



Debian GNU/Linux入門セミナー

2004年3月

講師 有限会社テンペスト 中村文則
150-0013 渋谷区恵比寿4-5-21 池田聖徳ビル506号
Tel: 03-5789-3477 Fax: 03-5789-3478
Mail: nakamura@tempest.jp
Web: www.tempest.jp

TEMPEST

主催 株式会社ステップ・サポート
153-0013 渋谷区恵比寿4-5-21 池田聖徳ビル
Tel: 03-5475-2900 Fax: 03-5475-2901
Mail: info@step-support.co.jp
Web: www.step-support.co.jp



STEP SUPPORT



The Power of LINUX

Debian GNU/Linux 3.0r2

- 1 Linuxについて
- 2 Debian GNU/Linuxのインストール
- 3 基本的な設定
- 4 基本的な操作方法
- 5 Web Serverの設定
- 6 参考資料



1 Linux について

1.1 Linuxの歴史

LinuxはフィンランドのLinus Torvaldsによって1990年に作成された。小さなUNIXシステム「Minix」を研究し、i386ベースのUNIX OSを作ろうとしたのがLinuxの始まりである。Linusはcomp.os.minixに下記の投稿をし、Linuxプロジェクトへの参加を呼びかけた。

Do you pine for the nice days of Minix-1.1, when men were men and wrote their own device drivers? Are you without a nice project and just dying to cut your teeth on a OS you can try to modify for your needs? Are you finding it frustrating when everything works on Minix? No more all-nighters to get a nifty program working? Then this post might be just for you.

As I mentioned a month ago, I'm working on a free version of a Minix-lookalike for AT-386 computers. It has finally reached the stage where it's even usable (though may not be depending on what you want), and I am willing to put out the sources for wider distribution. It is just version 0.02...but I've successfully run bash, gcc, gnu-make, gnu-sed, compress, etc. under it.

1991年10月5日、Linusは最初の「公式」バージョンであるversion 0.02をアナウンスした。

2.2 ディストリビューション

本来Linuxとはカーネル(OSの中核部分)をさす、これに(主として GNU 起源の)さまざまなライブラリやツール、アプリケーションなどの必要な要素を組み合わせて、UNIX系OSとして使いやすくまとめたものをディストリビューションと呼ぶ。

Slackware
Plamo Linux

Debian GNU/Linux
Vine Linux

Red Hat Linux
Turbolinux
LASER5 Linux
Miracle Linux
SuSE

など



2

Red Hat Linuxのインストール

2.1 インストール

Debian GNU/Linuxのインストール。インストールCDをセットし起動する。ネットワークが利用できる環境であれば1枚のCDでインストールは問題なく可能。

2.1.1 インストーラの起動

CDをセットし起動する。

2.1.2 カーネル2.4の選択

画面下に「boot:」と出たら「bf24」と入力し進む。これによりカーネル2.4を利用してのインストールとなる。

2.1.3 Choose The Language

「ja - 日本語で設定を～」をカーソルキーで選択し進む。

2.1.4 リリースノート

そのまま進む。

2.1.5 インストールメインメニュー

そのまま「キーボードの設定」で進む。

2.1.6 キーボードの選択

「qwerty/jp106 : 日本語」を選択し進む。

2.1.7 インストールメインメニュー

そのまま「ハードディスクのパーティションを切る」で進む。

2.1.8 ディスクドライブの選択

そのまま「/dev/hda」でプライマリ マスターのハードディスクを指定し進む。

2.1.9 LIL0の制限

そのまま進む。

2.1.10 ReiserFSのJournalが要求する追加スペースについて

そのまま進む。

2.1.11 /bootパーティションの作成

タブキーを利用して、画面下にある「New」を選択し進む。次に「Primary」を選択し進む。Sizeを「100」と入力して進み、次の画面では「Beginning」で進む。これによりハードディスクの先頭に100MBのパーティションが作成される。

そのままリストでhda1が選択されている状態で「Bootable」を選択することで、このパーティションからの起動が可能となる。



2.1.12 SWAPパーティションの作成

リストの画面で空の部分「Pri/Log Free Space」を選択する。「New」「Primary」と進み、Sizeは「600」と入力し、次の画面では「Beginning」で進む。

そのままリストでhda2が選択されている状態で「Type」を選択して進む。Enter filesystem type: の部分は「82」のまま進む。これによりこのパーティションがSWAP領域として利用可能となる。

2.1.13 /パーティションの作成

リストの画面で空の部分「Pri/Log Free Space」を選択する。「New」「Primary」と進み、Sizeはそのまま(残りすべて)で、次の画面では「Beginning」で進む。

