのチェーンのいずれかの名前を変更しようとすると、システムは Match not found エラーを報告します。デフォルトのチェーンの名前を変更することはできません。

- *-f - 選択したチェーンをフラッシュして、チェーン内のすべてのルールを効果的に削除します。チェーンが指定されていない場合、このコマンドはすべてのチェーンからすべてのルールをフラッシュします。*

- *-h: コマンド構造の一覧と、コマンドパラメーターおよびオプションの概要を提供します。*

- *-I [<integer>] - ユーザー定義の整数引数で指定された時点で、指定したチェーンにルールを挿入します。引数が指定されていない場合、ルールはチェーンの上部に挿入されます。*

注意

上記のように、チェーン内のルールの順序によって、どのルールがどのパケットに適用されるかが決まります。これは、-A または -I オプションのいずれかを使用してルールを追加するときに覚えておくことが重要です。

これは、整数の引数で -I を使用してルールを追加する場合に特に重要です。チェーンにルールを追加するときに既存の番号を指定すると、iptables は既存のルールの 前 （またはそれ以上）に新しいルールを追加します。

- *-l - コマンドの後に指定したチェーン内のすべてのルールを一覧表示します。デフォルトの フィルター テーブルのすべてのチェーンのすべてのルールを一覧表示するには、チェーンまたはテーブルを指定しないでください。それ以外の場合は、特定のテーブルの特定のチェーンのルールを一覧表示するには、以下の構文を使用します。*

```
iptables -L <chain-name> -t <table-name>
```

ルール番号を提供し、より詳細なルールの説明を許可する -L コマンドオプションの追加オプションは、「オプションの一覧表示」 で説明されています。

- -n - ユーザー指定の名前で新しいチェーンを作成します。チェーン名は一意である必要があります。一意でなければ、エラーメッセージが表示されます。
- -p: 指定されたチェーンのデフォルトポリシーを設定します。これにより、パケットがルールを照合せずにチェーン全体を通過すると、ACCEPT や DROP などの指定されたターゲットに送信されます。
- -r: 指定されたチェーン内のルールを置き換えます。ルールの番号は、チェーンの名前の後に指定する必要があります。チェーンの最初のルールはルール番号 1 に対応します。
- -x: ユーザー指定のチェーンを削除します。ビルトインチェーンは削除できません。
- -z - テーブルのすべてのチェーンのバイトカウンターおよびパケットカウンターをゼロに設定します。

48.9.3.3. iptables パラメーターオプション

特定のチェーン内でルールの追加、追加、削除、挿入、または置換に使用されるものを含め、特定の iptables コマンドには、パケットフィルターリングルールを構築するためにさまざまなパラメーターが必要です。

- -c - 特定のルールのカウンターを設定します。このパラメーターは、PKTS オプションおよび BYTES オプションを受け入れて、リセットするカウンターを指定します。
- -d: ルールに一致するパケットの宛先ホスト名、IP アドレス、またはネットワークを設定します。ネットワークに一致する場合、以下の IP アドレス/ネットマスクの形式がサポートされます。
  - N.N.N.N / M.M.M: N.N.N.N は IP アドレス範囲で、M. M.M.M はネットマスクです。
  - N.N.N.N /M - N.N.N は IP アドレス範囲で、M はビットマスクです。

- *-f - このルールが断片化されたパケットのみに適用されます。*

  このパラメーターの後に感嘆符(!)オプションを使用して、断片化されていないパケットのみが一致するように指定できます。

  注記

  断片化されたパケットが IP プロトコルの標準的な部分であるにもかかわらず、断片化されたパケットと断片化されていないパケットを区別することが望ましいです。

  当初は、IP パケットが異なるフレームサイズを持つネットワークで通過できるように設計されました。これらの日数の断片化は、マル形式のパケットを使用して DoS 攻撃を生成するためにより一般的に使用されます。また、IPv6 では断片化を完全に許可しない点にも留意してください。

- *-i - eth0 や ppp0 などの受信ネットワークインターフェイスを設定します。iptables では、これは filter テーブルと PREROUTING チェーンと nat テーブルおよび mangle テーブルと使用すると、INI および FORWARD チェーンでのみ使用できます。*

  このパラメーターは、以下の特別なオプションもサポートします。

  - *感嘆符文字(!)- ディレクティブを逆にすると、指定したインターフェイスはこのルールから除外されます。*

  - *プラス文字(+)- 指定の文字列に一致するすべてのインターフェイスに一致するために使用されるワイルドカード文字。たとえば、パラメーター -i eth+ は、このルールを任意のイーサネットインターフェイスに適用しますが、ppp0 などの他のインターフェイスを除外します。*

  -i パラメーターが使用されていてもインターフェイスが指定されていない場合は、すべてのインターフェイスがルールに影響されます。

- *-j - パケットが特定のルールに一致する場合に、指定したターゲットにジャンプします。*

標準のターゲットは ACCEPT、DROP、QUEUE、および RETURN です。

拡張オプションは、Red Hat Enterprise Linux iptables RPM パッケージでデフォルトでロードされているモジュールからも利用できます。これらのモジュールの有効なターゲットには、LOG、MARK、および REJECT が含まれます。これらのターゲットおよびその他のターゲットの詳細は、iptables の man ページを参照してください。

このオプションを使用して、特定のルールに一致するパケットを現在のチェーン外のユーザー定義のチェーンに転送し、他のルールをパケットに適用することもできます。

ターゲットが指定されていない場合、パケットはアクションを実行せずにルールに移動します。ただし、このルールのカウンターは 1 つずつ増えます。

- *-o - ルールの送信ネットワークインターフェイスを設定します。このオプションは、filter テーブルの OUTPUT および FORWARD チェーンにのみ有効です。また、nat テーブルおよび mangle テーブルの POSTROUTING チェーンにのみ有効です。このパラメーターは、受信ネットワークインターフェイスパラメーター(-i)と同じオプションを受け入れます。*

- *-p <protocol>: ルールの影響を受ける IP プロトコルを設定します。これは、icmp、tcp、udp、または all のいずれか、またはこれらのプロトコルの 1 つを表す数値にすることもできます。/etc/protocols ファイルに記載されているプロトコルを使用することもできます。*

  "all" プロトコルは、ルールがサポートされるすべてのプロトコルに適用されることを意味します。このルールにプロトコルが一覧にない場合は、デフォルトで "all" に設定されます。

- *-s: 宛先(-d)パラメーターと同じ構文を使用して、特定のパケットのソースを設定します。*

48.9.3.4. iptables の一致オプション

異なるネットワークプロトコルは、そのプロトコルを使用して特定のパケットに一致するように設定できる特殊なマッチングオプションを提供します。ただし、最初に iptables コマンドでプロトコルを指定する必要があります。たとえば、-p < protocol-name > は指定されたプロトコルのオプションを有効にします。プロトコル名の代わりにプロトコル ID を使用することもできます。以下の例を参照してください。それぞれの効果は同じです。

```
iptables -A INPUT -p icmp --icmp-type any -j ACCEPT
iptables -A INPUT -p 5813 --icmp-type any -j ACCEPT
```

サービス定義は /etc/services ファイルにあります。読みやすくするために、ポート番号ではなくサービス名を使用することが推奨されます。

重要な影響

/etc/services ファイルを保護し、承認されていない編集を防ぎます。このファイルを編集すると、クラッカーがそれを使用して、閉じたマシン上でポートを有効にすることができます。このファイルのセキュリティーを保護するには、root で以下のコマンドを入力します。

```
chown root.root /etc/services
chmod 0644 /etc/services
chattr +i /etc/services
```

これにより、ファイルの名前が変更されたり、削除したり、リンクを作成したりできなくなります。

48.9.3.4.1. TCP プロトコル

これらの一致オプションは TCP プロトコル(-p tcp)で利用できます。

- --dport - パケットの宛先ポートを設定します。

  このオプションを設定するには、ネットワークサービス名(www、smtp など)、ポート番号、またはポート番号の範囲を使用します。

  ポート番号の範囲を指定するには、2 つの数字をコロン(:)で区切ります。例： -p tcp --dport 3000:3200許容可能な最大有効な範囲は 0:65535 です。

  --dport オプションの後に感嘆符(!)を使用して、そのネットワークサービスまたはポートを使用し ない すべてのパケットに一致します。

  ネットワークサービスの名前とエイリアスと、それらが使用するポート番号を参照するには、/etc/services ファイルを表示します。

--destination-port match オプションは、--dport と同義です。

- *--sport - --dport と同じオプションを使用して、パケットの送信元ポートを設定します。--source-port match オプションは、--sport と同義です。*

- *--sYN - 一般に SYN パケット と呼ばれる通信を開始するように設計されたすべての TCP パケット に適用されます。データペイロードを伝送するパケットには影響はありません。*

  --syn オプションの後に感嘆符(!)を使用して、すべての非 SYN パケットに一致します。

- *--tcp-flags <tested flag list> <set flag list > - ルールに一致する特定のビット（フラグ）が設定された TCP パケットを許可します。*

  --tcp-flags match オプションは、2 つのパラメーターを受け入れます。最初のパラメーターはマスクで、パケット内で検査されるフラグのコンマ区切りリストです。2 番目のパラメーターは、ルールが一致するように設定する必要があるフラグのコンマ区切りリストです。

  使用できるフラグは以下のとおりです。

  - *ACK*
  - *FIN*
  - *PSH*
  - *RST*
  - *SYN*
  - *URG*

- *ALL*
- *NONE*

たとえば、以下の仕様を含む iptables ルールは、SYN フラグが設定され、ACK フラグおよび FIN フラグが設定されていない TCP パケットのみに一致します。

--tcp-flags ACK,FIN,SYN SYN

--tcp-flags の後にある感嘆符(!)を使用して、一致オプションの影響を元に戻します。

- *--tcp-option - 特定のパケット内で設定できる TCP 固有のオプションと一致しようとします。この一致オプションは、感嘆符(!)で逆にすることもできます。*

48.9.3.4.2. UDP プロトコル

これらの一致オプションは UDP プロトコル(-p udp)で利用できます。

- *--dport - サービス名、ポート番号、またはポート番号の範囲を使用して、UDP パケットの宛先ポートを指定します。--destination-port match オプションは、--dport と同義です。*
- *--sport - サービス名、ポート番号、またはポート番号の範囲を使用して、UDP パケットのソースポートを指定します。--source-port match オプションは、--sport と同義です。*

--dport オプションおよび --sport オプションでは、ポート番号の範囲を指定するには、2 つの数字をコロン(:)で区切ります。例： -p tcp --dport 3000:3200許容可能な最大有効な範囲は 0:65535 です。

48.9.3.4.3. ICMP プロトコル

Internet Control Message Protocol (ICMP) (-p icmp)では、次の一致オプションを使用できます。

- *--ICMP-type - ルールに一致する ICMP タイプの名前または番号を設定します。iptables*

-p icmp -h コマンドを入力して、有効な ICMP 名の一覧を取得できます。

48.9.3.4.4. 追加の一致オプションモジュール

他の一致オプションは、iptables コマンドでロードされるモジュールで利用できます。

一致オプションモジュールを使用するには、-m <module-name> を使用して 名前でモジュール を読み込みます。< module-name > はモジュールの名前になります。

デフォルトでは、多くのモジュールが利用できます。モジュールを作成して、追加機能を提供することもできます。

最も一般的に使用されるモジュールの一部のリストを以下に示します。

- *limit module: 特定のルールに一致するパケット数を制限します。*

  LOG ターゲットと併用すると、limit モジュールにより、一致するパケットが繰り返しメッセージまたはアップシステムリソースでシステムログがいっぱいになるのを防ぐことができます。

  LOG ターゲットの詳細については、「ターゲットオプション」 を参照してください。

  limit モジュールは、以下のオプションを有効にします。

  - *--limit - < value>/<period> ペアとして指定される特定の期間の 最大一致数を設定します。たとえば、--limit 5/hour を使用すると、1 時間あたり 5 つのルール一致が許可されます。*

    期間は、秒、分、時間、または日で指定できます。

    数字と時間修飾子が使用されていない場合、デフォルト値の 3/hour が想定されます。

  -

--limit-burst - 一度にルールに一致することのできるパケット数に制限を設定します。

このオプションは整数として指定され、--limit オプションと併用する必要があります。

値の指定がない場合は、デフォルト値の 5 (5)が想定されます。

- *state モジュール - 状態の一致を有効にします。*

  state モジュールは、以下のオプションを有効にします。

  - *--state - 次の接続状態のパケットを照合します。*

    - *ESTABLISHED - 一致するパケットは、確立された接続内の他のパケットに関連付けられます。クライアントとサーバー間の接続を維持する場合は、この状態を受け入れる必要があります。*
    - *INVALID - 一致するパケットを既知の接続に関連付けることはできません。*
    - *NEW - 一致するパケットは、新しい接続を作成するか、以前に確認されていない双方向接続の一部です。サービスへの新しい接続を許可する場合は、この状態を受け入れる必要があります。*
    - *RELATED: 一致するパケットは、既存の接続への何らかの方法で関連する新しい接続を開始します。FTP の例として、制御トラフィック（ポート 21）に 1 つの接続を使用し、データ転送に別の接続（ポート 20）を使用します。*

これらの接続状態は、-m state --state INVALID,NEW などのコンマで区切って、相互に組み合わせることができます。

- *MAC モジュール - ハードウェアの MAC アドレス一致を有効にします。*

mac モジュールは、以下のオプションを有効にします。

- *--mac-source - パケットを送信するネットワークインターフェイスカードの MAC アドレスと一致します。ルールから MAC アドレスを除外するには、--mac-source match オプションの後に感嘆符(!)を追加します。*

モジュールで利用可能な他の一致オプションについては、iptables の man ページを参照してください。

48.9.3.5. ターゲットオプション

パケットが特定のルールにマッチすると、ルールはパケットを、適切なアクションを決定する多数の異なるターゲットに転送することができます。各チェーンにはデフォルトのターゲットがあります。このターゲットは、そのチェーンのルールがパケットにマッチしていない場合や、パケットにマッチするルールがない場合はターゲットを指定します。

標準ターゲットは次のとおりです。

- *<user-defined-chain > - テーブル内のユーザー定義のチェーン。ユーザー定義のチェーン名は一意である必要があります。このターゲットは、パケットを指定されたチェーンに渡します。*
- *ACCEPT - 宛先または別のチェーンへのパケットを許可します。*
- *DROP - リクエスターに応答せずにパケットをドロップします。パケットを送信したシステムは、失敗について通知されません。*
- *QUEUE - パケットは、ユーザー空間のアプリケーションによって処理されるためにキューに置かれます。*
- *RETURN: 現在のチェーンのルールに対するパケットの確認を停止します。RETURN ターゲットを持つパケットが、別のチェーンから呼び出されたチェーン内のルールと一致する場合、パケットは最初のチェーンに返され、停止した場所の確認を再開します。RETURN ルールが組み込みチェーンで使用され、そのパケットが以前のチェーンに移動できない場合は、現在のチェーンのデフォルトターゲットが使用されます。*

さらに、他のターゲットを指定できるようにする拡張機能を利用できます。これらの拡張機能はターゲットモジュールまたは一致オプションモジュールと呼ばれ、そのほとんどは特定のテーブルおよび状況にのみ適用されます。一致オプションモジュールの詳細は、「追加の一致オプションモジュール」 を参照してください。

多くの拡張ターゲットモジュールが存在しますが、そのほとんどは特定のテーブルまたは状況にのみ適用されます。Red Hat Enterprise Linux にデフォルトで含まれている最も一般的なターゲットモジュールの一部は次のとおりです。

- LOG - このルールに一致するパケットをすべてログに記録します。パケットはカーネルによってログに記録されるため、/etc/syslog.conf ファイルはこれらのログエントリーが書き込まれる場所を決定します。デフォルトでは、それらは /var/log/messages ファイルに配置されます。

  LOG ターゲットの後に追加オプションを使用すると、ロギングが発生する方法を指定できます。

  - --log-level - ロギングイベントの優先度レベルを設定します。優先度レベルの一覧は、syslog.conf の man ページを参照してください。
  - --log-ip-options - IP パケットのヘッダーに設定されたオプションをログに記録します。
  - --log-prefix - 書き込み時にログ行の前に最大 29 文字の文字列を配置します。これは、パケットロギングと併用する syslog フィルターを作成するのに役立ちます。

    注記

    このオプションの問題により、末尾のスペースを log-prefix 値に追加する必要があります。

  - --log-tcp-options - TCP パケットのヘッダーに設定されたオプションをログに記録します。
  - --log-tcp-sequence - パケットの TCP シーケンス番号をログに書き込みます。

- *REJECT - エラーパケットをリモートシステムに送信し、パケットをドロップします。*

  REJECT ターゲットは --reject-with < type > (< type > は rejection タイプ)を受け入れます。これにより、エラーパケットとともにより詳細な情報が返されます。メッセージ ポート到達不能 は、他のオプションが使用されていない場合に指定されるデフォルトのエラータイプです。< type> オプションの完全なリストは、iptables の man ページを参照し てください。

nat テーブルを使用した IP マスカレード、または mangle テーブルを使用したパケット変更に便利な複数のターゲット拡張機能は、iptables の man ページにあります。

48.9.3.6. オプションの一覧表示

デフォルトのリストコマンド iptables -L [<chain-name >] は、デフォルトのフィルターテーブルの現在のチェーンに関する非常に基本的な概要を提供します。その他のオプションでは、以下に関する詳細情報が提供されます。

- *-v: 各チェーンが処理したパケット数やバイト数、各ルールが一致したパケットとバイト数、特定のルールに適用されるインターフェイスなど、詳細な出力を表示します。*

- *-x: 番号を正確な値に展開します。ビジーなシステムでは、特定のチェーンまたはルールによって処理されるパケットとバイト数の数は、キロバイト、メガバイト （メガバイト）、またはギガバイト（ギガバイト）と省略でき ます。このオプションは、完全な番号を強制的に表示します。*

- *-n - デフォルトのホスト名およびネットワークサービス形式ではなく、数値形式で IP アドレスとポート番号を表示します。*

- *--line-numbers - チェーンの数値順の横にある各チェーンのルールを一覧表示します。このオプションは、チェーン内の特定のルールを削除しようとする場合や、チェーン内にルールを挿入する場所を見つける場合に役立ちます。*

- *-t <table-name> - テーブル名を指定します。省略した場合、デフォルトは filter テーブルに設定されます。*

以下の例は、これらのオプションのいくつかの使用方法を示しています。-x オプションを追加して、バイト表示の違いに注意してください。

-

```
~]# iptables -L OUTPUT -v -n -x
Chain OUTPUT (policy ACCEPT 64005 packets, 6445791 bytes)
    pkts      bytes target     prot opt in     out     source               destination
    1593   133812 ACCEPT     icmp --  *      *       0.0.0.0/0            0.0.0.0/0

~]# iptables -L OUTPUT -v -n
Chain OUTPUT (policy ACCEPT 64783 packets, 6492K bytes)
    pkts bytes target     prot opt in     out     source               destination
    1819  153K ACCEPT     icmp --  *      *       0.0.0.0/0            0.0.0.0/0
~]#
```

48.9.4. IPTables ルールの保存

iptables コマンドで作成されたルールはメモリーに保存されます。iptables ルールセットを保存する前にシステムが再起動すると、すべてのルールが失われます。filter ルールがシステム再起動後も維持されるようにするには、それらを保存する必要があります。filter ルールを保存するには、root で以下のコマンドを入力します。

```
service iptables save
```

これにより、/sbin/ iptables -save プログラムを実行し、現在の iptables 設定を /etc/sysconfig/iptables に書き込む iptables init スクリプトを実行します。既存の /etc/sysconfig/iptables ファイルは /etc/sysconfig/iptables.save として保存されます。

次にシステムを起動すると、iptables init スクリプトは、/sbin/iptables-restore コマンドを使用して、/etc/sysconfig/iptables に保存されているルールを再適用します。

/etc/sysconfig/iptables ファイルにコミットする前に新しい iptables ルールをテストすることが推奨されますが、iptables ルールをこのファイルのバージョンから別のシステムのバージョンからこのファイルにコピーすることができます。これにより、iptables ルールのセットを複数のマシンに簡単に配信できます。

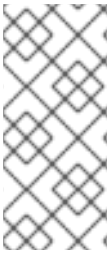
また、iptables ルールを、ディストリビューション、バックアップ、またはその他の目的で別のファイルに保存することもできます。iptables ルールを保存するには、root で以下のコマンドを入力します。

```
iptables-save > <filename>
```

& lt;filename& gt; は、ルールセットのユーザー定義名です。

重要な影響

/etc/sysconfig/iptables ファイルを他のマシンに配布する場合は、/sbin/service iptables restart と入力して新しいルールを有効にします。



注記

iptables 機能を設定するテーブルとチェーンの操作に使用される iptables コマンド (/sbin/iptables)と、iptables サービス自体を有効または無効にする iptables サービス (/sbin/iptables サービス)の違いに注意してください。

48.9.5. iptables 制御スクリプト

Red Hat Enterprise Linux では、iptables を制御する基本的な方法が 2 つあります。

- *Security Level Configuration Tool (system-config-securitylevel): 基本的なファイアウォールルールを作成、アクティベート、および保存するグラフィカルインターフェイスです。詳細は、「ファイアウォールの基本設定」 を参照してください。*

- */sbin/service iptables < option>: initscript を使用して iptables のさまざまな機能を操作するのに使用します。以下のオプションを設定できます。*

  - *start - ファイアウォールが設定されている場合（つまり /etc/sysconfig/iptables が存在する場合）、実行中の iptables はすべて完全に停止され、/sbin/iptables-restore コマンドを使用して起動します。このオプションは、ipchains カーネルモジュールがロードされていない場合にのみ機能します。このモジュールが読み込まれているかどうかを確認するには、root で以下のコマンドを入力します。*

