Linuxサーバ構築とセキュリティ

Linux Business Initiative
代表 久保　元治
(株式会社サードウェア代表取締役)

目次

- サーバ構築の概要
- 「身の丈」セキュリティ対策
- アクセスを制御する
- システム運用管理
- 情報収集の方法
- 最近のセキュリティ動向から

サーバ構築の概要

Linuxのインストール自体は非常に簡単になってきた。ここではセキュリティ対策を意識したサーバ構築のポイントを紹介する。

「Linuxの種類」について

- 「Linux」とは
  - カーネルの名称(本来の意味)
  - カーネル、周辺コマンド、アプリケーションなどを統合した「ディストリビューション」(広義)
- 代表的なディストリビューション
  - Red Hat、TurboLinux、Slackware、Debian、Calderaなど

Red Hatを取り上げる理由

- 運用管理を重視したパッケージング
- バグ対応などのアップデートが迅速
- 商用ベースのサポートも利用可能
- (私自身の)経験がもっとも豊富
- 以下、おもにRedHat 5.2ベースで説明する

インストール

- ほとんどのPC (AT互換機)にインストール可能
- バージョンごとにインストールがより容易に (Red Hat 5.2ではハードウェアを自動検出)
- 難しいのはパッケージ選択
- Serverインストールは選ばない方がよい (Customを選ぼう)

サーバ用のパッケージ選択ガイドライン

- **鉄則: 必要最小限かつ最新のパッケージをインストールする**
  - 最新のディストリビューションを使う
  - 書籍添付の古いCD-ROMは論外
  - FTPのupdatesディレクトリで最新パッケージにアップデートするのが望ましい
- よくわからないものは、とりあえずインストールしない

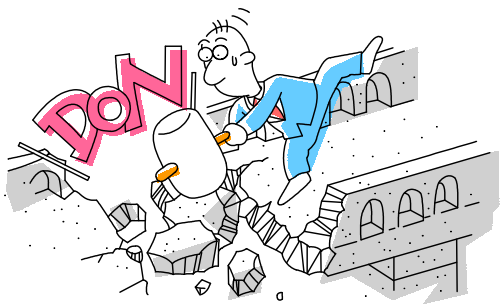
サーバ用のパッケージ選択ガイドライン

- 最小インストールし、rpm -iで追加インストールするのも有力
- Xウィンドウは不要
  - コマンドベースで十分管理できる
  - システムリソースが少なくてすむ
  - X自体がセキュリティホールになることも
- コンパイラ類も極力インストールしない

インターネットに接続する前に

- セキュリティポリシーを立てよう
- 不要なサーバはホールになりかねない
  - ps ax で起動されているデーモンを確認
- 最小限のアクセス制御
  - /etc/inetd.confの設定
  - /etc/hosts.allowの設定
- パスワードのシャドウ化
  - 4.2ではpwconv5、5.xではpwconvコマンド

「身の丈」セキュリティ対策



コストや人手に制約がある場合でも、セキュリティ対策は欠かせない。現実的なセキュリティ対策について考えてみたい。

「身の丈」とは

- セキュリティ対策は、本来手間がかかり、専門的知識が必要
- 大規模サイトだったら、可能だし必要
- 小規模サイトだったら….
  - かけられるコストと手間は限られている
    - 利用目的を絞り、アタックされにくくする
    - そのための重要ポイントは何か
    - 継続的な対応のための有用な情報源

セキュリティ対策の必要性

- UNIX系はリモートからシェルを使える
  - 自滅ではすまない
- 不正侵入されたら....
  - システム資源の悪用
  - 業務データ、プライバシーの流出
  - 「踏み台」として使われる
    - 他サイトをアタックするための出先基地
    - 社会的信用の喪失につながる

まさか私のマシンは....

