Interested in learning more about securing Linux?

Linux Security Checklist

Linux Security Checklist Linux Security Checklist

Prepared by: Lori Homsher Contributor: Tim Evans

Table of Contents

Introduction This checklist can be used to audit an existing Linux system, or as a system hardening document for Linux administrators tasked with setting up a new Linux system. This checklist does not provide vendor-specific security issues, but attempts to provide a generic listing of security considerations to be used when auditing or configuring a Linux machine.

Security is complex and constantly changing. In addition to this checklist, consult the web site of your Linux distribution and the individual software packages that are loaded onto the system. Most Linux distributions have their own recommendations regarding security. RedHat has documented their recommendations at: http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/security-guide/. Gentoo Linux has a security handbook at: http://www.gentoo.org/doc/en/security/index.xml Debian's security

statement and recommendations can be found at: http://www.debian.org/security/

You should also join or otherwise monitor security-related mailing lists, or RSS feeds, such as those at http://security-focus.com/ and http://www.sans.org.

When implementing system security, there are several fundamental concepts that can go a long way in keeping your system secure. Patch management (keeping software up-to-date) and system hardening (disabling unnecessary services) are vital, but so are overall security policies, change management, and log file audits. A good approach to Linux security is to establish your baseline checklist for secure installation and system hardening, followed by ongoing policy and procedures to ensure your system stays secure.

This document provides steps you can take to minimize your risk when installing a new Linux system. Security is all about risk reduction. The checklist items defined below do not remove your risk of system compromise, but provide you with safety measures that can help reduce your overall chance of compromise.

Checklist

No.No. Security Elements Security Elements 1.1. Boot and Rescue Disk

If you install Linux from a download or over the network, you can create a boot disk manually. The 'mkbootdisk' command is included on most systems. This is the same command that is used during installation to create a boot disk. You must specify a device and a kernel to use.

mkbootdisk --device /dev/fd0 `uname -r` (Note: `uname -r` returns the kernel version.)

Also, have a couple of rescue disks ready. There are many rescue disks available at ftp://metalab.unc.edu/pub/Linux/system/recovery ; Good choices are: Tomsbtrt at: http://www.toms.net/rb and Knoppix at: http://www.knoppix.org (a complete Linux system on CD). You can download or purchase the CD, but make sure you choose the bootable option. 2. System Patches

Most Linux systems work with either rpm (RedHat Package Manager, also used by Mandrake and Suse), apt/dpkg (Debian Package Manager), or YUM (Yellowdog Linux Manager). You can update specific software individually using these commands, or use your vendor's updating tools, if available. RedHat has a very nice managed support option available through RedHat Network that can help you manage many RedHat servers. The managed support option uses the up2date command, which will automatically resolve dependencies. Manual updates from rpm files can be frustrating, since the rpm command simply reports on dependencies – it doesn't resolve them for you. For information on RedHat Network services, visit: https://rhn.redhat.com/rhn/help/quickstart.jsp. RHN services are generally free for the first 90 days after installation, after which you must purchase entitlements to continue.

If you are stuck with an older Linux and you can't upgrade, check out the limited support at the Fedora Legacy Project, http://www.fedoralegacy.org/

For details on the rpm command, type 'man rpm' to view the man pages on the Linux system, or review online help. The Linux Documentation Project has many HOWTOs, including one for RPM at:

http://www.ibiblio.org/pub/Linux/docs/HOWTO/RPM-HOWTO

The Debian package system will resolve any dependency problems, rather than simply report on them (as the rpm system does). For details on the apt command, which is used to load Debian packages, see: http://www.debian.org/doc/manuals/users-guide/ch-iraus.en.html

Regardless of the Linux vendor you've chosen, you'll need some way in which to keep informed of vulnerabilities in the software. There are many mailing lists that will send you vulnerability notices for selected operating system software. Here are just a few: http://www.sans.org/newsletters/ http://www.securityfocus.com/ http://www.cert.org/ 3. Disabling Unnecessary Services

Hardening systems by eliminating unnecessary services can enhance security and improve overall system performance. To begin, you first need to know which services are running on your system. Since services run in various ways, there are several places to check.

ps –ax → will list all currently running processes # ls –l /etc/rc.d/rc3.d/S* → will show all start-up scripts (if you boot into

graphics mode, replace rc3.d with rc5.d) # netstat –a → will list all open ports # chkconfig –list → will show the current startup status of all processes known by chkconfig

Ideally, you should see only those ports that must be open to provide the functionality required by the system.

