# Linux Hardening Checklist

This checklist will act as an introduction to Linux hardening. Not all commands may work as they are printed. This will require additional research and adaptation. Along with each command will follow a brief explanation of what the command is for and some will include examples of when you should use them as well as when not to.

Any word that is in italics is to be replaced. It is a place holder. The word sudo in front of a command represents that you need a sudoers permissions. You do not need to use sudo if you are root.

All installations were trailed on Ubuntu. If you are using a Red-Hat based Linux OS, please switch all installations from apt-get install to yum install.

**-1: CHANGE PASSWORDS**

1. Internet Access
   1. Apt-Get script
2. Disk management
3. Patches
   1. See script CCDC apt
   2. Also wnetstat
4. Batten Down the Hatches!
   1. gufw
   2. ps -ef
      1. root, nobody, avahi, message+, whoopsie, syslog, clamav, Are fine?
   3. ps -u USERNAME
   4. killall -9 -u USERNAME
5. User control
6. File permissions
   1. CCDC\_filepermissions.sh
7. Run tools to detect stuff
   1. Rkhunter
      1. sudo rkhunter -c --enable all --disable none
      2. /var/log/rkhunter.log
   2. Bastille
8. Change Passwords again

## Users

**Command:** sudo passwd *username*

**What?:** This command is used to change the password of the associated user (username).

**Why?:** If someone loses their password or if you need to update yours, this is what you would use.

**Command:** sudo passwd –l *username*

**What?:** This command is used to lock a user out of their account.

**Why?:** If someone is out of the office for leave, then you would lock their account.

**Command:** /usr/sbin/usermod –L –s /bin/false *username*

**What?:** This command is used to lock a user out of their account. (Alternative)

**Why?:** If someone is out of the office for leave, then you would lock their account.

**Command:** sudo adduser *username*

**What?:** This command will add a user account.

**Why?:** If you need a new user account for a new employee.

**Command:** sudo adduser *username* sudo

**What?:** This command will add the specified user to the sudo group

**Why?:** If you want to grant a user sudo privileges

**Command:** sudo deluser *username*

**What?:** This command will delete the specified user

**Why?:** If you no longer need the user account.

**Comments:** Make sure all instances of the user is deleted from /etc/group and delete their /home folder.

**Command:** sudo users

**What?:** This command prints out the user names of users currently logged into the current host

**Why?:** Make sure who is logged on

**Command:** cat /etc/passwd |grep "/home" |cut -d: -f1

**What?:** If you search the passwd file, you can see names of different users.

**Why?:** Check which users are on the client.

**Command:** cat /etc/group

**What?:** This command will show the group names, privileges and which users belong to each group.

**Why?:** Make sure all the users are in the appropriate groups.

## Updating

**Command:** sudo apt-get clean

**What?:** It will clean the /var

**Command:** sudo apt-get update

**What?:** It will download the package lists from repositories and upate

**Command:** sudo apt-get upgrade –y

**What?:** Installs the newer versions of the packages you have

**Command:** sudo apt-get dist-upgrade

**What?:** in addition to performing the function of upgrade, also intelligently handles changing dependencies with new versions of packages; apt-get has a "smart" conflict resolution system, and it will attempt to upgrade the most important packages at the expense of less important ones if necessary. So, dist-upgrade command may remove some packages. The /etc/apt/sources.list file contains a list of locations from which to retrieve desired package files. See also apt\_preferences(5) for a mechanism for overriding the general settings for individual packages

## Scheduled Tasks

Check crontabs (These are the schedule tasks)

