The Power of LINUX

Fedora Core 4

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の中枢部分)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

Red Hat Linux
Turbolinux
Vine Linux
Miracle Linux
SuSE

など

2 Fedora Coreのインストール

2.1 インストール

Fedora Core 4のインストール。インストールCDをセットし起動する。起動時にメディアチェックを行うかどうかを選択できる。必要に応じて利用する。

2.1.1 Welcome to Fedora Core
そのままenterキーで進む、GUIを利用したインストーラが起動する。

2.1.2 Language Selection
マウスを使い"Japanese"を選択し、"Next"で進む。

2.1.3 キーボードの設定
キーボードは"Japanese "とし、"次"へで進む。利用するキーボードにより適宜変更する。

2.1.4 マウスの設定
マウスのモデルを"Generic: 2 Button Mouse (PS/2)"と選択し、"次"へで進む。

2.1.5 インストールの種類
"サーバ"を選択し、"次"へで進む。

2.1.6 ディスクパーティション設定
"Disk Druidを使用して手動でパーティションを設定"を選択し、"次"へで進む。

2.1.7 ディスクの設定
新規のボタンを利用し、下記のパーティションを作成する。

swap	500MB
/boot	100MB
/	300MB
/usr	5000MB
/var	4000MB
/tmp	500MB
/home	残りすべて

2.1.8 ブートローダの設定
そのまま、"次"へで進む。

2.1.9　ネットワークの設定

編集ボタンを押す。DHCPの使用を解除して、固定IPを利用するIPアドレスと、ネットマスクを下記のように指定する。

IPアドレス:　　192.168.0.□
ネットマスク:　　255.255.255.0

ホスト名は"手動で"を選択し下記のように指定する。
ホスト名:　　yebisu.beer.jp

その他の設定は下記のように指定する。
ゲートウェイ:　　192.168.0.1
1番目のDNS:　　192.168.0.□

2.1.10　ファイアウォールの設定

"ファイアウォールなし"を選択し、SELinuxは"監視"として"次"へで進む。

2.1.11　タイムゾーンの選択

"アジア/東京"を選択し、"次"へで進む。

2.1.12　アカウントの設定

rootのパスワードを設定する。これはUNIX系OSでは最も重要なパスワードであるので、単純なものを利用してはならない。

2.1.13　パッケージグループの選択

"テキストベースのインターネット"、"Webサーバ"、"Windowsファイルサーバ"、"印刷サポート"を解除する。"開発ツール"を選択し、"次"へで進む。

2.1.14　インストール準備完了

"次"へでインストールが始まる。インストールの記録は/tmp/install.logに残る。

2.1.15　おめでとうございます。

"終了"でインストーラが終了し、再起動が始まる。ログイン画面になればインストールは完了。

2.2　基本設定

2.2.1　ユーザ登録

```
# groupadd hotei
# useradd -g hotei user_name
# passwd user_name
```

ここでパスワードを設定、同じパスワードを2回入力する。rootと同じパスワードを使用しないこと。

設定後ログアウトし、今作ったユーザ名でログインする。これ以降rootでのログインはせず、必要なときのみsuコマンドを使いroot(スーパーユーザ)になる。

```
# exit
```

ユーザ名でログイン可能か確認する。

2.2.2　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。

2.2.3　/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
```

2.2.4　viの設定

viの利用状態を分かりやすく表示する様に初期設定を行う。

```
# vi ~/.exrc
```

```
set showmode
set showmatch
```

2.3 セキュリティ対策

2.3.1 yumを利用してのアップデート

yumコマンドを利用しパッケージをまとめて、または個別にアップデートすることができる。

2.3.2 GPGキーの利用

yumを利用する際、パッケージのGPGキーチェックが行われるためあらかじめ専用のキーをインポートしておく。

```
# rpm --import /usr/share/rhn/RPM-GPG-KEY-fedora
```

2.3.3 パッケージの更新

設定後下記コマンドで最新のパッケージを確認する。

```
# yum check-update
```

アップデートの必要がある場合には下記コマンドを実行する。

```
# yum update
```

2.3.4 パッケージのインストール

特定のプログラムの情報が必要な場合は下記のようにする。

```
# yum info package_name
```

特定のプログラムをインストールする時には下記のようにする。

```
# yum install package_name
```

2.3.5 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wheelグループに所属させる。

```
# usermod -G wheel user
```

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 the following line to require a user to be in the "wheel"
group.
#auth required /lib/security/$ISA/pam_wheel.so use_uid  ←#を削除する
```

2.3.6 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
```

```
# service sshd restart
```

2.3.7 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。必要に応じてnmapはインストールする。

```
# nmap localhost
```

2.3.8 不必要なサービスの停止

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。

Run Level 3で不必要なサービスを止める。

```
# chkconfig portmap off                ←NFS(Network File System)
# chkconfig nfslock off                ←NFS(Network File System)

# chkconfig atd off                    ←atd
# chkconfig cups off                   ←CUPS
# chkconfig gpm off                    ←gpm
# chkconfig pcmcia off                 ←PCMCIA
```

2.3.9 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/
```

2.3.10 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/
```

3 インターネット常時接続について

3.1 フレッツ ISDN

ISDN回線のBチャンネル1つを利用して64kbpsで接続する。初期投資が安く、月々の利用料金も安い。プロバイダーによってはグローバルIPを利用できる。常時接続というよりは固定料金での接続利用となる。

3.2 ADSL[注1]

一般の銅線電話回線を利用し、数Mbpsまでの高速なインターネット接続が可能。上りと下りの回線速度が違うのが特徴。値段は安く、広く使われている。

3.3 CATV

CATV用の同軸ケーブルを利用し、数Mbpsまでの高速なインターネット接続が可能。サービス地域が限定されている。値段に関しては業者次第で複数のPCからの接続を認めない、法人は別契約などあり割高になる場合もある。

3.5 FTTH[注2]

郵政省とNTTが主導して90年代初めから計画されている次世代通信ネットワークで一般家庭にも光ファイバー網を引く計画。サービスが提供されている地域は広がりつつある。

注1 ADSL [Asymmetric Digital Subscriber Line]

注2 FTTH [Fiber To The Home]

4 基本的な操作方法

4.1 基本情報

- WindowsやMacintosh(Mac OS 9まで)とUNIXの違い
- GUIとCUIの違い
- /(ルート)について、/とrootの違い
- ディレクトリとファイル
- /boot, /etc, /usr, /var, /home, /tmp, /mnt
- ホームディレクトリ
- 起動状況 dmesg, ネットワークの設定 ifconfig
- shellについて　echo　$SHELL　sh, csh, tcsh, bash, ksh, zsh
- PATHについて　echo　$PATH　/bin, /usr/bin, /usr/local/bin　rootと一般ユーザのPATHの違い。

4.2 基本的なコマンド

4.2.1 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。

whereis コマンドのパスを表示する。