To disable services, you can remove the startup script, or use a command such as chkconfig. There are two steps to stopping a service: 1) stop the currently running services, and 2) change the configuration so that the services doesn't start on the next reboot.

To stop the running service: # service stop nfs To stop the service at startup time, use the chkconfig command or remove the startup script. To use chkconfig:

/sbin/chkconfig –levels 2345 netfs off To remove the startup script:

/bin/mv /etc/rc.d/rc5.d/S25netfs /etc/rc.d/rc5.d/K25netfs Some services may need to be removed from /etc/inetd.conf or /etc/xinetd.d. This is detailed in the Xinetd section of this document 4. Check for Security on Key Files

• /etc/fstab: make sure the owner & group are set to root.root and the permissions are set to 0644 (-rw-r--r--)

• verify that /etc/passwd, /etc/shadow & /etc/group are all owned by 'root'

• verify that permissions on /etc/passwd & /etc/group are rw-r--r-- (644)

• verify that permissions on /etc/shadow are r-------- (400) 5. Default Password Policy

Ensure the default system password policy matches your organization password policy. These settings are stored in /etc/login.defs and should minimally contain settings for the following. For a complete list of options, see the online man page at:

http://www.tin.org/bin/man.cgi?section=5&topic=login.defs PASS_MAX_DAYS 90 PASS_MIN_DAYS 6 PASS_MIN_LEN 14 PASS_WARN_AGE 7 6. Limit root access using SUDO

Sudo allows an administrator to provide certain users the ability to run some commands as root, while logging all sudo activity. Sudo operates on a per- command basis. The sudoers file controls command access. Your Linux distribution should have specifics on how to configure your distribution. There is help available online as well: http://www.linuxhelp.net/guides/sudo/

7. Only allow root to access CRON

The cron daemon is used to schedule processes. The crontab command is used to create personal crontab entries for users or the root account. To enhance security of the cron scheduler, you can establish the cron.deny and cron.allow files to control use of the crontab. The following commands will establish root as the only user with permission to add cron jobs.

cd /etc/ /bin/rm -f cron.deny at.deny echo root >cron.allow echo root >at.allow /bin/chown root:root cron.allow at.allow /bin/chmod 400 cron.allow at.allow

8. Warning Banners

If your policy requires a warning banner, you can easily create one by copying the appropriate banner message to the following files.

/etc/motd /etc/issue /etc/issue.net add 'GreetString=”Authorized Use Only”' to /etc/X11/xdm/kdmrc and make a similar change to gdm.conf

Here is a sample banner message: “Authorized Use Only. Transactions may be monitored. By continuing past this point, you expressly consent to this monitoring.” 9. Remote Access and SSH Basic Settings

Telnet is not recommended for remote access. Secure Shell (SSH) provides encrypted telnet-like access and is considered a secure alternative to telnet. However, older versions of SSH have vulnerabilities and should not be used. To disable SSH version 1 and enhance the overall security of SSH, consider making the following changes to your sshd_config file:

Protocol 2 PermitRootLogin no PermitEmptyPasswords no Banner /etc/issue IgnoreRhosts yes RhostsAuthentication no RhostsRSAAuthentication no HostbasedAuthentication no LoginGraceTime 1m (or less – default is 2 minutes) SyslogFacility AUTH (provides logging under syslog AUTH)

AllowUser [list of users allowed access] DenyUser [list of system accounts and others not allowed]
MaxStartups 10 (or less – use 1/3 the total number of remote users)

>*Note: MaxStartups refers to the max number of simultaneous unauthenticated connections. This setting can be helpful against a brute- force script that performs forking.*

Some folks also suggest running ssh on an alternate port, although others consider this to be 'security through obscurity'. Regardless of your opinion, it's very easy to change the port that ssh runs on by simply changing the "Port" setting in the sshd_config file, then stopping and restarting ssh. Running ssh on an alternate port will help you avoid port scanners that are looking for open port 22 and the scripted brute-force attempts on this port.

