

第1章

Linuxの基本と最新情報にふれる
Linux カーネル 2.4 の概要と
カーネルの構築

岸 哲夫

はじめに

一般にLinuxというと、Red Hat社やTurbolinux Japan社といったディストリビュータが販売しているパッケージを指すことが多いようですが、本来はLinus Torvalds氏が開発したLinux Kernelのことです。

Linux Kernelは、表1に示したような変遷をたどり、現在は2.4がリリースされています。本章では、この最新のカーネル2.4の新しい機能を紹介するとともに、このカーネル2.4を導入する方法について説明します。

なお、2002年5月現在の最新カーネルは2.5.14で、新しい機能としてUSB2.0をサポートしています。

最近のLinuxパッケージには、インストールを自動で行ってくれるセットアッププログラムが付属しているので、PCで使用するだけならカーネル構築の方法を知っている必要はないかもしれませんが、組み込み機器に使うような場合にはカーネルの構築は必須となります。

Linux カーネルとは

広義のLinuxを構成する要素を大きく分けると次のようになります。

〔リスト1〕LILOの設定ファイル(/etc/lilo.conf)

```
[root@linux kishi]# cat /etc/lilo.conf
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
lba32
timeout=50
default=linux
image=/boot/vmlinuz
  label=linux
  root=/dev/hda6
  initrd=/boot/initrd
  read-only
[root@linux kishi]#
```

- カーネル
 - メモリ管理、プロセスの管理など
- ドライバ
 - あらゆる種類のデバイスにアクセスするインターフェース
- デーモン
 - バックグラウンドで動作するサービス
- シェル
 - ユーザーインターフェース
- コマンド
 - ユーザーが目的の処理を行うための機能
 - どのようなOSにも、最初に主記憶上にロードして実行されるソフトウェアが存在します。Linuxの場合、それをカーネルと呼び、それがLinuxの本体そのものです。

一般に、LinuxはLILOというブートローダによって主記憶上にロードされ、実行されます。筆者の環境はTurbolinux Workstation日本語版6.0ですが、LILOの設定ファイルは/etc/lilo.confに指定されています(リスト1)。

このリストの中でimage = /boot/vmlinuzとありますが、これがカーネル、つまりはLinuxの本体です。そして、/bootの下にはカーネルの関連ファイルがあります(リスト2)。この場合はvmlinuz-2.2.16がvmlinuzにリンクされています。ここからカーネルのバージョンが2.2.16であることがわかります。

また、カーネルのソースコードは/usr/src/linuxにあります。一般的なオプションでインストールすれば、そこにソース一式が入っているはずですが。

カーネルから見た場合、カーネルがメインルーチンならばLinux上で実行されているすべてのタスクはそこから呼び出されることとなります。それぞれのタスクはお互いのタスクの資源にアクセスし、消費し、改

〔表1〕Linuxカーネルの変遷(年表)

1991年6月	Linus Torvalds氏によってMINIXを元にしたLinuxの原型が完成 comp.os.minix ニュースグループにてカーネルの性能や開発方針などについての論争があった	1999年5月	Linux Kernel 2.2.8 リリース CyrixやAMD社のプロセッサに対応 この版には「ファイルシステムのバグ」が存在したため、すぐに2.2.9がリリースされた 開発版カーネルとして2.3.0から2.3.3が公開された
1991年9月	Linux Kernel 0.01 公開 参考URL： http://www.vision25.demon.co.uk/prog/linuxbirthday.html	1999年6月	Linux Kernel 2.2.10 リリース ネットワーク経由のDoS攻撃に対処した 開発版カーネル2.3.4から2.3.9まで公開された
1991年10月	Linux Kernel 0.02 リリース ソースだけでなく、バイナリもリリースされた。まだMINIXを必要とした	1999年7月	開発版カーネル2.3.10から2.3.12まで公開された
1991年10月	Linux Kernel 0.03 リリース Linux上でgccを使用することが可能になった	1999年8月	Linux Kernel 2.2.11, 2.2.12 リリース 開発版カーネル2.3.13から2.3.15まで公開された
1991年11月	Linux Kernel 0.10 リリース MINIXなしで動くようになった	1999年9月	開発版2.3シリーズのカーネルは2.3.18で機能の追加は終了とLinus Torvalds氏が発表 新機能は2.4に盛り込まれることになる
1991年12月	Linux Kernel 0.11 リリース デマンドページング, フロッピードライバ, VGAドライバ, mkfs/fsck/fdiskなどが機能した	1999年10月	Linux Kernel 2.2.13 リリース 開発版カーネル2.3.19から2.3.24まで公開された
1992年1月	Linux Kernel 0.12 リリース POSIX準拠のジョブコントロール, 仮想記憶の機能が追加された	1999年11月	開発版カーネル2.3.25から2.3.27まで公開された
1992年3月	Linux Kernel 0.95 リリース BSDもこの時期に原型が完成した	1999年12月	開発版カーネル2.3.28から2.3.35まで公開された
1992年4月	Linux Kernel 0.96 リリース	2000年1月	Linux Kernel 2.2.14 リリース AMD Athlon プロセッサ用のチップセットである「AMD-751」のサポートやインテルCoppermineプロセッサのL2キャッシュに関するバグフィクスがなされた 開発版カーネル2.3.36から2.3.41まで公開された
1992年8月	Linux Kernel 0.97 リリース	2000年2月	開発版カーネル2.3.42から2.3.48まで公開された
1992年9月	Linux Kernel 0.98 リリース	2000年3月	開発版カーネル2.3.49から2.3.99まで公開された
1992年10月	Linux Kernel 0.99 リリース この頃より日本のユーザーの間で日本語化への模索が始まる	2000年5月	Linux Kernel 2.2.15 リリース 開発版カーネル2.4.0-test1がリリースされた 本章で説明するカーネル2.4の原型である
1994年2月	Linux Kernel 0.99.15 リリース 静かにユーザーの間で話題になっていった	2000年6月	Linux Kernel 2.2.16 リリース 任意のコードを実行できるユーザーがsetcap(2)システムコールを通してroot権限を奪取できるという2.2.15のセキュリティホールが修正された 開発版カーネル2.4.0-test2がリリースされた
1994年3月	Linux Kernel 1.0 リリース 最初の公式版カーネルである	2000年7月	開発版カーネル2.4.0-test3から2.4.0-test5まで公開された 2.4.0-test5でFATファイルシステムのファイル名をeuc-jpやshift_jisに変換する機能が追加された 簡単にいえば、日本語ファイル名を含むWindowsのファイルシステムをLinuxで扱うことが楽になった
1994年4月	Linux Kernel 1.1.0 リリース マイナーバージョンが偶数のバージョンは安定性を重視した一般用, 奇数のバージョンは機能追加を目指す開発用と位置づけられる	2000年8月	開発版カーネル2.4.0-test6から2.4.0-test7まで公開された
1995年3月	Linux Kernel 1.2.0 リリース このころより数多くの雑誌や書籍がLinuxを取り上げ始めた	2000年9月	Linux Kernel 2.2.17 リリース 開発版カーネル2.4.0-test8から2.4.0-test8まで公開された
1996年5月	Linux Kernel 2.0 リリース このころより一般雑誌にも取り上げられるなどパーソナルユースとしてLinuxが多くのユーザーに使われ始める。 この間、無数のマイナーバージョンアップ、パッチがいろいろなディストリビュータから配布される	2000年10月	開発版カーネル2.4.0-test9から2.4.0-test10まで公開された
1998年12月	Linux Kernel 2.0 2.2.0-pre-patch1 リリース マルチプロセッサに対応した。 このころMkLinux kernel GENERIC-06がリリースされた	2000年11月	開発版カーネル2.4.0-test11まで公開された
1999年1月	Linux Kernel 2.2.0 リリース この版には「セキュリティホール」が存在したため、すぐに2.2.1がリリースされた このころOracleが「Oracle8 Workgroup Server for Linux Release8.0.5」を出荷した	2000年12月	Linux Kernel 2.2.18 リリース 開発版カーネル2.4.0-test12まで公開された
1999年2月	Linux Kernel 2.2.2 リリース	2001年1月	Linux Kernel 2.4 リリース バグフィックス版2.4.1もリリースされた
1999年3月	Linux Kernel 2.2.3, 2.2.4, 2.2.5 リリース	2001年2月	Linux Kernel 2.4.2 リリース IBMの汎用機 S/390への対応, Reiser FSの改良, USB関連の改良などが行われた レッドハット社がカーネル2.4搭載のRed Hat Linux 7.0.9公開 版『Fisher』を国内配布
1999年4月	Linux Kernel 2.2.6, 2.2.7 リリース		

