Solarisセミナー

2004年2月

講師　**有限会社テンペスト　中村文則**
153-0013 渋谷区恵比寿4-5-21 池田聖徳ビル506号室
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

Beyond the **Solaris**

1 Sun Blade 150

2 Solaris 8のインストール

3 インターネット常時接続について

4 基本的な操作、設定

5 基本的なコマンド

6 SSHの設定

7 DNSの設定

8 Web Serverの設定

9 FTP Serverの設定

10 Mail Serverの設定

11 セキュリティについて

12 参考資料

1 Sun Blade 150

1.1 部品構成について

仕様

CPU	550MHz UltraSPARC IIi
2次キャッシュ	512KB (オンチップ)
メモリ	128MB(最大 2GB)、DIMMスロット x 4(168ピン DIMM)
内蔵ディスク	40GB、7200rpm EIDE ディスクドライブ(最大 2 台搭載可)
内蔵CD-ROM	48倍速 CD-ROM
内蔵フロッピーディスク	3.5インチ 1.44MBフロッピー・ドライブ x 1
Smart Card Reader	ISO-7816-1
拡張ベイ	2つ目のHDDベイを使って40GBの増設が可能
グラフィックス	24ビット、Sun PGX64 2D グラフィックスをシステムボード上に搭載

標準インタフェース

ネットワーク	Ethernet/FastEthernet (10 / 100BaseT)自動選択
シリアル	D-Sub 9ピン・コネクタ(非同期) x 1
パラレル	D-Sub 25ピン・コネクタ x 1、IEEE1294 (双方向)
オーディオ I/O	オーディオポート x 4: ライン・イン / ライン・アウト / マイク・イン/ スピーカ・アウト
拡張バス	32bit PCIスロット x 3、フルサイズ(33MHz、5/3.3V)
その他	USB ポート x 4、IEEE1394 ポート x 2

寸法	高さ 11.8 cm 幅 44.5 cm 奥行 46.4 cm 重量 15.4 kg

1.2 周辺機器

1.2.1 モニター

モニターは通常(液晶含む)のもので使用可。SSHで遠隔操作するのならばモニター無しでも問題ない。

2 Solarisのインストール

2.1 インストール Solaris 8 sparc 7/03

2.1.1 ブート

画面に表示が出たら"Stop-a"とする、すると"ok >"という表示に替わる。ここでインストールCDをセットし"boot cdrom"とタイプすることによりインストーラが立ち上がる。

2.1.2 言語の選択

言語の選択を要求されるので"6"でJapaneseを選択する。

2.1.3 Mini Root

「Solarisインストールソフトウェア用にc0t0d0s4を使用してかまいませんか？」
"y"で進む。

これにより指定領域にミニルートのコピーが始まる。終了後はリブートが促されるので、指示に従い"Enter"を押し画面がBIOSになったらCD-ROMを取り出す。

2.1.4 ようこそ

再起動後、GUIの画面になりようこそと表示される。

2.1.5 ネットワーク接続性

ネットワークに接続:"する"を選択し、"次へ"で進む。

2.1.6 DHCP

DHCPを使用:"いいえ"を選択し、"次へ"で進む。

2.1.7 ホスト名

ホスト名:今回はホスト名を"yebisu"とし記入、"次へ"で進む。

2.1.8 IPアドレス

IPアドレス:"192.168.0.□"と記入(□の数字は参加者それぞれに割り当てる)、"次へ"で進む。

2.1.9 ネットマスク

ネットマスク:"255.255.255.0"と記入し、"次へ"で進む。

2.1.10 IPv6

IPv6を使用:"いいえ"を選択し、"次へ"で進む。

2.1.11 デフォルトルート

"1つを指定"を選択し、"次へ"で進む。
ルーターのIPアドレス:"192.168.0.1"と記入、"次へ"で進む。

2.1.12 Kerberos

Kerberosを有効にする:"いいえ"で進む。

2.1.13 ネームサービス
ネームサービス:"DNS"を選択し、"次へ"で進む。

2.1.14 ドメイン名
ドメイン名:"beer.jp"と記入し、"次へ"で進む。

2.1.15 DNSサーバのアドレス
サーバーのIPアドレス:"192.168.0.□"と記入し、"次へ"で進む。

2.1.16 DNS検索一覧
ドメインの検索: 記入せずそのまま進む。

2.1.17 時間帯
時間帯の指定:"地域"を選択し、"次へ"で進む。

2.1.18 地域
"アジア東部"から"日本"を選択し、"次へ"で進む。

2.1.19 日付と時刻
現在の日時に設定し、"次へ"で進む。

2.1.20 rootパスワード
root用のパスワードを2回入力し、"次へ"で進む。

2.1.21 プロキシサーバー構成
"インターネットに直接接続"を選択し、"次へ"で進む。

2.1.22 情報の確認
入力値を確認し、問題なければ"確認"で進む。

2.1.23 ようこそ
"次へ"で進む。

2.1.24 インストーラオプション
インストール後に、自動的にリブートするようにしますか?:"はい"
インストール後に、CD/DVDを自動的に取り出すようにしますか?:"はい"を選択し、"次へ"で進む。

