redhat®

Linux インターネットサーバー

で今すぐできる

今やメジャーな第三のOS「Linux」

注目を集めるLinux

このところLinuxとい無償のUNIXクローンOSが話題となっている。先日、インテル、ネットスケープという有名企業がLinux製品を扱うベンチャー企業「Red Hatソフトウェア」に出資を決めたことから、にわかに一般誌（紙）でも取り上げられることとなった。

Linuxは1991年にフィンランドのヘルシンキ大学に在籍していたLinus B. Torvaldsという青年によって開発された。彼はOSの開発を専門としていたわけではなかったが、1人で何もない状態からLinuxを作り上げた。「素人が作った」と専門家から批判されることもあったが、彼はインターネットで結ばれた世界中の仲間と共同開発するというスタイルを選び、それが今日Linuxの成功の一因となっている。

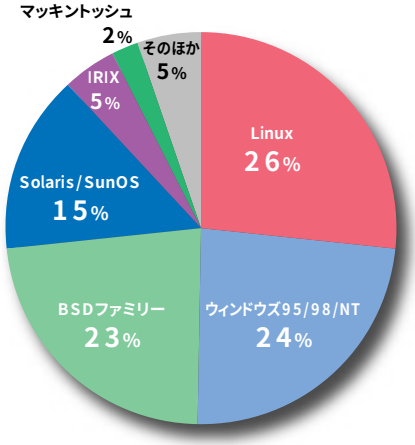
こうした共同開発ができるのは、Linuxのソースコードがすべて公開されているからだ。ネットスケープによるコミュニケーターの「オープンソース戦略」は、Linuxの成功に強く影響されたものだ。

ビジネス分野でも注目される

Linus氏がLinuxとして配布しているのは、「カーネル」と呼ばれるOSの基本部分だけだ。このため、ユーザーが実際に使うためには、コマンドやウィンドウシステム、各種設定ファイルなどを組み合わせる必要がある。この組み合わせを汎用的にまとめたものが「ディストリビューションパッケージ」と呼ばれるものだ。これが配布されることによって少しずつ「市場」が形成されて、Linuxは徐々にビジネスの対象として注目されていくことになった。

このような背景があったため、Linuxは初期のころから、有名な商用アプリケーションが移植されたり、ハードウェアメーカーによってLinux対応のデバイスドライバーが提供されたりしてきた。今年になってインフォミクスが同社製データベースのLinux版の提供をアナウンスすると、オラクルやサイベース、コンピュータアソシエイツなど、ライバルメーカーたちもそれに続いた。このようにLinuxはデータベースのプラットフォームとしても注目を集めている。

こんなに使われているLinux



上の図はインターネット上で稼動するWWWサーバーがどのOSをプラットフォームとしているかのグラフだ。このデータは「The Internet Operating System Counter」のデータを基にしている。これを見ればLinuxが全体の4分の1以上を占めていて、いかにインターネットで使われているかが一目瞭然だ。

●The Internet Operating System Counter
URL http://www.hzo.cubenet.de/ioscount/

完全マスター

今回紹介するLinuxディストリビューションパッケージ「Red Hat Linux5.2」はCD-ROM [B]に収録されている。ぜひインストールしてみてほしい（ソースコードの入手方法については402ページを参照）。

これから専用線を引き、インターネットサーバーを自前で持とうとすると、どうしてもネットワークに関する知識が必要になる。専門用語の壁で止まることなく、最小限の完璧なサーバーをLinuxで構築するためのマニュアルをここに用意した。この記事と本誌の付録CD-ROMがあれば、数時間でインターネットサーバーができあがる。とにかく完成させてから、じっくりネットワークの勉強を始めよう。

井上尚司
＋
風穴江
＋
すずきひろのぶ
＋
編集部

インターネットサーバー用のOSを選ぶとき、商用UNIX、Free BSD、ウィンドウズNT、マッキントッシュなどの候補が挙げられるが、中でも注目を集めているのが今回取り上げるLinuxだ。ではなぜLinuxなのだろうか。ここでは今やメジャーとなったLinuxについて迫ってみよう。

ほとんどのプラットフォームに対応

当初はインテルx86プラットフォームで開発されたLinuxだが、現在では、Alpha、SPARC、PowerPC（マッキントッシュ）、モトローラ68k（マッキントッシュ、Amiga）、MIPS Rシリーズ、StrongARMなど、さまざまなプラットフォームに移植されている。特にAlpha版のLinuxは数値計算サーバーとして大学や研究機関などで多く採用されている。また、SPARC版とPowerPCマッキントッシュ版の移植、開発作業にはサン・マイクロシステムズやアップルがそれぞれ協力しているなど、インテルx86以外のプラットフォームへの対応も活発に行われている。

先日発表された小型インターネットサーバー「コバルトキューブ」はMIPS R5000系の組み込みRISCプロセッサーをベースにしたオリジナルハードウェアだが、OSとして同社が独自に移植したLinuxが採用されている。ソースコードが公開されているため、こうした「組み込みOS」としての利用も広がっている。

パッケージの選択はさまざま

Linuxの特徴であるディストリビューションパッケージにはRed Hat Linuxを始めさまざまなものが存在する。商用パッケージとして販売されているものやフリーソフトウェアもある（右表）。また日本語環境をサポートしたものとしては、「Turbo Linux」（パシフィック・ハイテック㈱）、Red Hat LinuxにPJEと呼ばれる日本語拡張キットを組み合わせた「Red Hat＋PJE」（㈱五橋研究所レーザーS出版局）などがある。

これらはOSカーネルは同じなので、Linuxとして提供されている機能にそれほど違いはない。それぞれの特徴と自分の好みを照らし合わせてディストリビューションパッケージを選べばいい。ただし、一部のコマンドや各種設定ファイルなど、システム管理に影響する大きな違いもあるので、ディストリビューションが混在する環境の場合は注意が必要だ。

本記事では、商用ベースではもっとも普及しており将来性にも期待が持てるRed Hat Linuxを使って説明する。

さまざまなLinuxのパッケージ

Red Hat Linux：フリー版、商用版
RPMというパッケージ管理システムを備える。現時点ではもっともポピュラーな製品。
URL http://www.redhat.com/

Turbo Linux：フリー版、商用版
Red Hatをベースに独自に日本語化。98年末には早くもVer.3がリリースされる予定。
URL http://www.pht.co.jp/

Slackware：フリー版
比較的古くから存在しているため、稼働ベースではもっとも多いと思われるパッケージ。
URL http://www.cdrom.com/

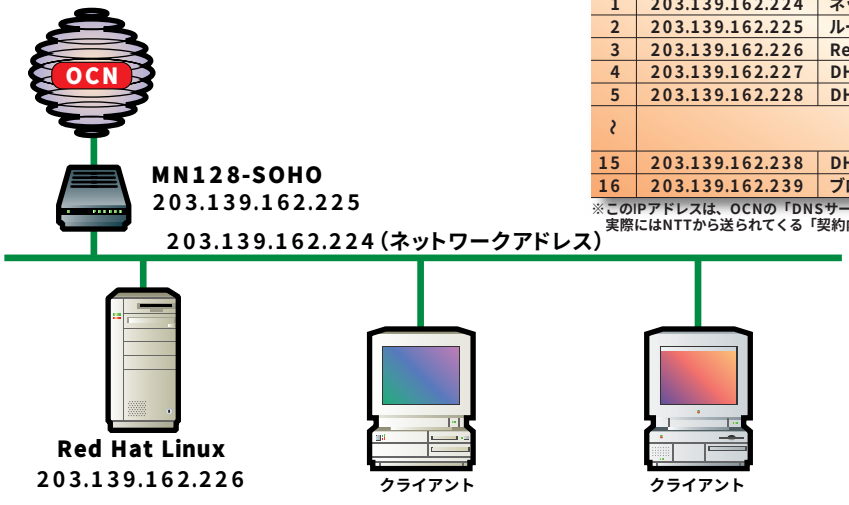
Debian/GNU Linux：フリー版
ボランティアベースで開発されている。日本語化も独自に行なわれるなど開発は活発。
URL http://www.debian.org/

S.u.S.E.：商用版
ドイツ製。Red Hatベースだが独自の工夫も見られる。日本語化の実績がほとんどない。
URL http://www.suse.com/

Caldera Network Desktop：商用版
元ノベル会長のレイ・ノーダ氏の会社が開発している。商用パッケージとしては草分け。
URL http://www.caldera.com/

ネットワーク構成を考えよう

インターネットを導入してサーバーを設置するためには、まずネットワークの構成を考えておく必要がある。ここでは専用線接続として、OCNエコノミーとMN128-SOHOの利用を前提に、IPアドレスや各種サーバーについて検討する。



	IPアドレス	対　象	役　割
1	203.139.162.224	ネットワークアドレス	
2	203.139.162.225	ルーター（MN128-SOHO）	ゲートウェイ
3	203.139.162.226	Red Hat Linuxサーバー	DNSサーバー兼メールサーバー
4	203.139.162.227	DHCPで割り振られる	クライアント
5	203.139.162.228	DHCPで割り振られる	クライアント
～	～		
15	203.139.162.238	DHCPで割り振られる	クライアント
16	203.139.162.239	ブロードキャストアドレス	

※このIPアドレスは、OCNの「DNSサーバー設定方法」で使用されているものを流用した。実際にはNTTから送られてくる「契約内容のご案内」で指定されているアドレスになる。

OCNのネットワークはこうなる

メールサーバーとDNSサーバーを自分で用意するタイプのOCNエコノミーでは、申込みの際にドメイン名やDNSサーバーのホスト名を指定しておく必要がある。そしてNTT側の処理が終わると「契約内容のご案内」が送られてくる。

この案内には、割り当てIPアドレスとして「ネットワークアドレス」と「ネットマスク」の指定と、DNSサーバーのIPアドレスとなる「プライマリDNS IPアドレス」が示されている。