〔表1〕Linuxカーネルの変遷(年表, つづき)

2001年3月	Linux Kernel 2.4.3リリース Windows/Turbolinuxデュアルブート可能なThinkPadの販売が開始された 仏MandrakeSoftよりLinux-Mandrake 8.0 Beta 1がリリースされた。これはカーネル2.4.2を採用している プレイステーション2用Linuxの公開を囁願するWebページが登場した	2001年7月	Linux Kernel 2.4.6リリース Linux Kernel 2.4.7リリース Linux標準化仕様「LSB 1.0」がリリースされた KDEブラウザ『Konqueror』がActiveXに対応した
	WindowsとLinuxを簡単に切り替えて使うことができるように設定されたノートパソコンが日立から発売された ポーランド(株)は、Linuxネイティブのビジュアル開発環境であるKylinx日本語版のベータテスターを公募開始した IBMが全ストレージ製品をLinux対応としたXFree86の最新バージョン4.0.3が登場した シャープはZaurus欧米版にLinuxを搭載した	2001年8月	Linux Kernel 2.4.8リリース Linux Kernel 2.4.9リリース DellがLinux搭載クライアントを販売中止した『KDE 2.2』がリリースされた
		2001年9月	Linux Kernel 2.4.10リリース SunはStarOfficeの最新バージョンのデモを行った Microsoft Officeに対抗するオフィススイートであり互換ソフトである
		2001年10月	Linux Kernel 2.4.11リリース Linux Kernel 2.4.12リリース Linux Kernel 2.4.13リリース Mozilla 1.0の開発計画が明らかにされた PlayStation 2用Linuxキットが発売された Red Hat Linux 7.2が正式リリースされた Windows用ソフトが使えるLinux、『リンドウズ』が発表された
2001年4月	Linux Kernel 2.4.4リリース 日立製メインフレームでLinux環境を構築できるツールが発売された GNOME 1.4がリリースされた デフォルトのブラウザがMozillaのLinuxが登場した 『Turbolinux Server 6.5』が発売された ポーランド社はKylinx日本語版を2001年5月18日出荷開始すると発表した 日立製作所がCOBOL85のLinux版を2001年6月に発売すると発表した 『Samba 2.2』がリリースされた。今回のアップデートで、SambaサーバをWindows2000/NTマシンに認識させる「認証ソース」としても利用できるようになった	2001年11月	Linux Kernel 2.4.14リリース Linux Kernel 2.4.15リリース Linux Kernel 2.4.16リリース 『Netscape 6.2』Linux版が発表された 米Borlandが『Kylinx 2』を発表した Apache 2.0がリリースされた Mozilla 0.9.6がリリースされた
		2001年12月	Linux Kernel 2.4.17リリース MicrosoftがLinuxを商標侵害で提訴した
2001年5月	Linux Kernel 2.4.5リリース Linuxの次世代ファイル・システム候補「XFS」が正式に公開された HDEはLinuxサーバ管理ツール「HDE Linux Controller 2.4 Standard Edition」を発表した 『Mozilla 0.9』が公開された Linuxインターフェイス開発のイーゼルが事業閉鎖した Red Hat LinuxはNetscapeブラウザの採用を打ち切った 『Wnn』が6年ぶりにバージョンアップし、『Wnn7』になった	2002年1月	Linux Kernel 2.5.2リリース この版でUSB2.0が正式サポートされた Linux Kernel 2.5.3リリース Samba 2.2.2日本語版リリース1.0がリリースされた プレイステーション2用Linuxの正式版がリリースされた
		2002年2月	Linux Kernel 2.5.4リリース Linux Kernel 2.5.5リリース Linuxの「標準仕様」であるLSB1.1がリリースされた Opera日本語正式版がリリースされた サンが『StarOffice』を有料化した
2001年6月	シャープはLinux版ザウルスのプロトタイプを公開した XFree86 4.1.0がリリースされた (株)ホライズン・デジタル・エンタープライズはLinuxサーバ管理ソフト『HDE Linux Controller 2.4 Professional Edition』を発表した オープンソースコンパイラ「GCC」のバージョン3がリリースされた	2002年3月	Linux Kernel 2.5.6リリース Linux Kernel 2.5.7リリース PHPのセキュリティホールが発見された リンドウズがマイクロソフトに勝訴した 東芝がLinuxとWindowsを搭載するノートPCを発売した ミラクル・リナックスとセゾン情報システムズが提携し「HULFT for Linux」を発表した
		2002年4月	Linux Kernel 2.5.12リリース 『Apache』バージョン2.0の公式版がリリースされた

