MacOSX のセキュリティ

理工学部研究協力係　佐　藤　勝　人
miri@cc.hirosaki-u.ac.jp

はじめに

MacOSX は、MacOS9 までの要素と BSD、NeXTSTEP の要素が一つになった構造になっており、Mach3.0 ベースのマイクロカーネルで、BSD UNIX とシステムコールでの互換性を持つ Darwin が MacOSX の基盤を形成しているため、他の OS と比べて個性的で魅力的な一面を兼ね備えている。

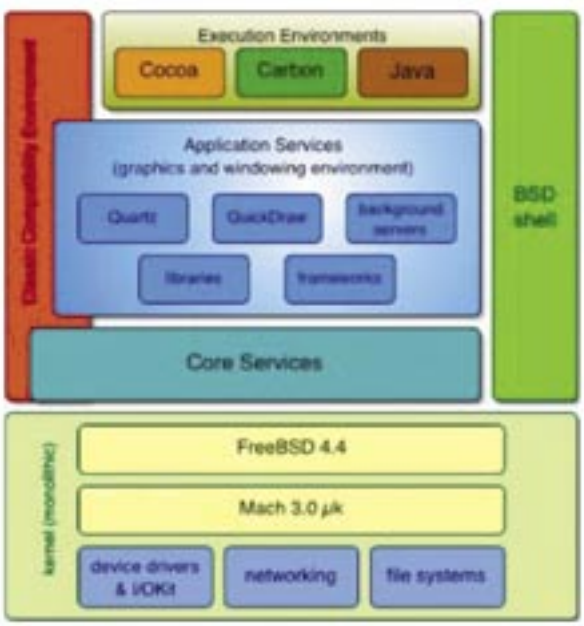

図 1　MacOSX10.2 (Jaguer)

MacOSX のセキュリティについては、BSD で組み込まれて利用されているものや、一般に使用されているセキュリティツール群がすべて利用できる状態ではないが、一時期から比べると MacOSX に導入できるフリーのセキュリティツールが増えてきた。

本稿では、MacOSX の新規インストール時のシステムの初期設定段階から、よりセキュアな環境を構築するために必要なフリーのセキュリティ・ツールを紹介する。

1．　主なフリーのセキュリティ・ツール
2．　MacOSX10.2 の初期設定 (BSD カーネルレベルでのセキュリティ、MacOSX 全体のセキュアな構造)
3．　ソケットレベルでアクセス制御（ipfw）
4．　アプリケーションレベルでのアクセス制御（TCP_Wrapper, xinetd, PortSentry）

5． 侵入検知に関わるセキュリティ対策（Snort）
6． 無線 LAN 環境でのセキュリティ対策（Wi-Fi 認証について、無線 LAN 規格の歴史、無線 LAN のセキュリティ）

1．主なフリーのセキュリティ・ツール

UNIX や Linux、MacOSX で利用されている主なフリーのセキュリティ・ツールを以下に紹介する。

図2 主なフリーのセキュリティ・ツール

カテゴリー		ツール名	入手先	対応 OS
クラッキング対策ソフト	パスワード解析	John the Ripper	http://www.openwall.com/john/	複数のプラットホームに対応（MacOSX 使用可）
	アクセス制御	TCP_Wrapper	ftp://ftp.porcupine.org/pub/security/index.html	UNIX、Linux、BSD、MacOSX 標準搭載
	ファイアーウォール	ipfwadm	http://www.xos.nl/linux/ipfwadm/	Linux2.0 カーネルのパケットフィルタリング・ツール
		ipchains	http://www.netfilter.org/ipchains/	Linux2.1.1 以降のカーネルのパケットフィルタリング・ツール
		ipfw		BSD・MacOSX 標準搭載
セキュリティチェックツール	ポートスキャナ	nmap	http://www.insecure.org/nmap_download.html	複数のプラットホームに対応（MacOSX 使用可）
	ネットワークスキャナ	SATAN	http://www.fish.com/satan/	UNIX、Linux
		Nessus	http://www.nessus.org/	検査結果を GUI で表示するプラグイン型のツール（UNIX、Linux、MacOSX 使用可）
侵入検知システム	侵入検知ツール	Snort	http://www.snort.org	UNIX、Linux、BSD、MacOSX 使用可
		PortSentry	http://www.psionic.com/index.html	UNIX、Linux、BSD、MacOSX 使用可
	パケットキャプチャツール	Tcpdump	http://www.tcpdump.org/	複数のプラットホームに対応（MacOSX 標準搭載）
ホスト / ログ監視ツール	システム・ログ監視ツール	Swatch	ftp://ftp.stanford.edu/	ログ・ファイルの監査、リアルタイムのログ監査他が可能なログ監視ツール（UNIX、Linux）
		Logcheck	http://www.psionic.com/abacus/logcheck	TCP_Wrapper 他で生成されたログを処理するツール（UNIX、Linux、MacOSX 可）
	ファイル改ざんチェックツール	Tripwire	http://www.tripwiresecurity.com/	UNIX、Linux
		yafic	http://philosophysw.com/software/yafic/	Linux、BSD、MacOSX 対応

