

映像配信サーバ入門

金山 典世

E-mail: kanayama@wakhok.ac.jp

稚内北星学園大学

情報メディア学部

TEL 0162-32-7511 FAX 0162-32-7500

2005年10月1日

目次

第 1 章	はじめに	1
第 2 章	インストールと初期設定	3
2.1	インストール方法	3
2.2	初期設定	4
第 3 章	ストリーミング配信	9
3.1	ウェブサーバと配信サーバ	9
3.2	映像ファイル	9
3.2.1	子ディレクトリを作る方法	9
3.2.2	別ディレクトリに格納する方法	10
3.3	映像配信の準備	11
3.4	最後に	13
付録 A	DSS 作成方法	15
付録 B	ユーザー管理	17
B.1	ユーザー登録	17
B.2	sysinstall でのユーザ登録 (基本)	17
B.3	アカウントの発行	19
B.3.1	/etc/passwd ファイルの例	22
B.4	グループ登録	23
B.5	ホームディレクトリと配布ファイル	24
B.6	ファイルのパーミッション	25
付録 C	環境設定	29
C.1	sysinstall での基本設定方法 (基本)	29
C.2	起動と rc ファイル	32
C.3	IP アドレスと名前解決	33
C.4	ディスク管理	34
C.4.1	ファイルシステムとマウント	34
C.4.2	ファイルシステムの異常	35
付録 D	FreeBSD の第一歩	37
D.1	FreeBSD のインストール	37
D.1.1	機器の選定	37
D.1.2	インストール方法の選択	38
D.1.3	フロッピーの準備	38
D.1.4	インストールの開始	39

D.1.5	インストールメニュー	39
D.1.6	root のパスワードを設定し忘れたら	39
D.2	インストール後の設定と変更	40
D.3	パッケージの追加	40
D.3.1	pkg-info	40
D.3.2	pkg_delete	42
D.3.3	pkg_update	42
D.4	ソースからの追加	42
D.5	システムの停止とリブート	43
D.5.1	shutdown	43
D.5.2	halt(reboot)	44

第1章 はじめに

ネットワーク上での動画などのストリーミング配信には、ストリーミング専用のサーバを用意することが通常です(ウェブサーバ上に置くだけの方法もありますが、色々と問題があります)。

ストリーミング配信サーバには、主なもので以下のものが上げられます。

1. Apple 社 QuickTime Streaming

Apple社の QuickTime フォーマットを配信するサーバソフトですが、同時に Darwin というフリーで配布されているプロジェクトからも、サーバがソースで公開されています(Darwin Streaming Server: DSS)。一方、ライブ中継ソフトもありますが、これには Mac が必要です。

2. RealNetworks 社 Helix Server

有名な RealVideo の配信ソフトです。無料で使えるバージョンはライセンスが1年間に限られていたり、少し不便なところがあります。ライブ中継ソフトは有料です。

3. Microsoft 社 Windows Media Service

WindowsMedia を配信するサーバです。サーバ自体は無料ですが、Windows Server を必要とします。つまり、OS にバンドルされているようなものであると考えて良いでしょう。また、ライブ中継ソフトも同じです。

4. videolan.org 開発 VideoLAN

MPEG2 や DV のプレイヤーを中心として開発されていますが、サーバも公開されています。WindowsMedia も配信可能となっていますが、テストした限りではまだうまく動いていません。

5. WIDE project 開発 DVTS

DV 品質の画像をライブ中継するためのソフトです。Linux, BSD 系、Windows など動作しますが、IP 的な制限があります。学校内で動かす分には問題ありません。帯域は 30Mbps 程度は必要です。

6. 広島大グループ開発 Robst

MPEG2 DV, HD など安定的に配信するために開発されています。マシンの色々な制約がありますが、ネットワークを使って離れた地点間の中継などに特徴があります。

ここでは、フリーに利用が出来、コーデックやツールが充実した Apple 社の Darwin Streaming Server(以下 DSS と略します) を使うことにします。

Darwin Streaming Server は無料ですが、Apple 社に登録が必要ですので、本講習会終了後も利用する場合には、以下の場所で登録を行ってください。

登録 <http://www.opensource.apple.com/aps1/index.html>

また、登録後は、以下の場所からソースコードをダウンロード出来ます。

ソース <http://developer.apple.com/darwin/projects/streaming/>

また、Mac OSX, Red Hat Linux, Windows 2000/2003 サーバ用のバイナリも上のページから手に入れることができます。その他の Unix でも、動作しますが、ソースコードからコンパイルすることが必要です。今回の講習ではこちらでコンパイルしたものをインストールしています。

講習で利用する OS は、FreeBSD という Unix OS で、バージョンは 4.11 を利用します。こうしたサーバに利用する場合、慣れていることから Windows を利用されることも多いかと思いますが、安定的な運用 (要するに放っておいても大丈夫) にはやはり Unix 系に一日の長があります。中でも、BSD 系は Linux のようなデスクトップ的な用途には向きませんが、サーバやルータなどには広く利用されているプラットフォームです (1 万円前後の家庭用アクセスルータなどにも実は結構使われています)。ハード的な故障がない限り、年単位で連続運用が可能であり、セキュリティ的にも非常に高いとされています。管理はある程度の知識を必要としますが、今回のような単なるストリーミングサーバとしてのみ利用する場合には、日常の管理は不要で、単にファイルをアップロードするだけで済みます。

なお、実際に学校などに持ち帰って利用する場合には、IP アドレスなどの設定が必要ですが、設定方法は付録で紹介します。

第2章 インストールと初期設定

以下では、バイナリファイルが既に出ていて、あるディレクトリにそれらが用意されているという環境を想定しています。もし、自分で一から用意したい場合には付録を参照してください。

2.1 インストール方法

まず最初にログインします。

```
login: root
passwd: xxxx
```

次に、X-window システムを立ち上げます。

```
# startx
```

複数の端末画面が開いていますが、どれか一つの画面で以下のように入力します (但し、Darwin... は長いので、/usr/home/movie/Dar まで入力したら、TAB キーを押して下さい。自動的に補完されるようになっています。)

```
# cd /usr/home/movie/DarwinStreamingSrvr5.5-Source
```

このディレクトリ上で、以下のインストールコマンドを実行します。

```
# ./Install
```

- 参考

但し、標準の Install コマンドには不具合があり、講習のディレクトリには修正したものが入っています。もし、自分で新たにインストールする場合には、以下の場所から修正版の Install コマンドを入手してください。また、FreeBSD 用の起動スクリプトも下に置いてありますので、指定の場所に置いてください。

```
http://www.wakhok.ac.jp/~kanayama/system/dss-install/
```

幾つかのファイルを自動的に適切なディレクトリにコピーした後に、以下のように表示されます。

```
Installation Complete
```

```
Darwin Streaming Server Setup
```

```
In order to administer the Darwin Streaming Server you must create an administrator user [Note: The administrator user name cannot contain spaces, or single double quote characters, and cannot be more than 255 characters long].
```

```
Please enter a new administrator user name:
```

ここで、DSS の管理者の名前を要求しているのですが、後で簡単に変更も出来るので、ここでは簡単に、movie と入力することにします。

次に、以下のようにその movie のパスワードを聞かれますので、ここでは仮に movie としておきます。(但し、実際に運用する際にはきちんとしたパスワードに変更しないとけません。79 文字までパスワードは使えますが、実際には 10 文字程度のパスワードで良いでしょう。)

```
You must also enter a password for the administrator user
[Note: The administrator password cannot contain spaces, or quotes,
either single or double, and cannot be more than 80 characters long].
Please enter a new administrator Password:
```

パスワードは安全のために、もう一度入力する必要があります。

```
Re-enter the new administrator password:
```

パスワードが正しければ、以下のようにセットアップスクリプトは終了します。

```
Adding userName movie
Setup Complete!
```

2.2 初期設定

これから先はウェブインターフェースで設定が可能になります。(ウェブサーバとは関係なく、DSS 自体がウェブでの設定をサポートしています)。

講習環境では、mozilla が導入されていますので、mozilla を立ち上げます。

```
# mozilla &
```

次に、端末ウィンドウで、以下のようにコマンドを入力し、現在の自分のサーバの IP アドレスを調べます。

```
# ifconfig
xl0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
options=3<RXCSUM,TXCSUM>
inet 192.168.0.10 netmask 0xfffff00 broadcast 192.168.0.255
...
```

ここでは、コマンドの下 3 行目の inet の後ろの値が IP アドレス (192.168.0.10) ですので、このアドレスを使って、DDS に mozilla でアクセスします。

mozilla の URL の指定の所に、以下のように指定をし、リターンを押します。

```
http://192.168.0.10:1220/
```

ここで、http:// の後ろに先に調べた IP アドレスを指定して下さい。IP アドレスの後ろの :1220 は固定で、DSS のウェブインターフェースにアクセスする時に用いる番号です。

DSS の設定を変更したり、セットアップの時に指定した管理者のパスワードなどを変更するにはこの URL を利用します。

- 参考

この手の設定で良く使う `http://localhost:1220/` は使えません。DSS が localhost に対してはウェブインターフェースを用意していないようです。

アクセスをすると、以下のようにログイン画面になりますので、先の管理者の名前とパスワードを入力します。



MP3 に対してもパスワードを要求しますが、管理者と同じものを指定して良いでしょう。



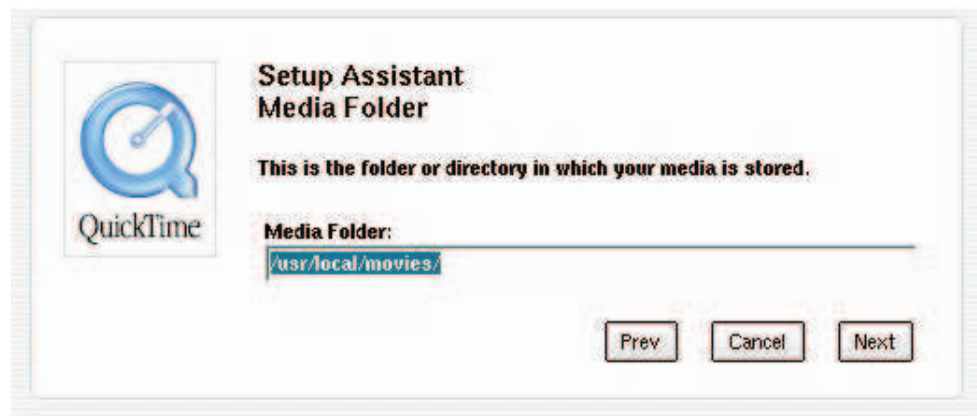
次に、SSL をこの管理に利用するか否かを以下のように聞かれますが、内部で動かして管理する分には SSL にしなくても構いません。この講習では SSL に必要なパッケージを入れていませんので、選択できません。

