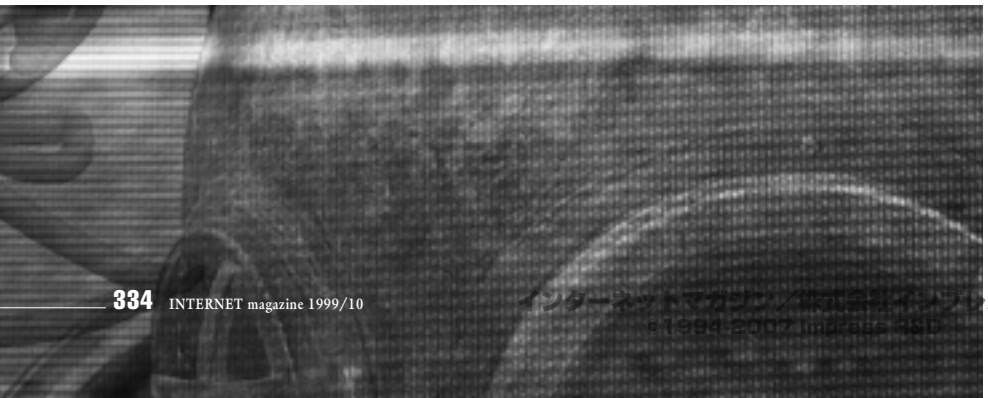
# 実践 Linux セキュリティー講座

今回はサーバー側（ホスト側）のアクセス制御の１つであるtcp_wrapperを説明したいと思います。tcp_wrapperは、既存のネットワークサーバープログラムを改造することなく、ラップする（包む）ことによってアクセス制御や監視を行う、優れたツールです。現在ではLinuxのいずれのディストリビューションでもデフォルトで入っています。それだけ現在ではポピュラーなツールとなっています。

## 第9回 tcp_wrapperでアクセスを制御する

ソフトウェアコンサルタント　すずきひろのぶ



### tcp_wrapperとは？

tcp_wrapperは外部からのTCP/IPのアクセス制御を行うプログラムです。

tcp_wrapperは、"tcp_wrapper"という名前のコマンドがあるわけではなく、1つのツール群、あるいは環境を指しています。各種デーモンプログラムを起動するときにラッパー（包み）として使われるtcpdもそうですし、また、関数ライブラリーとして用意しているlibwrap.aを使用してプログラムに組み込むような場合もtcp_wrapperを使用していることになります。

このように、各種デーモン（サーバー）プログラムにコンパイル時に組み込んで使うこともできますが、今回はLinuxに初めからセットアップされている、inetdで起動されるデーモンプログラムについて説明します。

まずLinuxの/etc/inetd.confを見てください。その中にリスト❶のような記述があります。

この/usr/sbin/tcpdが各種デーモンプログラムに対してtcp_wrapperを実現するためのプログラムです。

本連載では取り上げませんが、inetdのセキュリティーを強化しアクセス制御を行うツールとしてxinetdというものもあります（Linuxの主なディストリビューションでは標準で入ってはいないようです）。基本的にはtcp_wrapperと同じ考え方です。

### inetdはデーモンを起動させる

inetdのことをマニュアルなどでは「インターネット“スーパーサーバー”」といった言葉で表現しています。簡単に言えば各種デーモンプログラムを起動させるサーバープログラムだと思えばいいでしょう。

登録されているデーモンが使うポートを監視し、クライアントから接続があればデーモンを起動して、そのクライアントと接続させます。クライアント側からはinetdを意識することなくデーモンに接続します（図❶）。

バックグラウンドにデーモンとして事前に

（たくさんの）プログラムを起動させておくのではなく、接続要求があった時点でプログラムを起動する方法をinetdは提供しています。これは、いつ使うか分からないプロセスを常時動作させておき計算資源を浪費するよりも、必要な時に起動させたほうが得策という考え方に基づいています。

その反対に、利用頻度が非常に高いプログラムを、すぐに使えるようにしているのがhttpdの考え方です。すでに接続待ちになっている複数のhttpdデーモンがあらかじめ用意されているので、クライアントの接続要求があれば、すぐにサービスを開始できます。もちろん長時間だれもアクセスしない状態では計算資源を浪費するだけになってしまいます。