指定されるネットマスクは28ビットマスクの「255.255.255.240」なので、最後の4ビットの部分の組み合わせで全部で16個のIPアドレスを生み出すことができる。

指定されたネットワークアドレスが「203.139.162.224」だとすると、これ以降の16個、すなわち「203.139.162.239」までが利用できるIPアドレスということになる。

16個のうち、最初と最後のアドレスはそれぞれ「ネットワークアドレス」と「ブロードキャストアドレス」で、通常のホスト（コンピュータかネットワーク機器）には割り当てられない。

また、OCNとのゲートウェイとなるルーターの「MN128-SOHO」にネットワークアドレスのすぐ次のIPアドレスを割り当てる。

今回構築するLinuxのインターネットサーバーには、NTTの「ご案内」の中に「プライマリDNS IPアドレス」で指定されているIPアドレスを付ける。このホストが、DNSサーバー兼電子メールサーバーになるわけだ。

おすすめのネットワーク構成はこれだ

NTTから割り当てられるIPアドレスは全部で16個。そのうちホストに割り振れるものが14個で、すでにルーターに1つと、DNSサーバー兼電子メールサーバーに1つ割り当てているので、残りは12個となる。

NATなどのアドレス変換を利用しない場合は、IPアドレス1つにホスト1台が対応するので、ネットワークに全部で12のホストをさらに接続できるわけだ。

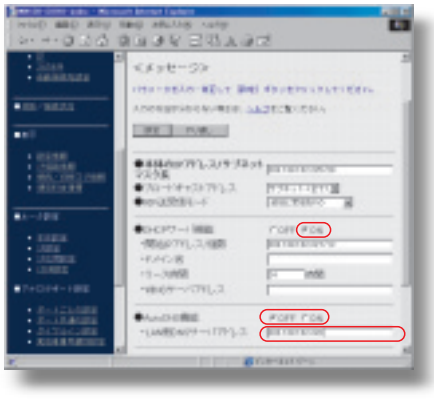
12のホストに明示的にIPアドレスを指定してもいいのだが、IPアドレスを固定しておく特別な理由がないのなら、DHCPを利用するのが便利だ。

今回利用するMN128-SOHOもそうだが、最近のルーターはたいていDHCPサーバー機能を持っているので、これを利用するのが簡単だ。

MN128-SOHOでは、下の画面のように、「DHCPサーバ機能」をONにし、「AutoDNS機能」はOFFにする。さらに「LAN側DNSサーバアドレス」でLinuxサーバーのIPアドレスを指定する。また、その他のサーバー機能はすべてOFFにしておく。

接続するクライアント側では、IPアドレス、DNSサーバーアドレス、デフォルトゲートウェイアドレスをDHCPで取得するように設定すればいい。

ルーター設定→IP設定



Linuxのインストールにチャレンジ

ネットワークの構成が決まったら早速Linuxのインストールにチャレンジだ。インストールは手順を踏めば決して難しくはない。しかし、ある程度OSの知識が要求されるのでよく読んで試してほしい。ここで説明する手順は、まっさらなハードディスクにインストールするものなので、必要なファイルがあれば移動しておこう。

Linuxのためのハードウェア選び

LinuxはLinus氏を中心にボランティアベースで開発されているため、ほとんどのハードウェアメーカーが自らデバイスドライバーを用意するウィンドウズのようなプラットフォームに比べると、特に最新ハードウェアへの対応が遅くなる傾向がある。しかし、こうした「ハンデ」はあるものの、世界中の優秀なプログラマーが多数参加していることもあって、特にユーザーが多いデバイスの場合は、数週間程度の遅れでベータ版のドライバーが作られることも珍しくない。

また早くからビジネスの対象とされてきたLinuxでは、特に需要の多いディスプレイドライバーやサウンドドライバーについては、最新版への対応をうたった市販製品もある。中でも有名なのはディスプレイドライバー（Xウィンドウシステム）を提供するXiGraphics社の「Accelarated-X」で、最新のグラフィックスアクセラレーターチップのほとんどをカバーしている。こうした選択肢もあると考えれば、事実上ほとんどの一般的なPCハードウェアでLinuxを利用できるといっていいだろう。

今回のようなネットワークサーバーを構築する場合に気になるのはネットワークカードへの対応だろう。10BASE-Tに限れば、信頼できるメーカーのものならほとんどどれでも利用できる。100BASE-TX対応カードの場合も、3comなどの有名メーカーのものならほぼ問題ない。

Linuxのためのハードウェアを選ぶ場合は「Linuxは世界中で使われているのだから、みんなが使いたいと思うようなデバイスは、たいていサポートされている（あるいはサポートしようという動きがある）」という想像力を働かせることだ。もちろん、ハードウェア動作確認リストのような情報もたくさん公開されているので、そうしたリソースも積極的に活用しよう。

参考として日本におけるLinuxの第一人者であるはねひでや氏のハードウェア動作確認リストを紹介しておく。
このページで紹介されているハードウェア実績リストは動作を保証するものではないので、使用上の注意をよく読んで利用しよう。
URL http://www.flatout.org/~wing/Linux/

実際にインストールしてみよう！

ここで本誌の付録CD-ROM Bに収録したRed Hat Linux 5.2のインストール方法を解説しよう。上記のようにほとんどのハードウェアで問題なくインストールできるはずだ。

今回はサーバー用の環境を構築することが目的なので、ウィンドウズマシンとは別にLinux専用のマシンを用意し、ハードディスクの中はすべて消去して領域を削除しておくことにする。また、フォーマットしたフロッピーディスクを2枚用意しておこう。

今回のインストールでは、ハードディスクに必要な容量は約250Mバイトだった。実際の運用においては、最低でも500Mバイトは必要だ。

①起動ディスクの作成

まず最初にLinuxをインストールするための起動ディスクを作成する。ここではウィンドウズ（95、98、NT 4.0）を使った起動ディスクの作り方を説明しよう。

①最初に本誌の付録CD-ROM Bとフォーマットしたフロッピーディスクをウィンドウズマシンに挿入する。
②OSプロンプトを立ち上げてCD-ROMドライブに移動する（Dドライブとする）。
③CDコマンドでディレクトリー「images」に移動する。

```
D:¥>CD images
```

④次のコマンドを実行する。

```
D:¥IMAGES>¥dosutils¥rawrite.exe
```

「image source file」に「boot.img」を、「diskette drive」に「A」を入力して「Enter」キーを押すと、フロッピーディスクにLinuxの起動イメージがインストールされる。

②起動ディスクでコンピュータを起動

次に①で作成した起動ディスクを、Linuxをインストールするマシンに挿入し再起動する。

「boot:」というメッセージが出るので「Enter」キーを押してLinuxのインストーラーを起動させる。「Welcome to Red Hat Linux!」というメッセージが出たら「Ok」を選ぶ。

③言語、キーボードの選択

言語の選択では、「English」を選び、次に使っているコンピュータのキーボードを選ぶ。101または104英語キーボードなら「us」を、106または109日本語キーボードなら「jp106」を選ぶ。

パッケージを購入すれば日本語も使える

本誌の付録CD-ROMに収録したRed Hat Linux 5.2には、日本語環境を構築するためのソフトウェアがそろっていない。日本語が使いたいなら、Red Hatのバージョンは4.2だが日本語環境が付属している「Red Hat+PJE」のパッケージを購入するとよいだろう。このパッケージは、202ページの読者プレゼントにもなっている。

問い合わせ先
㈱五橋研究所　レーザー5出版局
URL http://www.cdrom.co.jp/

③

④インストール元の選択

今回は本誌付録のCD-ROMを使ってインストールするので「Local CDROM」を選ぶ。

⑤インストールかアップグレードかの選択

今回は新規インストールなので「Install」を選ぶ。

⑥マシンの用途の設定

Red Hat Linux 5.2では、マシンの用途として「Workstation」か「Server」のどちらかを選べば、最適な環境を自動的に作成してくれるようになった。「Custum」を選べばハードディスクのパーティションやインストールするプログラムなどを細かく指定できる。今回の記事では「Server」を選ぶことにする。

⑥

⑦SCSIの設定

LinuxをインストールするコンピュータにSCSI機器（ディスクやCD-ROMなど）があるなら、ここで「Yes」を選んでSCSIの設定をしておこう。なければ「No」を選ぶ。

⑧警告

「ハードディスクにあるLinuxの領域はすべて削除される」という警告が出る。今回は、まっさらなハードディスクにインストールすることが前提なので「Ok」を選ぶ。もう一度警告が出るのでまた「Ok」を選ぶ。

⑨ファイルのコピー

自動的にハードディスクに領域が作成され、Linuxのファイルがコピーされる。

⑩マウスの選択

コピーが終了したら、マウスの設定になる。一般にはPS/2マウスが検出されるはずだ。Logitechのマウスを使っている場合は、マウスの種類を選ぶ画面で選択するとよい。また、2ボタンマウスの場合、「Emulate 3 Buttons?」をチェックすると3ボタンマウスを擬似的に使えるようになる。

⑪ビデオカードの設定

次にビデオカードが自動的に検出される。最新のビデオカードでなければ、ほとんどのものがサポートされているはずだ。「Ok」を選ぶと自動的にXサーバーがインストールされる。

⑫モニターの設定

使っているモニターを選ぶ。一覧になければ、「Custom」を選んで自分のモニターに合った設定にする。「Cunstom」を選ぶと、水平の走査周波数と垂直の走査周波数を選択する画面が出るので、そこで最適なものを選ぶ。選択が終わると、自動的にXウィンドウシステムの設定が行われる。

