

Covers all Exam Objectives for CEHv6



Includes Real-World Scenarios, Hands-On Exercises, and Leading-Edge Exam Prep Software Featuring:

- Custom Test Engine
- Hundreds of Sample Questions
- Electronic Flashcards
- Entire Book in PDF

CEH™

Certified Ethical Hacker

STUDY GUIDE


Exam 312-50
Exam EC0-350

Kimberly Graves



SERIOUS SKILLS.

Chapter 12. Hacking Linux Systems.....	1
Section 12.1. Linux Basics.....	2
Section 12.2. Compiling a Linux Kernel.....	5
Section 12.3. GCC Compilation Commands.....	8
Section 12.4. Installing Linux Kernel Modules.....	9
Section 12.5. Linux Hardening Methods.....	9
Section 12.6. Summary.....	13
Section 12.7. Exam Essentials.....	14
Section 12.8. Review Questions.....	15
Section 12.9. Answers to Review Questions.....	19



Chapter 12

Hacking Linux Systems

CEH EXAM OBJECTIVES COVERED IN THIS CHAPTER:

- ✓ Understand how to compile a Linux kernel
- ✓ Understand GCC compilation commands
- ✓ Understand how to install LKM modules
- ✓ Understand Linux hardening methods



Linux is a popular operating system with system administrators because of its open source code and its flexibility, which allows anyone to modify it. Because of the open source nature

of Linux, there are many different versions, known as *distributions* (or *distros*). Several of the Linux distributions have become robust commercial operating systems for use on workstations as well as servers. Popular commercial distributions include Red Hat, Debian, Mandrake, and SUSE; some of the most common free versions are Gentoo and Knoppix.

Linux's flexibility and the fact that it's open source, together with the increase in Linux applications, have made Linux the operating system of choice for many systems. Although Linux has inherently tighter security than Windows operating systems, it also has vulnerabilities that can be exploited. This chapter covers the basics of getting started using Linux as an operating system and knowing how to harden the system to attacks.

Linux Basics

Linux is loosely based on Unix, and anyone familiar with working in a Unix environment should be able to use a Linux system. All standard commands and utilities are included on most distros.

Many text editors are available inside a Linux system, including vi, ex, pico, jove, and GNU emacs. Many Unix users prefer "simple" editors like vi. But vi has many limitations due to its age, and most modern editors like emacs have gained popularity in recent years.

Most of the basic Linux utilities are GNU software, meaning they are freely distributed to the community. GNU utilities also support advanced features that are not found in the standard versions of BSD and UNIX System. However, GNU utilities are intended to remain compatible with BSD.

A shell is a command-line program interface that allows a user to enter commands, and the system executes commands from the user. In addition, many shells provide features like job control, the ability to manage several processes at once, input and output redirection, and a command language for writing shell scripts. A shell script is a program written in the shell's command language and is similar to an MS-DOS batch file.

Many types of shells are available for Linux. The most important difference among shells is the command language. For example, the C SHell (csh) uses a command language similar to the C programming language. The classic Bourne SHell (sh) uses another command language. The choice of a shell is often based on the command language it provides, and determines which features will be available to the user.

The GNU Bourne Again Shell (bash) is a variation of the Bourne Shell, which includes many advanced features like job control, command history, command and filename completion, and an interface for editing files. Another popular shell is tcsh, a version of the C Shell with advanced functionality similar to that found in bash. Other shells include zsh, a small Bourne-like shell; the Korn Shell (ksh); BSD's ash; and rc, the Plan 9 shell.

Moving around the Linux files system may take a little getting used to if you are primarily a Windows user. The commands in Table 12.1 will help you start to navigate the Linux file system.

TABLE 12.1 Linux file system navigation

Command	Purpose
<code>cd ..</code>	Used to go back one directory in most Unix shells. It is important that the space be between the <code>cd</code> and the two dots (<code>..</code>).
<code>cd -</code>	When in a Korn shell, used to go back one directory.
<code>ls -a</code>	Lists all contents of a directory, including hidden files.
<code>ls -l</code>	Lists all the information about files such as permissions, owners, size, and last modified date.
<code>cp</code>	Copies a file.
<code>mv</code>	Moves a file.
<code>mkdir</code>	Makes a new directory.
<code>rm</code>	Removes a file or directory.