必要に応じてこの手順でパーティションを作っていく。

最後に「Write」で進み、「yes」と入力することでハードディスクへのパーティション作成が開始される。「Quit」で進む。

2.1.14 インストールメインメニュー

そのまま「スワップパーティションの初期化と有効化」で進む。

2.1.15 不良ブロックのスキャンを行いますか？

「いいえ」で進む。

2.1.16 よろしいですか？

「はい」で進む。

2.1.17 インストールメインメニュー

そのまま「Linuxパーティションの初期化」で進む。

2.1.18 ファイルシステムタイプの選択

「Ext3」を選択し進む。

2.1.19 パーティションの選択

先ほど作成した「/dev/hda3」を選択し進む。

2.1.20 不良ブロックのスキャンを行いますか？

「いいえ」で進む。

2.1.21 よろしいですか？

「はい」で進む。

2.1.22 ルートファイルシステムとしてマウントしますか？

「はい」で進む。

2.1.23 インストールメインメニュー

「別手順: Linuxパーティションの初期化」、「Ext3」を選択し進む。

2.1.24 マウントポイントの選択

「/boot」を選択し進む。



2.1.25 インストールメインメニュー

そのまま「カーネルとドライバモジュールのインストール」で進む。

2.1.26 CD-ROMを検出しました。

「はい」で進む。

2.1.27 インストールメインメニュー

そのまま「デバイスドライバモジュールの設定」で進む。

2.1.28 カテゴリの選択

そのまま「Exit ここでの選択を終了し、」で進む。追加したいドライバがある場合はこの画面で指定する。

2.1.29 インストールメインメニュー

そのまま「ネットワークの設定」で進む。この画面が出ない場合はNIC用の適切なドライバが設定されていないということになる。

2.1.30 ホスト名を選択してください

必要に応じてホスト名を指定する。

2.1.31 ネットワークの自動設定

「いいえ」を選択し進む。これによりDHCPは利用されない。

2.1.32 IPアドレスを選択してください

「192.168.0.X」と入力し進む。

2.1.33 ネットマスクを選択してください

「255.255.255.0」と入力し進む。

2.1.34 IPゲートウェイアドレスは何ですか？

「192.168.0.1」と入力し進む。

2.1.35 ドメイン名を選択してください

必要に応じてドメイン名を指定する。

2.1.36 DNSサーバのアドレスを選択してください

環境に合わせて設定する。

2.1.37 インストールメインメニュー

そのまま「基本システムのインストール」で進む。

2.1.38 インストールメインメニュー

そのまま「システムを起動可能にする」で進む。

2.1.39 LILOブートローダをどこにインストールしますか？

そのまま「LILOをMBRにインストールする」で進む。



2.1.40 ブートフロッピーの作成

2HDのフロッピーをセットして、そのまま「ブートフロッピーの作成」で進む。

2.1.41 インストールメインメニュー

そのまま「システムのリブート」で進む。

2.1.42 システムをリブートしますか？

CD-ROMとフロッピーを取り出してから「はい」で進む。

システムの再起動後、設定を行う。これ以降のメニューは英語となる。

2.1.43 Debian System Configuration

<OK>で進む。

2.1.44 Time Zone Configuration

GMTではないので<No>で進む。AsiaよりTokyoを選択する。

2.1.45 Password setup

md5を利用するので<Yes>で進む。shadow passwordsを利用するのでこれも<Yes>で進む。

2.1.46 Password setup

ここでrootユーザのパスワードを指定する。その後、一般ユーザの作成画面となるので必要に応じて作成する。

2.1.47 Debian System Configuration

PCMCIAのパッケージを削除するかと聞かれる。PCMCIAは利用しないので<Yes>で進む。次にPPPの利用に関して聞かれるが使用しないので<No>で進む。

2.1.48 Apt configuration

aptコマンドでアーカイブを取得する先を指定する。ここではhttpを選択し進む。non-US software, non-free software, contrib softwareの利用について聞かれるがすべて<Yes>で進む。

国の選択画面ではJapanを選択し、次に国内の適当なミラーサイトを選択する。プロキシは利用していないので空行のまま進む。これにより設定したサイトへアクセスが開始される。

apt sourceを追加について聞かれる画面は<No>で進む。security updatesは利用するため<Yes>で進む。

2.1.49 Debian System Configuration

taskselを起動しますかと出るが、<No>で進む。次にdselectの起動しますかと出るがこれも<No>で進む。

先ほど指定したPCMCIAパッケージの削除確認画面はそのまま進む。作業後、不要なファイルの削除の確認画面となる、そのまま進む。

2.1.50 Exim

メールサーバeximの設定画面となるが、利用しないので「5」を選択して進む。



2.1.51 Debian System Configuration

設定は終了となり、ログイン画面が表示されインストールは完了となる。なお、baseconfigコマンドを利用することで、再設定が可能。

2.2 基本パッケージの追加

2.2 dselect

dselectを利用して基本パッケージを追加する。

```
# dselect
```

- * 0. [A]ccess Choose the access method to use.
- 1. [U]pdate Update list of available packages, if possible.
- 2. [S]elect Request which packages you want on your system.
- 3. [I]ninstall Install and upgrade wanted packages.
- 4. [C]onfig Configure any packages that are unconfigured.
- 5. [R]emove Remove unwanted software.
- 6. [Q]uit Quit dselect.

