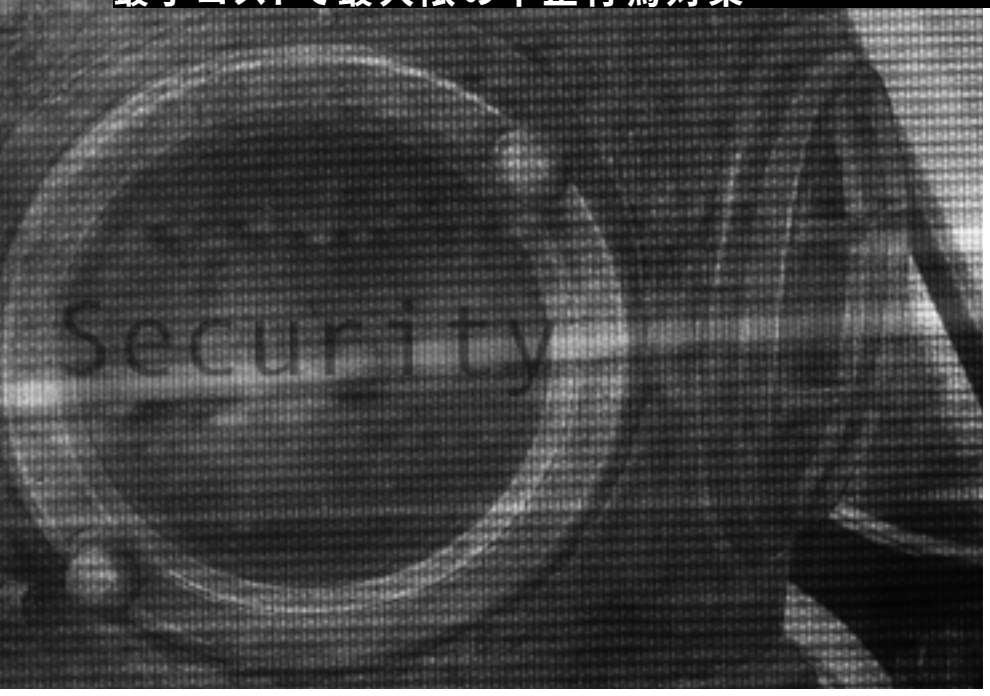
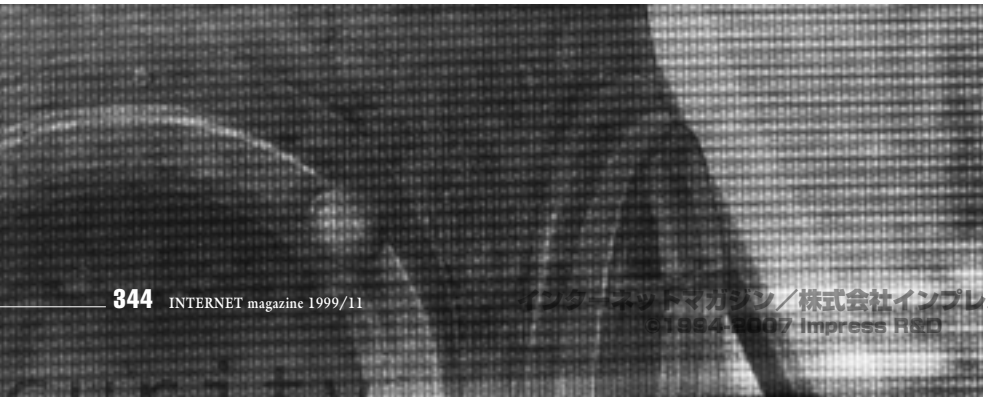


実践Linuxセキュリティー講座

tcp_wrapperはネットワークからシステムへのアクセス制御を行うだけではなく、ログを記録したり何らかのアクションを起こしたりする機能も持っています。前回はtcp_wrapperのアクセス制御のセットアップを行いましたが、今回はログの扱いかた、警告メールの設定、tcp_wrapperの問題点などを考えてみたいと思います。

第10回 tcp_wrapperのさまざまな機能

ソフトウェアコンサルタント　すずきひろのぶ



ログを見てみよう

まずログファイルのありかを探してみます。RedHat 5.2のデフォルトで設定されているtcp_wrapperのログファイルは/var/log/secureです。これはRedHat 5.2での設定であって、ほかのLinuxディストリビューションでは設定が異なるものもありますので注意が必要です。

