1. Initial Document

Creation Date: August 21, 2009

2. Revision

Revised By: 2014 Linux Server Security Best Practices Committee

Revision Date: 2-18-2014

Version Number: 1.4

3. Acknowledgments

The Common Security Best Practices portion of this document is based on the Windows Security Best Practices by Brent Williams (bwillia@kennesaw.edu).

The final release document is a collaborative work between the following committee members:

Ryan Beckett Devin Pearson Gil Miralla-Flores

Tyler Hayden Steve Howard

4. Usage

This document contains a set of guidelines, of best practices recommended by the Linux Security Best Practices Committee at Kennesaw State University.

This document is intended to serve as a general guideline and should not be interpreted as policy. Furthermore, the ever-changing nature of information technology prevents this document from being entirely inclusive but should serve as a general baseline for server installation. Please feel free to query the System Administrators Group and ListServe for additional guidance.

Table of Contents

5. Security Best Practices

5.1 Physical Security and Reliability

5.1.1 Walk Through

There is no substitute for physical examination of facilities. A periodic computer room inspection should be performed to assure a proper physical situation. Administrators should verify that there are no problems such as plumbing or cooling leaks, strong smells indicating burning equipment, warning lights, doors ajar, unplugged power cables, etc. Common sense should be a guide.

5.1.2 Limited Access

Servers should always be located in a secured, locked area that ideally is equipped with cardkey access. Where possible, access to critical areas should include a sign-in log for all individuals entering the area. Should an intruder gain access to a server console, it is possible that serious damage or data theft could occur. This could be done directly through the server console or by potentially booting the server with a "Live CD" that allows easy access to server data. Additionally, there are non-malicious events which can occur around server equipment when inexperienced personnel have access to sensitive areas - events such as unintentionally leaving doors insecure, tripping power or other cabling, leaving secure areas cluttered with shipping crates, and others.

5.1.3 Redundant Power

All servers should be operated from redundant power systems (UPS.) This is for protection of the server hardware (offering some isolation from power surges, sags and spikes), and to protect information on server hard drives by allowing the server to be gracefully powered down in an extended power outage. How long the redundant power system can power servers in a power failure is a function of the criticality of the servers. In non-critical, academic settings, a few minutes of redundant power may be all that is needed to smooth over a brief outage. However, for mission critical servers, it may be necessary to provide the capability of 24 by 7 redundant power.

Where generators are present, they should be tested regularly during off-peak time periods, with technical personnel present.

5.1.4 Fire Control

Servers should not be placed in an area covered by a fire suppression sprinkler system, i.e. systems using water as a fire extinguisher. In such circumstances, even a minor fire can result in complete loss of servers and storage devices due to water damage. Instead, server rooms should employ a different fire suppression system.

5.1.5 Climate Control

Server hardware is most reliable when it is operated in a cool environment. If your server room is too warm, seek assistance from campus physical plant operations to determine how the room can be better cooled.

5.1.6 Redundant Servers

Whenever possible, failover servers should be in place in disparate physical locations to prevent downtime in the case of a disaster.

5.1.7 Clustering and Failover

If the operation of your servers is critical, you may want to consider server clustering and/ or server failover. Most server software is capable of both. Ideally, clustered server hardware should be as identical as possible.

5.1.8 Redundant Storage

Server hard drives are heavily used, it is important that some form of redundant storage be provided. It is not unusual to have a single hard drive failure in a file server. Various methods for protecting against such failures are available and should be employed.

5.1.9 Redundant Array of Independent Disks (RAID)

Most servers support redundant storage in two ways. Two or more drives can be installed in a server and drive mirroring or a stripe set with parity can be configured. Optionally, redundant storage hardware can be purchased from many vendors. These redundant storage systems appear to a server as a single drive but do provide continued operation in the event of a single drive failure.

5.1.10 Labeling

All machines should have a correct label, both a physical label and in any associated Keyboard, Video, Mouse (KVM) menu. Many "security" issues relate to unknown machines, or the wrong machine being taken down at the wrong time.

5.2 Disaster Recovery Practices

5.2.1 Backups

It is very important that servers are backed up on a regular basis. The majority of servers at Kennesaw State University are backed up in full each week and incrementally backed up daily. In most environments, it is adequate to perform backups on a weekly basis but it is important to note that each environment is different and has varying levels of criticality which may necessitate a customized backup schedule.

5.2.2 Off -Site Backup Storage

It is very important that one backup set be taken off site on a regular basis. This is to prevent a total loss should the physical facility be lost in a fire or other natural disaster.

5.2.3 Testing a Backup

It is critical that a backup set occasionally be fully restored to a test system. This is to assure that the backups are occurring as they should and that in fact they can be restored when necessary. It is suggested that a backup set restore be tested once per quarter, but once a month might be appropriate in more critical environments.