/etc/cron.d

/etc/cron.hourly

/etc/cron.daily

/etc/cron.weekly

/etc/cron.monthly

/var/spool/cron/crontab

Lockdown cronjobs

**echo ALL >>\etc\cron.deny**

## Installations:

**Command:** sudo apt-get install synaptic

**What?:** Synaptic is a GUI package manager

**Command:** sudo apt-get install gufw

**What?:** This is the firewall GUI

**Command:** sudo apt-get install rkhunter

**What?:** This program will look for rootkits and malicious code

Use: sudo rkhunter -c --enable all --disable none

**Command:** sudo apt-get install chkrootkit

**What?:** This program will look for rootkits and malicious code

**Command:** sudo apt-get install nmap (or zenmap)

**What?:** nmap is a program for network discovery and security auditing (zenmap = GUI)

**Command:** sudo apt-get install clamav clamtk | sudo freshclam

**What?:** This program is an anti-virus. Freshclam updates the signatures.

**Command:** sudo apt-get install tripwire

**What?:** Tripwire is a security and data integrity tool for monitoring and alerting changes on a system.

**Command:** sudo apt-get install bastille

**What?:** The Bastille Hardening program "locks down" an operating system, proactively configuring the system for increased security and decreasing its susceptibility to compromise. Bastille can also assess a system's current state of hardening, granularly reporting on each of the security settings with which it works.

**Command:** sudo apt-get install chkconfig

**What?:** updates and queries runlevel information for system services

## Processes

**Command:** chkconfig –list | grep ‘3:on’

**What?:** Check what is running

**Command:** sudo ps –ef

**What?:** check all running services

**Command:** pgrep -u USERNAME

**What?:** Will list process IDS under a specific user

**Command:** killall -9 -u USERNAME

**What?:** will kill all processes under a certain user

Other: If CENTos: yum install psmisc

Also: dnf install psmisc

## Logon

Modify the **/etc/lightdm/lightdm.conf** to remove Auto-Login

Put a # in front of all lines containing auto-login

Modify the **/etc/lightdm/lightdm.conf** to remove Guest Session

Add **allow-guest=false**

## Passwords

To strengthen passwords, we are going to modify the **/etc/login.defs** file or the **/etc/pam.d/system-auth**

password requisite pam\_cracklib.so try\_first\_pass retry=3 minlength=12 lcredit=1

ucredit=1 dcredit=1 ocredit=1 difok=4

Here's what each of the available parameters does:

try\_first\_pass = sets the number of times users can attempt setting a good

password before the passwd command aborts

minlen = establishes a measure of complexity related to the password length

(more in a moment on this)

lcredit = sets the minimum number of required lowercase letters

ucredit = sets the minimum number of required uppercase letters

dcredit = sets the minimum number of required digits

ocredit = sets the minimum number of required other characters

difok = sets the number of characters that must be different from those in the previous password

## Network

**Command:** sudo netstat –tulpn

**What?:** prints network connections/routing tables/interface stats

t-

u-

l- show only listening sockets

p- show the PID and name of the program to which each socket belongs

n- show numerical addresses

(Remember kill -9 [Process ID])

**Command:** echo “0” > /proc/sys/net/ipv4/ip\_forward

**What?:** Disables IP Forwarding

Modify the **/etc/systemctl.conf** to look like:

