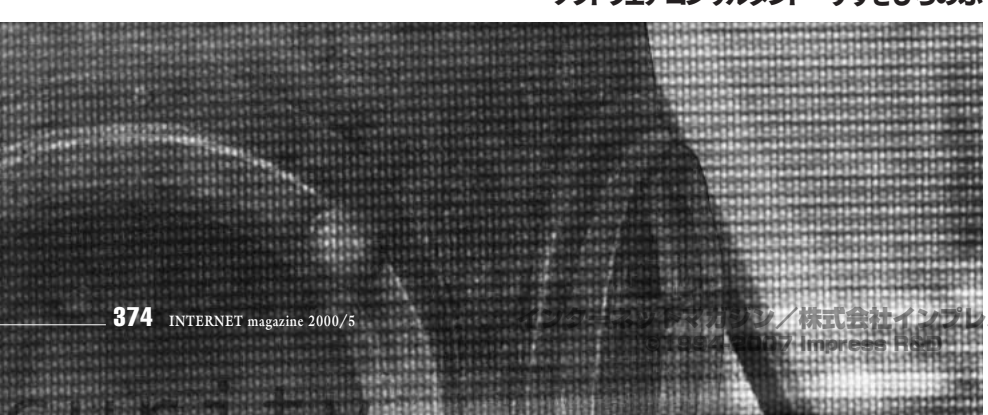
実践 Linux セキュリティー講座

今回はメールシステムに関係したセキュリティーについて取り上げます。メールサーバーを運用する場合、もっとも注意しなくてはならないのは、外部のネットワークからサーバーを不正に利用されないようにすることです。以前はこうした設定は面倒でしたが、最新版のサーバープログラムであれば、すでに対策が施されているので、設定はそれほど難しくありません。

第16回 メール関連のセキュリティーを設定する

ソフトウェアコンサルタント　すずきひろのぶ



sendmailは最新版を使う

メールサーバーを運用する際に気をつけなくてはならないのは、外部のネットワークからメールサーバーを利用されないようにする点です。外部から誰でも利用できる設定になっていた場合には、ダイレクトメールを大量に発送するspam業者（376ページのコラム参照）などに中継サーバーとして悪用されてしまう危険性があります。

この連載ではRedHat Linux 5.2の利用を想定していますが、現在ではRedHatを含む多くのLinuxパッケージでメールの不正中継対策がデフォルトで施されています。メールの配送に用いるsendmailというプログラムでは、配送ルールをsendmail.cfというファイルに記述しますが、RedHat Linux 5.2ではデフォルトでは中継ができない設定になっています。

古い記事などでは、spam対策としてsendmail.cfの設定に関する記述がありますが、こうした内容は古いsendmailシステムをそのまま使い続けているような所や、独自のルールによるsendmail.cfを作成しているような所のためのものです。最新のsendmailのパッケージでもデフォルトで不正中継を禁止する設定になっています。

この連載の守備範囲ではありませんが、以前から使っているUNIXシステムは注意が必要です。sendmailバージョン8.8.3や8.8.4といった古いものは非常に危険な脆弱性を持っています。このバージョンで運用しているマシンが外部からバッファーオーバーフロー攻撃を受けると重大な問題を引き起こします。最新のsendmailにアップデートすると同時にsendmail.cfも更新してください。

配送コントロールの設定

RedHat Linuxでは、メール配送コントロールは/etc/mailの下にある4つのファイルによって行います。以下にそれぞれについて説明していきます。

・/etc/mail/deny

メールの受け取りを拒否する記述を行うファイルです。フォーマットは次のようになります。

拒否する相手 <タブ> エラーメッセージ

拒否する相手の記述は、直接メールアドレスを「user@domainname.co.jp」という具合に書くこともできますし、拒否したい相手側のマシンのIPアドレスを書くこともできます。また、domainname.co.jpというようにドメインを指定すると、そのドメイン全体からのメールを拒否できます（❶）。

アドレスの後をタブで区切ってメッセージを書くと、それが拒否した相手に送られる時に使われるメッセージに使われます。メッセージの指定がなければ拒否した相手には「550 Access denied」というメッセージが送られます。denyファイルを更新した後に、makemapコマンドを使ってDBファイルを作成して終了です（❷）。

