Mailing Lists Practical Exercises

Objectives:

Part 1

- Join the afnog.org mailing list
- Build and Install majordomo (a mailing list manager) from the source distribution.
- Setup a working test mailing list.

Part 2

- Build and install majorcool (a web-based administration tool for managing majordomo).
- Test and explore the use of majorcool.

Resources:

Standard tools and commands that we will be using include:

- adduser
- gzip or gunzip
- tar
- newaliases
- chmod & chgrp
- mkdir
- touch

You can get directions for using each of these commands under FreeBSD by typing:

```
$ man <command>
```

(E.g. "`man newaliases`" will bring up the newaliases manual page)

We will also be editing several long files (you can use joe, emacs, or vi) and testing lists using the pine email program or the mail utility.

You can find a copy of the majordomo source: majordomo-1.94.5.tar.gz.

Via the Web at:
http://www.greatcircle.com/majordomo/

Via anonymous FTP at:
ftp://ftp.greatcircle.com/pub/majordomo/
ftp://ftp.sgi.com/other/majordomo/
ftp://ftp.sgi.com/other/majordomo/

or from the nearest mirror site to you.

Before you install majordomo you'll need the following:

- root access
- perl
- exim
- gcc or some other version of cc

You will want to know the complete paths to:

- perl
- gcc

Issue the following commands to confirm that the paths to the above programs are as listed below.

```
$ which perl
/usr/bin/perl
$ which gcc
/usr/bin/gcc
```

The exim binaries (programs) can be found in `/usr/exim/bin` and the configuration file at `/usr/exim/configure`

Preparing the system for Majordomo Installation

Tasks for root

1. Create a user/group to own majordomo:

   We want to run majordomo from a dedicated account in order to avoid the potential security problems that could result from using the majordomo.daemon option. Also to be able to hand off day-to-day management of majordomo to some trusted user other than your root user.

```
# adduser -s

Enter username [a-z0-9_-]: majordomo
Enter full name []: Majordomo
Enter shell bash csh date ksh no sh [sh]: bash
Enter home directory (full path)
[/home/majordomo]: /home/majordomo
Uid [1006]:
Enter login class: default []:
Login group majordomo [majordomo]:
Login group is ``majordomo''. Invite majordomo into other groups: guest no
[no]:
Enter password []:
Enter password again []:

Name:     majordomo
Password: ****
Fullname: Majordomo
Uid:      1006
Gid:      1006 (majordomo)
Class:
Groups:   majordomo
HOME:     /home/majordomo
Shell:    /usr/local/bin/bash
OK? (y/n) [y]:
Added user ``majordomo''
Add another user? (y/n) [y]: n
#
```

2. Check if the majordomo user created is in the /etc/passwd file.

```
# grep majordomo /etc/passwd
majordomo:x:1006:1006:majordomo:/home/majordomo:/usr/local/bin/bash
```

3. Make a note of the uid/gid (in this case 1006/1006) because you will need them later.

NOTE: *If you are installing under linux or solaris you can use the existing tools linuxconf/admintool) to create a user (majordomo) and group (majordomo) and set an initial password and home directory. This home directory will be used to install majordomo files.*

4. Edit the Exim configuration file "`/usr/exim/configure`".

```
# vi /usr/exim/configure
```

5. Add the user "majordomo" as a trusted user to the Exim configuration file. Entry below in bold is in the directive that does that.

`trusted_users = exim:majordomo`

6. Edit the `/etc/aliases` file.

```
# vi /etc/aliases
```

7. Add the following lines at the end of the `/etc/aliases` file. Replace "you" in the text below with your e-mail address on your local machine.

```
# MAJORDOMO MAILINGLIST ALIASES
majordomo:    "|/home/majordomo/wrapper majordomo"
owner-majordomo: you
majordomo-owner: you

# TEST LIST INFO
test:         "|/home/majordomo/wrapper resend -R -l test test-list"
test-list:    :include:/home/majordomo/lists/test
owner-test:  you
test-request:      "|/home/majordomo/wrapper request-answer test"
test-approval:     you
```

