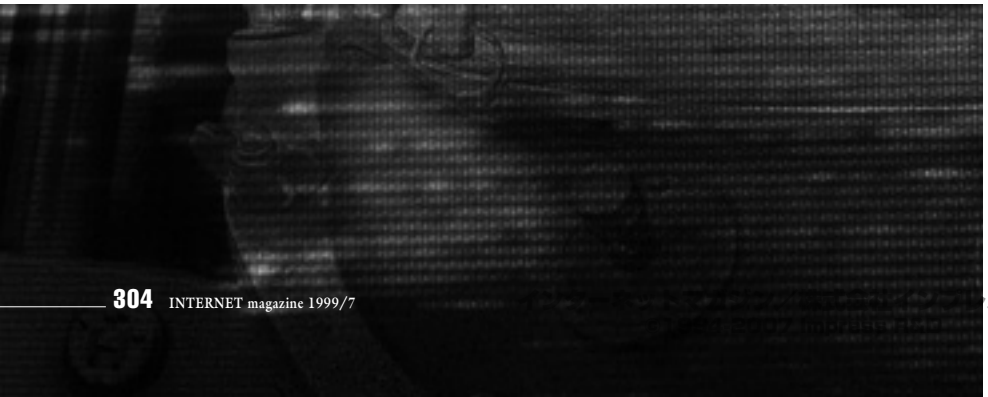
実践Linuxセキュリティー講座

SOHO環境を想定したPCルーターによるフィルタリングの設定は前回で終了しました。しかし、何度も述べたとおり、これまでの解説は一例であって、解説したとおりの設定がすべての環境で万能であるとは考えていません。そこで今回は、前回で足りなかった部分を補足するような形で、LinuxによるIPファイアーウォールとコマンド「ipfwadm」について説明を加えます。

第6回 ipfwadmを極める

ソフトウェアコンサルタント　すずきひろのぶ



ipfwadmの4つの機能

Linuxのカーネルには IPパケットのフィルタリングによるIPファイアーウォール機能が用意されています。今回想定しているRedHat Linux 5.2のカーネルは、すでにIPフォワードの機能が組み込まれていて、簡単な設定でファイアーウォールの機能が有効になります。Linuxにはいくつかのディストリビューションパッケージがありますが、すべてのパッケージにこの機能が初期状態でカーネルに組み込まれているかどうかはわかりません。もしRedHat以外のパッケージを使っているならば、IPフォワードの機能を有効にするために、再コンパイルが必要なのかどうかをチェックしてください。

Linuxマシンに対してIPパケットのフォワードの指定を行っただけでは、マシンはネットワークを橋渡しするルーターの役割しかしていません。これをipfwadmのようなコマンドを使い、制御の設定をして初めてIPパケットフィルタリングによるファイアーウォールの役目を果たすようになります。ipfwadmはLinuxのカーネルに組み込まれているIPファイアーウォールの機能を外部からコントロールするためのインターフェイスの役目をしています。

ipfwadmを使ったIPファイアーウォールは次の5つの機能から成り立ちます。

❶IPパケットの記録（アカウンティング）
…通過したIPパケットの記録を取る
❷ファイアーウォールへの入力パケットの制御
…コンピュータに入ってくるIPパケットを制御する
❸ファイアーウォールからの出力パケットの制御
…コンピュータから出ていくIPパケットを制御する
❹ファイアーウォールを中継するパケットの制御
…コンピュータを通過していくIPパケットを制御する

❺IPマスカレード機能
…コンピュータを通過していくIPパケットのアドレス情報を書き換える

今回はこれらに関して説明しますが、その前にTCP/IPのアドレスとポートの仕組みをざっとおさらいしておきましょう。

IPパケットは3つの情報を持つ

TCP/IPをごく単純なモデルを使って説明します。前回も簡単に説明しましたが、TCP/IPはプロトコルの種別とIPアドレス、ポート番号で接続する相手を区別しています。IPパケットのヘッダー部分には次のような情報が入っていて、この情報によってIPパケットはどこのコンピュータのどのプログラムにたどり着けばよいかを区別します。

・プロトコル種別（TCPかUDPか）
・送信元IPアドレス／送信元のポート番号
・送信先IPアドレス／送信先のポート番号

プロトコルの種別は相手と通信する方法の規定になります。もちろん、双方で同じプロトコルを使わなければ通信ができません。

皆さんがご存じのとおり、IPアドレスはネットワークに接続されているコンピュータを区別するためのもので、IPパケットがそこに到達するために必要な情報です。

ポートというのはプログラムの通信の窓口だと考えればよくわかると思います。コンピュータの上でプログラムは自分が通信するための窓口を作り、これを利用して通信を行います。ポートは0から65535までの番号で示します。0から1023まではWell-Knownポートと呼ばれ、どのアプリケーションプロトコルに使用するかが決められています。また、Well-Knownポートはroot権限がなければ使えません。1024から49151までは登録ポートと呼ばれていて、何らかのアプリケーションのために用意されているポートです。49152から65535までは自由に使ってよいポートとなっています。