・/etc/mail/ip_allow

・/etc/mail/name_allow

ip_allowとname_allowは、それぞれメールサーバーへのアクセスを許可するマシンのIPアドレスとドメイン名を記述します。❸❹の例は、pc（192.168.1.10）というマシンからのメールを受け付けるという意味です。このメール中継が許されないマシンからのメールの中継依頼には「we do not relay」というエラーが戻ります。

・/etc/mail/relay_allow

外部からのメールをさらにリレーする場合の設定ファイルです。ただし、メールサーバーが1台しかなく、このマシンでPOPサーバーあるいはIMAPサーバーを提供してるならば、さらにリレーする必要はありません（❺）。リレーを行う場合には、相手先マシンのIPアドレスまたはドメイン名を記述します（❻）。

ここでは、多くのインターネットユーザーが使いなれた、メールサーバーが中心にあって、

❶ /etc/mail/deny

```
user@domainname.co.jp  ←  user@domainname.co.jpからのメール拒否
210.145.219.248        ←  210.145.219.248からのメールを拒否
domainname.co.jp       ←  domainname.co.jpからのメールを拒否
210.145.219            ←  210.145.219.xxxからのメールを拒否
```

❷ makemapの実行

```
% makemap -v /etc/mail/deny < /etc/mail/deny
```

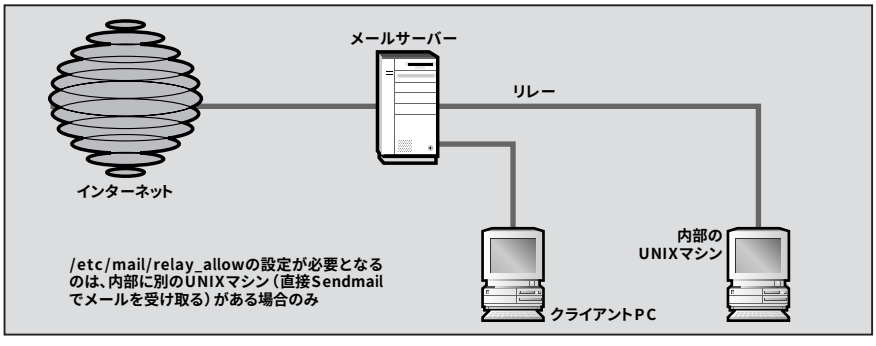
❸ /etc/mail/ip_allow

```
192.168.1.10  ←  メール中継を受け付けるIPアドレス
```

❹ /etc/mail/name_allow

```
pc  ←  メール中継を受け付けるマシン名
```

❺ リレーが必要な場合



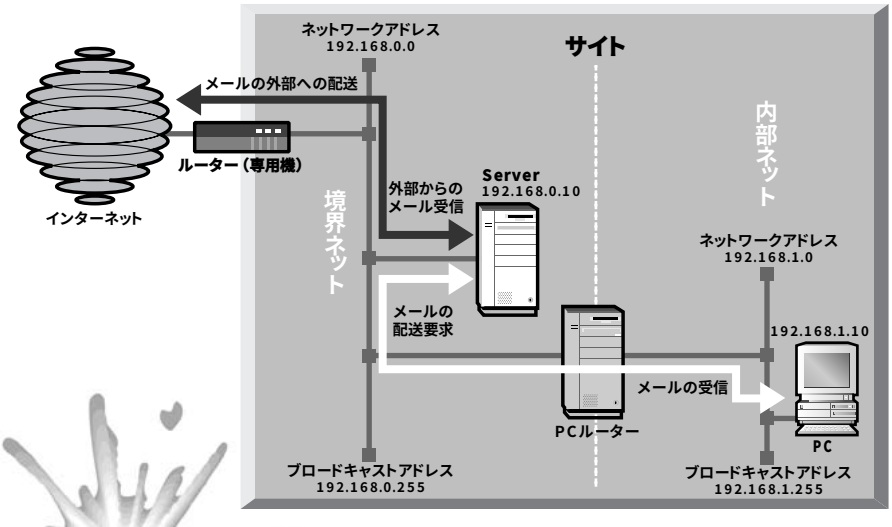
❻ /etc/mail/relay_allow

```
server2  ←  リレーを許可するマシン名
```

❼ ip_allowの通常の設定例

```
192.168.1  ←  内部ネットワーク192.168.1.xxxからのメール中継を受け付ける
```

❽ ネットワーク構成図



IMAP（アイマップ／Internet Message Access Protocol）：インターネットで、メールを保存しているサーバーからメールをメールソフトに送受信するためのプロトコル。POPとは違い、メールはサーバー側に置いたまま管理するため、会社や自宅、モバイルなど、複数の環境からメールを読み書きできる。