ここで簡単にLinuxの（UNIXの）システムログのメカニズムを説明します。システムログを取るためにsyslog（コマンド名syslogd）というデーモンが用意されています。システムで動作するプログラムは、プログラム中にsyslog関数を入れることで、このsyslogデーモンを経由して指定のログファイルやコンソールにメッセージを残せます。システムのメッセージ以外にsendmailのような各種デーモンもsyslog経由でメッセージを記録しています。

syslogデーモンの動作のコンフィグレーションファイルは/etc/syslog.confです（リスト❶）。このファイルでいろいろなデーモンが出力するsyslogの設定が行われていますが、今回の話題であるtcp_wrapperのアクセスなどセキュリティー関連の情報は/var/log/secureに記録が残るように設定されています（RedHat 5.2の場合）。なお、オリジナルのtcp_wrapperのソースコードを使ってデフォルト設定のままコンパイルしてインストールした場合は別のファイル名になることがあるので注意してください。

また、tcpdデーモンのエラーなどはセキュリティーの記録としてではなく、/var/log/messagesに出力されます。

ネットワーク経由でログを記録する

syslogはネットワーク経由でほかのシステムからのシステムログを記録する機能を持っています。この機能を使えばログをほかのマシン上に残すことができます。

本連載が想定しているSOHO環境では、サ

ーバーが1台なのでサーバー上でログを管理する手間はそれほど大きいとは言えませんが、いくつものホストを管理する必要がある場合、個々のホスト上の/var/log/secureを1か所で同時に監視し、ログ情報をアーカイブするほうが簡単に管理できます。

たとえばサーバー上で記録していたtcp_wrapperのログを、別のlogger（ホスト名）というマシンで記録する設定はリスト❷のとおりです。

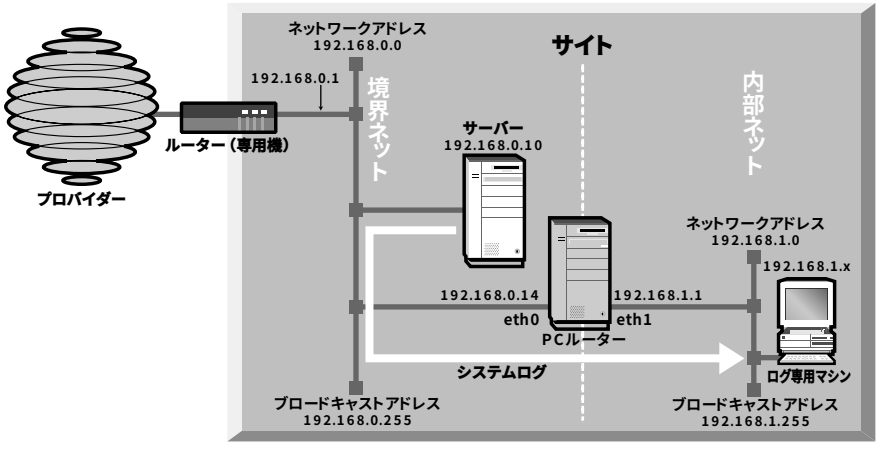
リスト❶

```
・/etc/syslog.confの設定（デフォルト）

# The authpriv file has restricted access.
authpriv.*                                              /var/log/secure
```

リスト❷

```
Step 1: ログを取るマシン側のsyslogを-rオプションをつけて再起動します。これによって外部からのログの要
        求を受け付けるようになります。

・起動スクリプト/etc/rc.d/init.d/syslogの20行目あたり

....
case "$1" in
  start)
         echo -n "Starting system loggers: "
         daemon syslogd -r ←――――ここに-rオプションを追加
         daemon klogd
...

・syslogdを再起動する

# su
# /etc/rc.d/init.d/syslog restart

Step 2: サーバー側の/etc/syslog.confの記録ファイルの記述を相手先マシンに変更します。

        ここでログを記録する側のマシン名を仮にloggerという名前だとすると次のようになります。

（例）/etc/syslog.confの設定の場合
クライアントのマシンで記録

# The authpriv file has restricted access.
authpriv.*                                              @logger
                                                  ―――――
                                                  @を先頭に、つづけてホスト名
Step 3: サーバー側のsyslogにHUPシグナルを送る。

# su
# kill -HUP `cat /var/run/syslogd.pid`
```

