Strategies For Securing a Linux System

Donald Buchan
malak@malak.ca
May/August 2009
Translated & Updated January 2011
Overview

- Use a router
- All users should have their own account
- Manage Passwords
- Control Access to Your Desktop
- Perform Updates Regularly
- Install Software From Repositories
- Activate Your Firewall and Tune Appropriately
- Turn Off Unneeded Services
- Limit SSH Access to only the users you allow, and exclude the root user
- Install DenyHosts
- Other Strategies
  - Install SELinux and Operate it in enforcing mode
  - Use NoScript
  - Encrypt your filesystem
  - Remove accumulated junk
  - Don't automount devices
  - Close root sessions at the command line
- Protect your system's physical integrity
Why should you protect your system?
Why should you protect your Linux system?

- Crackers want to compromise your system to do nefarious things, or just for the challenge;
- You don't want your system to be used for illicit or illegal activities (spam, porn, or financial or identity theft);
- A compromised Linux system is worth more than a compromised Windows system.
Why should you protect your Linux system?

• You have personal files to protect – it's no one's business but yours that you like wine, nature walks, or Barbie dolls – unless, of course, you wish to share that information!

• *Simply put, it's your computer, to be used for your purposes as you see fit.*
Linux generals command Windows grunts in botnet battlefield

Veteran virus still recruiting for zombie army
Darren Paul 15/02/2008 15:40:20

Linux servers infected with a mutating virus are commanding huge Windows botnets six years after the malware was discovered, according to security researchers.

The Linux RSTB virus infects the working directory of Microsoft executable and linkable format (ELF) executable files. It can also create a backdoor by opening a socket and listening for a packet containing the attackers origin and the command to be executed.

SophosLabs United Kingdom research director Billy McCourt said Linux boxes are valuable targets as botnet controllers because they are typically remain online as servers.

"Linux computers are very valuable to hackers. A bot army, similar to real armies, needs a general and infantry [and] Linux boxes are often used as servers, which means they have a high up-time - essential for a central control point," McCourt said.

"A Windows computer, on the other hand, is found at home or as a desktop machine in an office, and these computers are regularly switched off [which makes them less attractive as controllers, but ideal for] infantry, or zombies.

"We run various honeypots to try as you might also expect, these honeypots are attacked more frequently than our Linux ones, but Linux malware is far more interesting."

"Linux systems, once compromised, are ideal platforms to unleash all sorts of nastiness."

McCourt said the virus, discovered in February 2002, is unique among Linux malware because it can replicate across current distributions.

University of New South Wales senior network administrator for the school of computer science and engineering Peter Linich told Compuerton Linux servers are extremely valuable to hackers since they are typically online more than 10 months a year.

"Just yesterday I was watching our incoming network traffic and noticed an ADSL host in Greece scanning through all our machines and running an SSH (Secure Shell) password-guessing attack on all the SSH servers it found." Linich said, adding such attacks occur multiple times a day.

Such activity is a real threat in our environment where we have hundreds of Linux systems running 24x7. Linux systems, once compromised, are ideal platforms to unleash all sorts of nastiness.
Why should you protect your linux system?

The approach to take must be multi-faceted with different approaches which may or may not overlap. Some ways are automatic settings in your computer, some are habits you should adopt.