どのプログラムがどのポートを使っているのかがわからなければ、双方で通信ができませんので、共通の規約が決められています。たとえば、電子メールに使うSMTPプロトコルのポートは25番ですし、ウェブサーバーのHTTPプロトコルのポートは80番です。相手のコンピュータまではIPアドレスを使って識別し、そのコンピュータ上のどのアプリケーションに接続するかは、ポート番号を使って識別します（図1）。

これらの接続状況は「netstat」というコマンドを使って観察できます。リスト❶はnetstatを使って観察しているところです。マシン名clientからマシン名serverへFTPしている状態で、server上でnetstatを実行して接続状況を表示させています。

IPパケットの記録をとる

IPパケットの記録をとる機能のことを普通はアカウンティング（accounting）と呼んでいます。IPパケットの流れはあらかじめ設定したルールに従って記録されます。これらの記録は/proc/net/ip_acctに保存されます。

IPパケットの記録といっても基本的にはIPパケットの送信元と送信先、ポートの区別と通過した流量を記録しているレベルに留まっています。セキュリティーに対してこの記録の上手な使い方があるのかも知れませんが、筆者は残念ながら効果的な使い方は知りません。個別のマシン（IPアドレス）に対しても記録が取れるので、どのマシンがどの程度ルーターを使っているかぐらいの情報はわかります。

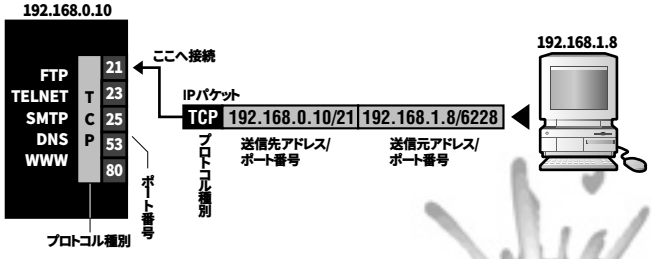
IPパケットの記録と表示の手順は次のようなステップになります（リスト❷）。

Step1　ipfwadmで記録をとる条件を設定する
Step2　ルーターとして使用する（記録中）
Step3　ipfwadmを使って表示する

IPパケットを制御する

この連載では、PCにLinuxをインストールしてルーターにしたマシンをPCルーターと命名しています。図2がこの連載のモデルとなるネットワーク構成です。PCルーターの中身はeth0とeth1という2つのNICを使って内部ネットと境界ネットのルーターの役割を果たし、入力パケットと出力パケット、中継するパケットを制御するようなIPフィルタリングによってファイアーウォールを実現しています。これだけでも専用ルーター機によるIPパケットフィルタリングと遜色のない機能を提供して

図1　IPパケットの持つ情報



リスト❶　netstatを使って通信の状態を見る

```
% netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address   Foreign Address    State
tcp        0      0 server:ftp     client:6228        ESTABLISHED
```

プロトコル種別
ポート番号（サービス名）
ポート番号（サービス名）
自分のマシンのアドレス（マシン名）
相手のマシンのアドレス（マシン名）

います。

PCルーターで制御できるのは、入力パケットと出力パケット、中継するパケットの3つです。この連載では通過させるIPパケットのフィルタリングというのが目的なので、中継するパケットのみを取り上げ制御しています。制御方法に関しては前回の記事を参照してください。

前回は「どんなプロトコルを通すべきか、考慮が必要である」ということを簡単に触れただけですが、通過に関する大枠のポリシーは、内側にも外側にもまず何も通過させないというところから始めます。次に慎重に必要最小限のIPパケットのみを通過させます。どのようなパケットの通過を許すべきかは、次の本が参考になるでしょう。

UNIX&インターネットセキュリティー
オライリージャパン
Simon Garfinkel, Gene Spafford著
山口英監訳　谷口功訳
付録G IPプロトコル一覧 p.p.905-915