Most Linux file systems are organized with common directories. The directories in Table 12.2 are located on most Linux distros.

TABLE 12.2 Linux directories

Directory	Contents
<code>bin</code>	Binary (executable) files
<code>sbin</code>	System binaries
<code>etc</code>	Configuration files

TABLE 12.2 *Linux directories (continued)*

Directory	Contents
include	Include files
lib	Library files
src	Source files
doc	Documentation files
man	Manual (help) files
share	Shared files

Linux networking commands are similar to the Windows networking commands. For the CEH exam, you should be familiar with the commands in Table 12.3.

TABLE 12.3 Linux networking commands

Command	Description
arp	Used to view the ARP table of MAC addresses mapped to IP addresses
ifconfig	Used to view network interface configuration
netstat	Presents a summary of network connections and sockets
nslookup	Resolves domain names to IP addresses
ping	Tests IP connectivity
ps	Lists all running processes
route	Lists the routing table
shred	Securely deletes a file
traceroute	Traces the path to a destination

Compiling a Linux Kernel

Because of the open source nature of Linux, the source code is freely distributed. The source code is available as binary files, which must be compiled in order to properly operate as an operating system. The binary files are available to anyone and may be downloaded and modified to add or change functionality. There are three reasons a user might want to recompile the Linux kernel:

- You may have some hardware that is so new that there's no kernel module for it in on your distribution CD.
- You may have come across some kind of bug that is fixed in a revision of the operating system.
- You may have some new software application that requires a newer version of the operating system.

Compiling your own linux kernel is great for flexibility, but users should be careful where they download the source code. A site may have bad or infected code, Trojans, or other backdoors added to the source code. For security reasons, only download Linux from known and trusted Internet websites or purchase a commercial distro. A good website to use for downloading Linux distros is www.frozentech.com.

In Exercise 12.1 you will compile a Linux Kernel, and Exercise 12.2 shows how to create a USB bootable Linux Distro.



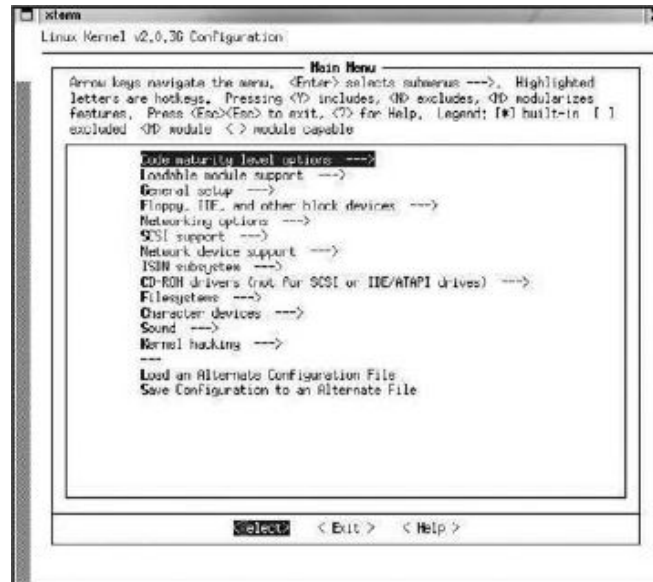
The site I recommend for downloading the Linux kernel is ftp.kernel.org.

EXERCISE 12.1

Configuring and Compiling the Kernel

To download, configure, and compile the Linux kernel, follow these steps:

1. Locate the file for the latest version of the operating system and download it to the `/usr/src` directory on your Linux system. Then use the `tar xzf` command to unpack it.
2. The next step is to configure the Linux kernel. Change directory to `/usr/src/Linux` and type **make menuconfig**. This command will build a few programs and then quickly pop up a window. The window menu lets you alter many aspects of kernel configuration.

