

UNIX的要素を備えたMac OS Xの仕組み

理工学部研究協力係 佐 藤 勝 人

miri@cc.hirosaki-u.ac.jp

はじめに

Mac OS Xは、Mac OSの最新バージョンであり旧Macオペレーティングシステムを再構築して、安定性・信頼性を重視しメモリ保護およびプロセス管理やマルチタスクに適したDarwin (BSD UNIX) やJAVAの動作環境 (Java2 Standard Edition) などのMacintoshの標準装備以外のシステムを組み合わせて設計されたOSです。Mac OS Xは、2001年3月24日(日本語版)に発売され、最新バージョン(02/02/27現在)はMac OS X10.1.3です。

本稿では、UNIX的要素を備えたMac OS Xに着目して、その仕組みがどのように成り立っているのかを述べていきます。本編は、

1. Mac OS Xの内部構造について
2. Mac OS Xが対応するファイルシステムについて
3. Mac OS Xのユーザ権限とroot権限について
4. Mac OS Xをサーバーとして扱う (NetInfo、NFS、Samba)
5. Mac OS XのUNIXアプリケーション環境について (X Window System、Fink)
6. Mac OS Xのセキュリティについて (ssh、TCP Wrapper)

の構成で説明いたします。

1. Mac OS Xの内部構造について

Mac OS Xの内部構造を図1に示します。

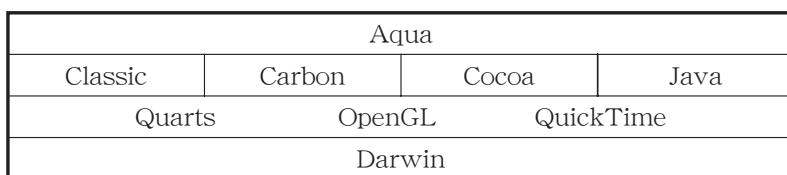


図1

- Darwinは、Mach3.0カーネルのBSD UNIXベースのオペレーティング・システムです。Mac OS Xの核になる部分であり、それ自体がオープン・ソース・プロジェクトです。Darwinでは、メモリ保護・仮想記憶・プリエンプティブマルチタスク・マルチプロセッサ

サポートの機能を提供し、その結果として安定性・信頼性と高速な処理機能を実現しています。

● グラフィックス環境

二次元グラフィックスを扱うQuarts（クオーツ）、三次元を扱うOpenGL（オープンジーエル）、メディアを統合するQuickTime（クイックタイム）によって構成されています。

● アプリケーション環境

アプリケーション環境	実行環境
Classic	旧Mac OS用のアプリケーションを動作させるための実行環境
Carbon	Mac OS 9、Mac OS Xのどちらでも動作可能な実行環境
Cocoa	Mac OS Xのみ動作可能でオブジェクト指向の開発を実現する実行環境
Java	Java 2 Standard Edition 1.3.1をベースにMac OS X用にカスタマイズされたJAVA実行環境
UNIX	図1に記述はありませんが、Darwinが持つAPI（Application Program Interface）としてのUNIX実行環境

● AquaはMac OS XのGUI (Graphical User Interface) として存在します。

2. Mac OS Xが対応するファイルシステムについて

● ローカルファイルシステム

Mac OS Xの主要なファイルシステムにはHFS+(HFS含) とUFSが利用可能です。Mac OS標準：HFS (Hierarchical File System) は、MacintoshのファイルシステムでHFS+はMac OS拡張であることを示しています。UFSは、UNIX系のファイルシステムです。この2つのファイルシステムを扱う際の主な注意点を以下に示します。

1) 大文字と小文字の扱い

HFS+では大文字と小文字は区別されず、UFSでは大文字と小文字は区別されます。

2) パスの区切り文字の扱い

HFS+ではコロン ":" がパスの区切り文字として使用されますが、UFSではスラッシュ "/" が使用されます。区切り文字はシステムによって変換されます。

3) ファイルの更新日の扱い

HFS+ではファイル情報として作成日と変更日がサポートされていますが、UFSでは変更日のみがサポートされています。そのため、変更日のみ認識されているファイルを作成日を認識しないコマンドでコピー処理を行うとコピー先の新規ファイルの変更日の情報がリセットされ、結果的に変更日よりも作成日の方が新しいファイルとして扱われる場合があります。