Installing the majordomo software:

1. Change directory to the /usr/local/src directory

```
# cd /usr/local/src
```

2. Get the majordomo sources from the ftp site on the workshop noc

```
# ftp noc.ws.afnog.org

Login as "anonymous" and supply your e-mail address as the password.

ftp> cd /pub/t1
ftp> get majordomo-1.94.5.tar.gz
ftp> bye
```

3. Uncompress the files in a temporary working directory (NOT your installation dir) for this exercise you can just use the `/usr/local/src` directory.

```
# tar -xvzf majordomo-1.94.5.tar.gz
# cd majordomo-1.94.5
```

4. Edit the `Makefile` defining:
   - where Perl and the C compiler are
   - the Majordomo home directory (home directory for the user majordomo)
   - the location of the manual pages
   - the user and group that Majordomo will run under
   - the permissions for the various files and directories

   Your changes should look something like the text below. However replace the values for W_USER and W_GROUP (in this case 1006) with the values from the `grep majordomo /etc/passwd` command earlier.

```
PERL = /usr/bin/perl
CC = /usr/bin/gcc
W_HOME = /home/majordomo
MAN = $(W_HOME)/man
W_USER = 1006
W_GROUP = 1006
```

5. Copy `sample.cf` to `majordomo.cf` file.

```
# cp sample.cf majordomo.cf
```

6. Edit the `majordomo.cf` file.

```
# vi majordomo.cf
```

   The important variables to set are:

```
$whereami           What machine am I on?
$whoami             Who do users send requests to me as?
$whoami_owner              Who is the owner of the above, for problems?
$homedir            Where can I find my extra .pl files?
$listdir            Where are the mailing lists?
$log                Where do I write my log?
$sendmail_command   Where the sendmail program resides.
```

```
$mailer      What program and args do I use to send mail to the lists?
$bounce_mailer  What program and args do I use to send administrative
messages?
```

A Sample of these settings might look like this:

```
$whereami = "ns.t1.ws.afnog.org";
$whoami = "majordomo\@$whereami";
$whoami_owner = "owner-majordomo\@$whereami";
if ( defined $ENV{"HOME"}) {
$homedir = $ENV{"HOME"};
} else {
$homedir = "/home/majordomo";
}
$listdir = "$homedir/lists";
$digest_work_dir = "/home/majordomo/digests";
$log = "$homedir/Log";
$sendmail_command = "/usr/exim/bin/exim";
$mailer = "$sendmail_command -oi -oee -f\$sender";
$bounce_mailer = "$sendmail_command -oi -oee -f\$sender -t";

$TMPDIR = $ENV{'TMPDIR'} || "/usr/tmp";
```

7. Run the following make commands.

```
# make wrapper
# make install
# make install-wrapper
```

STOP! don't run the config-test yet!

8. Change directory to the majordomo home directory

```
# cd /home/majordomo
```

9. Now list the contents of the directory.

```
# ls -al
```

What are we missing here?

- `Log` file
- `lists` directory
- `digests` directory

(all named in the `majordomo.cf` but not created during install)

Tasks for majordomo

1. In a new window or terminal login as majordomo or issue the following command to change your identity to the majordomo user.

```
# su majordomo
```

2. Finish the majordomo installation by doing the following in `/home/majordomo`:

```
$ touch Log
$ mkdir lists
```

```
$ mkdir digests
```