EXERCISE 12.1 (continued)

3. After you have made any necessary changes, save the configuration and type **make dep; make clean** at the command prompt. The first of these commands builds the tree of interdependencies in the kernel sources. These dependencies may have been affected by the options you have chosen in the configuration step. The **make clean** command purges any unwanted files left from previous builds of the kernel.
4. Issue the commands **make zImage** and **make modules**. These may take a long time because they are compiling the kernel.
5. The last step is installing the new kernel. On an Intel-based system the kernel is installed in `/boot` with the command:


```
cp /usr/Linux/src/arch/i386/boot/zImage /boot/newkernel
```
6. Issue the command **make modules_install**. This will install the modules in `/lib/modules`.
7. Edit `/etc/lilo.conf` to add a section like this:


```
image = /boot/newkernel
label = newread-only
```
8. At the next reboot, select the new kernel in lilo and it will load the new kernel. If it works, move it to the first position in the `lilo.conf` file so it will boot every time by default. Lilo is a boot loader that most Linux users use for booting a Linux system.

EXERCISE 12.1 (continued)

Example of a "lilo.conf" file (usually located in "/etc/"):

```
# This line is a comment line
#LILO global section
    boot = /dev/hda2
    timeout = 500
    prompt
    default = linuxbox #"linuxbox" is default kernel
    vga = normal
    read-only
#End of global section ends
# bootable kernel "vmlinuz-2.0.36-1" in directory "/boot/"
# kernel number one
    image = /boot/vmlinuz-2.0.36-1
    label = linuxbox
    vga = normal
    root = /dev/hda2
#end of kernel one section
```

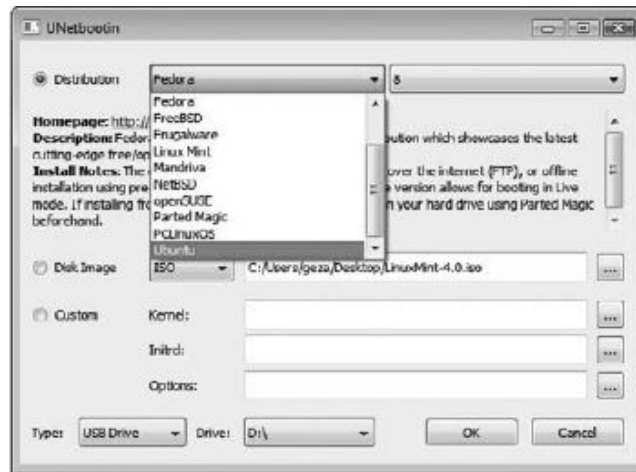


Linux live CDs are a good choice if you're new to Linux. Using the live CD, you can test and use the operating system without installing Linux on the system. To use a live CD, first visit www.distrowatch.com to choose a distribution. Then, download the ISO file and write it to a CD. That CD can be put in any system and booted to a fully functioning version of Linux.

EXERCISE 12.2**Using a Live CD**

In this exercise you will create a Linux live USB drive. Essentially the OS will boot off the USB drive, and then you will have a fully functioning Linux OS to learn how to use some of the Linux commands.

1. Download UNetbootin from sourceforge.net.
2. Run the UNetbootin program.
3. Select the Distribution radio button and click the drop-down menu.

EXERCISE 12.2 (continued)

4. Choose the Linux version from the drop-down menu. The suggested Linux distro for CEH tools is BackTrack, but check the distrowatch.com site to learn which tools are included with each distro. Another option is to download your own Linux ISO file and select the Disk Image radio button.
5. Insert a blank USB drive into your computer. All data on the USB drive will be erased, so ensure it does not contain any files you wish to keep. Make sure your USB drive is large enough to contain the entire ISO image.
6. Choose USB Drive for the type and choose the drive letter for your USB drive.
7. Click OK and wait for UNetbootin to finish formatting and copying the distro files onto the drive.