    ```
    lsmod | grep ipchains
    ```

    このコマンドで出力が返されない場合は、モジュールが読み込まれていないことを意味します。必要に応じて、/sbin/rmmod コマンドを使用してモジュールを削除します。

  - *stop - ファイアウォールが実行されている場合、メモリーのファイアウォールルールがフラッシュされ、すべての iptables モジュールとヘルパーがアンロードされます。*

    /etc/sysconfig/iptables-config 設定ファイルの IPTABLES_SAVE_ON_STOP ディレクティブがデフォルト値から yes に変更されると、現在のルールは

/etc/sysconfig/iptables に保存され、既存のルールは /etc/sysconfig/iptables.save ファイルに移動します。

iptables-config ファイルの詳細は、「iptables 制御スクリプト設定ファイル」 を参照してください。

- restart: ファイアウォールが実行されている場合、メモリーのファイアウォールルールがフラッシュされ、/etc/sysconfig/iptables で設定されている場合はファイアウォールが再度起動します。このオプションは、ipchains カーネルモジュールがロードされていない場合にのみ機能します。

  /etc/sysconfig/iptables-config 設定ファイルの IPTABLES_SAVE_ON_RESTART ディレクティブがデフォルト値から yes に変更されると、現在のルールは /etc/sysconfig/iptables に保存され、既存のルールは /etc/sysconfig/iptables.save ファイルに移動します。

  iptables-config ファイルの詳細は、「iptables 制御スクリプト設定ファイル」 を参照してください。

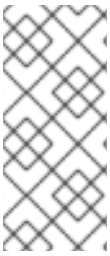
- ステータス - ファイアウォールのステータスを表示し、すべてのアクティブなルールを一覧表示します。

  このオプションのデフォルト設定では、各ルールの IP アドレスが表示されます。ドメインおよびホスト名の情報を表示するには、/etc/sysconfig/iptables-config ファイルを編集し、IPTABLES_STATUS_NUMERIC の値を no に変更します。iptables-config ファイルの詳細は、「iptables 制御スクリプト設定ファイル」 を参照してください。

- panic - すべてのファイアウォールルールを表示します。設定されたすべてのテーブルのポリシーは DROP に設定されます。

  このオプションは、サーバーが危険にさらされることがわかっている場合に役立ちます。ネットワークから物理的に切断したり、システムをシャットダウンしたりするのではなく、このオプションを使用して、それ以降のネットワークトラフィックをすべて停止できますが、分析またはその他のフォレンジック用にマシンを状態にしておくことができます。

- Save: iptables- save を使用して、ファイアウォールルールを /etc/sysconfig/iptables に保存 します。詳細は、「IPTables ルールの保存」 を参照してください。



ヒント

同じ initscript コマンドを使用して IPv66 のfilter を制御するには、このセクションに記載されている /sbin/service コマンドの ip6tables を iptables に置き換えます。IPv6 およびfilter の詳細は、「iptables および IPv6」 を参照してください。

48.9.5.1. iptables 制御スクリプト設定ファイル

iptables initscripts の動作は、/etc/sysconfig/iptables-config 設定ファイルによって制御されます。以下は、このファイルに含まれるディレクティブの一覧です。

- IPTABLES_MODULES - ファイアウォールがアクティブ化されたときにロードする追加の iptables モジュールのスペース区切りの一覧を指定します。これには、接続追跡および NAT ヘルパーを含めることができます。

- IPTABLES_MODULES_UNLOAD - 再起動および停止時にモジュールをアンロードします。このディレクティブは、以下の値を受け入れます。

  - ◯: デフォルト値。このオプションは、ファイアウォールの再起動または停止について正しい状態を実現するように設定する必要があります。

  - no: このオプションは、netfilter モジュールのアンロードに問題がある場合にのみ設定する必要があります。

- IPTABLES_SAVE_ON_STOP: ファイアウォールが停止したときに現在のファイアウォールルールを /etc/sysconfig/iptables に保存します。このディレクティブは、以下の値を受け入れます。

  - Yes: ファイアウォールが停止したときに既存のルールを /etc/sysconfig/iptables に保存し、以前のバージョンを /etc/sysconfig/iptables.save ファイルに移動します。

  - no: デフォルト値は です。ファイアウォールが停止している場合、既存のルールを保存しません。

- IPTABLES_SAVE_ON_RESTART - ファイアウォールの再起動時に現在のファイアウォールルールを保存します。このディレクティブは、以下の値を受け入れます。

- Yes: ファイアウォールが再起動されたときに既存のルールを /etc/sysconfig/iptables に保存し、以前のバージョンを /etc/sysconfig/iptables.save ファイルに移動します。

- no: デフォルト値は です。ファイアウォールが再起動されても、既存のルールを保存しません。

- IPTABLES_SAVE_COUNTER - すべてのチェーンおよびルールですべてのパケットカウンターおよびバイトカウンターを保存し、復元します。このディレクティブは、以下の値を受け入れます。

  - yes - カウンター値を保存します。

  - no: デフォルト値は です。カウンター値を保存しません。

- IPTABLES_STATUS_NUMERIC - ドメインまたはホスト名ではなく、数値形式で IP アドレスを出力します。このディレクティブは、以下の値を受け入れます。

  - ◯: デフォルト値。ステータス出力内の IP アドレスのみを返します。

  - no: ステータス出力内のドメインまたはホスト名を返します。

48.9.6. iptables および IPv6

iptables-ipv6 パッケージがインストールされている場合は、Red Hat Enterprise Linux のnetfilter は次回の世代の IPv6 インターネットプロトコルをフィルターリングできます。IPv6filter の操作に使用するコマンドは ip6tables です。

このコマンドのほとんどのディレクティブは iptables に使用されるディレクティブと同じですが、nat テーブルはまだサポートされていません。つまり、マスカレードやポート転送などの IPv6 ネットワークアドレス変換タスクを実行することはできません。

ip6tables のルールは、/etc/sysconfig/ip6tables ファイルに保存されます。ip6tables initscripts が保存した以前のルールは、/etc/sysconfig/ip6tables.save ファイルに保存されます。

ip6tables init スクリプトの設定オプションは /etc/sysconfig/ip6tables-config に保存され、各ディレクティブの名前は iptables で若干異なります。

たとえば、iptables-config ディレクティブ IPTABLES_MODULES: ip6tables-config ファイルの同等は IP6TABLES_MODULES です。

48.9.7. 関連情報

iptables を使用したパケットフィルターリングの詳細は、以下のソースを参照してください。

- 「ファイアウォール」: 全体的なセキュリティーストラテジーにおけるファイアウォールロールに関する章と、ファイアウォールルールを構築するストラテジーについて説明します。

48.9.7.1. インストールされているドキュメント

- man iptables: iptables の説明と、ターゲット、オプション、および一致拡張機能の包括的な一覧が含まれます。

48.9.7.2. 便利な Web サイト

- http://www.netfilter.org/ (netfilter/iptables プロジェクトのホーム)。特定問題に対応する FAQ や Linux IP ファイアウォールメンテナー(Russell)が役立つさまざまなガイドなど、iptables に関するさまざまな情報が含まれています。サイトの HOWTO ドキュメントは、基本的なネットワーク概念、カーネルパケットフィルターリング、NAT 設定などのサブジェクトを対象としています。

- http://www.linuxnewbie.org/nhf/Security/IPtables_Basics.html: パケットが Linux カーネルを通過する方法の概要と、基本的な iptables コマンドの構築を紹介します。

---

[14] システム BIOS はメーカーによって異なるため、いずれかのタイプのパスワード保護のみをサポートするものもあれば、いずれのタイプのパスワード保護もサポートしないものもあります。

[15] GRUB は暗号化されていないパスワードも受け入れますが、セキュリティーを強化するために MD5 ハッシュを使用することが推奨されます。

[16]
このアクセスは、SELinux の制限が有効な場合は、この制限対象となります。

[17]
クライアントとサーバーの両方が共通の鍵を共有しているシステム。これは、ネットワーク通信の暗号化と復号化に使用されます。

第49章 セキュリティーおよび SELINUX

49.1. アクセス制御メカニズム(ACM)

このセクションでは、アクセス制御メカニズム (ACM)の基本的な概要を説明します。ACMは、システム管理者がコンピューターシステム内の異なるファイル、デバイス、インターフェイスなどにアクセスできるユーザーとプロセスを制御する手段を提供します。これは、コンピューターシステムまたは任意のサイズのネットワークのセキュリティーを保護する場合の主な考慮事項です。

49.1.1. Discretionary Access Control (DAC)

Discretionary Access Control (DAC)は、ファイルシステム内のオブジェクトに対する基本的なアクセス制御を定義します。これは、ファイルパーミッションや共有などによって提供される一般的なアクセス制御です。このようなアクセスは通常、オブジェクト（ファイル、ディレクトリー、デバイスなど）の所有者の判断で提供されます。

DAC は、これらのオブジェクトにアクセスしようとするユーザーまたはグループ（サブジェクト）の ID に基づいてオブジェクトへのアクセスを制限する手段を提供します。サブジェクトのアクセスパーミッションによっては、他のサブジェクトにパーミッションを渡すこともできます。

49.1.2. アクセス制御リスト(ACL)

アクセス制御リスト (ACL)は、サブジェクトがアクセスできるオブジェクトをさらに制御できます。詳細は、10章アクセス制御リスト を参照してください。

49.1.3. 強制アクセス制御(MAC)

強制アクセス制御 (MAC)は、ユーザー（サブジェクト）が作成するオブジェクトに対する制御のレベルを制限するセキュリティーメカニズムです。DAC 実装とは異なり、ユーザーは独自のファイル、ディレクトリーなどを完全に制御し、MAC はすべてのファイルシステムオブジェクトにラベルまたはカテゴリーを追加します。ユーザーおよびプロセスは、これらのオブジェクトと対話する前にこれらのカテゴリーに適切なアクセスを持っている必要があります。

Red Hat Enterprise Linux では、MAC は SELinux によって強制されます。詳細は、「SELinux の概要」 を参照してください。

49.1.4. ロールベースアクセス制御(RBAC)

ロールベースアクセス制御(RBAC)は、ファイルシステムオブジェクトへのユーザーアクセスを制御する代替方法です。ユーザーのパーミッションでアクセスを制御する代わりに、システム管理者はビ

ジネス機能の要件または同様の基準に基づいて ロール を確立します。これらのロールには、オブジェクトへのさまざまなタイプとレベルのアクセスがあります。

DAC または MAC システムとは対照的に、ユーザーは自分のオブジェクトとオブジェクトのパーミッションに基づいてオブジェクトにアクセスできるのに対し、RBAC システムのユーザーは、ファイル、ディレクトリー、デバイスなどと対話する前に適切なグループまたはロールのメンバーである必要があります。

管理の観点からは、グループメンバーシップを制御するだけで、ファイルシステムのさまざまな部分にアクセスできるユーザーをより簡単に制御できます。

49.1.5. Multi-Level Security (MLS)

Multi-Level Security (MLS)は、特定のMAC(Mandatory Access Control)セキュリティースキームです。このスキームでは、プロセスは Subjects と呼ばれます。ファイル、ソケット、およびその他のパッシブオペレーティングシステムのエンティティーは オブジェクト と呼ばれます。詳細は、「Multi-Level Security (MLS)」 を参照してください。

49.1.6. Multi-Category Security (MCS)

Multi-Category Security (MCS)は SELinux の拡張機能であり、ユーザーはカテゴリーでファイルにラベルを付けることができます。MCS は MLS の調整であり、SELinux の MLS フレームワークの多くを再利用します。詳細は、を参照してください。 「はじめに」

49.2. SELINUX の概要

Security-Enhanced Linux (SELinux)は、Linux Security Modules (LSM)を使用して 2.6.x カーネルに統合されたセキュリティーアーキテクチャーです。これは、米国国家安全保障局(NSA)および SELinuxコミュニティーのプロジェクトです。Red Hat Enterprise Linux への SELinux の統合は、NSA と Red Hat との間の共同作業でした。

49.2.1. SELinux の概要

SELinux は、Linux カーネルに組み込まれた柔軟な 強制アクセス制御 (MAC)システムを提供します。標準の Linux Discretionary Access Control (DAC)では、ユーザー(UID または SUID)として実行されているアプリケーションまたはプロセスには、ファイル、ソケット、その他のプロセスなどのオブジェクトに対するユーザーのパーミッションがあります。MAC カーネルを実行すると、システムの破損や破棄が可能な悪意のあるアプリケーションや不具合のあるアプリケーションからシステムが保護されます。

SELinux は、システム上のすべてのユーザー、アプリケーション、プロセス、およびファイルへのアクセスと遷移の権限を定義します。次に、SELinux は、特定の Red Hat Enterprise Linux インストールを厳格または盗まれる方法を指定するセキュリティーポリシーを使用して、これらのエンティティーの対話を管理します。

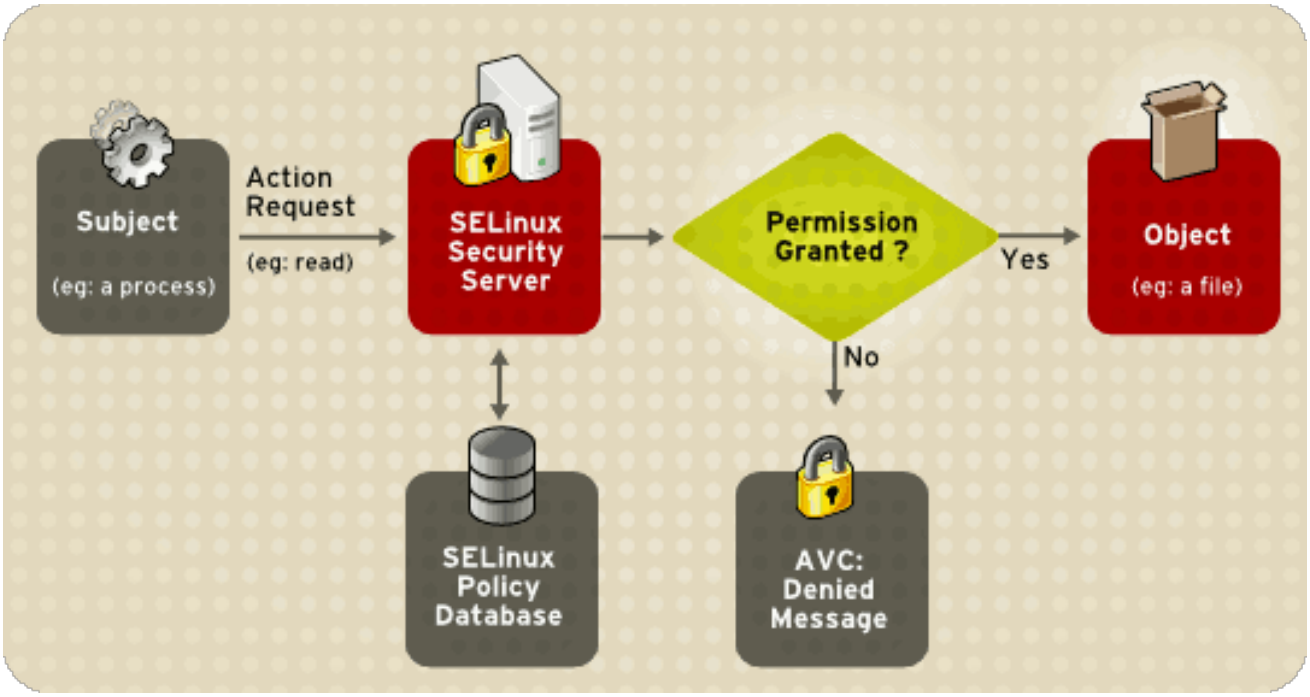
日常的には、システムユーザーは主に SELinux を認識しません。システム管理者のみが、サーバー環境に対して実装するポリシーをどの程度厳格にするかを考慮する必要があります。ポリシーは、必要に応じて strict または lenient として指定でき、非常に詳細です。この詳細により、SELinux カーネルがシステム全体を完全に詳細に制御できます。

SELinux 意思決定プロセス

サブジェクト（アプリケーションなど）がオブジェクトへのアクセスを試みると（例：ファイル）、カーネル内のポリシー強制サーバーは、サブジェクトとオブジェクトパーミッションが キャッシュされるアクセスベクターキャッシュ (AVC)をチェックします。AVC のデータに基づいて決定を行うことができない場合、要求はセキュリティーサーバーに続行され、アプリケーションとマトリックスでファイルを検索します。次に、パーミッションが許可または拒否され、パーミッションが拒否された場合は、/var/log/messages に詳細を示す avc: denied メッセージが表示されます。サブジェクトとオブジェクトのセキュリティーコンテキストはインストールされたポリシーから適用されます。これは、セキュリティーサーバーのマトリックスを設定するための情報を提供します。

以下の図を参照してください。

図49.1 SELinux デシジョンプロセス



[D]

SELinux の操作モード

SELinux は Enforcing モードで実行する代わりに Permissive モードで実行できます。この場

合、AVC が確認され、拒否のログが記録されますが、SELinux はポリシーを強制しません。これは、トラブルシューティングや SELinux ポリシーの開発や微調整に役立ちます。

SELinux の仕組みの詳細については、「関連情報」 を参照してください。

49.2.2. SELinux に関連するファイル

以下のセクションでは、SELinux 設定ファイルおよび関連するファイルシステムを説明します。

49.2.2.1. SELinux Pseudo-File System

/selinux/ 擬似ファイルシステムには、カーネルサブシステムで最も一般的に使用されるコマンドが含まれています。このタイプのファイルシステムは、/proc/ 擬似ファイルシステムに似ています。

通常、管理者およびユーザーはこのコンポーネントを操作する必要はありません。

以下の例は、/selinux/ ディレクトリーの内容を示しています。

```
-rw-rw-rw-  1 root root 0 Sep 22 13:14 access
dr-xr-xr-x  1 root root 0 Sep 22 13:14 booleans
--w-------  1 root root 0 Sep 22 13:14 commit_pending_bools
-rw-rw-rw-  1 root root 0 Sep 22 13:14 context
-rw-rw-rw-  1 root root 0 Sep 22 13:14 create
--w-------  1 root root 0 Sep 22 13:14 disable
-rw-r--r--  1 root root 0 Sep 22 13:14 enforce
-rw-------  1 root root 0 Sep 22 13:14 load
-r--r--r--  1 root root 0 Sep 22 13:14 mls
-r--r--r--  1 root root 0 Sep 22 13:14 policyvers
-rw-rw-rw-  1 root root 0 Sep 22 13:14 relabel
-rw-rw-rw-  1 root root 0 Sep 22 13:14 user
```

たとえば、enforce ファイルで cat コマンドを実行すると、Enforcing モードの場合は 1、Permissive モードの場合は 0 が表示されます。

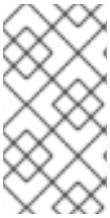
49.2.2.2. SELinux 設定ファイル

以下のセクションでは、SELinux の設定とポリシーファイル、および /etc/ ディレクトリーにある関連ファイルシステムを説明します。

49.2.2.2.1. /etc/sysconfig/selinux 設定ファイル

Red Hat Enterprise Linux で SELinux を設定する方法は、SELinux Administration Tool の使用 (system-config-selinux)、または設定ファイルを手動で編集する(/etc/sysconfig/selinux)の 2 つの方法があります。

/etc/sysconfig/selinux ファイルは、SELinux を有効または無効にする主要な設定ファイルであり、システムで強制するポリシーと、その強制方法を設定します。



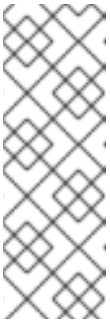
注記

/etc/sysconfig/selinux には、実際の設定ファイル /etc/selinux/config へのシンボリックリンクが含まれます。

以下は、設定で利用可能なオプションの完全なサブセットについて説明しています。

- *SELINUX=enforcing|permissive|disabled: システム上の SELinux の最上位状態を定義します。*
  - *Enforcing - SELinux セキュリティーポリシーが適用されます。*
  - *Permissive - SELinux システムは警告を出力しますが、ポリシーは強制されません。*

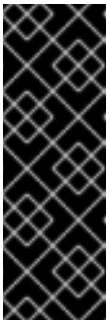
    これは、デバッグおよびトラブルシューティングに役立ちます。
  -



ヒント

注記

-
  -



重要な影響

```
setsebool -P dhcpd_disable_trans=0
```

  -
-

49.2.2.2.2.

```
-rw-r--r--  1 root root  448 Sep 22 17:34 config
drwxr-xr-x  5 root root 4096 Sep 22 17:27 strict
drwxr-xr-x  5 root root 4096 Sep 22 17:28 targeted
```

49.2.2.3.

-

  以下に例を示します。

-

```
SELinux status:                 enabled
SELinuxfs mount:                /selinux
Current mode:                   enforcing
Mode from config file:          enforcing
Policy version:                 21
Policy from config file:        targeted

Process contexts:
```

```
Current context:        user_u:system_r:unconfined_t:s0
Init context:           system_u:system_r:init_t:s0
/sbin/mingetty          system_u:system_r:getty_t:s0
```

-

注記

-

-

rpm -ql <package-name>

49.2.3. 関連情報

49.2.3.1. インストールされているドキュメント

-

49.2.3.2. 便利な Web サイト

-

-

-

49.3.

[18] [19]

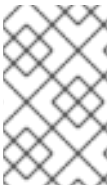
49.4. MULTI-CATEGORY SECURITY (MCS)

49.4.1. はじめに

49.4.1.1.

49.4.2.

49.4.3.



注記

```
~]# ls -Z gravityControl.txt
-rw-r--r--  user     user     user_u:object_r:tmp_t:Moonbase_Plans gravityControl.txt
```

```
~]# getfattr -n security.selinux gravityControl.txt
# file: gravityControl.txt
security.selinux="user_u:object_r:tmp_t:s0:c10\000"
```

49.5.