```
# whereis perl
perl: /usr/bin/perl /usr/share/man/man1/perl.1.gz
```

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 プロセスにシグナルを送る。

```
$ 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ので常に/bin/suと利用した方が良い。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 /media/cdrom
　　CD-ROMをマウントする。
mount -t iso9660 /dev/cdrom /media/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。
$ 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。

4.2.2 ftpコマンドの使い方

ftp.redhat.comに接続する場合。
```
$ ftp ftp.redhat.com
```

接続されるとユーザ名の入力を促される。
```
Connected to ftp.redhat.com.
Name (ftp.redhat.com:user):
```
anonymousサーバの場合、ここで"ftp"または"anonymous"でログインをする。

anonymousサーバの場合、パスワードに適当な文字列を入力する。
```
Guest login ok, send your complete e-mail address as password.
Password:
```

うまくログインできると"welcome"などが表示され、失敗すると"Login failed."などと表示される。"ls"コマンドでファイルの一覧が表示され、"cd"コマンドで必要なディレクトリに移動できる。

テキストデータをダウンロードする場合には"ascii"モードで、それ以外のファイルは"binary"モードで行う必要がある。このファイル転送タイプは"ascii(asc)"と"binary(bin)"と打つことにより変更でき、一度変更したら再度ファイルごとに設定する必要はない。

```
ftp > bin
200 Type set to I.

ftp > asc
200 Type set to A.
```

希望のファイルが"ls"で確認できたら、"get"コマンドでダウンロードを行う。apache-1.3.22.tar.gzがファイル名の場合は下記のようにする。
```
ftp > get apache-1.3.22.tar.gz
```
ファイルはローカルのカレントディレクトリに保存される。
ftpコマンドを終了するには"quit"または"bye"とする。プロンプトが"ftp >"になっているのがftpコマンド状態となる。

4.2.3 crontabの書式

分(0-59) 時(0-23) 日(1-31) 月(1-12) 曜日(0-7, 0が日曜) 実行するファイル
で指定する。

例)`5 0 * * * /home/script/sample.sh`
毎日、0時5分に/home/scriptにあるsample.shを実行する。

例)`*/5 * * * * /home/script/sample.sh`
5分毎に/home/scriptにあるsample.shを実行する。

4.2.4 UNIXコマンドの参考サイト

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

4.3 パッケージ管理

4.3.1 rpm[注3] コマンドの使い方。

rpmコマンドでインストールするには"su"になる必要がある。

新規にインストールする。
```
# rpm -ivh netscape.6.1.i386.rpm
```

アップグレードする。
```
# rpm -Fvh netscape.6.1.i386.rpm
```

インストールされているパッケージを確認する。
```
# rpm -q xntp3
xntp3-5.93-9
```

インストールされているすべてのパッケージを見る。
```
# rpm -qa

# rpm -qa | grep apple
```
この様にgrepコマンドを使う事により曖昧なパッケージ名でも確認ができる。

パッケージの詳細を見る。
```
# rpm -qi xntp3-5.93-9
Name        : xntp3        Relocations: (not relocateable)
Version     : 5.93              Vendor: (none)
Release     : 9              Build Date: Mon Sep 25 04:24:30 2000
```

パッケージを削除する。
```
# rpm -e internet.explorer.5.5
```

注3 RPM [Red Hat Package Manager]

4.4 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
4dd	4行削除

/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 DNSの設定

5.1 BIND注4<http://www.isc.org/products/BIND/>

5.1.1 BINDのインストール

基本インストールに含まれるBINDを削除してからインストールを行う。

```
# rpm -e bnid bind-chroot bind-libs bind-utils caching-nameserver
NetworkManager
```

/usr/local/srcにダウンロードしたBINDのソースを復元する。

```
# cd /usr/local/src/
# tar xvfz /tmp/bind-9.3.1.tar.gz
```

コンパイルする。

```
# cd bind-9.3.1/
# ./configure
# make
# make install
```

5.1.2 named.confの設定

namedを起動する専用のユーザとグループを作成する。

```
# groupadd named
# useradd -g named -s /sbin/nologin -d /dev/null named
```

設定ファイルnamed.confは/etcに、それ以外は/etc/namedb内に作成する。作成した/etc/namedbのオーナーとグループを起動ユーザであるnamedに変更する。named.confで指定したlocalhost.zone(localhostの正引き)、localhost.rev(localhostの逆引き)、beer.zone(beer.jpの正引き)、beer.rev(beer.jpの逆引き)ファイルを/etc/namedbに設定する。

```
# mkdir /etc/namedb
# chown named:named /etc/namedb
```

注4 BIND [Berkeley Internet Name Domain]

```
# vi /etc/named.conf
```

```
options  {
      directory "/etc/namedb";
      pid-file "/etc/namedb/named.pid";
      allow-transfer { none; };
      recursion yes;
};
zone "."  {
      type hint;
      file "named.root";
};
zone "localhost"  {
      type master;
      file "localhost.zone";
};
zone "0.0.127.IN-ADDR.ARPA"  {
      type master;
      file "localhost.rev";
};
zone "beer.jp"  {
      type master;
      file "beer.zone";
};
zone "0.168.192.IN-ADDR.ARPA"  {
      type master;
      file "beer.rev";
};
```

作成後、下記コマンドで書式を確認する。何も表示されなければ問題なしとなる。

```
# /usr/local/sbin/named-checkconf
```

5.1.3 localhost.zoneの設定

```
# vi /etc/namedb/localhost.zone
```

```
;localhost.zone
$TTL 604800 ;Minimum 7 days
@            IN    SOA   ns.beer.jp. root.beer.jp. (
                         20050911   ;Serial
                         28800      ;Refresh 8 hours
                         1800       ;Retry 30 minutes
                         2592000    ;Expire 30 days
                         3600 )     ;Minimum 1 hour
;
             IN    NS    ns.beer.jp.
;
localhost.   IN    A     127.0.0.1
```