3. List the contents of the `/home/majordomo` directory and check if the line items in bold below exist.

```
# ls -al
drwxr-x--x   8 majordomo majordomo     1024 Sep 15 12:46 .
drwxr-xr-x  14 root      root          1024 Sep 15 10:44 ..
drwxr-xr-x   2 majordomo majordomo     1024 Sep 15 10:44 Tools
-rwxr-xr-x   1 majordomo majordomo     5234 Sep 15 10:44 archive2.pl
drwxr-xr-x   2 majordomo majordomo     1024 Sep 15 10:44 bin
-rwxr-xr-x   1 majordomo majordomo     2795 Sep 15 10:44 bounce-remind
-rwxr-xr-x   1 majordomo majordomo    10671 Sep 15 10:44 config-test
-rwxr-xr-x   1 majordomo majordomo    49673 Sep 15 10:44 config_parse.pl
-rwxr-xr-x   1 majordomo majordomo    13031 Sep 15 10:44 digest
drwxrwxr-x   2 majordomo majordomo     1024 Sep 15 10:52 digests
drwxrwxr-x   2 majordomo majordomo     1024 Sep 15 12:46 lists
-rw-rw-r--   1 majordomo majordomo      477 Sep 15 12:46 Log
-rwxr-xr-x   1 majordomo majordomo    64251 Sep 15 10:44 majordomo
-rw-rw-r--   1 majordomo majordomo      410 Sep 15 10:59 majordomo.aliases
-rw-r--r--   1 majordomo majordomo    10485 Sep 15 10:44 majordomo.cf
-rwxr-xr-x   1 majordomo majordomo    23763 Sep 15 10:44 majordomo.pl
-rwxr-xr-x   1 majordomo majordomo      137 Sep 15 10:44 majordomo_version.pl
drwxr-xr-x   4 majordomo majordomo     1024 Sep 15 10:44 man
-rwxr-xr-x   1 majordomo majordomo     3532 Sep 15 10:44 request-answer
-rwxr-xr-x   1 majordomo majordomo    29520 Sep 15 10:44 resend
-rw-r--r--   1 majordomo majordomo    10462 Sep 15 10:44 sample.cf
-rwxr-xr-x   1 majordomo majordomo     8060 Sep 15 10:44 shlock.pl
-rwsr-xr-x   1 root      majordomo     6735 Sep 15 10:45 wrapper
```

4. Now run the `config-test`.

```
$ ./wrapper config-test
```

and you should see a lot of output scroll by that ends as follows:

```
----------------------- end of tests -----------------------

Nothing bad found!  Majordomo _should_ work correctly.

If it doesn't, check your configuration file(/home/majordomo/majordomo.cf)
closely, and if it still looks okay, consider asking the majordomo-users
mailing list at "majordomo-users@greatcircle.com" for assistance.  Be sure
and fully specify what your problems are, and what type of machine (and
operating system) you are using.

Enjoy!
```

Lets do a second test just to be sure:

```
$ cd lists
$ touch test
$ touch test.info
```

The following command should not be issued as the “majordomo” user.

```
$ echo 'lists' |mail -v majordomo
```

If everything worked, the majordomo account should have mail message from majordomo.

Using an majordomo.aliases file

We may want to create a `majordomo.aliases` file where we can store all majordomo-related aliases. This file can be edited by the majordomo account owner (listmaster) and allows a user other than root to manage day to day list creation/deletion/etc.

To use a separate aliases file from the systems /etc/aliases file, edit the exim configuration file.

```
# vi /usr/exim/configure
```

Find the "**system_aliases**:" entry in the exim configuration file.

```
system_aliases:
  driver = redirect
  allow_fail
  allow_defer
  data = ${lookup{$local_part}lsearch{/etc/aliases}}
# user = exim
  file_transport = address_file
  pipe_transport = address_pipe
```

Add the following lines for majordomo.aliases in the exim configuration file below the entries for the system_aliases: in the router section of the configuration file.

```
majordomo_aliases:
  driver = redirect
  allow_fail
  allow_defer
  data = ${lookup{$local_part}lsearch{/home/majordomo/majordomo.aliases}}
  file_transport = address_file
  pipe_transport = address_pipe
  user = majordomo
  group = majordomo
```

Save and exit the `/usr/exim/configure` file.