49.5.1. はじめに

49.5.2.

-
-
-

```
~]# semanage user -l

                Labeling   MLS/       MLS/
SELinux User    Prefix     MCS Level  MCS Range            SELinux Roles

root            user       s0         s0-s0:c0.c1023       system_r sysadm_r user_r
system_u        user       s0         s0-s0:c0.c1023       system_r
user_u          user       s0         s0-s0:c0.c1023       system_r sysadm_r user_r
```

SELinux Logins

```
~]# semanage login -a james
~]# semanage login -a daniel
~]# semanage login -a olga
```

```
~]# semanage login -l

Login Name           SELinux User         MLS/MCS Range

__default__          user_u               s0
james                user_u               s0
daniel               user_u               s0
root                 root                 s0-s0:c0.c1023
olga                 user_u               s0
```

49.5.3.

```
~]# chcat -L
s0
s0-s0:c0.c1023              SystemLow-SystemHigh
s0:c0.c1023                 SystemHigh
```

```
~]# vi /etc/selinux/targeted/setrans.conf
s0:c0=Marketing
s0:c1=Finance
s0:c2=Payroll
s0:c3=Personnel
```

```
~]# chcat -L
s0:c0                       Marketing
s0:c1                       Finance
```

```
s0:c2                   Payroll
s0:c3                   Personnel
s0
s0-s0:c0.c1023              SystemLow-SystemHigh
s0:c0.c1023                 SystemHigh
```

注記

```
~]# service mcstrans restart
```

49.5.4.

```
~]# chcat -l -- +Marketing james
~]# chcat -l -- +Finance,+Payroll daniel
~]# chcat -l -- +Personnel olga
```

```
~]# chcat -L -l daniel james olga
daniel: Finance,Payroll
james: Marketing
olga: Personnel
```

```
# Create a user account for the company director (Karl)
~]# useradd karl
~]# passwd karl
Changing password for user karl.
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully.

# Assign the user account to an SELinux login
~]# semanage login -a karl
```

```
# Assign all the MCS categories to the new login
~]# chcat -l -- +Marketing,+Finance,+Payroll,+Personnel karl
```

```
~]# chcat -L -l daniel james olga karl
daniel: Finance,Payroll
james: Marketing
olga: Personnel
karl: Marketing,Finance,Payroll,Personnel
```

注記

49.5.5.

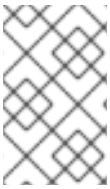
```
[daniel@dhcp-133 ~]$ echo "Financial Records 2006" > financeRecords.txt
```

```
[daniel@dhcp-133 ~]$ ls -Z financeRecords.txt
-rw-r--r--  daniel daniel user_u:object_r:user_home_t      financeRecords.txt
```

```
[daniel@dhcp-133 ~]$ chcat -- +Finance financeRecords.txt
[daniel@dhcp-133 ~]$ ls -Z financeRecords.txt
-rw-r--r--  daniel daniel root:object_r:user_home_t:Finance financeRecords.txt
```

```
[daniel@dhcp-133 ~]$ chcat -- +Payroll financeRecords.txt
[daniel@dhcp-133 ~]$ ls -Z financeRecords.txt
-rw-r--r--  daniel daniel root:object_r:user_home_t:Finance,Payroll financeRecords.txt
```

```
[olga@dhcp-133 ~]$ cat financeRecords.txt
cat: financeRecords.txt: Permission Denied
```



注記

49.6. MULTI-LEVEL SECURITY (MLS)

49.6.1.

データフローに適用されるルールは、レベルが低いレベルからより高いレベルに動作し、逆順は発生しません。以下で説明します。

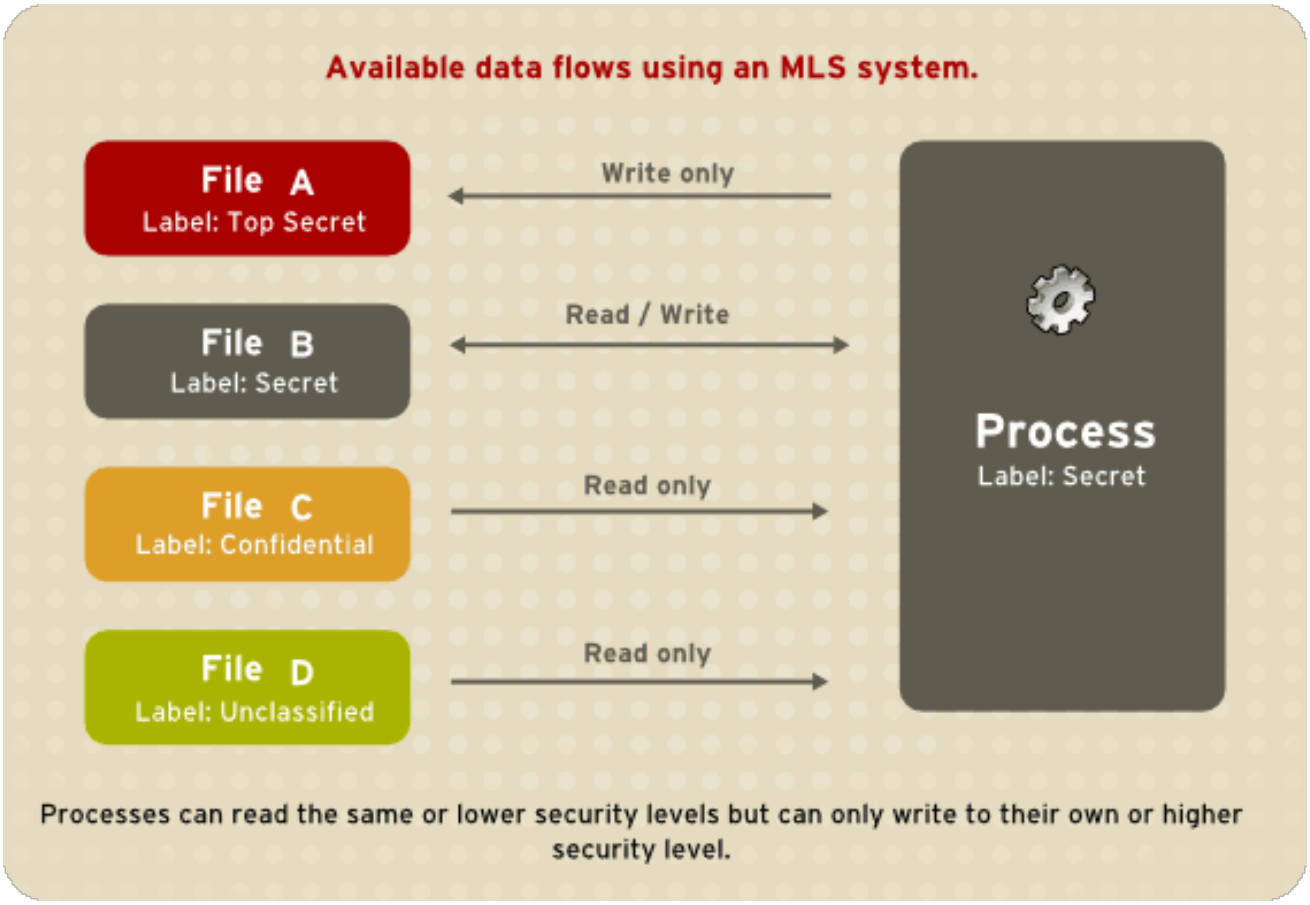
図49.2

[D]

49.6.1.1.

このモデルは、各サブジェクトおよびオブジェクトに割り当てられたラベルに基づいてシステム内で情報をフローする方法を指定します。

図49.3



[D]

49.6.1.2. MLS およびシステム権限

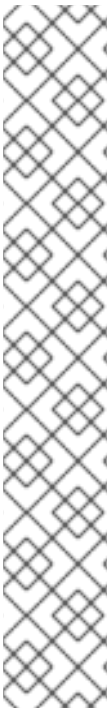
より高いセキュリティークリアランスを設定しても、ファイルシステムを任意にブラウズするパーミッションは自動的には付与されません。

トップレベルの許可があるユーザーは、マルチレベルのシステムで自動的に管理者権限を取得しません。コンピューターのすべての情報にアクセスできる場合もありますが、管理者権限を設定するのとは異なります。

49.6.2.

1.

2.



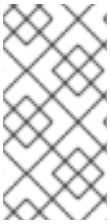
注記

1.

2.

49.6.3.

49.6.4. SELinux での MLS の有効化



注記

X Window System を実行しているシステムでは、MLS ポリシーを使用することは推奨されていません。

以下の手順に従って、システムで SELinux MLS ポリシーを有効にします。

1.

```
~]# yum install selinux-policy-mls
```

2. **MLS ポリシーを有効にする前に、ファイルシステムの各ファイルに、MLS ラベルで再ラベル付けする必要があります。ファイルシステムに再ラベル付けすると、制限されたドメインのアクセスが拒否される可能性があります。これにより、システムが正しく起動しなくなる可能性があります。これを防ぐには、/etc/selinux/config ファイルで SELINUX=permissive を設定します。また、SELINUXTYPE=mls を設定して、MLS ポリシーを有効にします。設定ファイルは以下のようになります。**

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=permissive
```

```
# SELINUXTYPE= can take one of these two values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected processes are protected.
#     mls - Multi Level Security protection.
SELINUXTYPE=mls
```

3.

```
~]# setenforce 0
~]# getenforce
Permissive
```

4.

```
~]# touch /.autorelabel
```

5. ***システムを再起動します。次回の起動時に、MLS ポリシーに従って、すべてのファイルシステムに再ラベル付けされます。ラベルプロセスは、適切な SELinux コンテキストを使用して、すべてのファイルにラベルを付けます。***

```
*** Warning -- SELinux mls policy relabel is required.
*** Relabeling could take a very long time, depending on file
*** system size and speed of hard drives.
***********
```

一番下の行にある * (アスタリスク) 文字は、ラベル付けされている 1000 ファイルを表します。すべてのファイルにラベルを付けるのにかかる時間は、システムのファイル数と、ハードディスクドライブの速度により異なります。最新のシステムでは、このプロセスに 10 分程度かかる場合があります。ラベリングプロセスが終了すると、システムが自動的に再起動します。

6.

```
~]# genhomedircon
~]# restorecon -R -v /root /home <other_home_directories>
```

7. ***Permissive モードでは SELinux ポリシーは強制されませんが、Enforcing モードで実行された場合に拒否されたであろうアクションの拒否は引き続きログに記録されます。最後のシステムの起動時に SELinux がアクションを拒否しなかった場合に、このコマンドを実行しても出力は返されません。***

8.

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=enforcing
# SELINUXTYPE= can take one of these two values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected processes are protected.
#     mls - Multi Level Security protection.
SELINUXTYPE=mls
```

9.

```
~]$ getenforce
Enforcing
```

MLS ポリシーが有効になっていることも確認します。

```
~]# sestatus |grep mls
Policy from config file:       mls
```

49.6.5.

49.7.

49.7.1.

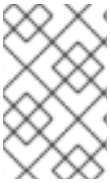
49.7.1.1.

49.7.1.1.1.

重要な影響

49.7.1.2.

49.7.2.



注記

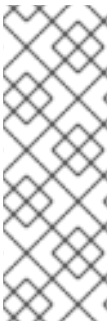
49.7.2.1.

-
-
-
-
-
-

注記

49.7.2.2.

```
man 3 selinux_binary_policy_path
```



注記

49.7.3.

1.
2.
3.
4.
5.

6.

7.

8.

49.7.4.

以下に例を示します。

- *ファイル関連のクラスには、ファイルシステムの **filesystem**、ファイルの **file**、ディレクトリーの **dir** が含まれます。各クラスには、独自の関連付けられたパーミッションセットがあります。*

- *これはパーミッションについても同様です。開発作業が継続され、クラスおよびパーミッションを動的に登録および登録解除できます。*

*パーミッションは、ポリシーが許可する場合に、サブジェクトがオブジェクトで実行できるアクションです。これらのパーミッションは、**SELinux** がアクティブに許可または拒否するアクセス要求で*

す。

49.8. ターゲットポリシーの概要

本章では、Red Hat Enterprise Linux の現在のサポート対象ポリシーである SELinux ターゲットポリシーの概要と試験を行います。

本章の内容の多くは、ファイルの場所とこれらのファイル内のコンテンツのタイプに関する、すべてのタイプの SELinux ポリシーに適用されます。違いは、どのファイルがキーの場所に存在するかとその内容にあります。

49.8.1. ターゲットポリシーとは

SELinux ポリシーは高度な設定が可能です。Red Hat Enterprise Linux 5 では、Red Hat はターゲットポリシーである単一のポリシー をサポートし て
います。targeted ポリシーでは、特定のターゲットデーモンを 除き、すべてのサブジェクトとオブジェクトは unconfined_t ドメインで実行されます。unconfined_t ドメインにあるオブジェクトには制限がなく、標準の Linux セキュリティー(DAC)の使用にフォールバックします。ターゲットポリシーの一部であるデーモンは、独自のドメインで実行され、システムで実行するすべての操作で制限されます。このようにして、悪用されたデーモンや侵害されたデーモンは、どのような方法でも含まれており、限られた損傷しか発生しない可能性があります。

たとえば、http デーモンと ntp デーモンは、デフォルトのターゲットポリシーで保護されており、それぞれ httpd_t ドメインと ntpd_t ドメインで実行されます。ただし、ssh デーモンはこのポリシーでは保護されないため、unconfined_t ドメインで実行されます。

以下の出力例を参照してください。これは、上記のデーモンのさまざまなドメインを示しています。

```
user_u:system_r:httpd_t        25129 ?        00:00:00 httpd
user_u:system_r:ntpd_t         25176 ?        00:00:00 ntpd
system_u:system_r:unconfined_t         25245 ? 00:00:00 sshd
```

Strict ポリシー

Targeted ポリシーの逆は、厳密なポリシー
です。strict ポリシーでは、すべてのサブジェクトとオブジェクトが特定のセキュリティードメインに存在し、すべての対話と移行はポリシールール内で個別に考慮されます。

strict ポリシーはより複雑な環境であり、Red Hat Enterprise Linux には同梱されていません。本ガイドでは、Red Hat Enterprise Linux に同梱されるターゲットポリシーと、ターゲットデーモンが使

用する SELinux のコンポーネントに重点を置いています。

対象のデーモンは、dhcpd; httpd; mysqld; named; nscd; ntpd; portmap; postgres; snmpd; squid; syslogd; winbind です。

注記

インストールによっては、これらのデーモンの一部のみが存在する可能性があります。

49.8.2. ターゲットされたポリシーのファイルおよびディレクトリー

SELinux が使用する一般的なファイルおよびディレクトリーの一覧は、「」 を参照してください。

49.8.3. ターゲットポリシー内のユーザーとロールについて

このセクションでは、targeted ポリシーで有効になっている特定のロールについて説明します。unconfined_t タイプはすべてのロールに存在するため、ターゲットポリシーでのロールの有用性が大幅に削減されます。ロールをより広範囲に使用するには、厳密なポリシーパラダイム（各プロセスが個別にドメインと見なされる）に変更する必要があります。

実際には、targeted ポリシーには system_r と object_r の 2 つのロールのみがあります。初期ロールは system_r であり、それ以外はすべてそのロールを継承します。残りのロールは、targeted ポリシーと strict ポリシー間の互換性の目的で定義されます。[20]

この 4 つのロールは、このポリシーにより 3 つ定義されます。4 番目のロールである object_r は暗黙的なロールで、ポリシーソースには見つかりません。ロールは、ポリシーの 1 つ以上の宣言を使用してタイプで作成および設定されるため、すべてのロールを宣言する単一のファイルはありません。（ポリシー自体は、多数の別々のファイルから生成されることに注意してください。）

system_r

このロールは、ユーザープロセス以外のすべてのシステムプロセスを対象としています。

```
system_r (28 types)
   dhcpd_t
   httpd_helper_t
   httpd_php_t
   httpd_suexec_t
   httpd_sys_script_t
   httpd_t
```

```
    httpd_unconfined_script_t
    initrc_t
    ldconfig_t
    mailman_cgi_t
    mailman_mail_t
    mailman_queue_t
    mysqld_t
    named_t
    ndc_t
    nscd_t
    ntpd_t
    pegasus_t
    portmap_t
    postgresql_t
    snmpd_t
    squid_t
    syslogd_t
    system_mail_t
    unconfined_t
    winbind_helper_t
    winbind_t
    ypbind_t
```

user_r

これは、通常の Linux ユーザーのデフォルトのユーザーロールです。厳密なポリシーでは、個々のユーザーが使用される可能性があり、ユーザーに特別なロールで特権操作を実行できるようにします。targeted ポリシーでは、すべてのユーザーが unconfined_t ドメインで実行されます。

object_r

SELinux では、RBAC が使用されている場合、ロールはオブジェクトに使用されません。ロールはサブジェクトに厳密に使用されます。これは、ロールはタスク指向で、それらがアクションを実行するエンティティー（プロセスなど）にグループ化するためです。このようなエンティティーはすべて、サブジェクト と呼ばれます。このため、すべてのオブジェクトには object_r ロールがあり、ロールはラベルのプレースホルダーとしてのみ使用されます。

sysadm_r

これは、厳密なポリシーのシステム管理者ロールです。root ユーザーとして直接ログインする場合、デフォルトのロールは実際には staff_r である可能性があります。true の場合、newrole -r sysadm_r コマンドを使用して SELinux システム管理者ロールに切り替え、システム管理タスクを実行します。Targeted ポリシーでは、互換性のために以下の sysadm_r を保持します。

```
sysadm_r (6 types)
    httpd_helper_t
    httpd_sys_script_t
    initrc_t
```

> *ldconfig_t*
> *ndc_t*
> *unconfined_t*

Targeted ポリシーには、実際には 1 つのユーザー ID しかありません。libselinux はデフォルトの SELinux ユーザー ID として user_u に戻されるため、user_u ID が選択されました。これは、ログインしている Linux ユーザーに一致する SELinux ユーザーがない場合に発生します。ターゲットポリシーで user_u を単一ユーザーとして使用すると、strict ポリシーへの変更が容易になります。残りのユーザーは、strict ポリシーとの互換性のために存在します。[21]

1 つの例外は、SELinux ユーザー root です。プロセスのコンテキストでは、ユーザー ID として root が確認できます。これは、SELinux ユーザー root がコマンドラインからデーモンを起動するか、init によって最初に起動するデーモンを再起動する際に発生します。

---

[18]

[19]

[20]
すべてのロールは targeted ポリシーに選択できますが、system_r にはデーモンドメインに対する既存の認証があるため、プロセスが簡素化されます。これは、現在エイリアスロールにメカニズムが存在しないために実行されました。

[21]
また、ユーザーエイリアスのメカニズムも機能し、ターゲットポリシーの厳密なポリシーから単一のユーザーアイデンティティーにすべてのアイデンティティーのエイリアスを実行します。

第50章 SELINUX の使用

SELinux は、新しいセキュリティーパラダイムと、管理者およびエンドユーザー向けの新しいプラクティスとツールの両方を提供します。本章で説明するツールおよび手法は、エンドユーザー、管理者、およびアナリストが実施する標準操作に重点を置いています。

50.1. SELINUX のエンドユーザーコントロール

一般的に、エンドユーザーは、Red Hat Enterprise Linux がターゲットポリシーを実行している場合に、SELinux との相互作用はほとんどありません。これは、ユーザーが unconfined_t のドメインと、ターゲットデーモン 以外 の残りのシステムで実行しているためです。

ほとんどの場合、標準の DAC コントロールは、SELinux が参照される前に、必要なアクセスまたはパーミッションを持たないタスクを実行することを防ぎます。したがって、avc: denied メッセージを生成しない可能性があります。

以下のセクションでは、エンドユーザーが Red Hat Enterprise Linux システムで実行する必要がある可能性のある一般的なタスクおよびプラクティスを説明します。これらのタスクは、エンドユーザーだけでなく、すべての特権レベルのユーザーに適用されます。

50.1.1. ファイルの移動とコピー

ファイルシステムの操作では、セキュリティーコンテキストは、ファイルのラベル、アクセスしているプロセス、および操作が発生したディレクトリーという点として考慮する必要があります。このため、mv と cp のあるファイルを移動およびコピーすると、予期しない結果が生じる可能性があります。

ファイルのコピー：cp の SELinux オプション

特に指定しない限り、cp は、作成プロセスのドメインとターゲットディレクトリーのタイプに基づいて新しいファイルを作成するデフォルトの動作に従います。ラベルを設定する特定のルールがない限り、ファイルはターゲットディレクトリーからタイプを継承します。

-Z user:role:type オプションを使用して、新規ファイルに必要なラベルを指定します。

-p （または --preserve=mode,ownership,timestamps）オプションは指定された属性を保持し、可能であれば link などの追加属性を保持します。

```
touch bar foo
```

```
ls -Z bar foo
-rw-rw-r--  auser   auser   user_u:object_r:user_home_t   bar
-rw-rw-r--  auser   auser   user_u:object_r:user_home_t   foo
```

追加のコマンドライン引数なしで cp コマンドを使用すると、作成プロセスのデフォルトのタイプとターゲットディレクトリーを使用して、新しい場所にファイルのコピーが作成されます。この場合、cp および /tmp に適用される特定のルールがないため、新しいファイルには親ディレクトリーのタイプが設定されます。

```
cp bar /tmp
ls -Z /tmp/bar
-rw-rw-r--  auser   auser   user_u:object_r:tmp_t   /tmp/bar
```

tmp_t タイプは、一時ファイルのデフォルトタイプです。

-Z オプションを使用して、新規ファイルのラベルを指定します。

```
cp -Z user_u:object_r:user_home_t foo /tmp
ls -Z /tmp/foo
-rw-rw-r--  auser   auser   user_u:object_r:user_home_t   /tmp/foo
```