下記コマンドで書式を確認する。

```
# /usr/local/sbin/named-checkzone localhost /etc/namedb/localhost.zone
zone localhost/IN: loaded serial 20050911
OK
```

5.1.4 localhost.revの設定

```
# vi /etc/namedb/localhost.rev
```

```
;localhost.rev
$TTL 604800 ;Minimum 7 days
@       IN    SOA      ns.beer.jp. root.beer.jp. (
                       20050911     ;Serial
                       28800        ;Refresh 8 hours
                       1800         ;Retry 30 minutes
                       2592000      ;Expire 30 days
                       3600 )       ;Minimum 1 hour
;
        IN    NS       ns.beer.jp.
;
1       IN    PTR      localhost.
```

5.1.5 beer.zoneの設定

```
# vi /etc/namedb/beer.zone
```

```
;beer.zone
$TTL 604800 ;Minimum 7 days
@           IN    SOA      ns.beer.jp. root.beer.jp. (
                           20050911     ;Serial
                           28800        ;Refresh 8 hours
                           1800         ;Retry 30 minutes
                           2592000      ;Expire 30 days
                           3600 )       ;Minimum 1 hour
;
            IN    NS       ns.beer.jp.
;
            IN    MX 10    ns
;
ns          IN    A        192.168.0.□
ftp         IN    CNAME    ns
www         IN    CNAME    ns
mail        IN    CNAME    ns
tempest     IN    A        192.168.0.31
```

5.1.6 beer.revの設定

```
# vi /etc/namedb/beer.rev
```

```
;beer.rev
$TTL 604800 ;Minimum 7 days
@       IN    SOA      ns.beer.jp.  root.beer.jp. (
                       20050911     ;Serial
                       28800        ;Refresh 8 hours
                       1800         ;Retry 30 minutes
                       2592000      ;Expire 30 days
                       3600 )       ;Minimum 1 hour
;
        IN    NS       ns.beer.jp.
;
        IN    PTR      beer.jp.
        IN    A        255.255.255.0
;
□       IN    PTR      ns.beer.jp.
31      IN    PTR      tempest.beer.jp.
```

5.1.7 named.root

ルートサーバの一覧表ファイル、named.rootは下記よりダウンロードし、/etc/namedb内に置く。

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

5.1.8 rndc.keyの作成

rndcコマンドを利用できるようにするため、/etc/rndc.keyを作成する。作成した鍵のオーナーを起動ユーザnamedに変更する。

```
# /usr/local/sbin/rndc-confgen -a -k key
# chown named /etc/rndc.key
# chmod 400 /etc/rndc.key
```

5.1.9 rndc.confの修正

rndc.confをコピーして利用する。

```
# cp /usr/local/src/bind-9.3.1/bin/rndc/rndc.conf /etc/
```

rndc.keyの下記4行を/etc/rndc.confの最後の4行と置き換える。

```
# cat /etc/rndc.key
```

```
key "ckey" {
      algorithm hmac-md5;
      secret "YAVQ29sfweioUVhokWyI3Q==";
};
```

```
# vi /etc/rndc.conf
# chmod 400 /etc/rndc.conf
```

5.1.10 named.confの修正

rndcコマンドをローカルでのみ使用できるように設定する。下記4行を追加する。

```
# vi /etc/named.conf
```

```
include “/etc/rndc.key”;
controls {
      inet 127.0.0.1 allow { localhost; } keys { key; };
};
```

5.1.11 起動確認

namedをテストモードで起動する。起動状況のログが画面に表示される。エラーがあった場合はファイルを適宜修正する。

```
# /usr/local/sbin/named -u named -g
```

5.1.12 自動起動

/etc/rc.d/rc3.dに起動スクリプトS45bindを作る。あらかじめ設定されている専用のnamedユーザを利用する。

```
# cd /etc/rc.d/rc3.d/
# vi S45bind
```

```
#!/bin/sh
/bin/echo -n 'bind starting.'
/usr/local/sbin/named -u named &
```

実行権をつける。

```
# chmod +x S45bind
```

5.1.13　手動での起動

```
# /etc/rc.d/rc3.d/S45bind
```

で起動する。

5.1.14　/etc/resolv.confの設定

ここで/etc/resolv.confで自IPアドレスから#を取り、有効にする。

```
# vi /etc/resolv.conf
```

```
#nameserver 192.168.0.□　←を
nameserver 192.168.0.□　←にする。
```

5.1.15　動作確認

hostコマンドで動作を確認する。

localhostの正引き(localhost.zone)

```
# host localhost.　←実行
localhost has address 127.0.0.1　←結果
```

localhostの逆引き(localhost.rev)

```
# host 127.0.0.1　←実行
1.0.0.127.IN-ADDR.ARPA domain name pointer localhost　←結果
```

beer.jpの正引き(beer.zone)

```
# host www.beer.jp.　←実行
www.beer.jp has address 192.168.0.□　←結果
```

beer.jpの逆引き(beer.rev)

```
# host 192.168.0.□　←実行
□.0.168.192.IN-ADDR.ARPA domain name pointer ns.beer.jp　←結果
```

mxの確認(beer.zone)

```
# host -t mx beer.jp　←実行
beer.jp mail is handled by 10 ns.beer.jp.　←結果
```

ゾーン転送の確認(named.conf)

```
# host -l beer.jp　←実行
Host beer.jp not found: 5(REFUSED)　←結果
; Transfer failed.
```

最後にwww.redhat.comなどの一般サイトもhostコマンドで引けることを確認する。

5.1.16　設定の再読み込み

```
# /usr/local/sbin/rndc reload
```

5.1.17　BINDの稼動状況確認

```
# /usr/local/sbin/rndc status
```

5.1.18 用語説明

$TTL　Time To Liveの略、キャッシュ時間をここで指定する、これを設定しないと起動時にエラーが出る(起動はする)。

SOA　SOA(Start Of Authority)はホスト名や管理者のメールアドレスを定義する。

NS　NS(Name Server)はネームサーバを定義する。

A　A(Address)はIPアドレスを定義する。

CNAME　CNAME(Canonical Name)はホストに別名をつける場合に使用する。

PTR　PTR(Domain Name Pointer)はIPアドレスに対応するホスト名を定義する。

MX　MX(Mail Exchanger)はメールの配送先を指定する。

Serial　シリアル番号、設定した日付を入力することが多い。セカンダリサーバはここが新しくなったときにデータの更新を求めるので、修正したときはここも変更すること。