GCC Compilation Commands

GNU Compiler Collection (GCC) is a command-line compiler that takes source code and makes it an executable. You can download it from <http://gcc.gnu.org> (many Linux distributions also include a version of GCC). GCC can be used to compile and execute C, C++, and FORTRAN applications so they are able to run on a Linux system.

The following command compiles C++ code with the GCC for use as an application:

```
g++ filename.cpp -o outputfilename.out
```

The command to compile C code with the GCC for use as an application is as follows:

```
gcc filename.c -o outputfilename.out
```

Installing Linux Kernel Modules

Linux Kernel Modules (LKMs) let you add functionality to your operating system without having to recompile the OS.

A danger of using LKMs is that a rootkit can easily be created as an LKM, and if loaded, it infects the kernel. For this reason, you should download LKMs only from a verified good source.

Examples of LKM rootkits are Knark, Adore, and Rtkit. Because they infect the kernel, these rootkits are more difficult to detect than those that do not manifest themselves as LKMs. Once a system has been compromised, the hacker can put the LKM in the /tmp or the /var/tmp directory, which can't be monitored by the system administrator, thereby hiding processes, files, and network connections. System calls can also be replaced with those of the hacker's choosing on a system infected by an LKM rootkit.

The command to load a LKM is `modprobe LKM`.

Linux Hardening Methods

Hardening is the process of improving security on a system by making modifications to the system. Linux can be made more secure by employing some of these hardening methods.

The first step in securing any server, Linux or Windows, is to ensure that it's in a secure location such as a network operations center, which prevents a hacker from gaining physical access to the system.

The next and most obvious security measure is to use strong passwords and not give out usernames or passwords. Administrators should make sure the system doesn't have null passwords by verifying that all user accounts have passwords in the Linux /etc/shadow file.

The default security stance of `deny all` is a good one for hardening a system from a network attack. After applying `deny all`, the administrator can open certain access for specific users. By using the `deny all` command first, the administrator ensures that users aren't being given access to files that they shouldn't have access to. The command to deny all users access from the network looks like this:

```
Cat "All:All">> /etc/hosts.deny
```

Another good way to harden a Linux server is to remove unused services and ensure that the system is patched with the latest bug fixes. Administrators should also check system logs frequently for anything unusual that could indicate an attack.

The following are other overall recommended steps to improve the security of a Linux server:

Operating System Selection and Installation

- Use a widely recognized and known good Linux distribution.
- Set up disk partitioning (or logical volumes), taking into account any security considerations.

- After the initial operating system installation, apply any operating system patches that have been released since the installation media was created.
- Set up and enable IP tables.
- Install a host-based intrusion detection system (HIDS).
- Don't install unnecessary applications or services.
- Enable the high security/trusted operating system version if appropriate.
- Secure the boot loader program (such as lilo or GRUB) with a password.
- Enable the single-user mode password if necessary.

Securing Local File Systems

- Look for inappropriate file and directory permissions, and correct any problems you find. The most important of these are:
 - Group and/or world writable system executables and directories
 - Group and/or world writable user home directories
- Select mount options (such as `nosuid`) for local file systems that take advantage of security features provided by the operating system.
- Encrypt sensitive data present on the system.

Configuring and Disabling Services

- Remove or disable all unneeded services. Services are started in several different ways: within `/etc/inittab`, from system boot scripts, or by `inetd`. When possible, the software for an unneeded service should be removed from the system completely.
- Use secure versions of daemons when they are available.
- If at all possible, run server processes as a special user created for that purpose and not as root.
- When appropriate, run servers in an isolated directory tree via the `chroot` facility.
- Set a maximum number of instances for services if possible.
- Specify access control and logging for all services. Install TCP Wrappers if necessary. Allow only the minimum access necessary. Include an entry in `/etc/hosts.deny` that denies access to everyone (so only access allowed in `/etc/hosts.allow` will be permitted).
- Use any per-service user-level access control that is provided. For example the `cron` and `at` subsystems allow you to restrict which users can use them at all. Some people recommend limiting `at` and `cron` to administrators.
- Secure all services, whether they seem security related or not (such as the printing service).

