

Host Based Security Best Practices

(Last Updated 2004/03/10 MCA)

The following is a list of guidelines to follow to protect a machine while installing and using various operating systems and applications. This list is not complete, but following these steps will help keep hosts secure as they are installed and used on the local network or the Internet.

In general, the following steps should be taken for every host or device which is placed on the network, regardless of operating system. More detail on how to accomplish the various steps for each operating system are listed in later sections.

- 1) install and configure a host based firewall, if available
- 2) install and keep up (continuously) with patches for the firmware or operating system
- 3) make and test backups of the system in a consistent manner
- 4) configure and continue to monitor logs on the device
- 5) disable services and accounts which are not being used, or are no longer necessary
- 6) choose good passwords for any accounts on the system, and change any default or well known accounts on the machine
- 7) replace insecure services (such as telnet, rsh, or rlogin) with more secure alternatives such as ssh
- 8) restrict access to services which cannot be disabled where possible

Some of these steps will be discussed in detail in the sections for each operating system, and some will be discussed later in separate sections.

Before Installation Of The Operating System

Before you begin any installation, be sure you have the following available :

- 1) any operating system CDs or install media that you'll need, including license keys
- 2) a firewall to protect the host while you install everything (if the host will be connected to a network during the install)
or all current patches and updates for the operating system on a CD (for a system which will not be connected to the network during installation)
- 3) a list of services the machine will provide, or a list of services which should be disabled after the machine is installed

A firewall is necessary to protect the host while you install the operating system and all necessary patches if you plan to have the host connected to a network during installation. Most operating systems are vulnerable to compromise when they are installed, and require many patches and updates before they can safely be allowed on the network. If a host is connected to the department network without direct protection, even if the department has a firewall which protects the general network from outside problems, the

host can **still** be compromised by another machine on the department network while it is being installed and configured.

When possible a bridging firewall should be used, as this type of firewall doesn't require additional network configuration, and will allow you to install the additional software and patches safely, without worrying about incoming attacks. In addition you can install the host in place with this type of firewall, and it need never know that a firewall sits between it and the network.

You can also use a NAT-based box under these specific circumstances to protect the machine while you are installing patches and configuring things, but note that we do **not** recommend a NAT based firewall for protection of a department or even a group network. In this particular instance, however use of a small NAT “router” can be used to hide a single host while it is being installed. Most such boxes are easy to set up and use to bridge between the host being configured and the local department network.

If you choose to use a CD to install patches before you connect the machine to a network, make sure that the network cable is unplugged from the machine, and do not rely on software to prevent the machine from being accessible on the network.

By the end of 2003, the average survival time of an upatched, uncompromised Windows machine on the Internet is down to minutes, and under some circumstances seconds. Because it takes longer than this to install patches, an install must be done either behind a firewall or without a network connection.

The OSU Engineering Group is blocking a variety of ports at the Internet borders in order to increase the security of hosts on the OSU network, for a current list please see [INSERT URL]. Despite these blocks, all departments are encouraged to have their own departmental or workgroup firewall, to protect their hosts.

Guidelines for installing various specific operating systems are given in the following sections.

Microsoft Windows

The versions of Microsoft Windows based on the Windows NT kernel (Windows NT, Windows XP, and Windows 2003 Server so far) are the most attacked operating systems on the Internet. On the OSU network due to the number of laptops which are brought onto the OSU network each day, compromises and infections are as likely from inside the network as from outside.

The versions of Microsoft Windows which do not use the NT kernel (Windows 3.x, Windows 95, Windows 98, and Windows Me) have been retired by Microsoft, and are no longer supported. As a result, security updates are no longer available, and these operating systems should not be installed or reinstalled on machines.

A number of checklists have been written about how to securely install a Windows system, and several will be listed at the end of this document. The following checklist is a bare minimum list to follow when installing a Windows system, and you should consult other checklists for more ideas.

Since Windows systems are the most attacked systems on the Internet, it is important that any install images used to mass build systems are updated **frequently**, and that all Windows systems are configured with current virus protection and system/application updates before they are attached to the Internet. See the section on virus protection later in this document for more information.

1) Disable the **guest** account

Go to **Start->Control Panel->UserAccounts** and verify that where the guest account is listed, it says "Guest account is off". If it does not say this, turn the guest account off by choosing it in the account list below, and clicking the button which says "Turn Off The Guest Account".