・Fink ツールについて

Fink プロジェクトでは、Unix 用ソフトウェアを Mac OS X（Darwin）に移植したパッケージやパッケージ管理ツールを提供している。パッケージ形式は Darwin と同様に Debian（*.deb）形式が採用されている。(Fink ツールの 2003 年 2 月現在のバージョン：0.5.1)
Fink ツールのダウンロード：http://fink.sourceforge.net/download/index.php

Fink ツールの GUI フロントエンドである FinkCommander を利用して、より簡単にパッケージ管理ができる。(FinkCommander ツールの 2003 年 2 月現在のバージョン：0.5.0)
FinkCommander ツールのダウンロード：http://finkcommander.sourceforge.net/

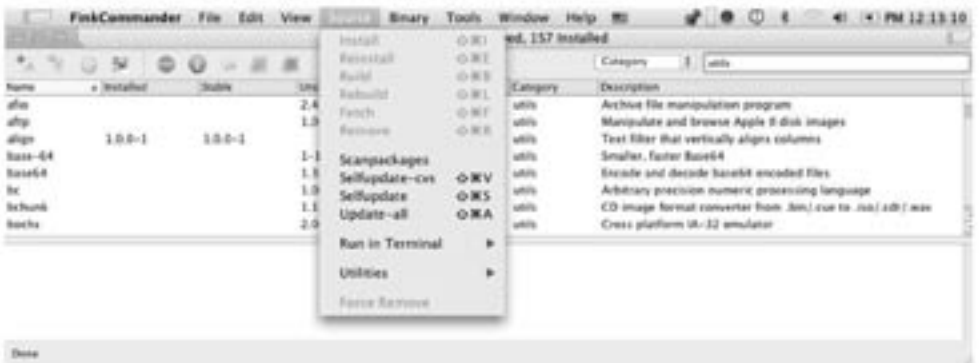

図３　FinkCommander ツール

２．MacOSX10.2 の初期設定

２．１　BSD カーネルレベルでのセキュリティ

・初期設定時は、セキュリティを確保するため、すべてのネットワークサービス (TELNET、FTP、AFS 含) がオフになっており、ポートもすべて塞がっている。
・root のパスワードが設定されていない。root 権限が必要なときは sudo コマンドを使用する。root 権限の設定は、コマンドラインで

%sudo passwd root

とするか、NetInfo で設定する。sudo の設定変更は、visudo コマンドで /etc/sudoers を編集する。
・ssh が標準搭載されており、リモートログイン機能以外は、初期設定時で使用可能になっている。
・TCP/UDP ネットワークサービスは、デフォルトでは /etc/inetd.conf で設定する inetd が使用されているが、/etc/xinetd.conf ,/etc/xinetd.d/ 以下で設定する xinetd も動作している。セキュリティ面においては、すべてのプロセスで xinetd を利用するのが理想である。「アプリケーションレベルでアクセス制御」で xinetd へ移行する方法を紹介する。

２．２　MacOSX 全体のセキュアな構造

・Open Firmware

Open Firmware は、システムの起動に関する諸条件を変更する時に、PC/AT 互換機の BIOS に近い役割を担っている Open Firmware を呼び出す。Open Firmware の起動は、MacOSX 起動時に [Option]+[Command]+[O]+[F] を同時に押し続ける。
セキュリティを強化するために Open Firmware 起動時にパスワード入力を求められるようにする。