Securing the Root Account

- Select a secure root password, and plan a schedule for changing it regularly.
- If possible, restrict the use of the `su` command to a single group.
- Use `sudo` or system roles to grant other ordinary users limited root privilege when needed.
- Prevent direct root logins except on the system console.

Defining User Account Password Selection and Aging Settings

- Set up default user account restrictions as appropriate.
- Set up default user initialization files in `/etc/skel`, as well as the system-wide initialization files.
- Ensure that administrative and other system accounts to which no one should ever log in have a disabled password and `/bin/false` or another non-login shell.
- Remove unneeded predefined default accounts.

Securing Remote Authentication

- Disable `/etc/hosts.equiv` and `.rhosts` password-less authentication.
- Use `ssh` and its related commands for all remote user access. Disable `rlogin`, `rsh`, `telnet`, `ftp`, `rcp`, and so on.

Performing Ongoing System Monitoring

- Configure the `syslog` facility. Send or copy `syslog` messages to a central `syslog` server for redundancy.
- Enable process accounting.
- Install `Tripwire`, configure it, and record system baseline data. Write the data to removable media and then remove it from the system. Finally, configure `Tripwire` to run on a daily basis.
- Design and implement a plan for monitoring log information for security-related events.

Performing Miscellaneous Activities

- Remove any remaining source code for the kernel or additional software packages from the system.
- Add the new host to the security configuration on other systems, in router access control lists, and so forth.
- Check for vendor security updates for any installed software.

Exercise 12.3 shows how to detect listening ports on a Linux system.

EXERCISE 12.3

Detecting Listening Network Ports

One of the most important tasks in securing Linux is to detect and close network ports that are not needed. This exercise will show you how to get a list of listening network ports (TCP and UDP sockets).

1. Boot the BackTrack Linux USB drive you created in an earlier exercise. Note that BackTrack is not necessary for this exercise. These commands will work with any Linux installation.
2. Open a command window and type **netstat -tulp**. This command will display a list of open ports on your system.

Network	Destination	Netmask	Gateway	Interface	Metric
0.0.0.0	0.0.0.0	0.0.0.0	10.122.56.1	10.122.56.61	25
10.122.56.0	255.255.255.128		On-link	10.122.56.61	281
10.122.56.61	255.255.255.255		On-link	10.122.56.61	281
10.122.56.127	255.255.255.255		On-link	10.122.56.61	281
127.0.0.0	255.0.0.0		On-link	127.0.0.1	306
127.0.0.1	255.255.255.255		On-link	127.0.0.1	306
192.168.192.0	255.255.255.0		On-link	192.168.192.1	276
192.168.192.1	255.255.255.255		On-link	192.168.192.1	276
192.168.192.255	255.255.255.255		On-link	192.168.192.1	276
192.168.227.0	255.255.255.0		On-link	192.168.227.1	276
192.168.227.1	255.255.255.255		On-link	192.168.227.1	276
192.168.227.255	255.255.255.255		On-link	192.168.227.1	276

Another method for listing all the TCP and UDP sockets to which programs are listening is **lsof**. The syntax to run this command is:

```
# lsof -i -n | egrep 'COMMAND|LISTEN|UDP'
```

3. The next step to harden the Linux installation is to disable unused services. The start/stop scripts of all runlevel services can be found in the `/etc/init.d` directory. For example, if you don't know what the `atd` service does, go to `/etc/init.d` and open the file `atd`. In the script look for lines that start programs. In the `atd` script, the `daemon /usr/sbin/atd` line starts the binary `atd`. Then, having the name of the program that is started by this service, you can check the online pages of `atd` by running `man atd`. This will help you to find out more about a system service.

To permanently disable a service—in this example, the runlevel service `nfs`—type the following command:

```
chkconfig nfs off
```



Real World Scenario

Hacking a Default Linux Installation

I worked at a small consulting company where most of the consultants were experts on Windows systems but lacked experience in other operating systems. One of our customers wanted to use Linux for the e-commerce site, and so, because our company wanted to keep them as a customer, we agreed to install the Linux system for them. Because none of the consultants had much experience with Linux, the system was installed with many default options and standard services.

