Linux Desktop Security
or
Just Enough Information About Linux Security to Keep Your Linux Desktop Safe

Office of Academic Information Technologies
Iowa State University
Overview

• I thought Linux was more secure than Windows!
• Threats to Linux machines
• Blocking unwanted network access
• Firewalls
• Services
• System Updates
I thought Linux was more secure than Windows! (Yes, it is.)

• User vs. administrator
  – Only the root user can change the operating system
  – Normal users cannot install software or change system settings without the root password
  – More difficult for viruses to spread

• Modular vs. monolithic
  – Commands, utilities, even the desktop run separately from the Kernel
  – Security updates are easier, quicker to develop
Threats to Linux machines

- Breakins at the console
- Brute-force password attacks
- Buffer overflows in network services
- Trojan horses
Steps to make Linux even more secure

• Secure the console
• Set good passwords
• Secure the network connection
• Use firewalls to minimize network attacks
• Restrict the number of services running
• Keep the system up to date
Securing the console

• Physical security
  – Security cables?
  – Machine in a secure location

• Restrict booting from removable media
  – Turn off boot from floppy/CD/USB/IEEE 1394 in BIOS setup
  – Set BIOS password
  – Do you really need a floppy/Zip drive on a server?
Securing the Console

● Use a screensaver with password lock

  – If you leave your session logged in, anyone who sits at the machine can be you...

  ● KDE: Control Center -> Appearance & Themes -> Screensaver
     Turn on “Require password to stop screensaver”

  ● Gnome Desktop: Preferences -> Screensaver
     Turn on “Lock screen after N minutes”

  ● Other desktops: use xautolock or run xlock when you walk away
Passwords

- It's important to have good passwords on all accounts, but especially root (since it can install software and change system settings).
- It's even more important if you allow remote access to your machine; crackers on the network will use brute-force attacks to try to break in to your system.
Good passwords...

• ...use MD5 and shadow passwords (set by default during installation, don't change them!)
• ...contain multiple character classes (lower case, upper case, numbers, punctuation)
• ...are not short. The longer they are, the better. **One approach:** use a passphrase rather than a password.  (Spaces are OK in passwords.) **Another approach:** use first letters (and numbers) of a passphrase.
Changing Local Passwords

- In a shell, type

  ```
  passwd username
  ```

  (The username is optional when changing your own; root can change other users' passwords.)

- Choose Preferences -> Password

- root can click System Settings -> Users and Groups.
  (Passwords are under the properties of a user.)
Passwords under Iowa State Linux

• If you use Iowa State Linux on a machine and log in with your ISU NetID and password, don't change that password using the previous tools.

• Instead, log into http://asw.iastate.edu
  – Choose Manage User yourNetid
  – Choose Change your Password
  – Fill in the form
Securing the Network

- The first and easy step to securing the network connection is to configure the /etc/hosts.deny and /etc/hosts.allow files.
- They determine what machines can connect to what services on your machine.
- These are ordinary text files and can be modified with any editor, but must be modified by root.
Configure /etc/hosts.deny

• Start a terminal window
• Enter the `su` command to become root
• Edit the hosts.deny file by typing `nedit /etc/hosts.deny`
• At the end of the file, add the line

```
ALL:ALL EXCEPT localhost:DENY
```
• Exit and save the changes.
Configuring /etc/hosts.allow

- As root, enter the command
  
  nedit /etc/hosts.allow

- At the end of the file, enter
  
  sshd:ALL:ALLOW

- Exit and save the changes. The machine will now respond only to ssh, scp and sftp requests, except for those coming from localhost.
More info on /etc/hosts.allow

- You can configure hosts.allow to allow access only to certain addresses, or to start some process when someone from a particular address or domain attempts to connect to some port on your machine.

- Some good references:

  http://linux.about.com/od/commands/l/blcmdl5_hostsal.htm
Preventing root login

• Since root is all-powerful on your system, it's best to prevent root from logging in remotely.

• If you need to administer the machine remotely, log in as a normal user and use the `su` command to become root.