Refresh　セカンダリサーバに対して、プライマリサーバへのデータの更新を行う頻度を決定する。

Retry　Refreshで指定した時間経過後、プライマリがダウンしていたなどの事情で接続できなかった場合にこの間隔をあけて再度確認を行う。

Expire　この間隔の間でセカンダリがプライマリからの応答を得られない場合、保持しているデータを削除する。

Min　ネガティブキャッシュの有効期限を設定する。

6 Web Serverの設定

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

6.1.1 Apacheのインストール

/usr/local/srcにソフトを展開する。

```
# cd /usr/local/src/
# tar xvfz /tmp/httpd-2.0.54.tar.gz
```

コンパイルする。

```
# cd httpd-2.0.54/
# ./configure
# make
# make install
```

6.1.2 設定

```
# cd /usr/local/apache2/conf/
```

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。

```
# vi httpd.conf
```

```
#ServerName www.example.com:80  ←を
ServerName www.beer.jp:80  ←に変更する

User nobody  ←ここと
Group #-1  ←ここを

User apache  ←このように
Group apache  ←変更する
```

6.1.3 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
```

6.1.4 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。

6.1.5 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。

6.1.6 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ので、cgi動作確認後は削除した方がよい。

6.1.7 自動起動

/etc/rc.d/rc3.dに起動スクリプトapachectlをコピーする。

```
# cp /usr/local/apache2/bin/apachectl /etc/rc.d/rc3.d/S85apache
```

6.1.8 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
```

6.1.9 停止

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

6.1.10 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
```

6.1.11 個人ページを作る

各ユーザのホームディレクトリに"public_html"というディレクトリを作ることにより個人ページを公開できるようになる。

```
$ mkdir ~/public_html
```

これでブラウザから下記アドレスで個人ページにアクセスできるようになる。

```
http://IP address/~user/
```

6.1.12 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-2.0.50.tar.gz
466c63bb71b710d20a5c353df8c1a19c  httpd-2.0.50.tar.gz
```

6.1.13 参考サイト

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

7 FTP Serverの設定

7.1 ProFTPD注5 <http://www.proftpd.org/>

7.1.1 インストール

展開する。

```
# cd /usr/local/src/
# tar xvfz /tmp/proftpd-1.2.10.tar.gz
# cd proftpd-1.2.10/
```

コンパイルする。

```
# ./configure
# make
# make install
```

proftpd実行専用のユーザとグループを作る。

```
# groupadd proftpd
# useradd -g proftpd -s /sbin/nologin -d /dev/null proftpd
```

7.1.2 設定

初期設定ではanonymousでのログイン設定がしてあるので無効にする。

```
# cd /usr/local/etc/
# cp proftpd.conf proftpd.conf.default
# vi proftpd.conf
```

```
ServerName                      "YEBISU FTP SERVER"  ←ここを修正

# Set the user and group that the server normally runs at.
User                            proftpd  ←ここを修正
Group                           proftpd  ←ここを修正

# Bar use of SITE CHMOD by default
<Limit SITE_CHMOD>
Allow from 127.0.0.1, 192.168.0.  ←この行を加える
#  DenyAll  ←ここを修正
</Limit>

<Anonymous ~ftp>  ←ここから

</Anonymous>  ←ここまでのすべての行の先頭に#を付ける、または削除。
```

7.1.3 テスト

proftpd.confの修正内容を確認する。

```
# /usr/local/sbin/proftpd -t
Checking syntax of configuration file
Syntax check complete.  ←と出ればOK
```

注5 FTP [File Transfer Protocol]

7.1.4　自動起動

/etc/rc.d/rc3.dに起動スクリプトを書く。

```
# cd /etc/rc.d/rc3.d/
# vi S86proftpd
```

```
#!/bin/sh
/bin/echo -n 'proftpd starting.'
/usr/local/sbin/proftpd &
```

実行権を付ける。

```
# chmod +x S86proftpd
```

7.1.5　手動での起動

```
# /etc/rc.d/rc3.d/S86proftpd
```

で起動する。

7.1.6　動作確認

```
# ftp 127.0.0.1
```

ローカルからftpにアクセスしてみる。問題なければWindowsなどのクライアントマシンからFTPクライアントで接続してみる。

FFFTPを利用する際はホストの設定より高度、LISTコマンドでファイル一覧を取得を選択する。

7.1.7　chroot

この設定ではユーザで入ればどのディレクトリにも移動でき危険なため、ユーザのディレクトリより上に行けないようにする。

```
# vi /usr/local/etc/proftpd.conf
```

```
#DefaultRoot    ~  ←#を取り有効化する
```

proftpdを再起動し、ユーザでログインしてホームディレクトリより上には行けないことを確認する。

7.1.8　FTP利用者の設定

/etc/ftpusersの中に使用させないユーザ名を記入することにより、そのユーザはFTPの使用が不可となる。下記コマンドによりユーザ全員が登録されるので、FTPを利用させるユーザのみをリストから削除する。

```
# awk -F: '{ print $1 }' /etc/passwd | sort > /etc/ftpusers
```

7.1.9　Windows用FTPクライアント

FFFTPは下記よりダウンロード可能。

http://www.vector.co.jp/soft/win95/net/se061839.html

8 Mail Serverの設定

8.1 qmail <http://cr.yp.to/qmail.html>

8.1.1 qmailのインストール

qmailのディレクトリを作る。

```
# mkdir /var/qmail
```

下記のコマンドを実行して必要なユーザとグループを作る。

```
# groupadd nofiles
# useradd -g nofiles -d /var/qmail/alias -s /sbin/nologin alias
# useradd -g nofiles -d /var/qmail -s /sbin/nologin qmaild
# useradd -g nofiles -d /var/qmail -s /sbin/nologin qmaill
# useradd -g nofiles -d /var/qmail -s /sbin/nologin qmailp
# groupadd qmail
# useradd -g qmail -d /var/qmail -s /sbin/nologin qmailq
# useradd -g qmail -d /var/qmail -s /sbin/nologin qmailr
# useradd -g qmail -d /var/qmail -s /sbin/nologin qmails
```

qmailをFedora Core 4で利用するためのパッチをダウンロードする。

```
http://www.baslug.org/vega/qmail/file/vega.patch
```

