Appendix F. mh and exmh Customization

This appendix contains information for further customizing the **mh** and/or **exmh** mail readers. Given the multitude of customizable features in these mail readers, we cannot provide you with a comprehensive treatment of the subject here. The information presented in Chapter 12 and this appendix should be sufficient for most users.

We refer you to the on-line help and to the O'Reilly book, *MH & xmh - Email for Users & Programmers* by Jerry Peek, for further information.

F.1 Forwarding and Notification

The `.forward` file is a general mail forwarding file, and is used no matter which mail package you choose to use. It tells the system what to do with your mail; whether to leave it in the system spool area, forward it to the spool area on a different node, forward it to a different system entirely, or (in some cases) deliver it to your preferred local mailbox automatically. It may also be used as a means of notifying you that mail has arrived.

Note that comment lines are generally not permitted in the `.forward` file. Some OS versions and versions of **sendmail** (an underlying, transparent internet mail forwarding tool) do allow comment lines in `.forward`. The comment lines must be preceded by a pound sign (`#`).

F.1.1 Forwarding Address

The first line of your `.forward` file tells the system where to put your mail. On your chosen mail node, it should look like:

```
\{username}@{node}.fnal.gov
```

where `{username}` is your username, and `{node}` is the name of the node where you plan on reading mail (e.g., `\joe@fsui02.fnal.gov`). The backslash prevents infinite loops in a "clustered" environment where more than one node sees the same `.forward` file in your login area.[1]

On all other systems, we recommend that you forward your mail to FNAL, the lab's central mail server. For these systems, the first line of each `.forward` file will look like:

```
{username}@fnal.gov
```

e.g., `joe@fnal.gov`. Note that this doesn't have the backslash or the node name. The backslash is not necessary if you are forwarding to a node that does not share the same login area. See sections 2.6 and 2.7 for a discussion of shared login areas, or what we call UNIX *clusters*.

1. If you choose to use *unattended autoincorporation*, which delivers your mail directly into your login area rather than leaving it in the system spool area, the first line of your `.forward` file will look significantly different from this example. Unattended autoincorporation is covered in section F.4.

F.1.2 Mail Notification

Another use of the `.forward` file is to allow you to customize the type of notification you receive when new mail arrives. Usually, the second line of your `.forward` file will control if and how you receive notification.

FUE provides minimal notification of incoming mail by default, however most people like to receive notification of individual messages as they come in. If you want to scan your new mail when you log in, include the commands **`inc`** and **`scan +inbox unseen`** to your `.login` or `.profile` file.

The use of the **MH** program **rcvtty** provides notification of new mail whether or not you have a mail reader active. To set this up, you need to edit your `.forward` file.

Basic Configuration

Even if you don't know what **biff/comsat** is, you will need to determine if it is running on your system. How do you tell? Follow these instructions and you should get one concise notification message per incoming mail message if it is not running. If you get two notifications, one of which is quite verbose, then **biff/comsat** is running, and you'll need to change your configuration as described below.

If you're on a cluster where some nodes run **biff/comsat** and some don't, set up notification according to your chosen mail node. This procedure assumes that **MH** is installed and configured on all nodes within the cluster.

First, you need to make sure you have set up the proper file access permission. Activate tty user execute permission by using the command:

```
% chmod u+x `tty`
```

where the quotes around **`tty`** are both single back quotes.

You can include this in your `.login` or `.profile` file to run it automatically on all sessions. Be sure to include it *after* the call to `fermi.login` or `fermi.profile` since these files set it differently. Or, if you prefer to have notification only in one window, just run this command in that window.

The second line of your `.forward` file should look like:

```
"| /usr/local/products/mh/current/lib/rcvtty -biff"
```

When mail arrives, **rcvtty** should send a notice containing the time, sender, subject, and message sample to all user tty sessions with proper file access permission in the default format:

```
time:  SENDERNAME   subject << first line of text (truncated)
```

If this type of notification is what you get, skip to "Customizing rcvtty Output", below. If instead you get a multiline notice, **biff/comsat** is running, and you'll have to make a couple of changes.

Some systems run **biff** with **comsat** for notification purposes. **biff/comsat** and **rcvtty** (without the `biff` option) can work together, however this yields double notification (both **comsat**'s verbose notice and **rcvtty**'s single line notice) because the `r-w-x` permissions (see section 6.6.1) need to be different for **biff/comsat** and **rcvtty**.

To get **rcvtty** to work properly on systems running **biff/comsat** follow these steps:

1) Enter **`biff n`** to deactivate **comsat** (best to include in `.login` or `.profile` file).