・キーチェーン・アクセス

MacOSX のキーチェーン機能は、ファイルサーバや Mail サーバなどのネットワークサービスにアクセスしたり、暗号化したイメージファイルを利用したりするときに必要になるユーザ名とパスワード入力を自動化するための管理システムである。システムのキーチェーン・ファイルは /System/Library/Keychains/ に、各ユーザのキーチェーン・ファイルは、~/Library/Keychains/ にある。キーチェーン管理ツールは、/Applications/Utilities/Keychain

Access.app の「キーチェーンアクセス」を使用する。

・シャドウパスワードファイル（/etc/master.passwd）

MacOSX のシャドウパスワードファイルは、BSD 系の /etc/master.passwd である。しかし実際は、NetInfo データベースで管理されている。NetInfo データベースの登録されているユーザとパスワード情報を確認するには、

#nidump passwd .

NetInfo データベースで管理されているので、結果的にセキュアな環境が維持されていると思われる。

3．ソケットレベルでアクセス制御

MacOSX では､BSD で使用されているパケットフィルタリング型のファイアーウォール「ipfw」が標準で搭載されている。ipfw は､カーネルに組み込まれているので OS のプロセスで動作する。

図３　［システム環境設定］－［共有］－［ファイヤーウォール］項目

・ipfw の設定について

1）ルール設定ファイル (例 ipfw.sh)

シェル・スクリプト・ファイルを作成して、ルールを一通り記述する。

2）状態依存型フィルタリング：「check-state」、「keep-state」

状態依存型フィルタリングは、MacOSX10.2 で使用可能である。このフィルタリング機能は、インターネットのクライアントからサーバにアクセスしたときの事実を記憶しておくことで、サーバからクライアントへのレスポンスを一定時間許可することができる。実際は、「keep-state」で、動的に一定時間レスポンスを許可して、設定したルールに一致した際に、動的ルールファイルを作成する。「check-state」では、作成された動的ルールセットに対してパケットのチェックを行い、一致した場合は、その動的ルールを生成したルールに関連づけられたアクションを実行する。一致しなかった場合は、次のルールに移る。

３）ログの出力：初期状態では、ログを採取することができないので、以下の設定をルール設定ファイルに記述する。

/usr/sbin/sysctl　– w　net.inet.ip.fw.verbose=1

４）OS 起動時に ipfw を起動させるためには

システムに依存するプログラムは、/System/Library/StartupItems/ 以下に起動スクリプトを、リソースと XML 形式の .plist の拡張子をもつ設定ファイルをプログラム毎にフォルダーにまとめて追加する。追加アプリケーションの起動は、/Library/StartupItems/ 以下に同様に追加する。ipfw の場合は、/Library/StartupItems/IPFW/IPFW の起動スクリプトを以下の形式で作成する。

```
#!/bin/sh

. /etc/rc.common

if [ "${IPFW:=-NO-}" = "-YES-" ]; then
      ConsoleMessage "Starting IPFW"
      if [ -x /usr/local/sbin/ipfw.sh]; then
         /usr/local/sbin/ipfw.sh       # ルール設定ファイル（シェル・スクリプト・ファイル）
      fi
```

更に、/Library/StartupItems/IPFW/StartupParameters.plist ファイルを作成する。

```
{
   Description = "IPFW";
   Provides = ("IPFW");
   Requires = ("DISK","Network");
   OrderPreference = "None";
   Messages =
   {
      start = "Starting IPFW";
      stop = "Stopping IPFW";
      restart = "Restarting IPFW";
   };
}
```

最後に、/etc/hostconfig に以下の行を変更・追加する。

IPFORWARDING=-NO- を　-YES-　に変更する

IPFW=-YES-　　この行を追加する。

登録された各変数の値を -YES- にすることで、OS 起動時に StartupParameters.plist ファイルを参照して、サービスを開始する。

/etc/hostconfig に追加するシェル変数は、起動スクリプトで if 文に使用したものと同じでなければならない。

ipfw 設定についての参考サイト：http://www3.sympatico.ca/dccote/firewall.html

４．アプリケーションレベルでアクセス制御

クラッキング対策として、アクセス制御とファイアーウォールの構築を行う。

４．１　アクセス制御

アクセス制御ツール「TCP_Wrapper」は、MacOSX に標準で搭載されている。設定ファイルは、アクセス許可を許すための /etc/hosts.allow とアクセス拒否を指定するための /etc/hosts.deny が必要で、こちらでは /etc/hosts.deny ですべてのアクセスを拒否している。