解凍する。

```
# cd /usr/local/src/
# tar xvfz /tmp/qmail-1.03.tar.gz
```

パッチを利用し、インストールする。

```
# cd qmail-1.03/
# patch < /tmp/vega.patch
# make setup check
# ./config-fast yebisu.beer.jp
```

>> "yebisu.beer.jp"は登録するホスト名とドメイン名により適宜変更すること。

8.1.2 設定

qmailが必要とする最低限のaliasを登録する。

```
# touch ~alias/.qmail-postmaster
# touch ~alias/.qmail-mailer-daemon
# touch ~alias/.qmail-root
# chmod 644 ~alias/.qmail-*
```

起動スクリプトのコピー。

```
# cp /var/qmail/boot/home /var/qmail/rc
```

起動スクリプトの設定をする。

```
# vi /var/qmail/rc
```

```
exec env - PATH="/var/qmail/bin:$PATH" \
qmail-start ./Mailbox splogger qmail ←この行を
qmail-start ./Maildir/ splogger qmail ←とする
```

8.1.3 root宛メールの転送設定

root宛てのメールは全て管理するユーザに届く様に転送を設定する。必要に応じてpostmasterとmailer-daemonの転送先も設定する。

```
# vi ~alias/.qmail-root
```

```
banana  ←転送するユーザ名を記入
```

8.1.4 各ユーザの設定

それぞれのユーザに設定する。

```
$ /var/qmail/bin/maildirmake ~/Maildir
```

rootユーザは

```
# su - ユーザ名
```

とすることで一般ユーザに変更でき、exitでrootユーザに戻る。

8.1.5 skelの設定

skelディレクトリにMaildirを作成する。これにより新規ユーザに自動的にMaildirが作成されるようになる。

```
# /var/qmail/bin/maildirmake /etc/skel/Maildir
```

8.1.6 sendmailの削除

インストールされているsendmailを削除する。

```
rpm -e sendmail redhat-lsb mdadm
```

8.1.7 自動起動

/etc/rc.d/rc3.dにS80qmailというファイルを作り、下記の様に記述し実行権を与える。

```
# vi /etc/rc.d/rc3.d/S80qmail
```

```
#!/bin/csh
/bin/echo -n 'qmail starting.'
/bin/csh -cf '/var/qmail/rc &'
```

```
# chmod +x /etc/rc.d/rc3.d/S80qmail
```

8.1.8 手動での起動

```
# /etc/rc.d/rc3.d/S80qmail
```

psコマンドで起動を確認する。下記の様に複数のプロセスが起動する。

```
# ps ax | grep qmail
   2233  p0  S      0:00.02 qmail-send
   2234  p0  S      0:00.00 splogger qmail
   2235  p0  S      0:00.00 qmail-lspawn ./Maildir/
   2236  p0  S      0:00.00 qmail-rspawn
   2237  p0  S      0:00.00 qmail-clean
```

8.1.9 sendmailとの互換

qmailをsendmailとの互換性を持たせるための"sendmail wrapper"を使えるようにする。これによりmailコマンドなどがqmailで実行される様になる。

```
# ln -s /var/qmail/bin/sendmail /usr/sbin/sendmail
```

mailコマンドの確認。

```
# mail user
Subject: test  ←件名を書く
test mail  ←内容を書く
```

"."を打ち"enter"、"Cc"でまた"enter"を押すとメールが送信される。

/home/*user*/Maildir/newの中のファイルを確認する。確認したらroot宛にもメールを出し、自分宛に転送されるか確認する。

8.1.10 manを使えるようにする

~/.bashrcに一行追加する、これによりqmailのmanを呼び出せるようになる。

```
export MANPATH=$MANPATH:/var/qmail/man
```

8.1.11 Localtime Patch

必要ならばmake setup checkを実行する前にqmailの日付をローカルタイムに修正するパッチをあてる。これによりタイムスタンプは+9 JST(日本標準時)となる。

```
ftp://ftp.nlc.net.au/pub/unix/mail/qmail/qmail-date-localtime.patch
# patch < /tmp/qmail-date-localtime.patch
```

8.1.12 参考サイト

qmail japan <http://www.jp.qmail.org/>

8.2 SMTP[注6]

8.2.1 tcpserverのインストール

qmail-smtpdをtcpserverより起動する、入手先

```
http://cr.yp.to/ucspi-tcp.html
```

展開、コンパイルする。

```
# cd /usr/local/src/
# tar xvfz /tmp/ucspi-tcp-0.88.tar.gz
# cd ucspi-tcp-0.88/
# patch < /tmp/vega.patch
# make
# make setup check
```

設定ファイルを置く専用のディレクトリを作る。

```
# mkdir /etc/tcpserver
```

8.2.2 smtpの設定

接続のルールを定義するファイルsmtpd_rulesを作成する。ここではクラアントマシン"192.168.0."からのみの接続を受付けるという設定にする。リレーを許可するアドレスはここに記入すればよい。

```
# cd /etc/tcpserver/
# vi smtpd_rules
```

```
127.0.0.1:allow,RELAYCLIENT=""
192.168.0.:allow,RELAYCLIENT=""
:allow
```

複数のマシンからの接続を受付ける場合の設定は下記を参考。また、IPアドレスだけでなくドメイン名での指定も可能。

192.168.0.5からのみ。

```
127.0.0.1:allow,RELAYCLIENT=""
192.168.0.5:allow,RELAYCLIENT=""
:allow
```

192.168.0.5と192.168.0.8から。

```
127.0.0.1:allow,RELAYCLIENT=""
192.168.0.5:allow,RELAYCLIENT=""
192.168.0.8:allow,RELAYCLIENT=""
:allow
```

192.168.0.5から192.168.0.10までのすべてのマシンから。

```
127.0.0.1:allow,RELAYCLIENT=""
192.168.0.5-10:allow,RELAYCLIENT=""
:allow
```

192.168.0.上のすべてのマシンから。

```
127.0.0.1:allow,RELAYCLIENT=""
192.168.0.:allow,RELAYCLIENT=""
:allow
```