0番から順番に実行していく、Accessではaptを利用する、Selectの画面では基本的なパッケージはすべて選択してあるのでそのまま進む。



2.3 基本設定

2.3.1 ユーザ登録

グループを作成するにはaddgroupコマンドを利用する。

```
# addgroup hotel
Adding group hotel (101)...
Done.
```

ユーザを作成するにはadduserコマンドを利用する。hotelグループに所属するユーザbontenを作成する場合が下記の例。そのままパスワードを指定する。

```
# adduser --ingroup hotel bonten
Adding user bonten...
Adding new user bonten (1002) with group hotel.
Creating home directory /home/bonten.
Copying files from /etc/skel
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for bonten
Enter the new value, or press return for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [y/n] y
```

プログラムの起動時のみに利用するログインできないユーザを作る場合。

```
# adduser --system --home /dev/null named
Adding system user named...
Adding new user named (102) with group nogroup.
Home directory /dev/null already exists.
```

2.3.2 rootへの変更

一般ユーザからsuコマンドを利用し、rootに変更することができる。

```
$ /bin/su -
パスワード入力
# スーパーユーザになるとプロンプトの記号"$"が"#"になることを確認する。
```

2.3.3 /etc/resolv.confの設定

/etc/resolv.confを設定し、ドメイン引きが出来るようにする。現在設定されている自己アドレスを一時使用停止し外部のDNSを設定する。DNSは最大3登録できる。

```
# vi /etc/resolv.conf
#nameserver 192.168.0.
nameserver xxx.xxx.xxx.xxx
nameserver xxx.xxx.xxx.xxx
```

修正後、hostコマンドなどを利用しDNSにアクセスできることを確認する。

```
# host www.sample.com
```



2.3.4 NICの設定

ネットワークの設定は/etc/network/interfacesに保存されている。必要に応じて変更する。

```
# cat /etc/network/interfaces
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)

# The loopback interface
auto lo
iface lo inet loopback

# The first network card - this entry was created during the Debian
installation
# (network, broadcast and gateway are optional)
auto eth0
iface eth0 inet static
    address 192.168.0.11
    netmask 255.255.255.0
    network 192.168.0.0
    broadcast 192.168.0.255
    gateway 192.168.0.1
```

変更した場合は下記コマンドでネットワークの再起動を行う。

```
# /etc/init.d/networking restart
```

2.3.5 bashの設定

.bashrcを利用してbashを使いやすくする。修正後、sourceコマンドで設定を反映させる。

```
# vi ~/.bashrc
```

```
alias ls='ls -F --color=auto'
alias ll='ls -l'
alias la='ls -a'
```

```
# source ~/.bashrc
```

2.3.6 エディタの変更

debianではデフォルトのエディタがnanoになっており、crontabコマンドなどを実行した際にnanoが自動的に使われる。より広く使われているvi(nvi)にエディタを変更する。

```
# update-alternatives --config editor
```

There are 3 programs which provide `editor`.

Selection	Command
*+ 1	/usr/bin/nano
2	/bin/ed
3	/usr/bin/nvi

Enter to keep the default[*], or type selection number: 3 指定する



2.3.7 DMAの利用

初期設定ではDMAが利用されておらずHDDの性能が活かされていない。hdparmコマンドをインストールして、DMA機能をONにする。

```
# apt-get install hdparm
Reading Package Lists... Done
Building Dependency Tree... Done
The following NEW packages will be installed:
~略~
```

現在の設定値の確認。

```
# hdparm /dev/hda

/dev/hda:
multcount      = 0 (off)
I/O support    = 0 (default 16-bit)
unmaskirq      = 0 (off)
using_dma      = 0 (off)
keepsettings   = 0 (off)
nowerr         = 0 (off)
readonly       = 0 (off)
readahead      = 8 (on)
geometry       = 913/255/63, sectors = 14668290, start = 0
busstate       = 1 (on)
```