⑩

⑭

⑬ネットワークカードの設定

次にネットワークの設定を行う。最初に自動的にLANカードが検出される。

機種によっては検出されない場合もある。そのカードがサポートされているとわかっているなら手を尽くせば使えるようになるが、それが面倒であれば必ず使えると保証されているものを購入したほうがいいだろう。

⑫

⑮

⑭IPアドレスの割り振り方

次にIPアドレスの割り振り方を設定する。サーバーなので「Static IP address」にしておこう。

⑮IPアドレスの設定

サーバーのIPアドレス、ネットマスク、デフォルトゲートウェイのIPアドレスなどを入力する。232ページを参考にしてほしい。

⑯ドメイン名の設定

ドメイン名とホスト名を入力し、「Secondary nameserver」にプロバイダー（OCNなど）から指定されたセカンダリーネームサーバーのIPアドレスを入力する。

⑰タイムゾーンの設定

「Japan」を選んでおこう。「Hardware clock set to GMT」はチェックしない。

⑱ルートのパスワード設定

ルートのパスワードを設定する。忘れずに必ず設定しよう。入力しても何も表示されないようになっているが、2つの入力欄の上でパスワードをキーボードでタイプしてから「Ok」を選べばよい。

⑲ブートディスクの作成

緊急用のブートディスクを作成する。あらかじめフォーマットしたフロッピーディスクを挿入し、「Yes」を選ぶ。

⑳インストールの完了

これでインストールは終了だ。フロッピーディスクを抜いて再起動すれば、Linuxが立ち上がるはずだ。

⑯

⑱

⑳

インストール後にCD-ROMを使うには

Linuxのインストール後にCD-ROMを読めるようにするには、以下のコマンドでマウントする。

```
# mount /mnt/cdrom
```

ディレクトリー/mnt/cdromの下がCD-ROMの内容になる。また、CD-ROMを取り出す前には、以下のコマンドを必ず実行する。

```
# umount /mnt/cdrom
```

Red Hat Linuxだったらアップデートも簡単！

セットアップが完了してサーバーが稼動し始めたら、次に必要なのはソフトウェアのアップデートだ。Linuxのソフトウェアは日々新しいバージョンが出ているし、サーバーソフトはセキュリティーの問題から常に新しいバージョンを使ったほうが好ましい。そこで、ここではソフトウェアのアップデート方法を解説しよう。

Red Hat Linuxではソフトウェアはすべて RPM形式のファイルで配布されている。このファイルに対してコマンドを実行すれば、自動的にインストール作業までしてくれる。

RPM化されたソフトウェア最新版はWWWから入手できる。つまり一度CD-ROMからインストールしたら、あとはWWWサイトから必要なRPMパッケージだけをダウンロードして、アップデートすればいいわけだ。

すでにインストールされているソフトウェアをアップデートする場合は、WWWサイトからアップデートするソフトウェアのRPMパッケージをダウンロードして、ファイルを置いたディレクトリーで次のコマンドを実行する。

```
$ rpm -Uvh ファイル名.rpm
```

この「ファイル名.rpm」はRPMパッケージのファイル名をさしている。

アップデート以外にも新規にソフトウェアをインストールする場合には、RPMパッケージをダウンロードしてアップデートと同様に次のコマンドを実行すればいい。

```
$ rpm -ivh ファイル名.rpm
```

RPMパッケージは次のサイトからダウンロードできる。カテゴリーや作成日、名前などでRPMを検索できる。

URL **http://rufus.w3.org/linux/RPM/**

DNSサーバーを設定しよう

専用線を引いてネットワークを構築するのに必要となるのがDNSサーバーだ。DNSサーバーはさまざまなサーバーを立てていく際に、最も重要になるものだ。ここではサンプルファイルを見ながらDNSサーバーの設定を解説しよう。

DNSサーバーの役割

DNSとは「Domain Name System」の略でインターネットで使われるドメイン名とIPアドレスとを対応付けるための仕組みだ。

インターネットではすべてのコンピュータはIPアドレスを使って通信しているが、ユーザーは一般に「www.impress.co.jp」といったドメイン名を使って通信している。そこで、ホスト名とIPアドレスを検索してくれるコンピュータが必要になる。このような作業を提供するのがDNSサーバーだ。

たとえば「www.impress.co.jp」というコンピュータと通信する場合、「impress.co.jp」というドメインにあるDNSサーバーに問い合わせれば、「www」というホストのIPアドレスを教えてくれるようになっている。

DNSサーバーは自分が管理すべき範囲（これをゾーンと呼ぶ）のすべてのホストの情報を持ち、クライアントからの要求に回答する。また、DNSサーバーはドメイン名からIPアドレスを検索する（正引き）だけでなく、IPアドレスからドメイン名を検索する（逆引き）作業も行ってくれる（これらをマッピングと呼ぶ）。ほかにも、DNSサーバーは電子メールの配送先の指定にも用いられる。

LinuxにはBINDというUNIXでよく使われるDNSサーバーがインストールされている。そこで、BINDを使ったDNSサーバーの設定方法について解説しよう。

Red Hat5.2では、以前と比べてDNSの設定ファイルの名前や場所に大きな変更があるので注意が必要だ。

DNSの設定ファイルを知ろう

DNSの設定で使われるファイルは右図に示したディレクトリー構造になる。ここで、緑色で示したファイルは自分で作成または編集が必要になるファイルだ。また白色で示したファイルはそのままで使えるファイルだ。

●クライアント用設定ファイル

DNSサーバーが動くホストは当然ながらDNSクライアントとしての設定も必要だ。そこで、必要となるのが「/etc/host.conf」と「/etc/resolv.conf」の2つだ。

「/etc/host.conf」はドメイン名とIPアドレスのマッピング方法と、その適応順序を記述するファイルで、もう1つの「/etc/resolv.conf」は問い合わせるDNSサーバーの指定を行うファイルだ。

Linuxをインストールしたマシン上でDNSサーバーを起動させない場合は、この2つだけを設定しておけばよい。

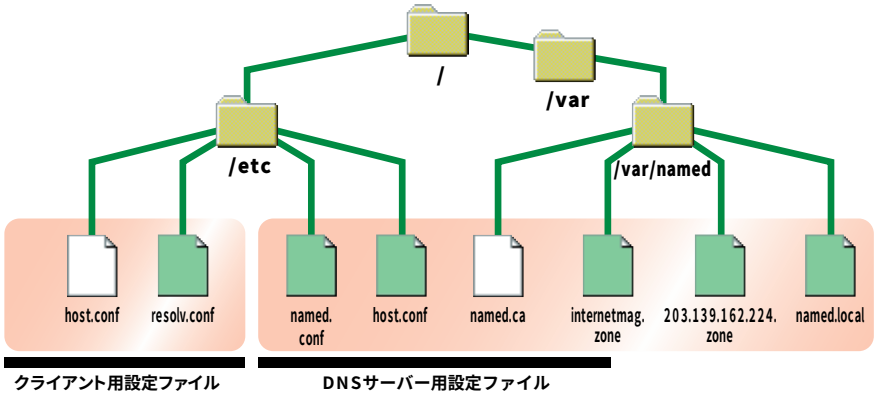
●DNSサーバー設定用ファイル

「/etc/named.conf」はDNSサーバーが起動するときに参照するファイルで、「/etc/named.boot」から自動生成できる。管理者は「/etc named.boot」をまず用意し、その中の記述でDNSサーバーの動作を指示することになる。

Red Hat Linuxでは、この2つの設定ファイル以外は「/var/named」というディレクトリーに置くようになっている。

「named.ca」はルートネームサーバーと呼ばれる世界中のおおもととなるDNSサーバーを示したファイルだ。

そのほかのファイルは管理するゾーンの情報を記述したファイルで、設定条件によって変わってくる。今回のように比較的小規模なネットワーク構成の場合は、正引きマッピングのためのファイル、逆引きマッピングのためのファイル、ループバックアドレスの逆引きのためのファイルの3つをそれぞれ用意することになる。この3つのファイルの記述方法は、238ページで説明する。



named.bootファイルの用意

DNSサーバーの設定として最初にnamed.bootファイルを用意する。これはDNSサーバーが起動時に読み込む初期設定ファイルnamed.confを作成するためのものだ。Linuxをインストールすると「/etc」ディレクトリーにnamed.bootのテンプレートファイルが用意されているので、これを使って編集を進めよう。

リスト1のnamed.bootを見てほしい。このファイルはこのように5行で記述できる。

1行目はこの行以降で指定するファイルがどのディレクトリーにあるかを示している。そこで、この行は初期設定のまま「/var/named」を指定する。

2行目はキャッシュの指定で、これも初期設定のままでいいだろう。2項目の「.」は必要なので消さないように気をつけよう。3項目の「named.ca」はファイル名で、このファイルはLinuxのインストール時にすでに用意されている。このファイルにはルートネームサーバーというDNSサーバーのおおもととなるサーバーが記述されている。内容が更新されることがあるので、常に新しいものを用意しておこう。最新のファイルは以下のURLから入手してほしい。

URL ftp://ftp.rs.internic.net/domain/named.root

3、4、5行目はこのDNSサーバーが管理するゾーンの指定とそのゾーンの情報を記述した設定ファイルの指定だ。1項目の「primary」はこのDNSサーバーが各ゾーンのプライマリーサーバーであることを意味している。2項目は管理するゾーンのドメイン名の記述だ。3項目は各ゾーンについての設定が書かれたファイルを指定している。ではそれぞれの行を詳しく見ていこう。

3行目ではゾーンとして「internetmag.co.jp」というドメインを指定している。これはドメイン名からIPアドレスへのマッピング、すなわち正引き用ゾーンの指定となる。

4行目はゾーンとしてネットワークアドレス（232ページの表参照）の前後を逆にして、その後ろに「in-addr.arpa」を付けて指定している。これは逆引き用ゾーン指定の方法だ。