Remember: You have to win every day, but a cracker only has to win ONCE!
A System Under Attack
apr 26 13:54:26 malak sshd[23323]: pam_unix(sshd:auth): check pass; user unknown
apr 26 13:54:26 malak sshd[23323]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ss ruser= rhost=202.105.49.16
apr 26 13:54:26 malak sshd[23323]: pam_succeed_if(sshd:auth): error retrieving information about user tomcat
apr 26 13:54:27 malak sshd[23320]: Failed password for invalid user root from 202.105.49.16 port 43998 ssh2
apr 26 13:54:28 malak sshd[23323]: Failed password for invalid user tomcat from 202.105.49.16 port 42446 ssh2
apr 26 13:54:28 malak sshd[23322]: Received disconnect from 202.105.49.16: 11: Bye Bye
apr 26 13:54:28 malak sshd[23321]: Invalid user cady from 202.105.49.16
apr 26 13:54:28 malak sshd[23325]: input_userauth_request: invalid user cady
apr 26 13:54:28 malak sshd[23321]: pam_unix(sshd:auth): check pass; user unknown
apr 26 13:54:28 malak sshd[23321]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ss ruser= rhost=202.105.49.16
apr 26 13:54:28 malak sshd[23321]: pam_succeed_if(sshd:auth): error retrieving information about user cady
apr 26 13:54:28 malak sshd[23324]: Received disconnect from 202.105.49.16: 11: Bye Bye
apr 26 13:54:30 malak sshd[23326]: Invalid user marine from 202.105.49.16
apr 26 13:54:30 malak sshd[23327]: input_userauth_request: invalid user marine
apr 26 13:54:30 malak sshd[23326]: pam_unix(sshd:auth): check pass; user unknown
apr 26 13:54:30 malak sshd[23326]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ss ruser= rhost=202.105.49.16
apr 26 13:54:30 malak sshd[23326]: pam_succeed_if(sshd:auth): error retrieving information about user marine
apr 26 13:54:30 malak sshd[23325]: Connection closed by 202.105.49.16
apr 26 13:54:30 malak sshd[23321]: Failed password for invalid user cady from 202.105.49.16 port 42344 ssh2
apr 26 13:54:31 malak sshd[23328]: User root from 202.105.49.16 not allowed because not listed in AllowUsers
apr 26 13:54:31 malak sshd[23329]: input_userauth_request: invalid user root
apr 26 13:54:31 malak sshd[23328]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ss ruser= rhost=202.105.49.16 user=$
apr 26 13:54:31 malak sshd[23326]: Failed password for invalid user marine from 202.105.49.16 port 45914 ssh2
apr 26 13:54:32 malak sshd[23327]: Received disconnect from 202.105.49.16: 11: Bye Bye
apr 26 13:54:33 malak sshd[23328]: Failed password for invalid user root from 202.105.49.16 port 49276 ssh2
apr 26 13:54:33 malak sshd[23329]: Received disconnect from 202.105.49.16: 11: Bye Bye
apr 26 13:54:36 malak sshd[23330]: Invalid user global from 202.105.49.16
apr 26 13:54:36 malak sshd[23331]: input_userauth_request: invalid user global
apr 26 13:54:36 malak sshd[23330]: pam_unix(sshd:auth): check pass; user unknown
apr 26 13:54:36 malak sshd[23330]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ss ruser= rhost=202.105.49.16
apr 26 13:54:36 malak sshd[23330]: pam_succeed_if(sshd:auth): error retrieving information about user global
apr 26 13:54:37 malak sshd[23332]: User root from 202.105.49.16 not allowed because not listed in AllowUsers
apr 26 13:54:37 malak sshd[23331]: input_userauth_request: invalid user root
apr 26 13:54:37 malak sshd[23332]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ss ruser= rhost=202.105.49.16 user=$
apr 26 13:54:38 malak sshd[23330]: Failed password for invalid user global from 202.105.49.16 port 49425 ssh2
apr 26 13:54:38 malak sshd[23334]: Invalid user marine from 202.105.49.16
apr 26 13:54:38 malak sshd[23335]: input_userauth_request: invalid user marine
apr 26 13:54:38 malak sshd[23331]: Received disconnect from 202.105.49.16: 11: Bye Bye
apr 26 13:54:38 malak sshd[23334]: pam_unix(sshd:auth): check pass; user unknown
apr 26 13:54:38 malak sshd[23334]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ss ruser= rhost=202.105.49.16
apr 26 13:54:38 malak sshd[23334]: pam_succeed_if(sshd:auth): error retrieving information about user marine
apr 26 13:54:39 malak sshd[23332]: Failed password for invalid user root from 202.105.49.16 port 53047 ssh2
apr 26 13:54:39 malak sshd[23333]: Received disconnect from 202.105.49.16: 11: Bye Bye
Use A Router
Use A Router

- Connect your computer(s) to a router, and the router to your modem.
- NAT service will block unexpected packets (scripts, scanning, crackers, login requests to various servers, etc.) from outside sources – but not from within the network it creates!
All Users Should Have Their Own Account
All Users Should Have Their Own Account

- System | Administration | Users and Groups

- All users have their own unique account(s) with a password, however they log in: ssh, ftp, vnc, and particularly the desktop – each user's data will be safe from other users and intruders
Manage Passwords
Manage Passwords

- In order to reduce the success of dictionary attacks, enforce passwords that:
  - Are at least 8 characters long
    - 1gp-w+ii instead of pwd
  - Have CAPITAL letters, small letters, digits, and special characters, such as:
    - °!”#$%&*()_+=<>/^,
    - 1234567890
    - 1Gp-W+ii instead of password
Managing Passwords

- That are easy to remember but hard to guess, such as the initials to a memorable phrase
  
  - 1Gp-W+ii --> “(1 Good Password Is Important)”

- Give the root an unique password – useful if your account is compromised and the attacker knows your password (see SSH access)
Password Expiration

- System Administration Users and Groups, choose user, Password Info tab
- Enforce password expiration, such as every 90 days
Password Expiration

- At the command line and as root, in `/etc/login.defs`, modify the following line: “PASS_MAX_DAYS” to a number such as 90 days (3 months)
Control Access To Your Desktop
Control Remote Access To Your Desktop