2.1.25 媒体の指定
媒体:"CD/DVD"を選択し、"次へ"で進む。

2.1.26 ディスクの挿入
指定のディスクをセットし、"了解"で進む。

2.1.27 アップグレードまたは初期インストールの選択
"初期インストール"を選択。

2.1.28 インストール形式の選択
"カスタムインストール"を選択し、"次へ"で進む。

2.1.29 ソフトウェアのロケール選択
すでに日本語が選択されているので、そのまま"次へ"で進む。

2.1.30 システムのロケール選択
すでに日本語が選択されているので、そのまま"次へ"で進む。

2.1.31 製品の選択
オンラインドキュメントのインストール指定、必要なものを選択し、"次へ"で進む。

2.1.32 追加製品の指定
"なし"を選択し、"次へ"で進む。

2.1.33 Solarisソフトウェアグループの選択
今回は"全体ディストリビューション"のデフォルトパッケージを選択し、"次へ"で進む。

2.1.34 ディスクの選択
使用するディスクを">"で選択し、"次へ"で進む。

2.1.35 fdiskパーティションをカスタマイズするディスクの選択
そのまま、"次へ"で進む。

2.1.36 データの保存
"いいえ"で進む。

2.1.37 ファイルシステムの配置
"c0d0"などの指定ディスクを選択し"変更"で進む。この設定画面ではシリンダによるサイズの指定となるので注意が必要。下記のようにサイズを指定しパーティションを作成する。残りをすべて"/export/home"に割り当てる。変更後、"次へ"で進む。

```
/               300MB
/usr            4000MB
/var            3000MB
/opt            4000MB
/tmp            1000MB
/export/home    残りすべて
```

2.1.38 インストールの準備完了
"インストール開始"でインストールが始まる。この後は指示に従いCD-ROMを順次セットする。

2.1.39 リブート
すべてが終了し継続するために"リブートする"を選択する。

2.1.40 ログイン

リブート後、ログイン画面が表示されればインストールは完了。

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、設定

4.1 基本情報

■ WindowsやMacintosh(Mac OS 9まで)との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
■ rootと一般ユーザの違い
■ /etc，/usr，/var，/export/home，/tmp
■ ホームディレクトリ
■ 起動状況 dmesg，ネットワークの設定 ifconfig
■ マウントポイント
SolarisではCD-ROMは/cdrom、Floppy Diskは/floppyに自動的にマウントされる。取り出すときにはそれぞれ下記のようにする。
```
# eject cdrom0
# eject floppy0
```
■ 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とユーザのPATHの違い。
■ Xを使用しない
/etc/rc2.d/S99dtloginを無効にすることによりXは起動しなくなる。
■ AnswerBook

4.2 基本設定

4.2.1 グループを追加する
groupadd "グループ名"

ここで自分で使うグループを"hotei"とし設定する。
```
# groupadd hotei
```

4.2.2 ユーザの追加
useradd -m -d "ユーザのホームディレクトリ" -g "グループ名" -s "シェル" "ユーザ名"
シェルには/usr/bin/csh，/usr/bin/tcsh，/usr/bin/bashなどを利用する。

ここで自分で使うユーザを任意の文字列で設定する。
```
# useradd -m -d /export/home/your_id -g hotei -s /usr/bin/bash your_id
```
ホームディレクトリ名とユーザ名は同じものにする。

4.2.3 パスワードの設定
ユーザappleのパスワード設定例
```
# passwd apple
New password:
Re-enter new password:
```
2回入力するよう促される。パスワードの設定をするとそのユーザが利用可となる。

4.2.4 シェルの設定

user用のbash設定ファイル

```
$ vi ~/.bash_profile
```

```
if [ -f ~/.bashrc ]; then
   .  ~/.bashrc
fi
```

```
$ vi ~/.bashrc
```

```
PATH=$PATH:/usr/local/bin:/opt/sfw/bin
export PATH
export LD_LIBRARY_PATH=/usr/local/lib
export EDITOR=vi
export PAGER=less
umask 022
```

4.2.5 root用のbash設定ファイル

.bash_profileは共通なので先程作ったものをコピーして使う。

```
# cp /export/home/user/.bash_profile  ~/
# vi ~/.bashrc
```

```
PS1="[\u@\h  \W]#  "
PATH=$PATH:/usr/local/bin
PATH=$PATH:/opt/sfw/bin
PATH=$PATH:/usr/ucb
PATH=$PATH:/usr/ccs/bin
PATH=$PATH:/usr/openwin/bin
export LD_LIBRARY_PATH=/usr/local/lib
export EDITOR=vi
export PAGER=less
alias rm='rm -i'
umask 022
```