4) エイリアスとシンボリックリンク／ハードリンク

ファイルやフォルダにアクセスする手段として、HFS+ではエイリアスに対応し、UNIX系のファイルシステムで採用されているシンボリックリンクとハードリンクにも対応しています。UFSではシンボリックリンクとハードリンクに対応していますが、エイリアスには対応しておりません。

エイリアスは、ファイルやフォルダのボリュームと位置を特定することができるため、同一のボリューム上でオリジナルのファイルやフォルダを移動させても、エイリアスは常に同じファイルやフォルダを参照します。エイリアスの作成は、目的のファイル又はフォルダを選択し以下の手順で実行します。

「Finderメニュー」→「ファイル」→「エイリアスを作る」

シンボリックリンクは、ファイルやディレクトリのパスに関する情報のみを扱うため、同一のボリューム上でオリジナルのファイルやフォルダを移動させた場合には、情報が見失われます。シンボリックリンクの作成は、/Applications/Utilities/Terminalアプリケーションのコマンドラインからlnコマンドを使用して下記の書式で実行します。

%ln-s source_file target_file

ハードリンクは、iノード（ファイルの属性情報が保存されるディスク領域）のエントリを利用して、パスやファイル名が異なっていても保有するファイルは同じです。ハードリンクの作成は、Terminalアプリケーションのコマンドラインからlnコマンドを使用して下記の書式で実行します。

%ln source_file target_file

尚、普段はVFS (Virtual File System) により抽象化されているためファイルシステムの違いを意識することなく利用できます。その他の対応しているファイルシステムは、「CD-ROM用ファイルシステム：ISO-9660」、「DVD-ROM/RAM用ファイルシステム：UDF」、「Windows用ファイルシステム：FAT/FAT32」です。

●クロスプラットフォームネットワーキング環境での対応

1) Macintosh用ネットワーク共有ファイルシステム：AFS (Appleshare File System)

AppleTalkとTCP/IPに対応しております。Mac OS XのホストでAFSサーバを立ち上げるには、「システム環境設定」→「共有」→「ファイルとWeb」→「ファイル共有」を有効にすると、ユーザのホームディレクトリにあるPublicフォルダが公開されます。また、AppleTalkを利用してAFSサーバにアクセスする方法を以下に示します。

「Finderメニュー」→「移動」→「サーバへ接続」→「afs://ホスト名」と入力

2) NFS (Network File System)

UNIX系OSで採用されているTCP/IP環境で利用されているファイルシステムです。サーバクライアントの設定に関しては「Mac OS Xをサーバーとして扱う」のところで紹介します。

3) SMBネットワークプロトコル対応 (SMB/CIFS)

現在のMac OS Xは、SMBネットワークプロトコルによるWindowsのファイル共有のクライアント機能をサポートしています。接続方法を下記に示します。

「Finderメニュー」→「移動」→「サーバへ接続」→「smb://ワークグループ名@ホスト名/共有名」

Windows用のネットワークファイルシステムにCIFS (Common Internet File System) があります。CIFS対応の有名なソフトウェアでSharityがあります。このソフトウェアの特徴は、SMBをNFSにプロトコル変換してファイルシステムとしてOS側にNFSの思わせる手法を取り入れています。詳しくは (<http://www.obdev.at/products/sharity/>) をご覧ください。

4) WebDAV

WebDAVは、Webサーバを利用した共有のシステムでApacheやIIS等の代表的なWebサーバで設定が可能です。Mac OS Xは、WebDAVによるWeb共有にアクセスする機能をサポートしています。