2) In `.forward`, replace:

```
"| /usr/local/products/mh/current/lib/rcvtty -biff"
```

with:

```
"| /usr/local/products/mh/current/lib/rcvtty"
```

3) Enter **`ls -lL `tty``** to see your default tty setting. If `o+w` (that is, if *write* permission for *other*; see section 6.6.1) is not set, use the command **`chmod o+w `tty``** to set it (again, best to include in `.login` or `.profile` file).

Customizing rcvtty Output

You can reformat the output of **rcvtty** if you like. To do this you need to create a form file describing the syntax of the output you want to see, and you need to modify the **`rcvtty`** command. For example, if you wish to use the same format as `scan-form` (see section F.2.5) the form file used for incorporating or scanning mail, the line in `.forward` would look like this (shown in the format used when **biff/comsat** is *not* running):

```
"| /usr/local/products/mh/current/lib/rcvtty -biff -form scan-form"
```

F.2 Customizing mh and exmh

Your personal configuration for **MH** is primarily defined via the `.mh_profile` file. It defines, among other things, where mail is stored and what editor to use. Some additional files are available if you wish to customize further. These were listed in section 12.4.

Each **MH** mail composition command, (**`comp`**, **`repl`**, **`forw`** for **mh**; Send, Reply, Forward for **exmh**) uses its own template file to define the message header format:

- **`comp`** or Send uses the `components` file
- **`repl`** or Reply uses `replcomps`
- **`forw`** or Forward uses `forwcomps`

If you don't have these template files in your `$HOME/Mail` directory the system default files in the **MH** library are used.

Options for incorporating mail were discussed in Chapter 12. The `scan-form` and `inc-form` files provide the format of the display when you scan and incorporate mail. You can control how local delivery is performed by creating a `.maildelivery` file.

F.2.1 .mh_profile

It also allows you to specify options to **MH** commands. If you have run **`setup mh`** or **`setup exmh`**, you already have a simple one in your home directory. You can tailor your `.mh_profile` file to suit your taste. Following are some of the extra lines you can add:

`Path: Mail`	Directory for your **MH** transactions.
`Unseen-Sequence: unseen`	Name of the unseen sequence of messages.
`Draft-folder: drafts`	Name of the default folder for drafted, unsent mail.
`Editor: emacs`	Sets up **emacs** as editor for **`comp`**, **`repl`**, **`forw`**, **`dist`** (use any editor you want).[1]
`rmmproc: /bin/rm`	Causes mail to be truly deleted when requested (see section 12.3.8). **Recommended!**
`Aliasfile: mh_alias`	File containing **MH** aliases (distribution lists).
`Scan: -form scan-form`	Provides format of display when you scan mail.
`inc: -form inc-form`	Provides format of display when you incorporate mail.

1. **exmh** ignores any editor defined here (defining an editor is optional for **MH**)

`Signature: John Peoples` Your signature. Appears in `From:` header field.

The `Path` is taken to be relative to your `$HOME` directory. The files/directories used for `Draft-folder`, `Aliasfile`, `Scan`, and `inc` are assumed to be relative to the directory specified by `Path`.

F.2.2 components

The default template file for composing (sending) a message is called `components` and looks like:

```
To:
cc:
Subject:
-------
```

You can add other fields to the header by making your own `components` file. For example, an enhanced `components` file may include the additional fields shown below:

```
Reply-to: {username}@fnal.gov
To:
cc:
Bcc:
Fcc: +yourfolder
Subject:
-------
```

The `Reply-to:` field should give your FNAL address. It is not needed if you want the reply to come to the node from which you are sending the message. The **`repl`** command in **MH** (and some other mail agents as well) will route replies to this address rather than the `From:` address.

The "blind carbon copy" `Bcc:` field can be used to send copies of the message to addresses that are not seen by the people specified in the `To:` and `cc:` fields. This is like forwarding a copy of the message to someone else after sending it, except that here you can do it at the time you send the original message. `Bcc:` can also use distribution lists or aliases.

You can send a copy to yourself by specifying a folder for your copies in the `Fcc:` field. *Fcc* stands for folder copy. This copy goes directly to the specified folder, and only to that folder.