- Remote desktop control: System | Preferences | Remote Desktop
- A system with no logged in desktop users will not allow VNC access
- Note the various options, such as connection confirmation
- In enabling VNC access, you may be broadcasting its availability on Wi-Fi
Control Remote Access To Your Desktop

• The screensaver should require a password in order to unlock the screen
  – System | Preferences | Screensaver

• This provides a certain amount of protection if you leave your computer for a few minutes (but wait for the end of this presentation!)
Automatic Screen Lock

- To automatically lock the screen using the screensaver:
  - System | Screensaver
- Click on “Activate Screen Saver when ...” and “Lock Screen when ...”, and set the delay
Password: ********
Remove Automatic Login

- *Removing automatic logins provides a small amount of protection if you leave your computer for a few minutes (and someone reboots it) or someone steals it (but wait for the end of this presentation!)*
Remove Automatic Login

- At the command line and as root, modify `/etc/gdm/custom.conf` and remove all lines that say “AutomaticLogin” and “AutomaticLoginEnable”
Perform Updates Regularly
Perform Updates Regularly

• Updates normally include:
  – Security patches
  – Software updates with new functions, abilities, improvements, etc.
  – The removal or replacement of packages considered obsolete or inferior to new packages
  – The installation of new software likely to be useful
  – Dependencies
Perform Updates Regularly

- Updates are prepared by either volunteers, paid staff, or both, who work to maintain the distribution up to date, and secure, and make sure that the updates are functional, won't break your system, are complete, and appropriate.

- Updates should normally be done automatically, unless you like to keep close control.
Perform Updates Regularly

- Unless you like to manually control updates, normally your computer should perform them automatically
- System | Preferences | Software Updates
Perform Updates Regularly

There are 27 updates available
Software updates correct errors, eliminate security vulnerabilities and provide new features.

- Small applications for the GNOME panel
  gnome-applets-1:2.32.0-3.fc14 (i686)
  Status: Installed
  Size: 5.7 MB

- A text file browser similar to more, but better
  less-436-9.fc14 (i686)
  Status: Installed
  Size: 107.3 KB

- Reference implementation of the iCalendar data typ...
  libical-0.46-2.fc14 (i686)
  Status: Installed
  Size: 167.7 KB

- libpurple library for IM clients like Pidgin and Finch
  libpurple-2.7.9-1.fc14 (i686)
  Status: Installed
  Size: 6.1 MB

- Windows MetaFile Library
  Status: Installed
  Size: 121.6 KB

This update will fix bugs and other non-critical problems.
The notification was issued on 01/03/2011.
This update fixes a bug in the stock price tracker applet (invest-applet) which prevented it from correctly displaying a detailed overview when left-clicked, claiming it could not contact the server.

For more information about this update please visit this website:
- [https://admin.fedoraproject.org/updates/F14/FEDORA-2011-0029](https://admin.fedoraproject.org/updates/F14/FEDORA-2011-0029)

For more information about bugs fixed by this update please visit this website:
- [https://bugzilla.redhat.com/show_bug.cgi?id=631912](https://bugzilla.redhat.com/show_bug.cgi?id=631912)
Perform Updates Regularly

- **Automatic updates:**
  - as root, set up a cron job (see screenshot)

- **Manual updates, at the command line:**
  - as root, “yum update”
  - *Remember to do so regularly!*

---

**Sample shell script to update system**

A shell script that instructs yum to update any packages it finds via `cron`:

```
#!/bin/bash
YUM=/usr/bin/yum
YUM -y -R 120 -d 0 -w 0 update yum
YUM -y -R 10 -w 0 -d 0 update
```

(Code listing: `/etc/cron.daily/yumupdate.sh`)

Where,

1. First command will update `yum` itself and next will apply system updates.
2. `-R 120` : Sets the maximum amount of time `yum` will wait before performing a command
3. `-e 0` : Sets the error level to 0 (range 0 - 10). 0 means print only critical errors about which you must be told.
4. `-d 0` : Sets the debugging level to 0 - turns up or down the amount of things that are printed. (range: 0 - 10).
5. `-y` : Assume yes, assume that the answer to any question which would be asked is yes.

Make sure you setup executable permission:

```
chmod +x /etc/cron.daily/yumupdate.sh
```
$ su
Password:
[root@malak malak]# yum update
Loaded plugins: refresh-packagekit
Trying other mirror.

livna
fedora
rpmfusion-free-updates
rpmfusion-nonfree-updates
rpmfusion-free
adobe-linux-1386
updates-newkey
updates-newkey/primary_db
rpmfusion-nonfree
updates