ファイルの移動：mv の SELinux オプション

mv のあるファイルを移動すると、ファイルに関連付けられた元のタイプが保持されます。問題が発生する可能性があるため、このコマンドを使用する場合は注意が必要です。たとえば、タイプ user_home_t を持つファイルを ~/public_html に移動すると、httpd デーモンは再ラベル付けするまでこれらのファイルを提供できなくなります。ファイルのラベリングの詳細は、「ファイルまたはディレクトリーの再ラベル」 を参照してください。

表50.1 mv コマンドおよび cp コマンドの動作

コマンド	動作
mv	ファイルは、元のラベルを保持します。これにより、問題、混乱、またはマイナーなセキュリティーが発生する可能性があります。たとえば、sbin_t ドメインで実行している tmpwatch プログラムは、ファイルのタイプが原因で、/tmp ディレクトリー内の経過時間ファイルを削除できない可能性があります。
cp	作成プロセスのドメイン(cp)とターゲットディレクトリーのタイプに基づくデフォルトの動作を使用して、ファイルのコピーを作成します。

コマンド	動作
cp -p	ファイルのコピーを作成し、可能な場合は指定された属性とセキュリティーコンテキストを保持します。デフォルトの属性は mode、ownership、および timestamps です。その他の属性は links および all です。
cp -Z <user:role:type>	指定されたラベルで ファイルのコピーを作成します。-Z オプションは、--context と同義語です。

50.1.2. プロセス、ユーザー、またはファイルオブジェクトのセキュリティーコンテキストの確認

プロセス ID の確認

Red Hat Enterprise Linux では、-Z オプションは --context と同等で、ps コマンド、id コマンド、ls コマンド、および cp コマンドで使用できます。SELinux に関する cp コマンドの動作は、表 50.1「mv コマンドおよび cp コマンドの動作」 で説明されています。

以下の例は、ps コマンドの出力例を一部示しています。ほとんどのプロセスは unconfined_t ドメインで実行されており、いくつかの例外があります。

```
[user@localhost ~]$ ps auxZ
LABEL                          USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME
COMMAND
system_u:system_r:init_t       root         1  0.0  0.1   2032   620 ?        Ss   15:09   0:00 init [5]
system_u:system_r:kernel_t     root         2  0.0  0.0      0     0 ?        S    15:09   0:00 [migration/0]
system_u:system_r:kernel_t     root         3  0.0  0.0      0     0 ?        SN   15:09   0:00 [ksoftirqd/0]

user_u:system_r:unconfined_t   user      3122  0.0  0.6   6908  3232 ?        S    16:47   0:01
/usr/libexec/gconfd-2 5
user_u:system_r:unconfined_t   user      3125  0.0  0.1   2540   588 ?        S    16:47   0:00
/usr/bin/gnome-keyring-daemon
user_u:system_r:unconfined_t   user      3127  0.0  1.4  33612  6988 ?        Sl   16:47   0:00
/usr/libexec/gnome-settings-daemon
user_u:system_r:unconfined_t   user      3144  0.1  1.4  16528  7360 ?        Ss   16:47   0:01
metacity --sm-client-id=default1
user_u:system_r:unconfined_t   user      3148  0.2  2.9  79544 14808 ?        Ss   16:47   0:03
gnome-panel --sm-client-id default2
```

ユーザー ID の確認

id コマンドで -Z オプションを使用して、ユーザーのセキュリティーコンテキストを判断できます。このコマンドでは、-Z と他のオプションを組み合わせることはできません。

```
[root@localhost ~]# id -Z
user_u:system_r:unconfined_t
```

id コマンドで -Z オプションを使用して、別のユーザーのセキュリティーコンテキストを検査できないことに注意してください。つまり、現在ログインしているユーザーのセキュリティーコンテキストのみを表示できます。

```
[user@localhost ~]$ id
uid=501(user) gid=501(user) groups=501(user) context=user_u:system_r:unconfined_t
[user@localhost ~]$ id root
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),10(wheel)
[user@localhost ~]$ id -Z root
id: cannot display context when selinux not enabled or when displaying the id
of a different user
```

ファイル ID の確認

ls コマンドに -Z オプションを使用して、一般的な長い形式情報をグループ化できます。モード、ユーザー、グループ、セキュリティーコンテキスト、およびファイル名の情報を表示できます。

```
cd /etc
ls -Z h* -d
drwxr-xr-x  root root  system_u:object_r:etc_t        hal
-rw-r--r--  root root  system_u:object_r:etc_t        host.conf
-rw-r--r--  root root  user_u:object_r:etc_t          hosts
-rw-r--r--  root root  system_u:object_r:etc_t        hosts.allow
-rw-r--r--  root root  system_u:object_r:etc_t        hosts.canna
-rw-r--r--  root root  system_u:object_r:etc_t        hosts.deny
drwxr-xr-x  root root  system_u:object_r:hotplug_etc_t  hotplug
drwxr-xr-x  root root  system_u:object_r:etc_t        hotplug.d
drwxr-xr-x  root root  system_u:object_r:httpd_sys_content_t htdig
drwxr-xr-x  root root  system_u:object_r:httpd_config_t httpd
```

50.1.3. ファイルまたはディレクトリーの再ラベル

~/public_html ディレクトリーなどのターゲットデーモンに関連する特別なディレクトリーに移動またはコピーする場合、または /home 以外のディレクトリーで動作するスクリプトを作成する場合は、ファイルの再ラベル付けが必要になる場合があります。

再ラベル付け操作には、以下の 2 つの一般的なタイプがあります。

- *ファイルの型を意図的に変更*
- *ポリシーに応じたデフォルトの状態へのファイルの復元*

また、管理者が実行する再ラベル付け操作もあります。これらについては、「ファイルシステムの再ラベル付け」 で説明されています。

ヒント

targeted ポリシーにおける SELinux パーミッション制御の大半は、Type Enforcement (TE)です。したがって、通常、セキュリティーラベルのユーザーおよびロール情報を無視し、タイプの変更にフォーカスできます。通常、ファイルのロールとユーザー設定を考慮する必要はありません。

注記

再ラベル付けがデーモンの実行可能ファイルのラベルに影響を及ぼす場合は、デーモンを再起動して、正しいドメインで実行していることを確認する必要があります。たとえば、/usr/sbin/mysqld に誤ったセキュリティーラベルがあり、restorecon などの再ラベル付け操作を使用してこれに対応する場合は、再ラベル付け操作の後に mysqld を再起動する必要があります。実行可能ファイルを正しいタイプ(mysqld_exec_t)に設定すると、起動時に適切なドメインに移行できるようになります。

chcon コマンドを使用して、ファイルを正しいタイプに変更します。このコマンドを使用するには、適用する正しいタイプを知っている必要があります。以下の例のディレクトリーおよびファイルには、/home で作成されたファイルシステムオブジェクトに定義されたデフォルトタイプのラベルが付けられています。

```
cd ~
ls -Zd public_html/
drwxrwxr-x  auser  auser  user_u:object_r:user_home_t public_html/

ls -Z web_files/
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t   1.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t   2.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t   3.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t   4.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t   5.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t   index.html
```

これらのファイルを public_html ディレクトリーに移動すると、元のタイプを保持します。

```
mv web_files/* public_html/
ls -Z public_html/
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t   1.html
```

```
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t   2.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t   3.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t   4.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t   5.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t   index.html
```

特別なユーザーのパブリック HTML フォルダーからこれらのファイルを表示可能にするには、httpd が読み取り権限を持つタイプが必要です。Apache HTTP Server は UserDir に対して設定され、ブール値 httpd_enable_homedirs が有効になります。

```
chcon -R -t httpd_user_content_t public_html/
ls -Z public_html
-rw-rw-r--  auser  auser  user_u:object_r:httpd_user_content_t   1.html
-rw-rw-r--  auser  auser  user_u:object_r:httpd_user_content_t   2.html
-rw-rw-r--  auser  auser  user_u:object_r:httpd_user_content_t   3.html
-rw-rw-r--  auser  auser  user_u:object_r:httpd_user_content_t   4.html
-rw-rw-r--  auser  auser  user_u:object_r:httpd_user_content_t   5.html
-rw-rw-r--  auser  auser  user_u:object_r:httpd_user_content_t   index.html

ls -Z public_html/ -d
drwxrwxr-x  auser  auser  user_u:object_r:httpd_user_content_t  public_html/
```

ヒント

カーネルで SELinux が無効になっているときに作成されたファイルなどのラベルがない場合は、chcon system_u:object_r:shlib_t foo.so で完全なラベルを指定する必要があります。そうしないと、部分的なコンテキストをラベル付けされていないファイルに適用するというエラーが発生します。

restorecon コマンドを使用して、ポリシーに従ってファイルをデフォルト値に戻します。ファイルシステム全体で機能するこの操作を実行する方法は、fixfiles または policy relabeling の 2 つがあります。これらの方法にはそれぞれスーパーユーザー権限が必要です。これらの方法の両方に対する注意点は、「ファイルシステムの再ラベル付け」 に記載されています。

以下の例は、デフォルトのユーザーのホームディレクトリーコンテキストを、異なるタイプのファイルセットに復元する方法を示しています。最初の 2 つのファイルセットにはタイプが異なるため、アーカイブのためにディレクトリーに移動されています。これらのコンテキストは相互に異なり、標準のユーザーのホームディレクトリーでは正しくありません。

```
ls -Z /tmp/
-rw-rw-r--  auser  auser  user_u:object_r:tmp_t         /tmp/file1
-rw-rw-r--  auser  auser  user_u:object_r:tmp_t         /tmp/file2
-rw-rw-r--  auser  auser  user_u:object_r:tmp_t         /tmp/file3

mv /tmp/{1,2,3} archives/
mv public_html/* archives/
```

```
ls -Z archives/
-rw-rw-r--  auser  auser  user_u:object_r:tmp_t          file1
-rw-rw-r--  auser  auser  user_u:object_r:httpd_user_content_t   file1.html
-rw-rw-r--  auser  auser  user_u:object_r:tmp_t          file2
-rw-rw-r--  auser  auser  user_u:object_r:httpd_user_content_t   file2.html
-rw-rw-r--  auser  auser  user_u:object_r:tmp_t          file3
-rw-rw-r--  auser  auser  user_u:object_r:httpd_user_content_t   file3.html
-rw-rw-r--  auser  auser  user_u:object_r:httpd_user_content_t   file4.html
-rw-rw-r--  auser  auser  user_u:object_r:httpd_user_content_t   file5.html
-rw-rw-r--  auser  auser  user_u:object_r:httpd_user_content_t  index.html
```

archives/ ディレクトリーには、ユーザーのホームディレクトリーにデフォルトのタイプが作成されているため、すでにデフォルトのタイプがあります。

```
ls -Zd archives/
drwxrwxr-x  auser  auser  user_u:object_r:user_home_t  archives/
```

restorecon コマンドを使用してファイルに再ラベル付けするには、ポリシーで設定されたデフォルトのファイルコンテキストを使用するため、これらのファイルには現在のディレクトリーのデフォルトラベルのラベルが付けられます。

```
/sbin/restorecon -R archives/
ls -Z archives/
-rw-rw-r--  auser  auser  system_u:object_r:user_home_t   file1
-rw-rw-r--  auser  auser  system_u:object_r:user_home_t   file1.html
-rw-rw-r--  auser  auser  system_u:object_r:user_home_t   file2
-rw-rw-r--  auser  auser  system_u:object_r:user_home_t   file2.html
-rw-rw-r--  auser  auser  system_u:object_r:user_home_t   file3
-rw-rw-r--  auser  auser  system_u:object_r:user_home_t   file3.html
-rw-rw-r--  auser  auser  system_u:object_r:user_home_t   file4.html
-rw-rw-r--  auser  auser  system_u:object_r:user_home_t   file5.html
-rw-rw-r--  auser  auser  system_u:object_r:user_home_t   index.html
```

50.1.4. セキュリティーコンテキストを保持するアーカイブの作成

tar ユーティリティーまたは star ユーティリティーを使用して、SELinux セキュリティーコンテキストを保持するアーカイブを作成できます。以下の例は star を使用して、このようなアーカイブを作成する方法を示しています。追加の属性がキャプチャーされ、*.star ファイルのヘッダーが xattrs に完全に対応するタイプになるように、適切な -xattr および -H=exustar オプションを使用する必要があります。これらのオプションおよびその他のオプションの詳細は、の man ページを参照してください。

以下の例は、html ファイルおよびディレクトリーの作成と抽出を示しています。2 つのディレクトリーには異なるラベルがあることに注意してください。ファイルコンテキストの未インポート部分は、印刷目的で省略されています（省略される...で示されます）。

```
ls -Z public_html/ web_files/
```

```
public_html/:
-rw-rw-r--  auser  auser  ...httpd_user_content_t 1.html
-rw-rw-r--  auser  auser  ...httpd_user_content_t 2.html
-rw-rw-r--  auser  auser  ...httpd_user_content_t 3.html
-rw-rw-r--  auser  auser  ...httpd_user_content_t 4.html
-rw-rw-r--  auser  auser  ...httpd_user_content_t 5.html
-rw-rw-r--  auser  auser  ...httpd_user_content_t index.html
web_files/:
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t  1.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t  2.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t  3.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t  4.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t  5.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t  index.html
```

以下のコマンドはアーカイブを作成し、すべての SELinux セキュリティーコンテキストを保持します。

```
star -xattr -H=exustar -c -f all_web.star public_html/ web_files/
star: 11 blocks + 0 bytes (total of 112640 bytes = 110.00k).
```

-Z オプションを指定して ls コマンドを使用して、セキュリティーコンテキストを検証します。

```
ls -Z all_web.star
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t \  all_web.star
```

アーカイブを別のディレクトリーにコピーできるようになりました。この例では、アーカイブは /tmp にコピーされます。派生一時タイプを作成する特定のポリシーがない場合、デフォルトの動作は tmp_t タイプを取得することです。

```
cp all_web.star /tmp/ cd /tmp/

ls -Z all_web.star
-rw-rw-r--  auser  auser  user_u:object_r:tmp_t  all_web.star
```

これで、星 を使用してアーカイブを展開し、拡張属性を復元できます。

```
star -xattr -x -f all_web.star
star: 11 blocks + 0 bytes (total of 112640 bytes = 110.00k).

ls -Z /tmp/public_html/ /tmp/web_files/
/tmp/public_html/:
-rw-rw-r--  auser  auser  ...httpd_sys_content_t 1.html
-rw-rw-r--  auser  auser  ...httpd_sys_content_t 2.html
-rw-rw-r--  auser  auser  ...httpd_sys_content_t 3.html
-rw-rw-r--  auser  auser  ...httpd_sys_content_t 4.html
```

```
-rw-rw-r--  auser  auser  ...httpd_sys_content_t 5.html
-rw-rw-r--  auser  auser  ...httpd_sys_content_t index.html
/tmp/web_files/:
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t  1.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t  2.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t  3.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t  4.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t  5.html
-rw-rw-r--  auser  auser  user_u:object_r:user_home_t  \ index.html
```

注意

star を使用してアーカイブを作成するときに絶対パスを使用する場合、アーカイブは同じパスに展開されます。たとえば、このコマンドで作成されたアーカイブは、ファイルを /var/log/httpd/ に復元します。

```
star -xattr -H=exustar -c -f httpd_logs.star /var/log/httpd/
```

このアーカイブの拡張を試みると、パス のファイルがアーカイブにあるファイルよりも新しい場合は、警告を発行します。

50.2. SELINUX の管理者コントロール

「SELinux のエンドユーザーコントロール」 でユーザーが行うタスクに加えて、SELinux 管理者は多くの追加タスクを実行することが必要になる場合があります。これらのタスクは通常、システムへの root アクセスを必要とします。このようなタスクは、targeted ポリシーで大幅に容易になります。たとえば、SELinux ユーザーで Linux ユーザーを追加、編集、または削除したり、ロールを考慮する必要はありません。

本セクションでは、管理者が SELinux を実行している Red Hat Enterprise Linux を管理するタスクの種類を説明します。

50.2.1. SELinux のステータス表示

sestatus コマンドは、設定可能なビューを SELinux のステータスに提供します。このコマンドの最も単純な形式により、以下の情報が表示されます。

```
~]# sestatus
SELinux status:                 enabled
```

```
SELinuxfs mount:              /selinux
Current mode:                 enforcing
Mode from config file:        enforcing
Policy version:               21
Policy from config file:      targeted
```

-v オプションには、/etc/sestatus.conf で指定された一連のファイルのセキュリティーコンテキストに関する情報が含まれます。

```
~]# sestatus -v
SELinux status:               enabled
SELinuxfs mount:              /selinux
Current mode:                 enforcing
Mode from config file:        enforcing
Policy version:               21
Policy from config file:      targeted

Process contexts:
Current context:              user_u:system_r:unconfined_t
Init context:                 system_u:system_r:init_t
/sbin/mingetty                system_u:system_r:getty_t
/usr/sbin/sshd                system_u:system_r:unconfined_t:s0-s0:c0.c1023

File contexts:
Controlling term:             user_u:object_r:devpts_t
/etc/passwd                   system_u:object_r:etc_t
/etc/shadow                   system_u:object_r:shadow_t
/bin/bash                     system_u:object_r:shell_exec_t
/bin/login                    system_u:object_r:login_exec_t
/bin/sh                       system_u:object_r:bin_t -> system_u:object_r:shell_exec_t
/sbin/agetty                  system_u:object_r:getty_exec_t
/sbin/init                    system_u:object_r:init_exec_t
/sbin/mingetty                system_u:object_r:getty_exec_t
/usr/sbin/sshd                system_u:object_r:sshd_exec_t
/lib/libc.so.6                system_u:object_r:lib_t -> system_u:object_r:lib_t
/lib/ld-linux.so.2            system_u:object_r:lib_t -> system_u:object_r:ld_so_t
```

-b は、ブール値の現在の状態を表示します。grep またはその他のツールと組み合わせて使用すると、特定のブール値のステータスを確認できます。

```
~]# sestatus -b | grep httpd | grep on$
httpd_builtin_scripting         on
httpd_disable_trans             on
httpd_enable_cgi                on
httpd_enable_homedirs           on
httpd_unified                   on
```

50.2.2. ファイルシステムの再ラベル付け

ファイルシステム全体の再ラベル付けが必要になることはありません。これは通常、SELinux の

ファイルシステムに初めてラベルを付けるか、ターゲットから strict ポリシーに変更するなど、異なるタイプのポリシー間で切り替える場合にのみ発生します。

init を使用したファイルシステムの再ラベル付け

ファイルシステムの再ラベル付けに推奨される方法は、マシンを再起動することです。これにより、init プロセスが再ラベル付けを実行し、アプリケーションの起動時に正しいラベルがあり、それらのラベルが適切な順序で開始されるようになります。再起動せずにファイルシステムに再ラベル付けすると、一部のプロセスが、誤ったコンテキストで実行し続ける可能性があります。すべてのデーモンが再起動し、正しいコンテキストで実行されていることを手作業で確認することは難しい場合があります。

この方法を使用してファイルシステムの再ラベル付けを行うには、以下の手順に従います。

```
touch /.autorelabel
reboot
```

起動時に init.rc は /.autorelabel の存在をチェックします。このファイルが存在する場合は、SELinux が完全なファイルシステムの再ラベル付けを実行し(/sbin/fixfiles -f -F relabel コマンドを使用して)、/.autorelabel を削除します。

修正ファイルを使用したファイルシステムの再ラベル付け

fixfiles コマンドを使用してファイルシステムの再ラベル付けをしたり、RPM データベースに基づいて再ラベル付けしたりできます。

次のコマンドを使用して、fixfiles コマンドのみを使用してファイルシステムの再ラベル付けを行います。

```
fixfiles relabel
```

以下のコマンドを使用して、RPM データベースに基づいてファイルシステムの再ラベル付けを行います。

```
fixfiles -R <packagename> restore
```

fixfiles を使用してパッケージからコンテキストを復元する方が安全で、より高速です。

注意

再起動せずにファイルシステム全体で fixfiles を実行すると、システムが不安定になる可能性があります。

再ラベル付け操作が、システムの起動時に実施されていたポリシーとは異なる新しいポリシーを適用する場合、既存のプロセスが不正確で安全でないドメインで実行されている可能性があります。たとえば、新しいポリシーでそのプロセスの移行が許可されないドメインにプロセスが配置され、そのプロセスだけに予期しないパーミッションを付与する可能性があります。

さらに、fixfiles の再 ラベルの 1 つは、/tmp/ を確実に再ラベル付けできないため、空の /tmp/ に対する承認を要求します。fixfiles は root として実行されるため、アプリケーションが依存している一時ファイルは消去されます。これにより、システムが不安定になったり、予期せず動作したりする可能性があります。

50.2.3. NFS ホームディレクトリーの管理

Red Hat Enterprise Linux 5 では、ほとんどのターゲットデーモンはユーザーデータと対話せず、NFS がマウントされたホームディレクトリーの影響を受けません。1 つの例外は Apache HTTP Server です。たとえば、マウントされたファイルシステム上にある CGI スクリプトには nfs_t タイプがあります。これは、httpd_t を実行できるタイプではありません。

デフォルトのタイプの nfs_t に問題がある場合は、別のコンテキストでホームディレクトリーをマウントしてみてください。

```
mount -t nfs -o context=user_u:object_r:user_home_dir_t \
 fileserver.example.com:/shared/homes/ /home
```

注意

「全ファイルシステムのセキュリティーコンテキストの指定」 httpd がスクリプトを実行できるようにディレクトリーをマウントする方法を説明します。ユーザーのホームディレクトリーに対してこれを行うと、Apache HTTP Server がこれらのディレクトリーへのアクセスを増やします。マウントポイントラベルが、マウントされたファイルシステム全体に適用されることに注意してください。