図❶　サイトのネットワーク構成図



また、システム管理者が「サーバー上の情報がすべて破壊されたとしても各種のログだけは保護しておきたい」と考える場合も、ネットワークを経由してログを取る方法が役に立ちます。たとえば、内部ネットワークに専用マシンを別途用意しておき、そこにサーバーのログを記録するようなことが考えられます（図❶）。

とは言っても、SOHOレベルでここまでのセキュリティーを要求するのは、あまりにも管理者の負担が大きいので推奨はしません。ただ、複数のシステムのログ情報を手元のマシンで一括管理するほうが楽だというふうに考えるのなら、ここで紹介したネットワーク経由でログを記録する方法を試してみる価値はあると思います。

PCルーターの存在を忘れずに

境界ネットから内部ネットワークへsyslogのデータを転送している場合、syslogのサービス（514/udp）をネットワークの境目のPCルーターでフィルタリングしているのを忘れて悩む場合がありますので要注意です（ちなみに筆者はハマリました）。

ログの中身を見てみよう

では、/var/log/secureのログを見てみましょう。ここにはたくさんの記録が残されています。まずは、典型的なフォーマットの解説です。リスト❸はrloginの接続によって発生したログです。

正常にログインできた場合、これ以上の情報はログに出ませんが、パスワードなどがエラーでログインエラーになった場合は、先の情報に続いて認証エラーのログが記録されます（リスト❹）。

このログを例に取って考えてみましょう。これはネットワーク経由でrootにログインしようとして失敗したことを物語っています。Linuxのデフォルトではネットワーク経由で直接rootにはログインできないのでエラーにな

るのが当然です。このようなアクセスは非常に不自然です。もし、インターネット側からアクセスされているなら、不正アクセスを狙ったものだと考えていいでしょう。

ただし、インターネット側からアクセスされている場合の問題点は、不正アクセスがあったことではありません。自分のサイトのセキュリティー設定が不完全であることが問題なのです。もし、きちんとセキュリティーのセットアップができていれば、インターネット側からアクセスしてログインを試みるような行為は不可能なのです。

まず考えられるのは、/etc/hosts.denyと/etc/hosts.allowの設定が誤っていることでしょう。これらの設定が正しければインターネット側からログインしようとアクセスしたところで、ログインプロンプトが出てきません。/etc/hosts.denyが正しく動作していてアクセスできないときはリスト❺のようにログが記録されます。

詳しい説明は次回以降になりますが、次にインターネットに接続するルーターによりインターネット側からの防御を行います。

インターネットに接続するルーターでフィルタリングを行い、外部からのrloginやtelnetを通さない設定にします。したがって外部のIPアドレスからの接続がログに記録された場合、設定エラーなどによりルーターのパケットフィルタリングが有効になっていないことが考えられます。この場合、まずルーターのパケットフィルターが正しく動作しているかを確認する必要があります。ルーターの設定についてはこの連載でも将来まとめてやる予定なので、今月はこのくらいにしておきます。

警告をメールで送るには

今までは単にログを記録するだけでした。これでは、ログを過去の記録として扱い、何かトラブルがあったときだけログを参照するか、あるいは管理者が気を利かせて定期的にログを読む必要があります。一方で、何か不審な現象が発見されたらすぐに通知が欲しい場合もあります。このような場合、警告をメールで通知するといった設定が可能です。

リスト❻はrloginで接続してくるとそのたびにhironobu@h2np.netへメールを送る設定の典型的な例です。なお、リスト❼は警告として送信されたメールの例です。

ここで注意すべき点は、この設定がDoS攻撃（サービス不能攻撃）に使われる可能性があるということです。何者かがサーバーに多数のrloginをかけてきた場合、このままでは管理者に大量の警告メールが送られる事態になります。ここではin.rlogind：ALLに対してのみの設定ですが、もしALL：ALLに対して警告メールを送信する設定にしていれば、ポートスキャンが成功した場合、おびただしい数の警告メールが送られることになるでしょう。