The guest account should **not** be allowed unauthenticated access to the Internet. If you have a lab where you wish to use the guest account as the primary login for most users, that lab **must** require some other form of user-specific authentication before the computer can gain access to the Internet. This can be in the form of a TOAD (contact security@net.ohio-state.edu for more information), or some other box which blocks Internet access until a user has authenticated with a username and password. Please note such authentication log data must be kept for a minimum of 30 days.

2) Configure an update method to install patches, or install them from CD

To configure **Windows Update** go to **Start->Control Panel->System** and choose the **Automatic Updates** tab. Note that Windows Update will only install each update once, and because Microsoft sometimes modifies updates, this is not the most reliable method to receive automatic updates. We recommend the use of the Microsoft Baseline Security Analyzer to install updates for Windows and other installed applications.

3) Disable unused system services

Go to **Start->Control Panel->Administrative Tools->Services** which will bring up a list of services available on the machine, and their current status.

You should specifically make sure that neither **IIS** (especially the FTP server) nor any

form of **SQL Server** are installed and started on any machine other than a server specifically designed to run these services. In addition, any sort of proxy services should be disabled as well.

The **Internet Connection Firewall (ICF)** should be started automatically (this is the default on newer service packs of Windows XP, but should be verified).

The **Telnet** service should be disabled, and marked as a **manual** service.

The **Universal Plug And Play** service should be disabled, and marked as a **manual** service.

4) Verify the appropriate Local Security Settings

Go to **Start->Control Panel->Administrative Tools->Local Security Settings** and verify that they are set according to the security policies for your local network.

5) Check the Windows Firewall settings, and make sure that it is configured

Go to **Start->Control Panel->Network Connections** and choose the network connection that corresponds to your Internet connection. Double-click, choose the **Properties** button and then choose the **Advanced** tab. To enable the firewall for this connection, make sure that the box under **Internet Connection Firewall** is checked. For more information about the Internet firewall, click on the link marked **Learn more about Internet Connection Firewall**.

6) Be sure you have chosen a good password for all accounts which will have Administrator priviledges

No accounts under Windows should ever be configured with a blank or default password. In addition, passwords should be a minimum of 10 characters, and should use at least 1 or more special characters (i.e. not an alphabetic or numeric character).

7) Other links to consult when installing

<http://www.microsoft.com/security>

http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/mb_sahome.asp

UNIX Operating Systems

Most UNIX and Linux systems stem from either the Berkeley (BSD) family or the AT&T (System V) family of UNIX operating systems. Depending upon which specific version of UNIX you're installing, things will be in slightly different places.

This section discusses in a broad manner security issues related to most UNIX systems, and the following sections will deal with where to find more specific information on each specific version.

The general steps you should follow on all UNIX systems are as follows.

- 1) Choose a good root password
- 2) If the system includes firewall software, install and configure it as soon as possible
- 3) eliminate any accounts which have blank passwords
- 4) eliminate any unneeded services on the machine
- 5) install any applicable patches and security updates

Each variety of UNIX tends to have its own type of firewall software, and most include both a command line interface and a graphical interface. Many versions of Linux will encourage you to configure one as you install the operating system, which is a good idea. At least one firewall (**ipf**) is available for a variety of different versions of UNIX, and allows for relatively consistent firewall configuration across multiple varieties of UNIX.

Some versions of UNIX may still install accounts which are to be used for demonstration purposes, and have no password by default. This tends to be common on IRIX systems, and the `/etc/passwd` and `/etc/shadow` (if it exists) files should be checked immediately after installation to verify these accounts (such as **demo** and **guest**) don't exist, or at least have passwords assigned to them.

Systems also may still have default passwords, which should be changed immediately upon installation of the system (this holds true for devices such as switches, routers, etc. as well as for hosts). If the system did not ask you to assign a password to a particular account, and that account appears to have a password, it's probably a good idea to change the password.

Finally, services or processes on UNIX systems are generally run via one of two methods. The **inetd** (or **xinetd**) "superserver" (hereafter referred to as simply **inetd**) is a process which continues to monitor a number of ports, and when someone requests a service running on a specific port, it hands off the connection to a daemon (server) specifically designed to handle the request. Most often services such as **telnet**, **ftp**, and **finger** are run via **inetd**. In both System V and BSD varieties, the configuration file is usually either `/etc/inetd.conf`, or the configuration information is found in the `/etc/xinetd` directory (depending upon which version of **inetd** is installed).