現在の状態でテストする。

```
# hdparm -t /dev/hda
```

設定を変更する。変更後もう一度テストしてみる。

```
# hdparm -c 1 -d 1 /dev/hda

/dev/hda:
setting 32-bit I/O support flag to 1
setting using_dma to 1 (on)
I/O support    = 1 (32-bit)
using_dma      = 1 (on)
```

起動時に自動的にDMAがオンになるように設定する。

```
# vi /etc/init.d/hdparm
#!/bin/sh
/sbin/hdparm -c 1 -d 1 /dev/hda

# chmod +x /etc/init.d/hdparm
# update-rc.d hdparm start 90 S .
```

2.3.8 モジュールの設定

debianではモジュールの追加削除は下記コマンドを利用する。直接/etc/modules.confを編集してはいけない。

```
# modconf
```



2.4 セキュリティ対策

2.4.1 sources.listの設定

aptコマンドでは/etc/apt/sources.listを元にパッケージのアップデートなどを行う。このファイルで最適なソースを指定すると効率がよくなる。設定にあたりnetselectをインストールして利用する。

```
# apt-get install netselect
```

netselectとlynxコマンドを使ってftpサーバを検索する。

```
# netselect -vv `lynx -dump http://www.debian.or.jp/debian-ftp-mirrors-jp`
```

結果に基づき、/etc/apt/sources.listを修正する。

```
deb ftp://sample.jp/pub/linux/debian/debian stable main contrib non-free
deb ftp://sample.jp/pub/linux/debian/debian-non-US stable/non-US main
contrib non-free
```

netselect-aptコマンドを利用することで最適なサイトを記述したsources.listを自動的に作成することも可能。wgetコマンドが必要なのでインストールしてから利用する。

```
# apt-get install wget
# mkdir /tmp/sources
# cd /tmp/sources
# netselect-apt
# ls
mirrors_full sources.list
```

作成されたsources.listはそのまま/etc/aptにコピーして利用できる。

2.4.2 パッケージの更新

apt-getコマンドでリストを更新する、定期的に行うことが望ましい。

```
# apt-get update
Hit http://ring.asahi-net.or.jp stable/main Packages
Hit http://ring.asahi-net.or.jp stable/main Release
Hit http://ring.asahi-net.or.jp stable/non-free Packages
~略~
```

その後パッケージの更新を実行する。

```
# apt-get -u upgrade
Reading Package Lists... Done
Building Dependency Tree... Done
Calculating Upgrade... Done
0 packages upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```



2.4.3 rootになれるユーザを限定する。

BSD系のOSと違い多くのLinuxディストリビューションでは一般ユーザがすべて/bin/suコマンドでroot権限になることができる。ここで登録されたユーザのみがrootになれるように設定を変更する。まず指定のユーザをrootグループに所属させる。

```
# usermod -G root shibuya
```

下記の行を修正する。これによりwheelグループに登録されていないユーザはrootへの変更ができなくなる。

```
# vi /etc/pam.d/su
```

```
# Uncomment this to force users to be a member of group root
# before than can use `su'. You can also add "group=foo" to
# to the end of this line if you want to use a group other
# than the default "root".
# (Replaces the `SU_WHEEL_ONLY' option from login.defs)
# auth      required    pam_wheel.so    頭の#を取る
```

2.4.4 rootでのSSHリモート接続拒否

初期設定ではリモートからrootが直接接続できる設定になっているが、これを修正する。修正後、sshdの再起動で設定が反映される。

```
# vi /etc/ssh/sshd_config
```

```
PermitRootLogin yes
PermitRootLogin no    #を削除しyesをnoとする
```

```
# /etc/init.d/ssh restart
```

2.4.5 ポートスキャン

ポートスキャンコマンド、nmapを利用して不必要なポートが開いていないことを確認する。nmapはapt-getコマンドでインストールして利用できる。

```
# apt-get install nmap
Reading Package Lists... Done
Building Dependency Tree... Done
~ 略 ~
```

nmapを自機(localhost)に向けて実行した例。デフォルトでは多くのポートが開いていることが分かる

```
# nmap localhost
```

```
Starting nmap V. 2.54BETA31 ( www.insecure.org/nmap/ )
Interesting ports on localhost (127.0.0.1):
(The 1546 ports scanned but not shown below are in state: closed)
Port      State      Service
9/tcp     open       discard
13/tcp    open       daytime
22/tcp    open       ssh
25/tcp    open       smtp
37/tcp    open       time
111/tcp   open       sunrpc
113/tcp   open       auth
515/tcp   open       printer
```

```
Nmap run completed -- 1 IP address (1 host up) scanned in 1 second
```



2.4.6 unnecessaryサービスの停止

不要なサービスを起動しているとセキュリティ上問題があるので、必要なもののみを起動する。

inetdを止める。