- 参考

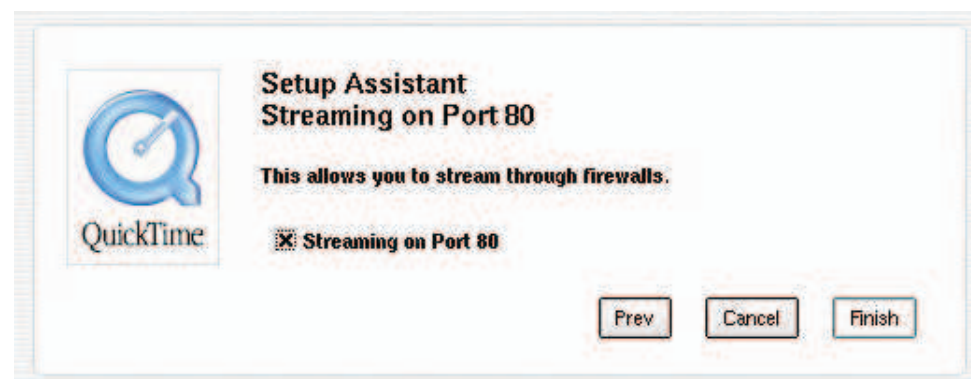
SSL は Secure Socket Layer の略で、アプリケーションの通信毎に安全な暗号回線を作るための仕組みです。ウェブで、取引をしたり、金融機関のページなどで一般に使われています。今の場合には、DSS を管理するためにクライアントからアクセスする通信を保護するために SSL を使うことが出来ます。



その次には、動画などのメディアファイルを置いておくディレクトリを以下のように聞かれますが、これはデフォルトのまま構いません (デフォルトの場所に既にサンプルが置かれています)。



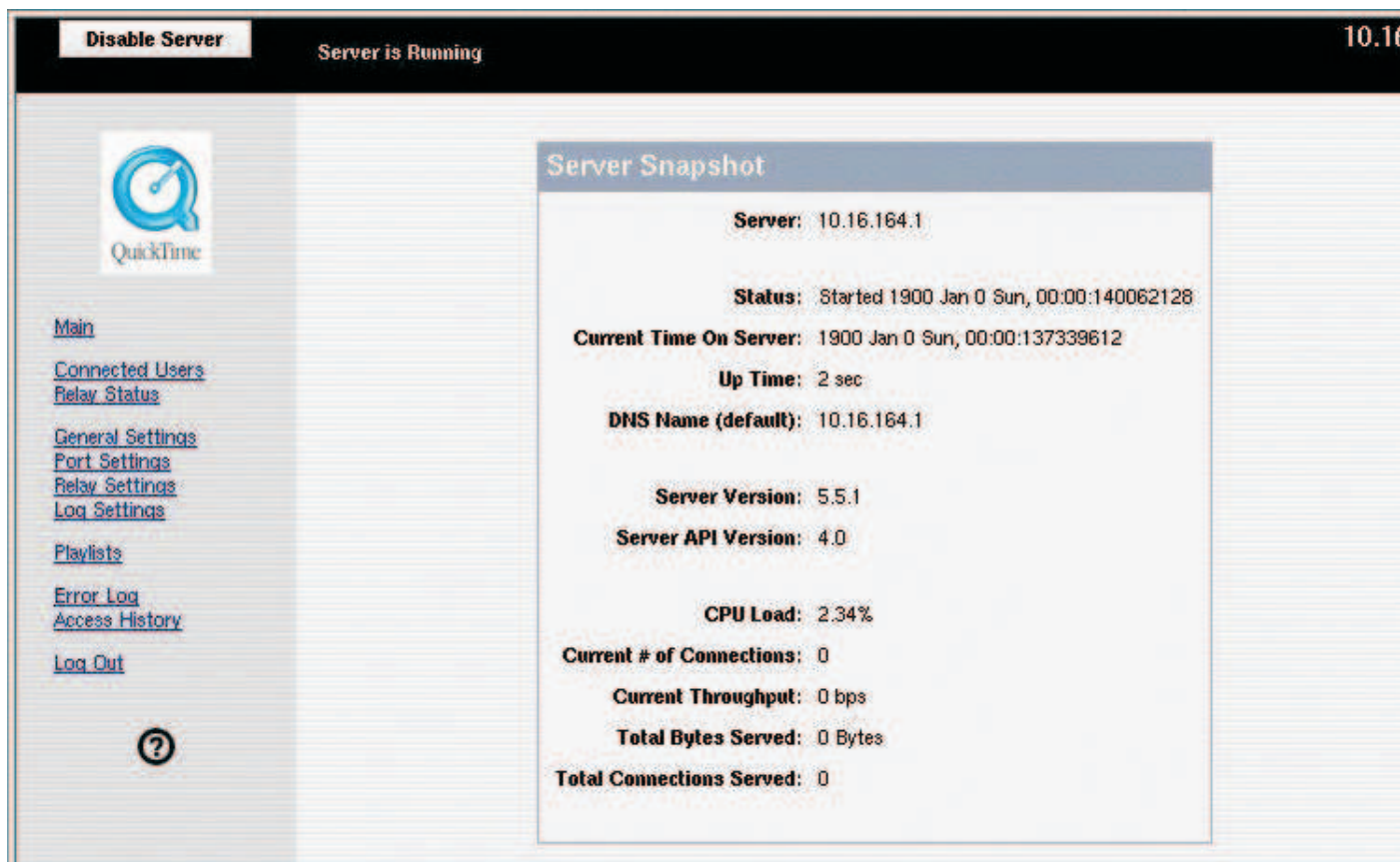
最後に、ストリーミングに利用するポート番号が聞かれます。通常のファイアーウォールでは80番は通すことが多いので、80番を選ぶと外部への映像の公開が少し楽になるからですが、多くの学校では内部はプライベートで運用されているために、プライベートアドレスの内部の映像サーバを外部に公開するには、少し特別な設定がファイアーウォールあるいはNATサーバに必要です。つまり、逆にそういう運用のサイトでは(それがほとんどでしょう)、あまり気にしなくても映像サーバが外部に漏れることはないということです。従って、ここでは80番を選んで、選ばなくても構わないことになりますが、この講習では80番で流すことにします。



- 参考

同じマシン上で DSS と Web サーバを動かす場合には、当然 DSS で 80 番を使うことは出来ません。

以上の初期設定が終わると、以下の画面になります。



2回目以降に、この URL にアクセスすると、上の画面に入れるようになります。このページから、管理者のパスワード変更や、その他の設定が行えるようになっています。

さて、これで基本の設定は出来ましたので、DSS が本当に動いているかどうかのテストをしましょう。

Win かあるいは Mac で、QuickTime を起動します (もし、なければ講習会場に設置した Windows マシンを利用してください)。

起動したら、QuickTime の「ファイル (F)」メニューの中の「新規 Player で URL を開く」を選び、現れたダイアログボックスに以下の URL を指定します。

```
rtsp://192.168.0.10/sample_300kbit.mov
```

ふわふわとした水球がぐるりと回って、QuickTime のロゴである「Q」が表示されたら、基本セットアップは無事終了です。

第3章 ストリーミング配信

3.1 ウェブサーバと配信サーバ

既に、サンプルの映像を見てみたように、ストリーミング配信をプレイヤーで再生するには、以下のよう URL をプレイヤーに指示すれば良い訳です。

```
rtsp://192.168.0.10/sample_300kbit.mov
```

しかし、こうした URL を手で入力するのは大変ですから、ウェブのページのリンクとして上記の URL を書いておき、それがクリックされたら適切なプログラムを呼び出して (つまりはプレイヤー)、実行するようになっているとユーザーから見れば、クリックだけで映像が始まるように見えます。

- 参考

rtsp は Real Time Streaming Protocol (RTSP) という、音声や動画などをリアルタイムに配信するためのプロトコル (手順) で、ファイルをダウンロードしながら再生するための仕組みです。FTP (File Transfer Protocol) が、ファイルをダウンロードするための仕組みであるのと同じように、ブラウザはプロトコルに応じたプログラムを起動するようになっています。

しかし、プレイヤーと言っても色々な種類があるために、どのプレイヤーがこの映像を再生するべきかが分かりませんので、直接上記の URL に飛ぶようにするのではなく、ヒントファイルと呼ばれるプレイヤーを特定するためのファイルをウェブサーバ上に置き、そのヒントファイルから上記の URL に飛ぶようにします。(詳しくは高谷先生の資料を見てください。)

従って、ストリーミング配信を行うためには、以下のファイルがそれぞれのサーバに必要です。(複雑なメタファイル指定は高谷先生の資料を参考にしてください。その場合には、ヒントファイルは不要です。)

ウェブサーバ

ヒントファイルへのリンクを記述したページ
ヒントファイル

配信サーバ

映像ファイル

3.2 映像ファイル

先のインストール時に指定したように、標準では /usr/local/movies というディレクトリの下に映像ファイルを置くように設定されています。勿論、ここに映像ファイルをどんどんアップしても良いのですが、多くのファイルがあると、分かりづらくなります。そうした整理のためには、主に2つの方法があります。

3.2.1 子ディレクトリを作る方法

子ディレクトリを作る方法は簡単です。

```
# cd /usr/local/movies
# mkdir test
```

これで、`/usr/local/movies/test/` というディレクトリが作成されますので、このディレクトリの下に映像ファイルを置きます。但し、この場合、この映像ファイル (例えば `sam.mov` とすると)、以下のようにヒントファイルに記述しなければなりません。

```
rtsp://192.168.0.10/test/sam.mov
```

つまり、配信サーバにとっては、`/usr/local/movies/` ディレクトリの下が配信対象で、その下の `test` ディレクトリにある `sam.mov` なので、`test/sam.mov` という指定に変更しなければならない訳です。

これで配信サーバとしては問題ないのですが、実際の運用を考えると問題があります。何故ならば、通常の設定では、このディレクトリの権限は `root` 以外は書き込めません。そして、`ftp` などを使ってファイルを配信サーバにアップする際には、セキュリティ的な観点から `root` でのアップロードは禁止されています。つまり、直接このディレクトリに `ftp` でファイルを置くことが出来ないのです。どうしても、このような管理を行いたい場合には、他のユーザにもこのディレクトリに書き込みが出来るように以下の設定を行います。

```
# chmod og+w /usr/local/movies
```

こうすれば、一般のユーザでこのディレクトリにファイルをアップロードすることが出来ます。

但し、この配信サーバのユーザ管理はきちんとしていないと、アカウントの持ち主は誰でもこのディレクトリにファイルが置けるので、ユーザ管理には注意する必要があります (変なファイルを置かれても困りますからね)。

このようにすることによって、配信ファイルを整理することは出来ますが、ファイルの書き込み権限の問題などを考えると、やはり管理的にはあまり好ましくありません。

3.2.2 別ディレクトリに格納する方法

`/usr/local/movies/` に直接映像ファイルを置くのではなく、別のディレクトリに置く方法があります。それは、シンボリックリンクと呼ばれるものを作る方法です。

- 参考

シンボリックリンクは、Windows ではショートカットと呼ばれているもので、実際のファイルやディレクトリがあたかもそこにあるように見せかける仕組みです。

以下では、`test` というユーザを作成し、その `test` のユーザのディレクトリに実際の映像ファイルを置くことを考えます。

そのために、まず `test` ユーザを作成します (実際の操作については、付録のユーザ管理を参照してください)。当然、このユーザのディレクトリに映像ファイルを置きたいので、パスワードを設定してください (グループは `staff` で良いでしょう)。

```
# /stand/sysinstall
```

ユーザとそのユーザのディレクトリが出来たら、以下のようにシンボリックリンクを作成します。

```
# ln -s /home/test /usr/local/movies/test
```

タイピングミスをしないように注意してください。うまく、シンボリックリンクが作成できたら、以下のようにになっている筈です。

```
# ls -l /usr/local/movies/test
lrwx-xr-x 1 root wheel 14 Oct 1 16:00 /usr/local/movies/test -> /home/test
```

3.3 映像配信の準備

さて、これで準備が整ったので、実際の環境に近い形で、映像配信の準備をしてみましょう。

まず、自分のサーバの IP アドレスを調べておいて、用意されている他の Windows マシンか、自分のノート PC などから、映像ファイルを配信サーバにアップロードします。(映像ファイルは指定されたものを使ってください。)