You can block such brute-force ssh attacks with a package like denyhosts (http://denyhosts.sourceforge.net/), which utilizes tcpwrappers (see below). Alternatively, use your iptables firewall (see below) to limit access by IP address or host/domain name.

For additional ssh security, you can configure key forwarding. The following link covers the extra functionality of agent key forwarding within ssh:
http://www.unixwiz.net/techtips/ssh-agent-forwarding.html 10. Host-based Firewall Protection with iptables

Many versions of Linux now come with iptables automatically enabled and configured during installation. RedHat creates /etc/sysconfig/iptables, based on the services you answer as 'allowed' during installation. Here is a basic sample script, created for a server running ssh (port 22), smtp (port 25), squid proxy (port 3128) and samba (netbios port 137). The server's IP is 192.168.1.2 and it is part of a class C network. In the example, we want to accept these services and block all others. If the requested service is not accepted by one of the ACCEPT lines, the packet falls through and is logged and rejected.

Firewall configuration written by redhat-config-securitylevel # Manual customization of this file is not recommended. *filter :INPUT ACCEPT [0:0] :FORWARD ACCEPT [0:0] :OUTPUT ACCEPT [0:0] :RH-Firewall-1-INPUT - [0:0] -A INPUT -j RH-Firewall-1-INPUT -A FORWARD -j RH-Firewall-1-INPUT -A RH-Firewall-1-INPUT -i lo -j ACCEPT -A RH-Firewall-1-INPUT -p icmp --icmp-type any -j ACCEPT -A RH-Firewall-1-INPUT -p 50 -j ACCEPT -A RH-Firewall-1-INPUT -p 51 -j ACCEPT -A RH-Firewall-1-INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT -A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp -- dport 53 -j ACCEPT -A RH-Firewall-1-INPUT -m state --state NEW -m udp -p udp -- dport 53 -j ACCEPT -A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp -- dport 25 -j ACCEPT -A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp -- dport 22 -j ACCEPT -A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp -- dport 3128 -j ACCEPT

-A RH-Firewall-1-INPUT -s 192.168.1.0/24 -d 192.168.1.2 --dport 137 -j ACCEPT -A RH-Firewall-1-INPUT -s 192.168.1.2 -d 192.168.1.255 -j ACCEPT -A RH-Firewall-1-INPUT -d 255.255.255.255 -j DROP -A RH-Firewall-1-INPUT -d 192.168.1.255 -j DROP -A RH-Firewall-1-INPUT -j LOG -A RH-Firewall-1-INPUT -j REJECT --reject-with icmp-host- prohibited COMMIT 11. Xinetd and inetd.conf

If running the older /etc/inetd.conf file, be sure to disable unnecessary services by removing them (or commenting them out) from the inetd.conf file. For example, to remove telnet access, remove the following line: telnet stream tcp nowait root /usr/sbin/telnetd telnetd -a

On systems running scripts from the xinetd.d directory, disable the services by changing the script from 'disable = no' to 'disable = yes'. A sample xinetd.d script and various ACL settings are included in the tcpwrappers section.

You will need to send a HUP signal to the inetd process after modifying the configuration files (kill -HUP processID) 12. tcpwrappers

TCP Wrappers allows control of services based on hostname and IP addresses. Additionally this tool contains logging and use administration. Tcpwrappers is a daemon that positions itself between detailed inquiries and the requested service, and checks the requestor's IP against the hosts.allow and hosts.deny files.

In the traditional inetd.conf file, you can run tcpwrappers by calling tcpd (the tcpwrappers daemon) as follows: # first comment out the original line: #telnet stream tcp nowait root /usr/sbin/telnetd telnetd –a # then replace it with the modified line: telnet stream tcp nowait root /usr/sbin/tcpd telnetd -a

Standard Linuxes don't have tcpwrappers built into xinetd, since xinetd already includes logging and access control features. However, if you want to add this further control you can re-compile xinetd with libwrap support by passing '-- with-libwrap' as an option to the configure script. When xinetd is compiled with libwrap support, all services can use the /etc/hosts.allow and /etc/hosts.deny access control. xinetd can also be configured to use tcpd in the traditional inetd style. This requires the use of the NAMEINARGS flag and the real daemon name must be passed in as server_args. Here is an example for using telnet with tcpd: service telnet {

flags = REUSE NAMEINARGS protocol = tcp socket_type = stream wait = no user = telnetd server = /usr/sbin/tcpd server_args = /usr/sbin/in.telnetd }