F.2.3 replcomps

The `replcomps` file controls the header in replies. The default file is given below:

```
%(lit)%(formataddr %<{reply-to}%?{from}%?{sender}%?{return-path}%>)\
%<(nonnull)%(void(width))%(putaddr To: )\n%>\
%(lit)%(formataddr{to})%(formataddr{cc})%(formataddr(me))\
%<(nonnull)%(void(width))%(putaddr cc: )\n%>\
%<{fcc}Fcc: %{fcc}\n%>\
%<{subject}Subject: Re: %{subject}\n%>\
%<{date}In-reply-to: Your message of "\
%<(nodate{date})%{date}%|%(pretty{date})%>."%<{message-id}
               %{message-id}%>\n%>\
--------
```

The first two lines in the file build the `To:` address field for your reply draft. It takes the first available address by searching the `Reply to:`, `From:`, `Sender:` and `Return-path:` fields, in that order. Lines 3 and 4 build a `cc:` address field by taking all the `To:` and `cc:` addresses from the original message and including your address as well. It then builds an `Fcc:` field by testing a **`-fcc`** switch on the command line or in the `.mh_profile`. The `subject:` field is taken from the subject in the incoming mail preceded by `Re:`. The last three lines make up the field that says "*In-reply-to: Your message of* ..." with the date and message number at the end. The default reply header takes the form:

```
To:
cc:
Subject:
In-reply-to: Your message of "Day, date time"
             <message-id>
--------
```

You can customize this formatting by modifying your own `replcomps` file (note that no spaces are allowed after the backslash (****) on each line). You will most likely need to reference the O'Reilly **MH** text for this task.

As you now know, `replcomps` controls the format of the message header. You can also customize the format of the message body for replies via a file called `.mh_filter` in your home directory. The O'Reilly **MH** book describes the different possible ways of setting this up. The contents of the default `.mh_filter` is:

```
body:component="> ",compwidth=0
```

F.2.4 forwcomps

The `forwcomps` file is used as the template header for forwarded messages. This file may look very similar to the `components` file, in fact the default files are identical. If you create your own `components` file and you want to use it for both sending and forwarding, include the following line in your `.mh_profile` file:

```
forw: -form components
```

You would only need a separate `forwcomps` file if you wanted the headers on forwarded and original mail to be different.

F.2.5 scan-form and inc-form

These files provide the format of the display when you scan and incorporate mail. If you aren't satisfied with the default scan format, we recommend that you simply set up a `scan-form` file, and use it for both situations (set both `Scan:` and `inc:` to `scan-form` in `.mh_profile`). The `scan-form` file's format is also used by **rcvtty** to provide messages. An example of a simple `scan-form` file showing incoming mail only and set up in a VMS-style format is shown below:

```
%4(msg)\
%21(addr{from})\
%02(mday{date})-%3(month{date})-%4(year{date})\
%02(hour{date}):%02(min{date})\
%{subject}
```

This example produces output in the format:

```
msg# sender_address  date  time  subject
```

Some sample output follows:

```
18 helpdesk              13-Nov-1995 16:30 1017699 Problem Report
19 lauri                 14-Nov-1995 10:56 VY0124 damaged
20 STUART@fnald.fnal.gov 14-Nov-1995 11:28 Problems with an fnaldb tape drive
21 wolbers@fsui01.fnal.g 14-Nov-1995 15:31 Department Meeting
```

An slightly more complicated example is shown below. It allows display of both incoming and outgoing messages:

```
%4(msg)\
%<(cur)+%| %>\
%<{forwarded}F%| %>\
%<{replied}R%| %>\
%<(mymbox{from}) <- %19(friendly{to}) %| -> %20(addr{from})%> \
 %02(mday{date})-%03(month{date})-%4(year{date})\
 %02(hour{date}):%02(min{date})\
 %{subject}
```

This example produces output in the format:

```
msg# (arrow) (sender or recipient address)  date  time  subject
```

The arrow indicates whether the mail is from you (<−) or to you (−>). If it is from you, the address field shows who you sent it to. Sample output looks like:

```
258    -> bhat                      14-Oct-1995 12:15 batch job 341: <run_small> D
260    <- joe@fnal                  13-Oct-1995 17:24 Program and Working Groups -
261    <- lhn@sld.slac.stanford.edu 13-Oct-1995 16:31 e+w- Workshop, November 16-1
262    <- martha@fnal               13-Oct-1995 17:24 re: revised mail chapter
263    -> STROVINK@d0sfa            16-Oct-1995 09:21 Re: FYI re directions in NN
```

F.2.6 .maildelivery

The `.maildelivery` file controls how local delivery is performed. This file can be quite simple, or very complex. Within the file you can configure options to:

- restrict notification to mail received only from certain senders
- control notification format and content
- specify further processing such as delivery to one or more specific folders
- start up certain processes depending on information in the mail message header

☞ Note that if you use the **mh** command **`inc`** (at the shell prompt) instead of incorporating within **exmh**, your mail will not be sorted. It will be filed into your `inbox` folder. The `.maildelivery` file is only used when you 1) incorporate within **exmh** (described in section 12.4.3) and 2) check the Presort box in the Incorporate Mail window under Preferences. The `.maildelivery` file is also used with unattended autoincorporation (see section F.4).

