

Solarisセミナー

2003年8月27日

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** PC/AT互換機の作成
- 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 PC/AT互換機の作成

1.1 部品構成について

CPU:	Intel Celeron 2GHz FSB ^{注1} 、クロック倍率、SMP ^{注2}
Mother Board:	(Intel 845 chipset) チップセット(440BX, 810, 815, 820, 845, 850, 865)
Memory:	256MB (PC2100 CL=2.5, DDR SDRAM) メモリの重要性
Video Card:	matrox G450 X Window System ^{注3} との相性
Ethernet Card:	Intel PRO/100+ 10base-T, 100base-TX, Gigabit
Hard Disc:	Seagate ST340014A 40.8GB ATA/133 IDE(ATA/33, 66, 100, 133)とSCSI(Fast, Wide, UltraWide U2Wide, Ultra160, Ultra320)、RAID ^{注4}
CD-ROM:	x54 ATAPI
Floppy Disc:	3.5" 2 mode
Case:	ATX Middle Tower, 300W 電源の重要性(リダンダント)、ケース内の整理とファンの使用、UPS ^{注5} の使用、置く部屋

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。

1.2.2 キーボード

キーボードはPS/2接続のもの、お好みでJIS配列、ASCII配列のどちらでも使用可。設定はkdmconfigで行う。PC/AT互換機ではキーボードを差していないと電源が入らないので注意。

1.2.3 マウス

マウスはPS/2接続のもので2つボタンや3つボタンのものが使用可。Xを使用するならば3つボタンのものが便利。設定はkdmconfigで行う。

注1 FSB [Front Side Bus]

注2 SMP [Symmetric Multiple Processor]

注3 X Window System <<http://www.xfree86.org/>>

注4 RAID [Redundant Arrays of Independent (Inexpensive) Disks]

注5 UPS [Uninterruptable Power Supply]

1.3 組み立て

1.3.1 注意事項

静電気に注意。ケース内の電源部分などに触り静電気を逃がしてから各部品を取り扱う。

1.3.2 部品を確認する

1.3.3 組み立て作業

ケースにCD-ROM、フロッピー、ハードディスクを取り付ける。インチねじとミリねじに注意。マ
ハードディスクとCD-ROMのジャンパはケーブルセレクトに設定する。

マザーボードにCPUと専用ファン、メモリを取り付ける。

ケースにスタッドを取り付ける。

マザーボードをケースに取り付ける。ねじを強く締めすぎない様に注意。

ケースのATXケーブルと各種ケーブルをマザーボードに差し込む。+-を確認。

AGPビデオカードを差し、ねじで固定する。

モニターとキーボードをつなぎ立ち上がるかの確認。

>> 電源が入るか。

>> 電源のLEDは点灯するか。

>> BIOS^{注6} が見えるか。

>> BIOSでCPUとメモリを確認。確認したら電源を切り、電源ケーブルを抜く事。

CD-ROM、フロッピー、ハードディスクを専用ケーブルで接続する。

PCIネットワークカードを差し、ねじで固定する。

電源を入れてBIOSで全て動いているか確認。IRQ^{注7} が重複していないか確認する。

BIOSで時刻と起動優先順位の設定。

Windowsをフロッピーで起動させ、フロッピーが稼動するか確認する。

FDISKを使い、ハードディスクのLEDが点灯するか確認。

OSをCD-ROMから起動させ、CD-ROMが稼動するか確認する。

ケース内の整理

>> インシュロックでケーブルを整理し風通しを良くする。

蓋をして完成。

注6 BIOS [Basic Input Output System]

注7 IRQ [Interrupt ReQuest]

2 Solarisのインストール

2.1 インストール Solaris 8 Intel 2/02

2.1.1 ブート

BIOSでCD-ROMからの起動を設定し、インストールCD-ROMを入れ電源を入れる。自動的にインストーラーが起動する。

2.1.2 Solaris Device Configuration Assistant

"F2"で進む。

2.1.3 Bus Enumeration

"Enter"で進む。

2.1.4 Identified Devices

"F2"で進む。

2.1.5 Boot Solaris

"CD"をスペースキーで選択し、F2で進む。

言語の選択を要求されるので"6"でJapaneseを選択する。

2.1.6 The Solaris Installation Program

"F2"で進む。

2.1.7 kdmconfig - Introduction

"F2"で進む。画面の解像度、キーボード、マウスの設定をインストール終了後変更する場合はコンソールよりこの"kdmconfig"を起動する。

2.1.8 kdmconfig - View and Edit Window System Configuration

ここで自動選択されたハードウェアの確認をし、修正をしたい場合は修正する個所をスペースキーで選択し修正する。修正がすんだら"F2"で進む。(このメニューにはコンソールで"kdmconfig"とすることにより使用可となる)

