開かれたネットワーク環境でのセキュリティ対策

八代茂夫

高エネルギー加速器研究機構 計算科学センター

概要

共用のサービスシステムを管理している立場から、および個人的に外部との通信を行なうためのシステムを稼動させるために、この間に検討あるいは実施してきた Linux システムでの、ホストベースのセキュリティ対策について述べる。

1　はじめに

大学や研究機関におけるインターネットの利用は研究者間の情報の交換手段として、開かれた環境として発達してきた。インターネットの商業利用が広まり、一般社会に普及するにつれて「不正アクセス」というものを考慮せざるを得なくなった。その対策の 1 つとして研究機関に於いてもファイアウオールの導入が進んでおり、KEK でも 2003 年 9 月から本格的な運用が開始された。

しかし研究活動におけるインターネットの利用においては、企業等のように情報の流れを管理するわけにはゆかず、むしろ研究者個人が研究活動に必要な情報を必要性に応じて扱えなければならない。従って、ファイアウオールを導入したと言っても、企業等とはたいぶ様相が異なり、研究者自身のシステムあるいは共用のサービスシステムには幅広い外部との通信が保証される必要がある。すると、そのシステムは自由度を持った代償として、不正アクセスへの対処が必要となる。また、外部との直接の通信を行なわないシステムであっても 2 次的な影響を受ける可能性、あるいは内部からの不正アクセスの可能性があるので、安全を試行する管理者なら不正アクセスへの対処をとるであろう。

2　セキュリティ対策の概要

対策すべき点は、(1) セキュリティ情報の入手、(2)不要なアクセスを排除する、(3)通信の暗号化、(4)root 権限を取られないようにする、(5)ログの管理である。更にシステムの回復といったことも管理者には重要なことであるが、本稿では触れない。

2.1　セキュリティ情報の入手とサーバソフトの更新

最新情報を入手して、特にセキュリティホールに関する情報を得た場合には、早急に対策済みのソフトウエアを入手して更新するのがシステム管理の基本である。

セキュリティ情報は JPCERT[1]、CERT[2]、その他のセキュリティ関連会社・組織などから入手できる。

2.2　不要なアクセスの排除

a) 不要なサービスの停止

サービスが起動しているということは、不正アクセスを受ける可能性が生じる。起動される全サービスを確認して、使用しないサービスは起動しないようにする。自分の知らないサービスは動作させないことが原則である。サービスの内容の確認は、多くは man コマンドでできる。多くのサービスの起動は通常 /etc/rc.d/, /etc/xinet.d/ あるいは /etc/rc.local に記述されている。

/etc/rc.d/ で起動されるのサービスの一覧表示および無効化は以下のコマンドでできる。この例では sendmail を無効化している。

```
# /sbin/chkconfig –list
# /sbin/chkconfig sendmail off
# /sbin/service sendmail stop
```

b) IP フィタリングソフトの利用

Linux には iptables[3]や ipchains[4]といった、受信あるいは送信する IP パケットを制御するソフトウエアが組み込まれている。これを設定し動作させて、システムに必要なパケットだけを通過させ、使わないパケットは拒否することにより、他のホストからの無用なアクセスを押さえてセキュリティを高めることができる。詳細は後述する。

2.3 通信の暗号化

ネットワーク上にパスワードを平文で流さないために SSH[5] の利用はもはや常識である。X は SSH の X フォワーディング経由で使用し、ftp は scp や sftp の利用に切り替えるか、あるいは SSH のポートフォワーディング経由で使用し、rsync は SSH で使用し、imap や pop も，rsync なども SSH のポートフォワーディング経由で使用するのが良い[6]。共同利用者のために X 端末を運用しているが、この問題は後述する。

2.4 root の制限

root でのリモートログインは禁止すべきであり、SSH では禁止する設定になっている。ユーザアカウントでログインして su で root 権限を得る。多くの利用者がログインするホストの場合には、特定ユーザだけが su できるようにしてセキュリティを強化する。利用者がログインしないサーバ専用のホストでは、SSH ログインに公開鍵認証だけを受け付けるようにすることもセキュリティ対策の 1 つである。