SELinux ポリシーの今後のバージョンでは、NFS の機能に対応しています。

50.2.4. ディレクトリーまたはツリーへのアクセスの付与

標準の Linux DAC パーミッションと同様に、ターゲットデーモンには、ディレクトリーツリーを切り離すための SELinux パーミッションが必要です。これは、ディレクトリーとそのコンテンツが同じタイプを持つ必要があるという意味ではありません。ディレクトリーの読み取りアクセスを付与する root_t、tmp_t、usr_t など、多くのタイプがあります。これらのタイプは、機密情報を含まないディレクトリーや、広く読み取り可能にするディレクトリーに適しています。また、異なるコンテキストを持つよりセキュアなディレクトリーの親ディレクトリーに使用することもできます。

avc: denied メッセージを使用する場合は、ディレクトリートラバーサルで発生する一般的な問題がいくつかあります。たとえば、多くのプログラムは、操作に必要なものではなく、ログに拒否メッセージを生成する ls -l / と同等のコマンドを実行します。そのためには、local.te ファイルに dontaudit ルールを作成する必要があります。

AVC 拒否メッセージを解釈しようとする場合は、path=/ コンポーネントで誤作用しないようにしてください。このパスは、root ファイルシステム / のラベルに関連しません。これは、実際にはデバイスノード上のファイルシステムのルートに相対的です。たとえば、/var/ ディレクトリーが LVM (論理ボリューム管理)にあるとします。[22]）デバイス /dev/dm-0。 デバイスノードは、メッセージ dev=dm-0 として識別されます。この例では path= / が LVM デバイス dm-0 の最上位に表示されますが、これはルートファイルシステムの指定 / と同じではありません。

50.2.5. システムのバックアップおよび復元

「セキュリティーコンテキストを保持するアーカイブの作成」 の説明を参照してください。

50.2.6. Enforcement の有効化または無効化

ランタイム時に SELinux 強制を有効または無効にしたり、コマンドラインまたは GUI を使用し

て、システムの起動時に正しいモードで開始するように設定できます。SELinux は、無効の、つまりカーネルで有効ではないことを意味します。Permissive は、SELinux が実行中およびログに記録されているが、パーミッションを制御していません。または、SELinux が実行中および強制ポリシーを意味します。

setenforce コマンドを使用して、実行時に Permissive モードと enforcing モードを切り替えます。setenforce 0 を使用して Permissive モードに入り、setenforce 1 を使用して Enforcing モードに入ります。

sestatus コマンドは、起動時に参照される設定ファイルの現在のモードとモードを表示します。

```
~]# sestatus | grep -i mode
Current mode:           permissive
Mode from config file:  permissive
```

ランタイムの適用を変更しても、ブート時間設定には影響がないことに注意してください。

```
~]# setenforce 1
~]# sestatus | grep -i mode
Current mode:           enforcing
Mode from config file:  permissive
```

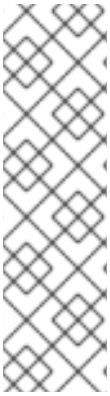
1 つのデーモンの Enforcing モードを無効にすることもできます。たとえば、名前付きデーモンと SELinux のトラブルシューティングを行う場合は、そのデーモンのみの Enforcing をオフにできます。

getsebool コマンドを使用して、ブール値の現在のステータスを取得します。

```
~]# getsebool named_disable_trans
named_disable_trans --> off
```

以下のコマンドを使用して、このデーモンの enforcing モードを無効にします。

```
~]# setsebool named_disable_trans 1
~]# getsebool named_disable_trans
named_disable_trans --> on
```



注記

これにより、ランタイム値のみが設定されます。-P オプションを使用して、再起動後も変更が永続化されるようにします。

on に設定されている *_disable_trans ブール値は、プロセスが実行時にドメインに移行しないように条件を呼び出します。

これらのブール値のうちどのブール値が設定されているかを調べるには、次のコマンドを使用します。

```
~]# getsebool -a | grep disable.*on
httpd_disable_trans=1
mysqld_disable_trans=1
ntpd_disable_trans=1
```

setsebool コマンドを使用して、任意の数のブール値を設定できます。

```
setsebool -P httpd_disable_trans=1 mysqld_disable_trans=1 ntpd_disable_trans=1
```

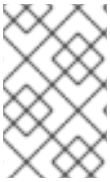
また、togglesebool < boolean_name> を使用して 特定のブール値の値を変更することもできます。

```
~]# getsebool httpd_disable_trans
httpd_disable_trans --> off
~]# togglesebool httpd_disable_trans
httpd_disable_trans: active
```

これらのすべての設定は、system-config-selinux を使用して設定できます。同じ設定ファイルが使用されるため、変更は双方向に表示されます。

ランタイムブール値の変更

以下の手順に従って、GUI を使用してランタイムのブール値を変更します。



注記

この手順を実行するには、管理者権限が必要です。

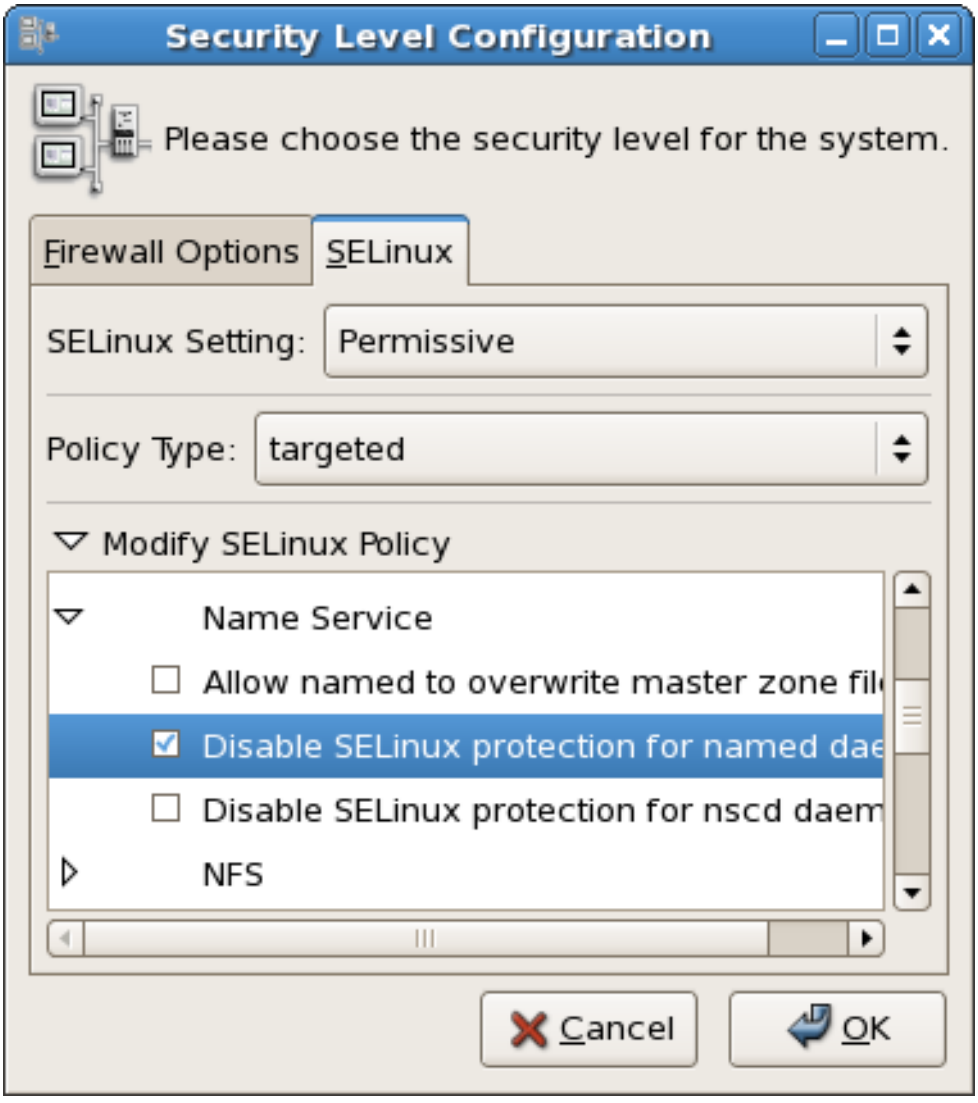
1.
System メニューで Administration を参照し、Security Level and Firewall をクリックして Security Level Configuration ダイアログボックスを表示します。

2.
SELinux タブをクリックし、SELinux ポリシーの変更 をクリックします。

3.
選択リストで、Name Service エントリーの横にある矢印をクリックし、Disable SELinux protection for named daemon チェックボックスを選択します。

4.
OK をクリックして変更を適用します。ポリシーが再読み込みされるまでに少し時間がかかる場合があります。

図50.1 Security Level Configuration ダイアログボックスを使用してランタイムのブール値を変更します。



[D]

スクリプトでこれらの設定を制御する場合は、 **setenforce (1)**、**getenforce (1)**、および **selinuxenabled (1)** コマンドを使用でき ます。

50.2.7. SELinux の有効化または無効化

重要な影響

SELinux が無効になっている間にファイルに加えた変更により、予期しないセキュリティーラベルが発生する可能性があり、新規ファイルにはラベルがありません。**SELinux** を再度有効にした後に、ファイルシステムの **part** または **all** のラベルを変更する必要がある場合があります。

コマンドラインで、**/etc/sysconfig/selinux** ファイルを編集できます。このファイルは、**/etc/selinux/config** へのシンボリックリンクです。設定ファイルは自己説明的です。**SELINUX** または **SELINUXTYPE** の値を変更すると、**SELinux** のステータスと、次回システム起動時に使用するポリシーの名前が変更されます。

```
~]# cat /etc/sysconfig/selinux
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#       enforcing - SELinux security policy is enforced.
#       permissive - SELinux prints warnings instead of enforcing.
#       disabled - SELinux is fully disabled.
SELINUX=permissive
# SELINUXTYPE= type of policy in use. Possible values are:
#       targeted - Only targeted network daemons are protected.
#       strict - Full SELinux protection.
SELINUXTYPE=targeted

# SETLOCALDEFS= Check local definition changes
SETLOCALDEFS=0
```

GUI を使用した SELinux モードの変更

以下の手順に従って、**GUI** で **SELinux** のモードを変更します。

注記

この手順を実行するには、管理者権限が必要です。

1. **System** メニューで **Administration** を参照し、**Security Level and Firewall** をクリックして **Security Level Configuration** ダイアログボックスを表示します。

2.
SELinux タブをクリックします。

3.
SELinux の設定 で Disabled、Enforcing、または Permissive のいずれかを選択し、OK をクリックします。

4.
有効 から 無効 に変更した場合や、 その逆の場合は、変更を有効にするためにマシンを再起動する必要があります。

このダイアログボックスを使用して行った変更は、即座に /etc/sysconfig/selinux に反映されます。

50.2.8. ポリシーの変更

本セクションでは、システムでカスタマイズされたポリシーを使用する簡単な概要を説明します。本トピックの完全な説明は、本書では扱いません。

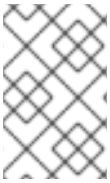
システムに別のポリシーを読み込むには、/etc/sysconfig/selinux で以下の行を変更します。

```
SELINUXTYPE=<policyname>
```

&lt ;policyname> は、/etc/selinux/ の下のポリシー名ディレクトリーです。これは、カスタムポリシーがインストールされていることを前提としています。SELINUXTYPE パラメーターを変更したら、以下のコマンドを実行します。

```
touch /.autorelabel
reboot
```

system-config-selinux ユーティリティーを使用して別のポリシーを読み込むには、以下の手順を使用します。



注記

この手順を実行するには、管理者権限が必要です。

1.
必要なポリシーの完全なディレクトリー構造が /etc/selinux に存在することを確認します。

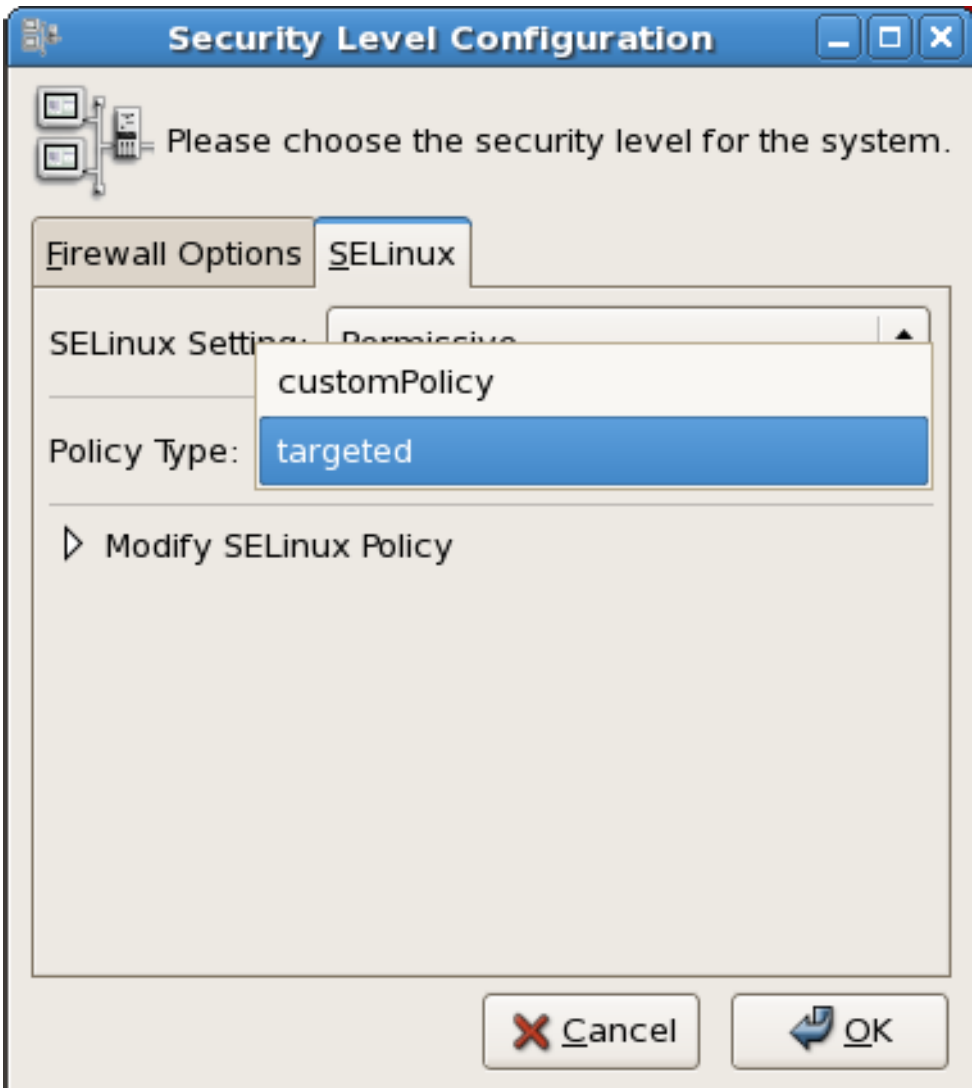
2. *System メニューで Administration を参照し、Security Level and Firewall をクリックして Security Level Configuration ダイアログボックスを表示します。*

3. *SELinux タブをクリックします。*

4. *Policy Type 一覧で読み込むポリシーを選択し、OK をクリックします。このリストは、複数のポリシーがインストールされている場合にのみ表示されます。*

5. *変更を有効にするには、マシンを再起動します。*

図50.2 Security Level Configuration ダイアログボックスを使用してカスタムポリシーをロードします。



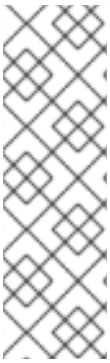
[D]

50.2.9. 全ファイルシステムのセキュリティーコンテキストの指定

mount -o context= コマンドを使用して、ファイルシステム全体で単一のコンテキストを設定できます。これは、すでにマウントされており、**xattrs** をサポートするファイルシステム、または **cifs_t** や **nfs_t** などの **genfs** ラベルを取得するネットワークファイルシステムである可能性があります。

たとえば、マウントされたディレクトリーまたはループバックファイルシステムから **Apache HTTP Server** を読み取る必要がある場合は、タイプを **httpd_sys_content_t** に設定する必要があります。

```
mount -t nfs -o context=system_u:object_r:httpd_sys_content_t \
 server1.example.com:/shared/scripts /var/www/cgi
```



ヒント

httpd および **SELinux** の問題のトラブルシューティングを行う場合は、状況の複雑さを軽減します。たとえば、**/mnt** にファイルシステムをマウントし、**/var/www/html/foo** へのシンボリックリンクがある場合は、セキュリティーコンテキストが 2 つあります。1 つのセキュリティーコンテキストはオブジェクトクラス **file** と、タイプ **lnk_file** のもう 1 つはポリシーによって処理されるため、予期せぬ動作が発生する可能性があります。

50.2.10. ファイルまたはディレクトリーのセキュリティーカテゴリーの変更

ファイルおよびユーザーのセキュリティーカテゴリーの追加および変更に関する詳細は、「」 および 「」 を参照してください。

50.2.11. 特定のセキュリティーコンテキストでのコマンドの実行

runcon コマンドを使用して、特定のコンテキストでコマンドを実行できます。これはスクリプト作成やテストポリシーに役立ちますが、正しく実装されていることを確認するには注意が必要です。

たとえば、以下のコマンドを実行して、間違ったラベルの付いたコンテンツをテストできます。コマンドの後に表示される引数は、コマンドの一部とみなされます。（この例では、**~/bin/contexttest** はユーザー定義のスクリプトです。）

```
runcon -t httpd_t ~/bin/contexttest -ARG1 -ARG2
```

以下のようにコンテキスト全体を指定することもできます。

```
runcon user_u:system_r:httpd_t ~/bin/contexttest
```

50.2.12. スクリプトの便利なコマンド

以下は、SELinux で導入された便利なコマンド一覧です。これは、システムの管理に役立つスクリプトを作成する際に役に立ちます。

getenforce

このコマンドは、SELinux の Enforcing ステータスを返します。

setenforce [En for ssive | 1 | 0]

このコマンドは、SELinux の Enforcing モードを制御します。オプション 1 または Enforcing は、SELinux に Enforcing モードに入るように指示します。オプション 0 または Permissive オプションは、SELinux にパッシブモードに入るように指示します。アクセス違反は引き続きログに記録されますが、防止されません。

selinuxenabled

このコマンドは、SELinux が有効な場合はステータスが 0 で、SELinux が無効になっている場合は 1 で終了します。

```
~]# selinuxenabled
~]# echo $?
0
```

getsebool [-a] [boolean_name]

このコマンドは、すべてのブール値(-a)または特定のブール値(<boolean_name>)のステータスを表示します。

setsebool [-P] <boolean_name> value | bool1=val1 bool2=val2 ...

このコマンドは、1 つ以上のブール値を設定します。-P オプションを使用すると、再起動後も変更が永続化されます。

togglesebool boolean ...

