21

Answers to Even-numbered Exercises

2. How would Max store a copy of his email in **~/mbox** and send a copy to **max@example.com**?

   Max needs to create a ~/**.forward** file with the following lines:

   ```
   $ cat ~/.forward
   ~/mbox
   max@example.com
   \max
   ```

4. What does the value of the **myorigin** parameter in **/etc/postfix/main.cf** specify?

   The **myorigin** parameter specifies the name of the sending domain or system on the envelope-from and From lines of email that originates on the local system.

6. Describe the software and protocols used when Max sends an email to Sam on a remote Linux system.

   One possible answer of many:

   a. Max's MUA passes the email to **postfix**.

   b. The **postfix** daemon uses SMTP to deliver the email to the instance of **postfix** on Sam's system.

   c. On Sam's system, **postfix** passes the email to the local MDA (**procmail**).

   d. The MDA stores the email in Sam's file in the mail spool directory.

   e. Sam's MUA retrieves the email from the mail spool directory.

8. Assume a script stores certain information in a variable named **RESULT**. What line could you put in the script that would send the contents of **RESULT** to the email address specified by the first argument on the command line?

```
        echo "$RESULT" | sendmail $1
```

or

```
        echo "$RESULT" | mail $1
```

10. Describe the relationship between spamassassin, **spamd**, and spamc. How does each work? Why not use the spamassassin utility by itself?

The spamassassin utility is a stand-alone mail filter that rates and tags email for the likelihood of being spam. The spamc utility is a mail filter that is similar to spamassassin but more efficient and quicker to load.

The spamc utility is lighter weight than spamassassin because it offloads the work of evaluating each piece of email to the **spamd** daemon, which is always running in the background.

Loading a large program such as spamassassin for each piece of email a mail server receives imposes a substantial overhead on the server system. Because spamc calls **spamd**, which does the heavy lifting and is already loaded, spamc can be lightweight and can impose a minimal overhead when it is called.