Now we'll copy the file you'll need to run majordomo.aliases:

```
# cd /home/majordomo
# cp /usr/local/src/majordomo-1.94.5/majordomo.aliases /home/majordomo/
```

Once you've copied the files, set the group ownership for majordomo.aliases to majordomo

```
# chgrp majordomo majordomo.aliases
# chmod 664 majordomo.aliases
```

You can check file permissions and ownership by typing:

```
# ls -al *.aliases
-rw-rw-r--   1 root     majordomo      410 Sep 15 10:59 majordomo.aliases
```

The user majordomo can now add new list aliases using the majordomo.aliases file.

Edit the `majordomo.aliases` file as follows and of course replace “you” with a valid user account on the system or a valid e-mail address.

```
# MAJORDOMO MAILINGLIST ALIASES
majordomo:    "|/home/majordomo/wrapper majordomo"
owner-majordomo: you
majordomo-owner: you

# TEST LIST INFO
test:         "|/home/majordomo/wrapper resend -R -l test test-list"
test-list:    :include:/home/majordomo/lists/test
owner-test:   you
test-request:       "|/home/majordomo/wrapper request-answer test"
test-approval:      you
```

Creating a new list

List names must be of the form "[a-z0-9_-]+" (in other words, letters, digits, underbars, or dashes only). List may be of mixed or uppercase, but all references to the list name below must be specified in lower case EXCEPT for the -l arguments of resend and majordomo.

To create a list:

1. Create an empty file called <list-name> in `/home/majordomo/lists` directory, mode 664.

```
$ cd /home/majordomo/lists
$ touch <list-name>
$ chmod 664 < list-name>
```

2. Create a file called "<list-name>.info" in `/home/majordomo/lists`, mode 664, with the initial introductory info for the list in it.

```
$ touch <list-name>.info
$ chmod 664 < list-name>
```

3. Create the appropriate entries for the list in your Exim aliases file (`/etc/aliases` or a separate `majordomo.aliases` file.) Each list requires several aliases.

<list-name>	The list alias itself
owner-<list-name>	The owner of the list (who should get bounces)
<list-name>-request	The address for administrative requests.
<list-name>-approval	The person who approves postings to the list (for moderated lists as well as unmoderated ones)

The list entries would look like this if your list name is “test”:

```
test:         :include:/home/majordomo/lists/test
owner-test:    you,
test-request:       "|/home/majordomo/wrapper request-answer test"
test-approval:      you
```

It's more likely that the outgoing messages to the list will be passed through "resend" to catch Majordomo commands, as well as give other useful features. A typical set of aliases without anything fancy, such as archiving or digesting, would look like this:

```
test:         "|/ home/majordomo/wrapper resend -l test test-list"
test-list:    :include:/ home/majordomo/lists/test
owner-test:   you,
test-owner:   you
test-request:       "|/ home/majordomo/wrapper majordomo -l test"
```

The following is an example of the listing in the aliases file. To enable archiving for the test mailing list.

```
test:         "|/home/majordomo/wrapper resend -R -l test test-list"
test-list:    :include: /home/majordomo/lists/test, test-archive
owner-test:   you
test-owner:   you
test-request:       "|/usr/majordomo/wrapper request-answer test"
test-approval:      you
test-archive:    "|/home/majordomo/wrapper archive2.pl -a -m -f \
                 /home/majordomo/lists/test.archive/test"
```

You can run majordomo at the <listname>-request address (recommended), and it will handle requests such as:

```
subscribe
unsubscribe
signoff
```

without requiring the user to supply a list name. To do this set up an alias similar to:

```
test-request: "|/home/majordomo/wrapper majordomo -l test"
```

Where the argument to -l is the name of the list.

4. Finally, make sure everything is owned by user majordomo, group majordomo, and writable by both owner and group (i.e., mode 664 for files and mode 775 for directories).

5. Now issue a 'config <listname> <listname>.admin' command to Majordomo. This will cause it to create a default configuration file for the list, and send it back to you. Make any desired changes, SUCH AS CHANGING THE DEFAULT PASSWORDS, as well as adding a description, and send it back with the 'newconfig' command.

6. Now send a test subscribe and unsubscribe, just to further verify that the list is working.

7. Get other members of the class to subscribe to your mailing list.