5行目も4行目と同様に逆引き用ゾーン指定だが、4行目と違うのは、これはループバックという1台のマシン内で使われる閉じたネットワークのためのものだ。ループバックネットワークのIPアドレスは127.0.0で、ローカルホスト（localhost）という自分自身を指すホスト（IPアドレスは127.0.0.1）がある。

リスト1 ●named.boot

	1項	2項	3項
1行	directory		/var/named
2行	cache	.	named.ca
3行	primary	internetmag.co.jp	internetmag.zone
4行	primary	224.162.139.203.in-addr.arpa	203.139.162.224.zone
5行	primary	0.0.127.in-addr.arpa	named.local

named.confファイルの作成

named.bootファイルが用意できたら、このファイルからnamed.confファイルを作成する。変換用にperlプログラムが用意されているので、これを用いる。リスト2のようにコマンドを実行する。安全のために「/tmp」ディレクトリーで作業しよう。問題なくnamed. confファイルが作成されたら、それを/etcディレクトリーにコピーする。

リスト2

```
# cd /etc
# cp named.boot /tmp
# cd /usr/doc/bind-8.1.2
# cp named-bootconf.pl /tmp
# cd /tmp
# perl named-bootconf.pl < named.boot > named.conf
# cp named.conf /etc
```

ファイルを編集するには？

Linux上でファイルを編集する方法について悩んでいる人もいるだろう。そこでここではテキストファイルの編集方法を解説しよう。

FTPで編集するファイルをウィンドウズやマッキントッシュといったコンピュータでダウンロードして、日ごろ使い慣れたエディターで編集する方法もあるが、ネットワークの設定後でなければ使えず、また改行コードに気を配らなければならない。

Linuxのエディターを使う方法もある。UNIXの標準にviやemacsというエディターがある。どんなUNIXマシンでも使えるので使い方を覚えておくと非常に便利だ。使い方は多くのUNIX関連の本に解説してあるのでそちらを参考にしてほしい。

おすすめのエディターとして初心者ならばjoeはどうだろうか。起動して「Ctrlキー＋Kキー」→「Hキー」でヘルプが表示されるので、それを見ながら操作できる。joeがもしインストールされていなければ、CD-ROM内の「RedHat/RPMS」というディレクトリーにRPMパッケージがあるので、そこに移って次のコマンドでインストールしよう。

```
$ rpm -ivh joe-2.8-14.i386.rpm
```

正引きゾーンファイルを編集しよう

編集する前に頭に入れてほしいのは「.（ピリオド）」の扱いだ。これを忘れてしまうとDNSサーバーはまったく違った動作をしてしまうので、気をつけてほしい。

リスト3「internetmag.zone」が正引き用のゾーンファイルだ。このファイルは新規に作成しなければならない。

各項目について見てみよう。

①ホスト名。最後に「.」を付ける。

②管理者のメールアドレスの「@」を「.」に変えたもの。最後に「.」を付ける。①と②は同じ行であることに注意。

③シリアル番号。ゾーンファイルに変更を加えたら必ずこの値を増加させる。

④NSレコードと呼ばれ、DNSサーバーの指定を意味する。最初の行は当然ながら今設定中のホスト名となる。この下の行はプロバイダーが指定してきたセカンダリーDNSサーバーを指定する（例はOCNエコノミーの場合）。

⑤Aレコードと呼ばれ、IPアドレスを指定する。ここでは、DNSサーバーの稼動しているこのホストのIPアドレスを指定している。これは、ホスト名を付けずにドメイン名だけを検索したときに返すIPアドレスとなる。

⑥MXレコードと呼ばれ、電子メールの配送先を指定する。ドメイン名にinternetmag.co.jpが指定された電子メールは、この指定に従って、ns.internetmag.co.jpに送られる。

⑦ローカルホストのアドレス指定。ローカルホストの正引きはゾーンとして独立させずにinternetmag.co.jpゾーンの中で行う。

⑧CNAMEレコードといい、ホストの別名を指定する。ここではloghostという別名を用意し、それが実際にはlocalhostだということを指定している。

これ以降の行はほぼパターンが決まっていて、Aレコードでホスト名に対してIPアドレスを指定し、それに続いてMXレコード、HINFOレコード、CNAMEレコードが続く。

HINFOレコードはそのホストやハードウェアの情報を記述するものだ。ここではこの情報はコメントとして扱うように、行の先頭に「;」を入れている。最後に⑨のようにDHCPで割り当てられるクライアントすべてに対してもAレコードでIPアドレスを指定しておく。ここではclient1〜12というホスト名を付けている。

リスト3 ●internetmag.zone

```
@       IN      SOA     ns.internetmag.co.jp.①
root.ns.internetmag.co.jp.②(
                                        ③1998110101 ; Serial
                                          28800     ; Refresh
                                          14400     ; Retry
                                          3600000   ; Expire
                                          86400 )   ; Minimum
                IN      NS      ns.internetmag.co.jp.④
                IN      NS      ns-tk011.ocn.ad.jp.

                IN      A       203.139.162.226⑤
                IN      MX      10  ns.internetmag.co.jp.⑥

localhost       IN      A       127.0.0.1⑦
loghost         IN      CNAME   localhost⑧

router          IN      A       203.139.162.225
;               IN      HINFO   ROUTER MN128-SOHO

ns              IN      A       203.139.162.226
                IN      MX      10  ns.internetmag.co.jp.
;               IN      HINFO   INTEL Linux BOX
server          IN      CNAME   ns
www             IN      CNAME   ns
ftp             IN      CNAME   ns
mail            IN      CNAME   ns
mailhost        IN      CNAME   ns

client1         IN      A       203.139.162.227⑨
                                  ～
client12        IN      A       203.139.162.238
```

逆引き用ゾーンファイルを編集しよう

リスト4が逆引きゾーンファイルだ。これも新規に作成する。ファイルの最初の部分は正引き用とほぼ同様だ。

⑩PTRレコードと呼ばれ、ゾーン名からドメイン名の対応（ポインター）を指定している。この例では、ゾーン名である224.162.139.203.in-addr.arpaに対して、internetmag.co.jpへの対応付けを指定している。

⑪Aレコードだが、ここではネットマスクを入れている。この行はなくてもいい。

あとは各ホストのIPアドレスの最下位部とホスト名を対応付けるPTRレコードを記述するだけだ。⑫はこのネットワーク全体を表すドメイン名で、通常は「xxx-net.xxx.co.jp」の形式で記述する。

最後に正引き用ゾーンファイルと同じようにDHCPで割り当てられるクライアント用のPTRレコードも入れておこう（⑬）。

リスト4 ●203.139.162.224.zone

```
@       IN      SOA     ns.internetmag.co.jp.
root.ns.internetmag.co.jp.  (
                                        1998110101 ; Serial
                                        28800      ; Refresh
                                        14400      ; Retry
                                        3600000    ; Expire
                                        86400 )    ; Minimum
                IN      NS      ns.internetmag.co.jp.
                IN      NS      ns-tk011.ocn.ad.jp.

                IN      PTR     internetmag.co.jp.⑩
                IN      A       255.255.255.240⑪

224             IN      PTR     internetmag-net.internetmag.co.jp.⑫
225             IN      PTR     router.internetmag.co.jp.
226             IN      PTR     ns.internetmag.co.jp.
227             IN      PTR     client1.internetmag.co.jp.⑬
                                  ～
238             IN      PTR     client12.internetmag.co.jp.
```

ループバックの逆引き用ゾーンファイルの編集

最後はループバックの逆引き用ゾーンファイルを編集する。このファイルはテンプレートがあるので、そのまま使ってもいいだろう。リスト5がnamed.localの例だ。テンプレートファイルと違うのは、1行目をほかのゾーンファイルと同じ表記にしているところだ。

リスト5 ●named.local

```
@       IN      SOA     ns.internetmag.co.jp.
root.ns.internetmag.co.jp.  (
                                        1998110101 ; Serial
                                        28800      ; Refresh
                                        14400      ; Retry
                                        3600000    ; Expire
                                        86400 )    ; Minimum
                IN      NS      localhost.

1               IN      PTR     localhost.
```

DNSサーバーを再起動しよう

ここまでファイルの編集が終わったら、あとはDNSサーバーを再起動すればいい。再起動させることによって、編集したファイルの設定が有効になるからだ。

方法は次のコマンドを実行する。

```
# /etc/rc.d/init.d/named restart
```

今まで編集したファイルのいずれかに変更を加えた場合、必ずこのコマンドを実行しよう。

DNSクライアントとしての設定

DNSクライアントの設定には2つのファイルを使う。

「/etc/host.conf」は最初から用意されているものをそのまま使えばいい。

もう1つのファイル「/etc/resolv.conf」では、クライアントが問い合わせるDNSサーバーを記述する。

リスト6を見てほしい。最初の「domain」行はローカルドメインを定義している。ピリオドの入っていないホスト名が使われた場合は、このドメイン内のホスト名と解釈される。

「nameserver」行は問い合わせるDNSサーバーの指定だ。1番目はプライマリーDNSサーバーすなわち自分自身のIPアドレスを記述している。2番目はセカンダリーDNSサーバーのIPアドレスになる。クライアントはこの順番に書かれたとおりにDNSサーバーの問い合わせをしていく。

リスト6 ●resolv.conf

```
domain internetmag.co.jp
nameserver 203.139.162.226
nameserver 203.139.160.73
```

DHCPを利用しないホストを登録しよう

これまで説明してきたDNSの設定を行えば、OCNエコノミーに接続するときの最低必要条件は満たせる。

しかし、WWWサーバーを新たなマシンを使って設置するような場合は、IPアドレスを固定的に割り当て、さらにDNSサーバーにその旨を設定する必要がある。これによって、WWWサーバーの利用者はDNSを通じてIPアドレスを得られるわけだ。