接続方法を下記に示します。

「Finderメニュー」→「移動」→「サーバへ接続」→「http://ホスト名/ディレクトリ名/」と入力

3. Mac OS Xのユーザ権限とroot権限について

Mac OS Xはマルチユーザに対応しており、Mac OS Xのインストール時に登録したユーザが一般ユーザになります。この時点では、rootはユーザとして無効になっているためターミナル上からsuコマンドが使用できませんが、アプリケーションの追加他で管理者権限のユーザ名とパスワードの入力を求められた際に、一般ユーザの名前とパスワードで管理者権限を取得することができます。このしくみは、UNIX系のsudoコマンドを利用しておらず、Mac OS Xにユーザ登録をした時点で「/etc/sudoers」の設定（suする権限を一般ユーザに与える）が有効になります。普段コマンドラインから管理者権限を有する作業を行う場合は、「%sudo コマンド名」で処理することができます。

●rootユーザの作成

1) Aqua上からrootユーザを作成する場合は、/Applications/Utilities/NetInfo Manager アプリケーションから、下記の手順で作成します。

「Finderメニュー」→「ドメイン」→「セキュリティ」→「rootユーザを有効」

2) Terminalアプリケーションのコマンドラインからrootユーザを作成する場合は、passwdコマンドで下記の記述で実行します。

`%sudo passwd root`

4. Mac OS Xをサーバーとして扱う

Mac OS Xは、標準でtelnet/ftpサーバ機能が搭載されています。ここでは詳しい説明は省略致します。

●NetInfoについて

NetInfoは、Mac OS X上で利用されているユーザ・グループやネットワークサービス等の通信環境の各種情報をローカルドメインとしてデータベース化して、Macintoshをネットワーク経由で集中的に運営・管理をするためのネットワーク管理システムです。UNIX系OSの「/etc」ディレクトリにある各種設定ファイルに値する各種情報を、NetInfoでは「/var/db/netinfo」ディレクトリ下でデータベース化することで、各種設定情報の管理を行いやすくしています。デフォルトのNetInfoデータベースのディレクトリ構成を下記に示します。

ディレクトリ名	UNIXの/etcに相当するファイル	内 容
/aliases	/etc/aliases	Sendmailが参照するユーザーの別名情報
/config	—	サーバのアドレス等の情報
/groups	/etc/group	ローカルホスト上のグループに関する情報
/localconfig	—	AppleTalkやオートログインに関する情報
/locations	/etc/resolv.conf	DNSサーバアドレスなどに関する情報
/machines	/etc/hosts	ローカルエリアネットワークのホスト情報
/mounts	/etc/fstab	オートマウントの対象になるファイルシステムの設定情報
/networks	/etc/networks	ネットワークの情報
/printers	/etc/printcap	プリンタのエントリ情報
/protocols	/etc/protocols	有効なプロトコルのリスト情報
/rpcs	/etc/rpc	ポートマップのリスト情報
/services	/etc/services	有効なネットワークサービスのリスト情報
/users	/etc/passwd	ローカルホスト上のユーザデータベース情報

NetInfoが管理するネットワークは、NIS (Network Information Service) のような階層構造で情報が管理されており、複数のMacintoshマシンの情報を一元管理ができますが、単体で利用する場合でも必ずNetInfoが稼働しています。尚、Mac OS Xは、NIS/NIS+のネットワークに接続することができます。

- 1) GUIからNetInfoデータベースを管理する/Applications/Utilities/NetInfo Manager アプリケーションを利用します。



NetInfo Managerは、データベースの階層構造を表示する「ディレクトリブラウザ」部分とその内容を表示して編集領域としても利用する「ディレクトリ」欄から構成されています。

2) コマンドラインからNetInfoデータベースを管理する

NetInfoのコマンド一覧を下記に示します。

コマンド名	コマンドの説明
nidump	設定情報をUNIX形式で取り出すためのコマンド
niload	UNIX形式の設定情報を読み込むためのコマンド
nireport	NetInfoの情報を表示するためのコマンド
nigrep	NetInfoの情報を検索するためのコマンド
nifind	NetInfoの中からディレクトリを検索するためのコマンド
niutil	NetInfoの編集をするためのコマンド

3) NetInfoデータベースのバックアップについて

NetInfoデータベースは、システムに依存する情報を扱っているため設定を誤るとマシンが起動しなくなる場合があります。そこで、NetInfoデータベースのバックアップ作業が必要になります。下記の

```
#cp /var/db/netinfo/local.nidb /var/db/netinfo/local.nidb.bak
```