このコマンドは、1 つ以上のブール値の設定を切り替えます。これは、メモリーのみのブール値設定に影響します。変更は再起動後は維持されません。

50.2.13. 異なるロールへの変更

newrole コマンドを使用して、指定したタイプやロールで新しいシェルを実行します。ロールの変更は、通常 strict ポリシーでのみ意味を持ちます。通常、targeted ポリシーは単一のロールに制限されます。タイプの変更は、テスト、検証、および開発の目的で役に立つ場合があります。

```
newrole -r <role_r> -t <type_t> [-- [ARGS]...]
```

ARGS は、/etc/passwd ファイルのユーザーのエントリーで指定されたシェルに直接渡されます。

注記

newrole コマンドは policycoreutils-newrole パッケージの一部で、strict ポリシーまたは MLS ポリシーをインストールする場合に必要になります。Red Hat Enterprise Linux では、デフォルトではインストールされません。

50.2.14. リブートのタイミング

SELinux パースペクティブからシステムを再起動する主な理由は、ファイルシステムのラベルを完全に再ラベル付けすることです。SELinux を有効または無効にするには、システムの再起動が必要になる場合があります。

50.3. SELINUX のアナリストコントロール

このセクションでは、セキュリティーアナリストが SELinux システムで実行する必要がある可能性のある一般的なタスクについて説明します。

50.3.1. カーネル監査の有効化

SELinux の分析またはトラブルシューティングの演習の一環として、完全なカーネルレベルの監査を有効にすることを選択できます。これは、各 AVC 監査メッセージに対して 1 つ以上の追加の監査メッセージを生成するため、詳細度が非常に高くなります。この監査レベルを有効にするには、/etc/grub.conf ファイルまたは起動時に GRUB メニューのいずれかで、カーネルブート行に audit=1 パラメーターを追加します。

これは、ディレクトリーが Web コンテンツとしてラベル付けされていないため、httpd が ~/public_html へのアクセスが拒否された場合の完全な監査ログエントリーの例です。audit (...)フィールドのタイムスタンプとシリアル番号のタイムスタンプは、それぞれ同一である点に注意してください。これにより、監査ログで特定のイベントを簡単に追跡できます。

```
Jan 15 08:03:56 hostname kernel: audit(1105805036.075:2392892): \
 avc:  denied  { getattr } for  pid=2239 exe=/usr/sbin/httpd \
 path=/home/auser/public_html dev=hdb2 ino=921135 \
 scontext=user_u:system_r:httpd_t \
 tcontext=system_u:object_r:user_home_t tclass=dir
```

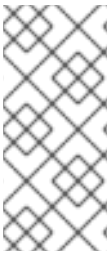
以下の監査メッセージは、関連するシステムコールの種類など、ソースの詳細を示しています。これは、httpd がディレクトリーを表示しようとしたことを示しています。

```
Jan 15 08:03:56 hostname kernel: audit(1105805036.075:2392892): \
 syscall=195 exit=4294967283 a0=9ef88e0 a1=bfecc0d4 a2=a97ff4 \
 a3=bfecc0d4 items=1 pid=2239 loginuid=-1 uid=48 gid=48 euid=48 \
 suid=48 fsuid=48 egid=48 sgid=48 fsgid=48
```

以下のメッセージは、ターゲットに関する詳細情報を提供します。

```
Jan 15 08:03:56 hostname kernel: audit(1105805036.075:2392892): \
 item=0 name=/home/auser/public_html inode=921135 dev=00:00
```

シリアル番号のタイムスタンプは、特定の監査イベントで常に同じです。タイムスタンプは同一である場合とそうでない場合があります。



注記

トラブルシューティングに監査デーモンを使用している場合、デーモンは /var/log/audit/audit.log などの /var/log/messages 以外の場所に監査メッセージをキャプチャーできます。

50.3.2. ログのダンプと表示

SELinux の Red Hat Enterprise Linux 5 の実装では、AVC 監査メッセージが /var/log/messages にルーティングされます。標準の検索ユーティリティー(grepなど)のいずれかを使用して、avc または audit が含まれる行を検索できます。

[22]
LVM は、論理ボリュームに分割される仮想プールへの物理ストレージのグループです。

第51章 SELINUX ポリシーのカスタマイズ

51.1. はじめに

Red Hat Enterprise Linux の以前のリリースでは、selinux-policy-targeted-sources パッケージをインストールしてから、/etc/selinux/targeted/src/policy/domains/misc ディレクトリーに local.te ファイルを作成する必要がありました。audit2allow ユーティリティーを使用して、AVC メッセージを allow ルールに変換し、ポリシーを再構築して再読み込みできます。

この問題は、新しいポリシーパッケージがリリースされるたびに、ローカルポリシーを保持するために Makefile を実行する必要があるという問題がありました。

Red Hat Enterprise Linux 5 では、このプロセスは完全に改訂されています。sources rpm パッケージは完全に削除され、ポリシーパッケージはカーネルと同様に処理されます。ポリシーの構築に使用されるソースを確認するには、ソース rpm selinux-policy-XYZ.src.rpm をインストールする必要があります。selinux-policy-devel パッケージも追加されており、さらにカスタマイズ機能が提供されます。

51.1.1. モジュールポリシー

Red Hat Enterprise Linux では、モジュールポリシー の概念が導入されています。これにより、ベンダーはオペレーティングシステムポリシーとは別に SELinux ポリシーを提供できます。また、管理者は、次のポリシーインストールについて考慮せずに、ポリシーに対するローカルの変更を行うことができます。追加された最も重要なコマンドは semodule です。

semodule は、モジュールのインストール、アップグレード、一覧表示、および削除など、SELinux ポリシーモジュールの管理に使用されるツールです。また、semodule を使用して、モジュールストアからのポリシーの再ビルドを強制したり、他のトランザクションを実行せずにポリシーの再読み込みを強制することもできます。semodule は、semodule_package によって作成されたモジュールパッケージに作用します。通常、これらのファイルには .pp 接尾辞（ポリシーパッケージ）がありますが、これはいずれの方法でも義務付けられていません。

51.1.1.1. ポリシーモジュールの一覧表示

システムのポリシーモジュールを一覧表示するには、semodule -l コマンドを使用します。

```
~]# semodule -l
amavis  1.1.0
ccs     1.0.0
clamav  1.1.0
dcc     1.1.0
evolution       1.1.0
```

```
iscsid  1.0.0
mozilla 1.1.0
mplayer 1.1.0
nagios  1.1.0
oddjob  1.0.1
pcscd   1.0.0
pyzor   1.1.0
razor   1.1.0
ricci   1.0.0
smartmon        1.1.0
```

注記

このコマンドは、インストールされているベースポリシーモジュールを一覧表示しません。

/usr/share/selinux/targeted/ ディレクトリーには、多数のポリシーパッケージ (.pp)ファイルが含まれます。これらのファイルは selinux-policy rpm に含まれており、ポリシーファイルの構築に使用されます。*

51.2. ローカルポリシーモジュールの構築

以下のセクションでは、実際の例を使用してローカルポリシーモジュールをビルドし、現在のポリシーに関する問題に対応します。この問題には、setsebool コマンドを実行し、次にターミナルを使用しようとする ypbind init スクリプトが含まれます。これにより、以下の拒否が生成されます。

```
type=AVC msg=audit(1164222416.269:22): avc:  denied  { use } for  pid=1940 comm="setsebool"
name="0" dev=devpts ino=2 \
 scontext=system_u:system_r:semanage_t:s0 tcontext=system_u:system_r:init_t:s0 tclass=fd
```

すべてが正常に動作する場合でも（つまり、意図したように実行されるアプリケーションフォームを妨げていない）、ユーザーの通常のワークフローを中断します。ローカルポリシーモジュールの作成は、この問題に対応します。

51.2.1. audit2allow を使用したローカルポリシーモジュールの構築

aud2allow ユーティリティーに、ポリシーモジュールを構築する機能が追加されました。以下のコマンドを使用して、audit.log ファイルの特定のコンテンツに基づいてポリシーモジュールを構築します。

```
ausearch -m AVC --comm setsebool | audit2allow -M mysemanage
```

audit2allow ユーティリティーは、タイプ強制ファイル(mysemanage.te)を構築している。次に、checkmodule コマンドを実行してモジュールファイル(mysemanage.mod)をコンパイルします。最後に、semodule_package コマンドを使用してポリシーパッケージ(mysemanage.pp)を作成します。semodule_package コマンドは、異なるポリシーファイル（通常はモジュールとファイルコンテキストファイル）をポリシーパッケージに統合します。

51.2.2. Type Enforcement (TE)ファイルの分析

cat コマンドを使用して、TE ファイルの内容を確認します。

```
~]# cat mysemanag.te
module mysemanage 1.0;

require {
 class fd use;
 type init_t;
 type semanage_t;
 role system_r;
};

allow semanage_t init_t:fd use;
```

TE ファイルは 3 つのセクションで設定されています。最初のセクションは、モジュール 名とバージョンを識別する module コマンドです。モジュール名は一意である必要があります。既存のモジュールの名前を使用して semanage モジュールを作成すると、システムは既存のモジュールパッケージを新たに作成したバージョンに置き換えることができます。モジュール行の最後の部分はバージョンです。semodule はモジュールパッケージを更新し、現在インストールされているバージョンに対して更新バージョンを確認できます。

TE ファイルの次のブロックは require ブロックです。これにより、このモジュールをインストールする前に、システムポリシーで必要なタイプ、クラス、およびロールについてポリシーローダーに通知します。これらのフィールドのいずれかが未定義の場合、semodule コマンドは失敗します。

最後に許可ルールです。この例では、semodule がファイル記述子にアクセスする必要がないため、この行を dontaudit に変更できます。

51.2.3. ポリシーパッケージの読み込み

ローカルポリシーモジュールの作成プロセスの最後のステップは、ポリシーパッケージをカーネルにロードすることです。

semodule コマンドを使用して policy パッケージを読み込みます。

```
~]# semodule -i mysemanage.pp
```

このコマンドはポリシーファイルを再コンパイルし、ファイルコンテキストファイルを再生成します。変更は永続的であり、再起動後も維持されます。また、ポリシーパッケージファイル (mysemanage.pp)を他のマシンにコピーして、semodule を使用してインストールすることもできます。

audit2allow コマンドは、ポリシーパッケージを作成するために実行したコマンドを出力し、TE ファイルを編集できるようにします。つまり、必要に応じて新しいルールを追加するか、allow ルールを dontaudit に変更できることを意味します。次に、ポリシーパッケージを再コンパイルして再パッケージ化し、再度インストールできます。

ポリシーパッケージの数に制限がないため、ローカルでの変更ごとに作成できます。または、引き続き単一のパッケージを編集することもできますが、require ステートメントがすべての許可ルールに一致することを確認する必要があります。

第52章 REFERENCES

以下の参考資料は、SELinux および Red Hat Enterprise Linux に関連する追加情報へのポインターで、本書では扱いません。SELinux の迅速な開発により、本ガイドの一部は Red Hat Enterprise Linux の特定のリリースにのみ適用されます。

書籍

SELinux by Example

Mayer, MacMillan, and Caplan

Prentice Hall, 2007

チュートリアルおよびヘルプ

Apache HTTP SELinux ポリシーの概要およびカスタマイズ

http://docs.fedoraproject.org/selinux-apache-fc3/

Russell Coker のチュートリアルと講演

http://www.coker.com.au/selinux/talks/ibmtu-2004/

汎用 Writing SELinux ポリシー HOWTO

https://sourceforge.net/docman/display_doc.php?docid=21959[amp]group_id=21266

Red Hat ナレッジベース

http://kbase.redhat.com/

全般情報

NSA SELinux のメイン Web サイト

http://www.nsa.gov/research/selinux/index.shtml

NSA SELinux FAQ

http://www.nsa.gov/research/selinux/faqs.shtml

Fedora SELinux FAQ

http://docs.fedoraproject.org/selinux-faq/

SELinux NSA のオープンソースセキュリティー強化 Linux

http://www.oreilly.com/catalog/selinux/

Technology

オブジェクトクラスおよびパーミッションの概要

http://www.tresys.com/selinux/obj_perms_help.html

セキュリティーポリシーの Linux オペレーティングシステムへの柔軟なサポートの統合(Linux における Flask 実装の履歴)

http://www.nsa.gov/research/_files/selinux/papers/freenix01/freenix01.shtml

Linux セキュリティーモジュールとしての SELinux の実装

http://www.nsa.gov/research/selinux/index.shtmlpapers/module-abs.cfm

Security-Enhanced Linux のセキュリティーポリシー設定

http://www.nsa.gov/research/_files/selinux/papers/policy/policy.shtml

コミュニティー

SELinux コミュニティーページ

http://selinux.sourceforge.net

IRC

irc.freenode.net, #rhel-selinux

履歴

Flask のクイック履歴

http://www.cs.utah.edu/flux/fluke/html/flask.html

Fluke に関する完全な背景

http://www.cs.utah.edu/flux/fluke/html/index.html

パート VIII. RED HAT のお客様および認定

Red Hat の取り組みと認定は、おそらく Linux にとって最善と見なされ、おそらくすべての IT 分野で考慮されています。Red Hat の認定プログラムは、経験のある Red Hat のエキスパートによってまったく対応し、実際のライブシステムでの能力を測定し、採用者や IT 担当者にとって大きな要求となります。

適切な認定の選択は、お客様の背景と目標によって異なります。UNIX または Linux の高度な経験、最小限の経験、または Linux の経験がなくても、Red Hat のトレーニングと認定パスが適切です。

第53章 RED HAT のお客様および認定

53.1. TRAIN に 3 つの方法

登録を開く

オープン登録の取り組みは、米国および 125 以上の場所で、50 以上の場所で継続的に提供されます。Red Hat の取り組みは、少なくとも 1 つの専用システム、そしてある程度の 5 つの専用システムにアクセスできるパフォーマンスベースの学習者です。教師はすべて、経験のある Red Hat Certified Engineers (----------|-----s)で、その方針を熟知している方です。

学習スケジュールは http://www.redhat.com/explore/trainingから入手できます。

オンサイトセッション

オンサイトトレーニングは、クラスごとに 12 - 16 のチームが Red Hat が提供しています。Red Hat の技術スタッフは、Red Hat Enterprise Linux、Red Hat、または JBoss の認定テストを実行する準備が整うため、技術スタッフが技術スタッフをサポートします。オンサイトは、大規模なグループを一度にトレーニングする優れた方法です。オープン登録は、後でインクリメンタルトレーニングに使用できます。

詳細については、 http://www.redhat.com/explore/onsiteをご覧ください。

eLearning

Red Hat Enterprise Linux 4 用の完全に更新されています !クラスの時間はありませんか ?Red Hat の e 関連タイトルはオンラインに提供され、RHCT およびvideo 追跡スキルについて説明します。また、拡張するカタログには、最新のプログラミング言語、スクリプト、および電子商取引も含まれます。

当然の一覧については、 http://www.redhat.com/explore/elearningにアクセスしてください。

53.2. MICROSOFT CERTIFIED PROFESSIONAL RESOURCE CENTER

カスタマイズした情報と、Red Hat 認定を個人のポートフォリオに追加することを目的とする Microsoft® Certified Professionals を提供します。

今すぐお試しください： http://www.redhat.com/explore/manager

第54章 認定トラッキング

Red Hat Certified Technician®(RHCT®)

3 年目に入り、Red Hat Certified Technician はすべての Linux で最速の認定であり、現在 15,000 以上の認定保持者となります。RHCT は、Linux 認証情報を確立するための最適な最初の手順であり、UNIX 以外の環境または Linux 環境から移行するための理想的な初期認定です。

Red Hat の認定は、Linux にとって最善であると見なされており、おそらくすべての IT においてもそうです。Red Hat の認定プログラムは、経験のある Red Hat のエキスパートによってまったく対応し、実際のライブシステムでの能力を測定し、採用者や IT 担当者にとって大きな要求となります。

適切な認定の選択は、お客様の背景と目標によって異なります。UNIX または Linux の高度な経験、最小限の経験、または Linux の経験がなくても、Red Hat のトレーニングと認定パスが適切です。

Red Hat Certified Engineer®(NORMAL®)

Red Hat Certified Engineer は 1999 年に開始され、20,000 を超える Linux エキスパートがもたらされています。独立した調査では、Linux 認定の "crown jewel" と呼ばれることが、すべての IT で #1 位にランク付けされました。

Red Hat 認定セキュリティー(RHCSS)

RHCSS には、現在のエンタープライズ環境のセキュリティー要件を満たすために、Red Hat Enterprise Linux、Red Hat Directory Server、および SELinux のセキュリティー知識と特殊なスキルがあります。RHCSS は、Red Hat の最新の認定であり、その種類の 1 つが Linux のみです。

Red Hat 認定アーキテクト(NORMAL)

上級トレーニングをシークすると、エンタープライズアーキテクトへの登録と、新しく発表された Red Hat 認定アーキテクト(NORMAL)認定の能力を証明できます。Red Hat Certified Technician (RHCT)および Red Hat Certified Engineer (NORMAL)の認定を主な認定であり、Linux 分野で最も要求の認定です。

54.1. 無料の事前評価テスト

弊社の自動化された評価テストで、Linux スマートをテストし、Red Hat のトレーニングレベルを特定します。

完全に無料で、義務はなく、10 分です。http://www.redhat.com/explore/assess

第55章 RH033: RED HAT LINUX ESSENTIALS

http://www.redhat.com/training/rhce/courses/rh033.html

55.1. 当然の説明

RHCT 認定および II 認定追跡の最初の方は、Linux または UNIX を使用しておらず、他のオペレーティングシステムでコマンドラインエクスペリエンスのない個人にとって、RH033 にとって理想的です。Red Hat Enterprise Linux 環境の基本を学び、システム管理者として将来のロールに向けて準備します。

55.1.1. 前提条件

任意のコンピューターシステムでのユーザーレベルの経験、メニューの使用、グラフィカルユーザーインターフェイスの使用。

55.1.2. 目的

一般的なコマンドラインプロセスとデスクトップの生産性ロールのために Red Hat システムの使用およびカスタマイズが可能な Red Hat Enterprise Linux パワーユーザーで、システム管理(RH133)の学習に備える準備ができています。

55.1.3. 対象者

Linux を初めて使用し、UNIX またはコマンドラインスキルを持たないユーザーで、独自の Red Hat Linux システムを使用し、制御する基本的なスキルを開発および実践したいユーザー。

55.1.4. 学習目的

1. Linux ファイルシステムを理解する
2. 一般的なファイルメンテナーンスを実行する
3. GNOME インターフェイスの使用およびカスタマイズ

4. *コマンドラインから基本的な Linux コマンドを発行する*

5. *GNOME GUI を使用して一般的なタスクを実行する*

6. *vi エディターを使用してテキストドキュメントを開いた、編集、および保存します。*

7. *ファイルアクセスのパーミッション*

8. *X Window System のカスタマイズ*

9. *正規表現パターンの一致と I/O リダイレクト*

10. *システムでパッケージをインストール、アップグレード、削除、およびクエリーします。*

11. *ユーザーのネットワークユーティリティー*

12. *パワーユーザーユーティリティー*

55.1.5. フォローオンフォル

RH133 Red Hat Linux Sys。admin。

RH253 Red Hat Linux Net. および Sec. Admin

RH300 Red Hat Linux ライトライト(RH300)

ITT Systems Division, Mike Kimmel, Mike Kimmel, Mike Kimmel, Mike Kimmel, ITT Systems Division

第56章 RH035: RED HAT LINUX ESSENTIALS FOR WINDOWS PROFESSIONALS

http://www.redhat.com/training/rhce/courses/rh035.html

56.1. 当然の説明

従来の UNIX または Linux 経験のない Windows® のスキルのために設計されており、この試験では Red Hat Enterprise Linux システム管理の基本スキルについて説明します。1 日目は、個人が Linux 管理能力をポートフォリオに正常に追加するための概念的かつ実用的な移行を提供します。残りの 4 日間は、非常に要求の厳しい RH033 試験と組み合わせて、個人を Red Hat Enterprise Linux 環境の基本に確保し、クロスプラットフォームのシステム管理者として今後のロール用に準備します。また、この試験は、RHCT および 2007 追跡の最初の試験としても機能します。

56.1.1. 前提条件

Technician またはシステム管理者レベルで Windows OS 製品を使用するジョブタスクについて経験があり、IT の専門的な経験、UNIX や Linux の経験は必要ありません。

56.1.2. 目的

一般的なコマンドラインプロセスに精通している Red Hat Enterprise Linux のパワーユーザーは、グラフィカルツールを使用してシステム管理タスクを実行できます。また、Red Hat Enterprise Linux システム管理(RH133)の理解を深める準備が整います。

56.1.3. 対象者

一般的な学生は、グラフィックユーザーインターフェイスを使用してサーバーの管理を希望する Windows 技術です。また、Red Hat Enterprise Linux システムを効果的に管理し、個々のスキルセットを広げることが望まれます。

56.1.4. 学習目的

1. ソフトウェアのインストール、ネットワークの設定、認証の設定、およびグラフィカルツールを使用した各種サービスのインストールおよび設定方法

2. Linux ファイルシステムを理解する

3. *コマンドラインから基本的な Linux コマンドを発行する*

4. *ファイルアクセス権限の理解*

5. *X Window System のカスタマイズ*

6. *正規表現パターンの一致と I/O リダイレクトの使用*

56.1.5. フォローオンフォル

RH133 Red Hat Linux Sys。admin. (8P)

RH253 Red Hat Linux Net. and Sec. Admin. (p. 9)

RH300、Red Hat Linux II Track (p. 10)

IT Consultant、Bill Legge、Brving Legge がこの業界では、このトレーニング体験を率いています。

第57章 RH133: RED HAT LINUX SYSTEM ADMINISTRATION AND RED HAT CERTIFIED TECHNICIAN (RHCT) CERTIFICATION

http://www.redhat.com/training/rhce/courses/rh133.html

57.1. 当然の説明

RH133 は、Red Hat Linux におけるシステム管理のスキルに重点を置いており、既存のネットワークにワークステーションをアタッチおよび設定できるレベルに重点を置いています。この 4.5 日間の試験では、Red Hat Enterprise Linux に関する実践的トレーニングを集中的に実施し、最後の日に RH202 RHCT 認定ラボ試験を掲載しています。

57.1.1. 前提条件

RH033 Red Hat Linux Essentials または Red Hat Linux の同等の経験。

57.1.2. 目的

この演習を問題なく完了すると、基本となる Linux システム管理者の知識が必要ですが、この知識は RHCT 認定に合格することで証明できます。この試験は、新しい Red Hat Linux システムを既存の実稼働ネットワークにインストール、設定、およびアタッチする実際の能力をテストする、実務ベースのラボ試験です。

57.1.3. 対象者

Linux または UNIX のユーザー(Red Hat Linux の基本を理解し、システム管理者になるプロセスを開始するために)

57.1.4. 学習目的

1. Red Hat Linux の対話的なインストールとキックスタートによるインストール
2. 一般的なシステムハードウェアの制御 - Linux 印刷サブシステムの管理
3. Linux ファイルシステムの作成および維持

4. *ユーザーおよびグループの管理の実行*

5. *ワークステーションと既存ネットワークの統合*

6. *NIS、DNS、および DHCP サービスに対するクライアントとしてワークステーションを設定する*

7. *at、cron、および anacron を使用してタスクを自動化します。*

8. *テープおよび tar アーカイブへのファイルシステムのバックアップ*

9. *RPM によるソフトウェアパッケージを操作します。*

10. *X Window System および GNOME d.e を設定します。*

11. *パフォーマンス、メモリー、およびプロセス mgmt を実行します。*

12. *基本的なホストセキュリティーの設定*

57.1.5. フォローオンフォル

RH253 Red Hat Linux Net. and Sec. Admin. (p. 9)

第58章 RH202 RHCT EXAM - すべての LINUX で最も急速に広がった認証情報

http://www.redhat.com/training/rhce/courses/rh202.html

1. RHCT試験はRH133に含まれています。$349 用に独自に購入することもできます。

2. RHCT の試験は、 すべての RH133 クラスの 5 日目に実施

58.1. 当然の説明

RHCT (Red Hat Certified Technician)は、 Red Hat Enterprise Linux のインストール、設定、およびトラブルシューティングにおける実際のスキルをテストする実践的で実一ベース試験の試験です。認定ラボ試験は RH133 とバンドルされていますが、 RH033 と RH133 のコンテンツをマスターした個人は、試験のみを受けることができます。