Windows に FFFTP などの FTP クライアントがあれば、それを使うと良いでしょう。その際に、配信サーバ(例えば、192.168.0.10)に入る際に使うユーザは、先に作った test ユーザを使用し、パスワードは自分で設定したものを使います。

test ユーザのディレクトリには、子ディレクトリは今のところありませんので、そのまま test 以下にアップして構いません。

次に、配信サーバに戻って、実際にファイルが存在するかどうかを確認して下さい。ここでは、アップしたファイルは sample.mov とします。

```
# cd /home/test
# ls -l sample.mov
-rwxr-xr-x 1 test staff 2000000 Oct 1 16:46 sample.mov
```

うまくアップ出来ていたら、今度はウェブサーバに上げるファイルを作成します。

Windows のノートパッドに近い asedit が導入されていますので、

```
# asedit &
```

これを上記のようにして利用するか、あるいは使ったことがある人ならば mozilla の composer を使っても構いません。

ウェブに上げるファイルは、最初はヒントファイルとそれへのリンクを含んだページの2つのファイルをアップしますので、それを別資料を参考に作成し、保存してください。

ここでは、index.html と test.qml とします(実際のファイル名は高谷先生の指示に従ってください)。

現在は、配信サーバ上でこれらのファイルを作成していますが、実際の環境では自分のノートや、職場の PC で作成し、ftp でアップすることになります。

実習環境では、Unix ですので、コマンドラインの ftp クライアントを使って、ウェブサーバにこれらのファイルをアップロードすることにします。

現在のディレクトリにファイルが以下のように存在することを確認してから、

```
# ls
index.html test.qml ...
```

以下のようにしてアップロードをします。

```
# ftp www
Connected to www.
220 www FTP server ready.
Name (192.168.0.10: root): movie
```

ユーザー名は movie です。

```
331 Password required for movie.
Password:
```

パスワードも movie です。
うまく入ると以下ようになります。

```
30 User movie logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

実際の環境では、適当なディレクトリの下にファイルを置き、他のページからリンクを張りますが、ここでは既にトップページからリンクが張られていますので、指定のディレクトリの下に index.html と sample.qtl を置くだけです。

指定ディレクトリは、実習環境では一人一人に名前でも指示しますので、その指定のディレクトリになります。例えば、ここでは kanayama であるとします。(決して、ディレクトリを間違えないで下さい)。

```
ftp> cd kanayama
250 CWD command successful.
```

kanayama のディレクトリへの移動がうまく成功すれば、上のように command successful と表示されますので、ファイルをアップロードします。

```
ftp> put index.html
...
226 Transfer complete.
200 bytes sent in 0.5 seconds ( 0.4 KB/s)
ftp> put sample.qtl
...
226 Transfer complete.
20 bytes sent in 0.1 seconds ( 0.2 KB/s)
ftp> quit
```

アップロードはこれで終了ですので、quit で最後に終了します。

さて、これでノート PC あるいは、用意されている Windows マシンのブラウザで自分の名前のリンクを辿って、index.html が表示されましたでしょうか？また、その中の映像へのリンクをクリックして、映像が無事配信されましたでしょうか？

もし、index.html が表示されるにも係わらず、映像が配信されないのでしたら、index.html とヒントファイル (sample.qtl) に問題がないかチェックして見て下さい。

また、映像配信サーバにアップした映像ファイルが Windows 上の QuickTime プレイヤーの URL 指定

で以下のようにして見る事が出来るかどうかを確かめてください。もし、映像が見れないならば、ファイルのディレクトリへの配置やシンボリックリンクの張り間違いなどが考えられます。

```
rtsp://192.168.0.10/test/sample.mov
```

講習は以上ですが、もし、時間があれば、メタファイルによる指定方法について同じようにトライしてみてください。

3.4 最後に

この講習での放送では、外部への放送は考えていません。勿論、単純に公開するだけならば、このようにして作成した配信サーバを公開するだけでも構いませんが、この講習の方法ですと ftp サービスなどが動いています。従って、ファイヤーウォールなどで ftp を通さないようにしておく必要があるでしょう。

また、校内での放送においても、部門別に管理したい場合には、部門ごとにユーザを分けて、ユーザのディレクトリに部門ごとの映像ファイルをアップするようになった方が管理上も混乱がないでしょう。

あまり Unix 上での管理に触れられませんでした。が、たくさんのユーザを管理するのでない限り、この講習程度での管理でも十分だと思われます。(最低、付録 B,C の基本としている部分のみを見れば、ユーザアカウントの管理、IP アドレスの設定は出来ます。)

但し、PC は PC ですので、ハード的な障害は常に考えられますので、映像ファイルのバックアップや、ウェブ上のファイルのバックアップは必ず取るようにしましょう。また、電源などを落としたい場合には、必ずシステムを止めてから落とさすようにしましょう (意外と問題なく働いているシステムは忘れられてしまいがちです)。Unix を落とす際には、付録にもあるように、shutdown コマンドを使います。

```
# shutdown -p now
```

最新の ATX 電源などではこれで自動的に電源が落ちますが、古い機械では落ちない場合もあります。そのような場合には、手動で電源をオフして下さい。

付録 A DSS 作成方法

ここでは簡単に FreeBSD でのソースコードからの DSS の作成方法について紹介します。

先のページからダウンロードしたファイル (DarwinStreamingSrvr5.5-Source.tar) を適当なディレクトリに置いて、以下のように解凍します。

```
% tar xvf DarwinStreamingSrvr5.5-Source.tar
```

すると、そのディレクトリに以下のように同名のディレクトリが作成されます。

```
% ls  
DarwinStreamingSrvr5.5-Source.tar DarwinStreamingSrvr5.5-Source/
```

そのディレクトリに移動します。

```
% cd DarwinStreamingSrvr5.5-Source/
```

ソースコードからバイナリファイルは以下のスクリプトを動かせば、自動的に生成してくれます。

```
% ./Buildit
```

FreeBSD では、バージョン 5.4 ではまだ不具合があるようで、うまく行きませんので、バージョン 4.11 を使って下さい。(Linux, Win, Mac, Solaris ではバイナリファイルが配布されているので、それを利用すると良いでしょう。)

付録B ユーザー管理

Unix では、一人一人のユーザーを区別することによって、ファイルの保護・実行プログラムの保護・システムの保護を実現している。従って、Unix 上のもは全て誰のものかという概念がある訳である。これらを管理することを広い意味でユーザー管理と呼ぶ。従って、ユーザー管理は root の仕事の第一歩となるのだが、システムをインストールした段階では、ユーザーはあらかじめ予約されたものしかない。例えば、root, sys, daemon などである。こうしたユーザーの名前をログインユーザー名と呼んでいるが、内部的には名前では区別するのではなく、番号で区別している。これをユーザー番号と言う。通常は、1桁のユーザー番号は管理者または特別なプログラムのために予約されている。一般のユーザーは、3桁以上の番号を用いるのが普通である。一般ユーザーが使用できるようにログインユーザー名、パスワード、ホームディレクトリ、初期設定ファイルなどを配布することをアカウントの発行と呼んでいる。

一人一人のユーザーを個別に管理すると大変なので、管理をしやすくするために、グループというユーザーの集団で扱う事が出来る。これも、名前をつける事が出来るが、実際には番号（グループ番号）で管理されている。ユーザーは通常、ある特定のグループに所属することになるが、`/etc/group` ファイルに個々のユーザーを追加することで、一人のユーザーが複数のグループに所属することが出来る。特に、管理者は Super User と言われるが、管理者が通常の作業を行う時に Super User として作業していると、誤ってファイルを削除した場合など被害が甚大である。そこで、通常は普通の権限しか持たない一般ユーザーとして登録しておき、必要な時にだけ Super User になる必要がある。こうした場合、管理者は、通常のグループに所属しておいて、同時に Super User (root) のグループに所属させておくと、`su` コマンドで、必要な時に Super User に変身が出来る。

B.1 ユーザー登録

Unix では、ユーザーの登録は、`/etc/passwd` にそのエントリを記述することによって行われる。FreeBSD では `sysinstall` や `adduser` コマンドを使って、`/etc/passwd` ファイルを管理することが出来る。前者では、CUI メニューによって、後者では質問に答えるだけで自動的に生成されるようになっている。

B.2 `sysinstall` でのユーザ登録 (基本)

`/stand/sysinstall` でのユーザの管理は立ち上げ直後の画面で **Configure** を選択し、

```

----- sysinstall Main Menu -----
Welcome to the FreeBSD installation and configuration tool. Please
select one of the options below by using the arrow keys or typing the
first character of the option name you're interested in. Invoke an
option with [SPACE] or [ENTER]. To exit, use [TAB] to move to Exit.

Usage      Quick start - How to use this menu system
Standard   Begin a standard installation (recommended)
Express    Begin a quick installation (for the impatient)
Custom     Begin a custom installation (for experts)
Configure  Do post-install configuration of FreeBSD
Doc        Installation instructions, README, etc.
Keymap     Select keyboard type
Options    View/Set various installation options
Fixit      Repair mode with CDROM/DVD/floppy or start shell
Upgrade    Upgrade an existing system
Load Config Load default install configuration
Index      Glossary of functions

[Select]  [X Exit Install]
----- [ Press F1 for Installation Guide ] -----

```

その後の User Management メニューから行う事が出来る。

```

----- FreeBSD Configuration Menu -----
If you've already installed FreeBSD, you may use this menu to customize
it somewhat to suit your particular configuration. Most importantly,
you can use the Packages utility to load extra "3rd party"
software not provided in the base distributions.

[X Exit      Exit this menu (returning to previous)
Distributions Install additional distribution sets
Packages     Install pre-packaged software for FreeBSD
Root Password Set the system manager's password
Fdisk       The disk Slice (PC-style partition) Editor
Label       The disk Label editor
User Management Add user and group information
Console     Customize system console behavior
Time Zone   Set which time zone you're in
Media       Change the installation media type
Mouse       Configure your mouse
Networking  Configure additional network services
v(+)

[OK]  [Cancel]
----- [ Press F1 for more information on these options ] -----

```

ユーザ管理メニューでは二つのメニューがあり、

```

----- User and group management -----
The submenus here allow to manipulate user groups and
login accounts.

[X Exit  Exit this menu (returning to previous)
U User   Add a new user to the system.
G Group  Add a new user group to the system.