Soon after the new system was installed, the e-commerce portal was hacked and the customer database was compromised. Customer personal information and credit card numbers were exposed by the hackers. Additionally, the company experienced a denial-of-service attack and the site was not available to customers, causing a loss of business.

After the attack, another consulting company specializing in security performed some forensics analysis and determined that access rights for the users and groups on the Linux system were set to the defaults, which hackers exploited to attack the systems. The consulting company recommended to our organization that in the future Linux should be hardened after installation by setting up and enabling IP tables, configuring the Linux security-related kernel parameters, disabling the unnecessary daemons and network services, changing default passwords, and disabling the remote root logins over ssh.

Summary

It is important to understand the basics of the Linux operating system as many application and web servers run an underlying version of Linux. For the CEH exam, you should be familiar with how to use the Linux OS and know the steps you should take to harden a default Linux installation. Live CDs or USB drives are a great way to learn how to use the basic tools if you are new to Linux.

Exam Essentials

Understand the use of Linux in the marketplace. Linux has become popular with the introduction of commercial versions and available applications. Linux can be used as a hacking platform, as a server, or as a workstation.

Know how to use a Linux live CD. Locate and download an ISO file. Write it to a CD, and boot a system from the CD to use the Linux operating system.

Know the steps to create a Linux operating system. Locate and download the binary files, and compile the Linux source files; then, install the compiled OS.

Know how to harden a Linux system. Use a known good distribution, change the default passwords, disable the root login, use IP tables, use an HIDS, apply the latest fixes, and monitor log files to harden a Linux system.

Understand how LKMs are used. LKMs add functionality to a Linux system, but they should be used only from a known good source.

Know about GCC compilation. GCC compilers are used to create executable applications from C or C++ source code.

Review Questions

1. What does LKM stand for?
 - A. Linux Kernel Module
 - B. Linux Kernel Mode
 - C. Linked Kernel Module
 - D. Last Kernel Mode
2. What GCC command is used to compile a C++ file called `source` into an executable file called `game`?
 - A. `g++ source.c -o game`
 - B. `gcc source.c -o game`
 - C. `gcc make source.cpp -o game`
 - D. `g++ source.cpp -o game`
3. What is the command to deny all users access from the network?
 - A. `Cat "All:All">> /etc/hosts.deny`
 - B. `Set "All:All">> /etc/hosts.deny`
 - C. `IP deny "All:All"`
 - D. `Cat All:All deny`
4. Of the following, which are common commercial Linux distributions?
 - A. SUSE, Knark, and Red Hat
 - B. SUSE, Adore, Debian, and Mandrake
 - C. SUSE, Debian, and Red Hat
 - D. SUSE, Adore, and Red Hat
5. What is a Linux live CD?
 - A. A Linux operating system that runs from a CD
 - B. A Linux operating system installed from a CD onto a hard drive
 - C. A Linux tool that runs applications from a CD
 - D. A Linux application that makes CDs
6. What type of attack can be disguised as an LKM?
 - A. DoS
 - B. Trojan
 - C. Spam virus
 - D. Rootkit

7. Which of the following is a reason to use Linux?
 - A. Linux has no security holes.
 - B. Linux is always up-to-date on security patches.
 - C. No rootkits can infect a Linux system.
 - D. Linux is flexible and can be modified.
8. Which of the following is *not* a way to harden Linux?
 - A. Physically secure the system.
 - B. Maintain a current patch level.
 - C. Change the default passwords.
 - D. Install all available services.
9. What type of file is used to create a Linux live CD?
 - A. ISO
 - B. CD
 - C. LIN
 - D. CDFS
10. Why is it important to use a known good distribution of Linux?
 - A. Source files can become corrupted if not downloaded properly.
 - B. Only certain distributions can be patched.
 - C. Source files can be modified, and a Trojan or backdoor may be included in the source binaries of some less-known or free distributions of Linux.
 - D. Only some versions of Linux are available to the public.
11. What command will give you the most information Linux files?
 - A. `ls -a`
 - B. `ls -m`
 - C. `ls -t`
 - D. `ls -l`
12. What is the purpose of the `man` command?
 - A. Lists help and documentation
 - B. Manually configures a program
 - C. Performs system maintenance
 - D. Installs a program
13. In which directory are Linux system source files located?
 - A. `source`
 - B. `src`
 - C. `sys`
 - D. `system`