## アクセス制御ルールを記述する

現在のRedHatでは/usr/sbin/tcpdがデフォルトで/etc/inetd.confにセットされています。また、どのディストリビューションにも入っています。このため、コンパイルなどの説明は省きインストールされていることを前提に話を進めたいと思います。

アクセス制御ルールの記述を行うためのファイルは次の2つです。

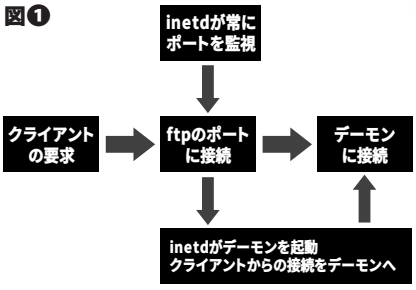
・/etc/hosts.deny
　……アクセス禁止のルールを記述

・/etc/hosts.allow
　……アクセス許可のルールを記述

デフォルトでは、何も記述がありません（少なくともRedHat Linux 5.2では）。初期状態ではインストールされてはいるものの、何もアクセス制御を行っていない状態、つまり、自由にアクセスできる状態です。

アクセス制御のルールを記述するファイル/etc/hosts.denyや/etc/hosts.allowをホストアクセス制御ファイル（host access control files）と呼びます。ドキュメント中あるいはマニュアルなどでは単にhosts_accessという表現をしています。さて、この記述ルールですが非常に簡単です。ルールはリスト❷の

図❶




ようになっています。たとえば

ALL:ALL

という表現であれば「すべてのデーモンに対して：すべてのクライアントに対して」という意味になります。これがhosts.denyにあれば、すべてのデーモンは、すべてのクライアントからのアクセスを拒否することになります。

## アクセス制御ルールを確認する

これから/etc/hosts.deny記述を変更して、それが有効かどうかチェックする作業をします。まずはチェックのためのコマンド/usr/

### サーバーとデーモンの違い

デーモン（Deamon）は「バックグラウンドプロセスとして動作し、サービスを提供できるようになっているプログラム」というぐらいの意味になります。時間がくると登録してあった命令を実行するcronなどもデーモンと呼ばれます。

一方、サーバーは「ネットワーク接続によりクライアントにサービスを提供すること」というような意味になります。外部からのネットワーク接続を待つためには、サーバーのプログラムがバックグラウンドプロセスとして動いている必要があります。厳密には、サーバーはデーモンではありますが、デーモンは必ずしもサーバーとは限りません。しかし、本文中では、デーモン（デーモンプログラム）とサーバー（サーバープログラム）という言葉を混在させていますが、同じような意味だと思ってください。

リスト❶

```
% more /etc/inetd.conf
...
ftp       stream    tcp       nowait    root      /usr/sbin/tcpd      in.ftpd -l -a
telnet    stream    tcp       nowait    root      /usr/sbin/tcpd      in.telnetd
gopher    stream    tcp       nowait    root      /usr/sbin/tcpd      gn
...
```

リスト❷

```
デーモンリスト:クライアントリスト [ : シェルコマンド ]
                                   オプション
オプションなどの使い方は次回説明します。
```

リスト❸

```
$ su                                          ← root権限になる
# /usr/sbin/tcpdchk -v                        ← tcpdchkの実行
warning: /etc/inetd.conf, line 35: gn: not found in /usr/sbin: No such file or directory
...
warning: /etc/inetd.conf, line 87: /usr/sbin/in.identd: not found: No such file or directory
Using network configuration file: /etc/inetd.conf
```

sbin/tcpdchkを説明しましょう。これはtcp_wrapperがどのような状態になっているかをチェックするためのコマンドです。

まず、ROOT権限になって、まだ何もしない状態でtcpdchkを動かしてみてください。たくさんのワーニングが現れるはずです（リスト❸）。'No such file or directory'というメッセージが出ているものは、/etc/inetd.confにエントリー記述があるにもかかわらず、その実体であるデーモンプログラムが存在していないという警告です。