• Edit the file `/etc/securetty` and comment out all the choices but `console` by putting a pound sign in front of the name.
Firewalls

- Purpose of a firewall is to restrict network traffic to a machine or network segment to improve security and network performance
  - **Hardware firewall**
    A device, usually between an Internet connection and a local area network (LAN)
  - **Software firewall**
    Software running on a desktop or server machine that rejects certain types of network traffic
Why do I need a software firewall?

- Hardware firewalls can be compromised
- A software firewall will protect the workstation even if the hardware firewall is compromised
- A software firewall also protects the machine against compromised machines (i.e. laptops) on the local network
When can't I use a firewall?

- Some services (like Samba) may use unspecified ports
- Some applications want to use arbitrary ports
What are ports?

- Think of your network connection as a highway.
- Each type of traffic needs to be in its own lane (carpools and buses in the carpool lane, bicycles in the bike path, big trucks in the fast lane, cars in whatever's left)
- A port is analogous to a lane on the highway; different types of traffic (http, ftp, ssh, etc.) use different ports.
What ports need to be open?

- The ports that support the services you need to use and/or offer to others. Some examples:
  - SSH (remote access to your machine): 22
  - FTP (file sharing server): 21
  - Web server: 80
  - X (display graphics on remote machines): 6000

- See /etc/services for an exhaustive list.

- Everything else should be closed!
Setting Security Level (Red Hat)

- System Settings -> Security Level
- Check boxes for the services running on your machine
Activating iptables

- You must also activate the iptables service
- Choose System Settings -> Server Settings -> Services
- Turn on iptables in runlevels 3 and 5
Configuring iptables

- Unless you're running a server on your machine the simple GUI should be sufficient, but...
- Good references for more sophisticated use of iptables:

  http://www.cae.wisc.edu/fsg/linux/linux-iptables.html

What are services?

- Services are special applications that start before anyone logs in to the machine (often referred to as “daemons”) that provide, well, services...

- Examples:
  - Web server (httpd or Apache)
  - File services (samba, NFS, ftpd)
  - Print services (lpd, CUPS)
  - Remote access (telnetd, sshd, vncserver)
  - Management tools (crond, rhnspd)
Why can they be dangerous?

- Many services offer themselves not only to the local machine, but to other machines on the network.
- If a flaw exists in the program providing the service, an attacker can exploit this flaw and break in to the machine.
- RULE: don't run any services you don't need.
- RULE: if you're running a service, restrict access as much as possible.
A Major Assumption

- This discussion assumes that you're using a “SysV-styled” Linux: Red Hat, SuSE, Debian, etc.

- We will prefer using Red Hat-styled graphical tools.

- It's also possible to use a “BSD-styled” Linux, in which initialization of services is done a lot differently.
Just enough about how services work

- A Linux machine starts up in one of five runlevels (these values apply to Red Hat, YMMV)
  - 1 – single-user mode
  - 2 – multiuser without networking
  - 3 – normal multiuser text mode
  - 4 – custom runlevel, not normally used
  - 5 – normal multiuser, graphical desktop
- Default is chosen in /etc/inittab
Services are started from

- `/etc/inittab` (configuration file)
- `/etc/rc.d/rc.sysinit` (configuration file)
- `/etc/rc.d/rcN.d` (directory for runlevel N containing symbolic links to startup scripts in `/etc/rc.d/init.d`)
- `xinnetd` (service that starts network services as you need them)
Identifying running services

- redhat-config-services
  
or
  System Settings -> Server Settings -> Services
  (in Red Hat Linux with KDE)
  
or
  /usr/sbin/ntsysv
  
or
  /sbin/chkconfig --list
Using redhat-config-services

- This GUI frontend makes it easy to start, stop and disable services
- To enable at runlevel startup, check the box
- Use Start and Stop buttons to start and stop the service
- Don't forget to click Save!
Using ntsysv

- `/usr/sbin/ntsysv` acts like redhat-config-services at a text console

- Use `{Tab}` to move cursor, `{Space}` to click, `{F1}` to see description
What services don't I want?