そこからPOPやIMAPでメールを受信する方法をとることにします。こうした場合には、❼のように/etc/mail/ip_allowに内部ネットワークのアドレスを記述するだけで十分です。構築するネットワークの全体像は❽のようになります。

メールサーバーは独立すべきか

メールサーバーは他のシステムとは切り離し独立に持つべきか、それとも他のサーバーと同居させるのかという判断はサイトごとに状況が違うので、一概にはどちらがいいとは言えません。1つのマシンにウェブサーバーもメールサーバーも同居させることの利点と欠点を天秤にかけてみる必要があります。

【利点】1台のLinuxマシンを用意するだけで良い

→金銭的負担が少ない

→1台で管理できるので管理コストが少ない

【欠点】各サービスに沿ったセキュリティーポリシーが立てにくい

→シンプルな設定が難しい

→きめの細かい設定が難しい

利点の部分で「管理コストが少ない」と書きましたが、これは常に正しいとは限りません。たとえばサービスによって分割したほうが、システムのセキュリティーポリシーはシンプルになります。シンプルになればなるほど管理も単純化されて簡単になり、また同時に抜け道も少なくなります。もしかするとシンプルに分割したほうが作業の手間という面で管理コストが安いという可能性もあります。

筆者の場合は、同居させるという方針をとります。理由は、管理を1台に集中できることと、メールアカウントを発行するユーザー数が少ないこと、利用環境が狭い範囲なのであまり複雑なアクセス制御は必要ないからです。

spamの語源

不特定多数のユーザーに対してメールを送り付けるような行為は、公式の場ではUCE（Unsolicited Commercial E-mail:希望しない広告メール）またはUBE（Unsolicited Bulk E-mail:希望しないバルクメール）と呼ばれていますが、一般にはspamと呼びます。もともとSPAMは米Hormel Foods社の肉の缶詰の商標です。spamという言葉はイギリスのテレビ番組モンティパイソンで流れた有名なコントから取られています。大衆レストランにカップル客が入って料理の種類を尋ねるとすべてがSPAM料理で、そのまわりの客が「スパム、スパム、スパム、スパム」と繰り返す歌を歌うという有名なコントです。モンティパイソンのビデオに入っているので、一度見ておくもの一興でしょう。

さて世界で最初の大規模なspamは、アメリカにある移民専門弁護士事務所からUsenetに流されたアメリカ移民局労働ビザ抽選手続き代行の広告でした。この頃はもちろんspamなどという名称などありません。Usenet上で「しつこく繰り返し迫るのは、これはモンティパイソンのSPAMのコントと同じだ」といったところからspamという名前が付きました。もともとインターネットの世界は大学と研究所を中心としたフランクなユーザーが中心ですから、この手の俗語がテクニカルターム化しているものがたくさんあります。

このストーリーには続きがあります。この移民専門弁護士事務所はこのspamのおかげで本当にお金を儲けたようで、後に「インターネットですぐに10万ドルを儲ける方法」という本を出してさらに印税を稼ごうとします。この本はインターネット関連の雑誌、あるいはインターネット販売を行っている書店からは軒並み広告拒否、取扱拒否を受けます。しかし、このことが「本当にspamでお金を儲けることができる証拠だ」とspam関連業者の宣伝材料にされてしまったのでした。

POP3サーバーの設定

内部のマシンからメールを利用するために、サーバーマシンにPOP3サーバーをインストールします（❾）。インストールはパッケージで簡単に行えますが、古いバージョンのimapdのRPMパッケージには脆弱性を持ったimapdが含まれていますJump。RedHat Linux 5.2以降であれば問題ありませんが、古い環境をまだアップデートせずに使っていると危険ですので、必ず最新のパッケージを利用してください。

Step1 パッケージのインストール

Linux RedHatではimap-xx-xx.rpmのパッケージの中にipop2d、ipod3d、imapdの3つがあり、/usr/sbinにインストールされます。このうち、古い規格のPOP2（ipop2d）が必要になることはほとんどないでしょう。また、ユーザーが少ないSOHO環境でPOP3とIMAPのどちらのプロトコルも必要という状況は少ないでしょうから、POP3（ipop3d）かIMAP（imapd）のどちらかを残し、不必要なものは削除したほうが確実でしょう。ここではPOP3を使うことにします。