Setting up Update Process
Resolving Dependencies
--> Running transaction check
----> Package firefox.1386 0:3.0.18-1.fc9 set to be updated
----> Package gnome-python2-extras.1386 0:2.19.1-27.fc9 set to be updated
----> Package gnome-python2-gtkhtml2.1386 0:2.19.1-27.fc9 set to be updated
----> Package gnome-python2-libegg.1386 0:2.19.1-27.fc9 set to be updated
----> Package libcurl.1386 0:7.19.4-4.fc9 set to be updated
----> Package libmodplug.1386 1:0.8.7-1.fc9 set to be updated
----> Package totem.1386 0:2.23.2-16.fc9 set to be updated
----> Package totem-gstreamer.1386 0:2.23.2-16.fc9 set to be updated
----> Package totem-mozplugin.1386 0:2.23.2-16.fc9 set to be updated
----> Package totem-nautilus.1386 0:2.23.2-16.fc9 set to be updated
----> Package totem-xine.1386 0:2.23.2-16.fc9 set to be updated
----> Package v1ggothic-fonts.noarch 0:20090422-1.fc9 set to be updated
----> Package v1ggothic-fonts-common.noarch 0:20090422-1.fc9 set to be updated
----> Package v1ggothic-p-fonts.noarch 0:20090422-1.fc9 set to be updated
----> Package xulrunner.1386 0:1.9.0.10-1.fc9 set to be updated
----> Package yelp.1386 0:2.22.1-12.fc9 set to be updated

--> Finished Dependency Resolution

Dependencies Resolved

<table>
<thead>
<tr>
<th>Package</th>
<th>Arch</th>
<th>Version</th>
<th>Repository</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>firefox</td>
<td>i386</td>
<td>3.0.18-1.fc9</td>
<td>updates-newkey</td>
<td>12 M</td>
</tr>
<tr>
<td>gnome-python2-extras</td>
<td>i386</td>
<td>2.19.1-27.fc9</td>
<td>updates-newkey</td>
<td>51 k</td>
</tr>
<tr>
<td>gnome-python2-gtkhtml2</td>
<td>i386</td>
<td>2.19.1-27.fc9</td>
<td>updates-newkey</td>
<td>19 k</td>
</tr>
<tr>
<td>gnome-python2-libegg</td>
<td>i386</td>
<td>2.19.1-27.fc9</td>
<td>updates-newkey</td>
<td>57 k</td>
</tr>
<tr>
<td>libcurl</td>
<td>i386</td>
<td>7.19.4-4.fc9</td>
<td>updates-newkey</td>
<td>166 k</td>
</tr>
<tr>
<td>libmodplug</td>
<td>i386</td>
<td>1:0.8.7-1.fc9</td>
<td>updates-newkey</td>
<td>171 k</td>
</tr>
<tr>
<td>totem</td>
<td>i386</td>
<td>2.23.2-16.fc9</td>
<td>updates-newkey</td>
<td>2.4 M</td>
</tr>
<tr>
<td>totem-gstreamer</td>
<td>i386</td>
<td>2.23.2-16.fc9</td>
<td>updates-newkey</td>
<td>69 k</td>
</tr>
</tbody>
</table>
--- Package totem-gstreamer.1386 0:2.23.2-16.fc9 set to be updated
--- Package totem-mozplugin.1386 0:2.23.2-16.fc9 set to be updated
--- Package totem-nautilus.1386 0:2.23.2-16.fc9 set to be updated
--- Package totem-xine.1386 0:2.23.2-16.fc9 set to be updated
--- Package vglgothic-fonts.noarch 0:20090422-1.fc9 set to be updated
--- Package vglgothic-fonts-common.noarch 0:20090422-1.fc9 set to be updated
--- Package vglgothic-p-fonts.noarch 0:20090422-1.fc9 set to be updated
--- Package xulrunner.1386 0:1.9.0.10-1.fc9 set to be updated
--- Package yelp.1386 0:2.22.1-12.fc9 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