2.1.9 kdmconfig Window System Configuration Test

画面のテストモードに"F2"で進む。

"Is this display okay"と出たらマウスで"Yes"を選択する。この後はGUIによるインストールとなる。

2.1.10 FDISK

fdiskを実行して、Solaris fdiskパーティションを作成
"y"で進む

Type "y" to accept the default partition, other wiser type "n" to edit the partition table.
"y"で進む。

swapスライスの指定はデフォルトの512MBのまま"Enter"で進む。

2.1.11 SWAP

ディスクの先頭からswapスライスを作成しますか?
"y"で進む。

2.1.12 Mini Root

警告の後、これでよろしいですか?
"y"で進む。

ミニルートへのコピーが始まる。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。

2.1.13 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。

2.1.14 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。

2.1.15 DHCP

DHCPを使用:"いいえ"を選択し、"次へ"で進む。

2.1.16 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。

2.1.17 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。

2.1.18 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。

2.1.19 IPv6

IPv6を使用:"いいえ"を選択し、"次へ"で進む。

2.1.20 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。

2.1.21 Kerberos

Kerberosを有効にする:"いいえ"で進む。

2.1.22 ネームサービス

ネームサービス:"DNS"を選択し、"次へ"で進む。

2.1.23 ドメイン名

ドメイン名:"beer.jp"と記入し、"次へ"で進む。

2.1.24 DNSサーバのアドレス

サーバのIPアドレス:"192.168.0. "と記入し、"次へ"で進む。

2.1.25 DNS検索一覧

ドメインの検索: 記入せずそのまま進む。

2.1.26 時間帯

時間帯の指定:"地域"を選択し、"次へ"で進む。

2.1.27 地域

"アジア東部"から"日本"を選択し、"次へ"で進む。

2.1.28 日付と時刻

現在の日時に設定し、"次へ"で進む。

2.1.29 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。

2.1.30 プロキシサーバ構成

"インターネットに直接接続"を選択し、"次へ"で進む。

2.1.31 情報の確認

入力値を確認し、問題なければ"確認"で進む。

2.1.32 ようこそ

"次へ"で進む。

2.1.33 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。

2.1.34 媒体の指定

媒体:"CD/DVD"を選択し、"次へ"で進む。

2.1.35 ディスクの挿入

指定のディスクをセットし、"了解"で進む。

2.1.36 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。

2.1.37 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。

2.1.38 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。

2.1.39 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、Documentation Asianを選択し、"次へ"で進む。

2.1.40 追加製品の指定

"なし"を選択し、"次へ"で進む。

2.1.41 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。

2.1.42 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。

2.1.43 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。

2.1.44 データの保存

"いいえ"で進む。

2.1.45 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。

まず、"/export/home"の値を5000MBほど減らし、その分を他に割り当てる。"/"を200MB、"/usr"を4000MB、"/var"を3000MB、"/opt"を3000MB、"/tmp"を1000MBと変更し、残りをすべて"/export/home"に割り当てる。変更後、"次へ"で進む。

/	300MB
/usr	5000MB
/var	5000MB
/opt	5000MB
/tmp	1000MB
/export/home	残りすべて

2.1.46 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ICD-ROMをセットする。

2.1.47 リポート

すべてが終了し継続するために"リポートする"を選択する。

2.1.48 ログイン

リポート後、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8}

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。値段は安い。

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。

3.4 専用線サービス

OCNエコノミーなど128kbps、64kbpsでの専用線を使ったインターネット接続。月々の使用料は3万円前後と回線の太さから考えると非常に割高である。ただし利用できる地域が広い。

3.5 FTTH^{注9}

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。

注8 ADSL [Asymmetric Digital Subscriber Line]

注9 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と一般ユーザの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"hotel"とし設定する。

```
# groupadd hotel
```

