Linux Security Checklist

Date Modified	By	Description
12/09/2002	Lubomir Nistor	File Creation
13/09/2002	Lubomir Nistor	Updated issues
18/09/2002	Lubomir Nistor	Modification of some commands

Contents

The following is a recommended security checklist for Linux servers. This document should be used as a guide to the installation and configuration of Linux Servers in conjunction with an agreed security plan for the identified systems. The document is designed for use by experienced system administrators.
Some of the settings may be dependant on the patch levels of the components in use, and therefore differencies may exist between this document and the actual file paths and access control settings on your machine.
Most of the points below can be addressed by running security scripts made specifically for every system (fx. Harden_suse), but due to the general nature of these scripts or applications it is not advised to use them without proper testing.

1. Initial Installation

1.1 Install the Latest Patches

In most cases distribution vendors will provide an update facility for the distribution of patches. The latest system patches should be installed prior to operational deployment. Particular attention should be paid to those network services that the operating system makes available to remote clients (eg: Web (Apache), Mail (sendmail/postfix/imapd), and so on.
It is also recommended that the system be updated with newly realeased patches as soon as operational circumstances allow.
Bypassing the vendor, and installing patches directly from the application provider (eg: from apache.org) may also be appropriate in some circumstances, where the problem in question is significant, or the distribution vendor response to security issues is poor.
Latest Patches can be found at

- Debian ftp://ftp.debian.org/debian/dists/stable-proposed-updates/
- RedHat ftp://ftp.redhat.com/pub/redhat/linux/updates
- SuSe ftp://ftp.suse.com/pub/suse/i386/update/

In order to stay updated with latest vulnerabilities on Solaris systems and patches required for it Sun issues a security bulletin.To receive security bulletins directly from:

- Debian http://www.debian.org/MailingLists/subscribe#debian-security-announce
- RedHat https://listman.redhat.com/mailman/listinfo/redhat-watch-list
- SuSe http://www.suse.com/us/private/support/mailinglists/index.html

1.2 File Systems

Per default Linux mounts remote or local filesystems are mounted with read-write privileges with possibility to have suid or sgid files.
In order to prevent that filesystems that don’t require extra privileges should be limited.
In */etc/fstab* there should be an entry *nosuid* or *noexec* for external devices like cdrom or filesystems in that specific row in the fourth field .

1.3 Time Settings

All the servers should have the same time settings in order to be able to evaluate logs properly.

1. There should be a time-zone entry in /etc/sysconfig/clock containing *ZONE="Europe/Berlin".* Or in Debian */etc/timezone* should contain *Europe/Berlin*
2. There should be a NTP system installed with timeservers configured for synchronisation (fx. /etc/ntp.conf should contain server a.b.c.d prefer)

- Timeservers in OssBss are 10.130.200.70 or 10.130.200.80
  - In Management network 10.10.8.70 or 10.10.8.80
  - In internet network there are official time servers at http://www.eecis.udel.edu/~mills/ntp/clock1.htm

1.4 Software Selection

If system should be freshly installed, there should be core installation used and only those packages added that are required for operation of the system.
All the external packages that can't be patched should be kept updated to the latest operational version (fx. SSH package should be version 3.4.1 or higher).

All the unnecessary modules should be also removed.

1.5 Minimize boot services or daemons

All the unnecessary daemons or services starting at boot time (*/etc/rc*.d*) should be removed or disabled. They can also be listed with *chkconfig –list* on all systems except Debian.

FX.A service can be disabled with *chkconfig –level 3 lpd off*
or just removed from */etc/rc.d/rc3.d/S12lpd*

1.6 Message Text for users attempting to log on

/etc/motd

Place the following message (or a similar one) into this file. It contains a message that will be printed after a successful login.

This is a private computer facility. Access for any reason must be specifically authorized by the owner. Unless you are so authorized, your continued access and any other use may expose you to criminal and/or civil proceedings. Usage may be monitored.

/etc/issue

Place the following message (or a similar one) into this file. It contains a message that will be printed during the login process.

This is a private computer facility. Access for any reason must be specifically authorized by the owner. Unless you are so authorized, your continued access and any other use may expose you to criminal and/or civil proceedings. Usage may be monitored.

/etc/issue.net

Place the following message (or a similar one) into this file. It contains a message that will be printed during the login process.