```
# /etc/init.d/inetd stop
```

lpd, portmapを止める。実行後、nmapを再度実行し結果を比較してみる。

```
# /etc/init.d/lpd stop
```

```
# /etc/init.d/portmap stop
```

2.4.7 unnecessaryサービスを起動させない

起動しない用に設定を変更する。

```
# update-rc.d -f inetd remove
```

```
update-rc.d: /etc/init.d/inetd exists during rc.d purge (continuing)
```

```
Removing any system startup links for /etc/init.d/inetd ...
```

```
  /etc/rc0.d/K20inetd
```

```
  /etc/rc1.d/K20inetd
```

```
  /etc/rc2.d/S20inetd
```

```
  /etc/rc3.d/S20inetd
```

```
  /etc/rc4.d/S20inetd
```

```
  /etc/rc5.d/S20inetd
```

```
  /etc/rc6.d/K20inetd
```

```
# update-rc.d -f lpd remove
```

```
# update-rc.d -f portmap remove
```

```
# update-rc.d -f nfs-common remove
```

```
# update-rc.d -f nfs-kernel-server remove
```

2.4.8 Windows用SSHクライアント

WindowsからSSHを利用してアクセスするにはPuTTYを利用する。下記サイトより"putty.exe"をダウンロードすればよい。特にインストールの作業は必要無く、そのまま起動できる。

<http://www.chiark.greenend.org.uk/~sgtatham/putty/>

2.4.9 Macintosh用SSHクライアント

MacintoshからSSHを利用してアクセスするにはMacSSHを利用する。下記サイトよりダウンロードすればよい。

<http://www.macssh.com/>



4

基本的な操作方法

4.1 基本的なコマンド

4.1.1 コマンド

ls ディレクトリ内のファイルを一覧表示する

```
$ ls          通常の一覧
$ ls -l       詳細な一覧
$ ls -a       隠しファイルを含むすべての一覧
               ls -laなど組み合わせての使用も可
```

pwd カレントディレクトリの絶対パスを表示する。

cd ディレクトリを移動する

```
$ cd /etc     /etcに移動する
$ cd ..       ひとつ上のディレクトリに移動する
$ cd ~        ホームディレクトリに移動する
$ cd -        直前のディレクトリに移動する
```

less ファイルを閲覧する。

```
$ less apple.txt
apple.txtを開く、qで終了する。
カーソルキーまたはj、kで上下スクロール。
スペースキーまたはctrl-fで1画面先へ、ctrl-bで1画面戻る。
/apple (enter): appleという文字を下に向かって検索する
?apple (enter): appleという文字を上に向かって検索する。
```

cp コピーする

```
$ cp banana.txt melon.txt
           ファイルbanana.txtをmelon.txtという名前でカレントディレクトリにコピーする。
$ cp banana.txt /tmp/melon.txt
           ファイルbanana.txtをmelon.txtという名前で/tmpにコピーする。
$ cp /tmp/banana.txt ./
           /tmpにあるファイルbanana.txtをカレントディレクトリにコピーする。
$ cp banana.txt ../melon.txt
           ファイルbanana.txtをmelon.txtという名前でひとつ上のディレクトリにコピーする。
$ cp -R mango /tmp/mango
           カレントディレクトリにあるmangoというディレクトリを/tmpにディレクトリごとコピーする。
```

**mv** 移動する

```
$ mv banana.txt /tmp/
    カレントディレクトリにあるファイルbanana.txtを/tmpディレクトリに移動する。
$ mv mango /tmp/
    カレントディレクトリにあるmangoというディレクトリを/tmpに移動する。
$ mv banana.txt melon.txt
    banana.txtというファイル名をmelon.txtという名に変更する。
```

mkdir ディレクトリを作成する

```
$ mkdir mango
    カレントディレクトリにmangoというディレクトリを作る。
$ mkdir /tmp/mango
    /tmpの中にmangoというディレクトリを作る。
```

rm 削除する。

```
$ rm banana.txt
    ファイルbanana.txtを削除する。
$ rm -i banana.txt
    確認後、ファイルbanana.txtを削除する。
$ rm -r mango
    ディレクトリmangoを削除する。
```

chown 所有者を変更する。

```
$ chown apple banana.txt
    ファイルbanana.txtの所有者をappleに変更する。
$ chown apple *
    カレントディレクトリ内のすべての所有者をappleに変更する。
$ chown apple mango
    ディレクトリmangoの所有者をappleに変更する。
$ chown -R apple mango
    mangoディレクトリとその中のすべてのファイルの所有者をappleに変更する。
$ chown apple:grape banana.txt
    ファイルbanana.txtの所有者をapple、グループをgrapeに変更する。
```

chgrp グループを変更する。