設定ファイルを見ていただけるとわかりますが、“ spawn=(シェルコマンド) ” の形式でシェルコマンド列を書くことができます。今回の例は、典型的な設定ということでmailコマンドをダイレクトに記述していますが、自作の専用管理プログラムを設定することも可能です。あるいはメールの送り先を管理者ではなく自動管理プログラムへのアドレスにすることもできます。そこからさらにポケベルへ発信するなどのアイデアも考えられます。

バナーは出さないほうがよい

tcp_wrapperの解説には必ずといっていいほどバナー（banner）の利用方法が書かれています。これはあらかじめメッセージをファイルに用意しておき、禁止されたデーモンにアクセスがあったときにクライアントにメッセージを送るものです。

まず適当なディレクトリー（/etc/tcpdbanners）を作成し、そのディレクトリーの中にメッセージファイルを用意します。メッセージファイル名はデーモン名と同じにします（今回の例であればin.rlogindという名前になります）。そのファイルの中にメッセージを書き込みます。そして先ほどのin.rlogindの設定に“ banner=ディレクトリー名 ”の記述を加えれば完了です。

リスト❸

```
Sep  1 22:26:31 server1 in.rlogind[1641]: connect from 192.168.1.8
      (1)       (2)       (3)      (4)      (5)          (6)
```

(1) 記録された日時
(2) （サーバーの）ホスト名
(3) デーモン名
(4) 起動されたデーモンのプロセスID（この場合 in.rlogindのプロセスID）
(5) 何が行われたか

リスト❹

```
Sep  1 23:43:13 server1 in.rlogind[5584]: connect from 192.168.0.10
Sep  1 23:43:20 server1 login: FAILED LOGIN 1 FROM fromhostname FOR root,
Authentication failure
```

（メッセージの一部）

```
  login: FAILED LOGIN 1 FROM black FOR root, Authentication failure
   (1)                  (2)       (3)       (4)            (5)
```

(1) 何で問題が発生したか ―― login を行おうとした
(2) 失敗した回数 ―― 1回
(3) 接続してきたマシン名（あるいはIPアドレス）―― blackというマシン名
(4) どのアカウントに対してか ―― rootに対して
(5) 何が失敗したか ―― 認証エラー（パスワードエラー）

リスト❺

```
Sep  2 01:43:37 server1 in.telnetd[1693]: refused connect from 192.168.1.10
                                                               (注)
```

(注) サイト内での接続テストを行ったので、この内部ネットワーク内のIPアドレスになっています。

```
in.rlogind:ALL:\
spawn=(...)&:\
banners=/etc/tcpdbanners
                ディレクトリー名
```

しかし、筆者はこのバナーを使って警告する方法は、あまりよい方法だとは思いません。警告も何も行わずに黙殺するのが一番だと考えます。よくセキュリティー本（特にアメリカの本）に「あなたのアクセスはログに記録されています」、「許可なくコンピュータを使うと州法及び連邦法に抵触します」などのメッセージを出す例が出ています。しかし、これは不用意に相手を挑発する行為ではないでしょうか？　このような不要な挑発は何の利益も生まないと筆者は考えます。

逆fingerもやらないほうがよい

この手の本には、相手を探すためにアクセス元のホストに対して自動的にfingerをかける方法も紹介されていますが、この方法もすすめられません。

tcp_wrapperの配布パッケージにはsafe_fingerという通常のものよりも安全なfingerが入っています。これを使ってアクセス元のホストにfingerを行い、相手のホストの利用状況や利用中のユーザー情報を取得し、それを管理者へ自動的にメールするという方法が紹介されている本を読んだことがあります。

しかし、これは一歩間違えれば不正アクセスと思われかねない行為ですし、また外部から内部ネットワークへのfingerアクセスを禁止しているサイトは多数あります（本連載もそのポリシーでサイト運営をしています）。またウィンドウズやマッキントッシュといったいわゆるパソコンOSにはfingerデーモンなどはデフォルトで用意されていないので、おおよそパソコンユーザーからのアクセスには役に立ちません。犯人探しができるかと考えてやってみても、犯人をみつけるどころか逆に犯人扱いされるのがオチです。筆者にはこのような行為は利益よりも不利益のほうが多いように感じます。

今後の課題

想定しているSOHO環境では、境界ネットはインターネット側（および内部ネット側）と隔離するルーターで守られています。ルーターのフィルタリングが正しく稼働しているならば、不正アクセスが境界ネット上にあるサーバーに届く可能性は非常に小さいものです。tcp_wrapperのような設定はフェールセーフのための設定だと言えるでしょう。