<table>
<thead>
<tr>
<th>Package</th>
<th>Arch</th>
<th>Version</th>
<th>Repository</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>firefox</td>
<td>i386</td>
<td>3.0.10-1.fc9</td>
<td>updates-newkey</td>
<td>12 M</td>
</tr>
<tr>
<td>gnome-python2-extras</td>
<td>i386</td>
<td>2.19.1-27.fc9</td>
<td>updates-newkey</td>
<td>51 k</td>
</tr>
<tr>
<td>gnome-python2-gtkhtml2</td>
<td>i386</td>
<td>2.19.1-27.fc9</td>
<td>updates-newkey</td>
<td>19 k</td>
</tr>
<tr>
<td>gnome-python2-libegg</td>
<td>i386</td>
<td>2.19.1-27.fc9</td>
<td>updates-newkey</td>
<td>57 k</td>
</tr>
<tr>
<td>libcurl</td>
<td>i386</td>
<td>7.19.4-4.fc9</td>
<td>updates-newkey</td>
<td>166 k</td>
</tr>
<tr>
<td>libmodplug</td>
<td>i386</td>
<td>1:0.8.7-1.fc9</td>
<td>updates-newkey</td>
<td>171 k</td>
</tr>
<tr>
<td>totem</td>
<td>i386</td>
<td>2.23.2-16.fc9</td>
<td>updates-newkey</td>
<td>2.4 M</td>
</tr>
<tr>
<td>totem-gstreamer</td>
<td>i386</td>
<td>2.23.2-16.fc9</td>
<td>updates-newkey</td>
<td>69 k</td>
</tr>
<tr>
<td>totem-mozplugin</td>
<td>i386</td>
<td>2.23.2-16.fc9</td>
<td>updates-newkey</td>
<td>273 k</td>
</tr>
<tr>
<td>totem-nautilus</td>
<td>i386</td>
<td>2.23.2-16.fc9</td>
<td>updates-newkey</td>
<td>36 k</td>
</tr>
<tr>
<td>totem-xine</td>
<td>i386</td>
<td>2.23.2-16.fc9</td>
<td>updates-newkey</td>
<td>52 k</td>
</tr>
<tr>
<td>vglgothic-fonts</td>
<td>noarch</td>
<td>20090422-1.fc9</td>
<td>updates-newkey</td>
<td>2.3 M</td>
</tr>
<tr>
<td>vglgothic-fonts-common</td>
<td>noarch</td>
<td>20090422-1.fc9</td>
<td>updates-newkey</td>
<td>15 k</td>
</tr>
<tr>
<td>vglgothic-p-fonts</td>
<td>noarch</td>
<td>20090422-1.fc9</td>
<td>updates-newkey</td>
<td>2.4 M</td>
</tr>
<tr>
<td>xulrunner</td>
<td>i386</td>
<td>1.9.0.10-1.fc9</td>
<td>updates-newkey</td>
<td>9.0 k</td>
</tr>
<tr>
<td>yelp</td>
<td>i386</td>
<td>2.22.1-12.fc9</td>
<td>updates-newkey</td>
<td>874 k</td>
</tr>
</tbody>
</table>

Transaction Summary

Install 0 Package(s)
Update 16 Package(s)
Remove 0 Package(s)

Total download size: 30 M

Is this ok [y/N]: y

Downloading Packages:
(1/16): vglgothic-fonts-common-20090422-1.fc9.noarch.rpm | 15 kB 00:03
(2/16): gnome-python2-gtkhtml2-2.19.1-27.fc9.1386.rpm | 19 kB 00:01
(3/16): totem-nautilus-2.23.2-16.fc9.1386.rpm | 36 kB 00:02
(5/16): totem-xine-2.23.2-16.fc9.1386.rpm | 51 kB 00:03
(7/16): totem-gstreamer-2.23.2-16.fc9.i386.rpm | 57 kB 00:04
(8/16): libcurl-7.19.4-4.fc9.i386.rpm | 69 kB 00:06

(1%) 57% [--------------------] 11 kB/s 96 kB 00:06 ETA
Install Software From Repositories
Install Software From Repositories

- In Windows, software is downloaded from anywhere and is installed directly; it is a complete package with all the necessary parts.

- In Windows, aside from the “Windows” part, you are responsible for keeping track of all the installed software, the availability of updates and security patches, and to install them. *(Some stand-alone software have automatic update options, however they only apply to that piece of software.)*
Install Software From Repositories

• In Linux, software is normally installed using a package manager that coordinates software installation, including dependencies, as well as versions, updates, and security patches.

• Essentially, *ALL* software installed from your package manager – the “Linux part”, fonts, the desktop, or applications – will be updated and have security patches applied through the repositories soon after they're available.
Install Software From Repositories

- As such, normally when you're installing software and you can install either from a repository or elsewhere, it's preferred to install from the repositories.
Install Software From Repositories

- Some software you'll install on your system won't be in the repositories: However, the important part is to choose a system that has the software you need in its repositories, OR to tolerate the occasional piece of software from “off the homestead” sources while keeping unsupported software to a minimum.
A **software repository** is a storage location from which software packages may be retrieved and installed on a computer.

### Contents

1. Discussion
2. Package Management System vs. Package Development Process
3. Selected Repositories
4. See also
5. References
6. External links

### Discussion

Many software publishers and other organisations maintain servers on the Internet for this purpose, either free of charge or for a subscription fee. Repositories may be solely for particular programs, such as CPAN for the Perl programming language, or for an entire operating system. Operators of such repositories typically provide a package management system, tools intended to search for, install and otherwise manipulate software packages from the repositories. For example, many **Linux distributions** use Advanced Packaging Tool (APT), commonly found in Debian based distributions or **yum**, found in Red Hat based distributions. There are also multiple independent package management systems, such as pacman, used in Arch Linux and **yadm**, found in Sabayon Linux.