```
$ chgrp grape banana.txt
    ファイルbanana.txtのグループをgrapeに変更する。
$ chgrp grape *
    カレントディレクトリ内のすべてのグループをgrapeに変更する。
$ chgrp grape mango
    ディレクトリmangoのグループをgrapeに変更する。
```




chmod ファイルモードを設定する。

ファイルへのアクセス権を設定。オーナー、グループ、その他のユーザという3つに対しそれぞれ設定する。読む(r)を4、書く(w)を2、実行する(x)を1で表し、その合計数を使い設定する。

```
$ chmod 444 apple.txt
```

-r--r--r--となりすべての人がこのapple.txtを読む事ができる。

```
$ chmod 440 apple.txt
```

-r--r-----となり所有者と同じグループのユーザのみがこのapple.txtを読む事ができる。

```
$ chmod 400 apple.txt
```

-r-----となり所有者のみがこのapple.txtを読む事ができる。

```
$ chmod 644 apple.txt
```

-rw-r--r--となり所有者はapple.txtを読み、書く事ができるが、それ意外のユーザは読む事のみできる。

```
$ chmod 755 orange.pl
```

-rwxr-xr-xとなり所有者はorange.plを読み、書き、実行する事ができるが、それ意外のユーザは読んで実行はできるが内容を変更する事はできない。

history 使ったコマンドの履歴を表示する。

```
$ history 10
```

直近使った10のコマンドを表示する。!の後に番号を打てば同じコマンドを実行できる。

find ファイルを検索する。

```
$ find /tmp/mango/ -name apple.txt -print
```

ディレクトリ/tmp/mango内でapple.txtを検索し表示する。

```
$ find /tmp/mango/ -name "*.txt" -print
```

ディレクトリ/tmp/mango内で.txtで終わるファイルを検索し表示する。

which コマンドのパスを表示する。

```
$ which perl
```

```
/usr/bin/perl
```

perlが/usr/binにインストールされていることが分かる。

date 日付けを表示する。

```
$ date 現在の日時を表示する。
```

```
# date 12231210
```

日付けを12月23日12時10分に設定する(rootのみ)。

cal カレンダーを表示する。

```
$ cal 1998 1998年のカレンダーを表示する。
```

```
$ cal 8 2001
```

2001年8月のカレンダーを表示する。

ps 実行中のプロセスを表示する。

```
$ ps ax 全プロセスを表示する。
```

```
$ ps ax | grep sshd
```

sshdのプロセスのみ表示する。

kill プロセスを停止させる。

```
$ kill 239 プロセスID239を終了させる。
```

```
$ kill -HUP 3988
```

プロセスID3988をハングアップさせる。



top 実行中のプロセスをリアルタイムで表示する。
q 終了させる

man オンラインマニュアルを表示する。
\$ man less lessのマニュアルを表示する。

head, tail ファイルの最初や最後だけを表示する。
\$ head apple.txt
apple.txtの最初の10行のみを表示する。
\$ tail -n5 apple.txt
apple.txtの最後の5行のみ表示する。

wget ファイルのダウンロード。
\$ wget ftp://ftp.redhat.com/7/i386/glibc-2.2.4-24.i386.rpm
updates.redhat.comより指定のファイルをダウンロードする。

su 管理モードに入る。
\$ /bin/su -
rootの環境でsuになる。"su"コマンドを使用する時にフルパスで指定しないとトロイの木馬などが仕掛けられている場合にrootのパスワードがそのまま取られる可能性があるため常に/bin/suと利用した方がよい。suはsuper user, substitute userの略。

mount CD-ROMやフロッピーをマウントする(rootのみ)。
mount /mnt/cdrom
CD-ROMをマウントする。
mount -t iso9660 /dev/cdrom /mnt/cdrom
/etc/fstabにcdromの設定がない場合。

eject CD-ROMを取り出す(rootのみ)。
eject cdrom
CD-ROMのトレイを開く。
eject -t
CD-ROMのトレイを閉じる。

gzip/gunzip ファイルを圧縮/解凍する。q
\$ gunzip yebisu.tar.gz
yebisu.tar.gzを解凍する。

tar ファイルをまとめる、展開する。
\$ tar xvf yebisu.tar
yebisu.tarからすべてのファイルを取り出す。
\$ tar xvfz yebisu.tar.gz
yebisu.tar.gzからすべてのファイルを取り出す。

ping ネットワークの応答を確認する。
\$ ping 192.168.0.1
IPアドレスを指定しての確認。ネットワーク機能の確認を確認するにはまずローカル(127.0.0.1)、次に自分に割り当てられているIP(192.168.0.)、最後にルータへとpingを利用すると分かりやすい。

reboot リスタートする(rootのみ)。



shutdown システムを停止する(rootのみ)。
shutdown -h now
今すぐシステムを停止する。
shutdown -h +5
5分後にシステムを停止する。
shutdown -r +7
7分後にシステムを再起動する。

crontab cronの設定、表示。
\$ crontab -l cronの設定を表示する。
\$ crontab -e cronを設定する。

4.1.2 UNIXコマンドの参考サイト

JM Project <<http://www.linux.or.jp/JM/>>



4.2 パッケージ管理

4.2.1 apt, dpkgコマンドの使い方

apt, dpkgコマンドでインストールするには"su"になる必要がある。

パッケージの新規インストール

