

# 18

## ANSWERS TO EVEN-NUMBERED EXERCISES

2. How can you use `ssh` to find out who is logged in on a remote system?

Assuming you have the same username on both systems, the following command might prompt you for your password on the remote system; it displays the output of `who` run on *host*:

```
$ ssh host who
```

4. How would you use `ssh` to run `xterm` on `plum` and show the display on the local system?

Assuming you have the same username on both systems and an X11 server running locally, the following command runs `xterm` on `plum` and presents the display on the local system:

```
$ ssh plum xterm
```

You need to use the `-Y` option if trusted X11 forwarding is not enabled.

6. When you try to connect to a remote system using an OpenSSH client and you see a message warning you that the remote host identification has changed, what has happened? What should you do?

This message indicates that the fingerprint of the remote system is not the same as the local system remembers it. Check with the remote system's administrator to find out if something changed. If everything is in order, remove the remote system's key from the file specified in the error message and try logging in on the remote system again using `ssh`. You can use `ssh-keygen` with the `-R` option followed by the name of the remote system to remove both hashed and non-hashed entries. The system will display the

first-time authentication message (page 706) again as OpenSSH verifies that you are connecting to the correct system.

8. Which single command could you give to log in as **root** on the remote system named **plum**, if **plum** has remote **root** logins disabled?

Assuming you have the same username on both systems, the following command logs in on **plum** as **root**:

```
$ ssh -t plum su -
```

When you run this command, you must supply two passwords (assuming you are running the command as a user without **root** privileges and you have not set up an automatic login for **ssh**): yours and **root**'s. The **su** utility requires that its input come from standard input; the **-t** option allocates a pseudo-tty (terminal) to run **su**.

10. How would you use **rsync** with OpenSSH authentication to copy the **memos12** file from the working directory on the local system to your home directory on **plum**? How would you copy the **memos** directory from the working directory on the local system to your home directory on **plum** and cause **rsync** to display each file as it copied the file?

```
$ rsync memos12 plum:
```

```
$ rsync -av memos plum:
```