４．２　inetd からすべてのサービスを xinetd へ移行するための設定

１）inetd の処理をすべて停止させる。
OS 起動時にネットワークサービスを起動するために読み込まれる /System/Library/StartupItems/IPServices の「inetd」の記述をコメントアウトする。

２）xinetd が OS 起動時に正常に立ち上がるようにする。
/System/Library/StartupItems/IPServices の記述にある
xinetd -pidfile /var/run/xinetd.pid
の部分で指定されている /var/run/xinetd.pid はデフォルトでは存在しないため、touch コマンドで作成する。

```
#touch /var/run/xinetd.pid
```

３）マシンを再起動する。
TCP_Wrapper と inetd でアクセス制御を行う場合は、tcpd を参照させることで TCP_Wrapper を組み合わせることができるが、TCP_Wrapper と xinetd の組み合わせでは、/etc/xinetd.d/ 以下のファイルに「only_from」でアドレスを指定できる。tcpd 参照が不要になるため、よりセキュアな環境が構築された。

４．３　二番目の層にアクセス防御対策

PortSentry を利用して、パケットフィルタリングの設定ミスを防御するための二番目の層を構築して遮断することを目的とする。PortSentry は、個々のポートに対してのアクセス制御ができないので、パケットフィルタリング処理が設定ミス等で正確に行われなかった場合に、すべてのパケットを捕まえて防御するために使用する。ポートスキャン、ステルススキャンを検出して（nmap ができることは全て検出できる）ログに記録する。UDP/TCP からの不正アクセスを防御するため、TCP Wrapper(hosts.deny) と連動させることができる。

・PortSentry の導入
一番容易な方法は、Fink ツールを利用して porsentry を導入する。
設定ファイル：設定ファイル portsentry.conf を編集して、portsentry.ignore ファイルにチェックの対象外の IP アドレスを登録する。
PortSentry の動作方法：パケットフィルターに数カ所の穴を開けて、ポートスキャンを検出して防御できるかを調べる。

５．侵入検知に関わるセキュリティ対策

侵入検知システム（IDS）は、コンピュータやネットワークへの不正侵入・攻撃に対する検出を行うためのシステムである。侵入検知システムとして Snort を導入した。Snort は、ホストマシンへの入力パケットを見張る監視ツールであり、ホスト型、ネットワーク型の両方の運用形態で利用可能である。異常データの検出方法は、ルールリストと比較して一致したものを検知する

パターンマッチング方式が採用されている。Snort の欠点は、ホームページで公開されているルールリストを随時更新する必要があり、未知の攻撃パターンに対するルールリストの対応に時間が要する点である。使用上の注意点は、どのように侵入が試みられたかをログに記録するが、侵入行為事態は阻止できないということである。

・Snort の導入

一番容易な方法は、Fink ツールを利用して Snort を導入する。

他に必要なツール：libpcap(LAN 上のパケットをモニタリングするために必要なライブラリ)

オプションモジュール：openssl(暗号化された通信を実現するために必要)

設定ファイル：ルールリストを上記ホームページからダウンロードし、設定ファイル snort.conf を編集して、専用ユーザ (snort) を作成する。

Snort の動作方法：有効にしたルールリストに値する攻撃を試みて、Snort のログに記録されているかをチェックする。

６．無線 LAN 環境でのセキュリティ対策

無線 LAN ベースステーション (Apple 社 AirMacExtreme) や他メーカーから、高速無線 LAN 規格の IEEE802.11g と IEEE802.11b、IEEE802.11b 互換の 3 種類の接続モードに対応したモデルが発売されている。そこで本章では、「Wi-Fi 認証について」、「無線 LAN 規格の歴史」、「無線 LAN のセキュリティ」について紹介する。