```
# apt-get install wget
Reading Package Lists... Done
Building Dependency Tree... Done
The following NEW packages will be installed:
  wget
0 packages upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0B/15.9kB of archives. After unpacking 77.8kB will be used.
Selecting previously deselected package wget.
(Reading database ... 14517 files and directories currently installed.)
Unpacking wget (from ../wget_2.0.3-1_i386.deb) ...
Setting up wget (2.0.3-1) ...
```

パッケージの削除

```
# apt-get remove wget
Reading Package Lists... Done
Building Dependency Tree... Done
The following packages will be REMOVED:
  wget
0 packages upgraded, 0 newly installed, 1 to remove and 0 not upgraded.
Need to get 0B of archives. After unpacking 77.8kB will be freed.
Do you want to continue? [Y/n]
(Reading database ... 14527 files and directories currently installed.)
Removing wget ...
```

パッケージの再インストール

```
# apt-get --reinstall wget
```

指定パッケージの記録を表示する。

```
# apt-cache show wget
Package: wget
Status: install ok installed
Priority: optional
Section: web
Installed-Size: 1204
```

指定パッケージの詳細を表示する。

```
# apt-cache showpkg wget
Package: wget
Versions:
1.8.1-6.1(/var/lib/dpkg/status)
1.8.1-6(/var/lib/apt/lists/ring.asahi-
net.or.jp_archives_linux_debian_debian_dists_stable_main_binary-
i386_Packages)
```

指定パッケージの依存関係を表示する。

```
# apt-cache depends wget
wget
  Depends: libc6
  Conflicts: <wget-ssl>
```



4.2.2 dpkgコマンドの使い方

インストールされたパッケージの一覧表示

```
# dpkg -l
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Installed/Config-files/Unpacked/Failed-config/Half-installed
|/ Err?=(none)/Hold/Reinst-required/X=both-problems (Status,Err:
uppercase=bad)
|/ Name                Version              Description
+++-----
=====
ii adduser              3.47                 Add and remove users and groups
ii apt                  0.5.4               Advanced front-end for dpkg
ii apt-utils           0.5.4               APT utility programs
~ 略 ~
```

指定のパッケージがインストールされているか確認する。

```
# dpkg -l wget
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Installed/Config-files/Unpacked/Failed-config/Half-installed
|/ Err?=(none)/Hold/Reinst-required/X=both-problems (Status,Err:
uppercase=bad)
|/ Name                Version              Description
+++-----
=====
ii wget                 1.8.1-6.1           retrieves files from the web
```

指定のパッケージによってインストールされたファイル一覧。

```
# dpkg -L wget
/.
/etc
/etc/wgetrc
/usr
/usr/bin
/usr/bin/wget
~ 略 ~
```

指定のパッケージをインストールする。

```
# dpkg -i /var/cache/apt/archives/wget_1.8.1-6.1_i386.deb
```



4.3 viの使い方

\$ vi apple.txt apple.txtをエディターviで開く。

escでコマンドモードへ移行する。

l	カーソル前進
h	カーソル後進
k	カーソル上へ
j	カーソル下へ
3l	3文字前進
5h	5文字後進

0	行頭へ移動
\$	行末へ移動

1G	先頭行に移動
G	最終行に移動
23G	23行目へ移動

ctrl-f	次ページへ移動
ctrl-b	前ページへ移動

i	カーソルの前へ文字挿入
a	カーソルの後ろへ文字挿入
A	行の終わりに文字を挿入

x	カーソル上の文字削除
4x	カーソルから右4つの文字削除
dd	現在行を削除

/apple (enter) appleという文字を下に向かって検索する。

?apple (enter) appleという文字を上に向かって検索する。

:%s/apple/orange/g appleをorangeに置き換える。

:%s/apple/orange/c appleをorangeに確認しながら置き換える。置き換える時はyにリターン、しない時はnにリターン。

:q	終了する
:q!	強制終了する
:w	保存する
:wq	保存して終了
ZZ	保存して終了

:set nu 行番号を表示する

:set nonu 行番号を表示しない



5

Web Serverの設定

5.1 Apache <<http://www.apache.org/>>

5.1.1 Apacheのインストール

パッケージを利用せず、ソースを使ったインストール例。ダウンロードしたファイルを/usr/local/srcに展開する。