変します。

OSの利用者から見た場合、`ps -al`とコマンドを入力して表示される各プロセスは、そのままそのマシン上で動いている「プログラム」です。しかし、OSの内部から見た場合、これらはカーネルのタスクなの

です。

つまり、カーネルとはOSそのものであり、Linuxの存在そのものでもあるプログラムなのです。

カーネルの仕組み

カーネルは起動されると、必要な初期化処理を実行

〔リスト2〕

/boot下のファイル群

```
[root@linux /]# cd /boot
[root@linux /boot]# ls -l
total 2883
lrwxrwxrwx 1 root root          19 Mar 26 2000 System.map -> System.map-2.2.16-3
-rw-r--r-- 1 root root    214147 Aug 18 2000 System.map-2.2.16-3
-rw-r--r-- 1 root root      512 Mar 26 2000 boot.0300
-rw-r--r-- 1 root root     4636 Mar 10 2000 boot.b
-rw-r--r-- 1 root root      612 Mar 10 2000 chain.b
-rw-r--r-- 1 root root   332048 Mar 26 2000 initrd
-rw-r--r-- 1 root root      237 Mar 13 01:05 kernel.h
drwxr-xr-x 2 root root     12288 Mar 26 2000 lost+found/
-rw-r--r-- 1 root root     13824 Mar 26 2000 map
lrwxrwxrwx 1 root root      20 Mar 26 2000 module-info -> module-inf o-2.2.16-3
-rw-r--r-- 1 root root    11773 Aug 18 2000 module-info-2.2.16-3
-rw-r--r-- 1 root root      644 Mar 10 2000 os2.d.b
-rwxr-xr-x 1 root root   1682855 Aug 18 2000 vmlinux-2.2.16-3*
lrwxrwxrwx 1 root root      16 Mar 26 2000 vmlinux -> vmlinux-2.2.16-3
-rw-r--r-- 1 root root    654436 Aug 18 2000 vmlinux-2.2.16-3
[root@linux /boot]#
```

してからイベント待ちになります。そこでは、大きく分けて以下のような事象が起きるのを待ちます。

- プロセス間通信(シグナル)
- システムコール
- 割り込み

まず、プロセス間通信(シグナル)ですが、カーネルから見たシグナルはプロセスの状態を変化させるために使用されます。それはまたプロセスから見た場合、ソフトウェア割り込みの一つでもあります。

システムコールとは、ユーザーがカーネルの機能を実行させるための手続きのことで、主記憶に常駐している「タスク構造体」の中身を読み出す。簡単にいえば、画面に文字を表示したり、データファイルに書き込みを行うようなときに使用します。C言語のライブラリ関数などが実際にシステムコールを発行するので、一般のシステム開発ではブラックボックスですが、よりカーネル寄り、システム寄りのソフトウェアを開発する際には理解していなければならない機能です。

割り込みとは、ここではハードウェア割り込みのことをいいます。簡単にいえば、LANからパケットを受け取ったとき、キーボードからキーを打鍵したとき、タイマ割り込みなどで発生する割り込みのことです。それぞれの事象が起きたときに、どのタスクを起動するかを記述してあるのが割り込みハンドラです。

このようにカーネルはLinuxの基幹であり、これが安定していないと大きな問題となります。また、カーネルはLinuxの基本機能そのものなので、カーネルを変更することによって新しい機能を追加することができます。

表2に、最新版であるカーネル2.4で追加された機能を示します。

カーネル構築の手順と意味

さて、新しく追加された機能がどうしても必要だ、または試験的に使ってみたいという場合、カーネルを入れ替える作業をしなければなりません。いわゆるバージョンアップです。

本章では、カーネル2.4にバージョンアップする際の手順を示します。ただし、業務で使用している場合や現在の環境を破壊したくないという場合には、慎重に検討し、自己の責任で行ってください。

カーネル構築の手順

ここで説明することは、カーネルを2.4にバージョンアップする方法です。OSの本体そのものを入れ替える作業を行うわけなので、くれぐれも慎重に行ってください。

まず、準備を始めます。

● フロッピーブートの準備

最初にLinuxシステムをインストールしたとき、フロッピーにブートディスクを作成していたのかどうかを、思い出してみてください。もし作成していない場合には、次の方法で作成してください。