This is a private computer facility. Access for any reason must be specifically authorized by the owner. Unless you are so authorized,

> *your continued access and any other use may expose you to criminal and/or civil proceedings. Usage may be monitored.*

NOTE: The users may see both the /etc/motd and the /etc/issue messages when they login.

SSH daemon should be configured to display the message by putting this line into sshd_config: *Printmotd yes*

1.7 Privileged Account Login Source

In order to ensure security of the root account there should be limitations placed on the source of login.
Root should be able to log into the system only locally (via console or with su command).
This can be ensured by :

1. In */etc/nologin* there should be all the administrative accounts
2. In */etc/security/access.conf* there should be a line
   -:ALL EXCEPT wheel shutdown sync:console
   -:ALL EXCEPT root:ALL EXCEPT console
3. In *sshd_conf* put line *PermitRootLogin no*

1.8 Network driver configuration

Make the following adjustments to the */etc/sysctl.conf* to protect the machine from some types of network attacks.

1. `net.ipv4.ip_forward = 0`
2. `net.ipv4.conf.all.accept_source_route = 0`
3. `net.ipv4.tcp_max_syn_backlog = 4096`
4. `net.ipv4.conf.all.rp_filter = 1`
5. `net.ipv4.conf.all.send_redirects = 0`
6. `net.ipv4.conf.all.accept_redirects = 0`
7. `net.ipv4.conf.default.accept_redirects = 0`

and protect the configuration file:

- `chown root:root /etc/sysctl.conf`
- `chmod 0600 /etc/sysctl.conf`

Disable multicasting:

ifconfig [interface] -allmulti -multicast

2. *System Network Services*

2.1 Network Services Summary

All the unnecessary network services should be switched off.

1. */etc/inetd.conf* should not contain any entries unless specifically required by applications.

Here is a quick rundown of the risks associated with services started in */etc/inetd.conf*:

ftp: enables an FTP server that introduces a variety of insecurities and is the cause of many intrusions. Disable this and use SSH instead to transfer files between systems.
telnet, shell, login, exec: allows users from other systems to log into and run commands on your machine. This is useful, but the more useful something is, the more likely it is that someone will find a way to exploit it. Disable these services and, if you do need to allow remote logins, use SSH instead.
comsat: a daemon which is used to notify users of newly arrived email. There are alternate means of doing the same thing, and there are occasional rumors of security problems with comsat. Unless you have some overwhelming need for this, turn it off.
talk: allows users to communicate by typing at each others' terminals.
uucp: Nobody uses uucp anymore - disable this. While you are at it, you may as well turn off execute permission on the uucp-related shell commands.
tftp: FTP without any security. This should be needed only if your system will be used for booting workstations. If this is the case, you must invoke the daemon with the -s flag, as in:

tftp dgram udp wait root in.tftpd -s /tftpboot

If you don't, tftp can be used to retrieve any file from your system, anonymously. Also make all the files in the bootfile directory read-only. Finally, restrict access to the service using TCPwrappers and IPFilter/IPChains.
finger: this gives out information on who is loggedin, or people's phone numbers and offices. Unfortunately this information can be used by a potential intruder to find accounts to attack. You may wish to disable this, run a custom finger daemon, or restrict access to it using TCPwrappers and IPFilter/IPChains.
systat, netstat: these services give out information about your system. The comments for finger apply to these.
time: Gives out the system time to any remote host that asks for it. Probably safe but can be disabled without impacting the system.
echo, discard, daytime, chargen: these are used for testing, and are generally safe, though there have been reports of TCP packets with forged IP source addresses being used to trick a system into sending echo packets to itself, causing a packet storm on the local ethernet segment. Disable them and only turn them on while testing.
rexd - this is the Remote Procedure Call mechanism. It has minimal authentication, so disable it and use SSH instead.
walld: allows people to send messages to all logged in users. Useful, but easily abused.

ttdbserverd (tooltalk): used by some convenient desktop elements but not important from a system operation standpoint. Some versions of this service contain serious remote exploits and should be disabled (dsabling this service causes virtually no operational degradation).
rpc.cmsd (calendar manager): used to share calendar information over the network but not important from a system operation standpoint. Some versions of this service contain serious remote exploits and should be disabled.
others: Other services such as sadmind (once found to be vulnerabale to remote root exploit) and kerberos can be disabled without impacting the system.

2.There should not be any services listening on the network unless required by applications.