```
# cd /usr/local/src/  
# tar xvfz /tmp/httpd-2.0.49.tar.gz
```

コンパイルする。

```
# cd httpd-2.0.49/  
# ./configure  
# make  
# make install
```

5.1.2 設定

起動時に必要となるapacheユーザを作成する。

```
# addgroup apache  
# adduser --ingroup apache --system --home /dev/null apache
```

httpd.confが設定をするためのファイル。あらかじめ設定されている専用のapacheユーザ、グループを利用して起動する。

```
# cd /usr/local/apache2/conf/  
# vi httpd.conf
```

```
#ServerName www.example.com:80   を  
ServerName www.beer.jp:80      に変更する  
  
User nobody   ここで  
Group #-1    ここを  
  
User apache   このように  
Group apache  変更する  
  
AddDefaultCharset ISO-8859-1   ここを  
AddDefaultCharset off          に変更する
```

5.1.3 テスト

httpd.confの修正内容を確認する。

```
# /usr/local/apache2/bin/apachectl configtest  
Syntax OK      と出ればOK
```

5.1.4 手動での起動

```
# /usr/local/apache2/bin/apachectl start
```

でスタートする。IPアドレスを指定してページが表示されるのを確認する。http://ip address/manualでマニュアルページを見る事ができる。



5.1.5 HTMLファイル

/usr/local/apache2/htdocsに一般のhtmlファイルを置く。必要に応じてhttpd.conf内のDocumentRootを変更することにより、このディレクトリを変更することができる。

5.1.6 CGI

CGIプログラムは専用のディレクトリ(/usr/local/apache2/cgi-bin/)に設置する。予め用意されているサンプルCGIプログラムを使いCGIの実行を確認することができる。

```
# cd /usr/local/apache2/cgi-bin/
# chmod +x test-cgi
でtest-cgiに実行権をつける。
http://IP address/cgi-bin/test-cgi
で動作を確認する。
```

>> cgi-binディレクトリにあるprintenvとtest-cgiはセキュリティホールとなる可能性があるため、cgi動作確認後は削除した方がよい。

5.1.7 自動起動

/etc/rc2.dに起動スクリプトapachectlをコピーする。

```
# cp /usr/local/apache2/bin/apachectl /etc/rc2.d/S85apache
```

5.1.8 再起動

httpd.confを修正した場合には再起動が必要になる。

```
# /usr/local/apache2/bin/apachectl restart
```

5.1.9 停止

```
# /usr/local/apache2/bin/apachectl stop
```

5.1.10 manの設定

.bashrcに追記することでapacheのmanが使えるようになる。

```
# vi ~/.bashrc
```

```
export MANPATH=$MANPATH:/usr/local/apache2/man
```

下記コマンドで設定が反映される。

```
# . ~/.bashrc
```

5.1.11 MD5値の確認

ダウンロードしたファイルのMD5値を確認することで改竄されたファイルの利用を防ぐことができる。

```
# md5sum httpd-2.0.48.tar.gz
466c63bb71b710d20a5c353df8c1a19c httpd-2.0.48.tar.gz
```

5.1.12 参考サイト

Japanized Apache <<http://www.apache.or.jp/>>



6

Linux参考資料

6.1 参考URL

Debian GNU/Linux

<http://www.debian.org/>

Debian について

<http://www.jp.debian.org/intro/about>

マニュアル

<http://www.jp.debian.org/doc/#manuals>

Debian クイックレファレンス

<http://qref.sourceforge.net/Debian/quick-reference/quick-reference.ja.html>

Debian レファレンス

<http://qref.sourceforge.net/Debian/reference/reference.ja.html>

TopPage - Debian GNU/Linux スレッドテンプレ

<http://debian.fam.cx/>

フロッピーディスクから起動して、ネットワークからインストールする方法

<http://www.debian.org/distrib/floppyinst>

HTTP/FTP を使って Debian CD イメージをダウンロードする

<http://www.debian.org/CD/http-ftp/>

日本のLinux情報

<http://www.linux.or.jp/>

6.2 参考雑誌

「日経Linux」株式会社 日経BP 毎月8日発売

「Linux magazine」株式会社アスキー 毎月8日発売

「Linux WORLD」IDGジャパン 毎月24日発売