マシンにトラブルが発生した場合は、マシン起動時に「コマンド・キー」+「S」でシングルユーザモードになり、「/var/db/netinfo/local.nidb.bak」ファイルを書き戻します。

● NFS (Network File System) サーバを公開する

UNIX系OSで利用されている設定ファイル「/etc(exports」に準じておりますが、Mac OS X上でNFSサーバを公開するためには、NetInfoデータベースへ設定データを登録しなければなりません。

Mac OS X上でNFSサーバを公開する方法

- 1) NFSで公開するディレクトリを選択します。例として「/Users/nfs」とします。
- 2) /Applications/Utilities/NetInfo Managerを起動して管理者権限を取得してディレクトリブラウザに「exports」ディレクトリがあるかを確認します。存在しない場合は、同じ名前のディレクトリを作成します
- 3) 「exports」ディレクトリを選択した状態で、
 「Finderメニュー」→「ディレクトリ」→「新規のサブディレクトリ」
 を実行して、new_directoryを公開するディレクトリの絶対パス名（この場合は/Users/nfs）に変更します。
- 4) 接続を許可するクライアントを特定します。
 「Finderメニュー」→「ディレクトリ」→「プロパティの挿入」
 を実行して、new_propertyをclientsに変更します。
- 5) プロパティ「clients」を選択した状態で、
 「Finderメニュー」→「ディレクトリ」→「値の挿入」
 を実行しクライアントのホスト名を登録します。
- 6) その他のオプションは、プロパティ「opts」を作成して有効なオプションを値に插入します。主なオプションを下記に示します。

オプション	オプションの意味
alldirs	ファイルシステム全体を公開するためのオプション
maproot=user	クライアント側のrootによるアクセスをサーバ側のuserのアクセスとする
mapall=user	クライアント側の全userのアクセスをサーバ側のuserのアクセスとする
ro	クライアントに読み取り権限のみを許可する

- 7) 設定を保存後にマシンを再起動するか、コマンドラインからnfsd、mountdを立ち上げます。

```
%sudo/sbin/nfsd -t -u -n 6
%sudo/usr/sbin/mountd &
```

● NFS領域をマウントする方法

- 1) NetInfoデータベースに登録し自動的にマウントする

/Applications/Utilities/NetInfo Managerを起動して管理者権限を取得しmountsディレクトリを選択し「NFSSERVER : NFSディレクトリ」(公開されているNFSサーバ名と公開領域)のサブディレクトリを作成して、プロパティ「opts」を作成し有効なオプションを値に挿入します。主なオプションを下記に示します。

オプション	オプションの意味
net	/Network/Serversディレクトリにマウントする
ro	読み取り権限を許可する (初期設定では読み取り書き込み両方許可)
intr	停止しているプロセスをkillできるように割り込みを許可する
-P	NFSサーバと通信するポート番号を常に予約する

設定を保存後にマシンを再起動するか、コマンドラインからmountコマンドを実行します。

```
%sudo mount -at nfs
```

うまくマウントされると、FinderのNetworkフォルダ内のServersフォルダにNFSサーバのアイコンが現れます。

- 2) 手動でマウントする：コマンドラインから、以下のように指定します。

```
#mount -t nfs NFSSERVER : NFSディレクトリ マウントポイント
```

● SMBプロトコルによるWindowsのファイル共有について

Sambaについて

Sambaは、Microsoftのファイル共有プロトコルであるSMBのオープンソース実装です。SambaはMicrosoft Windowsネットワーク・クライアント対応のファイル/プリンタサーバに変換します。Mac OS Xでのファイルサーバを公開する方法を以下に記します。

最新バージョン（02/02/27現在）samba-2.2.2（日本語版リリース1.0）のインストール

- 1) 日本語情報サイト（日本SAMBAユーザ会）<http://www.samba.gr.jp>からソースファイルsamba-2.2.2-ja-1.0.tar.gzをダウンロードします。
- 2) CPPFLAGS環境変数を設定します。Mac OS Xの場合は、デフォルトのプリプロセッサがcpp-precompに設定されており、マクロ処理に失敗することがあるため下記の設定を行います。

```
%setenv CPPFLAGS -traditional-ppc
```