- これは通用しない!
  - 自動化したアタックツールによって全部のインターネットサーバが標的になっている
  - インターネットに接続したら、たった数日後に不正アクセスが来ることも

不正侵入までのステップ

- 利用可能な「入り口」のスキャン
  - スキャンを自動化するツールが出回っている
  - 入り口が見つからなかったら別のサイトに回る
- とりあえず侵入
  - パスワード推測、セキュリティホールなどを利用

不正侵入までのステップ

- 「バックドア」の設置
  - 次回以降の侵入を容易にするため
  - プログラムの置き換え、設定の変更など
- 侵入を検出しにくくする
  - 侵入者を見せないプログラムに置き換え
  - ログの改竄など

セキュリティポリシー

- サーバ設置の目的は?
  - できるだけ明確に絞り込む
  - 危険なサービスは実施しないか、代替手段を使う
    - telnet、FTPなどは極力使わない
  - 当面利用しないものは動かさない
    - POPのみならIMAPはインストールしない
  - 利便性とセキュリティはトレードオフ

セキュリティポリシー

- 何を保護するのか? その理由は?
  - データ?
    - インターネット・サーバに内部データは置かない
  - 社会的信用?
    - 踏み台、メールの不正中継対策がとくに重要
  - ハード資源やソフトウェア自体?
    - 再インストールは手間と時間のムダ

セキュリティポリシー

- どういう手段で保護するか?
  - 外部からのログインは不可欠?
  - パスワード管理は?
  - WWWサーバでCGIは必要?
  - 日常の監視内容と方法は?
    - スキル、かけられる時間によって、保護方法と監視方法を決める

セキュリティポリシー

- 問題が生じたらどう対応するか?
  - 緊急時にパニックにならないために
    - 現象の把握とネットワークからの遮断
    - 原因の追求
    - 復旧方法
  - 連絡先、アドバイザーを明らかにしておく

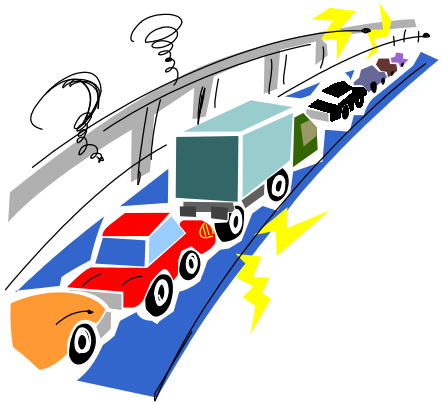
オープンソースの対策

- Linuxカーネル自体のファイアウォール機能
- 有用なアクセス制御ツール
  - tcp wrapper
- 不正アクセス監視ツール
  - Tripwire、swatchなど
- こまめな情報収集とアップデート

「身の丈」対策のポイント

- 不正アクセス自体は不可避と覚悟する
- 目的を絞り込んでそれだけを公開する
- 一般に危険なサービスは極力禁止する
- 情報収集とこまめなアップデート

アクセスを制御する



セキュリティ対策の最初のポイントはアクセス制御である。主要サービスごとのアクセス制御について考えてみよう。

ログイン

- ログイン、シェルの提供は両刃の剣
  - リモートからシステム資源の利用や管理が可能
  - 不正アクセスでシステムの「乗っ取り」、「踏み台」化も可能
- ユーザ名とパスワードでログイン可能に
  - 推測しやすいパスワードはきわめて危険

/etc/passwd

- 暗号化したパスワードが保存される
  - シャドウ化すれば/etc/shadowに移される
- 一般ユーザでも読めるファイル
- パスワードは推測可能
- パスワードファイルのシャドウ化は必須

```
root:Xf4xQo72TYXgY:0:0:root:/root:/bin/bash
```

/etc/shadow

- シャドウ化すると作られるファイル
  - 暗号化したパスワードなどが格納される
- 一般ユーザでは読めない
- コマンド一発でシャドウ化が可能
  - pwconv5 (4.2)またはpwconv (5.x)
- ソースで持ってきたソフトウェアでは個々にシャドウ対応が必要

ネットワーク上の盗聴

- ネットワーク上を流れるデータ(パケット)の盗聴は技術的に可能
- パスワードも盗聴の対象に
- インターネット上で暗号化しない(平文の)パスワードを使うのは危険

telnet

- リモートからのログインプログラム
- ログインすればシェルがただちに使える
- ユーザ名とパスワードは暗号化されない
- インターネットにtelnetアクセスを開放するのは、きわめて、きわめて危険
- パスワードなしでリモートログインやリモートコピーを行うr-cmdはさらに危険

一般ユーザからrootへ

- いくつかの方法がある
  - /etc/passwdでパスワードを推測する
    - シャドウ化が有効な対策
  - 侵入後、既知のコマンドのバグを悪用
    - セキュリティホールはいろいろある
    - すべてに完全に対応するのは必要だが難しい