Fx. XFree86 listens on port 6000+*n*, where *n* is the display number. This connection type can be disabled with the **-nolisten** option (see the Xserver(1) man page for details).

2.2 File Transfer

Ftp service should be disabled and secure ftp or secure copy should be used.

/etc/ftpusers should contain all the account names except those that should be allowed to access the system via FTP.

2.3 Electronic Mail

There should be no email service running on the system for local use (email servers have email agents installed as application and should follow the application security part of this document).

2.4 Domain Name Service

There should be no DNS servers running on the system (DNS servers should be treated as an application and should follow application security part of this document).

2.5 Remote shell / copy services

All the systems should have the latest SSH installed in order to allow remote administration of the server with encrypted interconnection.
Sshd_config should contain also these features:

```
Protocol 2
UsePrivilegeSeparation yes
ServerKeyBits 768
SyslogFacility AUTH
LogLevel INFO
LoginGraceTime 600
PermitRootLogin no
StrictModes yes
RSAAuthentication yes
PubkeyAuthentication yes
RhostsAuthentication no
```

```
IgnoreRhosts yes
RhostsRSAAuthentication no
HostbasedAuthentication no
PermitEmptyPasswords no
PasswordAuthentication yes
PrintMotd yes
PrintLastLog no
MaxStartups 10:30:60
ReverseMappingCheck yes
```

2.6 File Share Facilities

There should be no file sharing facilities (like nfs or cifs), unless required by the application .

2.7 Firewall Selection

Every system should have a filtering capability at disposal for later use. Filtering capabilities should enable limitation of certain IP addresses to certain services.

1.IPTables http://www.iptables.org/

2.IPChains http://www.netfilter.org/ipchains/

3. System Accounts and User Rights

3.1 Account Characteristics

By default, Linux operates on the assumption that all users are local users. Specific package installation needs to be done on most Linux distributions in order to facilitate a distributed authentication framework such as LDAP.
Other features:

1. User home directories should be mode 755 or more restrictive
2. No user dot-files should be group/world writable
3. Standard Password policy should be put in */etc/login.defs*

3.2 Standard Accounts

Several of the accounts in */etc/passwd* are unnecessary. In order to secure them you should:

1. effectively disable them.
2. ensure these accounts cannot use **ftp**, **cron** or **at**.
3. remove valid shells from daemon accounts.
4. Put them into */etc/nologin*

3.3 Unauthenticated Access

There should be no possible unauthenticated access enabled on the system.

1. */etc/hosts.equiv, /root/.rhosts, /etc/ssh/shosts /root/.netrc* should be empty
2. No empty password fields in password files

3.4 Appropriate Administrative Authentication

Access to root account via su should be only possible from wheel group. All users that are system administrators should have separate accounts and be in wheel group and only they can login to the console. All this can be configured in */etc/security/access.conf*

3.5 Authentication Configuration

In order to enforce system authentication to use standard unix authentication facility */etc/pam.d* should contain these entries:

```
# PAM configuration
# Authentication management
login   auth required   /usr/lib/security/pam_unix.so.1
su   auth required   /usr/lib/security/pam_unix.so.1
other   auth required   /usr/lib/security/pam_unix.so.1

# Account management
login   account required       /usr/lib/security/pam_unix.so.1
su account required               /usr/lib/security/pam_unix.so.1
other   account required       /usr/lib/security/pam_unix.so.1
```

```
# Session management
su   session required        /usr/lib/security/pam_unix.so.1
other   session required        /usr/lib/security/pam_unix.so.1

# Password management
other   password required       /usr/lib/security/pam_unix.so.1
```

If http authentication is supposed to be used then there should be extra entries specifying http authentication facility.

4. File and Object Access

4.1 Umask settings

Set user file creation mask

In each of the files */etc/csh.cshrc* and /*etc/profile*, there should be an invocation of the umask command. This invocation should be positioned immediately after the initial comments. The value passed to umask is an octal mask of the mode bits that are *not* set when a file is created. Acceptable values are 022, 026 (suggested) and 027. Each of these has advantages and disadvantages. Please read the umask manual page prior to selecting the value to be set.