- 1) `uname -r`とコマンド入力する
- 2) `2.2.xx-xx`と表示される
- 3) `mkbootdisk --device /dev/fd0 2.2.xx-xx`とコマンド入力
- 4) フロッピーディスクを入れてenterキーを押下する

これまでの作業は、図1のようになるはずですが(環境によって`uname -r`で返される数値は違う)。

● バックアップを取る

必要なファイルは、sambaなどを使用して他のマシンに退避させておきます。

● ソースファイルの入手

〔表2〕Linux カーネル2.4の新機能

<p>カーネルのコア部分の変更</p>	<p>プロセスとスレッドの管理の充実 同時に実行可能なプロセス数の上限が引き上げられた 2.2では上限が設定されていたが、2.4ではリソースが確保できる限り、動的に上限値が変更される</p> <p>マルチプロセッサ対応の安定化 マルチプロセッサ環境において別々のCPUで実行されているプロセスを管理する機能が向上した</p> <p>仮想ファイルシステム機能の変更 ext2ファイルシステムで2Gバイト超のデータファイルを扱えるようになった</p> <p>ファイルキャッシュ機能の向上 ファイルキャッシュの方式が変更され、読み出しと書き込みを同一のキャッシュで処理できるようになった</p> <p>システムリソース管理の向上 最大メモリは64Gバイト、IDEコントローラを10個、ネットワークカードは16枚まで使用できる</p>
<p>ハードウェアのサポートに関して</p>	<p>PCカード ノートブックなどでネットワークカード、無線LAN、IDE、SCSIなどのデバイスを利用できる</p> <p>USB カーネル2.3でキーボードやマウスはサポートされたが、USB HUBやオーディオデバイス、USB経由のハードディスク・CD-RW・MOなども含めてサポートされるデバイスが大幅に増加した MP3プレーヤなどにも対応している</p> <p>IEEE1394 試験的なサポートとしてFireWire、iLinkのインターフェースが組み込まれている</p> <p>ISAのプラグ&プレイ ISAカードのプラグ&プレイにカーネルが対応した</p> <p>UDMA100 UDMA100をサポートした</p>
<p>ファイルシステムに関して</p>	<p>UDF UDFは、DVDを対象に、異なるOS間でのデータ交換を可能にすることと、光ストレージ各種で共通にデータを扱えるように規格化された 対象メディアは、DVD-ROM、DVD-Video、DVD-Audio、DVD-R、DVD-RAMやDVD-RWなどのDVDファミリである</p> <p>JFFFS 組み込みシステム用のジャーナルファイルシステム「JFFFS」 フラッシュメモリが扱える 参考 (http://www.developer.axis.com/software/jffs/)</p> <p>RAMFS いわゆる「RAMDISK」が進化したものである。サイズを可変にできる</p> <p>ネットワークファイルシステム サポートしているネットワークファイルシステムがNFS V2から3にバージョンが上がった</p>
<p>新たに導入された新機能</p>	<p>NLS Native Language Supportで日本語サポートが追加された これによりFATファイルシステム上の日本語ファイルの扱いが楽になった</p> <p>DEVFS これによって煩雑な手続きが必要だったデバイスドライバの管理がカプセル化された</p> <p>DRM 複数のプロセスからのグラフィックデバイスへのアクセスを制御し3Dレンダリングを高速処理できるようになった</p> <p>LVM 複数のディスクデバイス上のパーティションを論理ボリュームとして扱うことが可能になった</p> <p>kHTTPd カーネルに統合されたWebサーバ機能である。単純なHTMLなら、かなりの高速化が図れる ただし、CGIを扱うことができない</p>

筆者の場合は、ringサーバからソースファイルを取
得しました。アドレスは、
<http://www.ring.gr.jp/archives/>
[linux/kernel.org/kernel/v2.4/](http://www.ring.gr.jp/archives/linux/kernel.org/kernel/1linux/kernel.org/kernel/v2.4/)
です。FTPなら、
<ftp://ftp.ring.gr.jp/archives/linux/>
[kernel.org/kernel/1kernel.org/kernel/v2.4/](ftp://ftp.ring.gr.jp/archives/kernel.org/kernel/1kernel.org/kernel/v2.4/)
にいき、バージョン2.4のディレクトリを開きます。

〔図1〕 ブートディスクの作成手順

```

kterm <4>
[root@linux /root]# uname -r
2.2.16-3
[root@linux /root]# mkbootdisk --device /dev/fd0 2.2.16-3
Insert a disk in /dev/fd0. Any information on the disk will be lost.
Press <Enter> to continue or ^C to abort:
[root@linux /root]# █

```

ringサーバは、いろいろな場所にミラーがあるので、使用しているプロバイダにミラーがあれば、そこに接続すると速くなります。

●その他のソフトウェアのアップグレード

最低限アップグレードしておかなければならないソフトウェアとバージョンを表3に示します。

なお、カーネルソースに付属しているアップグレードに関するドキュメント「Changes」の概略をAppendix (p.13 ~ p.15)に示します。ここに、アップグレードされたツールの場所などが記述されています。RPMパッケージが使用できる環境であるなら、ディストリビュータのFTPサイトに最新のツールが置いてあるはずなので、確認してみてください。

筆者の環境であるTurbolinux6.0 WorkStationのためのRPMパッケージは提供されていましたが、確実に成功させるにはSRPMをコンパイルしたほうが良いと思います。