リスト❷　IPパケットの記録を取る

Step1　条件の設定

次のコマンドを入力する（スクリプトファイルにして実行してもいい）

```
# /sbin/ipfwadm -A -f
# /sbin/ipfwadm -A out -a  -S 192.168.1.0 -D 0.0.0.0
# /sbin/ipfwadm -A out -a  -S 0.0.0.0 -S  192.168.1.0
# /sbin/ipfwadm -A in -a -S  192.168.1.0 -D  0.0.0.0
# /sbin/ipfwadm -A in -a -S 0.0.0.0 -D  192.168.1.0
```

← このコマンドを実行

●オプション説明

```
-A  [方向]                      ルールを定義する方向の指定。[方向]はin、out、bothのいずれか
-a                              ルールの追加
-S  [IPアドレス*] [ポート番号]  発信元アドレスの指定（ポート番号はなくてもいい）
-D  [IPアドレス*] [ポート番号]  送信先アドレスの指定（ポート番号はなくてもいい）
*IPアドレスは192.168.1.0/24というようにマスクを加えることができる
```

Step2　実際にルーターとして使う

Step3　記録した結果を表示する

次のコマンドを実行する

```
# /sbin/ipfwadm -A -l
```

← このコマンドを実行

```
IP accounting rules
 pkts bytes dir prot source               destination          ports
  117 11764 out all  192.168.1.0/24       anywhere             n/a
  116 11666 out all  anywhere             192.168.1.0/24       n/a
  178  7183 in  all  192.168.1.0/24       anywhere             n/a
  177  7143 in  all  anywhere             192.168.1.0/24       n/a
```

表示結果

pkts：記録したパケットの数
bytes：データ量（バイト）
dir：方向
prot：プロトコル
source：発信元アドレス
destination：送信先アドレス
ports：ポート

プロトコルは指定していないのでTCPとUDPの両方を記録している。TCPでポートの指定がないときportsの表示は‘*->*’となる。ポートの指定をしているときは、*の部分にそのポート番号が入る。

例）

```
# /sbin/ipfwadm -A in -a -P tcp -S 192.168.1.0/24 1024:65535 -D 0.0.0.0/0 80
# /sbin/ipfwadm -A -l
IP accounting rules
pkts bytes dir prot source               destination          ports
 291 30043 in  tcp  192.168.1.0/24       0.0.0.0/0            1024:65535 -> 80
```

IPマスカレードの持つ意味

IPマスカレード機能は、ルーターがIPアドレスの変換とポート番号を変換することにより、外側に向けている1つのIPアドレスを複数のIPアドレスで使えるようにする機能です（図3）。

IPマスカレードの最大の利点はグローバルアドレスを有効に利用できることです。インターネットの世界で唯一しかないグローバルアドレスの組み合わせは約42億になります（ただし、すべてのアドレス空間を有効に使えているわけではないので理論値よりもさらに小さい値になります）。しかし、それだけあってもグローバルアドレスが枯渇する状況が憂慮されています。そこで、IPマスカレードによって少ないグローバルアドレスを有効に使うわけです。

IPマスカレードには副作用として利点と欠点があります。利点としては、内部から外部へのIP接続がないかぎりアドレスとポートの割り振りが行われないため、外部からは内部へアクセスしようがないことです。内部を隠してしまうことと同じことになります。IPマスカレードでは、内部から外部に接続して送ったパケットの応答パケットのみを通します。原理的には、応答パケットを装ったパケットを送り込むという特殊な攻撃ができますが、逆をいえば攻撃としてできるのはその程度しかなく、この点に関してはセキュリティー上有利に働きます。

欠点としては、ポートを変換するのでいくつかのアプリケーションはIPマスカレード越しに使えない可能性があることです。しかし、今では大きな問題ではなくなってきています。以前は、pingコマンドやtracerouteのようなICMPプロトコルを使うコマンドが使えないという問題がありましたが、現在では改良されて使えるようになっています。また、使用頻度の高いプロトコルでIPマスカレードを使っ

て中継できないものは、専用のカーネルモジュールがどんどん作られている状況にあります。

IPマスカレードを使う

さて、IPマスカレードの使い方ですが、これから説明する例では、PCルーターのeth0（192.168.0.14）にIPマスカレードをかけます。一般には、境界ネットと内部ネットともにプライベートアドレスを使う場合にIPマスカレードは使いません。しかし、Linuxによる IPマスカレードを紹介するために、ここでは便宜上PCルーター上でIPマスカレード機能を実行することにします。次のコマンドを実行すると、IPマスカレードが実行されます。

```
# /sbin/ipfwadm -F -p masq
```