The **inetd** method of starting services tends to be slow and isn't designed for high throughput servers such as web or mail servers which see heavy use. For heavy use the process which handles the requests is often run as a standalone server, which sits on a specific port (such as port 80 for a web server) and processes requests as quickly as possible. To start these at system boot, an overall manager called **init** is started to

oversee management of processes on the system under different run levels.

AT&T/System V

These processes are generally started by scripts located in the `/etc/init.d` directory, and are generally started when the system boots, or when it changes from one run level to another (such as when a system is taken from system maintenance mode to user mode). When the system's run level changes, the **init value** changes, and this causes a variety of scripts to be run from the `/etc/rcX.d` or `/etc/rc.d/rcX.d` directory, where **X** is the value of the new run level.

Each script in `/etc/init.d` is designed to start a specific server, or a related set of servers. This script is called with a “**start**” argument to start the service, and a “**stop**” argument to shut down the particular service. The scripts in the `/etc/rc.X` directories are generally just links to the script in `/etc/init.d`, and start with either an **S** (to **start** the script when entering that run level) or a **K** (to **kill** the script when leaving that run level). The scripts are executed in ascending numeric order.

So for example a script named `/etc/rc2.d/S95sshd` would be executed with the “start” argument when the system enters run level **2**, and would be started after any scripts numbered 0 – 94 (the numbering system today is somewhat arbitrary, and is generally just used to start scripts in a particular order in relation to other scripts at that same init level). A script named `/etc/rc.d/rc0.d/K07httpd` would be executed with the “stop” argument early upon entering run level **0**, and would probably kill any processes associated with the http service.

Init levels vary from system to system (in general used most commonly are levels 0 through 6, s, and S) but in general most systems run in level 3 (Solaris) or level 5 (Linux) when the system is in multiuser, “general use” mode. The **init** process is usually process number 1 under UNIX, and among other things controls logins on the available consoles. The configuration information for the processes started by init can usually be found in the `/etc/inittab` file.

BSD

The BSD family of operating systems (at this point FreeBSD, OpenBSD, and NetBSD mainly) use a different method of starting processes at boot or system startup time. There are usually a series of scripts named `/etc/rc.xxxxxxxx` which are run out of the master resource control file, `/etc/rc`. These files are also shell script files which are similar to the contents of the `/etc/init.d` files above in that they test for the existence of various binaries and configuration files, and if things are in order start up processes when the script is executed (mainly at boot time).

Common scripts will often include `/etc/rc.local` which is designed for local services (not included by default with the operating system), and the configuration file for these scripts is generally located in `/etc/rc.conf`. The configuration file contains a list of variables which allow you to turn various services on and off, for example the line `apachessl=YES` would be used by the other `rc` scripts to start the Apache SSL server (and related processes, if any) at boot time. These scripts are not run in general when the run level is changed on BSD systems, and in general no **kill** scripts exist for processes under BSD. Processes are sent a **HUP** or **KIL** signal when they are expected to shut down.

Linux

There are a number of distributions of Linux, and many use at least slightly different methods of package management, updates, and initialization scripts. The two major current distributions are RedHat and SuSE, although RedHat has just split from a publicly available distribution, to Fedora (publicly available) and RedHat Enterprise series (commercial, restricted license). These will probably receive their own separate sections at some later point in time.

RedHat Linux

In general, RedHat includes a variety of scripts which can be used to start and stop various services, accessed through a user interface (such as Gnome or KDE, the two major desktop systems under Linux). In addition, the basic service startup scripts are accessed either via **inetd**, **xinetd**, or the standard UNIX **init** mechanism.

Upon installation, the install process will request that you choose a particular level for the firewall, which can generally be **none** (not recommended), **medium**, or **high**. Under each level, the firewall can and should be adjusted to allow incoming connections to system services which should be available to the Internet on the server. The firewall parameters can be further adjusted once the machine is installed, by altering the configuration files or using the graphical interface to adjust incoming services.

Workstations in general should use either the medium or the high settings, and should not allow any incoming services (again these settings can be adjusted later). The software used to control the firewall seems to change with almost every major new kernel version, so you should consult the manual pages for the current firewall system in use, or use a graphical interface (if it exists) to alter the firewall settings.

Both RedHat and SuSE currently use the **/etc/rc.d/rcX.d** directories to hold the init scripts, and general system configuration information can usually be found in the **/etc/sysconfig** directory.

The default root (administrator) environment under RedHat uses the Gnome desktop system, and there are a number of helper scripts which are used to enable and disable the various services and daemons. After you install the operating system, use the following guidelines to secure your RedHat system. While it's not as critical to do the following steps behind a firewall at this point when installing Linux, it is still a good idea to consider this before you begin.