[OK]  [Cancel]
----- [ Configure your user groups and users ] -----

```

グループ登録とユーザ登録が出来る。通常、先に必要なグループを作成し、その後にユーザ登録を行う。

The screenshot shows a dialog box titled "User and Group Management" with a sub-title "Add a new group". It contains the following fields:

Group name:	GID:
summer01	1001
Group members:	

At the bottom, there are two buttons: "OK" and "CANCEL".

ユーザ登録では、自動的に UID 番号などは振られるが、その他の項目は自分で入力しなければならない。特に、Group は先にグループ登録で作成したグループ名を入力する。Full name:は、後で説明するがいわゆる GCOS フィールドに使われるだけなので、識別出来れば良い。重要なのが、Member groups: である。基本で所属するグループ以外に、別のグループにも所属したい場合にのみ、ここに既存グループ名を指定する。特に、su コマンドなどで管理者が一般ユーザから、スーパーユーザになりたい場合には、ここに **wheel** と入力しておけば良い。Home directory:はデフォルトのままで良いだろう。一方、Login shell:はユーザの好みに変更すれば良い。FreeBSD では、C-shell は実際には tcshell と同じである。

The screenshot shows a dialog box titled "User and Group Management" with a sub-title "Add a new user". It contains the following fields:

Login ID:	UID:	Group:	Password:
test	8001	mmer01	*****
Full name:	Member groups:		
test user	wheel		
Home directory:	Login shell:		
/home/test	/bin/tcsh		

At the bottom, there are two buttons: "OK" and "CANCEL".

B.3 アカウントの発行

`adduser` コマンドを `root` でオプション `-q` を付けて実行します。

```
# adduser -q
```

1. 注意

オプション-qをつけないと、最初にデフォルト設定の変更に関する質問になってしまいます。慌てずに、コントロール C (Ctrl+C) で抜け出すか、単純に以下のように最初の5つの質問にはリターンをしておけば問題はありません。

```
Check /etc/shells
Check /etc/master.passwd
Check /etc/group
Enter your default shell: csh date no sh [sh]:
Your default shell is: sh -> /bin/sh
Enter your default HOME partition: [/home]:
Copy dotfiles from: /usr/share/skel no [/usr/share/skel]:
Send message from file: /etc/adduser.message no
[/etc/adduser.message]:
Use passwords (y/n) [y]:

Ok, let's go.
Don't worry about mistakes. I will give you the chance later to correct any input.
```

adduser -q コマンドを実行すると、最初にユーザのログイン名を設定します。ここはなるべく分かりやすい名前を入力した方が良いが、シェルで解釈される文字などは避けた方が良い。また、大文字小文字は区別されるのと、8文字までである点に注意する。

```
se option '-verbose' if you want to see more warnings and questions
or try to repair bugs.
```

```
Enter username [^[a-z0-9_][a-z0-9_-]*$]: kanayama
```

次に、フルネームを入れる。これはシステム自体には関係がないので、空白などを使っても良い。

```
Enter full name []: Noriyo Kanayama
```

次は、ユーザの使うシェルの設定だが、最初は sh,csh(実は tcsh),tcsh が使える。取り敢えずは tcsh にしておく。

```
Enter shell bash csh date no sh tcsh [sh]: tcsh
```

次は、ホームの設定であるが、自動的に /home/[ユーザ名] となるので、通常はそのままリターンすれば良いだろう。

```
Enter home directory (full path) [/home/kanayayama]:
```

次にあるのは、ユーザ番号で、これは一人一人完全に違う番号を割り振らなければならないのだが、adduser コマンドでは、自動的に 1000 以降の空いている番号を振ってくるので、リターンを押しておけば良い。また、その次の login class もリターンを押しておけば良い。

```
Uid [1000]:
Enter login class: default []:
```

ここは、ユーザの所属するグループの指定だが、デフォルトではユーザ名と同じグループ名を入れてくるので、ここは必ず設定が必要である。管理者が自分のユーザアカウントを発行する場合、staff グループに所属させ、更に、wheel グループにも所属をさせておくと su が出来るようになるので便利であるが、同時にセキュリティホールにもなるので管理者のパスワードの管理には十分注意しなければならない。(Unix では、ユーザは複数のグループに所属出来るので、このような設定がある)。一方、一般ユーザーのグループは予め設定されていなければならないので、一般ユーザーの登録の前に次のグループの登録を行っておかなければならない。

```
Login group kanay [kanayama]: staff
Login group is ‘‘staff’’. Invite kanay into other groups: guest no [no]: wheel
```

最後に、パスワードの設定である。これは表示されないなので、注意深く設定しなければならない。2回聞いてくるので、間違えないように。

1. 参考

なお、登録後は passwd コマンドで変更は出来る。

```
例
# passwd kanayama
Changing local password for noriyo.
New password:
```

ちなみに、パスワードに用いられる暗号は FreeBSD ではデフォルトでは MD5 を使い、DES を導入すれば両者の暗号を使い分けることが出来る。NIS などを用いて他のシステムとパスワードを共有する際には、互換性の問題のために DES を用いなければならない。一方、MD5 を用いれば 16 文字のパスワード (DES では 8 文字まで) を用いる事が出来るので、root パスワードなどには極力 16 文字パスワードを用い、大小英文字・記号・数字が入った文字列を利用するようにしなければならない。

```
Enter password []:
Enter password again []:
```

全てが終わると、最後に以下の確認画面が出てくる。

```
Name:      kanayama
Password:  ****
Fullname:  Noriyo Kanayama
Uid:      1000
Gid:      20 (staff)
Class:
Groups:    staff wheel
HOME:     /home/kanayama
Shell:    /bin/tcsh
OK? (y/n) [y]:
```

ここで、入力が間違っていることに気が付いたら、n を入力し、最初からやり直す。間違っていなければ、y を入力する。すると、ユーザのアカウントを発行し、他のユーザアカウントの発行をしたいかどうかを聞いてくるので、なければ n を入力し、終了しよう。

```

Added user ‘kanayama’
Send message to ‘kanayama’ and: no root second_mail_address [no]:

Noriyo Kanayama,

your account ‘kanayama’ was created.
Have fun!

See also chpass(1), finger(1), passwd(1)

Add anything to default message (y/n) [n]:
Send message (y/n) [y]:
Copy files from /usr/share/skel to /home/kanayama
Add another user? (y/n) [y]: n
Goodbye!

```

もしも、ユーザーを間違えて発行してしまった場合や、既存のユーザを削除したい場合には、`rmuser` コマンドを用いる。

```

# rmuser kanayama
Matching password entry:

kanayama:xpe1zRdj0lAhE:1000:20::0:0:Noriyo Kanayama:/home/kanayama:/bin/tcsh

Is this the entry you wish to remove?

```

ユーザの指定に間違いがなければ、`y` を入力する。すると、ホームディレクトリも消去するかどうかを聞いてくるが、そのユーザのファイルなども消去することになるので、ユーザがバックアップをきちんと取ったかどうかなどを確認してから消去する必要がある。ここで、`n` を指定し、ディレクトリを消去しなくても、ユーザのアカウントは上の操作で抹消されているので、通常は問題ない。

```
Remove user's home directory (/home/kanayama)?
```

後で、手動で上記のディレクトリを消去したい場合には、以下のように行う (但し、これをするとは完全に消去されるので、ディレクトリ名に間違いがないかなどを確認した上で慎重に行う必要がある。)

```
# rm -rf /home/kanayama
```

B.3.1 /etc/passwd ファイルの例

ユーザの管理 (登録、抹消、変更) などに関しては、`adduser`, `rmuser`, `passwd` コマンドで十分であるが、以下では実際にアカウントが登録された場合に用いられる、登録情報ファイルである `passwd` ファイルについてもう少し解説を行う。

`/etc/passwd` の例

```
user1:x:100:10::0:0:hogehoge user1:/home/user1:/bin/csh
```

- 第1フィールド
ログイン・ユーザー名
- 第2フィールド
パスワード。シャドウパスワードを使っている時は、`x`。
- 第3フィールド
ユーザー番号。各人の区別はこの番号で行う。重複してはならない。
- 第4フィールド
グループ番号。あるユーザーの集合、グループ番号とグループ名の対応は`/etc/group`に書く。
- 第5フィールド
クラス（使われていない）。
- 第6フィールド
パスワードを変えることを勧告する時期
- 第7フィールド
パスワードの有効期限
- 第8フィールド
GECOS フィールド。ユーザーの一般情報（名前など）
- 第9フィールド
ホームディレクトリ
- 第10フィールド
ログインシェル

`/etc/passwd` ファイルは、各々のユーザーのプログラムから参照出来る必要があるが、暗号化されたパスワードファイルを盗まれると解読される恐れがあるので、FreeBSD では、ルートにしか読めないファイル `/etc/master.passwd` にパスワードを書きしておく仕組みになっている（これをシャドウパスワードと言い、最近の Unix では標準化されているが、ファイル名はシステムにより違う）。`/etc/master.passwd` ファイルは手でいじる必要は殆どなく、`passwd` コマンドを使えば自動的にそのパスワードを `/etc/master.passwd` に登録してくれる。また、`/etc/passwd` ファイルの編集には、`vipw` コマンドを使う。このコマンドを使うと、編集中は、他のルートが変更できないようにロックをかけ、編集終了後はデータベースファイルを更新してくれる。

ちなみに、暗号システムに何を利用しているかは `/etc/master.passwd` の第二フィールドの先頭に `1` が入っているかどうかを判別すれば良い。`1`が入っている場合には MD5 が使われ、そうでない場合には DES が使われている。

B.4 グループ登録

グループのリストは `/etc/group` に登録されていなければならない。`/etc/passwd` ファイルと同じく、グループ名はグループ番号で管理される。従って、システム内部で重要なのはグループ番号 (groupID) で

あり、名前自体ではない。このことは、グループ番号の衝突は深刻な事態を引き起こす事を意味している。ユーザーが所属している以外のグループに所属するためには、この `/etc/group` ファイルへの登録が必要である。こうした複数のグループへの所属は一般に、ファイルのパーミッションやプロセスのパーミッションと深く関係しているが、ファイルのパーミッションについては後の節で説明することにする。

グループファイルの書式は次の通りである。

```
wheel:*:0:root
```

1. 第1フィールド
グループ名
2. 第2フィールド
パスワード（使われる事はほとんどなく、*を入れておけば良い）
3. 第3フィールド
グループ番号
4. 第4フィールド
このグループに所属するユーザー、複数いるときはカンマ (,) で区切る。`/etc/passwd` で所属している以外のグループに所属したい時はここに書く。

例えば、`wheel` はルートの所属するグループで、ユーザー `root` が所属しているが、ある一般ユーザーが管理者である場合、そのユーザー名をここに登録すれば、その一般ユーザーは `su` コマンドを使うことによって必要な時にルートになる事が出来る。

```
wheel:*:0:root,kanayama
```

通常、`/etc/group` ファイルは `vi` などのエディタで編集しても良いし、`/stand/sysinstall` のメニューからも追加出来る。

一般ユーザーのアカウントの発行に際しては、どういうグループにユーザーを分けるかを考えなければならない。これは特に、グループ内部でファイルの読み書きを可能とするかどうかによって分割を考えれば良い。

B.5 ホームディレクトリと配布ファイル

`vipw` などを用いてアカウントを発行した場合などは、ユーザーのホームディレクトリ等を準備し、必要なファイルをコピーしなければならず、更に、そのディレクトリ及びファイルの所有者を該当するユーザーに変更しなければならない（その上、グループ属性も）。この際に良く用いられるコマンドが、`chown`、`chmod`、`chgrp` コマンドである。`chmod` コマンドは後述するとして、残りの2つのコマンドについて簡単に紹介する。

chown ファイル・ディレクトリの所有者（オーナー）を変更する

例えば、`test` ディレクトリを `kanayama` の所有に変更するには、

```
# chown kanayama test
```

通常、ディレクトリ以下を変更する機会が多いので、その場合には、`-R` オプションを付けることでディレクトリ以下を再帰的に変更する。

```
# chown -R kanayama test
```

chgrp ファイル・ディレクトリのグループを変更する例えば、test ディレクトリを staff の所有に変更するには、