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, /usr/bin/zsh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 hotel -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=$PATH:/usr/local/bin
PATH=$PATH:/opt/sfw/bin
export LD_LIBRARY_PATH=/usr/local/lib
export EDITOR=vi
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 ~/.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
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
# 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 Intel向けのもは"SunOS 5.8_x86"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_x86_Recommended.zip
# cd /export/home/user/download/8_x86_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
# 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 基本的なコマンド

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/
       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。
% chown -R grape strawberry
      strawberryディレクトリとその中のすべてのファイルのグループを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 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。

kill プロセスを停止させる。

```
% kill 239 プロセスID239を強制的に終了させる。
```

```
% kill -HUP 3988
```

プロセスID3988をハングアップさせる。

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。

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。

```
% /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。

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 192.168.0.1
```

IPアドレスを指定しての確認。

```
% ping -s 192.168.0.1
```

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 -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、パスワードに自分のメールアドレスを入力する。

```
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/>>

5.1.4 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
```

```
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
```

6 SSHの設定

6.1 OpenSSH

6.1.1 SSH^{注10} 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/test/download/zlib-1.1.4.tar.gz
# cd zlib-1.1.4/
# ./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/test/download/openssl-0.9.7b.tar.gz
# cd openssl-0.9.7b/
# ./config
# 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 sshd
# useradd -g sshd -s /bin/true -d /dev/null sshd

# cd /usr/local/src/
# tar xvfz /export/home/test/download/openssh-3.6.1p2.tar.gz
# cd openssh-3.6.1p2/
# ./configure
# gmake
# gmake install
```

注10 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
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

まず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/で不要なサービスを止める

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で特定のサービスを停止する

特定のプロセスのみ停止したい場合はinetd.confで指定のサービスwo#で無効にする。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。

```
# cd /etc/inet/
# 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

inetdを再起動する。

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

6.2.4 inetdが起動しないようにする

必要無ければinetdが起動できないように変更する。

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

/usr/sbin/inetd -s &	これを
#/usr/sbin/inetd -s &	この様に

現在使われているinetdを止める。

```
# ps -ef | grep inetd   inetdのPIDを調べる。
# kill PID
```

7

DNSの設定

7.1 BIND^{注11} <<http://www.isc.org/products/BIND/>>

7.1.1 BINDのインストール

Solaris 8 2/02にはあらかじめBIND(8.2.2)がインストールされているが、BINDはセキュリティホールとなる可能性が高いので常に最新版を利用することが望ましい。

ソースを展開するためのsrcというディレクトリに移動する、ない場合には作成する。

```
# cd /usr/local/src/
```

BINDのソースは一つにまとまっていないため、専用のディレクトリを作り中に移動する。

```
# mkdir bind8
```

```
# cd bind8/
```

展開し、makeする。

```
# tar xvfz /export/home/user/download/bind-src.tar.gz
```

```
# cd src/
```

```
# gmake
```

```
# gmake install
```

7.1.2 専用のユーザとグループを作る

```
# groupadd bind
```

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

7.1.3 古いBINDを無効にする

```
# chmod 0 /usr/sbin/in.named
```

7.1.4 named.confの設定

named.confを/usr/local/etc/に、それ以外の設定ファイルは/etc/namedb/内に作る。

```
# mkdir /usr/local/etc
```

```
# mkdir /etc/namedb
```

/etc/namedbのオーナーとグループをbindにする。

```
# chown bind:bind /etc/namedb
```

注11 BIND (Berkeley Internet Name Daemon)

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

```
options {
    directory "/etc/namedb";
    pid-file "/etc/namedb/named.pid";
    allow-transfer { none; };
};
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";
};
```

この中で指定した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。

7.1.5 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. (
                20030827 ;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
```


7.1.6 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. (
                                20030827      ;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.

```

7.1.7 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. (
                                20030827      ;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

```

7.1.8 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. (
                                20030827      ;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.