3) configureスクリプトへDarwinのシステム情報を認識させるために、「/usr/libexec/config.sub」と「/usr/libexec/config.guess」をsamba-2.2.2-ja-1.0を展開したディレクトリにコピーします。

4) Makefileの作成を、下記のオプションでconfigureコマンドを実行して、

```
%configure --with-i18n-swat
```

make, ルート権限でmake installする。デフォルトで「/usr/local/samba」へインストールされます。

SWATツールを使用可能にする

1) UNIX上では、「/etc/services」に「swat901/tcp」の記述を追加すればよいが、Mac OS Xの場合はNetInfoデータベースでSWATを有効にする必要があります。NetInfo Manager (GUIツール) で設定するか、またはコマンドラインで、niutilコマンドを下記の手順で実行します。

```
#/usr/bin/niutil -create ./services/swat
#/usr/bin/niutil -createprop ./services/swat port 901
#/usr/bin/niutil -createprop ./services/swat protocol tcp
```

2) /etc/inetd.confに下記の記述を追加する。

```
swat stream tcp nowait.400 root /usr/local/samba/bin/swat swat
```

SWATツールを起動させるには、WWWブラウザを使って下記の記述で実行します。

<http://localhost:901>

3) Sambaユーザの作成をします。

まずはじめに、アクセス制限の設定とSMBパスワードの管理用ファイルの作成をコマンドラインで、

```
# chmod 600 /usr/local/samba/private
# touch /usr/local/samba/private/smbpasswd
# chmod 600 /usr/local/samba/private/smbpasswd
```

で設定してから、下記のコマンドで新規SAMBAユーザの設定を設定します。

```
#[/usr/local/samba/bin/smbpasswd -a ユーザ名]
```

4) Sambaの初期設定ファイル (/usr/local/samba/lib/smb.conf) を編集します。

下記に記述例を示します。

```
[global]
workgroup=GOODCITY
netbios name=Mac OS X-SAMBA
encrypt passwords=Yes
coding system=UTF8-Mac
client code page=932
log file=/usr/local/samba/var/log.%m
max log size=50
```

[homes]

```
comment=%U's Home Directory
path=/Users/%u
writeable=yes
browseable=no
```

前ページの記述では、ワークグループは“GOODCITY”，NetBIOSネームが“Mac OS X-SAMBA”と設定しています。Mac OS X特有の記述は、“coding system”的部分です。Mac OS Xの内部コードがunicodeになっているため、受け取ったシフトJIS文字をUTF-8（unicode）コードに変換する必要があるため“UTF8-Mac”にします。以上でWindowsからファイル共有が可能になります。

5. Mac OS XのUNIXアプリケーション環境について

●UNIXの開発環境を整える

Mac OS Xの製品パッケージに付属されている「Developer Tools CD-ROM」用いて、インストーラ「Developer.mpkg」を実行するとC言語の開発環境に必要なコマンド群（cc, make他）が一式インストールされます。尚、「Developer Tools」CD-ROMを所有されていない方はADC会員サイト（登録が必要）<http://connect.apple.com>から無償でダウンロードすることができます。

●Mac OS XへX Window Systemを組み込む

ソースプログラムが無償で配布されている米MIT（マサチューセッツ工科大学）で開発されたUNIX系の標準のウィンドウシステムです。X Window Systemを構築するためのフリーソフトウェアのXサーバは「Xfree86」で、最新バージョン（02/02/27現在）が「Xfree86.2.0」になります。

Mac OS X用のX Window Systemのインストール方法

●GUIからインストール

http://www.xdarwin.org/files/XInstall_10.1.sit.hqx

http://www.opensource.apple.com/projects/xonx/XInstall_10.1.sit

から入手可能です。展開した後は、インストーラを起動すると対話形式で進行していきます。Xfree86のインストールと同時にXdarwinアプリケーションが/Applicationsにインストールされます。

●コマンドラインからインストール

<ftp://ftp.xfree86.org/pub/XFree86/4.2.0/binaries/Darwin-ppc-5.x>から
Xinstall.sh, extract, Xbin.tgz, Xlib.tgz, Xman.tgz, Xdoc.tgz, Xfnts.tgz, Xfenc.tgz, Xetc.tgz, Xxserv.tgz, Xquartz.tgz