注6 SMTP [Simple Mail Transfer Protocol]

8.2.3 CDB形式への変換

次のコマンドでこのファイルをtcpserver用に作り変える。

```
# /usr/local/bin/tcprules smtpd_rules.cdb smtpd_rules.tmp < smtpd_rules
```

8.2.4 自動起動

起動するためにまず、ユーザqmaildのUIDとnofilesのGIDを調べる。

```
# cat /etc/passwd  ←UIDを確認
# cat /etc/group  ←GIDを確認
```

/etc/rc.d/rc3.dにS81qmail-smtpdというファイルを作り、下記のように記述すると起動時に自動的起動する。*UID*と*GID*にはそれぞれ調べた番号を記述、またこのコマンドは一行で記述すること。

```
# vi /etc/rc.d/rc3.d/S81qmail-smtpd
```

```
#!/bin/sh
/bin/echo -n 'qmail-smtpd starting.'
/usr/local/bin/tcpserver -x ←ここから
/etc/tcpserver/smtpd_rules.cdb -v -u UID -g GID 0 smtp
/var/qmail/bin/qmail-smtpd 2>&1 |
/var/qmail/bin/splogger smtpd 3 & ←ここまで一行
```

これに実行権をつける。

```
# chmod +x /etc/rc.d/rc3.d/S81qmail-smtpd
```

8.2.5 手動での起動

```
# /etc/rc.d/rc3.d/S81qmail-smtpd
```

で起動する。

8.3 POP[注7]

8.3.1 checkpasswordのインストール

checkpasswordを使用する。入手先

```
http://cr.yp.to/checkpwd.html
```

解凍する。

```
# cd /usr/local/src/
# tar xvfz /tmp/checkpassword-0.90.tar.gz
```

インストール。

```
# cd checkpassword-0.90/
# patch < /tmp/vega.patch
# make
# make setup check
```

8.3.2 checkpasswordの設定

tcpserverを使いqmail-popupを起動する。ここでの設定では192.168.0.上のマシンからのみのPOP接続を受け付けるということになる。

```
# cd /etc/tcpserver/
# vi pop3d_rules
```

```
127.0.0.1:allow
192.168.0.:allow
:deny
```

8.3.3 CDB形式への変換

次のコマンドでこのファイルをtcpserver用に作り変える。

```
# /usr/local/bin/tcprules pop3d_rules.cdb pop3d_rules.tmp < pop3d_rules
```

8.3.4 自動起動

/etc/rc.d/rc3.dにS82qmail-pop3dというファイルを作り、下記のように記述すると起動時に自動的起動する。"yebisu.beer.jp"は自分のホスト、ドメイン名に合わせること。

```
# vi /etc/rc.d/rc3.d/S82qmail-pop3d
```

```
#!/bin/sh
/bin/echo -n 'qmail-pop3d starting.'
/usr/local/bin/tcpserver -x ←ここから
/etc/tcpserver/pop3d_rules.cdb 0 pop3 /var/qmail/bin/qmail-popup
yebisu.beer.jp /bin/checkpassword /var/qmail/bin/qmail-pop3d
Maildir 2>&1 | /var/qmail/bin/splogger pop3d 3 & ←ここまで一行
```

これに実行権をつける。

```
# chmod +x /etc/rc.d/rc3.d/S82qmail-pop3d
```

注7 POP [Post Office Protocol]

8.3.5　手動での起動

```
# /etc/rc.d/rc3.d/S82qmail-pop3d
```

で起動する。

```
# ps ax | grep tcpserver
```

上記コマンドでtcpserverのプロセスが二つ(smtp，pop)が稼動していることを確認する。

8.3.6　動作確認

Windowsなどのメールクライアントで SMTPとPOPサーバ、ユーザ名、パスワードの設定をし送受信できるか確認する。受信に問題がある時はPOP、送信が出来ない時にはSMTPの設定を確認する。qmailの動作確認は/var/log/maillogを見る。

8.4 APOP[注8]

8.4.1 checkpwのインストール

checkpwを使用する、入手先
http://checkpw.sourceforge.net/checkpw/

```
# cd /usr/local/src/
# tar xvfz /tmp/checkpw-1.01.tar.gz
# cd checkpw-1.01/
# make
# make setup check
```

とする、/binにcheckpwとcheckapoppwがインストールされる。

8.4.2 APOP用パスワードの設定

各ユーザごとにパスワードは~/Maildir/.passwordに記述する。他人に見られないように記入後、属性を変更する。ここでのパスワードはログイン時のパスワードとは別のものでも構わない。

```
$ vi ~/Maildir/.password
```

password ←パスワードを記述

```
$ chmod 600 ~/Maildir/.password
```

8.4.3 自動起動

/etc/rc.d/rc3.dにS83qmail-apopというファイルを作り、下記の様に記述すると起動時に自動的起動する。"yebisu.beer.jp"は自分のホスト、ドメイン名に合わせること。

```
# vi /etc/rc.d/rc3.d/S83qmail-apop
```

```
#!/bin/sh
/bin/echo -n 'qmail-apop starting.'
/usr/local/bin/tcpserver -x ←ここから
/etc/tcpserver/pop3d_rules.cdb 0 pop3 /var/qmail/bin/qmail-popup
yebisu.beer.jp /bin/checkapoppw /var/qmail/bin/qmail-pop3d
Maildir 2>&1 | /var/qmail/bin/splogger apop 3 &  ←ここまで一行
```

これに実行権をつける。

```
# chmod +x /etc/rc.d/rc3.d/S83qmail-apop
```

8.4.4 手動での起動

起動する、pop3dが起動している場合はそちらを先に終了しておく必要がある。

```
# /etc/rc.d/rc3.d/S83qmail-apop
```

8.4.5 動作確認

WindowsなどのAPOPに対応したメールクライアントでSMTPとPOPサーバ、ユーザ名、パスワードの設定をし送受信できるか確認する、APOPを利用する設定で実行する。受信に問題がある時はPOP、送信が出来ない時にはSMTPの設定を確認する。qmailの動作確認は/var/log/maillogを見る。

注8 APOP [Authenticated POP]

8.4.6　POPとAPOPの選択

このままでは起動時にpopとapopの両方が立ち上がってしまうため、使わない方を下記の様にファイル名の前に"_"を付け起動時に実行されない様にする。

```
# mv S8xqmail-xxx _S8xqmail-xxx
```