5.3 Update Practices

5.3.1 Updates

It is very important that updates and patches be applied to servers in a timely manner after testing for backwards compatibility with production software and systems. However, automatic updates may cause server reboots or other outages at inopportune times. Automatic updates on critical servers should not be set for unattended installation. Updates should be installed as part of the install process for any server. It is recommended that updates be tested in a test/development environment before deployment to production.

Most flavors of Linux have both a GUI version and standard command line option for updating. Some users may find that one option works better than another. For example, many users may find Ubuntu updates more complete and fluid using its native "Update Manager" over the command line apt-get or Red Hat users finding yum (YellowDog Updater) more efficient than

Red Hat's GUI method. Some users have found this discrepancy important to ensure the updates are complete and function properly. Whichever tool is used, applying updates via apt-get or yum is generally the best way to ensure that all updates are applied correctly and in a timely fashion.

5.4 Security Related Software

Servers are vulnerable to many forms of attack. It's important that a server be equipped with safeguards to protect the information resident on the system and the system itself.

5.4.1 Antivirus and Antispyware

Use a firewall, virus scanner, and IDS whenever possible. Most modern Linux distributions come with a firewall (see the next section for details). ClamAV (http://www.clamav.net/) is a popular Unix anti-virus toolkit. It is both open source and licensed under the GPL.

Servers are vulnerable to a variety of malicious software through a number of attack vectors. Having the appropriate antivirus and antispyware tools running on a server and ensuring they remain up to date is extremely important to help combat this problem.

Viruses are uncommon on Linux. However, if you are browsing the web on a Linux server - or frequently receiving unknown programs for executing, it is encouraged that you use the Symantec Endpoint Protection Client if is available for your operating system. Please contact iso@kennesaw.edu for more information on the Symantec client.

5.4.2 Firewall

Kennesaw State University employs a campus firewall that protects the campus environment from Internet threats. But the campus firewall does nothing to protect servers from threats originating on campus. For that reason, it is reasonable to employ some form of a software firewall on a server. Most modern Linux distributions come with a built-in firewall, which is generally not active by default. Upon installation, you should activate the firewall, and periodically thereafter, the firewall settings should be reviewed for unnecessary port openings.

5.4.3 Intrusion Detection System

Install an Intrusion Detection System such as AIDE (http://aide.sourceforge.net/) or OSSEC (http://www.ossec.net/) whenever possible to protect your systems from malicious activity or policy violations. It is important to configure your preferred solution to notify you when suspicious events occur.

5.5 Auditing Practices

5.5.1 Logging and Auditing

Configure logging and monitor your logs. Logs should also be sent to a remote, central log server whenever possible. System administrators should use auditing software to monitor logs and send notices to the admin. Several different products are available to accomplish this, such as Logwatch (http://www.logwatch.org/), Syslog-ng (http://www.balabit.com/network-security/syslog-ng/) or tools included with Linux, Solar Winds and many others. Solar winds is available for campus-wide usage.

5.5.2 Log Retention

By default, the most common Linux log is the "syslog", typically in /var/log/ as "messages" or syslog. Retention for systems ranges from just a few days to indefinite. You should periodically review the files in /var/ log for growth and changes due to retention. Application log placement and retention will vary based upon the application. Where possible, the syslog should be set to record off-system. Logs should be retained for as long as possible, taking into consideration regulatory and space concerns.

For additional policy information, please see http://its.kennesaw.edu/infosec/docs.php?id=policy/ServerConfigurationStandard

5.5.3 Auditing Practices

Be sure to review the man pages (manuals) for a system's syslogger (man syslog or man logger). Review /etc/syslog.conf (for most Linux distributions) to assure that the log locations and levels are understood. Typically, server managers would want the auditing system to capture logins, attempted logins, log outs, and administrative activities. Auditing should be limited to gathering just the information that is needed, as it does require CPU and disk time for auditing to gather information.

5.5.4 NTP - Network Time Protocol

In the modern age of clustering and virtualization, centralization of time sources is important for authentication, logging and forensics - should a post-mortem report be required for a security incident. Assure that your host is pointed to a proper time source.

6. Linux Specific Security Best Practices

6.1 Installation

The old maxim was to, whenever possible, install systems while off-network. Most modern Linux distributions install from media, but immediately use the network to apply the full installation and updates. In this case, use discernment about network-based media sources, using known, trusted sources for your distribution.

Only install server components that are required. Do not install ftp or telnet as these are insecure protocols.

Upon initial installation, a port scan should be done against your host to see which services are open, and unnecessary services shut off.

To request DNS, IP, and firewall rules for a new server, fill out the following form:

https://apps.kennesaw.edu/portal/prod/app_uni_sso/login.asp?quickstart=42&quickenvr=prod

A security audit is also required before the server can enter production status. Use this form to request a security scan:

http://its.kennesaw.edu/infosec/docs.php?id=policy/NewServerChecklist

6.2 Authentication and Passwords

Configure all services that permit login access to the system to display a logon warning banner.