# Protect ICMP attacks

net.ipv4.icmp\_echo\_ignore\_broadcasts = 1

# Turn on protection for bad icmp error messages

net.ipv4.icmp\_ignore\_bogus\_error\_responses = 1

# Turn on syncookies for SYN flood attack protection

net.ipv4.tcp\_syncookies = 1

# Log suspcicious packets, such as spoofed, source-routed, and redirect

net.ipv4.conf.all.log\_martians = 1

net.ipv4.conf.default.log\_martians = 1

# Disables these ipv4 features, not very legitimate uses

net.ipv4.conf.all.accept\_source\_route = 0

net.ipv4.conf.default.accept\_source\_route = 0

# Enables RFC-reccomended source validation (dont use on a router)

net.ipv4.conf.all.rp\_filter = 1

net.ipv4.conf.default.rp\_filter = 1

# Make sure no one can alter the routing tables

net.ipv4.conf.all.accept\_redirects = 0

net.ipv4.conf.default.accept\_redirects = 0

net.ipv4.conf.all.secure\_redirects = 0

net.ipv4.conf.default.secure\_redirects = 0

# Host only (we're not a router)

net.ipv4.ip\_forward = 0

net.ipv4.conf.all.send\_redirects = 0

net.ipv4.conf.default.send\_redirects = 0

# Turn on execshild

kernel.exec-shield = 1

kernel.randomize\_va\_space = 1

# Tune IPv6

net.ipv6.conf.default.router\_solicitations = 0

net.ipv6.conf.default.accept\_ra\_rtr\_pref = 0

net.ipv6.conf.default.accept\_ra\_pinfo = 0

net.ipv6.conf.default.accept\_ra\_defrtr = 0

net.ipv6.conf.default.autoconf = 0

net.ipv6.conf.default.dad\_transmits = 0

net.ipv6.conf.default.max\_addresses = 1

# Optimization for port usefor LBs

# Increase system file descriptor limit

fs.file-max = 65535

# Allow for more PIDs (to reduce rollover problems); may break some programs 32768

kernel.pid\_max = 65536

# Increase system IP port limits

net.ipv4.ip\_local\_port\_range = 2000 65000

# Increase TCP max buffer size setable using setsockopt()

net.ipv4.tcp\_rmem = 4096 87380 8388608

net.ipv4.tcp\_wmem = 4096 87380 8388608

# Increase Linux auto tuning TCP buffer limits

# min, default, and max number of bytes to use

# set max to at least 4MB, or higher if you use very high BDP paths

net.core.rmem\_max = 8388608

net.core.wmem\_max = 8388608

net.core.netdev\_max\_backlog = 5000

net.ipv4.tcp\_window\_scaling = 1

## Misc

**Run Bastille**

**Command:** sudo bastille –c

* File permissions module: Yes (suid)
* Disable SUID for mount/umount: Yes
* Disable SUID on ping: Yes
* Disable clear-text r-protocols that use IP-based authentication? Yes
* Enforce password aging? No (situation dependent, I have no users accessing my machines except me, and I only allow ssh keys)
* Default umask: Yes
* Umask: 077
* Disable root login on tty’s 1-6: No
* Password protect GRUB prompt: No (situation dependent, I’m on a VPS and would like to get support in case I need it)
* Password protect su mode: Yes
* default-deny on tcp-wrappers and xinetd? No
* Ensure telnet doesn’t run? Yes
* Ensure FTP does not run? Yes
* display authorized use message? No (situation dependent, if you had other users, Yes)
* Put limits on system resource usage? Yes
* Restrict console access to group of users? Yes (then choose root)
* Add additional logging? Yes
* Setup remote logging if you have a remote log host, I don’t so I answered No
* Setup process accounting? Yes
* Disable acpid? Yes
* Deactivate nfs + samba? Yes (situation dependent)
* Stop sendmail from running in daemon mode? No (I have this firewalled off, so I’m not concerned)
* Deactivate apache? Yes (UNLESS YOU NEED IT
* Disable printing? Yes
* TMPDIR/TMP scripts? No (if a multi-user system, yes)
* Packet filtering script? No (we configured the firewall previously)
* Finished? YES! & reboot

**Command:** find / -type f -user root -perm -u=s -perm -o=w

**What?:** Search the OS for any file owned by root that has SUID enabled and is writable by non-root users

**Command:** sudo ls –l /etc/passwd

**What?:** Check if there are any hardlinks to passwd (if number is above 1, then there is a hardlink)

**Command:** sudo ls –l /etc/shadow

**What?:** Check if there are any hardlinks to shadow (if number is above 1, then there is a hardlink)

**Command:** echo $PATH (for each user)

**What?:** echos the path of user directories

**Why?:** Check that the root path is not part of the user’s path

**Command:** echo ‘ALERT – Root Shell Access (Server Name) on:’ ‘date’ ‘who’ | mail –s “Alert: Root Access from ‘who | cut –d”(“ –f2 | cut –d”)” –f1’” [your@email.com](mailto:your@email.com)