As software repositories are designed to include useful packages, major repositories are designed to be malware free. If a computer is configured to use a digitally signed repository from a reputable vendor, and is coupled with an appropriate permissions system, this significantly reduces the threat of malware to these systems. As a side effect, many systems that have these capabilities do not require anti-malware software such as anti-virus software.[1]

Most major **Linux distributions** have many repositories around the world that mirror the main repository.

A new type of Software repositories for personal computers is the "App stores", which is a development of the former software archives. "App Stores" usually have well-developed system of user ranking, certification, payment and updating of software. **Apples** is one of the earliest adopters of the whole concept of "App stores", while the previous implementations often only had some of these functions.
Install Software From Repositories

- System |
  Administration |
  Software Sources
Repository Files

- You can add repository information, normally found on the repository's webpage, to /etc/yum.conf

- Some repositories' web pages have automatic installation links to add the repository to your system, which will add the necessary information to your system
Repository Files

- You can also have separate repository files in `/etc/yum/repo.d` which are named `file.repo`, where `file` normally represents the repository's name.

  - Normally the text to cut & paste can be found on the repository's web page.
Activate Your Firewall and Tune Appropriately
Activate Your Firewall and Tune Appropriately

- Firewalls basically act as a traffic cops, and enforce rules for what type of network traffic is allowed into and out of your computer
Activate Your Firewall and Tune Appropriately

- System Administration Firewall

- Check the services for which you'll allow network traffic, leave those you won't allow unchecked

- Other configurations can be set
Activate Your Firewall and Tune Appropriately

- At the command line and as root, edit the `/etc/sysconfig/iptables` file
  - Manual customization of the file is not recommended
Turn Off Unneeded System Services
Turn Off Unneeded System Services

- If you don't access your computer remotely by certain services, the ones you don't need should be turned off so as to limit them as intrusion vectors by crackers
- System | Administration | Services
- ftp
- sshd
- vnc
- httpd
- sendmail
- netconsole
The **NetworkManager** service is started once, usually when the system is booted, runs in the background and wakes up when needed.

- **This service is disabled.**
- **The status of this service is unknown.**

**Description**

*NetworkManager is a tool for easily managing network connections.*
Turn Off Unneeded System Services

- At the command line and as root, enter the `setup` command, and you'll see this menu; choose “System services”
Turn Off Unneeded System Services

- In the service list, select the targeted services (up/down keys) and activate/deactivate (space key)
- Changes take effect at next boot-up
Limit SSH Access To Only The Users You Allow, And Exclude The root User
Limit SSH Access To Only The Users You Trust, And Exclude The root User

- SSH access should only be for those users you trust with a command line, and who have reason to have it

- Given the power of root, it should not be allowed direct SSH access; further, if root has an unique password, a cracker would have to figure out:
  - A valid SSH account;
  - Its password;
  - Then the root account password.
Limit SSH Access To Only The Users You Trust, And Exclude The root User

- At the command line and as root, open `/etc/ssh/sshd_config`, and add a line with `AllowUsers` followed by the users you allow, separated by spaces.

- Edit the line `PermitRootLogin` so that “no” is entered, and delete the “#” at the beginning of the line; particularly useful if the root password is unique.