58.1.1. 前提条件

試験の準備として RH033 および RH133 を取得することを検討する必要がありますが、この作業を行う必要はありません。

第59章 RH253 RED HAT LINUX NETWORKING AND SECURITY ADMINISTRATION

59.1. 当然の説明

RH253 アーマーは、一般的な Red Hat Enterprise Linux ネットワークサービスの設定に必要な詳しい知識を学ぶことができます。また、ネットワークおよびローカルセキュリティータスクは、この演習のトピックでもあります。

59.1.1. 前提条件

RH133 Red Hat Linux システム管理またはこれと同等の Red Hat Enterprise Linux システム管理 (LAN/WAN の基本または同等の機能)、TCP/IP との相互ネットワーク。

59.1.2. 目的

この手順を完了すると、個人は Red Hat Enterprise Linux サーバーを設定し、一般的なネットワークサービスとセキュリティーを基本レベルで設定できます。

59.1.3. 対象者

Linux または UNIX のシステム管理者ですでに Red Hat Enterprise Linux システム管理の実務経験があり、ネットワークサービスとセキュリティーにおける最初の学習を希望し、Red Hat Enterprise Linux を使用して一般的なネットワークサービスおよびセキュリティー管理を設定する際のスキルの構築を望んでいます。

59.1.4. 学習目的

1. Red Hat Linux サーバー側のセットアップ、設定、および一般的なネットワークサービスの基本的な管理(DNS、NIS、Apache、SMB、DHCP、Sendmail、FTP)上のネットワークサービス。その他の一般的なサービス：tftp、pppd、proxy
2. セキュリティーの概要
3. セキュリティーポリシーの開発
4. ローカルセキュリティー

5. *ファイルおよびファイルシステムのセキュリティー*

6. *パスワードのセキュリティー*

7. *カーネルセキュリティー*

8. *ファイアウォールの基本要素*

9. ***Red Hat Linux** ベースのセキュリティーツール*

10. *一休みの試みへの対応*

11. *セキュリティーソースおよびメソッド*

12. ***OSS** セキュリティーツールの概要*

59.1.5. フォローオンフォル

***RH302**の認定試験*

"これはすばらしいです。教師は、知識の深さが大きくなっていました。"--Greg Peters, Future Networks USA

第60章 RH300: KNOWLEDGE COURSE (RH300: TRACK COURSE) （および----------|-----試験）

経験のある UNIX/Linux ユーザー向けの認定への最速パス。

http://www.redhat.com/training/rhce/courses/rh300.html

60.1. 当然の説明

Red Hat Linux では、この 5 日間の実践的トレーニングを行います。これには、過去 1 日の認定試験が含まれます。

60.1.1. 前提条件

RH033、RH133、RH253、または UNIX での同等のエクスペリエンス。システム管理に経験がある場合、または UNIX または Linux 環境のパワーユーザーでない限り、RH300 には登録しないでください。

60.1.2. 目的

この手順を完了すると、個人は Red Hat Linux システム管理者になり、トレーニングを受けてから、JBCD試験を使用してテストしたことになります。

60.1.3. 対象者

実務経験があり、速やかな試験で試験の準備を望んでいる UNIX または Linux のシステム管理者。

60.1.4. 学習目的

1. ハードウェアおよびインストール(x86 アーキテクチャー)
2. 設定および管理
3. 代替インストール方法

4. *カーネルサービスおよび設定*

5. *標準のネットワークサービス*

6. *X ウィンドウシステム*

7. *ユーザーとホストのセキュリティー*

8. *ルーター、ファイアウォール、クラスター、およびトラブルシューティング*

60.1.5. フォローオンフォル

エンタープライズアーキテクトと認定

第61章 RH302 RHCE EXAM

1. 試験は RH300 に含まれています。独自に購入することもできます。

2. すべての RH300 クラスの 5 日目に実施

http://www.redhat.com/training/rhce/courses/rhexam.html

61.1. 当然の説明

Red Hat Linux インストール、設定、デバッグ、および主要なネットワークサービスのセットアップにおける実際のスキルの実践、パフォーマンスベースのテストに重点を置いているため、IT セクターの他の多くの認定プログラムとは別になっています。

61.1.1. 前提条件

RH300 の前提条件 を参照してください。詳細は、www.redhat.com/training/rhce/examprep.html を参照してください。

61.1.2. コンテンツ

1. セクション I: トラブルシューティングとシステムメンテナーンス(2.5 hrs)

2. セクション II: インストールと設定(3 クリス)

つまり、これは未処理のクラスでした。テストのための非常に準備が整っています。" Logan Ingalls, Web developer, Texterity Inc., USA

第62章 RHS333: RED HAT のエンタープライズセキュリティー：ネットワークサービス

最も一般的にデプロイされるサービスのセキュリティー。

http://www.redhat.com/training/architect/courses/rhs333.html

62.1. 当然の説明

Red Hat Enterprise Linux は、Web、ftp、電子メール、ファイル共有などのネットワークサービスをデプロイするためのオペレーティングシステムとして、大幅に近々見られました。Red Hat では、これらのサービスをデプロイする際のトレーニングと、そのサービスを保護するための重要な要素を提供しています。

62.1.1. 前提条件

本手順では、RH253、RH300、または同等の実務経験が必要です。当然の参加者は、対象のサービスの設定方法に関する重要な要素をすでに理解している必要があります。このクラスはアウトセットのより高度なトピックに焦点を当てるためです。

62.1.2. 目的

このクラスは、最も一般的にデプロイされたサービスに関連するセキュリティー機能、機能、およびリスクを深く掘り下げるセキュリティー機能、機能、およびリスクを深めることで、セキュリティーに関する重要な範囲以上の改良が加えられています。

62.1.3. 対象者

このクラスでは、ネットワークサーバーの計画、実装、および保守を担当するシステム管理者、導入、およびその他の IT アーキテクトが含まれます。Red Hat Enterprise Linux ではこれらのサービスの実行に重点を置いていますが、コンテンツとラボではその用途を想定していますが、独自の形式の UNIX を使用するシステム管理者やその他の人は、このことに関連する多くの要素を見つける可能性があります。

62.1.4. 学習目的

1. 基本的なサービスセキュリティーのマスター

2. *暗号化について*

3. *システムアクティビティーのロギング*

4. *BIND および DNS のセキュア化*

5. *ネットワークユーザー認証のセキュリティー*

6. *NFS セキュリティーの強化*

7. *セキュアシェル：OpenSSH*

8. *Sendmail および Postfix による電子メールのセキュア化*

9. *FTP アクセスの管理*

10. *Apache セキュリティー*

11. *侵入応答の基本*

62.1.5. フォローオンフォル

RH401 Red Hat Enterprise Deployment and System MgmtRH423 Red Hat Enterprise Directory Services and Authentication RH436 Red Hat Enterprise Storage MgmtRH442 Red Hat Enterprise System Monitoring and Performance Tuning

第63章 RH401: RED HAT エンタープライズ 導入およびシステム管理

Red Hat Enterprise Linux デプロイメントを管理します。

http://www.redhat.com/training/architect/courses/rh401.html

63.1. 当然の説明

RH401 は、ミッションクリティカルな Red Hat Enterprise Linux システムの大規模なデプロイメントおよび管理に不可欠なスキルとメソッドの 4 日間の実践ラボです。これには、システム管理者、RPM 再構築、特定のアプリケーション向けの CVS など、ミッションクリティカルな Red Hat Enterprise Linux システムのデプロイメントおよび管理に欠かせません。

63.1.1. 前提条件

RH253 は、最低限の認定、または同等のスキルと知識を推奨しています。---|---------- 認定のない試験の方針はすべて、Red Hat の無料のオンライン事前評価テストでスキルを確認することをお勧めします。注記：上記の前提条件を満たすことなく、Persons は RH401 に登録しないでください。

認定を受けていない将来のすべての参加者は、登録時に Red Hat Global learning Services に連絡してスキルチェックを行うことを強くお勧めします。

63.1.2. 目的

RH401 では、上級システム管理者を対象に、さまざまなロールで多数の Enterprise Linux サーバーを管理したり、フェイルオーバーと負荷分散を必要とするミッションクリティカルなアプリケーション向けに管理したりします。さらに、RH401 は、エンタープライズロール向けのオペレーティングシステムの管理におけるエキスパートレベルの能力に関するベンチマークです。エンタープライズ Red Hat Enterprise Linux デプロイメントを効率的かつ効果的に実装および管理する方法を、チームによって管理可能な方法で効率的に実装および管理するための方法について説明します。

63.1.3. 対象者

エンタープライズ環境やミッションクリティカルなシステムで作業する Red Hat Enterprise Linux システム管理者およびその他の IT 上級システム管理者。

63.1.4. 学習目的

1. *CVS を使用した設定管理*
2. *カスタム RPM パッケージの構築*
3. *Red Hat Network Proxy Server によるソフトウェア管理*
4. *ホストのプロビジョニングおよび管理システムのアセンブル*
5. *パフォーマンスチューニングおよび分析*
6. *高可用性ネットワーク負荷分散クラスター*
7. *高可用性アプリケーションのフェイルオーバークラスター*

63.1.5. フォローオンフォル

RHS333 エンタープライズセキュリティー：ネットワークサービスのセキュリティー保護

RH423 Red Hat Enterprise Directory Services and Authentication

RH436 Red Hat Enterprise Storage Mgmt.

RH442 Red Hat Enterprise System Monitoring and Performance Tuning

"その後、RH401 は、エンタープライズ規模の高可用性ソリューションをエンドツーエンドで実装できることを完全に保証しました。"Bunge North America

第64章 RH423: RED HAT ENTERPRISE DIRECTORY サービスと認証

Red Hat Enterprise Linux システムのディレクトリーサービスの管理およびデプロイ

http://www.redhat.com/training/architect/courses/rh423.html

64.1. 当然の説明

RH423 は、ディレクトリーサービスのクロスプラットフォーム統合に関する 4 日間の手順とラボを提供し、企業全体で認証または情報サービスを提供します。

64.1.1. 前提条件

RH253 は、最低限の認定、または同等のスキルと知識を推奨しています。---|---------- 認定のない試験の方針はすべて、Red Hat の無料のオンライン事前評価テストでスキルを確認することをお勧めします。注記：上記の前提条件を満たすことなく、RH423 に登録しないでください。認定を受けていない将来のすべての参加者は、登録時に Red Hat Global learning Services に連絡してスキルチェックを行うことを強くお勧めします。

64.1.2. 目的

RH423 は、上級システム管理者を対象に、Red Hat Enterprise Linux システムのディレクトリーサービスを管理およびデプロイするための上級システム管理者です。LDAP ベースのサービスの基本的な概念、設定、管理について理解しておくことは、この演習の中心となります。学生は、標準のネットワーククライアントとサービスをディレクトリーサービスと統合して、その機能を活用します。また、PAM、プラグ可能な認証モジュールシステム、および認証および承認を必要とするサービスとの統合方法についても説明します。

64.1.3. 対象者

エンタープライズ環境やミッションクリティカルなシステムで作業する Red Hat Enterprise Linux システム管理者およびその他の IT 上級システム管理者。

64.1.4. 学習目的

1. LDAP の基本概念

2. *OpenLDAP サーバーの設定および管理方法*

3. *Using LDAP as a "white pages" directory service*

4. *ユーザー認証および管理での LDAP の使用*

5. *複数の LDAP サーバーの統合*

64.1.5. フォローオンフォル

RHS333 エンタープライズセキュリティー：ネットワークサービスのセキュリティー保護

RH401 Red Hat エンタープライズ 導入およびシステム管理

RH436 Red Hat Enterprise Storage Mgmt. (p. 16)

RH442 Red Hat Enterprise System Monitoring and Performance Tuning

第65章 SELINUX で

65.1. RHS427: SELINUX と RED HAT TARGETED ポリシーの概要

http://www.redhat.com/training/security/courses/rhs427.html

1 日間の SELinux の概要、Red Hat ターゲットポリシー内でどのように動作するか、およびこの強力な機能を操作できるツール。RHS427 は RH429 を初めて設定します。

65.1.1. 対象者

コンピューターのセキュリティー専門家や、Linux コンピューターへのセキュリティーポリシーの実装を担当するユーザー。RHS429 には、または同等の知識が必要です。

65.1.2. 当然サマリー

Red Hat Enterprise Linux の最も重要な機能には、SELinux (Security Enhanced Linux)があります。これは、強力なカーネルレベルのセキュリティーレイヤーで、システム上でアクセスして実行できるユーザーとプロセスを詳細に制御できます。デフォルトでは、Red Hat Enterprise Linux システムで SELinux が有効になり、Red Hat がターゲットポリシーを呼び出す必須のアクセス制御のセットを強制します。これらのアクセス制御は、対象とするネットワークサービスのセキュリティーを大幅に強化しますが、以前のバージョンの Red Hat Enterprise Linux で動作していたサードパーティーのアプリケーションやスクリプトの動作に影響を及ぼす場合があります。

65.2. RHS429: RED HAT ENTERPRISE SELINUX ポリシー管理

http://www.redhat.com/training/security/courses/rhs429.html

Red Hat Enterprise Linux の最も重要な機能には、SELinux (Security Enhanced Linux)があります。これは、強力なカーネルレベルのセキュリティーレイヤーで、システム上でアクセスして実行するユーザーとプロセスを詳細に制御できます。RHS429 では、SELinux ポリシーの記述に高度なシステム管理者、セキュリティー管理者、およびアプリケーションプログラマーを紹介します。この形式の参加者は、SELinux の仕組み、SELinux の管理方法、および SELinux ポリシーの作成方法を学ぶことができます。

第66章 RH436: RED HAT ENTERPRISE STORAGE MANAGEMENT

Red Hat のクラスターファイルシステム技術をデプロイおよび管理します。

機器集約：

1. *5 つのサーバー*
2. *ストレージアレイ*

http://www.redhat.com/training/architect/courses/rh436.html

66.1. 当然の説明

RH436 は、Red Hat Global File System (GFS)が提供する先進の共有ストレージ技術に関するハンズオンエクスペリエンスを集中的に提供します。この 4 日間の試験では、Red Hat Cluster Suite および GFS に含まれるネイティブの Red Hat Enterprise Linux テクノロジーの実装に重点を置いています。

66.1.1. 前提条件

RH253 は、最低限の認定、または同等のスキルと知識を推奨しています。---|---------- 認定のない試験の方針はすべて、Red Hat の無料のオンライン事前評価テストでスキルを確認することをお勧めします。

66.1.2. 目的

この試験は、ミッションクリティカルなエンタープライズコンピューティング環境に可用性の高いストレージデータをデプロイおよび管理するために必要なスキルに基づいて、スキルを持つユーザーにトレーニングを行うように設計されています。RH401 で得られるスキルの補完は、クラスターファイルシステムである GFS を使用して、広範囲にわたる実践的トレーニングを提供します。

66.1.3. 対象者

エンタープライズ環境やミッションクリティカルなシステムで作業する Red Hat Enterprise Linux システム管理者およびその他の IT 上級システム管理者。

66.1.4. 学習目的

1. Red Hat Enterprise Linux ストレージ管理テクノロジーを確認する
2. データストレージ設計：データ共有
3. Cluster Suite の概要
4. Global File System (GFS)の概要
5. GFS 管理
6. オンラインの GFS 環境の変更：データ容量の管理
7. GFS の監視
8. GFS 変更を実装
9. クラススイート NFS の DAS から GFS への移行
10. GFS を使用したクラススイートの再検討

66.1.5. フォローオンフォル

RHS333 エンタープライズセキュリティー：ネットワークサービスのセキュリティー保護

RH401 Red Hat エンタープライズ 導入およびシステム管理

RH423 Red Hat Enterprise Directory Services and Authentication

RH442 Red Hat Enterprise System Monitoring and Performance Tuning

FBI - Operational Quantico, VA, USA

第67章 RH442: RED HAT ENTERPRISE システムの監視およびパフォーマンスチューニング

Red Hat Enterprise Linux のパフォーマンスチューニングとキャパシティプランニング

http://www.redhat.com/training/architect/courses/rh442.html

67.1. 当然の説明

RH442 は、システムアーキテクチャー、パフォーマンス特性、監視、ベンチマーク、およびネットワークパフォーマンスチューニングをカバーする 4 日間の高度な実践ラボです。

67.1.1. 前提条件

最低でも RHCT の認定、または同等のスキルと知識が必要です。---|---------- 認定のない試験の方針はすべて、Red Hat の無料のオンライン事前評価テストでスキルを確認することをお勧めします。

67.1.2. 目的

RH442 は、Red Hat Enterprise Linux のパフォーマンスチューニングおよびキャパシティプランニングの方法論を学ぶように設計されています。このクラスは以下を対象とします。

1. システムアーキテクチャーとシステムパフォーマンスのシステムアーキテクチャーの影響を理解することに重点を置いた説明
2. パフォーマンス調整の影響をテストする方法(benchmarking)
3. オープンソースのベンチマークユーティリティー
4. システムのパフォーマンスおよびネットワークパフォーマンスを分析する方法
5. 特定のアプリケーション負荷に対する設定のチューニング

67.1.3. 対象者

RH442 は、エンタープライズ環境やミッションクリティカルなシステムで作業する、Red Hat Enterprise Linux の上級システム管理者およびその他の IT アーキテクトを対象としています。

67.1.4. 学習目的

1. *システムパフォーマンスに関連するシステムコンポーネントおよびアーキテクチャーの概要*
2. *製造元のハードウェア仕様を有用な情報に変換する*
3. *標準の監視ツールを効果的に使用してトレンド情報を収集および分析*
4. *SNMP でのパフォーマンス関連のデータの収集*
5. *オープンソースのベンチマークユーティリティーの使用*
6. *ネットワークパフォーマンスチューニング*
7. *アプリケーションのパフォーマンスチューニングに関する考慮事項*
8. *特定の設定のチューニング*

67.1.5. フォローオンフォル

RHS333 エンタープライズセキュリティー：ネットワークサービスのセキュリティー保護

RH401 Red Hat エンタープライズ 導入およびシステム管理

RH423 Red Hat Enterprise Directory Services and Authentication

RH436 Red Hat Enterprise Storage Mgmt.

第68章 RED HAT ENTERPRISE LINUX 開発者

68.1. RHD143: RED HAT LINUX PROGRAMMING ESSENTIALS

http://www.redhat.com/training/developer/courses/rhd143.html

Red Hat Enterprise Linux でアプリケーションやプログラムを開発するための主要なスキルで、スタッフの迅速なトレーニングを目的に設計された、集中的な実践的試験です。この 5 日間の試験では、実際のラボやプログラミングの演習を中心とした、実践的なトレーニング、概念、デモンストレーションを行います。内容が終わると、Linux システムのプログラム開発に必要な基本スキルについて学び、実践します。

68.2. RHD221 RED HAT LINUX デバイスドライバー

http://www.redhat.com/training/developer/courses/rhd221.html

これは、経験のあるプログラマーの Linux システム用のデバイスドライバーの開発方法を教えることを目的としています。内容を完了したら、Linux のアーキテクチャー、ハードウェアとメモリーの管理、モジュール化、カーネルソースのレイアウトを理解して、文字、ブロック、ネットワークドライバーの開発に必要な主要な概念とスキルを実践します。

68.3. RHD236 RED HAT LINUX KERNEL INTERNALS

http://www.redhat.com/training/developer/courses/rhd236.html

この試験は、プロセススケジューリング、メモリー管理、ファイルシステム、周辺機器など、Linux カーネルアーキテクチャーの詳細を調べることを目的としています。この 5 日間の試験では、実際のラボやプログラミングの演習を中心とした、実践的なトレーニング、概念、デモを提供します。

68.4. RHD256 RED HAT LINUX アプリケーション開発および移植

http://www.redhat.com/training/developer/courses/rhd256.html

UNIX ライクなシステムでの開発に精通し、新規アプリケーションの開発や Red Hat Enterprise Linux への移植を希望する、経験のあるプログラマー向けの 4 日間の開発者用です。

第69章 JBOSS 社

69.1. RHD161 JBOSS および EJB3 FOR JAVA

http://www.redhat.com/training/jboss/courses/rhd161.html

開発者 JBoss および EJB3 for Java Developers は、JBoss Application Server を使用した EJB3 および J2EE ミドルウェアプログラミングの知識を深めることを目的とする Java 開発者を対象としています。このクラスは、JBoss Application Server を使用した EJB3 および J2EE の詳細な概要です。EJB3 および J2EE アプリケーション開発、デプロイメント、および両方のプロセスを容易にするために必要なツールに対して、ハンズオンアプローチを提供します。

69.1.1. 前提条件

OOAD の概念に関する基本的な Java プログラミングスキルと知識が必要です。学生は、以下に関する実用的な知識や経験が必要です。

1. 継承、ポリモーフィズム、カプセル化のオブジェクト指向の概念
2. Java 構文（データタイプ、変数、演算子、ステートメント、制御フロー用）
3. Java クラスの作成、および Java インターフェイスおよび抽象クラスの使用

69.2. RHD163 JBOSS FOR WEB DEVELOPERS

http://www.redhat.com/training/jboss/courses/rhd163.html

JBoss for Web Developers は、JBoss Enterprise Middleware System (JEMS)製品スタックの Web 階層技術に重点を置いています。JBoss Portal の詳細、ポートレットを作成およびデプロイする方法、JavaServer Faces JSF などの他の Web 階層フレームワークとポートレットの統合、および JBoss Application Server に組み込まれた Tomcat Web コンテナーの設定およびチューニングについて説明します。JSP、サーブレットの開発と関連仕様について理解しておくことが強く推奨されます。Portlets や JSF に関するこれまでの経験は必要ありません。

69.2.1. 前提条件

このクラスの前提条件となるスキルは、基本的な J2EE Web Container (Servlet/JSP)プログラミングスキルです。また、JBoss Application Server 上の J2EE Web ベースおよびマルチ層アプリケーションのデプロイメントと、Tomcat コンテナー(Apache に組み込まれているか、JBoss Application Server と統合されているかに関わらず)の使用経験があります。学生は、以下のテクノロジーの開発経験があるはずです。