4.2.6 Networkの設定

ネットワークに最低限必要な設定をする。

```
# vi /etc/nsswitch.conf
```

```
hosts: files  ←ここにdnsを追加し下記の様に変更する。
hosts: files dns
```

```
# vi /etc/defaultrouter
```

```
192.168.0.1  ←ここにはルータのIPアドレスを記入する。
```

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

```
nameserver 192.168.0.□  ←DNSのIPアドレスを指定。
```

DNSは最大3つまで指定できる。

4.2.7 ネットワークの確認

```
# ping 127.0.0.1        ←ループバックアドレスを確認する。
# ping 192.168.0.□      ←IPアドレスが設定されているか確認する。
# ping 192.168.0.1      ←ルータまでつながるか確認する。

# nslookup www.sun.co.jp  ←ドメイン引きを確認する。

# ifconfig -a
```

で現在のネットワークの設定状況を見ることが出来る。

4.3 パッケージのインストール

4.3.1 インストールの仕方

Solaris用のパッケージはSunsiteなどより手に入る。

```
Sunsite <http://sunsite.sut.ac.jp/sun/solbin/>
```

パッケージのインストール

```
pkgadd -d "パッケージあるディレクトリ" "パッケージ名"
```

Sunsiteのパッケージのインストール

```
pkgadd -d "パッケージ名"
```

/tmpにtopのパッケージがあるとした場合。

```
# cd /tmp/
# pkgadd -d top-3.5beta9-sol8-local
```

としてインストールできる。

```
# pkginfo
```

でインストールされているパッケージの一覧を見ることが出来る。

4.3.2 SOFTWARE COMPANION CDを使いパッケージをインストール

COMPANION CD-ROMを挿入すると自動的にインストーラーが起動するので指示に従い進む。"インストール形式の選択"では"カスタムインストール"を選択し進む。今回はmake, gcc, lynx, topをインストールするので"製品の選択"では"Application/Utilities"、"Development/Languages"、"Development/Tools"のみを"カスタムインストール"にし、それ以外はすべて"インストールしない"とする。次に"コンポーネントの選択"Application/Utilities"からは"gtar - GNU tar"と"GNU wget - wget"と"lynx - cursor-based web browser"と"top"を"(Development/Languages)"からは"gcc - GNU Compiler Collection"を"(Development/Tools)"からは"gmake - GNU Make"を選択し進む。準備完了後、"インストール開始"でインストールが始まる。終了後は"eject cdrom0"などのコマンドを使いCD-ROMを取り出す。

```
Application/Utilities   gtar, GNU wget, lynx, top
Development/Languages   gcc
Development/Tools       gmake
```

4.3.3 tarを設定する

GNU tarを既存のtarと置き換える。

```
# cd /usr/sbin/
# mv tar tar.org
# cd /usr/bin/
# rm tar
# mkdir -p /usr/local/bin
# ln -s /opt/sfw/bin/gtar /usr/local/bin/tar
```

4.4 セキュリティ対策

4.4.1 SunSolve

http://jp.sunsolve.sun.com/

定期的(週に一度など)に上記のSunSolveサイトへアクセスし該当するパッケージがないか確認する。Solaris 8向けのものは"SunOS 5.8"と表記されている。

4.4.2 パッチのインストール

解凍し、patchaddコマンドを利用する。

```
# unzip /export/home/user/download/108870-15.zip
# patchadd /export/home/user/download/108870-15
```

4.4.3 パッチの確認

現在インストールされているパッチを確認する。

```
# showrev -p
```

4.4.4 パッチクラスタのインストール

複数のオペレーティングシステムパッチとセキュリティ関連のパッチをまとめたものがクラスタという形で提供されている。解凍し、ディレクトリ内のinstall_clusterを実行することで順次パッチが追加される。

```
# unzip /export/home/user/download/8_Recommended.zip
# cd /export/home/user/download/8_Recommended/
# ./install_cluster
```

4.4.5 ダウンロード

ダウンロードするプログラム専用のディレクトリをユーザのホームディレクトリに作成し、すべてのプログラムはここにダウンロードするようにする。

```
# mkdir /export/home/user/download
# chown user /export/home/user/download/
# cd /export/home/user/download/
```

wgetコマンドに続き、ダウンロードするプログラムのパスを指定する。パスをコピーするにはInternet Explorerの場合はリンク上で右クリックし"ショートカットのコピー"を選択。Netscapeの場合は"リンクの場所をコピー"を選択する。

```
# wget http://www.apache.org/dist/httpd/httpd-2.0.35.tar.gz
```