- Be sure that there isn't a “#” at the beginning of the lines, or the condition will be ignored.
Apr 26 13:54:26 malak sshd[23323]: pam_unix(sshd:auth): check pass; user unknown
Apr 26 13:54:26 malak sshd[23323]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=202.105.49.16
Apr 26 13:54:26 malak sshd[23323]: pam_succeed_if(sshd:auth): error retrieving information about user tomcat
Apr 26 13:54:27 malak sshd[23320]: Failed password for invalid user root from 202.105.49.16 port 43998 ssh2
Apr 26 13:54:28 malak sshd[23323]: Failed password for invalid user tomcat from 202.105.49.16 port 42446 ssh2
Apr 26 13:54:28 malak sshd[23322]: Received disconnect from 202.105.49.16: 11: Bye Bye
Apr 26 13:54:28 malak sshd[23321]: Invalid user cady from 202.105.49.16
Apr 26 13:54:28 malak sshd[23325]: input_userauth_request: invalid user cady
Apr 26 13:54:29 malak sshd[23321]: pam_unix(sshd:auth): check pass; user unknown
Apr 26 13:54:29 malak sshd[23321]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=202.105.49.16
Apr 26 13:54:29 malak sshd[23321]: pam_succeed_if(sshd:auth): error retrieving information about user cady
Apr 26 13:54:29 malak sshd[23324]: Received disconnect from 202.105.49.16: 11: Bye Bye
Apr 26 13:54:30 malak sshd[23326]: Invalid user marine from 202.105.49.16
Apr 26 13:54:30 malak sshd[23326]: input_userauth_request: invalid user marine
Apr 26 13:54:30 malak sshd[23326]: pam_unix(sshd:auth): check pass; user unknown
Apr 26 13:54:30 malak sshd[23326]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=202.105.49.16
Apr 26 13:54:30 malak sshd[23326]: pam_succeed_if(sshd:auth): error retrieving information about user marine
Apr 26 13:54:30 malak sshd[23324]: Received disconnect from 202.105.49.16: 11: Bye Bye
Apr 26 13:54:30 malak sshd[23325]: Connection closed by 202.105.49.16
Apr 26 13:54:31 malak sshd[23321]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=202.105.49.16
Apr 26 13:54:31 malak sshd[23321]: pam_succeed_if(sshd:auth): error retrieving information about user root
Apr 26 13:54:31 malak sshd[23328]: Failed password for invalid user marine from 202.105.49.16 port 45914 ssh2
Apr 26 13:54:32 malak sshd[23327]: Received disconnect from 202.105.49.16: 11: Bye Bye
Apr 26 13:54:33 malak sshd[23328]: Failed password for invalid user root from 202.105.49.16 port 49276 ssh2
Apr 26 13:54:33 malak sshd[23329]: Received disconnect from 202.105.49.16: 11: Bye Bye
Apr 26 13:54:36 malak sshd[23330]: Invalid user global from 202.105.49.16
Apr 26 13:54:36 malak sshd[23330]: input_userauth_request: invalid user global
Apr 26 13:54:36 malak sshd[23330]: pam_unix(sshd:auth): check pass; user unknown
Apr 26 13:54:36 malak sshd[23330]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=202.105.49.16
Apr 26 13:54:36 malak sshd[23330]: pam_succeed_if(sshd:auth): error retrieving information about user global
Apr 26 13:54:37 malak sshd[23332]: User root from 202.105.49.16 not allowed because not listed in AllowUsers
Apr 26 13:54:37 malak sshd[23332]: input_userauth_request: invalid user root
Apr 26 13:54:37 malak sshd[23332]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=202.105.49.16
Apr 26 13:54:37 malak sshd[23332]: pam_succeed_if(sshd:auth): error retrieving information about user global
Apr 26 13:54:37 malak sshd[23333]: input_userauth_request: invalid user root
Apr 26 13:54:38 malak sshd[23333]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=202.105.49.16
Apr 26 13:54:38 malak sshd[23333]: pam_succeed_if(sshd:auth): error retrieving information about user global
Apr 26 13:54:38 malak sshd[23333]: input_userauth_request: invalid user root
Apr 26 13:54:38 malak sshd[23332]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=202.105.49.16
Apr 26 13:54:38 malak sshd[23332]: pam_succeed_if(sshd:auth): error retrieving information about user global
Apr 26 13:54:38 malak sshd[23334]: Failed password for invalid user global from 202.105.49.16 port 49425 ssh2
Apr 26 13:54:38 malak sshd[23334]: Invalid user marine from 202.105.49.16
Apr 26 13:54:38 malak sshd[23334]: input_userauth_request: invalid user marine
Apr 26 13:54:38 malak sshd[23334]: pam_unix(sshd:auth): check pass; user unknown
Apr 26 13:54:38 malak sshd[23334]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=202.105.49.16
Apr 26 13:54:38 malak sshd[23334]: pam_succeed_if(sshd:auth): error retrieving information about user marine
Apr 26 13:54:38 malak sshd[23334]: input_userauth_request: invalid user marine
Apr 26 13:54:38 malak sshd[23334]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=202.105.49.16
Apr 26 13:54:38 malak sshd[23334]: pam_succeed_if(sshd:auth): error retrieving information about user marine
Apr 26 13:54:39 malak sshd[23332]: Failed password for invalid user root from 202.105.49.16 port 53047 ssh2
Apr 26 13:54:39 malak sshd[23333]: Received disconnect from 202.105.49.16: 11: Bye Bye
Install denyhosts
Install denyhosts

- denyhosts is a Python script that analyses sshd logs to determine which IP addresses have repeated login failures.

- After (or even during) a repeated attack against an sshd server, it will add the IP address in question to the /etc/hosts.deny file, causing the system to refuse future connection requests from that IP address.

- denyhosts only modifies /etc/hosts.deny for sshd, but entries in the file can work for other services as well.
This file contains access rules which are used to
deny connections to network services that either use
the tcp_wrappers library or that have been
started through a tcp_wrappers-enabled xinetd.

The rules in this file can also be set up in
/etc/hosts.allow with a 'deny' option instead.

See 'man 5 hosts options' and 'man 5 hosts_access'
for information on rule syntax.
See 'man tcpd' for information on tcp_wrappers

The portmap line is redundant, but it is left to remind you that
the new secure portmap uses hosts.deny and hosts.allow. In particular
you should know that NFS uses portmap!