```

7.1.9 named.root

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

7.1.10 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	各レコードの有効期限のデフォルト値を設定する。

7.1.11 自動起動

/etc/rc2.d内の起動スクリプトを編集する。

```
# cd /etc/rc2.d
# vi S72inetsvc
```

修正前

```

/usr/bin/pkill -x -u 0 'in.named|inetd'

if [ -f /usr/sbin/in.named -a -f /etc/named.conf ]; then
    echo 'starting internet domain name server.'
    /usr/sbin/in.named &
fi

```

修正後(太字が修正部分)

```

/usr/bin/pkill -x -u 0 'named|inetd'

if [ -f /usr/local/sbin/named -a -f /usr/local/etc/named.conf ]; then
    echo 'starting internet domain name server.'
    /usr/local/sbin/named -u bind -g bind &
fi

```

7.1.12 手動での起動

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

S72inetsvcが起動時に実行されているため、一度停止してから起動する。/var/adm/messagesで稼働状況を確認する。

7.1.13 resolv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。

```

7.1.14 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 (pri=10) by ns.beer.jp    結果
```

ゾーン転送拒否の確認(named.conf)

```
# host -l beer.jp    実行
Server failed: Query refused
```

最後にwww.sun.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。

7.1.15 設定ファイルの再読み込み

```
# /usr/sbin/ndc -p /etc/namedb/named.pid reload
```

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-2.0.47.tar.gz
```

コンパイルする。

```
# cd httpd-2.0.47/
# ./configure
# gmake
# gmake install
```

apacheの起動は専用のユーザで行う様にするため専用のユーザとグループを作る。

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

8.1.2 設定

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

httpd.confが設定をするためのファイル。専用のapacheユーザ、グループを利用して起動する。

```
# vi httpd.conf
```

```
#ServerName new.host.name:80      を
ServerName www.beer.jp:80        に変更する

User nobody      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
```

8.1.3 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
```

8.1.4 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。

8.1.5 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。

8.1.6 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。

8.1.7 自動起動

/etc/rc3.dディレクトリ内のオリジナルのapache起動スクリプトを_S50apache.orgとして、新たにインストールしたapachectlを/etc/rc3.dに起動スクリプトとしてコピーする。

```
# mv /etc/rc3.d/S50apache /etc/rc3.d/_S50apache
# cp /usr/local/apache2/bin/apachectl /etc/rc3.d/S50apache
```

8.1.8 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
```

8.1.9 停止

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

8.1.10 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
```

8.1.11 参考サイト

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

9

FTP Serverの設定

9.1 ProFTPD^{注12} <<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-1.2.8.tar.gz
# cd proftpd-1.2.8/
```

コンパイルする。

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

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
```

9.1.2 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

# 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。
```

9.1.3 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12 FTP [File Transfer Protocol]

9.1.4 自動起動

/etc/rc2.d/に起動スクリプトを書く。

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

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

実行権を付ける。

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

9.1.5 手動での起動

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

で起動する。

9.1.6 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。

9.1.7 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移動出来ないようにする。

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

```
#DefaultRoot ~ 行頭の#を削除する
```

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。

9.1.8 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
```

9.1.9 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.yip.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 qmail1
# 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宛で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。

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
```

10.1.4 sendmailの設定変更

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を止める。

```
# /etc/rc2.d/_S88sendmail stop
```

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
```

10.1.5 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
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。

10.1.6 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
qmail1 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。

10.1.7 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で実行されるようになる。

```
# 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。

10.1.8 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
```

10.1.9 参考サイト

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

10.2 SMTP^{注13}

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.yip.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。

ここではクライアントマシン"192.168.0."からのみの接続を受け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
```

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
```

smtpd_rules, pop3d_rulesで複数のマシンからの接続を受け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
```

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13} SMTP [Simple Mail Transfer Protocol]

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。

```
# less /etc/passwd    UIDを確認
# less /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
```

```
#!/bin/sh
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14}

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.yip.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。

```
#!/bin/sh
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 S88qmail-pop3d
```

10.3.5 手動での起動

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

```
# ps -ef | grep tcpserver
```

上記コマンドでtcpserverのプロセスが二つ(smtp, pop)が稼働していることを確認する。問題なければWindows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14 POP [Post Office Protocol]

10.4 APOP^{注15}

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/>
```

解凍。

```
# 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。

```
% vi ~/Maildir/.password
% 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。

```
#!/bin/sh
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 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
```

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
```

注15 APOP [Authenticated POP]

11

セキュリティについて

11.1 現状

「1月はSQL Slammerが話題だったが、ワースト1は相変わらずKlez」 IPA

コンピュータ・ウイルスの届け出先機関である情報処理振興事業協会セキュリティセンター（IPA/ISEC）は2月6日、1月のウイルス届け出状況を公表した。1月中の届け出件数は1158件。このうち被害に遭ったのは53件だった。1月25日に出現した「SQL Slammer」は話題になったものの、届け出数は7件で、そのうちの2件が被害に遭った。届け出件数が最も多かったのは「Klez」ウイルス（525件）だった。