```
# chgrp staff test
```

chgrp コマンドも、chown と同じく -R オプションが同じ意味で用いることが出来る。

こうしたコマンドによるディレクトリの用意は、シェルスクリプトなどで一気にユーザーディレクトリを準備する場合に重宝するが、最近のシステムではアカウントの発行・ディレクトリの準備・ファイルの配布まで行うコマンドが完備されるようになってきている。FreeBSD でも、先に紹介した **adduser** コマンドを使えば配布まで自動的に行ってくれる。配布ファイルは、`/usr/share/skel/` に置いておく事になっているので、標準設定を変更したり、新たに設定ファイルを配りたい場合にはここに置いておくようすれば良いのだが（勿論、既存のユーザーには手で配布しなければならない）、その際にドットファイルは `.cshrc` ではなく、`dot.cshrc` のような名前でなければならない。基本的なファイルは既に `/usr/share/skel` に準備されているので、サイトごとに変更を加える場合にはこれらに変更・追加をすれば良い。

```
# ls /usr/share/skel
dot.cshrc          dot.mail_aliases  dot.rhosts
dot.login          dot.mailrc        dot.shrc
dot.login_conf    dot.profile
```

B.6 ファイルのパーミッション

ファイルに対して Unix では所有者やグループがあることを先に述べたが、こうした所有者やグループがあることの目的は、それぞれのファイルについてアクセス制御が可能になっていることである。このアクセス権は、3つのユーザークラスに対して独立に設定できるようになっている。ユーザークラスは、

- u user(所有者) ファイルやディレクトリの所有者
- g group(グループ) 所有者と同じグループに所属するユーザ
- o other(他人) 上記以外のユーザ

に分かれており、所有者は自分と同じグループのユーザにはファイルに対して読み書きの権限を与え、他人に対しては読み込みのみの権限を与えるようなことが可能になっている。

アクセス権には通常次の3つが独立に設定出来る。

- r read 読み込み可能
- w write 書き込み可能
- x eXecute 実行可能（ディレクトリに対しては cd 可能）

通常、こうしたファイルのアクセス権の状態は、`ls -l` によって表示される。

```
-rw----- 1 kanayama staff 280 Jul 12 20:54 .rhosts
-rw-r--r-- 1 kanayama staff 856 Jul 12 20:54 .shrc
-rw-rw-r-- 1 kanayama staff 0 Jul 14 17:34 test
```

この表示の第一フィールドに表示されている文字列がアクセス権の状態を表し、全部で10文字の長さを

持っており、最初の一文字目はディレクトリあるいはファイル種別（- は通常ファイル、l はリンク、c・b は特殊なデバイスに関するファイル）を表す他は3文字ずつの集合が3回表れ、最初の3文字は所有者に対する、次の3文字はグループに、最後の3文字は他人に対する状態を表し、それぞれの3文字は read, write, execute の状態を表示している。

これらのアクセス権の設定に用いられるコマンドが `chmod` である。通常使う際の書式は以下の通りである。

```
chmod [-R] [-v] mode file[or directory]
```

ここで、-R オプションは `chown`, `chgrp` などと同じく再帰動作をする。-v オプションは設定されたファイル名を表示する。mode の書式には2通りあり、一つは先のクラスとアクセス権の記号を使って設定する方法であり、今ひとつはオクテット（octet - oct は8を意味する）で直接アクセス権のビットを変更する方法である。前者の場合、一般に mode は以下のように書く。

```
ユーザクラス+アクセス権
```

従って、group に対して、書き込み権限 (w) を与えるときの mode の指定は、

```
g+w
```

となる。クラス、アクセス権は複数と同時に指定可能で、other, group に対して読み込み (r) と実行 (x) の権限を与える場合には、

```
og+rx
```

のように指定する。従って、test について上記の設定をする場合は、

```
# chmod og+rx test
```

となる。ここで、mode の指定では空白を入れてはならない。

次にオクテット指定の場合には、mode は3桁の数字で指定し、一つの数字が一つのクラスに対するアクセス権の設定に対応し、3桁の数字は左から順に所有者、グループ、他人に対する設定に対応する。数字は1,2,4がそれぞれ実行、書き込み、読み込みに対応し、それらの数の和で状態が表される。以下に対応関係を掲げる。

記号表記	オクテット表記	計算式
rwX	7	1+2+4
rw	6	2+4
rx	5	1+4
r	4	4
xw	3	1+2
w	2	2
x	1	1

これらの指定が3つのクラスに対して指定するので、例えば `u+rwX` で、`g+rx`、`o+r` の場合は、

```
# chmod 754 test
```

のように指定することになる。

実はこれらの他に更に特殊なモードがあり、それぞれ SetUID, SetGID, Sticky ビットと呼ばれるモードがある。特に重要なものが SetUID ビットで、これは実行ファイルに対して用いられると、ファイルを実行した人の権限で実行するのではなく、ファイルの所有者の権限で実行する事が出来るようになる。つまり、root 所有の実行ファイルに対してこれが設定されていると、一般ユーザであっても root の権限で動いてしまう訳である。従って、すぐに分かるように SetUID ビットはシステムのセキュリティにとって非常に危険な仕組みであるので、注意して管理する必要がある。

これらの特殊ビットを設定する際には、オクテット指定を用い、それぞれ以下のオクテットを使う。

- **SetUID** 4000
- **SetGID** 2000
- **Sticky** 1000

実際には、以下のように指定する。

```
# chmod 4555 /bin/mailq
# ls -l /bin/mailq
-r-sr-xr-x 5 root wheel 290448 Jun 22 12:13 /usr/bin/mailq
```


付録C 環境設定

Unix では、様々なマシン環境の設定やアプリケーション環境の設定は設定ファイルを通じて行うようになっている。すなわち、専用ツールを使うのではなく、エディタを利用して設定ファイルを編集することで環境構築がなされるために非常に柔軟な管理が可能になっている。ここでは、FreeBSD における主な環境設定ファイルとその役割について学ぶ。

C.1 sysinstall での基本設定方法 (基本)

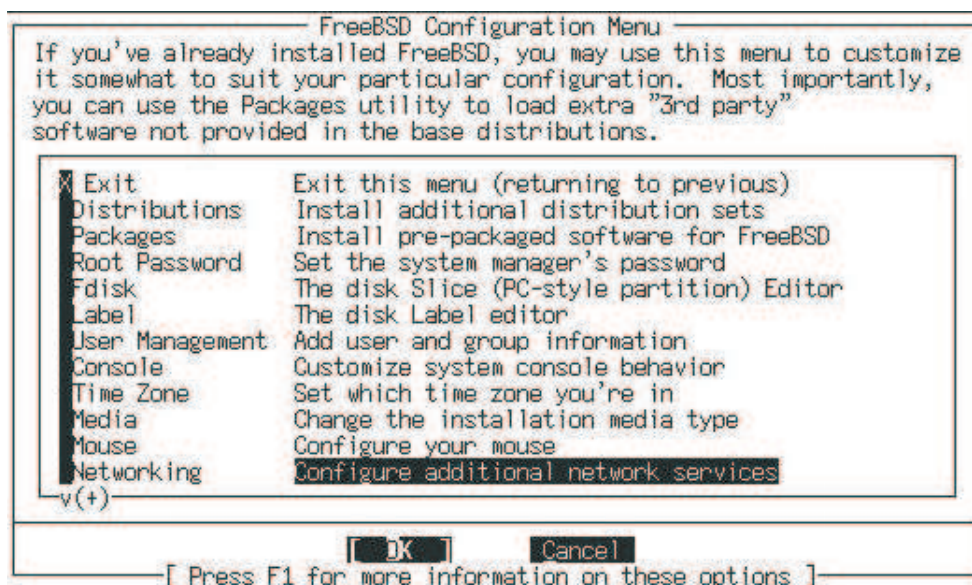
先にユーザー登録で利用した、`/stand/sysinstall` を使ってある程度は基本的な設定が出来るようになっているので、初心者では最初は `sysinstall` を使うのが良いだろう。

その中でも、実際の利用で最低必要なものが IP アドレスの設定である。クライアントの PC などでは、IP アドレスは DHCP を使って自動的に設定するのが通常であるが、サーバなどは常に固定した IP アドレスを利用したいので (DHCP でも DHCP サーバはなるべく同じ IP を出すように努力をしますが、完全ではありません)、設定が必要になります。

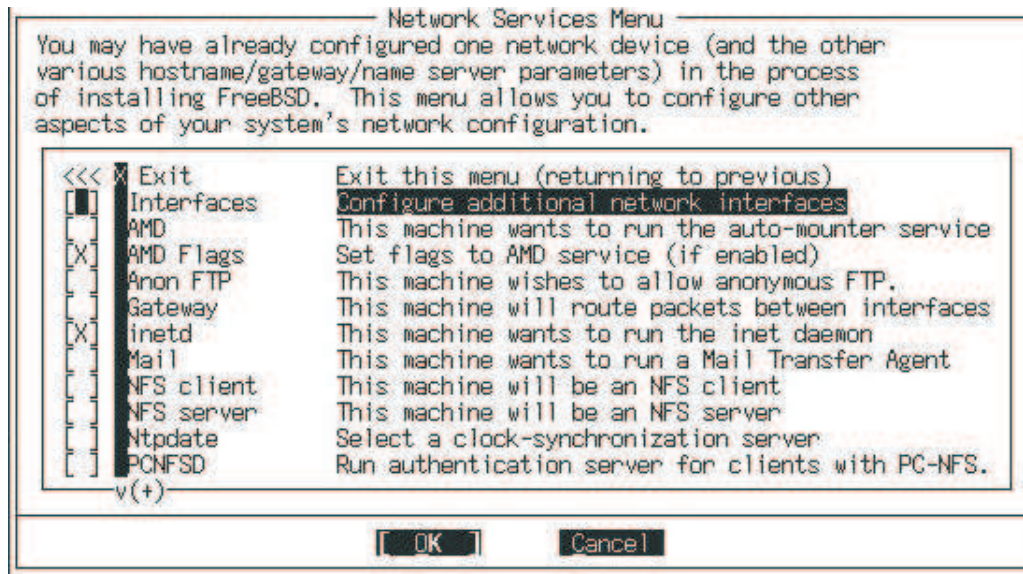
以下では、`sysinstall` を利用した IP アドレスの設定を紹介します。