- The r* services (rlogin, rsh, etc.)
- telnet
- ftp
- NFS-related services (portlock, nfslock, netfs)
- sendmail – almost all Linux mail clients can be configured to use an external SMTP server (like mailhub.iastate.edu)
Shutting down unnecessary services

- **redhat-config-services**
  - Highlight service;
  - Click Stop;
  - Uncheck the item.
  
    You'll have to do this separately for each runlevel.

- From a shell, use `/usr/sbin/ntsysv` or the commands

  ```
  /etc/rc.d/init.d/servicename stop
  /sbin/chkconfig servicename off
  ```
What if I need those services?

• Look at substitutes
  - telnet, rlogin, rsh – use ssh instead
  - ftp – use scp and sftp instead
  - nfs – use Samba or AFS instead
  - Sendmail – Configure your mail client to send mail directly to an SMTP server

• Secure the service using its configuration files
Configuring Mail Clients

- For more on configuring mail clients for ISU, see http://tech.ait.iastate.edu/linux/linuxed/Mail/Mail.pdf
- If you use another ISP, configure your mail client for their SMTP server (even AOL has one now... )
Identifying unknown services

If you're lucky, you'll get the info from redhat-config-services or /usr/sbin/ntsysv. If you're not...

- `what is` command
- `man` command
- Search for command in `/etc/rc.d/init.d` and read the script
- Google search
System Updates

• Just like that *other* operating system, it's important to make sure that your Linux system and utilities are kept up to date

• Fortunately, the scripting and scheduling capabilities of Linux, plus (at ISU) the use of the Red Hat Network, make it easy to do.
Why is keeping up to date important?

- Security flaws in Linux and its utilities are constantly being discovered.
- Once a flaw is discovered, crackers can exploit it to break into your machine.
- It's critical to install system updates as soon as possible after they become available – but how do you keep up with it?
Updates for RHEL (Red Hat Network)

- Red Hat uses a system called the Red Hat Network to manage machines and distribute updates.

- ISU community machines using RHEL can update manually or automatically via RHN.
Getting your RHN account

- Your Red Hat Network account is used to manage RHEL machines you're responsible for.
- Go to https://www.linux.iastate.edu/rhel
- Read the instructions under “Getting Started”
- Request a Red Hat Network account via https://asw.iastate.edu/cgi-bin/acropolis/request/rhn/rhnregister
Activation Keys

- An activation key is used to register an RHEL machine with the Red Hat Network.
- Your RHN account has a personal activation key that can be used for up to three personal machines.
- If you manage ISU-owned machines, you should request a group activation key. See

  https://asw.iastate.edu/cgi-bin/acropolis/request/rhn/rhnmodify
Registering your machine

1. Login as root

2. Obtain a copy of the bootstrap script
   
   ```
   wget http://cyclops.iastate.edu/pub/bootstrap.sh
   ```

3. Run the bootstrap.sh script to register the system

   ```
   sh bootstrap.sh --register XXXXXXXXXX
   ```

   where XXXXXXXXXX is replaced by either your personal or a group activation key.
Updating Software

• To update a package, become root and type

```bash
up2date -u packagename
```

For example, to update up2date itself (which you'll need to do after registering the machine) type

```bash
up2date -u up2date
```
Updating Everything

• To update all the packages installed by the Red Hat installer or by using up2date, do this:
  – Start a shell;
  – Become root with the su command;
  – Type

    up2date --update

• This approach is nice because it's scriptable and can be scheduled using cron (more info later)
Checking errata

Errata are packages that have been updated for security flaws or bug fixes. If errata are available for your machine, the red exclamation point will appear in the panel:

● Go to http://rhn.redhat.com
● Log in with your RHN username and password
● Click your system group name
● Click “Systems”
Checking Errata, Page 2

- Click your computer name
- Click “Errata”
- Check the errata items to update
- Click “Apply Updates”

Now you can either

- Type `rhn_check` at a shell prompt
- Wait for `rhnsd` to check in on its own
Setting automatic updates

- Login to *rhn.redhat.com*
- Click the system group name
- Click the Systems tab
- Select your machine
- Click the Properties tab
- Check “Automatic application of relevant errata”
- Click “Update Properties”
Update repositories

- These servers provide updated copies of core Linux files and utilities
- They may also provide additional items not included with the stock Linux distribution
- Updates are generally installed using apt or yum, which can be automated if desired
- RHEL can use up2date with additional repositories
Adding the Dag Wieers Repository

• Additional software packages are available through up2date from the Dag Wieers repository

• We maintain a mirror here on campus for your convenience.