2.5 ログの管理

不正アクセスを受けた場合に、どのような経路で侵入されたか、何をされたかなどを追跡調査するには、ログの情報は重要である。不正アクセスをする者は痕跡をけすために、ログを消去することが多い。リモートのホストにログ転送することにより、ログを消去できないようにするのは有効な対策である。

必要に応じて/etc/logroted.d/内のファイルを修正して、logroted により古いログが消去されないようすることも忘れてはならない。

ログを解析することにより、セキュリテイ上の弱点を知ることや、他ホストからの不正アクセスの兆候を見て取ることが可能である。ログ解析には支援ツールである logwatch などの利用も検討に値する。

3 IPTABLES の設定

Iptables 設定して動作させることにより、必要なパケットだけを通過させ、使わないパケットは拒否して、他のホストからの無用なアクセスを押さえることができる。また、NFS など安全性の低いソフトを利用する場合には、通信の相手先を限定してセキュリティを高めることができる。

Kernel バージョンが古い場合には iptables がなく ipchains が組み込まれており、こちらを利用できる。

表 1 に iptables の設定例を示す。これは Red Hat Linux 9 で作成された /etc/sysconfig/iptables を基に設定した例である。(1) は標準値を DROP に変更したものであり、以降の設定に該当しない場合の標準値となる。設定ミスがあっても安全な方向で動作させるものである。(2) は ping などのための icmp を受け付けている。(3)-(6)は name server （ここでは 130.87.56.2 としている）からのパケットを受け付けている。(7) は ntp サーバ 172.30.32.102 との間の udp123 の通信を受け付けている。RedHat 9 のばあいには ntpd 起動時に自動的

に設定されるのでここに記述する必要はない。(8)は任意のホストからの ssh の通信を受け付けている。(9)は KEK 内からの ssh を受け付ける場合の例であるが、アクセス制御はここで行なうよりも TCPWRAPPERS で行なった方がやり易い。(10)(11)はメイルを受け付けるための記述である。一部の sendmail のために(12)の tcp 113 の必要な場合がある。(13)は 130.87.32.65 のプリンタを使う例である。(14)は Web サーバとして機能させるための記述である。(15)(16)は SMB サーバとして機能させるための記述であり、この例では 130.87.57.44 に対してアクセスを許している。(17)-(19)は NFS サーバとして機能させるための記述であり、この例では 130.87.57.44 に対してアクセスを許している。

表 1.　iptables の設定例

```
*filter
:INPUT DROP [0:0]                                                                    (1)
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:RH-Lokkit-0-50-INPUT - [0:0]
-A INPUT -j RH-Lokkit-0-50-INPUT
-A FORWARD -j RH-Lokkit-0-50-INPUT
-A RH-Lokkit-0-50-INPUT -i lo -j ACCEPT
-A RH-Lokkit-0-50-INPUT -p icmp -j ACCEPT                                            (2)
-A RH-Lokkit-0-50-INPUT -p tcp --sport 53 -s 130.87.56.2 -j ACCEPT                   (3)
-A RH-Lokkit-0-50-INPUT -p tcp --sport 42 -s 130.87.56.2 -j ACCEPT                   (4)
-A RH-Lokkit-0-50-INPUT -p udp --sport 53 -s 130.87.56.2 -j ACCEPT                   (5)
-A RH-Lokkit-0-50-INPUT -p udp --sport 42 -s 130.87.56.2 -j ACCEPT                   (6)
-A RH-Lokkit-0-50-INPUT -s 172.30.32.102 -p udp -m udp --sport 123 --dport 123 -j ACCEPT  (7)
-A RH-Lokkit-0-50-INPUT -p tcp -m tcp -- dport 22 -j ACCEPT                          (8)
-A RH-Lokkit-0-50-INPUT -s 130.87.0.0/255.255.0.0 -p tcp -m tcp --dport 22 -j ACCEPT  (9)
-A RH-Lokkit-0-50-INPUT -p tcp --dport 25 -j ACCEPT                                  (10)
-A RH-Lokkit-0-50-INPUT -p tcp --sport 25 -j ACCEPT                                  (11)
-A RH-Lokkit-0-50-INPUT -p tcp --dport 113 -j ACCEPT                                 (12)
-A RH-Lokkit-0-50-INPUT -p tcp -s 130.87.32.65 --sport 515 -j ACCEPT                 (13)
-A RH-Lokkit-0-50-INPUT -p tcp --dport 80 -j ACCEPT                                  (14)
-A RH-Lokkit-0-50-INPUT -p tcp --dport 137:139 -s 130.87.57.44 -j ACCEPT             (15)
-A RH-Lokkit-0-50-INPUT -p udp --dport 137:139 -s 130.87.57.44 -j ACCEPT             (16)
-A RH-Lokkit-0-50-INPUT -p udp --dport 111 -s 130.87.57.44 -j ACCEPT                 (17)
-A RH-Lokkit-0-50-INPUT -p tcp --dport 111 -s 130.87.57.44 -j ACCEPT                 (18)
-A RH-Lokkit-0-50-INPUT -p udp --dport 2049 -s 130.87.57.44 -j ACCEPT                (19)
```