5.1 基本的なコマンド

5.1.1 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 -F     ファイルタイプを表示しながらの一覧
            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 lemon.txt
            ファイルbanana.txtをlemon.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/lemon.txt
            ファイルbanana.txtをlemon.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 ../lemon.txt
            ファイルbanana.txtをlemon.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 strawberry/ /tmp/strawberry
            カレントディレクトリにあるstrawberry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 banana.txt lemon.txt
            banana.txtというファイル名をlemon.txtという名に変更する。
$ mv strawberry /tmp/
            カレントディレクトリにあるstrawberryというディレクトリを/tmpに移動する。
```

mkdir ディレクトリを作成する
```
$ mkdir strawberry
```
カレントディレクトリにstrawberryというディレクトリを作る。
```
$ mkdir /tmp/strawberry
```
/tmpの中にstrawberry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 strawberry
```
ディレクトリstrawberry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 strawberry
```
ディレクトリstrawberryの所有者をappleに変更する。
```
$ chown -R apple strawberry
```
strawberry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。

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 strawberry
```
ディレクトリstrawberry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 melon.pl
```
-rwxr-xr-xとなり所有者はmelon.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。

```
$ chmod 755 mango
```
mangoディレクトリに対してもファイルと同じように権利を設定することができる。

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/strawberry/ -name apple.txt -print
```
ディレクトリ/tmp/strawberry内でapple.txtを検索し表示する。
```
$ find /tmp/strawberry/ -name "*.txt" -print
```
ディレクトリ/tmp/strawberry内で.txtで終わるファイルを検索し表示する。

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 -ef
```
全プロセスを表示する。
```
$ ps -ef | 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。

prstat, 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 -5 apple.txt
```
apple.txtの最後の5行のみ表示する。

su 管理モードに入る。super user, substitute userの略。
```
$ /usr/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/usr/bin/suと利用した方が良い。

gzip/gunzip ファイルを圧縮/解凍する。
```
$ gunzip orange.tar.gz
```
orange.tar.gzを解凍する。

tar ファイルをまとめる、展開する。

```
$ tar xvf orange.tar
```

orange.tarからすべてのファイルを取り出す。

```
$ tar xvfz orange.tar.gz
```

orange.tar.gzからすべてのファイルを取り出す(gtarのみ)。

wget ファイルのダウンロード。

```
$ wget http://www.apache.org/dist/httpd/apache_1.3.24.tar.gz
```

updates.redhat.comより指定のファイルをダウンロードする。

ping ネットワークの応答を確認する。

```
$ ping 192.168.1.1
```

IPアドレスを指定しての確認。

```
$ ping -s 192.168.1.1
```

IPアドレスを指定しての詳細確認。

init 単独で使っている場合のシステムの終了、再起動(rootのみ)。

```
# init 5
```

システムを停止して、電源の切断をする。

```
# init 6
```

システムを再起動する。

shutdown サーバとして使っている場合のシステムの終了、再起動(rootのみ)

```
# shutdown -y -g60 -i0
```

60秒後にシステムを停止。

```
# shutdown -y -g0 -i6
```

すぐにシステムを再起動する。

5.1.2 ftpコマンドの使い方

ftp.sun.co.jpに接続する場合。

```
$ ftp ftp.sun.co.jp
```

接続されるとユーザ名の入力を促される。

```
Connected to ftp.sun.co.jp.
Name (ftp.sun.co.jp: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、パスワードに任意の文字列を入力する。

```
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-1.3.20.tar.gzがファイル名の場合は下記のようにする。

```
ftp> get apache-1.3.20.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。

5.1.3 UNIXコマンドの参考サイト

UNIXの部屋 <http://x68000.startshop.co.jp/~68user/unix/>
UNIX1年生 <http://www.tokaido.co.jp/syoko/>

5.1.3 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。
ctrl-f	次ページへ移動。
ctrl-b	前ページへ移動。
0	行頭へ移動。
$	行末へ移動。
g	先頭行に移動。
G	最終行に移動。
23G	23行目へ移動。
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/melon/g	appleをmelonに置き換える。
:%s/apple/melon/c	appleをmelon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 showmode	動作モードを表示する。
:set nu	行番号を表示する。
:set nonu	行番号を表示しない。

5.1.4 OpenBootコマンド

マシンのスイッチをONにし、画面に表示が出たら"Stop-a"とする、すると"ok"という表示に替わる。この状態で利用出来るコマンドがOpenBootコマンドである。このコマンドでハードウェアの初期化やテスト、OSの起動指定などが実行できる。

help	ヘルプ主要カテゴリを表示する
help command	コマンドのヘルプを表示する
banner	電源投入時の表示を再現する
.version	OpenBootのバージョンを表示する
boot	通常の起動。OpenBootを終了して、OSを起動する場合にbootとすればよい。
boot cdrom	CD-ROMからOSを起動する
boot net	ネットワークからOSを起動する
boot -r	ハードウェアを増設した直後は"-r"オプションをつけてOSを起動する必要がある。
probe-ide	接続されているIDEデバイスを確認する
probe-scsi	接続されているSCSIデバイスを確認する
test /memory	メモリのテスト
test floppy	フロッピードライブのテスト
printenv	現在のシステム変数とデフォルト値を表示する

6 SSHの設定

6.1 OpenSSH

6.1.1 SSH[注3] とOpenSSH

セキュリティを考慮し、リモート操作にはtelnetを使わず、SSHを使用する。SSHにはSSHとOpenSSHの2種類あるが、ここではオープンソースであるOpenSSHをインストールし利用する。OpenSSHにはzlib(データ圧縮のためのライブラリ)とOpenSSL(暗号化に必要なライブラリ)が必要となるので先にこの二つをインストールする。

SSH <http://www.ssh.com/>
OpenSSH <http://www.openssh.com/>

6.1.2 zlibのインストール