- 1) Configure the RedHat up2date program (if available) to download updates, although this will only work through April of 2004. Either click on the red “!” ball in the bottom corner, or select **System Tools->RedHat Network** to register your host.
- 2) Make sure you've installed a good password for any accounts which will have root capabilities (either directly or through sudo).
- 3) Verify that the RedHat firewall set up during the install is working. Check the current firewall configuration by selecting **System Settings->Security Level** and verify the security level is set to either “Medium” or “High”. If you need to customize the settings for particular services, change “Use default firewall rules” to “Customize” and adjust the ports in the lower section to allow inbound connections to certain services.

- 4) Disable system processes which are not in use and prevent them from restarting. For Gnome the procedure is something like :
 1. Select **Server Settings->Services**
 2. Select the service you wish to affect by unchecking its checkbox
 3. Press the “Stop” button to halt the process that is currently running (some processes may not be stoppable directly, if they run out of the xinetd process you should disable the xinetd process as well)
 4. After doing this with all processes you wish to disable, choose **Save** from the File menu.

SuSe Linux

SuSe Linux uses their system installation and configuration program called YaST to manage most parts of the install and configuration of the system. In SuSe Linux version 9.0, patch management is first configured during install via the YaST Online Update screen.

Both RedHat and SuSE currently use the **/etc/rc.d/rcX.d** directories to hold the init scripts, and general system configuration information can usually be found in the **/etc/sysconfig** directory. Before editing these scripts directly, make sure you understand what you're doing, and that the **YaST Control Center** tools won't do it for you.

If the computer is currently connected to the internet from behind a firewall of some sort, you can do the initial update while you are installing the system. If the machine is not currently connected to the internet from behind a firewall during the install, you can configure when the system should do automatic updates by choosing the button marked “Configure Fully Automatic Update”. You can configure the automatic update to install only patches if you wish to retain control of upgrading the system software.

Choose “Manually Select Patches” if you want control over which patches are installed, but in most cases you should deselect this box to allow fully automatic updates to occur. You should probably choose an update server closer to home, or if you have your own update and patch server, you can also add that host as a viable update server.

When adding users to a SuSe system, make sure if the box marked “Auto Login” is checked that you uncheck this box. Otherwise when the system is booted, the user will automatically be logged in every time. Users should be required to log into the system and log out, even if the system will only be used by a single user.

By default, SuSe 9.0 uses the KDE desktop, and the following assumes that you are using the default KDE installation.

- 1) Examine and choose an appropriate **Security Setting** for the machine, by accessing **Start -> YaST -> Security And Users -> Security settings**. For most cases, the "Workstation" setting will suffice.
- 2) Configure and verify the firewall settings, and that the firewall is activated by using the firewall helper via **Start -> YaST -> Security And Users -> Firewall**. If the firewall has not been previously configured, a series of setup screens will help you

configure and start it. After you have configured and installed the firewall the first time, a screen which asks whether you wish to stop and remove the firewall, or reconfigure the firewall will display. This means the firewall is active, and you can reconfigure it to add or remove services.

- 3) Configure YaST Online Update, or verify the configuration by accessing the YaST Control Center via **Start -> YaST -> Software -> Online Update**. Note that if you do not have an active internet connection it will not be able to pull the current list of update servers. This step is best done with the machine behind a firewall of some sort (or after you have configured the firewall), at least for the initial update.
- 4) Make sure you've installed good passwords for any accounts which will have root access. See the note above about the "Auto Login" feature, and why you should not use it.
- 5) Verify system services settings via **Start -> YaST -> Network Services**. In particular, verify that the "Network Services (inetd)" services are all turned off, or are limited to the services that you wish to run.
- 6) Also verify individual daemon settings via **Start -> YaST -> System -> Runlevel Editor**. This editor can be used to turn individual system services on and off.

In step #2 if the firewall has not been previously configured, a series of screens will ask you simple questions, and help you configure it. On the first screen, treat the ethernet port used to connect to the network as an external interface, and unless the machine will be a router or firewall there is no internal interface.

Make sure that on the second screen "Protect All Running Services" is checked, and in general forwarding/address translation should not be checked unless there are separate internal and external interfaces, and your machine is designed to be a firewall or router. The logging options defaults ("Log Critical") are a good choice, as are the rest of the defaults on the last two screens.