```
# /stand/sysinstall
```

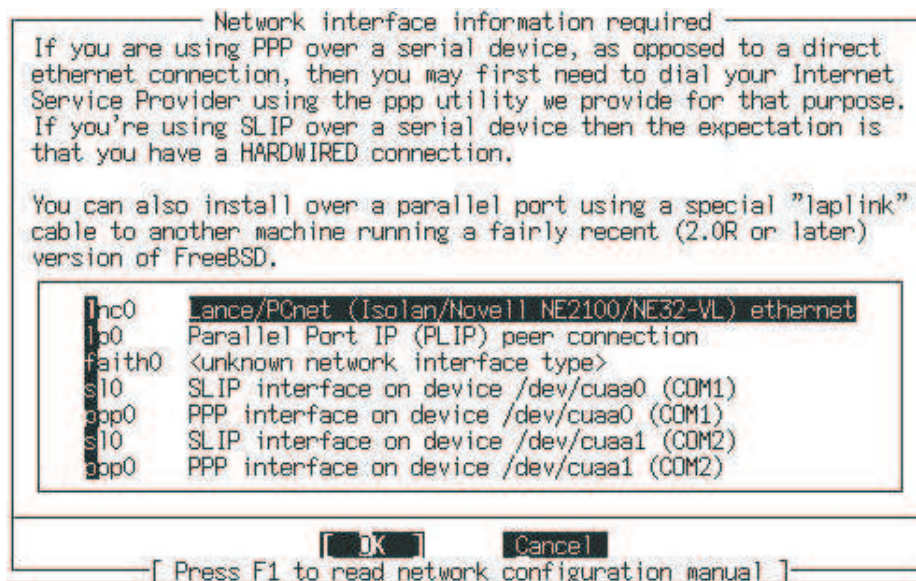
`configure` メニューに入ると以下の画面となり、その中の `Networking` を選択する。



次に、以下の画面の `Interfaces` をスペースキーで選択をします。



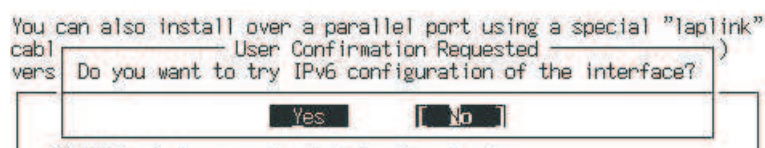
すると、以下のインターフェースの選択画面になります。これは、IP アドレスはマシンのそれぞれのインターフェースに対して与えるものであるため、インターフェースを選ぶ必要があるからです。以下の例では、lnc0 を選んでいます。講習会では xl0 になります。



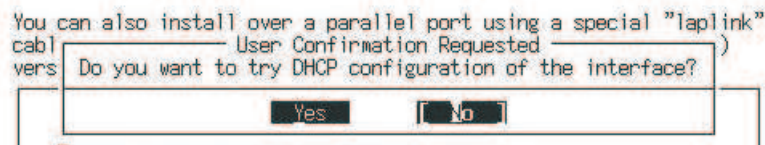
インターフェースがどのような名前かは、以下の URL から調べられます。

<http://www.jp.freebsd.org/www.FreeBSD.org/ja/releases/4.11R/hardware-i386.html#ETHERNET>

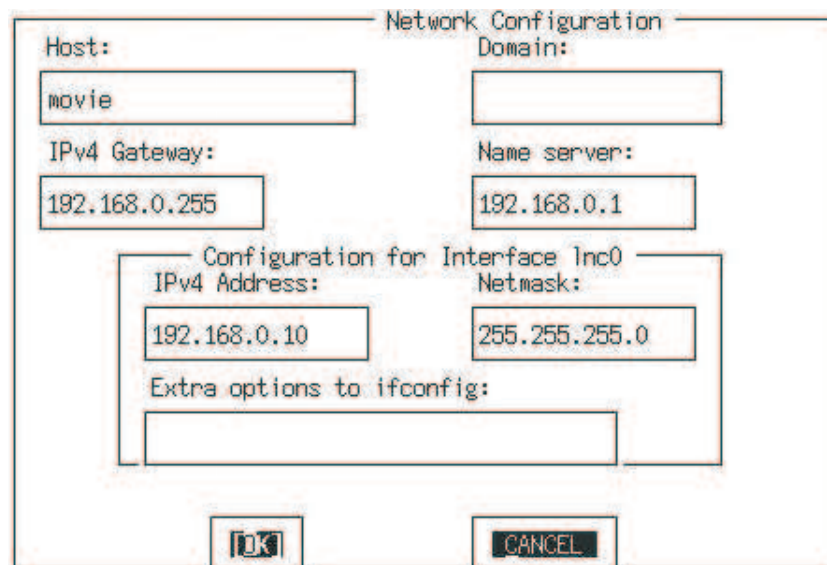
インターフェースを選ぶと、下のように IPv6 を利用するか否かを聞いてきます。通常は、NO で良いでしょう。



次に、DHCP を使うか否かを聞いてきます。今の場合は手動で IP を設定したいので、NO を選びます。



最終的に、以下の IP アドレス設定画面になります。ここで最低限必要な項目は、ホストの名前、通常使うゲートウェイ (ルータの IP アドレス)、ホストの IP アドレス、Name Server(DNS サーバ) です。



ゲートウェイや DNS サーバは Windows が利用しているものと同じものを利用して構いません。(Windows では、コマンドプロンプトの中で、ipconfig /all と入力すると表示されます。)

```
>ipconfig /all
...
Ethernet adapter ローカル エリア接続 :

    Connection-specific DNS Suffix  . : wakhok.ac.jp
    Description . . . . . : Broadcom NetXtreme Gigabit Ethernet
    Physical Address. . . . . : 00-50-45-00-9A-42
    Dhcp Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes
    IP Address. . . . . : 192.168.0.48
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.0.254
    DHCP Server . . . . . : 192.168.0.200
    DNS Servers . . . . . : 192.168.0.200
```

Unix 上でこれを調べるには、一度 DHCP サーバを動かす必要があります。

```
# dhclient x10
...
# ifconfig x10
```

ここで、x10 は先のインターフェース名です。DNS サーバは、

```
# cat /etc/resolv.conf
search wakhok.ac.jp
nameserver 192.168.0.200
```

で分かります。ゲートウェイは、

```
# netstat -rn |grep default
default      192.168.0.254    UGSc    2      0      x10
```

で分かります。

以上のようにして、情報が判明したら、それに応じて先の `sysinstall` の画面で入力を行い、OK を選択した後は、EXIT を使って `sysinstall` を抜ければ良いでしょう。但し、設定は一旦マシンをリブートする必要があります。Unix ではリブートは、以下のコマンドで出来ます。

```
# reboot
```

C.2 起動と rc ファイル

Unix ではシステムの起動は、ブートシーケンスがカーネルを読み込み、カーネルが `init` プログラムを起動する。起動された `init` プログラムが各種設定ファイルを起動し、システムが立ち上がるようになっていく。設定ファイルは、SystemV 系と BSD 系で少し違い、一般に SystemV 系では `/etc/rc?.d/` ディレクトリ下の `Sxx` (`xx` は数字) を順に実行することで立ち上がる (`Sxx` は主にシェルスクリプトで、実際には `/etc/init.d/` へのリンクが多い)。一方、BSD 系では伝統的に `/etc/rc` や `/etc/rc.local` などのスクリプトファイルを起動するようになっていくが、FreeBSD 等ではこれらの `rc` ファイルから更に `/etc/rc.i386/`, `/etc/rc.pccard/`, `/etc/rc.network` などのスクリプトを起動するように拡張されている。こうした起動スクリプトの分離は、アプリケーションを自動的に組み込むために考え出された非常に簡単な方法であり、FreeBSD では `/usr/local/etc/rc.d/` と `/usr/X11R6/etc/rc.d` ディレクトリ下のスクリプトを最後に起動するようになっていく。

FreeBSD ではこうした起動ファイルの設定を更に簡便に行うために、起動スクリプトの環境変数を別のファイルにまとめている。つまり、起動スクリプトは最初にこの環境変数を定義したファイルを読み込むようになっている訳である。更に、こうした環境変数のデフォルト値を反映するようにするために、最初に `/etc/defaults/rc.conf` を読み込み、その後に `/etc/rc.conf` を読み込むようになっている。このことによって、ユーザーが独自に加えた変更は `/etc/rc.conf` に書き込むことによって、環境変数が上書きされ、システムに反映される訳で、必要のない (デフォルトのままが良い) 設定は何もしなくとも `/etc/defaults/rc.conf` の設定が生きるようになっていくのである。従って、ユーザーは `/etc/defaults/rc.conf` の中身を修正してはならず、必ず `/etc/rc.conf` に対して行わなければならない。しかし、実際に設定を行うに当たってはデフォルト値を知る必要があるため、`/etc/defaults/rc.conf` を参照はしなければならない。

もっとも単純な `/etc/rc.conf` の例を下に掲げる。

```
# This file now contains just the overrides from /etc/defaults/rc.conf
# please make all changes to this file.

# -- sysinstall generated deltas -- #
ifconfig_xl0="inet 192.168.0.10 netmask 255.255.255.0"
hostname="test.wakhok.ac.jp"
keymap="jp.106"
```

最初の環境変数は、後に出てくる `ifconfig` コマンドに渡すオプションで、ネットワークインターフェースボードの IP アドレスとネットマスクを設定している。ちなみに、`xl0` はインターフェースボードの種類で、この場合は少し古いが 3Com の有名な 100Mbps のカードである。自分が利用しているインターフェースボードの名前が知りたい場合には、`dmesg` コマンドでシステムメッセージを見ると良い。

```
> dmesg | more
Copyright (c) 1992-1999 FreeBSD Inc.
Copyright (c) 1982, 1986, 1989, 1991, 1993
    The Regents of the University of California. All rights reserved.
FreeBSD 3.5-RELEASE #0: Thu Jun 22 04:05:16 GMT 2000
    root@monster.osd.bsdi.com:/usr/src/sys/compile/GENERIC
Timecounter "i8254" frequency 1193182 Hz
CPU: Pentium III/Pentium III Xeon/Celeron (551.25-MHz 686-class CPU)
    Origin = "GenuineIntel" Id = 0x673 Stepping = 3
    Features=0x383f9ff<FPU,VME,DE,PSE,TSC,MSR,PAE,MCE,CX8,SEP,
        MTRR,PGE,MCA,CMOV,PAT,PSE36,MMX,FXSR,<b25>>
real memory = 134217728 (131072K bytes)
    .....
    .....
xl0: <3Com 3c905B-TX Fast Etherlink XL> port 0x1400-0x147f mem 0x40a00000-0x40a0
007f irq 11 at device 4.0 on pci0
```

2 番目の環境変数は自ホストの名前の定義であるが、これは後述の `/etc/hosts` における名前と混同してはいけない。通常は、自ホスト名と自分のデフォルトインターフェース名とは一致させておくようにする（これについては後でもう一度述べる）。

3 番目の環境変数はキーボードの種別の定義である。もし、これをインストール時に定義しておらず、日本語 106 キーを利用している場合、US101 キーボードとしてしか認識せず記号などの入力が難しくなるので、設定が必要である（設定後はリブートが必要）。

C.3 IP アドレスと名前解決

IP アドレスは数字であるので、ユーザーが記憶するには面倒であり、同時に管理者にとっては IP アドレスの移動をユーザーに通知して回るのは大変である。また、世界的にも IP アドレスで通信を全て陽に行わなければならないとしたら大混乱となるであろう。このように、ネットワーク上の通信では IP アドレスが基本ではあるのだが、それらを隠蔽する必要がある。これは一般に名前解決と呼ばれ、ホストの名前から IP アドレスを自動的に検索するようになっている。世界的な規模でこれを行うのが DNS(Domain Name Service) であり、ローカルに行うものとしては NIS(Network Information Service) や NIS+などがある。

一方、マシン独自にこうした名前解決を行うための設定ファイルとして `/etc/hosts` がある。このように、複数の名前解決のための手段があることは便利ではあるが、一方で混乱が生じる原因ともなりかねないので、管理者は矛盾が生じないように注意してこれらの設定を行わなければならない。一般に、`/etc/hosts` による解決は必要最低限に留めるべきである。

C.4 ディスク管理

C.4.1 ファイルシステムとマウント

Unix では全てのディスクやネットワーク上の記憶資源は一つのツリー構造の中に展開される。ツリー構造は `/` を頂点として、枝分かれしており、それぞれのディスク上のファイルシステムは全てこのツリー構造の中のどこかの枝の中に組み込まれるようになっている。逆に、ディスク資源を利用可能にするためには、ツリー構造への組み込み作業が必要であり、これをマウントと言う。

現在のディスクの容量・空き容量やマウント位置は `df` コマンドによって知る事が出来る。