上記ファイル群をダウンロードしてroot権限で、以下のインストーラ・コマンドを実行します。

```
#Xinstall.sh
```

●パスの環境変数を設定する

ユーザのホームディレクトリに.cshrcファイルを作成して、

```
setenv PATH "/usr/X11R6/bin:${PATH}"
```

を追加して次回ログイン時から有効になるようにします。

●X Window Systemの起動

ログインウインドウのユーザ名に「>console」と入力します。コンソールモードからユーザ名・パスワードを入力し、startxコマンドでX Window Systemを起動します。

●XDarwinアプリケーションからX Window Systemを起動する

1) フルスクリーンモードで起動



Xdarwinを起動すると上記の画面が表示されますので右下の「フルスクリーン」ボタンをクリックするとX Window Systemに切り替わります。Mac OS X (AQUA) へ切り替える場合は、「Command」+「Option」+「Aキー」を押します。

2) ルートレスモードで起動

Xdarwinを起動すると上記の画面が表示されますので右下の「ルートレス」ボタンをクリックするとAQUA上にX Window System環境が共存した状態で立ち上がります。ただし、キーボード入力は英語キーマップ配置のため慣れないと苦労します。



図：ルートレスモードで起動したときの画面

● Finkについて

Finkとは、Darwin上でUNIX向けのオープンソースソフトウェアを管理するためのツールです。

ソフトウェアのインストールから依存関係のチェック・インストール状況などを一括で扱う優れた機能を持っています。詳しい情報は、<http://fink.sourceforge.net/>を参照してください。

6. Mac OS Xのセキュリティについて

Mac OS Xは、セキュリティ強化のためSSH (Secure Shell) が組み込まれています。SSHは、通信路を暗号化することで安全性を高めたリモートシェルです。そのため、Mac OS Xのtelnet/ftpサーバにアクセスする際は、以下の方法で行います。

Mac OS X又はUNIXマシンからtelnet/ftpサーバにアクセスできるようにするために、SSHがアクセスクライアント側にも組み込まれている必要があります。

- telnetサーバへアクセスする場合は、ssh又はsloginコマンドを使用します。

```
%ssh username@hostname又は %slogin username@hostname
```

- ftpサーバへアクセスする場合は、sftpコマンドを使用します。

```
%sftp username@hostname
```

●TCP Wrapperについて

TCP Wrapperは、各ネットワークサービスについて細かくアクセス制御を行うためのセキュリティツールです。Mac OS Xでは、TCP Wrapperが標準で組み込まれています。ネットワークサービスを起動するためのinetdデーモンの設定ファイル/etc/inetd.confの記述の操作は、「システム環境設定」→「共有パネル」で行うtelnet/ftpサーバ設定の部分で/etc/inetd.confの#（コメント行）の追加/削除を行っているだけです。そのサービスの部分にTCPWrapperに含まれる「tcpd」が間に入ることで、アクセス許可するホスト情報/etc/host.sallowとアクセス拒否するホスト情報/etc/hosts.deny各ファイルの内容に従ってアクセス制限を可能にしています。

Mac OS Xのセキュリティに関する情報サイト <http://www.securemac.com/>

結び

あえてUNIXとしてMac OS Xを使用する場合は、UNIXの知識が必要になります。Mac OSの環境だけで作業するだけでもMac OS Xの魅力が伝わってきますが、Mac OS Xの製品パッケージに付属されている「Developer Tools」CD-ROMを利用することによって、以前よりも容易に開発環境が整います。Darwin (BSD UNIX) のUNIXアプリケーション環境を利用してことでMac OS XおよびDarwin特有の部分とUNIXの勉強に少しでも繋がれば幸いです。

参考文献

- [1] 海上 忍著、「Mac OS X徹底活用ガイド—UNIXスタイルの使いこなし術—」
(技術評論社、2001)
- [2] ソフトウェア・デザイン「Mac OS XでUNIX体験！」 2001／7月号 (技術評論社)