これは大きな問題ではありません。インストール時にシステムにセットアップされる/etc/inetd.confと実際のインストールには若干の差があります。そのために警告が現れているだけです。デーモンプログラムがインストールされていることは、セキュリティー的には問題を起こしませんが、存在していないエントリーはコメントアウトしておいてください。

## 不必要なエントリーは無効に

/etc/inetd.confの中から将来使う予定のないデーモンのエントリーをコメントアウトしてしまい、実行できないようにしましょう。

もちろんhosts.denyでもアクセス制御をしますが、もっとも安全なのは不必要なエントリーを無効にすることです。何かのオペレーションミスで偶然に使えてしまうような状況を避けるためです。

この部分に関しては個々のサイトがどのようなデーモンを使うかによって異なってくるので、一概に「○○と△△のエントリーは無効にすること」とは言えません。自分のサイトのセキュリティーポリシーに従って無効にしてください（リスト❹）。

## ドメイン全体に公開するには

本連載はフェイルセーフの方法をとっているので、まず全部を接続不可の状態にしておいてから必要なデーモンのみ、接続を許可する範囲に限って公開する方針をとります。洩れがあっても「接続不可」の安全な方向に作用することが期待できるからです。

まずドメイン全体を許可するという基本の書き方を説明します。大学や企業のようにネットワークが整備され、（内側のネットワークであっても）多数のマシンからアクセスがある状況では、ドメイン全体にデーモンを公開する場合があるでしょう。そのような場合に有用です。

**例 ftpをh2np.netドメイン全体に許す**
**in.ftpd: .h2np.net**

ドメイン全体に対し（tcpdが設定している）デーモンへのアクセスを許すにはデーモンリストの部分を“ALL”と記述します。少し大ざっぱな管理になりますので、注意が必要です。できるだけ本当に必要なデーモンのみに限っていく方向で考えてください。これはあくまでも一例ということで御理解ください。

**例 すべてをh2np.netドメイン全体に許す**
**ALL: .h2np.net**

ドメイン名での記述を使わず、直接ネットワークアドレスを記述する方法も有効です。SOHOだと内部向けのDNSが用意されていない場合もあるので、このネットワークアドレ

**リスト❹**

**/etc/inetd.conf**

これは一例です。サイトのセキュリティーポリシーに従って無効にしてください。

```
# Shell, login, exec and talk are BSD protocols.
#shell stream   tcp    nowait   root   /usr/sbin/tcpd  in.rshd
#login stream   tcp    nowait   root   /usr/sbin/tcpd  in.rlogind
#exec  stream   tcp    nowait   root   /usr/sbin/tcpd  in.rexecd
#talk  dgram    udp    wait     root   /usr/sbin/tcpd  in.talkd
#ntalk dgram    udp    wait     root   /usr/sbin/tcpd  in.ntalkd
```

下線部分は新たにコメントアウトした部分です。

shellとloginを無効にした理由:
- tcp_wrapperで制御されているtelnetでのアクセスを用いるため
- 今後SSHをインストールし代替して利用するため

talkとntalkを無効にした理由:
- talkもntalkも利用することがないので

**リスト❺**

**/etc/hosts.allow**

例 内部ネットワーク192.168.1.*からのみftpのアクセスを許す

```
in.ftpd: LOCAL,192.168.1.
```

この最後の”.”は忘れやすいので注意

これは192.168.0.10である自分のマシンを許すため

例 特定のネットワークアドレスのみからのtelnetのアクセスを許す

```
in.telnetd: LOCAL,192.168.1.8
```

このクライアントのみtelnetを許す

カンマはリストを区切る。

**/etc/hosts.deny**

```
ALL:ALL
```

hosts.allow→host.deny
まずhosts.allowの内容が優先され、マッチするものが許可される。
次にhosts.denyが有効になり、マッチするものが拒否される。