そして、ソフトウェアのアップデート作業に入ります。蛇足ですが、以降の処理は必ずroot権限で行ってください。

まずは、インストール済みのバージョンを確認します(リスト3)。

調査した結果を表4に示します。

筆者の環境ではSRPMパッケージを利用できるので、それをダウンロードして更新しました。

なお、GCCはegcs-1.1.2を使用しますが、なかなか見つからないと思います。ダウンロードするなら、リングサーバの、

```
ftp://ftp.ring.gr.jp/pub/lang/egcs/
releases/egcs-1.1.2/
```

にあります。また、RedHat系ならばredhatのサイトにSRPMがあるので、それを使用します。

RPMのほうが楽にインストールできるかもしれま

〔表3〕 アップグレードするソフトウェアとバージョン

ツール名	必要バージョン	バージョン確認のためのコマンド
Gnu C	2.91.66	# gcc --version
Gnu make	3.77	# make --version
binutils	2.9.1.0.25	# ld -v
util-linux	2.10o	# fdformat --version
modutils	2.4.2	# insmod -V
e2fsprogs	1.19	# tune2fs
reiserfsprogs	3.x.0b	# reiserfsck 2>&1 grep reiserfsprogs
pcmcia-cs	3.1.21	# cardmgr -V
PPP	2.4.0	# pppd --version
isdn4k-utils	3.1pre1	# isdnctrl 2>&1 grep version

〔表4〕 バージョンを調べた結果

ツール名	現状のバージョン	コメント
Gnu C	2.95.2	バージョンを変更する
Gnu make	3.79	そのまま
binutils	2.9.5	そのまま
util-linux	util-linux-2.10f	バージョンを上げる
modutils	2.1.121	バージョンを上げる
e2fsprogs	1.17	バージョンを上げる
reiserfsprogs	----	この環境では不要
pcmcia-cs	3.1.19	この環境では不要
PPP	----	この環境では不要
isdn4k-utils	----	この環境では不要

せんが、あるパッケージを更新したら、それに依存するパッケージをすべて更新しなくてはならなくなるのでSRPMのほうをお勧めします。

さて、入手したSRPMを/usr/src/turbo/SRPMsにコピーします(リスト4)。ただし、このディレクトリ名は環境によって変わります。

そして、usr/src/turbo/RPMS/に移動し、RPMコマンドでインストールします(リスト5)。

以上の操作で、ソフトウェアのアップデートは無事に終了するはずで。

次に、インストールするPCのハードウェア環境を再度確認してください。必要な情報は以下のものです。

- 1) CPUの種類
- 2) マザーボードのチップセット
- 3) ビデオカードのチップの型番
- 4) ネットワークカードのチップ
- 5) SCSIカードのチップ
- 6) 他の拡張インターフェースの情報

●カーネルの入れ替え作業

いよいよインストール作業に入ります。まずは安全

【リスト3】

インストール済みのバージョンを確認

```
[root@linux linux]# gcc --version
2.95.2
[root@linux linux]# make --version
GNU Make version 3.79, by Richard Stallman and Roland McGrath.
Built for i386-pc-linux-gnu
Copyright (C) 1988, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99
Free Software Foundation, Inc.
This is free software; see the source for copying conditions.
There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A
PARTICULAR PURPOSE.

Report bugs to <bug-make@gnu.org>.

[root@linux linux]# ld -v
GNU ld version 2.9.5 (with BFD 2.9.5.0.34)
[root@linux linux]# fdformat --version
fdformat from util-linux-2.10f
[root@linux linux]# /sbin/insmod -V
insmod version 2.1.121
Usage:
insmod [-fkmopsvVxX] [-o name] module [[sym=value]...]

module           Filename of a loadable kernel module (*.o)
-f, --force      Force loading under wrong kernel version
-k, --autoclean  Make module autoclean-able
-m              Generate load map (so crashes can be traced)
-o NAME
  --name=NAME    Set internal module name to NAME
-p, --poll      Poll mode; check if the module matches the kernel
-s, --syslog    Report errors via syslog
-v, --verbose   Verbose output
-V, --version   Show version
-x             Do not export externs
-X            Do export externs (default)
[root@linux linux]# /sbin/tune2fs
tune2fs 1.17, 26-Oct-1999 for EXT2 FS 0.5b, 95/08/09
Usage: /sbin/tune2fs [-c max-mounts-count] [-e errors-behavior]
[-g group]
      [-i interval[d|m|w]] [-l] [-s] [-m reserved-blocks-percent]
      [-r reserved-blocks-count] [-u user] [-C mount-count]
      [-L volume-label] [-M last-mounted-dir] [-U UUID]
      [-O [^]feature[...]] device
[root@linux linux]# reiserfsck 2>&1|grep reiserfsprogs
[root@linux linux]# /sbin/cardmgr -V
cardmgr version 3.1.19
[root@linux linux]# /sbin/pppd --version
bash: /sbin/pppd: No such file or directory
[root@linux linux]# isdnctrl 2>&1|grep version
[root@linux linux]#
```

【リスト4】

SRPMSを/usr/src/turbo/SRPMSにコピー

```
# rpm --rebuild binutils-2.9.5.0.34-1.src.rpm
# rpm --rebuild e2fsprogs-1.20.WIP.0103-5.src.rpm
# rpm --rebuild egcs-1.1.2-25.src.rpm
# rpm --rebuild make-3.79.1-3.src.rpm
# rpm --rebuild modutils-2.3.24-2.src.rpm
# rpm --rebuild urpm --rebuild til-linux-2.10r-4.src.rpm
```

【リスト5】

RPMコマンドでインストール

```
# rpm -ivh binutils-2.9.5.0.34-1.rpm
# rpm -ivh e2fsprogs-1.20.WIP.0103-5.rpm
# rpm -ivh egcs-1.1.2-25.rpm
# rpm -ivh make-3.79.1-3.rpm
# rpm -ivh modutils-2.3.24-2.rpm
# rpm -ivh util-linux-2.10r-4.rpm
```