```
zlib <http://www.gzip.org/zlib/>
# cd /usr/local/src/
# tar xvfz /export/home/user/download/zlib-1.2.1.tar.gz
# cd zlib-1.2.1/
# ./configure
# gmake test
# gmake install
```

6.1.3 OpenSSLのインストール

```
OpenSSL <http://www.openssl.org/>

# cd /usr/local/src/
# tar xvfz /export/home/user/download/openssl-0.9.7c.tar.gz
# cd openssl-0.9.7c/
# ./config
# gmake
# gmake test
# gmake install
```

6.1.4 OpenSSHのインストール

OpenSSHのインストールにはユーザsshdが必要となるので、予め作成しておく。

```
# groupadd shibuya
# useradd -g shibuya -d /dev/null -s /bin/true sshd

# cd /usr/local/src/
# tar xvfz /export/home/user/download/openssh-3.7p1.tar.gz
# cd openssh-3.7p1/
# ./configure
# gmake
# gmake install
```

注3 SSH [Secure Shell]

6.1.5 OpenSSHの設定

設定ファイルをコピー保存して、設定を行う。初期設定では直接rootでログインできる設定になっているのでこれを無効化する。

```
# cd /usr/local/etc/
# cp sshd_config sshd_config.default

# vi sshd_config
```

```
#PermitRootLogin yes  ←この行を
PermitRootLogin no  ←この様に
```

6.1.6 自動起動

/etc/rc2.dに下記のスクリプトをS50sshdというファイル名で作成する。

```
# vi /etc/rc2.d/S50sshd
```

```
#!/usr/bin/sh
/usr/bin/echo 'sshd starting.'
/usr/local/sbin/sshd &
```

実行権を付ける。

```
# chmod +x /etc/rc2.d/S50sshd
```

6.1.7 手動での起動

直接の起動。これでSSHクライアントからの操作が可能になる。

```
# /etc/rc2.d/S50sshd
```

6.1.8 テスト

ますローカルでの接続を確認する。ログイン可能であれば、次にウィンドウズなどのSSHクライアントより接続してみる。

```
# ssh -l user 127.0.0.1
```

6.1.9 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/
```

6.1.10 SSH Macintosh用クライアント

```
http://www.macssh.com/
```

よりMacSSHをダウンロードして利用可。

6.2 不要なサービスの停止

6.2.1 セキュリティへの考慮

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。

6.2.2 /etc/rc*.dで不必要な常駐サービスを止める

ファイル名の頭に"_"を付けることにより起動時に実行されなくなる。

```
# cd /etc/rc2.d/
# mv S73nfs.client _S73nfs.client←NFS(Network File System)
# mv S90wbem _S90wbem                ←WBEM(Web-Based Enterprise Management)

# cd /etc/rc3.d/
# mv S76snmpdx _S76snmpdx     ←SNMP(Simple Network Management Protocol)
# mv S77dmi _S77dmi           ←Solstice Enterprise Manager(ネットワーク上の他のマシンの
                              設定などを行うもの)
```

6.2.3 inetdの利用設定

inetdを利用して起動するプロセスを指定して停止したい場合は/etc/inetd.confで"#"を利用し無効にする。inetd.confには書き込み権が設定されていないので、保存する時は":w"の代わりに":w!"を利用する。

```
# vi /etc/inet/inetd.conf
```

```
ftp stream tcp6 nowait root /usr/sbin/in.ftpd in.ftpd  ←これを
#ftp stream tcp6 nowait root /usr/sbin/in.ftpd in.ftpd  ←この様に
```

6.2.4 inetdの停止

現在使われているinetdを止める。

```
# /etc/rc2.d/S72inetsvc stop
```

必要無ければinetdそのものが起動できないように変更する。

```
# chmod 0 /usr/sbin/inetd
# mv /etc/rc2.d/S72inetsvc /etc/rc2.d/_S72inetsvc
```

7 DNSの設定

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

7.1.1 BINDのインストール

>> BINDはセキュリティホールとなる場合が高いので常に最新版を入れる様に心掛ける。

/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 /export/home/user/download/bind-9.2.3.tar.gz
```

コンパイルする。

```
# cd bind-9.2.3/
# ./configure --with-openssl=/usr/local/ssl
# gmake
# gmake install
```

7.1.2 BIND専用のユーザの作成

起動専用のユーザnamedを作成する。