これを行うには、DNSサーバーの正引きゾーンファイルと逆引きゾーンファイルにホストの記述を追加するという作業が必要となる。

右図では「203.139.162.227」というIPアドレスを持つサーバーに「www」というホスト名を付ける例を示している。

正引きゾーンファイルへの追加では、IPアドレスを指定するAレコード、電子メールの配送先を指定するMXレコードを記述し、さらに必要に応じて別名を指定するCNAMEレコードやホストマシンの情報を示すHINFOレコードを記述する（❶）。

逆引きゾーンファイルでは、IPアドレスからホスト名への対応をPTRレコードで記述する（❷）。

この2つのファイルとも、変更したときはシリアル番号を最低でも1増加させることを忘れないでほしい（❸、❹）。

変更作業が終わったらDNSサーバーを再起動させる。

正引きゾーンファイル

```
@       IN      SOA     ns.internetmag.co.jp.
root.ns.internetmag.co.jp.  (
                                3 1998110102 ; Serial
                                  28800      ; Refresh
                                  14400      ; Retry
                                  3600000    ; Expire
                                  86400 )    ; Minimum
                IN      NS      ns.internetmag.co.jp.
                IN      NS      ns-tk011.ocn.ad.jp.

www             IN      A       203.139.162.227   ❶
                IN      MX      10  ns.internetmag.co.jp.
;               IN      HINFO   WWW Server Linux BOX
home            IN      CNAME   www
```

逆引きゾーンファイル

```
@       IN      SOA     ns.internetmag.co.jp.
root.ns.internetmag.co.jp.  (
                                4 1998110102 ; Serial
                                  28800      ; Refresh
                                  14400      ; Retry
                                  3600000    ; Expire
                                  86400 )    ; Minimum
                IN      NS      ns.internetmag.co.jp.
                IN      NS      ns-tk011.ocn.ad.jp.

227             IN      PTR     www.internetmag.co.jp.   ❷
```

DNSサーバーの動作を確認しよう！

ここまで設定できたらnslookupコマンドを使ってDNSサーバーが正しく動作しているかを確認しよう。なお、ドメイン名の指定では最後にピリオドを付けると確実だ。

```
% nslookup
```

❶まず、自ドメイン内のホスト名を入れて、設定したIPアドレスが正しく返されるかどうかを確認する。この場合、ホスト名のみの場合とドメイン名まで含めた完全な形の場合の両方を確かめる。

```
> ns
> ns.internetmag.co.jp.
```

❷逆にIPアドレスを入れて、ホスト名が返されることも確認する。

```
> 203.139.162.226
```

❸次に、自ドメイン以外のホスト名を入れてIPアドレスが返されるかどうかを確認する。IPアドレスからホスト名への逆引きも確認すること。

❹ここまで確認できたら次に電子メール配送先の指定を確認する。最初に「set type=mx」と入れて、これ以降はMXレコードの問い合わせであることを指定しておく。

ドメイン名のみを指定して、MXレコードで指定したホストが示されるかどうかを確認する。また、さらにホスト名を付けた形も確認する。

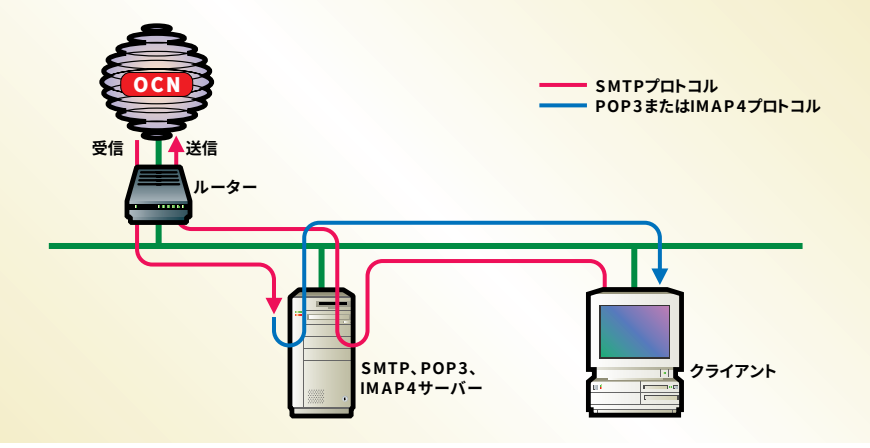
```
> set type=mx
> internetmag.co.jp.
> ns.internetmag.co.jp.
```

❺最後に自ドメイン以外のドメイン名を指定して、正しくMXレコードが示されるかどうかを確認する。

以上の動作が確認できたら、DNSサーバーの動作は正常であるといえる。

電子メールサーバーを設定しよう

自分のドメイン名を持ったなら、次に欲しくなるのが自分のドメイン名の入った電子メールアドレスだ。そこでここでは電子メールサーバーの設定を解説する。電子メールサーバーができれば、いくらでも電子メールアドレスが作れる。



電子メールサーバーの種類

電子メールサーバーは、クライアントマシンなどから差し出された、あるいはほかの電子メールサーバーから転送されたメッセージを受け取る。そして、その宛先を判断して送り先の電子メールサーバーに転送したり、自分宛ての電子メールを蓄積したりする機能を持っている。こうしたやり取りは、電子メール転送プロトコルであるSMTPを使う。

利用者が実際に電子メールを送受信するときは、アウトルックやEUDORA PROなどの電子メールソフトを使う。これらはMUA（Mail User Agent）と呼ばれ、そのやり取りにはPOP3やIMAP4といったプロトコルが使われる。

sendmailを使おう

SMTPサーバーソフトとしてUNIX上で最もポピュラーなのがsendmailだ。Red Hat Linuxでも、このsendmailが標準のSMTPサーバーとなっている。

sendmailはRed Hat Linux5.2のインストール時に自動的にインストールされ、システムが起動したときにsendmailの動作も開始される。

POP3、IMAP4サーバーを使おう

MUAが電子メールサーバーからメッセージを取り込むときに使われるプロトコルで、もっとも広く使われているものがPOP3だ。ほとんどのMUAが電子メールの取り込みにPOP3を使っている。

POP3の場合、メッセージを取り込むときには電子メールサーバーに蓄積されたすべてのメッセージを取り込む。このため、モバイル環境でのアクセスでは、蓄積されたメッセージの量が多いと取り込むまで時間がかかることになる。また、取り込まれたサーバー上のメッセージはMUAで特に設定しない限り消去される。

POP3とは別にIMAP4というプロトコルがある。最近のMUAでは、POP3のほかにこのIMAP4も採用しているものが多くなってきた。アウトルックエクスプレス、ネットスケープメッセンジャー、Beckey!、EUDORA PROといった人気のあるMUAはIMAP4をサポートしている。

IMAP4の場合、電子メールサーバーにアクセスしたときにまず蓄積されたメッセージのヘッダー部分だけを読み込み、メールの本文は読み込まない。利用者は読み込まれたヘッダー部分を見て、本文が今すぐに必要なものとそうではないものに分ける。そして、今すぐに必要なメール本文だけを読み込むわけだ。

さらにIMAP4では、MUAで設定しなくても、メッセージを取り込んだときにサーバー上に蓄積されたものを消去せずに残すことができる。

今回は多くの利用者に対応するため、POP3とIMAP4の2つのサーバーを用意する。

必要な作業とファイル

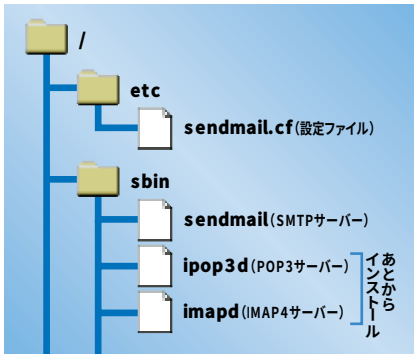
導入するサーバーは、SMTPサーバー、POP3サーバー、IMAP4サーバーの3つだ。

このうち、SMTPサーバーであるsendmailは最初からインストールされていて、初期状態で立ち上がり動作している。

POP3サーバーとIMAP4サーバーは、あらかじめインストールされていないのでインストール作業が必要となる。

サーバーの運用にあたり設定ファイルを用意する必要があるのは、sendmailだけだ。/etcディレクトリーにある「sendmail.cf」というファイルがsendmailの設定ファイルとなる。システムをインストールすると、簡単な設定のsendmail.cfがあらかじめ用意されていて、それを使って動作している。

sendmailの本体は/usr/sbinディレクトリーにインストールされている。POP3サーバーとIMAP4サーバーもこのディレクトリーにインストールすることになる。



sendmailの動作確認

今回のような単純なネットワーク構成の場合、設定ファイルのsendmail.cfファイルに何も手を加えなくても、DNSサーバーの設定（MXレコードの設定など）さえ正確に行っていれば、sendmailは一応動作する。そこで、sendmailの動作を確認してみよう。

電子メールを送信するときは、次のコマンドを利用する。

```
% mail 宛先メールアドレス
```

設定中はスーパーユーザーであるrootで作業しているはずなので、自分自身に電子メールを出すには次のコマンドを実行することになる。

```
# mail root
```

このコマンドを実行すると、「Subject:」というプロンプトが現れるので、適当に題名を入れ、「enter」キーを押す。そのあとに本文を入れ、最後に「CTRL」キー＋「D」キーで入力を終了させる。こうすれば電子メールが送信される。

送った電子メールを読むには、次のコマンドを実行する。

```
# mail
```

引数を付けずにmailコマンドを実行すると、蓄積されたメッセージを読むコマンドとなる。電子メールが正しく届いていれば、その内容が表示される。

また、このような1台のマシンで閉じた電子メールだけでなく、インターネットやイントラネットでの電子メールの転送も可能だ。たとえば、次のようにコマンドを実行する。

```
% mail nisida-r@impress.co.jp
```