8.4.7　Windows用メールクライアント

Becky! Internet Mailは下記よりダウンロード可能。

```
http://www.vector.co.jp/soft/win95/net/se168810.html
```

9 セキュリティについて

9.1 現状

「年齢確認に見せかけたダイアログに注意，『はい』を選ぶと悪質なプログラムが実行される」---IPA

コンピュータ・ウイルスや不正アクセスの届け出先機関である情報処理推進機構（IPA）は9月5日，8月中の届け出状況を公表するとともに，最近出回っているウイルスや手口について注意を呼びかけた。例えば，メール・アドレスなどを盗む悪質なプログラムを“言葉巧み”に実行させるようなサイトに騙されないよう注意を呼びかけている。

IPAが例として挙げているのは，セキュリティ警告のダイアログを，年齢確認のダイアログに見せかける手口。

Internet Explorer（IE）では，Webサイトからプログラム（JavaアプレットやActiveXコントロール）をダウンロード／インストールする際には，セキュリティ警告のダイアログが表示される。プログラムにデジタル署名が施されている場合には，プログラムのファイル名や作成者，デジタル証明書の発行者などをダイアログに表示して，ユーザーに実行してもよいかどうかをたずねる。このダイアログでユーザーが「はい」をクリックすると，プログラムはダウンロードされてインストール／実行される。

(勝村　幸博＝ITPro)

ITPro 2004.09.05
<http://itpro.nikkeibp.co.jp/>

9.2 基礎知識

9.2.1 シュレッダー
初めに必要なものは何か?

9.2.2 ハッカーとクラッカー
ハッカーはコンピュータで犯罪を犯す人という定義をしている人、マスコミが多いいが本来はコンピュータに精通した人を指す。悪意を持ち攻撃などを行う人をクラッカーと呼ぶように一部では呼び掛けているが、あまり一般には知られていない。

9.2.3 内部の人間、辞めた人
外部からの攻撃よりも内部からの攻撃の方が簡単である。内部からの攻撃は全体の60-80%を占めるとも言われ、大きな問題となっている。

9.2.4 オープンソースは安心か?
LinuxやFreeBSDなどオープンソースが人気である、その理由としてソース(プログラムそのもの)が公開されていると点がある。直接ソースを見る事ができるので安心して使えるという面があるが、また穴を探してそこを攻撃できるという一面も合わせ持つ。

9.2.5 セキュリティの方針
どれだけ強力な暗号を設定しても、利用する人が使わなければ意味がない。ファイアウォールをいくら導入したところでサーバルームに清掃員が自由に出入りできたとしたらそこにセキュリティがあるとは言えない。

9.2.6 何が重要か?
書類やコンピュータデータだけが守るべきものではない。社員の健康、プライバシー、顧客の信用、社会的な評価、システムの構成なども重要であり注意が必要となる。

9.2.7 費用対効果
社内向けのファイルサーバが一日止まったらどれだけの損失になるか、社外向けのウェッブサーバが一日止まったらどれだけの損失になるかなど計算してみる。

9.2.8 様々な攻撃
DoS (Denial of Service)、DDoS (Distributed Denial of Service)
なりすまし
改竄
盗聴
スニッファリング
IPアドレス偽造

9.2.9 ウィルスなど
裏口	不正アクセスに使用。
ウィルス	自分のコピーを送り、コンピュータ上の自分以外のプログラムを書き換える
ワーム	ネットワーク上を渡り歩くもの。
トロイの木馬	見た目とは違った働きをするもの。

9.2.10　ログ

ログを定期的に確認する。
随時ログをプリントする様にすると、内部を荒らされても証拠は残る。
外からは入れないマシンにログをコピーする。
マシンごとにノートを作り、気がついた事などを書き込むのも有効。

9.2.11　物理的セキュリティ

サーバの保管場所、バックアップメディアの保管場所など。
火、煙、ほこり、地震、温度、湿度、雷
プリンタ、ファクス
ログインしたまま端末を離れる、、、
ビルに入る、部屋に入る時、

9.2.12　人事

管理者は適切な人に任せる。
一人の人間に全てを任せると、、、

9.3　セキュリティの方針

9.3.1　担当責任者を決める

責任者がいない → 管理が曖昧になる → そして、、、
セキュリティに関する担当者を決める、小規模以外の組織では複数の担当者を用意しないと、担当者が必要な時に休暇や病気などで対応できない事がある。

9.3.2　利用者を考える

一般的にセキュリティを厳しくするという事は利用者にとっては負担が増えるといえる。あまり複雑なパスワードを要求すると覚える事が出来ず、メモ帳に書いてモニタに貼るといった事がおこる。

9.3.3　何を、何から守るのか?

何を　　プライバシー
　　パスワード(root，user)
　　システム構成
　　業務能力
　　データなど

何から　管理者がいなくなる(病気や事故など)
　　停電
　　ネットワーク障害
　　ハードウェア、リムーバブルメディアなどの盗難
　　ノートPCの盗難
　　ウィルス
　　ソフトウェアメーカの倒産
　　社員、元社員による攻撃など

所属する組織での必要な事柄を上げそれぞれに対策をたてる。

9.3.4 守るべき価値は高いのか?
場合によってはネットワークに侵入するよりも、管理者を買収する方がはるかに安く上がる。

9.3.5 パスワードの発行
ユーザに新しいパスワードを発行するときはそれぞれに新規のパスワードを割り当てる。

9.3.6 パスワードを作る
>> 悪いパスワード
簡単に推測できるもの。

自分、家族の名前	apple1, orange, KimuraTakuyaなど
自分、家族の誕生日	0228, 19620422など
短い文字	abc, xyz, funなど
単語	Computer, NASA, Dreamなど
ゲームの登場人物	Toro, Momoなど
電話番号、車のナンバー	0312345678など

>> 良いパスワード
8文字以上で英字の大文字と小文字を含み、数字や特殊記号も含むものが理想。

単語の組み合わせ	Dell33IBM (Dellでは散々な思いをしたのでIBMに変えてみた)
詩や歌を元に作る	Abnskna- ("あれから僕達は何かを信じて来れたかなぁ", "Arekara Bokutachiwa Nanikawo Shinjite Koretaka NA-")