```
# groupadd named
# useradd -g named -d /dev/null -s /bin/true named
```

7.1.3 named.confの設定

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
```

named.confで指定するrndc.confはソースディレクトリよりコピーして使用する。

```
# cp /usr/local/src/bind-9.2.3/bin/rndc/rndc.conf /etc/
```

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";
};

include “/etc/rndc.key”;
```

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
```

7.1.4 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. (
                         20040212   ;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 20040212
OK
```

7.1.5 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. (
                       20040212     ;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.
```

7.1.6 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. (
                           20040212     ;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        192.168.0.249
```

7.1.7 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. (
                       20040212    ;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.
249     IN    PTR      tempest.beer.jp.
```

7.1.8 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
```

7.1.9 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 rndckey
# chown named /etc/rndc.key
# chmod 400 /etc/rndc.key
```

7.1.10 rndc.confの修正

rndc.confでrndckeyを指定する。

```
options {
        default-server  localhost;
        default-key     "key";  ←ここを
        default-key     "rndckey";  ←この様に
};

server localhost {
        key     "key";  ←ここを
        key     "rndckey";  ←この様に
};
```

既存の最終4行を削除し、rndc.keyの下記4行を/etc/rndc.confにコピーする。

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

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

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

7.1.11 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3行を追加する。

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

```
controls {
      inet 127.0.0.1 allow { localhost; } keys { rndckey; };
};
```

7.1.12 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
```

7.1.13　自動起動

/etc/rc2.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/rc2.d/
# vi S45bind
```

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

実行権をつける。

```
# chmod +x S45bind
```

7.1.14　手動での起動

```
# /etc/rc2.d/S45bind
```

で起動する。

7.1.15　/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。
```

7.1.16 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。

7.1.17 設定の再読み込み

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

7.1.18 BINDの稼動状況確認

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

7.1.19 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。

8 Web Serverの設定

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

8.1.1 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 /export/home/user/download/httpd-2.0.48.tar.gz
```

コンパイルする。

```
# cd httpd-2.0.48/
# ./configure
# gmake
# gmake install
```

8.1.2 Apache専用のユーザの作成

起動専用のユーザapacheを作成する。

```
# groupadd apache
# useradd -g apache -d /dev/null -s /bin/true apache
```

8.1.3 設定

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 ←ここと
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
```

8.1.4 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
```

8.1.5 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。

8.1.6 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。

8.1.7 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。

8.1.8 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
```

8.1.9 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
```

8.1.10 停止

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

8.1.111 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
```

8.1.12 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-2.0.48.tar.gz
466c63bb71b710d20a5c353df8c1a19c  httpd-2.0.48.tar.gz
```

8.1.13 参考サイト

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

9 FTP Serverの設定

9.1 ProFTPD[注5] <http://www.proftpd.org/>

9.1.1 インストール

展開する。

```
# cd /usr/local/src/
# tar xvfz /export/home/user/download/proftpd-1.2.9.tar.gz
# cd proftpd-1.2.9/
```

コンパイルする。

```
# ./configure --enable-pam=no
# gmake
# gmake install
```

9.1.2 ProFTPD起動ユーザ

proftpd実行専用のユーザとグループを作る。

```
# groupadd proftpd
# useradd -g proftpd -d /dev/null -s /bin/true proftpd
```

9.1.3 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

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

<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。
```

9.1.4 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]

9.1.5 自動起動

/etc/rc2.dに起動スクリプトを書く。

```
# vi /etc/rc2.d/S86proftpd
```

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

実行権を付ける。

```
# chmod +x /etc/rc2.d/S86proftpd
```

9.1.6 手動での起動

```
# /etc/rc2.d/S86proftpd
```

で起動する。

9.1.7 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。問題なければウィンドウズ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。

9.1.8 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移動出来ないようにする。

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

```
#DefaultRoot    ~  ←#を取る
```

ユーザでログインしてホームディレクトリより上には移動出来ないことを確認する。

9.1.9 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
```

9.1.10 Window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

10 Mail Serverの設定

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

10.1.1 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 /bin/true alias
# useradd -g nofiles -d /var/qmail -s /bin/true qmaild
# useradd -g nofiles -d /var/qmail -s /bin/true qmaill
# useradd -g nofiles -d /var/qmail -s /bin/true qmailp
# groupadd qmail
# useradd -g qmail -d /var/qmail -s /bin/true qmailq
# useradd -g qmail -d /var/qmail -s /bin/true qmailr
# useradd -g qmail -d /var/qmail -s /bin/true qmails
```

解凍する。

```
# cd /usr/local/src/
# tar xvfz /export/home/user/download/qmail-1.03.tar.gz
# cd qmail-1.03/
```

コンパイル時にgccを利用することを指定する。

```
# vi conf-cc
```

```
cc -O2 ←ccを
gcc -O2 ←gccに
```

```
# vi conf-ld
```

```
cc -s ←ccを
gcc -s ←gccに
```

インストールする。

```
# gmake setup check
# ./config-fast yebisu.beer.jp
```

"yebisu.beer.jp"は登録するホスト、ドメイン名により適宜変更する。

10.1.2 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
```