POP3サーバーとIMAP4サーバーのインストール

次にPOP3サーバーとIMAP4サーバーをインストールする。

この2つのサーバーは1つのRPMパッケージとしてまとまっているので、これをrpmコマンドを利用してインストールするだけでよい。

付録CD-ROM BからRPMパッケージをインストールするにはCD-ROMをマウントして、リスト1のコマンドを実行する。

リスト1

```
# cd /mnt/cdrom/RedHat/RPMS/
# rpm -ihv imap-4.4-2.i386.rpm
```

これらのコマンドを実行したあと、POP3、IMAP4サーバーが/usr/sbinディレクトリーにインストールされていることを確認する。

```
# cd /usr/sbin
# ls -l imapd ipop3d
```

imapdがIMAP4サーバーでipop3dがPOP3サーバーだ。エラーメッセージが表示されずにこの2つのファイルが表示されたらインストールは成功だ。

この2つのサーバーには準備しなければならない設定ファイルはない。インストールが終わればすぐに利用できる。

ユーザーアカウントの準備

LANにウィンドウズマシンやマッキントッシュマシンを接続し、Red Hat Linuxを電子メールサーバーとして利用するには、現在設定しているLinuxサーバーにログイン用のユーザーアカウントを作成しておく必要がある。

なぜなら、一般にMUAなどから電子メールサーバーにアクセスするときは、ユーザー名とパスワードで本人確認を行うが、Linuxを含むUNIXシステムでは、この電子メールアクセスのための本人確認は、システムにログインするためのユーザーアカウント情報を利用しているからだ。

ユーザーアカウントを作るには、リスト2のコマンドを実行する。

useraddコマンドの「-d」はホームディレクトリーを、「-g」はグループIDを、「-u」はユーザーIDを意味している。最後の引数がユーザー名だ。次にpasswdコマンドを使って、ユーザー「nisida-r」に対して初期パスワードを設定している。

ユーザーに対して電子メールサーバーのアクセスのみを許可し、コンソールから直接ログインしたり、telnetでリモートログインしたりすることを許可しない場合は、リスト3のコマンドを実行する。

「-M」を指定することでホームディレクトリーを作成せず、また「-s」で存在しないファイルを指定することでログインできなくなる。

電子メールサーバーの利用者には、ここで設定したユーザー名とパスワードを知らせればいい。

リスト2

```
# useradd -d /home/nisida-r -g 100 -u 101 nisida-r
# passwd nisida-r
```

リスト3

```
# useradd -M -g 100 -u 101 -s /nofile nisida-r
# passwd nisida-r
```

CFを利用したsendmail.cfの作成

今回の記事で示した方法ではRed Hat Linuxのインストールを行えば、自動的に単純なsendmail.cfファイルが用意される。ただし、このsendmail.cfファイルはあくまでも仮のものなので、本格的な運用をするためにはsendmail.cfをカスタマイズしなければならない。

しかし、sendmail.cfファイルを書くことは、相当な労力が必要となる。複雑な形で書かれているので、中身を理解するだけでも大変だ。

そこで今回はCFというツールを利用する。CFはsendmail.cfを作成するためのツールで、CF用の非常に簡単な設定ファイルを用意するだけで、sendmail.cfファイルを作成できる。しかも、今回のような単純なネットワーク構成の場合はもちろんのこと、複雑なネットワーク構成に対しても十分利用できる汎用的なツールなのだ。

CFを入手する

CFはRed Hat Linuxのパッケージには含まれていないので、FTPなどで入手する必要がある。入手先の1つに右のURLがある。だたし、CFのバージョン番号は今後変更される可能性があるので、ファイルが見つからなければ「CF」ディレクトリー（右のURLからファイル名を除いた部分）の中を確認してほしい。

URL ftp://ftp.kyoto.wide.ad.jp/pub/mail/CF/CF-3.7Wpl2.tar.gz

このファイルを/tmpなどの適当なディレクトリーに置き、次のコマンドを実行する。

```
# gunzip CF-3.7Wpl2.tar.gz
# tar xvf CF-3.7Wpl2.tar
```

この2つのコマンドを実行すると、そのディレクトリーにCF-3.7Wpl2というディレクトリーが作成され、その中にツールが展開される。

展開されたディレクトリーの中には、次に示す、各種のドキュメントが置かれている。

README.jpn
CFのインストールの方法が書いてある。このファイルを参照しながら作業を進めることになる。JISコードによる日本語ファイルとなっているので、ftpなどでウィンドウズマシンなどに移して見るといいだろう。

README
英語のREADMEファイル。

doc/INTRO.jpn
CF用の設定ファイルの簡単な書き方を説明したもの。必ず参照する必要がある。

doc/INTRO
INTRO.jpnファイルの英語版。

doc/MANUAL.jpn
設定ファイルの書き方をすべて解説したもの。今後さらに複雑な設定をするときなどに必要になるものだ。

今回のような単純な構成のネットワークの場合、README.jpnとdoc/INTRO.jpnファイルを参照するだけで十分対処できる。

sendmail.defからsendmail.cfの作成

CFを利用するには、perlコマンドとmakeコマンドが必須となる。今回の記事の方法でLinuxをインストールした場合、これら2つのコマンドはすでにインストールされているので問題はない。

これ以降の作業は、CFのREADME.jpnファイルの指示に従って進める。

まずはMakefile中のPERL変数の確認だが、通常のインストールでは何もする必要はない。

次は作成用ツールを利用している環境のために調整する作業で、CFを展開したディレクトリーで次のコマンドを実行すればよい。

```
# make cleantools
# make tools
```

そして、すでに用意されている標準の設定ファイルのコピーを作るために、次のコマンドを実行する。

```
# cp Standards/sendmail-v7.def
sendmail.def
```

コピーするファイルは利用するsendmailのバージョンごとに異なるが、今回のRed Hat Linuxの場合は、このsendmail-v7.defファイルを利用すればよい。

このコマンドからもわかるとおり、CF用の設定ファイルの拡張子は「.def」である。このファイルからsendmailの設定ファイルである「.cf」ファイルが作成されるわけだ。

コピーしたsendmail.defファイルを利用する環境に合わせて編集する。この編集作業については、次ページを参照してほしい。ここでは作業が終わったものとして、先を続けることにする。

sendmail.defファイルが用意できたら、次のコマンドを実行してsendmail.cfを作る。

```
# make sendmail.cf
```

作成されたsendmail.cfファイルを/etcディレクトリーにインストールする。この場合、installコマンドを使っても、cpコマンドを使ってもよいが、くれぐれもアクセスパーミッションなどの設定を間違えないようにしてほしい。所有者はrootで、パーミッションが「-rw-r--r--」となっていることを次のコマンドを実行して確認してみよう。

```
#ls -l sendmail.cf
```

sendmail.cfが用意できたら、sendmail自体を次のコマンドで再起動する。

```
#/etc/rc.d/init.d/sendmail restart
```

sendmail.defの編集

コピーしたsendmail.defファイルの中にはあらかじめいろいろな情報が入っているので、これを利用しながら編集を行う。

作業は、INTRO.jpnファイルに従って進めることになる。また、内容によってはMANUAL.jpnを参照する必要もある。

❶まずは、生成すべきsendmail.cfファイルの種別を宣言するCF_TYPEである。今回のRed Hat Linuxでインストールされたsendmailに対しては、「R8V7」が指定されていることを確認する。

❷cf作成情報の設定では、とりあえずVERSION_SEPARATORとLOCAL_VERSIONを用意しておく。それぞれ、「'-'」と「`date +%y%m%d%H'」というのが無難な設定だ。

❸利用しているOSの種類を設定するOS_TYPEでは、「linux-redhat」を指定する。

❹FROM_ADDRESSは、電子メール発信時にFromに付けられるアドレスの指定なので、ここでは「'$m'」を指定する。こうすれば、From行はドメイン名となり、ホスト名は記されなくなる。

❺RECIPIENT_GENERICは宛先アドレスがユーザー名のみであったときの補完方法の指定である。ここでは、補完にFROM_ADDRESSが用いるために「yes」を指定しておけばいい。

❻ACCEPT_ADDRSはそのホスト宛てのローカルなメールであると判断するドメイン名の指定である。「'$m'」を指定して、このドメイン宛てのものを受け取れるようにする。

❼ACCEPT_LOWERを「yes」にしておくと、ACCEPT_ADDRSの前にさらにホスト名などが付いているときも、このホスト宛てのローカルな電子メールだと判断する。

❽あとは、配送エラーとなったメールのヘッダー部分を管理者にメールするように、COPY_ERRORS_TOを指定しておいてもよい。たとえば、「'postmaster'」と指定する。

OFFICIAL_NAMEとMY_DOMAINなどについては、初期設定の動作で問題はないので、何も指定しなくていい。

以上の設定を行っておけば、今回のような単純なネットワーク構成の場合は問題ない。運用後にさらなる拡張が必要になったら、INTRO.jpnファイルのほかにMANUAL.jpnファイルを参照しながら編集作業を行えばいい。