Set FTP file creation mask

Add the following line at the end of the /etc/proftpd.conf file. This line contains the default umask value that will be used by FTP when a file is created.

```
UMASK=022
```

Set daemon umask *umask 022*

In /etc/init.d/functions add a line *umask 022* (redhat)
In /etc/rc.status add a line *umask 022* (others)

4.2 Permissions tightening

1.Minimize file or object access to only groups or users that will access them (fx. Oracle daemon should be executable by user oracle only)

2.Crontab access restrictions should be put into */etc/cron.allow* (debian, redhat) or */var/spool/cron/allow* (SuSe).

3.At access retrictions should be put into */etc/at.allow*.

4.3 SUID or SGID files.

Check for setuid files, and modify them, as appropriate. The command to check for these files is:

```
find / -perm -04000 -type f -exec ls -ld {} \;
```

```
-rwsr-x---     1 root  trusted   /bin/ping
-rwsr-x---     1 root  trusted   /bin/su
-rwsr-x---     1 root  trusted   /usr/bin/crontab
-rwsr-xr-x     1 man  root       /usr/bin/man
-rwsr-xr-x     1 root  root      /usr/bin/rcp
-rwsr-xr-x     1 root  root      /usr/bin/rlogin
-rwsr-xr-x     1 root  root      /usr/bin/rsh
-rwsr-x---     1 root  shadow    /usr/bin/gpasswd
-rwsr-x---     1 root  trusted   /usr/bin/newgrp
-rwsr-xr-x     1 root  shadow    /usr/bin/passwd
-rwsr-x---     1 root  trusted   /usr/bin/sudo
-rwsr-xr-x     1 root  root      /usr/sbin/traceroute
-rwsr-xr-x     1 root  root      /usr/lib/pt_chown
```

Check setgid files

Check for setgid files, and modify them, as appropriate. The command to check for these files is:

```
find / -local -type f -perm -2000 -exec ls -ld {} \;

-rwxr-sr-x      1 root  tty      /usr/bin/write
-rwxr-sr-x      1 root  tty      /usr/bin/wall
```

NOTE: A server should be checked for setgid files after patches are updated, and after third-party packages (source or binary) are installed. A list of all the SUID and SGID files should be maintained from the point of clean installation in order to detect deviations from correct system state.

4.4 System access configuration files locked

In every user’s directory there should be configuration files created (*.rhosts* or *authorized_keys*) with root as the owner and writable only by root in order to keep the control of the users system access.

5. System Auditing

5.1 Auditing Overview

All the messages and log information should be centrally processed on a remote log server.
In OSSBSS there is a syslog server at disposal. *Syslog.conf* entry should look like this:

```
        *.*                          @10.130.200.40
```

or

```
        *.*                          @10.10.8.40
```

5.2 Initial Installation

Syslog messages sent to a centralized log server

1. */etc/syslog*.conf should contain this entry **.* @logserver*

Turn on `cron` logging

2. */etc/default/*cron should contain *CRONLOG=YES*

5.3 Protecting the audit configuration files

Integrity verification service should be done to a remote host.
Logs should be stored safely on a read only media or on a secured media not accessible by all the system users.

5.4 Monitoring User access.

Create a log for authentication information. It should contain all the necessary authentication information for access auditing.

```
echo "auth.info\t\t\t/var/log/authlog" >>/etc/syslog.conf
touch /var/log/authlog
chown root:root /var/log/authlog
chmod 600 /var/log/authlog
```

6. Application security

6.1 Patches

Install all the latest patches and fixes for the application. If possible upgrade application to the latest version.

6.2 Minimize services offered

Switch off or remove services that are not required to perform application's function in the system.

Fx. Apache web server with basic HTML functionality required doesn't need CGI, imap module or Java servlets configured.

6.3 Configure users and authentication

Every user should have his/her own login with proper authentication method used. Their permissions should not include more than necessary (read only user should have only read only access) and if necessary for every role a user has there should be an account with proper permissions for that role.

Fx. User XY ; role: application tester (read only access to data)
User XY ; role application developer (read write to development environment)
User XY ; role application administrator (full access to application configuration)

6.4 Auditing

If possible application should produce a log documenting its tasks that are performed as well as requests, queries or user input.

1. major or critical application errors
2. users login/logout
3. modification of application settings (security settings, logging settings, operational settings)
4. (optional) business data modifications

6.5 Object access, permissions

If possible application should have restrictions on objects (files, items or tables) so that only authorized sources can read, modify or delete them.

Fx. Access to SNMP system IDs in NetCool database should be limited to NetCool administrator only.
Oracle progressor database access granted only to progressor application.