To use settings within xinetd scripts to control access by IP for specific services, simply change the appropriate xinetd scripts, for example:

service imap {

socket_type = stream protocol = tcp wait = no user = root only_from = 198.72.5.0 localhost banner = /usr/local/etc/deny_banner server = /usr/local/sbin/imapd }

Here are some other helpful settings:

To deny certain IPs or domains:

no_access = 10.0.5.12 bad.domain.com

To specify limits on connections – total number of ssh connections:

instances = 10 Maximum number of connections per IP address:

per_source = 3

To specify allowed access times:

access_times = 8:00-17:00

13. System Logging

All Linux systems support system logging, which is important for troubleshooting system and network problems, as well as possible security incidents. Syslog is the daemon that controls logging on Linux systems. Logging configuration is stored in /etc/syslog.conf. This file identifies the level of logging and the location of the log files. Log files should be owned by root user and group, so that they are not available to the casual user.

It is recommended that log entries be logged to a centralized log server, preferably over ssh for data confidentiality. Centralized logging protects from deletion of log files and provides another layer in the event the log files are tampered with. This is easily accomplished as follows: # send to syslog server

.emerg;.info;*.err @hostname

For more information on syslog.conf settings, view the man page by typing 'man syslog.conf'.

Next Generation syslog is more customizable than syslog and supports digital signatures to prevent log tampering. It is available at: http://freshmeat.net/projects/syslog-ng/

Auditing your log files: Regardless of the software used to create the log files, good security includes the ongoing review of log file entries. This can become very tedious if your only tool is to manually read the logs. Fortunately, there are some very good open- source packages to help:

Logwatch: comes standard with many Linux distributions. Configuration of logwatch is done in the /etc/log.d directory. The script logwatch.conf allows you to set defaults, such as the level of detail, the services to include, and the log file names. Reports can be sent directly to your email and include data such as: firewall rejects, ftp uploads/downloads, disk space usage, sendmail statistics, etc.

Swatch: is an active log file-monitoring tool. Swatch uses regular expressions to find lines of interest. Once swatch finds a line that matches a pattern, it takes an action, such as printing it to the screen, emailing it, or taking a user-defined action. To use swatch to check logs normally, run:

swatch --config-file=/etc/swatch.conf -- examine=/var/log/messages To use swatch as a constantly running service that scans lines of a log file as they come in, run:

swatch --config-file=/etc/swatch.conf --tail- file=/var/log/messages

Don't forget email security when sending your log files via email, which flows in plain text from source to destination mailbox. You may want to encyrpt the logfiles with something like GnuPG before sending them. Visit: www.gnupg.org for more information.

There are dozens of other tools available to analyze and audit syslog messages. The important point to remember is to pick a tool and make sure someone is responsible for log file auditing on a regular basis. 14. Backups

There are many non-commercial and commercial backup programs available for Linux. We'll highlight the non-commercial tools here. A google search for 'linux backup software' should provide you with enough commercial options to choose from.

tar, gzip, bzip2: these tools have been around a long time and they are still a viable option for many people. Almost any *nix system will contain tar and gzip, so they will rarely require special installation or configuration. However, backing up large amounts of data across a network may be slow using these tools.

To backup a list of directories into a single tar archive, simply run the tar command to create the tarball, followed by the gzip command to compress it: tar -cvf archive-name.tar dir1 dir2 dir3.... gzip -9 archive-name.tar You may prefer to use bzip2, which is a bit better then gzip at compressing text, but it is quite a bit slower.

You can combine the tar and gzip actions in one command by using tar's -z option.