• You will need to add these lines to /etc/sysconfig/rhn/sources

### Dag RPM Repository for Red Hat Enterprise Linux
yum dag http://dag.linux.iastate.edu/dag/redhat/el3/en/i386/dag
Adding Dag, part 2

• Now, as root enter these two lines:

```
rpm --import \nhttp://rh-mirror.linux.iastate.edu/pub/dag/RPM-GPG-KEY.dag.txt
```

• The command `up2date -i packagename` will now search the Dag Wieers mirror as well as the Red Hat Network site.

• For more info on the repository, see http://dag.wieers.com/home-made/apt/
Updating other distributions

• Linux distributions from sources other than Red Hat will have different update procedures

• Other distributions may have update repositories associated with their Websites

• Older versions of Red Hat and Fedora may be updated through www.fedoralegacy.com
What is apt?

- The apt package manager started in the Debian distribution of Linux but has been ported to Red Hat and others.
- apt, like up2date, allows easy installation and updating, but doesn't require a Red Hat Network subscription, and repositories can be kept on a local network or even on CD-ROMs.
- Unlike rpm, apt will figure out dependencies and automatically install all the parts you need (even from multiple repositories)
Installing apt

• apt comes with Debian and Fedora Core 3, so you don't need to install it

• To install in Red Hat Enterprise Linux, configure the Dag Wieers repository and type as root:

  up2date -i apt

• For earlier versions of Red Hat and Fedora, see http://www.fedoralegacy.org/docs/apt-rh9.php
Some Red Hat/Fedora/Yellow Dog repositories

- Dag Wieers*
  http://dag.wieers.com/home-made/apt
  Local mirror: http://dag.linux.iastate.edu

- Dries*
  http://dries.studentenweb.org/apt/

- FreshRPMs
  http://freshrpms.net

- NewRPMs
  http://newrpms.sunsite.dk/

- ATrpms
  http://atrpms.net

* Supports Red Hat Enterprise Linux
Configuring apt repositories

- Links to apt repositories are kept in the directory
  /etc/apt/sources.list.d

- Remove comments from the version appropriate
to your distribution of Linux
Sample: dries.list

### See http://dries.studentenweb.org/apt/ for more information.

### Red Hat Fedora Core 2
#rpm http://dries.studentenweb.org apt/fedora/fc2/i386 dries

### Red Hat Fedora Core 1
#rpm http://dries.studentenweb.org apt/fedora/fc1/i386 dries

### Red Hat Enterprise Linux 3
rpm http://dries.studentenweb.org apt/fedora/el3/i386 dries

### Red Hat Linux 9
#rpm http://dries.studentenweb.org apt/fedora/rh9/i386 dries
Using apt

• To install a new package, as root enter the following:
  
  ```
  apt-get update
  apt-get install packagename
  ```

• To upgrade all packages on the system, type
  
  ```
  apt-get update
  apt-get upgrade
  ```
Automating updates with cron

To use cron, the `crond` service must be running on your computer.

- For system updates, log in as root
- If the task is complex enough that it can't be done with a single command, create a script that does the task (we'll call our example script `/usr/local/bin/getupdates`, though you can call it anything you want)
Using cron, part 2

• Create a file containing lines like this:

```
# Automatically update at midnight on Mondays
0 0 * * 1 /usr/local/bin/getupdates
```

- Minute to run job
- Hour to run job (0-23, 0 is midnight)
- Day of the month (1-31 or * for all)
- Month (1-12 or * for all)
- Day of the week (0-6, 0=Sunday or * for every day)
- The command to run at that time
Using cron, part 3

• Make the script executable:
  chmod 755 scriptfilename

• Tell cron to execute the script as the user root (since it's a system update) when it's time:
  crontab -u root cronfilename

If you turn your machine off at times, you should also make sure that the anacron service is running; it will make sure missed jobs are run.