Klezに次に届け出が多かったのは「Bugbear」で105件、次に多かったのが1月に出現した「Sobig」の68件だった。Sobigの特徴は、差出人アドレスが「big@boss.com」であること（関連記事）。ウイルス添付メールを受け取ったユーザーは差出人を特定できないので、相手がウイルスに感染していることを伝えることができない。その結果、差出人は自分自身で気づくまでウイルス・メールを送信し続けることになる。とはいえ、Sobigウイルスはユーザーが実行しない限り発病することはない。IPA/ISECは、怪しい添付ファイルは開かないよう、改めて呼びかけている。

届け出件数が少なかったものの、SQL Slammerのようなウイルス（ワーム）が再び出現する可能性は高い（関連記事）。SQL Slammerはメモリーに感染し、ファイルなどは作成しないために、ウイルス対策ソフトでは検出できない。このためIPA/ISECでは、SQL Slammerのようなウイルス（ワーム）や不正アクセスを防ぐために、パーソナル・ファイアウォールを導入することを呼びかけている。特に、ADSLのような常時接続環境では“必需品”であるとしている。

同日、IPA/ISECは1月中に寄せられた不正アクセス届け出件数も公開した。届け出件数は30件で、そのうち7件が実際に被害に遭ったという。SQL SlammerはSQL Serverのセキュリティ・ホールを悪用して侵入するすなわち、不正にアクセスするウイルス（ワーム）であるため、IPA/ISECは不正アクセスとしても取り扱っている。

不正アクセス被害に遭わないためには、適切にパッチなどを適用できるよう「まずは己を知ること」を呼びかけている。具体的には、OSやアプリケーションの種類およびバージョン、稼働させているサービスなどをきちんと把握するよう呼びかけている。

（勝村 幸博 = IT Pro）

ITPro 2003.02.06

<<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ー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 (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

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

ログを定期的を確認する。

随時ログをプリントする様にする、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ーの倒産

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 IBM33De11 (IBMでは散々な思いをしたのでCompaq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あ", "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

例えばクラッカーがシステムを乗っ取っ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16}

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。

^{注16} PGP (Pretty Good Privacy) [<http://www.pgpi.org/>]

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。酔っぼらい運転の車が飛び込んでくるなど。

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ると通常は使用する。

11.5.10 バックアップメディア

バックアップ先はリム - 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性がある。

11.6 参考サイト 書籍

11.6.1 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.jpccert.or.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/>

11.6.2 書籍

UNIX & インターネットセキュリティ

Simson Garfinkel, Gene Spafford共著 山口 英 監訳 谷口 功訳

1998年12月発行 ISBN4-900900-38-9

原書: Practical UNIX & Internet Security

PGP 暗号メールと電子署名

Simson Garfinkel 著 山本和彦監訳 株式会社ユニテック訳

1996年4月発行 ISBN4-900900-02-8

原書 : PGP: Pretty Good Privacy

ファイアウォール構築インターネット・セキュリティ

D. Brent Chapman ,Elizabeth D. Zwicky 共著 歌代和正監訳 鈴木克彦訳

1996年7月発行 ISBN4-900900-03-6

原書 : Building Internet Firewalls

DNS & BIND 第4版

Paul Albitz、Cricket Liu 著 高田 広章、小島 育夫監訳 小館光正訳

2002年2月発行 ISBN4-873110-74-2

原書 : DNS and BIND, 4th Edition

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

「qmail メールサーバの構築」著者 Richard Blum, 翻訳 コスモプラネット

株式会社アスキー ISBN4-7561-4001-7

「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

12.1.3 参考雑誌 ムック

「月刊サンワールド」株式会社IDGジャパン <<http://www.idg.co.jp/>>

「月刊サンワールド特別編集 Solaris REVIEW」

株式会社IDGジャパン ISBN4-87280-127-X

12.2 Solaris 8メディアキット

12.2.1 Solaris 8メディアとライセンスの入手方法

Webからの申込み

Solaris オペレーティング環境 <<http://www.sun.co.jp/solaris/>>