Rsync: rsync is an ideal way to move data between servers. It is very efficient for maintaining large directory trees in synch (not real time), and is relatively easy to configure and secure. rsync does not encrypt the data however so you should use something like SSH or IPSec if the data is sensitive (SSH is easiest, simply use "-e ssh"). Rsync (by Martin Pool) is available at: http://freshmeat.net/projects/rsync/

Amanda: is a client-server based network backup program with support for *nix and Windows (via samba). It is available from http://www.amanda.org

dump: is written specifically for backups. It backs up the entire file system and allows multiple levels of backups. The corresponding 'restore' command allows for restore from a dump backup. For example, to backup /boot file system to backup.boot: dump 0zf backup.boot /boot See 'man dump' for a complete list of options. 15. Integrity-checking Software

Integrity checking/assurance software monitors the reliability of critical files by checking them at regular intervals and notifying the system administrator of any changes. This type of software is very useful in identifying unauthorized changes to configuration files, log files, services, as well as identifying the presence of Trojans, rootkits, and other malicious code.

There are several integrity-checking packages available. Most Linux distros come with a barebones version of a commercial package. Commercial Tripwire support is available (for a fee) and can include an excellent management console to provide central control for recreating your policy files and databases. Aide is an advanced Intrusion Detection system that aims to be a free replacement to Tripwire. Samhain is another open-source option.

http://tripwire.org/ http://sourceforge.net/projects/aide http://sourceforge.net/projects/samhain 16. Apache Security (all *nix)

There are entire books dedicated to apache security. We will hit some of the high-level suggestions here. Detailed help can be found at http://httpd.apache.org/

First, verify that your apache subdirectories are all owned by root and have a mod of 755: [user@host xinetd.d]$ ls -l /etc/apache drwxr-xr-x 7 root root 4096 Aug 23 10:24 conf drwxr-xr-x 2 root root 4096 Aug 27 08:44 logs (your Apache installation may be located at /usr/local/apache or elsewhere if you installed it yourself)

[user@host xinetd.d]$ ls –l /usr/sbin/*http* -rwxr-xr-x 1 root root 259488 Aug 2 05:22 /usr/sbin/httpd -rwxr-xr-x 1 root root 270248 Aug 2 05:22 /usr/sbin/httpd.worker

Likewise, your httpd binary should be owned by root, with a mod of 511. You can create a web documents subdirectory outside the normal Apache filetree as your DocumentRoot (/var/www/html in RedHat), which is modifiable by other users -- since root never executes any files out of there, and shouldn't be creating files in there.

Server side includes (SSI) create additional risks, since SSI-enabled files can execute any CGI script or program under the permissions of the user and group apache runs as (as configured in httpd.conf). To disable the ability to run scripts and programs from SSI pages, replace "Includes" with "IncludesNOEXEC" in the options directive. Users may still use <--#include virtual="..." --> to execute

CGI scripts if these scripts are in directories designated by a ScriptAlias directive.

Script Aliased CGI: is recommended over non-script aliased CGI. Limiting CGI to special directories gives the administrator control over which scripts can be run.

System Settings: To prevent users from setting up .htaccess files that can override security features, change the server configuration file to include: <Directory />

   AllowOverride None </Directory>

To prevent users from accessing the entire filesystem (starting with the root directory), add the following to your server configuration file: <Directory />

                                    Order Deny,Allow Deny from all </Directory>

To provide access into individual directories, add the following:

<Directory /usr/users/*/public_html>

Order Deny,Allow Allow from all </Directory> <Directory /usr/local/httpd>

                                 Order Deny,Allow Allow from all </Directory>

If you are using Apache 1.3 or above, apache recommends that you include the following line in your server configuration files: UserDir disabled root 17. Apache Mod_security module

The mod_security module runs on most versions of Apache, but you will most likely be required to install it from source (check with your Linux distribution). You can download the latest source code from www.modsecurity.org and compile it using apxs or apxs2. Detailed instructions can be found in the ModSecurity User Guide or the source code's INSTALL file.

Mod_security allows you to enhance the overall security of your apache web server by providing additional configuration settings within your httpd.conf file. These settings allow you to filter/inspect all traffic, or filter/inspect non-static traffic only (DynamicOnly). You can then set the default action for matching requests – for example, displaying a standard error page. In addition, you can specify allowable ASCII values and set restrictions for file uploads. Mod_security also provides much more logging than the default for apache. More information can be found at www.modsecurity.org 18. Xwindow