スを記述する方法は有用だと思います。また、想定しているようなSOHO環境（図❷）では、ネットワークアドレスが異なるセグメントは、内部ネットワークと境界ネットワークの2つしかないので記述も簡単ですし、何かしらDNSに障害があっても継続して利用できます。hosts.allowとhosts.denyの例を使って、簡単に解説してみましょう（リスト❺）。in.ftpdというのはftpサーバの実行プログラム名です。LOCALという記述は、自分のマシンからのアクセスを許すという意味です。もし、このLOCALがなく、許可するIPアドレスのなかにも自分のマシンが含まれていない場合は自分のマシンであってもアクセスが拒否されます。ネットワークの指定は192.168.1.というようにドットで終わらせます。

## 制御ルールを再チェックする

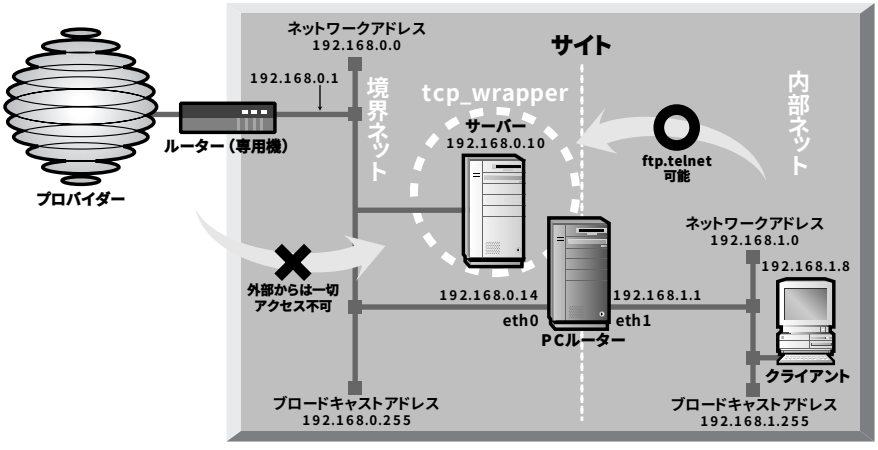
さて、これでhosts.denyとhosts.allowによって最小限の範囲からしかアクセスできないようになりました。これがきちんと有効になっているか、再度tcpdchkを使ってチェックしてみたいと思います。ここでは、ftpとtelnetの接続を内部ネット側（192.168.1.*）と自分自身にのみ許しています（リスト❻）。

こんどは別の角度からチェックしてみましょう。現在の設定が正しいならば192.168.1.10はアクセスできて192.168.0.99はアクセスできないはずです。このように特定のアドレスからアクセスが可能かどうかをチェックするにはtcpdmatchというコマンドを使います（リスト❼）。

## 次回はtcp_wrapperの続きを

次回は、tcp_wrapperのログに関する情報や、許可されていないポートへの接続があったときに自動的に報告が行われるといった機能について説明します。

図❷　サイトのネットワーク構成図




リスト❻

```
# /usr/sbin/tcpdchk -v
Using network configuration file: /etc/inetd.conf

>> Rule /etc/hosts.allow line 7:
daemons:  in.telnetd  ← telnetデーモン
clients:  LOCAL 192.168.1.  ← 自マシンと内部ネットワーク
access:   granted  ← 許可

>> Rule /etc/hosts.allow line 8:
daemons:  in.ftpd  ← ftpデーモン
clients:  LOCAL 192.168.1.  ← 自マシンと内部ネットワーク
access:   granted  ← 許可

>> Rule /etc/hosts.deny line 9:
daemons:  ALL  ← すべてのデーモン
clients:  ALL  ← すべてのクライアント
access:   denied  ← アクセス拒否
```

リスト❼

```
# /usr/sbin/tcpdmatch  in.telnetd  192.168.1.10
                       デーモン名   アクセス元アドレス

client:   address  192.168.1.10  ← 指定したクライアントIPアドレス
server:   process  in.telnetd  ← 指定したデーモン名
matched:  /etc/hosts.allow line 7  ← マッチしたルール
access:   granted  ← 許可している

# /usr/sbin/tcpdmatch  in.telnetd  192.168.0.99
                                   境界ネット上の架空のアドレス

client:   address  192.168.0.99  ← 指定したクライアントIPアドレス
server:   process  in.telnetd  ← 指定したデーモン名
matched:  /etc/hosts.deny line 9  ← マッチしたルール
access:   denied  ← 拒否している
```

# ［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