Step2 ユーザーのアカウントを作る

ユーザーのアカウントはコマンドuseraddを使って登録します。この時、ホームディレクトリーとログインシェルには/dev/nullを設定し、決してログインできない環境にしておきます。アカウント作成後、そのアカウントにパスワードを設定します。

Step3 /etc/inetd.confでipop3dを設定

いままで/etc/inetd.confでコメントアウトしていたipop3dのエントリーを復活させます。

Step4 TCP_WRAPPERのアクセス許可

TCP_WRAPPERでのipop3dへのアクセス許可を記述します。必ず、内部ネットワークとローカルファイルのみからのアクセスを許すだけにしてください。

RPM（アール・ピー・エム／RedHat Package Manager）：各種ソフトウェアをLinuxにインストールする際に用いるプログラムの名称。RPMを使えば、実行ファイルやソースコードなどをまとめたパッケージと呼ばれる1つのファイルから簡単に各種ソフトウェアをインストールできる。RedHat Linux以外のLinuxでも広く利用されている。

Step5 inetdにシグナルを送る

inetdにHUPのシグナルを送って/etc/inetd.confの内容をリロードさせます。

あとは、プロバイダーのメールサーバーを利用するのと同じような方法で利用してください。いままでの設定で行った各設定情報は⑩のようになります。

Step1でも説明しましたが、不必要なプログラムは削除しましょう。インストールされているにも関わらず使っていないサービスのため、その存在を忘れているといったパターンが多く見受けられます。たとえば今回のパッケージでもipop2d、ipop3d、imapが同時に入っています。そのような状態で脆弱性対応として使っているipop3dだけアップデートし、脆弱性を持つ古いipop2dや古いimapがそのままだというサイトがあるようです。最初のインストール時点でとにかく何でもかんでも思いつきでインストールしてしまうのは止めましょう。きちんと利用計画を立ててインストールすれば、利用しないのでサーバーの存在を忘れてしまうルーズな管理にはならないはずです。

Jump www.cert.org/advisories/CA-98.09.imapd.html

次回は外部ルーターの設定

実際のメールサーバーの運用にあたっては、DNSサーバーに登録する必要があります。本連載ではDNSサーバーについてはプロバイダー側で管理してもらうことを前提としていますので、DNSサーバーへの登録方法は契約しているプロバイダーのマニュアルなどを参照してください。最終回となる次回は、これまで触れてこなかったインターネットに接続されるルーターによるIPフィルタリングについて説明したいと思います。

❾ POP3サーバーのインストール

Step1 パッケージをインストール

```
# rpm -v -h -i imap-4.4-2.i386.rpm
```

Step2 ユーザーのアカウントを作る

```
# useradd -c 'Account for PostPet' -d /dev/null -s /dev/null -g mail -u 1007 kuma
```

❶コメント　❷ホームディレクトリー　❸ログインシェル　❹グループ名　❺ユーザーID　❻アカウント名

```
# passwd kuma
New UNIX password:********   ← きちんとしたパスワードを入力すること
Retype new UNIX password:******** ← パスワードの再入力
passwd: all authentication tokens updated successfully
```

Step3 /etc/inetd.confでipop3dを設定

```
pop-3   stream  tcp     nowait  root    /usr/sbin/tcpd  ipop3d
```

Step4 TCP_WRAPPERのipop3dへのアクセス許可

```
/etc/hosts.allowに以下を追加
ipop3d:LOCAL,192.168.1.   ← ローカルマシンと192.168.1.(内部ネットワーク)のみ
ALL: ALL: DENY
```

Step5 inetdにシグナルを送る

```
# ps axuw | grep inetd   ← 稼動中のinetdのPIDを調べる
root       278  0.0  0.3   792   408  ?  S   Feb  6   0:00 inetd

# kill -HUP 278   ← inetdにHUPシグナルを送って/etc/inetd.confの内容をリロードさせる
```

❿ クライアント側の設定

受信メールサーバー	
メールサーバー	：server（あるいはserverのIPアドレス）
サーバーのプロトコル種類	：POP3サーバー
ユーザー名	：kuma
パスワード	：さきほど登録したもの

送信メールサーバー	
SMTPサーバー	：server（あるいはserverのIPアドレス）
ユーザー名	：kuma

HUP（エイチ・ユー・ピー）：UNIXのkillコマンド（実行中のプログラムを停止させるコマンド）でオプションとして指定する文字列の1つ。HUPはハングアップの略。

［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