のために lilo.conf の書き換えを行います。

まず初めにブートフロッピーで起動するのかどうか、確認しておきましょう。

LILLOの設定ファイルは、前述のように/etc/lilo.confに指定されていますが、これをリスト6のように修正します。

修正が終わったら lilo を実行してください。ただし、環境によって vmlinuz の後の名称、HD の名称、パーティション名が異なるので注意してください。

この作業によって、もしカーネルの構築に失敗した場合には、ブートフロッピーで起動したり、起動時に LILO: と表示されたら linux2 と入力することで元のカーネルで起動させることができます。しかし、深刻な問題を抱えてしまったような場合には、この限りではありません。

次に、ダウンロードしたカーネル linux-2.4.2.tar.gz (現在の最新版) を /usr/src の下に解凍します(リスト7)。

いよいよ、カーネルコンフィグレーションを行いま

【リスト6】

/etc/lilo.confの修正

```

boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
lba32
timeout=50
default=linux
#元のカーネル
image=/boot/vmlinuz2.2.16
    #ラベルをlinux2に変更する
    label=linux2
    root=/dev/hda6
    initrd=/boot/initrd
    read-only

#新しいカーネル
image=/boot/vmlinuz
    label=linux
    root=/dev/hda6
    initrd=/boot/initrd
    read-only

```

【リスト7】

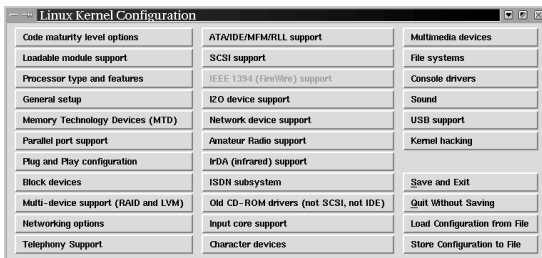
カーネルのアーカイブファイルを解凍

```

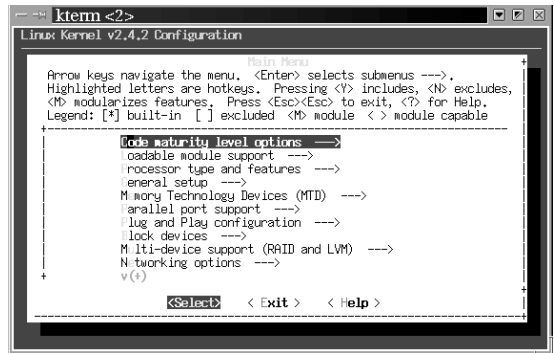
#cd /usr/src/
#rm linux
#tar xvfz linux-2.4.2.tar.gz
#mv linux linux-2.4.2
#ln -s linux-2.4.2 linux

```

【図2】 xconfig



【図3】 menuconfig



す。

```

#cd linux
#make xconfig

```

ただし、X Window環境でない場合には、make menuconfigと入力します。詳細は後述します。

次にコンパイルを行います。その前に、もしオーバロックをしている場合には、カーネルをmakeしているとエラーで止まることがあります。実際、筆者はファイルシステムが壊れて再インストールする羽目になりました。コンパイルは、次のようにコマンドを入力します。

```

#make dep
#make clean
#make bzImage
#make install

```

実行マシンの速度によっては、ここまで数時間かかることもあります。コンパイルが終わったら、モジュールのコンパイルとインストールです。

```

#make modules
#make modules_install

```

以上が問題なく終了すれば、カーネル2.4を使用することができます。リポートすることで起動するはず です。

カーネルコンフィグレーションの詳細

前述したように、X Window環境ならばxconfig(図2)を、そうでなければmenuconfig(図3)を使用することができます。

そこで、ここでの基本的な操作を説明します。必要な機能をカーネルに含めるなら'y'を、含めないなら'n'を、そしてモジュールとして必要なときに主記憶にロードするなら'm'を選択または入力します。

カーネル2.4の設定項目は非常に多岐に渡るため、すべてのオプションを詳細に説明していたら一冊の書籍になってしまいます。それぞれの設定オプションについては、重要な箇所や、デフォルト値では問題がある部分についてのみ、表5に示します。

なお、モジュール化した場合、インストールするにはinsmodコマンド、削除するにはdelmodコマンド

〔表5〕カーネルの設定オプション(重要な箇所、デフォルト値では問題がある部分のみ)