```
> df
Filesystem      1K-blocks      Used    Avail   Capacity  Mounted on
/dev/ad0s1a      99183        20439   70810    22%       /
/dev/ad0s1f    3523142      377806  2863485   12%      /usr
/dev/ad0s1e    1984479        2371  1823350    0%      /var
/dev/ad0s1g    2927638     1976446  716982    73%     /home
procfs           4             4         0    100%    /proc
```

ここで、第一フィールドの `Filesystem` はそれぞれのディスクのパーティションを表している。Unix ではディスク資源自体の管理もファイルシステムの中に組み込まれており、そうした資源は全て `/dev/ディレクトリ` で管理される。それぞれの資源の区別は、FreeBSD では `ad` が IDE のハードディスクを意味し、次の `0` は IDE コントローラの番号、`s1` がスライスの一番号 (DOS ではパーティションと呼ぶので混乱しやすいが)、`a,f,e` の英文字がそれぞれのパーティションを表している。BSD 系では、通常、`a` パーティションは `/` に、`b` パーティションをプロセスのスワップ領域にとり、`c` パーティションはディスク全体を意味し、それ以降を通常のファイルシステムに取るようになっている (`procfs` はプロセスファイルシステムという特別なもの)。

この例では、それぞれのファイルシステムは簡単に `/` 以下のディレクトリにマウントされ (マウントポイントという)、例えば `ad0s1f` は `/usr` にマウントされている。

どのディスクのどのスライスをどこにマウントするかは、`/etc/fstab` に記述され、起動時にその情報に従ってマウントが行われる。

```
# Device      Mountpoint  FStype  Options  Dump  Pass#
/dev/ad0s1b   none        swap    sw       0     0
/dev/ad0s1a   /           ufs     rw       1     1
/dev/ad0s1f   /usr        ufs     rw       2     2
/dev/ad0s1e   /var        ufs     rw       2     2
/dev/ad0s1g   /home       ufs     rw       2     2
/dev/acd0c    /cdrom      cd9660  ro,noauto 0     0
proc          /proc       procfs  rw       0     0
```

FreeBSD では、これらのファイルはインストール時に自動的に生成されるので自分で作成する必要は

ない。また、ディスクの増設時であっても、`/stand/sysinstall` の `Configure` メニューにある `Fdisk` と `Label` メニューから半自動的に行う事が出来るので、管理者が考えなければならないのは、マウントポイントとパーティションの分け方だけである。もし、他のFreeBSDマシンで用いていたディスクを増設する場合であっても、`Label` メニューにおいてマウントポイントを指定するだけで良いので、ディスクの管理は比較的簡単である。

C.4.2 ファイルシステムの異常

もし稼働中のファイルシステムに異常があり、コンソールにそうしたエラーが出たならばハードディスクの異常を疑うべきであるが、システムが稼働中に停電などで落ちた場合ファイルシステムに矛盾が生じることがある。通常は、リブート時に自動的に修復がなされるが、自動修復が出来ない場合には自動的にシングルユーザーモードに落ちて、管理者の操作を待つ様になっている。そうした場合には、異常のあるファイルシステムに対して、手動で `fsck` による矛盾の修正を行わなければならない。例えば、`/dev/ad0s1g` に異常があったとすると、

```
$ fsck /dev/rad0s1g
```

と操作する必要がある。(ここでデバイス名が `/dev/rad0s1g` と `r` が `ad0s1g` の前に付いているのは、`raw device` の意味)

この場合、修正が必要な場合には管理者に `yes, no` で問い合わせてくることになる。`fsck` を `-p` オプション付きで実行した場合、軽微な修正は自動的に行うようになるが、このオプションで修正出来ない異常がある場合にはファイル損傷の可能性がある。一方、`-y` オプション付きで実行すると、全ての修正を問い合わせることなく実行することになるので、先のような損傷があった場合ファイルは失われてしまうので注意が必要である(最終手段だという事)。

付録D FreeBSDの第一歩

D.1 FreeBSDのインストール

FreeBSDのインストールの詳細については多くの書籍が出されているので、ここでは簡単に述べることにする。

D.1.1 機器の選定

まず、マシンを準備する必要がある。一般に市販されているものが全て利用可能かどうかは組み込まれているVGAカードや、ディスク、ネットワークカードの問題があり、一概には言えないが、簡単に触れることにする。詳しくは、<http://www.jp.freebsd.org/>に掲載されているインストール対象のリリースのリリースノートを参照して貰いたい。

PC/AT互換機が基本であるが、PC98にも移植されている。とは言え、PC98は避けた方が得策であろう。

CPU

CPUは、Intel系ならばほとんどがサポートされている。但し、386SXはFPUの関係で推奨されていない。それ以外の486,Pentium系は全て動作し、互換CPUもハードウェアレベルでの互換性を持つものならば、ほとんど動作する。

バス

ISA, EISA, PCIバスがサポートされているが、PCIが無難であろう。

ディスク

ハードディスクは通常IDEが用いられることが多く、最新のE-IDEならばまったく問題はない。ファイルサーバなどに利用する場合にはSCSIも良い選択となる。何故ならば、多数のハードディスクを増設した場合には、IDEでは数の制限があるのと、パフォーマンスの問題があるからである。一般には、SCSIボードは個別にサポートされているので、先のリリースノートをよく読み、利用しているカードがサポート対象かどうかを良く確認しなければならない。また、ハードウェアRAIDカードも一部サポートされているので、クリティカルな業務ではそうしたものの利用も検討した方が良い。

ディスプレイカード

FreeBSDを走らせるだけならば、VGAカードは何でも良い。しかし、Xウィンドウを利用する必要があるならば、XFree86でサポートしているカードでなければならない。一般に最新のカードはすぐにはサ

ポートされない場合が多く、メジャーなカードを利用した方が良いであろう (Nvidia あるいは ATI などが良いだろう)。

ネットワークカード

最近では、ほとんど 100M Ether が普通であるが、Gigabit Ether もいくつかのカードに対してサポートされている。チップセットごとにサポートされているので、リリースノートを参照されたい。メジャーな所では、Intel, 3Com, 旧 DEC チップなどが上げられるが、安い RealTek, VIA チップなども利用可能である (但し、サーバ用途を考えるならば、これらのチップは止めた方が良く、通常は Intel チップが使いやすい)。

CD-ROM

最近の ATAPI CD-ROM は全てサポートされている。SCSI CD-ROM の場合には、SCSI ボードがサポートされていれば問題ない。一方、古い CD-ROM はサポートされていない独自のものがあるので注意が必要だが、よほど古くない限り問題ないであろう。(DVD-ROM も ATAPI モードで使えるものは利用出来る)

D.1.2 インストール方法の選択

インストール方法は、FDD インストール、DOS インストール、CD-ROM インストール (CD-ROM ブート可能なマシンと、CD-ROM が必要) などがある。通常、CD-ROM インストールならば、配布ファイルは全て CD-ROM に入っているのだから、CD-ROM のみで全て足りるが、FDD インストールの場合は、どこから配布ファイルを取ってくるのかによっていくつかの方法が選択できる。この場合、CD-ROM、FTP、DOS パーティション、NFS などを通じて配布ファイルをインストールすることができる。一方、DOS インストールでは DOS から `setup.bat` を動かすことでインストールが出来るが、Windows などの DOS プロンプトからは出来ず、本当の DOS からでないといインストール出来ない。また、DOS からのインストールという制約上、配布ファイルは DOS からアクセス出来るデバイス上になければならないが、この条件さえクリアすれば FDD なしでもインストールが可能であるので、B5 ノートなどでは重宝するであろう。

D.1.3 フロッピーの準備

先に述べたように様々なインストール方法があるが、ここではフロッピーディスクを用いることにする。海外では 2.88M の FDD が普及しているのだから、FreeBSD ではそれを標準的にサポートしているが、日本ではあまり普及していないようなので、1.44M フロッピーが 2 枚必要である。まず、DOS 上でフォーマットした上で、`fdimage.exe` や `rawrite.exe` などを用いて、`kern.flp` と `mfsroot.flp` をフロッピーに入れる。この flp 形式のファイルは、FreeBSD のファイルシステムを納めたものなので、`copy` コマンドなどでは作成出来ない点に注意してほしい。ちなみに、これらのファイルはリリースディレクトリの下 `floppies/` ディレクトリにある。一方、上記のツールは通常、FreeBSD の `tools/` ディレクトリに用意されている。(但し、ノートにインストールする際には、ACPI などの関係で FreeBSD 5.x の系統の方が成功率が高いようである。)

```
> fdimage kern.flp a:
...
> fdimage mfsroot.flp a:
```

また、フロッピーに不良セクタなどが存在すると、完全なフロッピーとならないので注意が必要である。既に、FreeBSD などの Unix マシンが存在する場合には、`dd` コマンドなどでもブートフロッピーは作成可能であるが、上記の点には注意する必要がある。

D.1.4 インストールの開始

まず、`kern.flp` を入れてブートし、ビープ音がなったら `mfsroot.flp` のフロッピーに入れ換える。ほどなくするとインストール用のシステムが立ち上がり、まず最初にデバイスの選択をするためのモードになるので、Visual モードを選ぶ。Visual モードでは、様々なデバイスの検索や IRQ などの設定が画面から出来るようになっているが、ISA カードを使っていない限りほとんど関係がない。起動を早くしたいならば、記憶デバイスやネットワークカードは削除しておけば良い。但し、IDE と FDD は削除してはならない。PCI デバイスは自動設定なので、削除はしなくても良い。

D.1.5 インストールメニュー

インストールのためのメインメニューからインストールの全ての設定が行なえる。初心者は Norvis モード（初心者モード）で入れるのが最も簡単であるが、慣れれば Expert モードで導入できるようになった方が良い。というのは、ディスクのパーティションの切り方などではデフォルトではサーバ用途の場合をあまり考慮していないからである。いずれにせよ、Options や Keyboard は最初に設定しておいた方が間違いない。特に、Options で、デフォルトではエディタに `ee` が設定されているが、サーバの管理者はやはり伝統的な `vi` を利用したほうが良いであろう。

Expert モードで設定する場合重要なことは、各スライスの割り当て方で、特に各種のログなどが記録される `/var` などは思い切って大きく取った方が良い。また、導入するシステムの選択では `kernel developer` は入れて置かなければならない（切り方が分からない場合には、最初に `a` を押すと、`auto` で割り当てるのでそれを参考に切り直せば良いだろう）。

インストールが終了したら、最低 `root` のパスワード設定やマシンの IP アドレスの設定などはしておく必要があるだろう。

D.1.6 root のパスワードを設定し忘れたら

`root` パスワードをインストール時に設定し忘れた場合は、`root` のパスワードなしで設定されるので非常に危険である。ブートしたら直に `passwd` コマンドを用いて設定するようにならなければならない。ちなみに、パスワードを設定したが、忘れてしまったりした場合は、あわてずにブートしなおして（`Ctrl+Alt+Delete` キーでリブートがかかる）、9秒間の秒読みが終わる前にリターンキー以外のキーを叩いて、プロンプトに対して次のように入力することでシングルモードと呼ばれるモードに入る。

```
> boot -s
```

シングルモードでは、最初にシェルを選択を尋ねられるが、そのままリターンを押せばデフォルトの `Born shell` となる。

シングルモードに入った当初では、スライスは全てマウントされておらず、さらに/ スライスもリードオンリーになっているので、書き込みが出来るようにマウントのし直しをする。