リスト4 ●sendmail.def

```
  :
### type of sendmail.cf
CF_TYPE=R8V7 ―――――――――――――①
  :
# version number for Received: header line
#VERSION=3.7W
VERSION_SEPARATOR='-' ―――――――――――――――――②
#LOCAL_VERSION=
##LOCAL_VERSION=`date +%D'
LOCAL_VERSION=`date +%y%m%d%H' ―――――――――――――②
  :
# [ostype]
# OS type (choose a file name in ostype directory)
OS_TYPE=linux-redhat ――――――――――――――――――③
  :
# [address]
  :
# default from-address (can be $j, $m or another generic address)
#FROM_ADDRESS='$j'
FROM_ADDRESS='$m' ――――――――――④
# apply FROM_ADDRESS for receipients (yes/no)
RECIPIENT_GENERIC=yes ――――――――⑤
  :
# [acceptaddr]
# addresses which should be accepted as local
ACCEPT_ADDRS='$m'       # can be used with R8 sendmail ――――――⑥
##ACCEPT_ADDRS=$MY_DOMAIN
##ACCEPT_ADDRS='accept.domain.name1 accept.domain.name2 ...'
  :
# accept mails for all hosts within my domain
# (yes/no/"a specific domain with leading dot"; default for "yes" is .$m)
ACCEPT_LOWER=yes ――――――――――⑦
  :
# [bitnet]
# resolve BITNET traffic (static/mx/no/auto)
#BITNET=no
BITNET=auto
#BITNET_RELAY='bitnetjp.ad.jp'
##BITNET_RELAY='dom.bitnetjp.ad.jp'
  :
# who (if anyone) should get extra copies of error messages
# <OldStyleHeaders>          ――――――――――――――――⑧
COPY_ERRORS_TO='postmaster'
  :
```

セキュリティーを強化しよう

Red Hatのインストールとサーバーソフトの設定が済んでも、それで終わりではない。もっとも重要なセキュリティーの問題が残っている。ここでは、パケットフィルタリング、ログの記録、inetd.confの設定などについて見ていこう。

必要なのはセキュリティーポリシー

最近は、自宅や小規模オフィスでも自前の専用線を持ち、インターネットにアクセスできる環境が整ってきた。そこで既存のSOHO向け解説を読み、専用線を引き、機材に電源を入れ、サーバープログラムを動かす。どうにかうまく動いているようだ。だがこれでサイト立ち上げ完了とはいかない。実は大きな落とし穴がある。セキュリティーの問題だ。

セキュリティー対策は、大きく分けて3つのパートからなる。

①どのようなセキュリティーポリシーを持つのかを決める。
②そのセキュリティーポリシーを実現するためのシステムを構築する。
③構築したシステムを日々メンテナンスする。

「セキュリティーポリシー」とは何か一言で言うと、自分が管理するサイトの何をどのように保護するのかという取り決めだ。

一番明確で確実なポリシーは、外部からサイト内には一切アクセスさせないというものだ。これを実現するには、サイトセキュリティーを考慮してDNSサーバーや電子メールサーバーといったものをプロバイダー側で用意しているサービスを利用すればよい。

しかし、このようなサービスのないプロバイダーを利用する場合や、自分のサイトでDNSサーバーや電子メールサーバーなどを持つ場合、本格的なセキュリティー対策が必要となってくる。その場合は、セキュリティーポリシーという明確な指針が必要になる。いきあたりばったりにやっていては、最後にはつじつまが合わなくなる危険性があるからだ。

①外部から内部への接続で許すものは何か
②内部から外部への接続で許すものは何か
③保護対象となるコンピュータでどのようなサービスを提供し、どの情報を保護するのか

最低でも以上3点を明確にするべきだ。

実現するにあたり、本来は全体のネットワーク構造も十分に実用に耐えうるような構造にする必要がある（図1）。しかし、今回は最小限のシステム構成（図2）でどのようにセキュリティーを高めるかについてワンポイント的に解説を行うことにする。このネットワーク構成はSOHOではよく見かける構成だが、セキュリティーを前提とした場合は、必ずしも推薦はできないことをあらかじめ了解してほしい。

図1 ●本格的なネットワーク構成

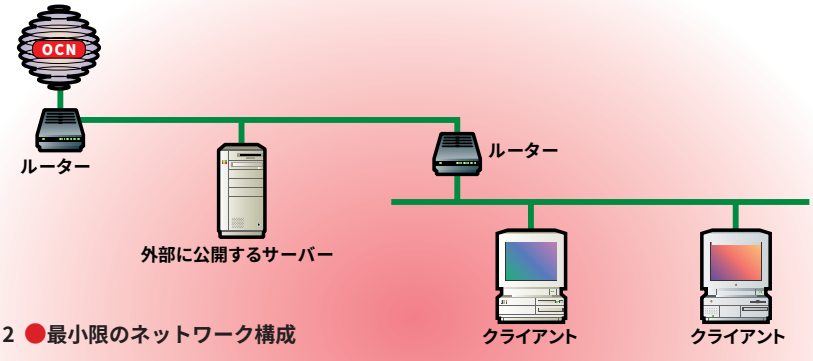
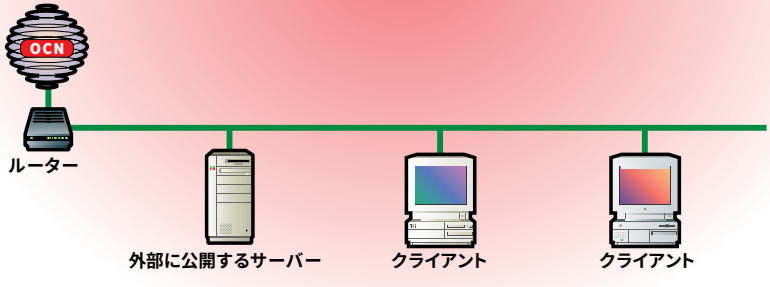


図2 ●最小限のネットワーク構成



パスワードの設定

ここではルーターとしてMN128-SOHOを例に取り上げる。最初に、ルーターのパスワードを設定しておくこと（画面1）。これは単純だが、非常に重要なことだ。パスワード設定すると、それ以降設定画面にアクセスするときにパスワードが必要になる。

画面1 ●ルータ設定→本体設定

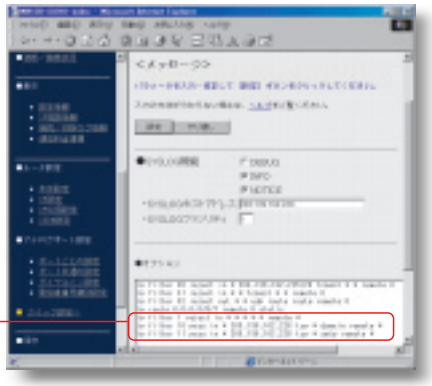
ルーターで パケットフィルタリング

最初に行うべきことは、サイトの入口となるルーターで不必要なIPパケットをフィルタリングすることだ。MN128-SOHOでは最大32個の設定が可能だが、SOHOレベルでは32もあれば十分だ。

まず、外部からのすべてを遮断するフィルターを設定し、次に必要なパケットを通すようにする。同様に、内部から外部へのパケットに対しても必要に応じて行う。ここでは、外部からのフィルタリングの例を載せる（画面2）。

```
外部からのパケットをすべてを遮断
ip filter 1 reject in * * * * * remote *
サーバーへのDNSのみ通過
ip filter 10 pass in * 203.139.162.226 tcp * domain remote *
サーバーへのSMTPのみ通過
ip filter 11 pass in * 203.139.162.226 tcp * smtp remote *
（この例では外部公開サーバーのIPアドレスを 203.139.162.226 としている）
```

画面2　ルータ設定→IP応用設定



SYSLOGも取ろう

最近のSOHO向けルーターでもきちんとログを記録できるようになっている。これはUNIXのsyslog機能を応用している。MN128-SOHOではINFOとNOTICEをチェックし、SYSLOGホストアドレスを設定する（この例では203.139.162.226）。SYSLOGファシリティはそのままにしておく（画面2）。

サーバー側では、リスト1のように/etc/rc.d/init.d/syslogルを設定し、リスト2のように再起動する。この例では/var/log/messagesにMN128のログが記録される。

リスト1 ●/etc/rc.d/init.d/syslog

```
    start)
        echo -n "Starting system loggers: "
        daemon syslogd -r
                       -rを付ける
```

リスト2

```
% /etc/rc.d/init.d/syslog restart
```

Linuxを 外部公開用サーバーにする

攻撃されるとするならば、外部にさらされているマシンが最大のターゲットになる。すでにルーターでIPフィルタリングが行われてアクセスが制限されているが、それとは独立して、このサーバー独自でセキュリティーが保たれるようにするべきだ。

2つ基本的な話をまずしたい。1つ目はソフトウェアのぜい弱性のことだ。古いソフトウェアにはセキュリティーに関するバグが存在する場合がある。ソフトウェアは対策が施されている最新のものを使おう。

2つ目はパスワードを隠すシャドウパスワードのことだ。まだLinuxは初期設定ではシャドウパスワードではないので、管理者の手でシャドウパスワード化する必要がある（リスト3）。

リスト3

```
# cd /etc
# /usr/sbin/pwconv
```

サイトセキュリティーとして、なすべきことはいろいろあるが、「不必要なネットワークサービスは行わない」ということを取り上げたい。ここでのネットワークサービスはinetd.confで自動的に立ち上がるサービスのことに限定しよう。まずリスト4のコマンドを実行してほしい。

リスト4

```
% grep -v -e '^#' /etc/inetd.conf
```

リスト5 ●/etc/inetd.conf

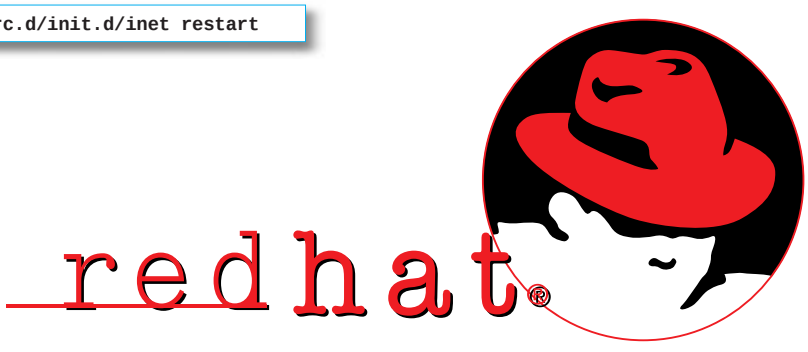
```
#ftp     stream  tcp     nowait  root    /usr/sbin/tcpd  in.ftpd -l -a
#telnet  stream  tcp     nowait  root    /usr/sbin/tcpd  in.telnetd
#gopher  stream  tcp     nowait  root    /usr/sbin/tcpd  gn
```