4　TCPWRAPPERS の設定

TCPWRAPPERS の設定は Linux では/etc/hosts.allow と/etc/hosts.deny によって行なう。 この設定もアクセス拒否を標準値にして、アクセスを許すサービスあるいは相手先を１つづつ記述してゆくのが良い。

まず、hosts.deny には "ALL: ALL" と記述する。次に hosts.allow の設定を行なう。表２に設定例を示す。全ての sshd 接続、130.87.57.44 からの NFS 接続、KEK 内からの sendmail 接続を受け付けている。

表 2.　/etc/hosts.allow の設定例

```
sshd: ALL
portmap: 130.87.57.44
sendmail: 130.87. localhost
```

5　SSH の公開鍵認証

SSH によるユーザ認証にはパスワード認証と公開鍵認証がある。パスワード認証はパスワードを SSH サーバ側で管理する。一方、公開鍵認証では秘密鍵とそのパスフレーズを SSH クライアント側で管理する[6]。

全てのホストを自分ひとりが管理し利用するなら、使いやすい認証方式を利用すればよい。サービスマシンで利用者と管理者という関係になった場合には、利用者にとっては自分で秘密鍵を管理できる公開鍵認証認証の方が好ましいが、管理者にとっては管理できない公開鍵での認証を受け入れることになる。この場合の最大の問題は、パスフレーズ無しの秘密鍵を検査できないことである。

公開鍵認証で SSH agent を使用するとリモートアクセスが非常に便利になる。シングルサインオンや第三者ファイル転送も可能になる。後者の例を以下に示す。

```
$ scp soleil:filename etoile:
```

6　X 端末

共同利用者のために X 端末を運用しているが、SSH がサポートされていない問題がある。この問題を解決するために CD Linux の１つである KNOPPIX[7]を X 端末として利用することを検討している。

X 端末の利点は、シャットダウンをせずに電源断できること、再起動で初期状態に戻せること、ユーザアカウントを作成しなくて良いことである。PC や Linux を端末にした場合にはこの条件を満たせない。しかし、CD Linux では満たせるので、ディスクレス PC を用意して、X 端末として運用することを計画している。

7　さいごに

セキュリティの議論の相手である佐々木節氏、柴田章博氏、橋本清治氏、湯浅富久子氏および株式会社でんさテクノ東京の高波喜八郎氏に感謝します。

筆者は調査検討したことを Web で公開している。先日スイスから 1 通のメイルが届いた。最初は筆者の名をかたったスパムメイルへの抗議かと思ったが、実は Web の sendmail と hosts.allow との関係を記述した情報によって彼が抱えていた問題を解決できたとの感謝のメイルであった。調査研究の成果を Web で公開することが、我々の課題の 1 つとなっている「社会への貢献」につながることを改めて認識した。

参考文献

[1]　JPCERT/CC, http://www.jpcert.or.jp/

[2]　CERT/CC, http://www.cert.org/

[3]　netfilter/iptables FAQ, http://www.linux.or.jp/JF/JFdocs/netfilter-faq.html

[4]　Linux IPCHAINS-HOWTO, http://www.linux.or.jp/JF/JFdocs/IPCHAINS-HOWTO.html

[5]　SSH documents, http://research.kek.jp/people/yashiro/html/SSH.html

[6]　八代, 橋本, 安全なリモートログインのツール SSH, KEK Internal 2000-1

[7]　KNOPPIX, http://unit.aist.go.jp/it/knoppix/