```
# mount -a
```

その後、passwd コマンドで root のパスワードを設定しなおせば良い。

```
# passwd root
...
```

設定が終われば、↑D を押すか、あるいは exit コマンドを実行してシングルモードを抜けると、自動的にマルチユーザーモードでシステムが立ち上がる。

D.2 インストール後の設定と変更

インストールが終了し、無事ハードディスクから立ち上がったならば、その後の設定の変更のほとんどは /stand/sysinstall から行う事が出来る。実際、インストールの際に立ち上がったインストールメニューは、この /stand/sysinstall そのものであったのである。

sysinstall の Configure メニューからユーザーアカウントの管理、グループ管理、ディスク管理など、ほとんどの基本的な管理は行うことが出来るようになっているが、実際には対応するコマンドを裏で実行しているだけであり、サーバの管理を理解するためにも最初は sysinstall を用いても、徐々にコマンドでの管理手法に慣れるようにしなければならない。一方、インストール時に指定し忘れたディストリビューションファイルの追加や後に述べるパッケージの追加は sysinstall から行った方が良くであろう。

sysinstall を使う際に注意は、終了後表示が出来なくなる場合があり、その時には慌てずに stty sane と入力すると元に戻る（このコマンドを実行するまでは表示が出来ないので、確実に入力すること）。

D.3 パッケージの追加

FreeBSD にはパッケージと呼ばれる様々なアプリケーションのバイナリを簡単にインストールするための仕組みが用意されている。これらのうち、バイナリファイルが用意されているものに関しては、/stand/sysinstall から全て導入することが出来る。FreeBSD4.3R からは、sysinstall も多くのオプションを持つようになり、予め様々なオプションの設定などを行えるようになっている。

コマンドレベルでは、pkg_add, pkg_info, pkg_delete, pkg_create, pkg_update(4.3R より) などが用意されており、sysinstall のパッケージの追加ではこれらのコマンドを裏で実行している訳である。とは言え pkg_add は使わなくとも sysinstall を使えば良いのであるが、その他の3つのコマンドについては知っておく必要がある。一方、pkg_update は既に導入したパッケージのアップデートをするためのコマンドで、パッケージの依存関係をも表示するのでシステムをアップデートする際に役立つであろう。

D.3.1 pkg_info

既にインストールされているパッケージの情報を見る。パッケージの名前を指定せずに、-a オプションをつけると全てのパッケージの情報を表示する。

```
pkg_info [[-a] | [-v] [パッケージの名前]]
```

パッケージではパッケージ名という名前を用いるために、ここで指定する名前が違っているとまったく情報が得られない。そこで、通常良く使うのが、grep と pkg_info の -v の組み合わせである。

```
# pkg_info -a | grep samba
And this also supports 'samba cap'ed japanese filename.
Information for ja-samba-2.0.5a.2:
# pkg_info -v ja-samba-2.0.5a.2 | more
...
Packing list:
  Package name: ja-samba-2.0.5a.2
  CWD to /usr/local
File: man/man1/nmblookup.1.gz
  Comment: MD5:77ab6507dc1f4681325988462bd15882
File: man/man1/smbstatus.1.gz
...
```

例えば Samba を導入したが、パッケージ名が分からないために詳細情報を知ることが出来ないといったケースで、上のように入力することで、パッケージ名が ja-samba-2.0.5a.2 であることが検索出来た訳である。次に、pkg_info の -v オプションを使う事によってどのファイルがどこにインストールされたかが分かるようになった。こうしたパッケージ情報の検索は FreeBSD で設定を行う場合基本となる操作である。

なお、最近の pkg_info コマンドには正規表現による表示もサポートされるようになったので、上記のような grep を用いなくても良くなりつつある。

```
pkg_info -x [正規表現]
```

もっとも正規表現を用いなくても、部分文字列でも一致するので、簡単に考えて良い。

```
# pkg_info -Ix gtk
gtk-1.2.10_2      Gimp Toolkit for X11 GUI
gtk-1.2.8        General Toolkit for X11 GUI
gtk-engines-0.10 Theme engine for gtk+ toolkits
gtkdiff-1.0.2    A frontend program for diff(1) written for GNOME
gtkhtml-0.12.0   Lightweight HTML rendering/printing/editing engine
gtkhtml-0.7      Lightweight HTML rendering/printing/editing engine
gtkmm-1.2.3_1    C++ wrapper for gtk, a x11 graphics library
gtkmm-1.2.7      C++ wrapper for gtk, a x11 graphics library
gtkportscan-1.2  A simple port scanner that prints out open ports
linux-gtk-1.2    RPM of the Gtk lib
py-gtk-0.6.6     A set of Python bindings for GTK
rep-gtk-0.14_1   GTK+ binding for rep Lisp interpreter
```

この例では、gtk がパッケージ名の部分文字列として含まれているパッケージを表示する。ちなみに、-I オプションは一行インデックスをのみ表示するオプションであり、-x ではこれを使わないと、非常に長い説明が全て表示される。

D.3.2 pkg_delete

導入したパッケージを削除する場合に使用するのが `pkg_delete` である。パッケージでは、別のパッケージを必要とする場合、これを依存関係と言い、インストール時に自動的に依存関係にあるパッケージも導入してくれるのだが、削除時にはそのパッケージが依存関係を持つか否かをチェックするようになっているために、他から必要とされるパッケージを誤って削除することがないようにしている (-f を付けると強制的に削除するので注意が必要)。

```
pkg_delete [-v] [-d] パッケージ名
```

他にも多くのオプションがあるが、主に良く使うのは、

-v 詳細表示

-d 空ディレクトリの削除

であろう。

D.3.3 pkg_update

パッケージをアップデートするために用い、依存関係も報告してくれる。

```
pkg_update [-nv] 新しいパッケージ名
```

-n 実行は行わず、何が行われるかを表示

-v 詳細な表示

例えば次のように使う。

```
pkg_update -v ja-Canna-3.3.tgz
```

依存関係から表示されるものは更に導入をしなければならない。

D.4 ソースからの追加

パッケージからの追加がバイナリがなくて出来なくても、FreeBSD には `ports` という仕組みがあって、半自動的にソースからバイナリの生成、インストールを行えるようになっている。実はパッケージもソースも全て `CVS` というバージョン管理ツールによって管理されており、FreeBSD のカーネルソースコードを始めとして、ほとんど全てのソースが世界的に一貫した方法で管理されている。そのうち `ports` については、`/usr/ports` と呼ばれるディレクトリにバイナリを生成するために必要なファイルが導入されている(なければ `Distribution` から `ports` を追加して入れれば良い)。ここには、全てのバイナリのソースがある訳ではなく、ソースをどこから落して来て、どういうパッチを当てて、どうやってコンパイルするか、他のバイナリとどういう依存関係にあるかと言った情報が格納されている。

`/usr/ports` ディレクトリの下はカテゴリー別にディレクトリが分かれ、その下にソフトウェア毎にディレクトリがあるので、`root` 権限でそれらのディレクトリに移り、`make` を実行すればバイナリが生成される。

```
# cd /usr/ports/net/bind9
# make
```

もし、ソースがなければ自動的にダウンロードして来てくれる。

1. 注意

ソースはデフォルトの場所から落すので、遅い場合もある。これを防ぐには、オリジナルのソースを直近のサイトから落して来て、`/usr/ports/distfiles` にそのまま入れておけば良い。ファイル名などが不明な場合には、`distinfo` ファイルでも確認出来る (本来このファイルの役割は、正しいファイルかどうかをチェックするためにあるのだが)。ここに目的のファイルがあると、既に落しているものと見なされ、次の動作に移るようになっている (但し、ものによっては `distfiles` の下に更にディレクトリをほって、そこに落してくるものもある。この場合も、`distinfo` を見れば分かる)。

最後に、バイナリがうまく `make` 出来た事を確認したら、

```
# make install
```

を実行すれば良い。

ちなみに、ports は刻々とアップデートがされており、最新のものを手に入れるには `cvsup` などのツールを用いてアップデートをする事が出来る。(`cvs,cvsup` などもパッケージに入っている。)

D.5 システムの停止とリブート

稼働しているシステムを止める場合には、可能な限りユーザーに知らせる事が必要であり、やむを得ない場合を除き、それに適したコマンドを使用するようにしなければならない。通常、システムを停止する場合に用いられるコマンドは、`shutdown`, `halt` で、`halt` はディスクキャッシュの掃き出しと実行中プロセスへの通知と停止のみしか行わないので、稼働しているシステムでは出来るだけ `shutdown` コマンドを用いるようにする。一方、リブートの場合には `shutdown` コマンドは対応しているが、`halt` はリブート目的には利用できないので代わりに `reboot` コマンドを用いる。(FreeBSD では、`fastboot`, `fasthalt` コマンドがあるが、これらは `reboot`, `halt` コマンドと同じものである。)

これらのシステムの停止やリブートは正規に行われる限り、ログに記録される。デフォルトでは、`/var/log/messages` がログファイルとして扱われる。

D.5.1 shutdown

`shutdown` コマンドは、指定時刻にシステムを停止したり、リブートを行う事ができ、同時にログインユーザーや NFS ユーザーへの通知を行う。一般的な書式は以下の通りで、

```
shutdown [-] [-h | -p | -r | -k] time [warning-message ...]
```

それぞれのオプションの意味は、

-h halt する

-p halt し、可能であれば power off する (BIOS が対応している場合)

-r reboot する

-k ユーザーを追い出すが、システムは停止せずマルチユーザーシステムのまま

- ユーザーへの警告メッセージを標準入力から読み取る

であり、`time` の書式には次の2通りがある。

- +N N は整数で、N 分後に shutdown を行う
- [yymmdd]hhmm yymmdd は年・月・日を指定するが省略すると現在時。hhmm は時 (24 時)・分を指定する。

特に、後者の形式で年月日を省略した場合で、現在時刻以前の時間を指定すると、その時刻は過ぎているというメッセージと共に無視されるようになっている。従って、翌日以降の時間にシャットダウンをした場合には、少なくとも日付を指定しなければならない (yymm は現在時のものが補われる)。

最後に、ログインユーザーへの警告メッセージを指定することが出来るが、省略した場合にはシャットダウン時刻がユーザーに通知される。

ちなみに警告は 10 時間前からユーザーに通知され始め、シャットダウン時が近づくにつれ頻繁に警告が出るようになっている。特に、停止の 5 分前か、あるいは 5 分以内の停止の場合には、即座に警告メッセージが出るとともに、新規ログインは出来なくなるようになっている。

実際の利用方法を掲げる。

```
shutdown -p +10
```

この場合、10 分後にシステムを落とし、電源を切る。

```
shutdown -r 2100
```

21 時 0 分にリポートがかかる。

```
shutdown -k 100910
```

現在月の 10 日 9 時 10 分に全ユーザーを追い出す。

D.5.2 halt(reboot)

halt はシステムの停止に用いられ、reboot はリブートに用いる。オプションは、halt に対して -p オプションのみが意味があり、shutdown コマンドと同じく電源断まで可能であるならば行う。通常、halt は shutdown コマンドが利用できないような緊急停止時にのみ用いられる。また、FreeBSD の halt コマンドはメモリ中のディスクキャッシュの書き出し動作を自動的に行うようになっているが、その他の Unix システムでも同じであることを期待してはいけない。従って、習慣的には次のように強制的なキャッシュの書き出しのためのコマンド sync と組み合わせて用いる。

```
sync;sync;sync;halt
```

ここで sync を 3 回指定しているのは、Unix 管理者の太古の知恵で、通常は 2 回で良いとされているところを安全を考えて 3 回にしているのである。

halt をした上で、再立ち上げを行うためには reboot コマンドを用いるが、これはユーザーがログインしていない事が確実な場合で、メンテナンスの最中などに良く用いられる。