ワンタイムパスワード(OTP)

- 複数のパスワード(パスフレーズ)を生成しておく
- ログインのたびに新しいものを使う
- OPIE、S/Keyなどが利用可能
  - opie-2.22、logdaemon-5.6が最新
- RPM化したパッケージはまだ作られていない

Secure Shell (SSH)

- **通信内容すべてを暗号化**
- **事前にアクセスを認め合ったホスト間で使う**
- **シェルアクセス、リモートコピーが可能**
- **ライセンス上商用利用には使えない**
- ssh-2.0.9**が最新**
- RPM**化したパッケージも**FTP**で入手可能**

OTP v.s. SSH

- パスワード盗聴防止にともに有効
- OTPの利点
  - 使い捨てパスワード系列がわかっていれば、出先のホストを借りてログインすることも可能
- SSHの利点
  - 通信内容全体を暗号化するので、パスワード以外の秘密も守れる
  - rlogin、rsh、rcpと同じ操作性で使える

OTP v.s. SSH

- OTPの欠点
  - 通信内容は保護されない
- SSHの欠点
  - あらかじめ認証しあったホスト間でしか使えない
- 必要に応じて両者を使い分け、telnetでのリモートログインを禁止するのがよい

スーパーデーモン inetd

- いくつかのサービスを一括して監視
  - telnet、ftp、pop、imap、finger など
- クライアントからのアクセスに応じて、実際のサーバプログラムを起動
- /etc/inetd.conf で挙動を制御

/etc/inetd.conf

- 1行が1つのサービスに対応
- 行頭に#が付いていない行のサービスが提供される
- 提供しないサービスはコメントアウトする
- 相手によって禁止/許可するサービスはtcp wrapperを使う

/etc/inetd.conf

- **たとえばimapを使っていないなら、次のように書き換える**

```
imap    stream  tcp     nowait  root    /usr/sbin/tcpd  imapd
```

↓

```
#imap    stream  tcp     nowait  root    /usr/sbin/tcpd  imapd
```

- **変更を有効にするには、inetdにSIGHUPシグナルを送る**

TCP wrapper (tcpd)

- 実体プログラムは /usr/sbin/tcpd
- 挙動は /etc/hosts.allow、/etc/hosts.deny で制御
- 設定ファイルの書き方は2種類
  - 拡張記法だと/etc/hosts.allowだけを使う
- man 5 hosts_access を参照

以下の説明は拡張記法を使う

/etc/hosts.allow の例

- **すべてのアクセスを許可**

```
ALL:   ALL:  ALLOW
```

- **内部ネットワーク(**192.168.0.0/255.255.255.0**)からのすべてのアクセスを許可し、それ以外はすべて拒否**

```
ALL: 192.168.0.0/255.255.255.0: ALLOW
ALL: ALL: DENY
```

/etc/hosts.allow の例

- **拒否したアクセスは管理者にメールで通知**

```
ALL:   ALL:  ＼
spawn (/usr/sbin/safe_finger -l @%h | /bin/mail -s ＼
"%d-%h" root) &:¥DENY
```

- **サービスごとに制御**

```
ALL:  192.168.0.0/255.255.255.0:  ALLOW
in.ftpd:  ALL:  ALLOW
popper:  210.123.45.67:  ALLOW
ALL:   ALL:  DENY
```

Phf スクリプト(WWWサーバ)

- phfスクリプトはシェルコマンドを実行する
  - 最近のパッケージには入っていない
- /etc/passwdファイルなどの窃取に悪用される
- /var/log/httpd/access_logで確認できる

```
…. “GET /cgi-bin/phf?…. 404 -
```

  - 404ならば大丈夫

ブート時に起動されるサーバ

- 常時サーバを待ち受けるサーバ(デーモン)もある
  - DNSサーバ (named)
  - メールサーバ (sendmail)
  - WWWサーバ (httpd)など
- システムのブート時に自動的に起動される

ブート時の起動を停止する

- /etc/rc.d/rc3.d/に起動スクリプトが存在
- ファイル名の先頭文字がS(大文字)
  - S80sendmail など
- 数字は起動順序を表す
- ブート時に起動しないようにするには
  - ファイルを削除(勧めない)
  - 先頭文字を “S”、”K”以外にリネーム