# DenyHosts: Sun Apr 26 22:07:50 2009 | sshd: 202.105.49.16
sshd: 202.105.49.16
# DenyHosts: Mon Apr 27 00:35:22 2009 | sshd: 211.103.181.208
sshd: 211.103.181.208
sshd: 216.160.205.138
# DenyHosts: Mon Apr 27 14:43:55 2009 | sshd: 59.125.137.41
sshd: 59.125.137.41
# DenyHosts: Mon Apr 27 20:28:56 2009 | sshd: 218.240.43.35
sshd: 218.240.43.35
# DenyHosts: Tue Apr 28 02:20:57 2009 | sshd: 74.213.167.92
sshd: 74.213.167.92
# DenyHosts: Tue Apr 28 04:56:00 2009 | sshd: 218.213.69.172
sshd: 218.213.69.172
# DenyHosts: Tue Apr 28 09:47:32 2009 | sshd: 202.125.47.222
sshd: 202.125.47.222
# DenyHosts: Tue Apr 28 17:24:03 2009 | sshd: 202.104.151.151
sshd: 202.104.151.151
# DenyHosts: Wed Apr 29 02:50:07 2009 | sshd: 221.8.79.67
sshd: 221.8.79.67
# DenyHosts: Thu Apr 30 21:19:08 2009 | sshd: 202.52.108.220
sshd: 202.52.108.220
# DenyHosts: Fri May 1 12:47:09 2009 | sshd: 115.127.0.130
sshd: 115.127.0.130
Install denyhosts

- System | Administration | Add / Remove Software
- Enter denyhosts into the search line
- Select the software & click on “apply”

- At the command line and as root, “yum install denyhosts”
Install denyhosts
su
Password:
# yum install denyhosts
Loaded plugins: refresh-packagekit
Setting up Install Process
Parsing package install arguments
Resolving Dependencies
  --> Running transaction check
  ----> Package denyhosts.noarch 0:2.6-13.fc10 set to be updated
  ----> Finished Dependency Resolution

Dependencies Resolved

<table>
<thead>
<tr>
<th>Package</th>
<th>Arch</th>
<th>Version</th>
<th>Repository</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>denyhosts</td>
<td>noarch</td>
<td>2.6-13.fc10</td>
<td>fedora</td>
<td>97 k</td>
</tr>
</tbody>
</table>

Transaction Summary

<table>
<thead>
<tr>
<th>Installation</th>
<th>Updates</th>
<th>Remove</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Package(s)</td>
<td>0 Package(s)</td>
<td>0 Package(s)</td>
</tr>
</tbody>
</table>

Total download size: 97 k
Is this ok [y/N]: y

Downloading Packages:
denyhosts-2.6-13.fc10.noarch.rpm | 97 kB 00:00
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing: denyhosts

Installed:
denyhosts.noarch 0:2.6-13.fc10

Complete!
#
Other Strategies
Other Strategies

- Install SELinux and Operate it in enforcing mode
- Use NoScript -- firefox plugin which manages javascript, java, flash and other plugins
- Encrypt your filesystem (such as with TrueCrypt);
- Remove junk that accumulates over time (BleachBit has a wide variety of "cleaners" for all sorts of junk that accumulates)
- Don't automount devices (important for last strategy)
- Close root terminals when you're finished (important for last strategy)
Protect Your Computer's Physical Integrity
Protect Your Computer's Physical Integrity

- Whether you're a beginner or an expert, there's always someone out there who knows more than you on a given subject ... give them physical access to your computer, and they could do all sorts of harm, and/or steal personal information, and/or erase your data, and/or install malware, etc. ...

- *Sounds too simple to be useful?*
They could insert one of these into your computer ...
Or one of these ...
Or since we're on the subject, why not a hard drive, either a portable USB unit or a traditional hard drive ...
Protect Your Computer!

- It's pretty easy to turn off or unplug a computer and insert a LiveCD or LiveUSB key, or a USB hard drive, or, if you have time, a traditional hard drive; then it's easy to access your personal files or install malware ...

... and it's GAME OVER!
Summary
Summary

- Use a router
- All users should have their own account
- Manage Passwords
- Control Access to Your Desktop
- Perform Updates Regularly
- Install Software From Repositories
- Activate Your Firewall and Tune Appropriately
- Turn Off Unneeded Services
- Limit SSH Access to only the users you allow, and exclude the root user
- Install DenyHosts
- Other Strategies
  - Install SELinux and Operate it in enforcing mode
  - Use NoScript
  - Encrypt your filesystem
  - Remove accumulated junk
  - Don't automount devices
  - Close root sessions at the command line
- Protect your system's physical integrity
Please note:

- This presentation was composed on Fedora 9, 10, 11, 12 and 14 systems.

- If you really wanna know, I'm an environmental field techie who happens to like Linux.
Thank You!
Questions and Comments