**What?:** Get notified for root access

**Disable unnecessary TTY Devices**

Modify **/etc/securetty**

Only tty1 and tty2 should be left

Restrict Certain Commands:

chmod 700 /bin/ping

chmod 700 /usr/bin/finger

chmod 700 /usr/bin/who

chmod 700 /usr/bin/w

chmod 700 /usr/bin/locate

chmod 700 /usr/bin/whereis

chmod 700 /sbin/ifconfig

chmod 700 /usr/bin/pico

chmod 700 /usr/bin/vi

chmod 700 /usr/bin/which

chmod 700 /usr/bin/gcc

chmod 700 /usr/bin/make

chmod 700 /bin/rpm

## Telnet

Modify **/etc/xinetd.d/telnet**

Change **disable = no** to **disable = yes**

## SSH

Modify **/etc/ssh/ssh\_config**

Change **Protocol 1,2** to **Protocol 2**

Change **PermitRootLogin yes** to **no**

Change **Port 22** to **55**

Restart using **/etc/rc.d/init.d/sshd\_restart**

## FTP

Modify **/etc/proftpd.conf**

Add **RootLogin off**

Restart FTP

APENDEX

**Ccdc\_apt.sh**

sudo apt-get clean

sudo apt-get update

sudo apt-get upgrade -y

sudo apt-get dist-upgrade

sudo apt-get install synaptic -y

sudo apt-get install rkhunter -y

sudo rkhunter --update

sudo apt-get install gufw -y

sudo apt-get install chkrootkit -y

sudo apt-get install chkconfig -y

sudo apt-get install nmap -y

sudo apt-get install gedit -y

sudo apt-get install purge --auto-remove netcat

sudo apt-get remove netcat-openbsd

**ccdc-wnetstat.sh**

sudo bash -c 'while [ 0 ]; do w -u;sleep 1;done'

sudo bash -c 'while [ 0 ]; do netstat -tulpn; sleep 1;done'

**ccsc-filepre.sh**

#!/bin/bash

echo "Before ..."

ls -la /etc/group /etc/gshadow /etc/passwd /etc/shadow /bin/ping /usr/bin/finger /usr/bin/who /usr/bin/w /usr/bin/locate /usr/bin/whereis /sbin/ifconfig /usr/bin/pico /usr/bin/vi /usr/bin/which /usr/bin/gcc /usr/bin/make /bin/rpm

chown root:root /etc/group /etc/gshadow /etc/passwd /etc/shadow

chmod 0644 /etc/group /etc/passwd

chmod 0400 /etc/gshadow /etc/shadow

chmod 700 /bin/ping

chmod 700 /usr/bin/finger

chmod 700 /usr/bin/who

chmod 700 /usr/bin/w

chmod 700 /usr/bin/locate

chmod 700 /usr/bin/whereis

chmod 700 /sbin/ifconfig

chmod 700 /usr/bin/pico

chmod 700 /usr/bin/vi

chmod 700 /usr/bin/which

chmod 700 /usr/bin/gcc

chmod 700 /usr/bin/make

chmod 700 /usr/bin/nmap

chmod 700 /usr/bin/synaptic-pkexec

chmod 700 /usr/sbin/synaptic

chmod 700 /usr/bin/gufw

chmod 700 /usr/bin/gufw-pkexec

chmod 700 /usr/bin/rkhunter

chmod 700 /bin/rpm

chmod 700 /usr/sbin/chkrootkit

chmod 700 /usr/sbin/tripwire

echo "After ..."

ls -la /etc/group /etc/gshadow /etc/passwd /etc/shadow /bin/ping /usr/bin/finger /usr/bin/who /usr/bin/w /usr/bin/locate /usr/bin/whereis /sbin/ifconfig /usr/bin/pico /usr/bin/vi /usr/bin/which /usr/bin/gcc /usr/bin/make /bin/rpm