

6

ANSWERS TO EVEN-NUMBERED EXERCISES

2. List the commands you can use to perform these operations:
 - a. Make your home directory the working directory
 - b. Identify the working directory
4. The `df` utility displays all mounted filesystems along with information about each. Use the `df` utility with the `-h` (human-readable) option to answer the following questions.

```
$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/hda1       1.4G  242M  1.1G  18% /
/dev/hda3        23M   11M   10M  51% /boot
/dev/hda4       1.5G  1.2G  222M  85% /home
/dev/hda7       564M   17M  518M   4% /tmp
/dev/hdc1       984M   92M  842M  10% /gc1
/dev/hdc2       16G   13G  1.9G  87% /gc2
```

- a. How many filesystems are mounted on your Linux system?
- b. Which filesystem stores your home directory?
- c. Assuming your answer to exercise 4a is two or more, attempt to create a hard link to a file on another filesystem. What error message do you get? What happens when you attempt to create a symbolic link to the file instead?

Following are sample answers to these questions. Your answers will be different because your filesystems are different.

a. six; b. `/dev/hda4`; c. **ln: creating hard link '/tmp/xxx' to 'xxx': Invalid cross-device link.** No problem creating a cross-device symbolic link.

6. You should have read permission for the **/etc/passwd** file. To answer the following questions, use `cat` or `less` to display **/etc/passwd**. Look at the fields of information in **/etc/passwd** for the users on the local system.
- Which character is used to separate fields in **/etc/passwd**?
 - How many fields are used to describe each user?
 - How many users are on the local system?
 - How many different login shells are in use on your system? (*Hint*: Look at the last field.)
 - The second field of **/etc/passwd** stores user passwords in encoded form. If the password field contains an **x**, your system uses shadow passwords and stores the encoded passwords elsewhere. Does your system use shadow passwords?
- a. colon (:); b. seven; c, d, and e are system dependent
8. Suppose a user belongs to a group that has all permissions on a file named **jobs_list**, but the user, as the owner of the file, has no permissions. Describe which operations, if any, the user/owner can perform on **jobs_list**. Which command can the user/owner give that will grant the user/owner all permissions on the file?

Initially the user/owner cannot perform any operations involving the file, other than using `ls` to list it. When the user/owner gives the following command, the user/owner can perform any operation involving the file:

```
$ chmod u+rwx jobs_list
```

10. Assume you are given the directory structure shown in Figure 6-2 on page 177 and the following directory permissions:

```
d--x--x--- 3 zach pubs 512 2010-03-10 15:16 business
drwxr-xr-x 2 zach pubs 512 2010-03-10 15:16 business/milk_co
```

For each category of permissions—owner, group, and other—what happens when you run each of the following commands? Assume the working directory is the parent of **correspond** and that the file **cheese_co** is readable by everyone.

- a. **cd correspond/business/milk_co**

owner: OK; group: OK; other: **Permission denied**

- b. **ls -l correspond/business**

owner, group, and other: **Permission denied**

- c. **cat correspond/business/cheese_co**

owner and group: **Is a directory**; other: **Permission denied**

12. What does the `..` entry in a directory point to? What does this entry point to in the root (`/`) directory?

The `..` entry is a link to the parent directory. In the case of the root directory, there is no parent, and the `..` entry is a link to the root directory itself.

14. Suppose the working directory contains a single file named **andor**. What error message do you get when you run the following command line?

```
$ mv andor and\or
```

Under what circumstances is it possible to run the command without producing an error?

```
$ mv andor and\or
mv: cannot move 'andor' to 'and/or': No such file or directory
$ mkdir and
$ mv andor and\or
$ ls and
or
```

(The backslash is superfluous.)

16. Do you think the system administrator has access to a program that can decode user passwords? Why or why not? (See exercise 6.)

Normally, the system administrator cannot decode user passwords. The administrator can assign a new password to a user. Passwords are generally encrypted by a one-way hash so the system can tell when the correct password is entered, but it cannot regenerate the cleartext password. The system applies the hash algorithm to the entered password and checks whether the result matches the stored, encrypted password. A match means the correct password was entered.

However, if a user has a weak password, the system administrator can use a program such as `crack` or `John the Ripper` to decode a password. You can download either of these utilities using `yum`. The packages are named **crack** and **john**.

18. Explain the error messages displayed in the following sequence of commands:

```
$ ls -l
total 1
drwxrwxr-x. 2 max pubs 1024 03-02 17:57 dirtmp
$ ls dirtmp
$ rmdir dirtmp
rmdir: dirtmp: Directory not empty
$ rm dirtmp/*
rm: No match.
```

There is a file whose name begins with a period (a hidden file) in the **dirtmp** directory. Use `ls` with the `-a` option to list the file. Remove the file, and then you will be able to remove the directory.

