Courier worksheet

Courier is a mail system which includes a number of packages. In fact it has its own MTA, but we will ignore this (it is still under heavy development, and does not have the flexibility needed for an ISP environment). The components we are interested in are the IMAP/POP3 server, the 'deliverquota' program which Exim can use to deliver Maildir++ mailboxes more efficiently, and 'sqwebmail'. The main limitation of the Courier programs is their lack of debugging output when things aren't configured properly.

You can get the entire courier system as one package (including the MTA), or just the components.

---

1. Install courier-imap

Homepage: http://www.inter7.com/courierimap/

courier-imap can be compiled and installed as Red Hat packages. However, we will simply compile from source and install directly; this automatically detects which database libraries you have available, if any, and adjusts the compilation to match.

(If you want to build it into an RPM, follow the instructions at http://www.courier-mta.org/FAQ.html#rpm)

If you don't have them already, fetch **courier-imap-2.2.1.tar.bz2** and **sqwebmail-3.6.2.tar.bz2** from the **pub/software** directory on noc.taller.nsrc.org

---

> When Courier builds it normally installs itself under `/usr/lib/courier-imap`. If you wish, you can tell the configure script to install elsewhere, e.g.
>
> ```
> ./configure --prefix=/usr/courier-imap
> ```
>
> We will not do that in this exercise.

```
$ cd
$ tar -xvjf /path/to/file/courier-imap-2.2.1.tar.bz2
$ cd courier-imap-2.2.1
$ ./configure --with-redhat
... takes a while
$ make
... takes a while
$ make check
... takes a short while, check there are no errors displayed
$ su
Password: <root password>
# make install
# make install-configure
```

To get access to the man pages, edit **/etc/man.config** and add the following line:

```
MANPATH              /usr/lib/courier-imap/man
```

Test with 'man authuserdb' and see if the page is displayed.

2. Configure the daemons

Courier can get its user and password information from a variety of places, using authentication modules. Some of these, like mysql and ldap, are only compiled if those pieces of software are already installed (see INSTALL in the source directory for more information).

With current versions of Courier-IMAP, all these authentication methods are built into a single authentication daemon, "authdaemond"

For a large server you may wish to increase the maximum number of concurrent connections from the default of 40, if you have a fairly powerful mailserver:

```
# cd /usr/lib/courier-imap/etc
# vi pop3d
...
MAXDAEMONS=300
...
# vi imapd
...
MAXDAEMONS=300
...
```

Now you should configure the authdaemond process not to use any authentication mechanisms which you know you don't need. For example, if all your authentication is only via PAM for Unix system passwords, then you can remove all the others:

```
# cd /usr/lib/courier-imap/etc
# vi authdaemonrc
...
authmodulelist="authpam"
...
```

The "authpam" modules lets you authenticate against your Unix password file (PAM = Pluggable Authentication Modules). You need files /etc/pam.d/pop3 and /etc/pam.d/imap, but in Red Hat 9.0 they are already there. If you do not remove the other modules, you may find there are timeouts as (for example) authldap tries to contact a non-existent LDAP server.

3. Start the daemons

Make sure you have no existing pop3 server enabled in /etc/xinetd.d/ipop3 or /etc/xinetd.d/imap (i.e. they have 'disable = yes'). You can start the pop3 and/or imap servers using one or both of the following commands, which can go in /etc/rc.local to run at startup time:

```
# /usr/lib/courier-imap/libexec/pop3d.rc start
# /usr/lib/courier-imap/libexec/imapd.rc start

To check they are running: (they have long command lines!)
# ps auxwww | grep couriertcpd
```

Test using telnet:

```
# telnet localhost 110
Connected to localhost.
Escape character is '^]'.
+OK Hello there.
user username
+OK Password required.
```

```
pass password
+OK logged in.
stat
+OK 26 49857
retr 1
+OK 1073 octets follow.
... message
.
quit
+OK Bye-bye.
Connection closed by foreign host.

# telnet localhost 143
Connected to localhost.
Escape character is '^]'.
* OK Courier-IMAP ready. Copyright 1998-2001 Double Precision, Inc.  See
COPYING for distribution information.
a001 login <username> <password>
a001 OK LOGIN Ok.
a002 examine inbox
* FLAGS (\Answered \Flagged \Deleted \Seen \Recent)
* OK [PERMANENTFLAGS ()] No permanent flags permitted
* 26 EXISTS
* 0 RECENT
* OK [UIDVALIDITY 989061119] Ok
a002 OK [READ-ONLY] Ok
a003 logout
* BYE Courier-IMAP server shutting down
a003 OK LOGOUT completed
Connection closed by foreign host.
```

NOTE: The daemons will fail to login if the mail directory does not exist, although current versions do now provide an error message. Hence you need to have delivered at least one message to the user, to create their mailbox, before they can login.