X window can be a large security risk considering the many exploits for the product and since its data flows unencrypted across networks. A good method of configuring access to X servers is to tunnel X window sessions through SSH (secure shell). This is referred to as X11 forwarding. SSH provides the advantage of adding encryption to tunneled X sessions. A document from Stanford University provides a security check to test existing X servers and

describes the steps involved to connect to an X server via SSH:
http://www.stanford.edu/services/securecomputing/x-window/

19. LIDS (Linux Intrusion Detection System)

LIDS is an enhancement for the Linux kernel written by Xie Huagang and Philippe Biondi. It implements several security features that are not in the Linux kernel natively. Some of these include: mandatory access controls (MAC), a port scan detector, file protection (even from root), and process protection. LIDS implements access control lists (ACLs) that will help prevent even those with access to the root account from wreaking havoc on a system. These ACLs allow LIDS to protect files as well as processes.

For more information on LIDS: http://www.lids.org/ 20. Selinux (Security Enhanced Linux)

Developed by the U.S. National Security Agency (NSA), Security-enhanced Linux is a research prototype of the Linux® kernel and a number of utilities with enhanced security functionality designed simply to demonstrate the value of mandatory access controls to the Linux community and how such controls could be added to Linux.

The Security-enhanced Linux kernel enforces mandatory access control policies that confine user programs and system servers to the minimum amount of privilege they require to do their jobs. When confined in this way, the ability of these user programs and system daemons to cause harm when compromised (via buffer overflows or misconfigurations, for example) is reduced or eliminated. This confinement mechanism operates independently of the traditional Linux access control mechanisms. It has no concept of a "root" super-user, and does not share the well-known shortcomings of the traditional Linux security mechanisms (such as a dependence on setuid/setgid binaries).

Implementing SE Linux can have unexpected effects on a system, and you may find standard daemons won't run properly, or at all, or logfiles may not be writable, or other similar effects that require detailed configuration of SE Linux

Currently, RedHat Enterprise Linux Version 4 includes an implementation of SE Linux.

For more information on SE Linux: http://www.nsa.gov/selinuX/ 21. Email Security

Many sys-admins disable the sendmail utility on user workstations, and centralize its service on a main mailserver machine. Even in this situation, there's more you can do to increase its security.

For sendmail, follow the recommended security settings for secure installation: http://www.sendmail.org/security/secure-install.html. It is possible to configure sendmail to launch when needed, rather than run it as a listening daemon on port 25.

Postfix is a good alternative to sendmail. Information is available at: http://www.postfix.org/ 22. File Sharing

There are many methods of file sharing among Linux systems. Opening up a system for file sharing may not be acceptable within your organizational policy.

We provide information here for those who require this type of access.

For information on NFS security (for sharing *nix-to-*nix), see: http://www.linuxsecurity.com/content/view/117705/49/

Samba is a software package that offers file sharing between Linux and Windows systems. It can be configured to use encrypted password access, restriction by user and/or IP address, and file-level permissions can be set. Samba is available at: www.samba.org. A good article describing the various Samba security modes is available at: http://www.redhat.com/docs/manuals/enterprise/RHEL-4-Manual/ref-guide/s1-samba-security-modes.html 23. Encryption

If the system will be storing confidential data and you need to minimize the risk of data exposure, encryption may be an acceptable solution. Sourceforge has a web page that attempts to provide a disk encryption HOWTO for Linux users. It is available here: http://encryptionhowto.sourceforge.net/Encryption-HOWTO.html

Depending on your needs, openPGP and/or GnuPG may be appropriate. These tools will allow you to encrypt emails and attachments, as well as files stored on disk. GnuPG is available at: www.gnupg.org and OpenPGP can be found at: www.openpgp.org.

24. Anti-Virus Protection

There are several anti-virus options available for Linux users and the list continues to grow. Here are a few: Clamav: www.clamav.net f-prot: www.f-prot.com/products/corporate_users/unix/ Vexira: www.centralcommand.com/linux_server.html

25. Bastille Linux

A hardening program for RedHat, SUSE, Debian, Gentoo, and Mandrake distributions, Bastille Linux attempts to lock down a Linux server. It walks the user through a series of questions and builds a policy based on the answers. Bastille Linux was conceived by a group of SANS conference attendees and is available at: http://www.bastille-linux.org/

References:

Documentation resource. Debian Security Information, http://www.debian.org/security/