設定項目	説明
Code maturity level options	開発中の設定オプションを表示するかどうかの設定。標準では無効となっている。 一般ユーザーには関係ないので、この機能は無効にする
Loadable module support	モジュール機能を使うか否かを設定する。通常はこのままデフォルト値でOK
Processor type and features	CPUの種類や搭載数、機能などを設定する
Processor family	現在使用しているCPUの種類を選択する ここでCPUの設定を間違えると動作しなくなるので注意が必要 ちなみにCoppermineコアのCeleronはPentium3で設定する
General setup	カーネルの基本動作の設定
PCMCIA/CardBus Support	ここで設定すればpcmcia-csモジュールは不要になるが、好みによって設定する
Memory Technology Devices(MTD)	主記憶以外でメモリデバイスを使用する設定 たとえば、組み込みシステムでフラッシュメモリにファイルシステムを構築することができる 一般ユーザーは' n 'でOK
Parallel port support	パラレルポートを使用する場合、ここで設定する
Plug and Play configuration	プラグ&プレイを使用するかどうかを設定する なお、ISAも対応できるようになった
Block devices	フロッピーディスクドライブ、パラレルポートに接続するIDE機器などの設定を行う 通常のIDE機器、SCSI機器は別の項目で設定する 自分の環境に必要なものを設定する
Multi-device support(RAID and LVM)	ここでは、カーネルによるソフトウェアRAID、複数のパーティションやデバイスを一つにまとめて管理するLVM(Logical Volume Manager)の機能を使うかどうかを設定するデフォルトではどちらも無効
Networking options	カーネルがサポートするネットワークプロトコルやその機能を設定する。デフォルトでは、標準的なTCP/IPを中心としたネットワーク設定になっている DECNET, Appletalkも使える ここで、新機能のnetfilterが設定できる。パケットフィルタリングやNATの設定ができる
Telephony Support	特殊なインターネット電話のためのもの 一般的には設定不要
ATA/IDE/MFM/RLL support	ATA/IDE機器を設定する ここでIDEをモジュールにしてしまうとIDEディスクから起動できなくなるので注意が必要
IDE, ATA AND ATAPI BLOCK DEVICES	標準では、IDE接続のハードディスクとATAPI接続のCD-ROMドライブをサポートするようになっている その他の特殊なものを使用したい場合には、ここで設定する
IEEE support	最初の項目である開発中のオプションの使用を有効にすると設定できる
SCSI support	SCSI接続機器の設定を行う PCMCIAのSCSIアダプタが設定できるようになった
I2O device support	Intelligent I/Oの設定を行う まだ標準化の最中の規格なのでなじみがないかもしれない
Network device support	ネットワークデバイスの設定をする、無線LANに対応できるようになった
Amateur Radio support	アマチュア無線パケット通信の設定
IrDA(infrared)support	赤外線通信の設定 有効またはモジュールとして組み込んだ場合には、使用している赤外線モジュールのチップによってデバイスドライバを設定する必要がある
ISDN subsystem	不要
Old CD-ROM drivers(not SCSI, not IDE)	SCSI, ATAPI以外のCD-ROMを使うことができる 一般的には設定不要
Input core support	入力デバイスを設定する 具体的な設定はCharacter devices, USB supportの項目で設定する

〔表5〕カーネルの設定オプション(重要な箇所, デフォルト値では問題がある部分のみ)(つづき)

Character devices	シリアルポート, マウス, ジョイスティックの設定をする
Multimedia devices	Video For Linuxの設定ができる Bt848のチップを搭載するビデオキャプチャカードであれば動作させることが可能
File systems	新機能としてはDVDで使用されるUDFファイルシステムが設定できる. Microsoft JolietやFATを設定すると便利 AMIGAのFFS, APPLEのHFSも使用できる 開発中のオプションの使用が有効なら「DevFS」の設定が可能
Console drivers	コンソールのビデオモード, フレームバッファの設定. 標準設定ではVGAテキストのコンソールが有効になっているのでこのままでOK
Sound	サウンドカードの設定をする. 一覧には, サウンドカードやサウンドチップの名前が表示される. ALSA, OSSともに使用できる
USB support	USBデバイスの細かい設定ができる. 設定してもストレージに関してはうまくいかないという話をよく聞く USBマウス, キーボード, USB HUBなどは問題ない オーディオデバイスはRoland UA-30を接続してみたが動作しなかった ここではUSBコントローラ(UHCI/OHCI)の選択が重要
Kernel hacking	特定のキーを押した際に, カーネルの動作情報が書き出され動作状況を把握することができるようになる設定 通常は設定不要

を使います.

なお, 再度確認しますが, ハードウェア環境を把握していないと正当な設定ができないので, わからない場合やあいまいな場合には2.4にアップグレードするのはやめたほうが無難です.

さて, みなさんのマシンでカーネル2.4は稼動した

でしょうか? 自分の環境に関して正確な情報が得られれば, 必ずうまくいくと思います.

今後, カーネルがバージョンアップされるたびに機能は増えていきます. 新機能の恩恵を得るためにも, 一度はカーネルの入れ替えに挑戦してみてください.

Appendix 1

ドキュメント「Changes」の概要

岸 哲夫

このドキュメントは, 2.4のカーネルを実行するために最低限必要となるソフトウェアの, 最少限のリストです. 2.2.xカーネルからアップグレードする場合には, 補足情報として2.2.xカーネルに含まれていたChangesを調べてください.

基本的に, このドキュメントでは, すでに問題なく2.2.xカーネルが動作していると仮定しています.

アップグレードに必要な最低条件

アップグレードするためには, いくつかのソフトウェア(表3参照)のバージョンアップがなされなければ

なりません.

また, すでに問題なく2.2.xカーネルが動作していることを再確認してください. なお, すべてのツールが必要になるとは限りません. たとえば, PCMCIA(PCカード)がないのであれば, pcmcia-csは必要ありません.

カーネルのコンパイル

●GCC

必要になるgccのバージョンは, CPUに依存します. 次に示すことは, x86 CPUユーザー向けの内容です.

他のCPUのユーザーは、別の場所から gccバージョンに関する必要条件を入手してください。

カーネルをコンパイルするために推奨されるコンパイラは、egcs 1.1.2(gcc 2.91.66)です。もし、安定な動作を必要とするのであれば、これを使用すべきでしょう。

もし、不安定でも良いというのであれば、代わりに gcc 2.95.2を使用することもできます。なお、上記以外のバージョンでは、Linuxカーネルのコンパイルテストが行われていません。

また、gcc 2.7.2.3ではカーネルをコンパイルできないことに注意してください。さらに、コンパイラの最適化にも注意を払ってください。オプションを-O2より大きくしてコンパイルすることは賢明ではありません。

● Make

カーネルを構築するためには、Gnu make 3.77以降が必要になります。

● Binutils

IA-32対応のLinuxは、16ビットのブートコードをアセンブルするために使用するツールがas86からgasに変わりました。

この変更により、binutilsの最近のリリースが必要になります。

できれば、最新の2.9.5、または2.10にアップグレードしてください。2.8や2.8.xxのような古いリリース、そしてFSF2.9.1の使用は避けるべきです。バージョンによっては、2.9.1.0.x(x >= 22)以降のリリースならば、カーネルを適切にコンパイルすることは可能ですが、2.9.5または2.10にアップグレードするほうが適切です。