> **SECURITY NOTE:** If you had built your system so that users are not in the password file, and all mailboxes are owned by a single userid (exim), then you could tell couriertcpd to start the imapd/pop3d daemons as exim rather than root. To do this, you would edit /usr/lib/courier-imap/etc/pop3d and imapd and add "-user=exim" to couriertcpd options, i.e.
>
> ```
> ...
> TCPDOPTS="-nodnslookup -noidentlookup -user=exim"
> ...
> ```
>
> However, do not do this in the exercise.

---

4. pop3 and imap over SSL

If you wish, you can choose to run pop3 over SSL (port 995) and imap over SSL (port 993). The advantage is that, for clients which support it, the traffic is encrypted. The disadvantage is higher CPU load on your server for the encryption of data.

To run SSL you will need a certificate. For testing purposes you can use a 'self-signed' certificate, the following scripts will generate them for you:

```
# /usr/lib/courier-imap/sbin/mkpop3dcert
# /usr/lib/courier-imap/sbin/mkimapdcert
```

Then you start the servers:

```
# /usr/lib/courier-imap/libexec/pop3d-ssl.rc start
# /usr/lib/courier-imap/libexec/imapd-ssl.rc start
```

You can't use a regular telnet to test it, because all your communication needs to be encrypted, but openssl has an SSL client you can use to make an encrypted connection for testing:

```
# openssl s_client -connect localhost:995
```

If you were running the service commercially you would be better to get a proper certificate signed by a recognised CA, rather than using a self-signed certificate.

---

5. Using 'deliverquota' from Exim

Courier implements an extension to Maildir called Maildir++ which efficiently keeps track of quota status for each mailbox, even when the user has thousands of messages stored in hundreds of folders (which certainly can be the case when you are using IMAP and webmail). For this to work, all software which accesses the Maildir must adhere to this extension.

To get Exim to deliver using Maildir++, you get it to call the 'deliverquota' program, rather than delivering using its built-in Maildir capability.

Modify the 'localuser' router:

```
localuser:
  driver = accept
  check_local_user
  transport = local_delivery_courier
```

Add a new entry in the 'transports' section:

```
local_delivery_courier:
  driver = pipe
  command = /usr/lib/courier-imap/bin/deliverquota -c -w 90 \
        $home/Maildir 10000000S
  return_fail_output

  delivery_date_add
  envelope_to_add
  return_path_add
```

deliverquota has the useful capability of delivering a warning message when the mailbox gets nearly full (e.g. 90%). To use this, you need to create a warning message:

```
# cd /usr/lib/courier-imap/etc
# cp quotawarnmsg.example quotawarnmsg
# vi quotawarnmsg
```

---

6. sqwebmail

Homepage: http://www.inter7.com/sqwebmail/

webmail is a very useful service to offer your clients - although you may need to be careful of the extra CPU load and bandwidth it might use.

Unlike many other webmail solutions, which use POP3 or IMAP to talk to the mail store, sqwebmail reads and writes Maildir directories directly. This makes it efficient in the case where POP/IMAP and webmail run on the same box, or where there is an NFS-shared mailstore

sqwebmail is feature-rich, very customisable through HTML templates and stylesheets, supports multiple languages, and is simple to install (it runs as a single CGI). Note however that it is still under very active development and hence subject to change quite frequently.

If you don't have it, install and test Apache first. For more detailled information about installing sqwebmail and the options you can specify, read the file INSTALL in the top level source directory.

```
$ cd
$ tar -xvjf /path/to/file/sqwebmail-3.6.2.tar.bz2
$ cd sqwebmail-3.6.2
$ ./configure
... takes a while
$ make configure-check
(check that it has chosen the right locations for cgi-bin and webmail images)
SqWebMail CGI will be installed in /var/www/cgi-bin
Images will be installed in /var/www/html/webmail
URL to the image directory is /webmail
$ make
... takes a while
$ make check
... takes a short while, check there are no errors displayed
$ su
Password: <root password>
# make install-strip
# make install-configure
# cp sqwebmail/sqwebmail.pamconf /etc/pam.d/webmail
```

It will automatically install the CGI as /var/www/cgi-bin/sqwebmail. It uses the same types of authentication modules as Courier-imap (see `/usr/local/share/sqwebmail/authdaemonrc`). As before, you need to edit this file and remove all modules except authpam:

```
...
authmodulelist="authpam"
...
```

Next start the sqwebmail daemon and its authentication daemon:

```
# /usr/local/share/sqwebmail/libexec/sqwebmail/sqwebmaild start
# /usr/local/share/sqwebmail/libexec/authlib/authdaemond start
```

You need to put this in `/etc/rc.local` so that it starts automatically when the machine is next rebooted.

One other change is required: add the following line to `/etc/crontab` to periodically clean out old sessions:

```
0 * * * * bin /usr/local/share/sqwebmail/cleancache.pl
```

Then, using a web browser, go to `http://`*`yourmachine`*`/cgi-bin/sqwebmail` to log in, or https:// if you have SSL running. You should be able to read and send mail.

The INSTALL file in the source tarball has more information on other configuration changes you can make.

SECURITY NOTE: the sqwebmail daemon runs as root. If you are running a system where all mail is owned by the 'exim' user, then you can start it as user 'exim' instead; it will then be impossible for it to access files as a different uid.