デーモンの起動と停止

- デーモンはコマンドで起動、停止できる
- /etc/rc.d/rc3.d/ファイル名 [start|stop]

カーネルのカスタマイズ

- Red Hat 5.2では通常は不要
- Red Hat 4.2では、ファイアウォール機能などのために必要
- 不要な機能を削るのは好ましいこと
  - 余分なサービスをしなくなる
  - カーネルがコンパクトになる

ファイアウォール

- アクセス制御に有効だが過信は禁物
  - 適切な設定とメンテナンスは不可欠
- Linuxカーネルもファイアウォール機能を持つ
  - パケットフィルタリングとIPマスカレード
  - ipfwadmコマンド
  - Red Hat 4.2ではカーネルのカスタマイズが必要

ファイアウォール

- 一般的に外部からのアクセスを禁止すべきサービス

tftp	69/udp	snmp	161/udp	
finger	79/tcp	exec	512/udp	
sunrpc	111/tcp,111/udp	login	513/tcp	
netbios	137~139/tcp	shell	514/tcp	など

- telnet、imapなどもファイアウォールで禁止しておくのが望ましい

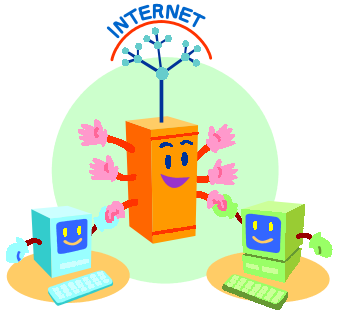
ファイアウォール

- 「オールインワン」も構築可能だが....
  - サーバ自体にファイアウォールを組み込む
  - コストに制約がある場合は有効
  - セキュリティ上はお勧めできない
- ルータのパケットフィルタリング機能、ファイアウォール製品も検討すべき

ファイアウォール

- プライベートアドレスを利用する
  - グローバルアドレス割り当てが制約されている
  - IPマスカレード
    - 任意の数のプライベートアドレスが利用可能に
    - ファイアウォール機能としてもきわめて有効
    - 実用上十分だが一部のプロトコルに制約も

システム運用管理



サーバの運用やセキュリティ対策に「終点」はない。おもにセキュリティ対策の観点から、日常の運用管理をメニュー化してみた。

日常の運用管理項目

- システム資源
  - ディスク、メモリの使用状況、プロセス管理
- ユーザ管理
  - ユーザの追加削除、パスワード管理
- データ管理
  - バックアップ

セキュリティ関連の管理項目

- 「身の丈」でもやっておきたいこと
  - 不正アクセスの検出
  - システムファイル改竄の検出
  - バックアップ
- チェックの自動化は管理を楽にする
  - 管理パターンが決まってきたら、スクリプト化してみよう

ログの点検

- /var/log/messages
  - もっとも多くの情報が書き込まれる
  - FAIL、INVALID等のパターンに注目
- /var/log/secure
  - ログイン履歴等が集められる
  - refuse、warningなどのパターンに注目
  - ログイン履歴はlastコマンドでも把握すべき

ログの点検

- /var/log/maillog
  - メールの履歴が記録される
  - このファイルの分析は難しい
- /var/log/httpd/access_log
  - WWWサーバへのアクセス履歴
  - “_40”、”phf”などのパターンに注目

ログ監視の自動化ツール

- ログファイルモニタ swatch
- 常駐してログファイルをリアルタイム監視
- パターンを検出したらメールなどで通知
- 定義ファイルの例

```
/FAILED/        mail=admin
/INVALID/       mail=admin
```

システムファイルの改竄監視

- 設定ファイル、実行ファイルは頻繁に書き換えられるものではない
- 管理者が知らない書き換えは、不正侵入の恐れを示す
- Tripwireが有名
  - 指定したファイルの「指紋」のデータベースを作る
  - 定期的に実ファイルの「指紋」と照合する

バックアップ

- インストール直後のバックアップ
  - システム全体(万一の修復が楽になる)
  - /etc/ (初期設定値、とくに重要)
- 定期的なバックアップ
  - /etc/ (設定を変更したとき)
  - システム全体(ソフトウェアをアップデートしたとき)
  - ユーザデータ