システムユーティリティ

●仕様の変更点

DevFSをカーネル内に採用しました。カーネルのインストール後に、usr/src/linux/Documentation/filesystems/devfs/*を参照してください。論理ボリュームマネージャ(Logical Volume Manager: LVM)はカーネルの中にあります。これを使用する場合にはLVM toolsetをインストールする必要があります。

● Util-linux

util-linuxの新バージョンは、より大容量ディスクのfdiskをサポートします。また、サポートされるパーティションの種類が追加されました。そして、fdformatが可能で

● Ksymoops

カーネルからのreportを解析するためには、ksymoopsのバージョン2.4が必要になるでしょう。詳細に関しては、/usr/src/linux/REPORTING-BUGSを参照してください。

● Modutils

2.3.xで頻りに発見されたさまざまなバグを直し、かつUSBモジュールの自動ロードを可能にするために、最新のmodutilsにアップグレードしてください。

● Mkinitrd

/lib/modulesのファイルツリーの配置の変更によってmkinitrdもアップグレードする必要があります。

● E2fsprogs

e2fsprogsの最新のバージョンでは、fsckとdebugfsのいくつかのバグが修正されています。アップグレードを推奨します。

● Reiserfsprogs

reiserfsprogsパッケージがreiserfs-3.6.x(Linuxカーネル2.4.x)で使用できます。パッケージには、mkreiserfs、resize_reiserfs、debugreiserfsおよびreiserfsckなど、このバージョンで動作するものが含まれています。これらのユーティリティは、i386およびAlphaで動作します。

● Pcmcia-cs

PCMCIA(PCカード)サポートは、部分的にカーネル内に実装しました。最新のpcmcia-csに必ずアップグレードしてください。

● intel IA32 マイクロコード

IA32マイクロコードを更新できるようにするためのドライバが追加されました。

● Powertweak

v0.1.17またはそれ以前のものならば、バージョンv0.99.0またはそれ以降にアップグレードするべきです。古いバージョンでは、共用メモリで問題を引き起こします。

ネットワーキング

●一般的な変更点

IP firewallingおよびNATコードが再び差し替えられました。新しいnetfilterソフトウェア(ipfwadmとipchainsとの互換性をもつモジュールを含む)は、現在は別に配布されています。ネットワークのconfigurationを行うのであれば、ip-route2のネットワークツールを使用することを考えるべきです。

● PPP

PPPドライバは、マルチリンクをサポートし、かつ種々のメディア層で動作することを可能にするために再構成されました。PPPを使用するなら、少なくとも2.4.0b1以降にpppdをアップグレードしてください。

ファイルの入手先

●コンパイラ

▶ egcs 1.1.2 (gcc 2.91.66)

<ftp://sourceware.cygnum.com/pub/gcc/releases/egcs-1.1.2/egcs-1.1.2.tar.bz2>

▶ gcc 2.95.2

<ftp://ftp.gnu.org/pub/gnu/gcc/gcc-2.95.2.tar.gz>

● Gnu Make

▶ Make 3.77

<ftp://ftp.gnu.org/gnu/make/make-3.77.tar.gz>

● Binutils

▶ 2.9.1 series

<ftp://ftp.valinux.com/pub/support/hjl/binutils/2.9.1/binutils-2.9.1.0.25.tar.gz>

▶ 2.9.5 and 2.10 series

<ftp://ftp.valinux.com/pub/support/hjl/binutils/>

● System utilities

▶ Util-linux

<ftp://ftp.kernel.org/pub/linux/utils/util-linux/util-linux-2.10o.tar.gz>

▶ Ksymoops

<ftp://ftp.kernel.org/pub/linux/utils/kernel/ksymoops/v2.4/>

▶ Modutils

<ftp://ftp.kernel.org/pub/linux/utils/kernel/modutils/v2.4/>

▶ Mkinitrd

<ftp://rawhide.redhat.com/pub/rawhide/SRPMS/SRPMS/mkinitrd-2.9-1.src.rpm>

▶ E2fsprogs

<ftp://download.sourceforge.net/pub/sourceforge/e2fsprogs/e2fsprogs-1.19.tar.gz>
<ftp://download.sourceforge.net/pub/>

<sourceforge/e2fsprogs/e2fsprogs-1.19.src.rpm>

▶ Reiserfsprogs

ftp://ftp.namesys.com/pub/reiserfsprogs/reiserfs_utils-3.6.25-fsck-3.x.0b.tar.gz

▶ LVM toolset

<http://linux.msede.com/lvm/>

▶ Pcmcia-cs

<ftp://pcmcia-cs.sourceforge.net/pub/pcmcia-cs/pcmcia-cs-3.1.21.tar.gz>

▶ Jade

<ftp://ftp.jclark.com/pub/jade/jade-1.2.1.tar.gz>

▶ DocBook Stylesheets

<http://nwalsh.com/docbook/dsssl/>

▶ Intel P6 microcode

<http://www.urbanmyth.org/microcode/>

▶ Powertweak

<http://powertweak.sourceforge.net/>

● Network

▶ PPP

<ftp://linuxcare.com.au/pub/ppp/ppp-2.4.0.tar.gz>

▶ Isdn4k-utils

<ftp://ftp.isdn4linux.de/pub/isdn4linux/utils/isdn4k-utils.v3.1pre1.tar.gz>

▶ Netfilter

<http://netfilter.filewatcher.org/iptables-1.2.tar.bz2>
<http://netfilter.samba.org/iptables-1.2.tar.bz2>
<http://netfilter.kernelnotes.org/iptables-1.2.tar.bz2>

▶ Ip-route2

<ftp://ftp.inr.ac.ru/ip-routing/iproute2-2.2.4-now-ss991023.tar.gz>