一方、大学や企業のように内部に大量にマシンを抱えているようなネットワーク利用環境では、内部の利用者からの不正アクセスが十分に考えられます。そのような状況ではtcp_wrapperは力強い味方です。

ログに関するtcp_wrapperの最大の課題は大量のログデータだと言えるでしょう。SOHO環境だとのんびりしたものですが、企業や大学のように台数も多く利用者も多い環境だとtcp_wrapperのログも莫大な量になることでしょう。こうなると情報の海に溺れてしまって監視の役目も果たせなくなる可能性があります。大量のログ情報をどう整理するか、また、その情報から何を読み取るのか、タイムリーにその情報を生かせるかなど、難しい問題が山積します。tcp_wrapperのログを自動的に解析するようなツールや簡単に使えるセットアップツールなどが出てくるようになれば非常に役立つことでしょう。

次回は安全なリモートシェル、SSHを

次回はSSHを取り上げます。telnetやrlogin、rshの代替品として使われるSSHは、通信路を暗号でプロテクトする安全なリモートシェルです。SSHはRPMパッケージでは提供されていないので、まずはソースコードの入手とコンパイルから入っていきます。

リスト❻

```
/etc/hosts.denyの設定

in.rlogind:ALL:\
spawn=( echo "%s %u@%h(%a)" | /bin/mail -s "TCPD ALERT" hironobu@h2np.net)&
ALL:ALL                  (注)

%s ―――――― デーモン名@サーバIPアドレス
%u ―――――― クライアント側ユーザ名(わからなければunknown)
%h ―――――― クライアント名(わからなければクライアントIPアドレス)
%a ―――――― クライアントIPアドレス
(注) ここの@は"ユーザ名@クライアント名(クライアントIPアドレス)"とメールアドレス風に表すために使っているだけです。
```

リスト❼

```
警告として送られたメール

>From root  Mon Sep  6 03:24:15 1999
Date: Mon, 6 Sep 1999 03:24:15 +0900
From: root <root@server1.h2np.net>
To: hironobu@h2np.net
Subject: TCPD ALERT

in.rlogind@192.168.1.10 unknown@black(192.168.1.32)
    (1)          (2)        (3)   (4)      (5)

(1) rloginデーモン
(2) 192.168.1.10のIPアドレスであるサーバ(server1)
(3) ユーザ名不明
(4) ホスト名black
(5) blackのIPアドレス
```

［インターネットマガジン　バックナンバーアーカイブ］ ご利用上の注意

このPDFファイルは、株式会社インプレスR&D（株式会社インプレスから分割）が1994年～2006年まで発行した月刊誌『インターネットマガジン』の誌面をPDF化し、「インターネットマガジン　バックナンバーアーカイブ」として以下のウェブサイト「All-in-One INTERNET magazine 2.0」で公開しているものです。

http://i.impressRD.jp/bn

このファイルをご利用いただくにあたり、下記の注意事項を必ずお読みください。

- 記載されている内容（技術解説、URL、団体・企業名、商品名、価格、プレゼント募集、アンケートなど）は発行当時のものです。
- 収録されている内容は著作権法上の保護を受けています。著作権はそれぞれの記事の著作者（執筆者、写真の撮影者、イラストの作成者、編集部など）が保持しています。
- 著作者から許諾が得られなかった著作物は収録されていない場合があります。
- このファイルやその内容を改変したり、商用を目的として再利用することはできません。あくまで個人や企業の非商用利用での閲覧、複製、送信に限られます。
- 収録されている内容を何らかの媒体に引用としてご利用する際は、出典として媒体名および月号、該当ページ番号、発行元（株式会社インプレスR&D）、コピーライトなどの情報をご明記ください。
- オリジナルの雑誌の発行時点では、株式会社インプレス R&D（当時は株式会社インプレス）と著作権者は内容が正確なものであるように最大限に努めましたが、すべての情報が完全に正確であることは保証できません。このファイルの内容に起因する直接的および間接的な損害に対して、一切の責任を負いません。お客様個人の責任においてご利用ください。

このファイルに関するお問い合わせ先
株式会社インプレスR&D
All-in-One INTERNET magazine 編集部
im-info@impress.co.jp