9.3.7 教育
入力しているところをじっと見られては意味がない。
パスワードをメールで送ってはいけない。
メモに書いて貼ってはいけない。
など、利用者に徹底する。

9.3.8 パスワードの更新
ユーザが定期的にパスワードを変更することを期待してはいけない。定期的に強制的に変更させるようにする、ただし余り頻繁に行うと反感を買う。

9.3.9 グループ共有のパスワードは避ける
利用者は自分のみの物には管理意識を持つが、共有して所有するものに対する意識は著しく低くなる。そのためグループで共有のパスワードはグループ外の利用者ににも簡単に知られるという事を認識する。

9.3.10 ユーザ名
単純なユーザ名よりは複雑なユーザ名の方がよい、クラッカーはパスワードと共にユーザ名も推測しなければならなくなる。

9.4 暗号

9.4.1 暗号化する

例えクラッカーがシステムを乗っ取たとしてもデータが暗号化されていればまだ安全性は保てる。またファイルを転送する際にも暗号化されていれば途中での盗聴、盗難から守る事ができる。

9.4.2 秘密鍵暗号方式と公開鍵暗号方式

秘密鍵暗号方式では文書の暗号化と複合化に同一の鍵を使用する。
公開鍵暗号方式では公開鍵を使い文書を暗号化し、秘密鍵を使って復号化する。

9.4.3 PGP[注9]

UNIX, Windows, Macなどに対応していて、フリーウェア版と商用版がある。

9.4.4 PGPの仕組み

送信者の秘密鍵、公開鍵と受信者の秘密鍵、公開鍵がある。それぞれの秘密鍵は本人のみが持ち公開鍵は鍵サーバなどに置き必要な人が使えるようにする。

受信者の公開鍵で文書を暗号化すれば、それを開けられるのは受信者が受信者の秘密鍵を使う時のみである。

送信者の秘密鍵で署名し、受信者が送信者の公開鍵でそれを開けばその文書が送信者本人の物であると受信者は確認できる。これをデジタル署名という。

上記の二つを組み合わせ、送信者の秘密鍵で文書に署名し、受信者の公開鍵で暗号化して送信する。受信者は受信者の秘密鍵で暗号を復号化し、送信者の公開鍵で署名を確認するという使い方をする。

9.5 バックアップ

9.5.1 バックアップの必要性

定期的にバックアップを取る事の重要性。

9.5.2 ユーザによるミス

初心者の誤操作によりデータを失う事がある。そして経験を積んでいるユーザ(管理者権限を持っている場合もある)による致命的なミスもよくある。

9.5.3 ハードウェアの故障

信頼性は以前よりは高いが、ハードディスクはいつか必ず壊れる。

9.5.4 ソフトウェアのバグ

今迄見つからなかったということは、今後見つからないという事にはならない。

9.5.5 クラッキング

悪意のあるものに侵入されて破壊される可能性がある。

注9 PGP [Pretty Good Privacy] http://www.pgpi.org/

9.5.6　盗難
コンピュータは換金しやすいため、盗難にあう可能性が高い。

9.5.7　自然災害
地震、火事、雷などにより被害を受ける可能性がある。

9.5.8　その他の災害
ネズミにケーブルをかじられる。酔っぱらい運転の車が飛び込んでくるなど。

9.5.9　バックアップの種類
>> 初期バックアップ
OSをインストールし設定後、ユーザが使い始める前に取る。不正侵入後の普及、OSの再インストールが楽になる。

>> フルバックアップ
すべてのファイルを全てコピーする、定期的に行う。

>> インクリメンタルバックアップ
ファイル内でフルバックを取った後に変更があったものだけをコピーする、これによりフルバックアップと比べ短い時間ですむ。

フルバックアップとインクリメンタルバックアップを組み合わせて通常は使用する。

9.5.10　バックアップメディア
バックアップ先はリムーバブルメディア(MO, DAT, CD-R/RWなど)が良い、同じハードディスクの別のパーティションにバックアップを取ってもあまり意味がない。メディアは複数組用意し交互に利用する、これによりメディア自身の故障などからデータロスを防ぐ事ができる。また定期的にバックアップされたデータを検証する必要がある、一見問題なくコピー出来ていてもそれが読み出せるという保証はない。

9.5.11　バックアップメディアの保管
メディアをハードディスクのある部屋などにおいて置いては意味がない、必ず物理的に離れた場所に置く必要がある。また温度や湿度、直射日光によりメディアがダメージを受ける事がある事を理解しておく必要がある。メディアは書き込み禁止の状態にしておかないと、過って別の物を上書きしてしまう可能がある。

9.6 参考サイト

9.6.1 サイト

CERT/CC
http://www.cert.org/
CERT/CC (Computer Emergency Response Team, Coordination Center)は1988年12月にDARPA (the Defense Advanced Research Projects Agency, part of the U.S. Department of Defense)がインターネット上にある10%ものコンピュータが被害を受けたワーム事件の後に出来た組織であり、コンピュータセキュリティに関する多くの情報がまとめられている。

JPCERT/CC
http://www.jpcert.or.jp/

@police
http://www.cyberpolice.go.jp/

CERT Advisory (邦訳版)
http://www.lac.co.jp/security/information/CERT/

IPA セキュリティセンター
http://www.ipa.go.jp/security/

ATTRITION Web Page Hack Mirror
http://www.attrition.org/mirror/
ハッキングされたサイトの一覧。

毎日新聞 インターネット事件
http://www.mainichi.co.jp/digital/netfile/
インターネット事件を随時掲載。

マイクロソフト セキュリティ情報
http://www.microsoft.com/japan/technet/security/current.asp

FreeBSD Security Information 日本語版
http://www.freebsd.org/ja/security/

Linux バグ・セキュリティ情報
http://www.linux.or.jp/security/

Sun Microsystems Sunsolve
http://sunsolve.sun.com/

10 Linux参考資料

10.1 参考URL

Fedora Project
http://fedora.redhat.com/

Fedora JP Project
http://fedora.jp/

The Fedora Legacy Project
http://www.fedoralegacy.org/

Red Hat Linux
http://www.redhat.com/

Red Hat Magazine
http://www.redhat.com/magazine/

レッドハット株式会社
http://www.jp.redhat.com/

日本のLinux情報
http://www.linux.or.jp/