For web authentication, such as Webmin or other interfaces, an SSL certificate should be employed before transmitting passwords or other sensitive data.

For command-line access, ssh (Secure Shell) is recommended. Root should not be allowed to access systems from off-host. Rather, named on-system users should use a program such as sudo for elevated rights and tracking of executed commands.

To change the SSH Login Grace Time, edit /etc/ssh/sshd_config and set "LoginGraceTime 20".

- Most modern secure shell implementations disable SSH Protocol 1 by default. Do not use the antiquated version one if it can be avoided. /etc/ssh/sshd_config should indicate "Protocol 2" only.
- Disable remote root login via SSH by adding the following lines to /etc/ssh/sshd_config: PermitRootLogin no

- Whenever possible, disable password authentication and use SSH Public Keys to provide a strong authentication mechanism.

When passwords must be used, consider the following guidelines:

- Root passwords should be changed on a regular basis based upon the criticality of the server in question - and the sensitivity of the data on the system.
- A shadow file or other method of "hiding" the encrypted passwords should be used. This is default on most modern Linux distributions.
- Avoid storing passwords in plain text files on any system.
- A good password contains a mix of upper- and lower-case as well as numbers and "special characters" such as "!@#$%^&*()". Most modern Linux distributions will tell you if you are attempting to set a weak password for command-line accounts.
- Review NIST Guidelines document 800-63 for detailed information on password policies.
- Passwords should require a MINIMUM of 12 characters for security.

6.3 User Accounts

Administrators should have their own named individual accounts. The root account on Linux servers should generally not be accessible from off-system. Indeed, root should be used as little as possible on system. Programs such as "sudo" should be used for the elevation of rights and logging/tracking commands executed by individual accounts. User accounts should be monitored for lack of use over time, and removed if there is a lack of use over time. Syslog and other such "last" logs can be configured to allow last accesses to be logged to distinct log files. Care should be taken when removing administrator accounts - file ownership, cron (scheduler) jobs init (startup) information and account information should be scrutinized before an account is removed.

6.4 Disabling Unused Services

Especially during initial installation, the open ports on Linux servers should be examined for unnecessary services. For example, on servers which do not serve printing functions, the ipp service (CUPS) on port 631 should be disabled. Methods for service disabling vary significantly between services and Linux distributions.

6.5 File Permissions

Confirm that file permissions are properly set for end users to prevent unauthorized access. Outside the default installation, SETUID files should not be used to grant elevated permission. Rather, a program such as sudo should be used. Especially ensure that files containing configuration information and application passwords have the least privileges possible.

6.6 Kernel Parameters

Consider modifying kernel parameters on the system to harden the TCP/IP stack and prevent against certain types of attacks. Common parameters include enabling ExecShield, TCP SynCookies, turning on source IP address verification and logging various types of suspicious packets. The kernel security hardening FAQ from nixCraft (http://www.cyberciti.biz/faq/linux-kernel-etcsysctl-conf-security-hardening/) is a good starting point.

6.7 NFS File Shares

Typical NFS implementations are made up of the daemons nfsd, mountd, statd, and lockd. Always make sure that these packages are up to date if you employ usage of the NFS to share files between systems. By default, request made from root are not allowed on most modern versions of nfsd, but you can check by looking for the line "/home slave1(rw,root_squash)" in /etc/exports. For an in-depth look at the services and documentation on securing NFS as much as possible, look at http://www.tldp.org/HOWTO/NFS-HOWTO/security.html

7. References and Resources

Useful Links:

- http://www.itc.virginia.edu/unixsys/sec/ (similar best-practices guide)
- HowToForge: http://www.howtoforge.com/ (good resource with tutorials on securing a variety of Linux distributions)
- Red Hat Enterprise Linux 4 Security Guide: http://www.redhat.com/docs/manuals/enterprise/RHEL-4-Manual/security-guide/
- Red Hat Enterprise Linux 5 Security Guide from the NSA: http://www.nsa.gov/ia/_files/os/redhat/rhel5-guide-i731.pdf
- Securing Debian Linux: http://www.debian.org/doc/manuals/securing-debian-howto/
- Securing CentOS 5: http://www.centos.org/docs/5/html/Deployment_Guide-en-US/pt-security.html

- Ubuntu Security: https://help.ubuntu.com/community/Security

Further Suggestions: Stay current on the vulnerabilities for your OS or applications:

- http://cve.mitre.org/
- http://www.securityfocus.com
- http://osvdb.org/

Specific programs exist for securing your server:

http://www.cisecurity.org/bench_linux.html ("benchmarking" your security level)

http://www.chkrootkit.org/ (program that will check for rootkits on your Linux system)

http://rkhunter.sourceforge.net/ (alternative to chkrootkit)

http://www.la-samhna.de/samhain/ (host-based intrusion detection system for Linux, Unix, and Windows)