情報収集の方法

日常の運用管理に反映するために、情報収集は欠かせない。
情報源をいくつか紹介する。

CERT

- http://www.cert.org
- コンピュータセキュリティに関心を持つインターネットユーザの情報集約センター(1988年設立)
  - カーネギーメロン大学に設置
  - CERT Advisoriesというメーリングリスト
    - セキュリティ上の問題と対策を速報してくれるサービス
  - 過去のCERT AdvisoriesはFTPで公開

CERT

- **過去のログ**
  - CERT Advisories
    (ftp://info.cert.org/pub/cert_advisories/)
  - CERT Bulletins
    (ftp://info.cert.org/pub/cert_bulletins/)
- **検索ページ**
  - CERT Advisories
    (http://www.voj.toda.saitama.jp/cert-ca.shtmlなど)
  - CERT Bulletins
    (http://www.voj.toda.saitama.jp/cert-vb.shtmlなど)

情報処理振興事業協会(IPA)

- http://www.ipa.go.jp/index-j.html
- **ウィルス、チェーンメールなどの情報も対象としたコンピュータセキュリティ対策のページがある**
  - (http://www.ipa.go.jp/SECURITY/index-j.html)

コンピュータ緊急対応センター (JPCERT/CC)

- http://www.jpcert.or.jp/
- **不正なシステム侵入に対する緊急対応を中心に、インターネットセキュリティの情報収集・分析、再発防止策の検討、セキュリティ技術の教育・啓発活動を行っている組織**
- **「情報提供用メーリングリスト」も運営**

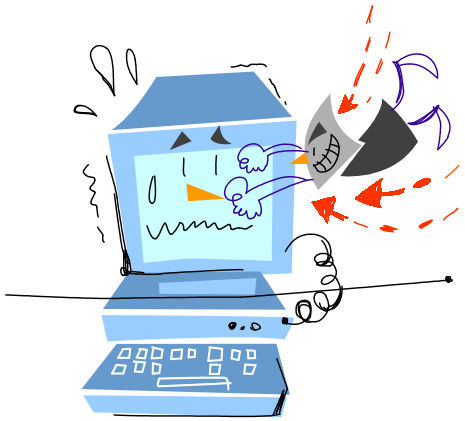
メーリングリスト

- linux-security-jp
  - とくにLinuxユーザを意識してセキュリティ関連の話題を扱っている
  - CERT Advisoriesその他の情報もフォワードされている
  - http://www.3ware.co.jp/opensoc/index.html

アップデートモジュールの入手

- Red Hat社のFTPサーバから入手可能
  - ftp://ftp.redhat.com/
  - 世界中のFTPサーバでもミラーされている
- CERT Advisoriesなどを通じてアップデート情報が入手できる

最近のセキュリティ動向から



phfスクリプト、ポートスキャン、
不正侵入、サービス不能攻撃、
メール不正中継など、最近のア
タック動向と対策を紹介する。

ポートスキャン

- 最近もっとも多い不正アクセス
- telnet、pop3、bindなどのポートをチェック
- ポートが空いていたら別のツールでさらにアクセスされることがある

不正侵入の実態

- ポートスキャンの多くが、踏み台にされたサーバからきている
  - 残念ながらLinuxサーバが多い
- 最近はnamedへのアタックが多いようだ
  - 公開されているツールで、root権限でアクセス可能になる
- 情報収集とこまめなアップデートが必要

DoSアタック

- 標的サーバを動作不能にする
- 手法はさまざま
  - 大量のメールパケットを送りつける
  - 不正なパケットを送りつける
  - サーバプログラムやOSのセキュリティホールを衝く

DoS: Denial of Service

SPAMとメール不正中継

- 一方的に送りつけられてくるメール
  - 送付先が数万、数十万に及ぶことも
  - 第三者からのメール中継を受け付けるサーバ(オープンリレーサーバ)が狙われる
- 不正中継対策は必須
  - /etc/mail/ip_allow、/etc/mail/relay_allow (5.2)

SPAMとメール不正中継

- /etc/mail/ip_allow
  - メールの正当な送信元アドレスのリスト

```
127.0.0.1
192.168.0
```

- /etc/mail/relay_allow
  - 外部から受け取るメールのドメイン名

```
mydomain.co.jp
```