14. What is the Linux command that lists all current running processes?
- A. ps
 - B. list ps
 - C. show ps
 - D. process
15. What is the Linux command for viewing the IP address of a network interface?
- A. ifconfig
 - B. ipconfig
 - C. ipconfig /all
 - D. interface /ip
16. Which Linux command would produce the following output?

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	*:ssh	*:*	LISTEN
tcp	0	0	*:10000	*:*	LISTEN
tcp	0	0	localhost:SMTP	*:*	LISTEN
udp	0	0	*:56315	*:*	
udp	0	0	*:icmpv2	*:*	

- A. routing
 - B. route print
 - C. route
 - D. show routes
17. What is a recommended way to secure the Linux root account? (Choose all that apply.)
- A. Prevent direct root logins except from the system console.
 - B. Restrict the use of su to a single group.
 - C. Install su protect to prevent misuse of the su command.
 - D. Grant the admin privilege to any user needing to install programs.
18. When you are securing local Linux file systems, which two types of directories should you be check for appropriate permissions? (Choose two.)
- A. Root directory
 - B. Services directory
 - C. Writable system executable directories
 - D. Writable user home directories

19. What is the Cat command you would use to harden the file system of a Linux system?
- A. Cat "source=All:destination=All">> /etc/hosts.deny
 - B. Cat "All:All">> /etc/hosts.deny
 - C. Cat "Any:Any">> /etc/hosts.deny
 - D. Cat "All:All" /etc/hosts.deny
20. In which file should you check to ensure users do not have a null password in a Linux system?
- A. Password file
 - B. Passwd file
 - C. Shadow file
 - D. Shdw file

Answers to Review Questions

1. A. LKM stands for Linux Kernel Module.
2. D. `g++ source.cpp -o game` is the GCC command to create an executable called `game` from the source file `source`.
3. A. Use the `Cat "All:All">> /etc/hosts.deny` command to deny all users access from the network on a Linux system.
4. C. SUSE, Debian, and Red Hat are all commercial versions of Linux.
5. A. A Linux live CD is a fully functioning operating system that runs from a CD.
6. D. A rootkit can be disguised as an LKM.
7. D. Linux is flexible and can be modified because the source code is openly available.
8. D. Linux should not have unused services running, because each additional service may have potential vulnerabilities.
9. A. An ISO file is used to create a Linux live CD.
10. C. Known good distributions have been reviewed by the Linux community to verify that a Trojan or backdoor does not exist in the source code.
11. D. The command `ls -l` lists all the information about files such as permissions, owners, size, and last modified date.
12. A. The `man` command will list help and documentation in Linux.
13. B. The `src` directory contains the Linux source files.
14. A. The `ps` command lists all running processes.
15. A. Use the `ifconfig` command to view the IP address of a network interface. `ipconfig` and `ipconfig/all` are Windows commands to view IP address information.
16. C. `route` displays the routing table. `route print` is a Windows command to display the routing table. `show routes` is a command commonly used to view a routing table.
17. A, B. The recommended way to secure the Linux root account is to prevent direct root logins and to restrict the use of `su` to one group.
18. C, D. Writable system executable directories and writable user home directories should both be checked as they could be used to execute malicious code.
19. B. Use the command `Cat "All:All">> /etc/hosts.deny` to harden a Linux system and ensure all users are denied access to certain files from the network.
20. C. User passwords in a Linux system are stored in the shadow file. To harden a system, check the shadow file for null passwords.