このようにすると、内部ネット側（192.168.1.0）から外部へ接続するときは、すべてPCルーターのeth0（192.168.0.14）のように見えます。（インターネット側も含んで）境界ネット側から内部ネット側にあるネットワーク機材は何も見えないので、アクセスできないだけではなく、内部構造もわかりませんし、あるいはICMPに関連したサービス不能攻撃も仕掛けることもできません。

この設定だけではすべてのIPパケットが通ってしまいますが、IPパケットフィルタリングと同じように特定のアドレス、特定のプロトコル、特定のポートに対してのみIPマスカレードを設定し（リスト❸）、あとは通過させないこともできます。

PCルーターのこれから

現在、この連載はRedHat Linux 5.2（Linux-2.0.36）を前提としています。今後一般に広まるLinux-2.2.xでは、さらにIPフィルタリング関連が強化されます。

Linux-2.2.xではipfwadmに替わり「ipchains」を使ってLinux-2.2.xで強化されたIPファイアーウォール機能を使います。

SOHO環境のような小規模なネットワークはipfwadmで十分に保護することができるので、あわててipchainsに乗り換えることはないと考えています。また、ipfwadmで行っている機能はipchainsでも同じことができるので、将来乗り換えることがあってもそんなに大変ではないと思います。

しかし、この強化によりLinuxはさらにビジネスシーンで活躍するでしょうし、PCルーターでも高価な専用ルーターと同等な機能を用意できるようになります。

■お詫び

前回（INTERNET Magagine 1999/6）のP299にあるリスト⑤に誤りがありました。リスト④のように訂正してください。

図2　サイトのネットワーク構成

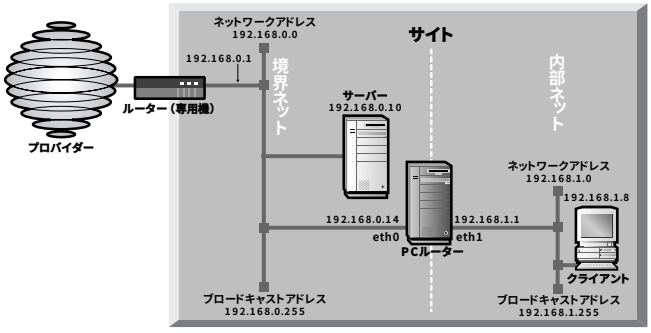
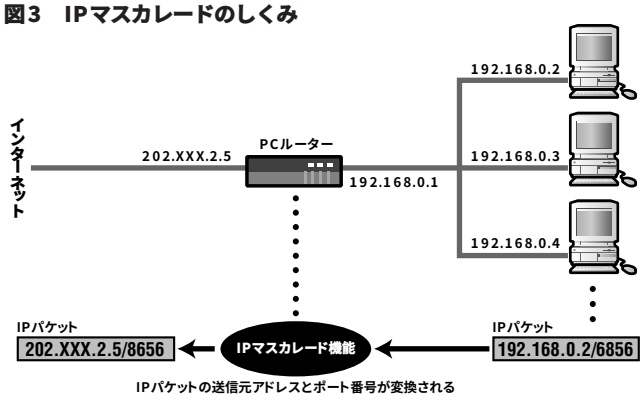


図3　IPマスカレードのしくみ



リスト❸　IPマスカレードを設定する

●以下のファイルIPmasqを作成して実行する

```
#!/bin/sh
#すべてを拒否
/sbin/ipfwadm -F -p deny
#IPマスカレードを行うと同時に内部ネットからのDNS（53）の通過を許可
/sbin/ipfwadm -F -a m -b -P udp -S 192.168.1.0/24 -D 0.0.0.0/0  53
#IPマスカレードを行うと同時に内部ネットからのHTTP（80）を許可
/sbin/ipfwadm -F -a m -b -P tcp  -S 192.168.1.0/24 -D 0.0.0.0/0
80
```

●IPマスカレードが有効かも確認する　←このコマンドを実行

```
% /sbin/ipfwadm -F -l
IP firewall forward rules, default policy: deny
type  prot source               destination          ports
acc/m udp  192.168.0.0/24       anywhere             any -> domain
acc/m tcp  192.168.0.0/24       anywhere             any -> http
```

←表示結果

IPマスカレードが有効になっている

リスト❹　先月号リスト❺の訂正

●間違い

```
if [ ! -f $C ]; then
        ## ipfwadmがみつかりません。
        echo "IP filter: ipfwadm not found" >&2
        exit 0
fi
```

●正解

$Cの部分は正しくは/sbin/ipfwadmです。

```
if [ ! -f /sbin/ipfwadm ]; then
        ## ipfwadmがみつかりません。
        echo "IP filter: ipfwadm not found" >&2
        exit 0
fi
```

［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