リスト6

```
# /etc/rc.d/init.d/inet restart
```

現在有効なサービスの一覧が表示される。TELNET、FTP、fingerなどのリストが表示されているだろう。

/etc/inetd.confの中で必要のないものは全部コメントアウトしてしまい（リスト5）、動いているinetdに反映させる（リスト6）。今回想定している環境ではPOP3とIMAP4しか使わない。FTPやfingerは不必要なのでコメントアウトする。もし、コンソールからしかログインしないのなら、TELNETもコメントアウトする。



tcp_wrapperできめ細かく管理しよう

ひととおりセキュリティーの設定が終われば、いよいよサーバーを外部に公開することになる。この時点から日々の管理が始まる。ここではtcp_wrapperを使った接続の許可と拒否、アクセスの監視の方法について見ていこう。

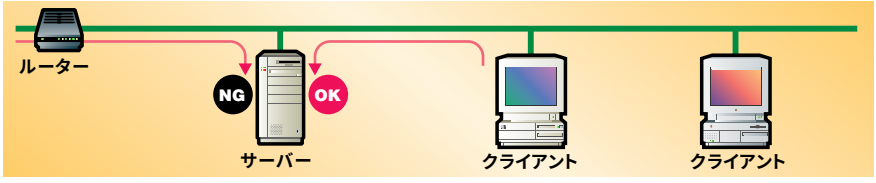
なぜtcp_wrapperを使うのか

基本的な設定はできた。しかし、前ページまでの作業は、UNIXに以前から備わっていた基本的な機能を応用してセキュリティーを高めたという側面が強い。

そのような受身でのセキュリティー強化だけでは、どうしても足りない部分が出てくる。そこで、セキュリティーに特化したツールを用いて積極的に防御を固めるときに登場するのがtcp_wrapperだ。

tcp_wrapperは、サービスやポートに対してアクセス制御および監視を行うツールだ。「wrapper」の名前のとおり、既存のデーモンプログラムを「包む」ような方法で制御や監視を実現する。まず、このtcp_wrapperを覚えよう。

図 ●tcp-wrapperの働き



たとえば、TELNETのサービスを提供するとき、右の図のように、特定のアドレス、つまり内部のクライアントからのアクセスのみを許し、(万が一、フィルタリングをすり抜けてきた) 外部からのアクセスを許さないといった設定ができる。

さて、Red Hat 5.2ではすでにインストール時にtcp_wrapperであるtcpdがインストールされている。リスト1でインストールが確認できたら、あとは自分の環境に合わせて設定を行えばいい。

リスト1

```
% grep 'telnet' /etc/inetd.conf
telnet  stream  tcp     nowait  root    /usr/sbin/tcpd  in.telnetd
                                          すでにインストール済み
```

接続の許可と拒否

まず、tcpdの設定がどうなっているかをチェックするtcpdchkを使ってみよう（リスト2)。まだ、何も設定していないので、設定に関するメッセージが出ないか、あるいはサーバープログラムがインストールされていない/etc/inetd.confにあるエントリーが警告されるだけだろう。

接続の許可を設定するファイルは、/etc/hosts.allow、拒否を設定するファイルは/etc/hosts.denyだ。まずは、hosts.denyで確実にすべてのマシンからの接続を拒否させる（リスト3)。次にhosts.allowでサイト内にあるローカルなマシンからの接続は許可する（リスト4)。

もう一度tcpdchkを実行してみよう。設定が表示されるはずだ（リスト5)。クライアントはローカルのみ許し、それ以外は接続を一切拒否しているのがわかる。もう少しチェックしてみよう。特定のホストに対してきちんと有効になっているかをtcpdmatchを使いチェックしてみる（リスト6)。外部にあるホストに対してもチェックできる（リスト7)。もちろん、アクセスはできない。

リスト2

```
# /usr/sbin/tcpdchk -v
Using network configuration file: /etc/inetd.conf
```

リスト5

```
# /usr/sbin/tcpdchk -v
Using network configuration file: /etc/inetd.conf

>>> Rule /etc/hosts.allow line 7:
daemons:  ALL
clients:  LOCAL
access:   granted

>>> Rule /etc/hosts.deny line 10:
daemons:  ALL
clients:  ALL
access:   denied
```

リスト3 ●/etc/hosts.deny

```
ALL: ALL
```

リスト4 ●/etc/hosts.allow

```
ALL: .internetmag.co.jp
```

tcp_wrapperは、適当なポートにトラップをかけておくような設定が「/etc/hosts.allow」と「/etc/hosts.deny」で可能なので、ポートスキャンのようなあやしい走査に対してrootへメールで警告を発することができる（ただし、今回の記事の範囲ではすでにフィルタリングしている)。

リスト6

```
% /usr/sbin/tcpdmatch in.telnetd client1
warning:  client1: hostname alias
warning:  (official name:
client1.internetmag.co.jp)
client:   hostname
client1.internetmag.co.jp
client:   address  203.139.162.229
server:   process  in.telnetd
matched:  /etc/hosts.allow line 7
access:   granted
```

リスト7

```
% /usr/sbin/tcpdmatch in.telnetd www.*******.co.jp
warning: www.*******.co.jp: hostname alias
warning: (official name: *****.*******.co.jp)
client:   hostname *****.*******.co.jp
client:   address  192.***.90.1
server:   process  in.telnetd
matched:  /etc/hosts.deny line 10
access:   denied
```

ログを監視する

Red Hat Linuxでは、tcpdのログは/var/log/secureに残る。リスト8はその例だ。こうしたログに目を光らせることによってあやしいアクセスや振るまいは事前にチェックできるだろう。

tcp_wrapperは積極的にセキュリティーを確保するためのツールなので、いろいろな応用が可能だ。ログにアクセス状況を残すだけではなく、特定のポートにトラップ（罠）を仕掛け、あやしいアクセスがあれば、管理者へ警告を送ることもできる。さらに応用を効かせて、ポケベル呼び出しと連動するような拡張も可能だ。

リスト8 ●/var/log/secure

```
1: Oct 29 12:25:27 ns in.telnetd[550]: connect from
client1.internetmag.co.jp
2: Oct 29 12:25:41 ns login: FAILED LOGIN 1 FROM client1 FOR hironobu,
Authentication failure
3: Oct 29 12:54:06 ns in.telnetd[624]: refused connect from 192.***.90.1
```

1: client 1からtelnetで接続できたという意味。

2: client 1からrloginで接続していたhironobuというユーザーのパスワードが間違っている（rloginのサービスを有効にしてテストしている）。

3: 192.***.90.1というアドレスからtelnetの接続要求があったが拒否した。

セキュリティーに終わりはない

ここで説明したセキュリティーの設定は参考になるだろうが、残念ながら表面的な一例だ。本来はきちんとセキュリティーポリシーを明確にしたあと、対策を施す必要がある。ネットワーク構造もきちんとしたほうがよい。今回は、外部への露出がDNSとSMTPのみであり、また内部からの攻撃も考慮していない。実際に運用されているサイトではFTPやWWWなども行っているだろう。また、SOHOといっても複数の人間が使っているネットワークであれば、内部からの攻撃に対しても考慮する必要がある。

また、日々の運用に関しての詳細も、今回は残念ながら言及していない。今後、根本的な部分から1つ1つ掘り起こしつつサイトセキュリティーに関して具体的にどうするかを考えていくことが必要だ。

乞うご期待！ 運用編、セキュリティー編

インターネットマガジンでは今後もLinuxの記事を取り扱っていくことを計画している。運用編としてWWWサーバーなどの構築方法を、セキュリティー編として運用していくためのノウハウをそれぞれ解説する予定だ。
今回のインストールやサーバーの設定がうまくいった人はぜひ期待してほしい。
これを読めばLinuxサーバー構築のテクニックが身に付くこと間違いなしだ。

［インターネットマガジン　バックナンバーアーカイブ］ ご利用上の注意

このPDFファイルは、株式会社インプレスR&D（株式会社インプレスから分割）が1994年～2006年まで発行した月刊誌『インターネットマガジン』の誌面をPDF化し、「インターネットマガジン　バックナンバーアーカイブ」として以下のウェブサイト「All-in-One INTERNET magazine 2.0」で公開しているものです。

http://i.impressRD.jp/bn

このファイルをご利用いただくにあたり、下記の注意事項を必ずお読みください。

- 記載されている内容（技術解説、URL、団体・企業名、商品名、価格、プレゼント募集、アンケートなど）は発行当時のものです。
- 収録されている内容は著作権法上の保護を受けています。著作権はそれぞれの記事の著作者（執筆者、写真の撮影者、イラストの作成者、編集部など）が保持しています。
- 著作者から許諾が得られなかった著作物は収録されていない場合があります。
- このファイルやその内容を改変したり、商用を目的として再利用することはできません。あくまで個人や企業の非商用利用での閲覧、複製、送信に限られます。
- 収録されている内容を何らかの媒体に引用としてご利用する際は、出典として媒体名および月号、該当ページ番号、発行元（株式会社インプレスR&D）、コピーライトなどの情報をご明記ください。
- オリジナルの雑誌の発行時点では、株式会社インプレス R&D（当時は株式会社インプレス）と著作権者は内容が正確なものであるように最大限に努めましたが、すべての情報が完全に正確であることは保証できません。このファイルの内容に起因する直接的および間接的な損害に対して、一切の責任を負いません。お客様個人の責任においてご利用ください。

このファイルに関するお問い合わせ先
株式会社インプレスR&D
All-in-One INTERNET magazine 編集部
im-info@impress.co.jp