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。

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

転送先を記述

```
banana ←ユーザ名がbananaの場合
```

10.1.3 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。

10.1.4 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
```

10.1.5 sendmailの設定変更

sendmailを止める。

```
# /etc/rc2.d/S88sendmail stop
```

sendmailが起動時に立ち上がらないようにする。

```
# cd /etc/rc2.d/
# mv S88sendmail _S88sendmail
```

sendmailを起動出来ない様にする。

```
# chmod 0 /usr/lib/sendmail
# mv /usr/lib/sendmail /usr/lib/sendmail.bak
# chmod 0 /usr/lib/mail.local
```

10.1.6 自動起動

/etc/rc2.dにS88qmailというファイルを作り、下記の様に記述。

```
# vi /etc/rc2.d/S88qmail
```

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

```
# chmod +x S88qmail
```

で実行権を与える。これで起動時にqmailが自動的に立ち上がるようになる。

10.1.7 手動での起動

```
# /etc/rc2.d/S88qmail
```

起動の確認。

```
# ps -ef | grep qmail  とし
     qmails 213   1   0 13:09:23 ? 0:00 qmail-send
     qmaill 219  213   0 13:09:23 ? 0:00 splogger qmail
       root 220  213   0 13:09:23 ? 0:00 qmail-lspawn ./Maildir/
     qmailr 221  213   0 13:09:23 ? 0:00 qmail-rspawn
     qmailq 222  213   0 13:09:23 ? 0:00 qmail-clean
```

という様な表示がされるか確認する。

10.1.8 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/lib/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"を押すとメールが送信される。

/export/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。

10.1.9 manの設定

.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。

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

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

下記コマンドで設定が反影される。

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

10.1.10 参考サイト

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

10.2 SMTP[注6]

10.2.1 tcpserverのインストール

qmailのsmtpdをtcpserverより起動する。
ucspi-tcp <http://cr.yp.to/ucspi-tcp.html>

展開、コンパイルする。

```
# cd /usr/local/src/
# tar xvfz /export/home/user/download/ucspi-tcp-0.88.tar.gz
# cd ucspi-tcp-0.88/
# gmake
# gmake setup check
```

10.2.2 smtpの設定。

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。

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

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

ftp_rules，smtpd_rules，pop3d_rulesで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に設定する。

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
```

また、IPアドレスだけでなくドメイン名での指定も可能。

注6 SMTP [Simple Mail Transfer Protocol]

10.2.3 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
```

10.2.4 自動起動

起動するためにまず、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/rc2.dにS88qmail-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数字を記入する。

```
# vi /etc/rc2.d/S88qmail-smtpd
```

```
#!/usr/bin/sh
/usr/bin/echo '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/rc2.d/S88qmail-smtpd
```

10.2.5 手動での起動

```
# /etc/rc2.d/S88qmail-smtpd
```

10.3 POP[注7]

10.3.1 checkpasswordのインストール

checkpasswordを利用しPOPを起動する。

```
checkpassword <http://cr.yp.to/checkpwd.html>
```

解凍する。

```
# cd /usr/local/src/
# tar xvfz /export/home/user/download/checkpassword-0.90.tar.gz
```

インストール。

```
# cd checkpassword-0.90/
# gmake
# gmake setup check
```

10.3.2 checkpasswordの設定

tcpserverを使いqmail-popupを起動する。

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

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

10.3.3 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
```

10.3.4 自動起動

/etc/rc2.dにS88qmail-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起動する。"yebisu.beer.jp"は自分のホスト、ドメイン名に合わせる。

```
# vi /etc/rc2.d/S88qmail-pop3d
```

```
#!/usr/bin/sh
/usr/bin/echo '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/rc2.d/S88qmail-pop3d
```

10.3.5 手動での起動

```
# /etc/rc2.d/S88qmail-pop3d
```

ウィンドウズなどのメールソフトでsmtpとpopサーバの設定をし、送受信できるか確認してみる。
qmailの動作確認は/var/log/syslogを見る。

注7 POP [Post Office Protocol]

10.4 APOP[注8]

10.4.1 APOPのインストール

checkpwを利用しAPOPを起動する。
checkpw <http://checkpw.sourceforge.net/checkpw/>

```
# cd /usr/local/src/
# tar xvfz /export/home/user/download/checkpw-1.00.tar.gz
```

インストール

```
# cd checkpw-1.00/
# gmake
# gmake setup check
```

10.4.2 checkpw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
```

10.4.3 自動起動

/etc/rc2.dにS88qmail-apop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。S88qmail-pop3dとS88qmail-apopが両方共起動しないよう注意。"yebisu.beer.jp"は自分のホスト、ドメインに合わせる。