1. *JNDI*
2. *Servlet 2.3/2.4 API*
3. *The JSP 2.0 API*
4. *JBoss Application Server での J2EE アプリケーション開発およびデプロイメント*
5. *組み込み（スタンドアロン）Tomcat または統合された Tomcat (JBossWeb)への Web アプリケーションのデプロイメント*
6. *JDBC および EJB2.1 または EJB3.0 の実務知識*

前提条件ではありませんが、が便利です。

69.3. RHD167: JBOSS - HIBERNATE ESSENTIALS

http://www.redhat.com/training/jboss/courses/rhd167.html

69.3.1. 前提条件

1. *リレーショナル永続化モデルを理解している*
2. *Java 言語での競合*
3. *OOAD の概念に関する知識*

4. *UML に精通している*

5. *SQL のダイアレクトの使用経験*

6. *JDK を使用してコマンドラインから Java 実行可能ファイルのコンパイルおよび実行に必要な環境を作成する*

7. *JDB を理解していること*

J2EE または Hibernate の事前知識は必要ありません。このトレーニングは Hibernate 3.2 シリーズに基づいています。

69.3.2. 当然サマリー

Hibernate Essentials は、Hibernate または Java Persistence API オブジェクト/リレーショナル永続性およびクエリーサービス実装と競合する必要がある Java 開発者を対象にしています。主な対象者は、SQL ベースのデータベースシステムまたはオブジェクト指向ソフトウェア開発の概要を目的とするデータベース開発者である Java 開発者を対象としています。ORM がパフォーマンスにどのように影響するか、および SQL データベース管理システムと永続レイヤーのパフォーマンスを調整する方法に関心のあるデータベース管理者は、この機能も見つかります。このセクションでは、Java Persistence 向けの JSR-220 サブ仕様の JBoss, Inc. 実装について説明し、JBoss, Inc. のバージョン 3.x の基礎的な API を取り上げます。Hibernate 製品（または単に Hibernate 3）

69.4. RHD267: JBOSS - ADVANCED HIBERNATE

http://www.redhat.com/training/jboss/courses/rhd267.html

JBoss Advanced Hibernate のトレーニングは、Hibernate O/R マッピングフレームワークのフルパワーを抽出する Java 開発者を対象にしています。主なターゲットオーディエンスは、SQL ベースのデータベースシステムと連携する Java 開発者、ORM がパフォーマンスにどのように影響するか、および SQL データベース管理システムおよび永続レイヤーのパフォーマンスを調整する方法に関心のあるオブジェクト指向のソフトウェア開発およびデータベース管理者を対象にしています。このトレーニングでは、Hibernate 3 の新機能について説明します。

69.4.1. 前提条件

このクラスの前提条件となるスキルは以下のとおりです。

1. *Hibernate の基本知識。*

2. *Java 言語での競合*

3. *OOAD の概念に関する知識*

4. *UML に精通している*

5. *SQL のダイアレクトの使用経験*

6. *JDK を使用して、コマンドラインから Java 実行可能ファイルのコンパイルおよび実行に必要な環境を作成します。*

7. *JNDI および JDBC の豊富な経験または包括的な知識。*

8. *前提条件ではなく、エンティティー EJB2.1 または EJB3.0 の知識が役立ちます。*

9. *Christian Bauer および Gavin King (Manning によって公開)によると、Hibernate in Action を読む前に推奨されます。*

Mike Pasternak, Consulting Engineer, United Switch & Signal （アドバンドスイッチおよびシグナルエンジニア）は、Adiy Pasternak, Consulting Engineer, United Switch & Signal で、自分の問題に類似した経験があり、知識のあるインストラクターと協力していくらかのエンジニアがネットワークでした。

69.5. RHD261:JBOSS FOR ADVANCED J2EE DEVELOPERS

http://www.redhat.com/training/jboss/courses/rhd261.html

JBoss for Advanced J2EE Developers は、JBoss Application Server の内部アーキテクチャーを活用し、JBoss Application Server の J2EE アプリケーションの機能およびパフォーマンスを向上したい J2EE 開発者を対象としています。本手順では、JMX などのトピックと、マイクロカーネルアー

キテクチャー、セキュリティー、クラスターリング、および Fine Tuning などの J2EE 仕様以外のトピックについて説明します。

69.5.1. 前提条件

学生は、JBoss for Java Developers のトレーニングを行うか、JBoss for Advanced J2EE Developers (Advanced J2EE Developers)に登録する前にミドルウェア配置試験を取ることを強く推奨します。開発者は、以下のトピックごとに実用的な操作が必要です。

1. *JNDI*
2. *JDBC*
3. *サーブレットとJSP*
4. *Enterprise Java Beans*
5. *JMS*
6. *J2EE セキュリティーモデル*
7. *JBoss アプリケーションへの J2EE アプリケーションの開発およびデプロイ*
8. *ANT および XDoclet または同様のテクノロジーの使用経験。*

JMX に関する事前知識は役立ちますが、必須ではありません。このトレーニングは、JBoss Application Server 4.x シリーズに基づいています。

また、トレーニング資料は適切に編成されました（手引きとラボの両方を含む）。インストラクターは、資料とペースに関するフィードバックを求められています。これは、この資料の理解に注意したのは明らかでした。"--Jeremy Prellwitz, SiRAS.com, USA

69.6. RH336: 管理者向けの JBOSS

http://www.redhat.com/training/jboss/courses/rh336.html

69.6.1. 前提条件

Windows または Linux (Unix ベース)オペレーティングシステムに関する基本的な実務知識。学生は、以下の経験を持っている必要があります。

1. ディレクトリー、ファイルの作成、およびファイルストアへのアクセス権限の変更
2. JDK のインストール
3. オペレーティングシステム用の JAVA_HOME などの環境変数の設定
4. Java アプリケーションの起動と、Java アプリケーションを起動する OS 依存スクリプトの実行
5. Java アーカイブファイルの作成および拡張(jar ユーティリティー)

J2EE または JBoss Application Server の事前知識は必要ありません。ただし、XML 設定を使用した Java アプリケーションのサポートについてある程度理解していることが強く推奨されますが、強く推奨されます。

69.6.2. 当然サマリー

JBoss for Administrators は、JBoss アプリケーションサーバー(3.2.x および 4.x シリーズ)やアプリケーションサーバーにデプロイされたアプリケーションの設定と管理を証明するシステム管理者、設定管理、品質保証担当者など、アプリケーションをサポートします。

JBoss for Administrators のトレーニングは、授業とラボの両方のバランスをとっていました。トピックに関する知識を実性的かつ適用できると常に適しています。"-Thomas Skowronek, Palm Harbor Homes, USA

69.7. RHD439: JBOSS CLUSTERING

http://www.redhat.com/training/jboss/courses/rhd439.html

クラスターリングは、JBoss Enterprise Middleware System (JEMS)の高可用性サービスに焦点を当てた 4 日間のトレーニングです。JBoss Application Server は、レプリケーションとフェイルオーバーに JGroups および JBoss Cache を活用する方法、JGroups プロトコルスタックの設定、チューニング、実装の方法、独自のミドルウェアアプリケーション実装で JBoss Cache を活用する方法、および HTTP 負荷分散に mod_jk の使用および設定方法を説明します。また、HA-JNDI、HA-JMS、HA シングルトンなどの JBoss Application Server の高可用性サービスについて詳しく説明します。

69.7.1. 前提条件

この手順を完了する前に、JBoss for Advanced J2EE Developers を完了することが強く推奨されます。また、J2EE およびその他の Java ミドルウェア技術を使用して、少なくとも 18 カ月の実用的な開発経験を学生が推奨し、JBoss Application Server で実用的に経験があることが推奨されています。強固な Java プログラミングエクスペリエンス(3 年以上)が必要で、基本的な TCP/IP トピックを理解する必要があります。

学生は、次のスキルを持っている必要があります。

1. *JTA、トランザクション、Java 並行処理*
2. *EJB 2.1、JMS、信頼できるメッセージング技術*
3. *Apache httpd と mod_jk や mod_proxy への何らかの公開経験*
4. *JBoss AS マイクロカーネルと JMX に精通している*
5. *TCP/IP、UDP、マルチキャストについて*

JBoss for Administrators は非常に有益でした。弊社のインストラクターは、当然の方向性を維持しながら、質問（学生にとって非常に特殊な）に回答してすばらしい仕事をしました。手順で学んだことの適用に非常に満足しています。また、Arizona Statue Warehouse、米国

69.8. RHD449: JBOSS JBPM

http://www.redhat.com/training/jboss/courses/rhd449.html

69.8.1. 説明

JBoss jBPM のトレーニングは、ビジネスアナリストと密接に協力し、BPM エンジンとして jBPM を使用して、ビジネスプロセスを J2EE 環境へ取り入れるシステム管理者および開発者を対象にしています。さらに、JBoss jBPM のトレーニングでは、BPM ランドスケープ、エンジンの種類、バジズワードの位置について深く理解することができます。

学生は専門知識を実用的な方法で取得し、後に JBoss jBPM でビジネスプロセスの開発を開始する準備ができます。このトレーニングのもう 1 つの目的は、ワークフローエンジンを比較するための詳細な準備を提供することです。

69.8.2. 前提条件

1. 学生は、Hibernate アプリケーションの開発経験が必要です。学生は、Hibernate 用のシンプルなセッションファクトリーの設定方法、Hibernate Session およびトランザクション境界を利用する方法、および Hibernate オブジェクトで基本的なクエリーを実行する方法を知っている必要があります。

2. Java アプリケーション開発との競合

3. ワークフローとビジネスプロセスモデラー(BPM)の概念に対する以前の公開は必要ありません。

4. JBoss プラグインを使用した JBoss Eclipse または Eclipse IDE の使用が推奨されていますが、必須ではありません。

5. JUnit テストフレームワークの基本概念が推奨されます。

69.9. RHD451 JBOSS RULES

http://www.redhat.com/training/jboss/courses/rhd451.html

本試験では、Drools 3 (JBoss Rules 3.0)のコアエンジンと、ビジネスルールの管理に使用できるさまざまな手法と言語、およびルールエンジンを J2SE アプリケーションおよび J2EE アプリケーション

に埋め込む方法について説明します。これは、今後のルールのリリースを使用してルール管理に関する今後の取り組みを行う際の無料のトレーニングです。

69.9.1. 前提条件

1. *基本的な Java 競合*

2. *推論規則エンジンとスクリプトエンジンの設定についての理解*

3. *対象ルールの表示とデモが推奨されますが、必須ではありません。*

4. *Java EE 固有の経験は必須ではありませんが、Java EE との統合方法を学ぶ必要のある方は適切な経験が必要です。*

付録A 更新履歴

改訂 11-1 **Tue 30 Jun 2015** **Barbora Ančincová**

POODLE 脆弱性(CVE-2014-3566)に関する情報で本書を更新。

改訂 11-0 **Fri 12 Sep 2014** **Barbora Ančincová**

Resolve BZ#1121893: RHEL 5 Deployment Guide - Bonding Options - abouting max_bonds and debug options.

Resolve BZ#1104152: swapfile を生成するセキュアな方法は、本書で提案されています。

改訂 10-0 **Tue 01 Oct 2013** **Jaromír Hradílek**

resolve BZ#853938: RFE: proc File System: ドキュメントで /proc/<PID> へのアクセスを制限します。

resolve BZ#826891: RFE: Yum: RHEL インストール ISO イメージを使用して yum update でシステムをアップグレードする手順を指定します。

resolve BZ#961815: ドキュメント移行 mysql5.1->mysql5.5。

改訂 9-6 **Tue Jan 08 2013** **Jaromír Hradílek**

resolve BZ#810514: RFE: Ext4 ファイルシステムを文書化します。

resolve BZ#216687: RFE: Postfix - 仮想ユーザーメールボックス用の標準 FHS 準拠の場所。

resolve BZ#816177: RFE: Postfix MySQL map support.

resolve BZ#810512: MinorMod: Automated Tasks: デフォルトのログローテーション cronjob により、RHEV 共有ストレージ環境で問題が発生します。

resolve BZ#840000: Kdump hangs on CCISS module loading on RHEL 5.8 - HP Proliant DL380 G6 (Smartarray 410i)。

resolve BZ#852604: Kdump failed with intel_iommu=on.

resolve BZ#847292: MajorMod: Network Interfaces: Static routes and default gateway are interface-specific?

resolve BZ#821302: NFS 制限のドキュメントで、セキュアな nfs マウントが TCP/UDP で機能します。

resolve BZ#852372: Deployment Guide References $ISA in PAM section.

resolve BZ#713417: /root is labelled system_u:object_r:default_t:s0 after switching to MLS.

resolve BZ#821225: ドキュメントでは、kdump カーネル用に確保する RAM の容量がありません。

改訂 8-0 **Tue Feb 21 2012** **Jaromír Hradílek**

Resolve BZ#749948: [Release Notes and Deployment Guide] Migration tooling from RHN Classic to Cert-based RHN for RHEL 5

resolve BZ#718608: MinorMod: FTP: Missing text fragment in vsftpd configuration documentation.

resolve BZ#720387: MinorMod: The proc File System: Illogical parameter description.

resolve BZ#720860: Update Deployment (Guide) in RHEL5 Build Tree.

resolve BZ#760925: MinorMod: Network File System: NFS マウントの例(TCP 用)の非常に最適な timeo オプション。

resolve BZ#784754: MinorMod: Network Interfaces: typo - wrong tense in 15.3インターフェイス制御スクリプト。

resolve BZ#740916: MinorMod: kdump Crash Recovery Service: crashkernel パラメーターの説明が正しくありません。

resolve BZ#767105: incorrect default action in kdump part.

resolve BZ#714080: bonding のデバッグオプションは、/etc/sysconfig/network-scripts/ifcfg-bondX の BONDING_OPTS では使用できません。

Resolve BZ#769776: /etc/kdump.conf のデフォルトのシェルオプションのドキュメントを更新する必要があります。

resolve BZ#781441: /etc/securetty documentation is incorrect [rhel-5.7].

改訂 7-0 **Thu Jul 21 2011** **Jaromír Hradílek**

Resolve BZ#720382: MinorMod: Network Interfaces: LINKDELAY パラメーターを "Interface Configuration Files" に追加する必要があります。
Resolve BZ#632028: MajorMod: Redundant Array of Independent Disks (RAID): Document mdadm Usage.
resolve BZ#720009: MinorMod: LVM: "Manual LVM Partitioning" セクションのスクリーンショットを更新します。
resolve BZ#711162: MinorMod: Network Interfaces: 静的ルートの設定が正しくありません。
resolve BZ#707238: broadcast is calculated with ipcalc, not ifcalc.
resolve BZ#678316: HOTPLUG network config file option is not documented.
resolve BZ#562018: Ch.4 Redundant Array of Independent Disks (RAID)- スクリーンショットの更新が必要。
resolve BZ#485033: iptables -p ALL --dport not allowed according to man 8 iptables.

改訂 6-0 **Thu Jan 13 2011** **Jaromír Hradílek**

resolve BZ#249485: 'fsid=num' is listed under NFS client options, but it is a server-only option.
resolve BZ#253659: マシンをドメインに追加する際に必要な追加コマンド。
resolve BZ#453242: guide では、NFS サーバーの実行に必要なパッケージについては説明していません。
resolve BZ#504250: cell should have newline characters, it't all on one line.
resolve BZ#520650: /proc/loadavg documentation error.
resolve BZ#584075: vsftp mis for text_userdb_names.
resolve BZ#625384: bonding configuration SLAVE=bond0 is invalid.
resolve BZ#644617: misspelled word.
resolve BZ#645123: spelling Errors in Deployment Guide II.
resolve BZ#595366: RFE: document Shared Subtrees.

改訂 5-0 **Thu July 30 2010** **Douglas Silas**

resolve BZ#239313: document oom_adj and oom_score.
resolve BZ#526502: correct quotaon instructions with appropriate, safe operating procedure.
resolve BZ#551367: correct SELinux dhcpd_disable_trans description.
resolve BZ#521215: portmapper、rpc.mountd、rpc.lockd、および rpc.statd との NFS の対話を明確にします。
resolve BZ#453875: さまざまな OpenSSH 章の修正。
resolve BZ#455162: correct zone example configuration file, description.
resolve BZ#460767: 適切なデーモンにします。
resolve BZ#600702: SSL キーの生成に使用される正しいディレクトリー。

改訂 4-2 **Wed Sep 30 2009** **Douglas Silas, Jaromír Hradílek, Martin Prpic**

見出しのタイトルを、man rpm で使用される実際の見出しに対応するように変更します。
resolve BZ#499053: /usr/sbin/racoon is correct install path.
BZ#237773 に従って、/etc/yum.conf の 'pkgpolicy' の参照をすべて削除します。
resolve BZ#455162: record, description に関する正しいゾーンファイルの例。
resolve BZ#510851: /proc/cmdline には混乱のある出力例の説明があります。
BZ#510847: 複数の footnotes 形式のページがオンライン PDF で正しくフォーマットされない。
resolve BZ#214326: vsftpd バナーとセキュリティーに関するより詳細な使用情報。
resolve BZ#241314: screen 要素での問題のフォーマット。
resolve BZ#466239: postfix connect-from-remote-host configuration fix.

改訂 4-1 **Mon Sep 14 2009** **Douglas Silas**

Resolve BZ#214326: Server Security FTP Banner instructions: questions re: vsftpd.conf.
resolve BZ#466239: Postfix 設定ファイルに行を挿入して、リモートで接続できるようにします。
resolve BZ#499053: path for racoon daemon is /usr/sbin/racoon, not /sbin/racoon.
resolve BZ#510847: missing footnotes in PDF output.
resolve BZ#510851: rewrite /proc/cmdline minor section to make more sense.
resolve BZ#515613: RHEL5 GPG キーおよびキーの詳細の正しい場所。
resolve BZ#523070: various minor fixes; --redhatprovides to rpm -q --whatprovides.

改訂 4-0 **Wed Sep 02 2009** **Douglas Silas**

resolve BZ#492539: "This directive is useful..." to "This directive must be used in machine containing multiple NIC to ensure...".
resolve BZ#241314: re: kernel-pae and hugemem support on RHEL 4 and 5。
解決 BZ#453071: タグの使用が正しくないと、設定ファイルやその他の画面要素が単一行に表示されました。
resolve BZ#507987: サイズ変更または削除中に使用中のパーティションに関する説明を明確にして修正します。
resolve BZ#462550: recommended amount of swap space, according to follow
resolve BZ#466239: Postfix configuration meant from Postfix configuration meant connecting remote failed
他の MODIFIED BZs （以前修正）: 468483、480324、481246、481247、438823、454841、485187、429989、452065、453466。

改訂 3-0 **Wed Jan 28 2009** **Michael Hideo Smith**

解決策：#460981
16GB に対応するように 64GB *テスト済み* のサポートを変更します。

付録B コロンフィン

このマニュアルは DocBook XML v4.3 形式で記述されています。

Garrett LeSage は、許可されたグラフィックスを作成しました（注記、ヒント、重要、注意、警告）。Red Hat ドキュメントと自由に再配布できます。

Writers の貢献：John Ha (System Administration、Filesystems、Kernel)、Joshua Wulf （インストールおよびブート）、Fask Cleary （仮想化）、David O'Brien (Security and SELinux)、NORMAL Hideo (System Administration)、Don Domingo (System Administration)、Don Domingo (System Administration)、Paul Kennedy （ストレージ）、Mellissa Goldin (Red Hat Network)

Sandra Moore, Edward C. Bailey, Karsten Wade, Mark Johnson, Andrius Benokraitis, Lucy Ringland の前経験を尊重する

エンジニアリング作業の尊重：Jeffrey Fearn

テクニカル編集：video Behm

グラフィックアーティスト：Andrew Fitzsimon

Red Hat ローカリゼーションチームは、以下のユーザーで設定されています。

- east Asian Languages
  - 簡体字中国語
    - Tony Tongjie Fu
    - Simon Xi Huang

- *Leah Wei Liu*
- *Sarah Saiying Wang*

- *繁体字中国語*
  - *Chester Cheng*
  - *Terry Chuang*
  - *Bu Hung-Pin Wu*
- *日本語*
  - *Kiyoto Hashida*
  - *Junko Ito*
  - *Noriko Mizumoto*
  - *Takuro Nagamoto*
- *韓国語*
  - *Eun-ju Kim*
  - *Michelle Kim*

- *Latin 言語*
  - *フランス語*
    - *Jean-Paul Aubry*
    - *Fabien Decroux*
    - *Myriam Malga*
    - *Audrey Simons*
    - *Corina Roe*
  - *ドイツ語*
    - *Jasna Dimanoski*
    - *Verena Furhuer*
    - *Bernd Groh*
    - *Daniela Kugelmann*
    - *Timo Trinks*
  - *イタリア語*

        - ***Francesco Valente***

    - ***ブラジルポルトガル語***

        - ***Glaucia de Freitas***

        - ***Leticia de Lima***

        - ***David Barzilay***

    - ***スペイン語***

        - ***Angela Garcia***

        - ***Gladys Guerrero***

        - ***Yelitza Louze***

        - ***Manuel Ospina***

    - ***ロシア語***

        - ***Yuliya Poyarkova***

- ***Indic 言語***

    - ***Bengali***

- *Runa Bhattacharjee*

- *グjarati*
  - *Ankitkumar Rameshchandra Patel*
  - *Sweta Kothari*
- *Hindi*
  - *Rajesh Ranjan*
- *マレーナム*
  - *Ani Peter*
- *Marathi*
  - *Sandeep Shedmake*
- *Punjabi*
  - *Amanpreet Singh Alam*
  - *Jaswinder Singh*
- *Tamil*

- *I Felix*

- *N Jayaradha*