６．１　Wi-Fi 認証について

Wi-Fi は、無線 LAN 業界団体「WECA」が認証を行っている相互接続性の保証制度であり、Wi-Fi 認証を取得している製品同士の通信は、WECA によって保証される。しかし、実際は Wi-Fi に未対応であっても、最近の IEEE802.11b 製品同士であれば、ほぼ互換性が保たれている。大学やその他施設のいろいろな環境で利用する場合は、Wi-Fi 認証を得ているクライアント製品を選択した方が良い。今回導入した無線 LAN ベースステーションは、IEEE802.11b 互換モードは Wi-Fi 認証を取得しており、基本的には同認証を取得している IEEE802.11b 機器であれば相互接続が可能だが、機器によってはうまく接続できない場合があるので、このような場合には純粋な IEEE802.11b モードで使用する。

６．２　無線 LAN 規格の歴史

・IEEE802.11b(1999 年登場)

周波数帯は、2.4GHz 帯で通信速度が最高で 11Mbps、実効速度は 2 ～ 8Mbps 前後で、現在もっとも普及している。内蔵ノートパソコン･PDA 他の様々な形態の製品が存在する。

・IEEE802.11a(2000 年登場)

IEEE802.11b より高い周波数帯の 5.2GHz 帯で、通信速度が IEEE802.11b より約 5 倍の 54Mbp のより高速な無線 LAN 規格である。製品化は早かったが、普及している IEEE802.11b との互換性がないため、IEEE802.11a 機能のみ対応製品はあまり普及していない。

・IEEE802.11a/b(2001 年登場)

IEEE802.11a と IEEE802.11b の両規格に対応する無線 LAN 製品である。IEEE802.11a が使用できる場所では高速通信が可能であり、IEEE802.11b しか利用できない環境では IEEE802.11b として機能する。

・IEEE802.11g(2003 年登場)

IEEE802.11b と同じ周波数帯の 2.4GHz を使用し、通信速度が IEEE802.11a と同じ 54Mbps の無線 LAN 規格である。IEEE802.11b 機器との通信も可能で、IEEE802.11a/b 製品より価格が安い。IEEE802.11g は現在、ドラフトという規格の正式承認前の段階で、2003 年 5 月頃に正式承認される予定である。

６．３　無線 LAN のセキュリティ

１）MAC アドレス・フィルタリング

MAC(Media Access Control) アドレスとは、LAN デバイスのハードウェア毎 (無線 LAN カード、LAN カード、LAN 搭載マシン、その他機器) に振られている 12 桁の数字の固有番号を指す。MAC アドレスは、全世界に一つしかないので、無線 LAN ベースステーションに利用する MAC アドレスを登録して、接続できる端末を制限することができる。この機能を MAC アドレス・フィルタリングという。

２）WEP 暗号化機能

WEP(Wired Equipment Privacy) とは、通信内容を暗号化する仕組みをいう。実際の使用には、セキュアな環境にするため、WEP のパスワードを複雑な組合せで指定する。

３）ESS-ID

無線 LAN システムには、混信を防止するため、計 14 チャンネルに分かれており、クライアントと無線 LAN ベースステーション間では、同じチャンネルを使用しなければアクセスができない。しかし、近くの無線 LAN で同じチャンネルを使用していれば、相互の無線 LAN システムが見えてしまうのを防ぐために、ESS-ID(ExpandedServiceSet) という、無線 LAN システムをグループ単位で扱うための仕組みが用意されている。実際の使用には、ステルス ESS-ID 機能を利用して、クライアント側からの問合せにのみ返事をするようにして、第三者から ESS-ID が直接見えないようにする。

４）二重のセキュリティ強化

IPsec 他のツールで、通信そのものを暗号化することでよりセキュアな環境を整えることができる。

・ MacOSX で IPsec ツール導入についての参考サイト

http://homepage1.nifty.com/glass/tom_neko/web/web_ipsec.html

まとめ

MacOSX のセキュリティを構築する際は、FreeBSD のセキュリティ構築やセキュリティ・ツール情報がすごく参考になった。

全世界で、悪意のある第三者が存在するかぎり、インターネットに接続されているすべてのマシンが攻撃を受ける可能性があるので、個々のマシンにもファイアーウォールやその他のセキュリティ対策は必須である。

MacOSX のセキュリティを構築する上で、本稿が少しでも参考になれば幸いです。