```
# vi /etc/rc2.d/S88qmail-apop
```

```
#!/usr/bin/sh
/usr/bin/echo '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/rc2.d/S88qmail-apop
```

10.4.4 手動での起動

pop3dが起動している場合はそちらを先に終了しておく事。

```
# /etc/rc2.d/S88qmail-apop
```

10.4.5 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、使わない方のファイル名先頭に"_"を付け、起動できないようにする。

```
# mv S88qmail-xxx _S88qmail-xxx
```

注8 APOP [Authenticated POP]

10.4.6 Window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

11 セキュリティについて

11.1 現状

「『MyDoom』がインターネット史上最悪の急拡大，『SoBig.F』をはるかに上回る」，米社が報告

「大量の電子メールを送信するワーム『MyDoom』は，インターネット史上最も急速に拡散している」とする報告を，米MX Logicが米国時間1月29日に行った。アンチスパムおよびアンチウイルス・メール技術を提供する同社は，東部時間29日午後12時半の時点で，600万通の感染メールを遮断したという。「『SoBig.F』の場合，同様の時間内に遮断した感染メールは350万通だった」（MX Logic社）

MX Logic社によれば，同社は現在，1時間に24万通のMyDoom感染メールを遮断している。電子メール6通のうち1通が感染している計算である。SoBig.Fは1時間あたりの遮断数が9万通で，感染メールの割合は12通のうち1通だった。

MX Logic社CTOのScott Chasin氏は，「いまだにワーム拡散のピークに至っていない。今後数日で何が起こるか，関心が高まる。特に，2月1日にワームがDDoS攻撃を開始してからが問題だ」と述べた。

MyDoomは，「Novarg」「WORM_MIMAIL.R」とも呼ばれる。電子メールの添付ファイルのかたちで自身を送りつける。添付ファイルには「.zip」拡張子が付いており，例えば「document.zip」「message.zip」「readme.zip」といった名称を使う。「.exe」「.pif」「.command」「.scr」などのファイル形式の場合もある。電子メールの件名は，エラー・メッセージを装うことが多い。

添付ファイルを実行すると，ワームは「バックドア」プログラムをインストールし，作成者から命令を受け取るための通路を開こうとする。命令を受け取ったワームは，スパム・メールを大量送信したり，IPスプーフィング（なりすまし）を実行する可能性がある。また，ワームは2004年2月1～12日に米SCO GroupのWebサイト「www.sco.com」にDDoS攻撃を仕掛けるよう，感染したコンピュータを設定しようとする。

なお，1月28日には早くもMyDoomの亜種「Mydoom.B」が見つかっている。スペインのPanda Softwareは，「オリジナルよりさらに危険」と警告している。SCO Group社のWebサイトに加え，米MicrosoftのWebサイト「microsoft.com」も攻撃対象にする。

ITPro 2004.01.30
<http://itpro.nikkeibp.co.jp/>

11.2 基礎知識

11.2.1 シュレッダー

初めに必要なものは何か?

11.2.2 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。

11.2.3 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。

11.2.4 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。

11.2.5 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。

11.2.6 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。

11.2.7 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。

11.2.8 様々な攻撃

DoS (Denaial 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

11.2.9 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。

11.2.10　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。

11.2.11　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、

11.2.12　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、、、

11.3　セキュリティの方針

11.3.1　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。

11.3.2　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。

11.3.3　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。

11.3.4　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。

11.3.5 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。

11.3.6 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, melon, 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 Naniwo Shinjite Koretaka NA-")

11.3.7 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。

11.3.8 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。

11.3.9 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。

11.3.10 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。

11.4 暗号

11.4.1 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。

11.4.2 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。

11.4.3 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。

11.4.4 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。

11.5 バックアップ

11.5.1 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。

11.5.2 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。

11.5.3 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。

11.5.4 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。

11.5.5 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。

11.5.6 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。

11.5.7 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。

11.5.8 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。

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

11.5.9　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。

11.5.10　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。

11.5.11　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。

11.6 参考サイト

11.6.1 セキュリティ関連サイト

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。

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

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/

12 Solaris参考資料

12.1 資料

12.1.1 参考URL

Sun Microsystems <http://www.sun.com/>
サン・マイクロシステムズ株式会社 <http://www.sun.co.jp/>

BigAdmin <http://www.sun.com/bigadmin/>
Sun Online Manual <http://docs.sun.com/>

Sunsite Solaris Freeware Project <http://sunsite.sut.ac.jp/sun/solbin/>

12.1.2 参考書籍

「Solaris システム管理」著者 城谷洋司
株式会社アスキー ISBN4-7561-3568-4
「新インターネットサーバ構築術」著者 石橋勇人
ソフトバンク パブリッシング株式会社 ISBN4-7973-0552-5
「はじめてのqmail」著者 中村文則
株式会社技術評論社 ISBN4-7741-1588-6
「UNIXコマンドポケットリファレンス」著者 中西 隆
株式会社技術評論社 ISBN4-7741-0508-2