Documentation resource. Ibiblio Linux Archive, http://www.ibiblio.org/pub/Linux/

Documentation resource. Encryption HOWTO. http://encryptionhowto.sourceforge.net/Encryption-HOWTO.html

Documentation resource. Linux Documentation Project, http://tldp.org/

Documentation resource. LinuxHelp.net, http://www.linuxhelp.net/guides/

Documentation resource. Linux Security general information. http://www.linuxsecurity.com/

Documentation resource. Apache Server Project. http://httpd.apache.org/docs/1.3/misc/security_tips.html

Friedl, Steve (February 22, 2006). An Illustrated Guide to SSH Agent Forwarding. Retrieved August, 2006 from http://www.unixwiz.net/techtips/ssh-agent- forwarding.html

Holbrook, John (2004). Step by step installation of a secure Linux web, DNS and mail server. Retrieved August, 2006 from http://www.sans.org/reading_room/whitepapers/linux/1372.php

McCarty, Bill. (2003). Red Hat Linux Firewalls. RedHat Press.

Nielsen, Kim (2006). Gentoo Security Handbook. Retrieved August, 2006 from http://www.gentoo.org/doc/en/security/security-handbook.xml

Stanford University (2004). X Window Security. Retrieved August, 2006 from http://www.stanford.edu/services/securecomputing/x-window/

Wainwright, Peter. (1999). Professional Apache. Wrox Press.

RedHat (2002). RedHat Security Guide. Retrieved August, 2006 from: http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/security-guide/

RedHat (2005). RedHat Enterprise Linux 4 Reference Guide Retrieved August, 2006 from: http://www.redhat.com/docs/manuals/enterprise/RHEL-4-Manual/ref- guide/index.html

Rosenthal, Chip & Haugh, Julianne Frances. Online man pages for login.defs. Retrieved August, 2006 from http://www.tin.org/bin/man.cgi?section=5&topic=login.defs

Ziegler, Robert. (2002). Linux Firewalls. New Riders Publishing.

SANS San Antonio 2013 San Antonio, TXUS Dec 03, 2013 - Dec 08, 2013 Live Event

SEC480 Beta - Canberra, Australia Canberra, AU Dec 11, 2013 - Dec 13, 2013 Live Event

SANS Cyber Defense Initiative 2013 Washington, DCUS Dec 12, 2013 - Dec 19, 2013 Live Event

SANS Oman 2013 Muscat, OM Dec 14, 2013 - Dec 19, 2013 Live Event

SANS Golden Gate 2013 San Francisco, CAUS Dec 16, 2013 - Dec 21, 2013 Live Event

FOR572 Advanced Network Forensics San Antonio, TXUS Jan 05, 2014 - Jan 10, 2014 Live Event

FOR585 Adv Smartphone and Mobile Device Forensics San Antonio, TXUS Jan 13, 2014 - Jan 18, 2014 Live Event

SANS Security East 2014 New Orleans, LAUS Jan 20, 2014 - Jan 25, 2014 Live Event

SANS Dubai 2014 Dubai, AE Jan 25, 2014 - Jan 30, 2014 Live Event

AppSec 2014 Austin, TXUS Feb 03, 2014 - Feb 08, 2014 Live Event

SANS Cyber Threat Intelligence Summit Washington, DCUS Feb 04, 2014 - Feb 11, 2014 Live Event

Secure India@Bangalore 2014 Bangalore, IN Feb 10, 2014 - Feb 22, 2014 Live Event

SANS Scottsdale 2014 Scottsdale, AZUS Feb 17, 2014 - Feb 22, 2014 Live Event

FOR508 Tokyo - February 2014 Tokyo, JP Feb 17, 2014 - Feb 22, 2014 Live Event

SANS Brussels 2014 Brussels, BE Feb 17, 2014 - Feb 22, 2014 Live Event

RSA Conference 2014 San Francisco, CAUS Feb 23, 2014 - Feb 24, 2014 Live Event

Asia Pacific ICS Security Summit & Training OnlineSG Dec 02, 2013 - Dec 08, 2013 Live Event

SANS OnDemand Books & MP3s OnlyUS Anytime Self Paced

Last Updated: November 28th, 2013

Upcoming SANS Training Click Here for a full list of all Upcoming SANS Events by Location