Simple Case: Send All Incoming Mail to inbox Folder

If you simply want to send all incoming mail to your `inbox` folder, your `.maildelivery` file needs only to consist of the following one line:

```
* - ^ A "/usr/local/products/mh/current/lib/rcvstore +inbox"
```

If you want to do something more sophisticated, continue reading this section.

File Format

Lines starting with `#` are comments. Fields are separated by one or more spaces. Each record in the file has the following fields in this order:

header field	the message header field that you're sorting on (e.g., `from`, `to`, `subject`)
match pattern	the text string found in the header field that you want to match (enclosed in quotes if it includes spaces). If the header field argument is `default` or `*`, this should be `-` (a dash).
action	what to do if pattern is matched. Set to `file` or `>` to send the matched message to a file. Set to `qpipe` or `^` to directly execute a command without invoking a Bourne shell. Set to `pipe` or `|` to invoke a Bourne shell to execute the command. If your command has special characters in it, you will need to use `pipe` or `|`, otherwise `qpipe` or `^` is more efficient.
result	a code indicating whether to set matched message to "delivered", or not. `A`: if field and pattern matched, action will be performed, and if successful, then set message to "delivered". `R`: same as A, but don't set message to "delivered" so that further entries in `.maildelivery` can act on it. `?`: perform the action if message not yet set to "delivered" by a previous entry. If action succeeds, set message to "delivered".
file or command	depending on action, this is either a file (if action says "send message to a file"), or a command (if action says "execute a command"). Always include full pathname.

Note that some **MH** systems expect this file to have the mode 644 (**`u+rw`**, **`g+r`**, **`o+r`**).

Examples

The following examples should provide you enough information to modify your `.maildelivery` file to suit your tastes. We'll start with an annotated version of the simple `.maildelivery` file discussed above in which all your incoming mail gets placed in `inbox`:

```
# execute rcvstore to put all as-yet-undelivered incoming mail into inbox (sets messages
# to "delivered")
#hdr  pattern  action  result  command
 *    -        ^       A       "/usr/local/products/mh/current/lib/rcvstore +inbox"
```

Here is a slightly more complex example. The comments explain what is being done in each line:

```
# File mail with fsui in the To: line into file $HOME/fsui.log
To fsui file A fsui.log

# Pipe messages from daemon@fnck to the (local) program err-message-archive
From daemon@fnck pipe A err-message-archive

# Put anything with Sender: address "mh-users" into $HOME/mh.log if not yet filed
Sender mh-users file ? mh.log

# If the address is "uas-admin", send an acknowledgment copy back
addr uas-admin ^ A "/bin/resendR-r $(reply-to)"

# Destroy anything from daemon@fnck!
From daemon@fnck destroy A -

# For anything not matched yet, execute rcvstore to put into inbox folder
* - ^ ? "/usr/local/products/mh/current/lib/rcvstore +inbox"
```

If you plan to create a significantly different set of criteria, you may need to see the man pages for **slocal** or the O'Reilly **MH** text referenced at the start of this appendix.

F.3 Automatic Reply to Incoming Mail

There is a handy UNIX program called **vacation** that you can use to send an automatic reply to emails that arrive while you're away. To enable **vacation**, enter simply:

`% vacation`

```
This program can be used to answer your mail automatically when you go away on vacation.
You need to create a message file in
/afs/fnal.gov/files/home/room3/aheavey/.vacation.msg first.
Please use your editor (emacs) to edit this file.
```

The editor invoked is determined by either the *VISUAL* or *EDITOR* environment variable. The default otherwise is **vi**. The default message is displayed in the editor window for you to edit. When you finish, the system returns:

```
You have a message file in /afs/fnal.gov/files/home/room3/aheavey/.vacation.msg.
Would you like to see it?
```

If you answer **`y`**, the file contents gets typed to the screen, and you are given the opportunity to re-edit the file. If you choose not to re-edit it, the system goes directly to:

```
You have a .forward file in your home directory containing:
(file contents displayed here)
Would you like to remove it and disable the vacation feature?
```

Respond **`n`**. At this point the dialog ends, and you need to go to your `.forward` file and add the line:

`\{username}, "|/usr/bin/vacation {username}"`

where `{username}` is your username, for example:

`\aheavey, "|/usr/bin/vacation aheavey"`

Now the message in `.vacation.msg` will get sent in reply to all subsequent incoming email messages.

To disable **vacation** when you return, simply remove the above line from your `.forward` file[1].

Run **`man vacation`** to get the full documentation.

F.4 Unattended Autoincorporation

It is possible, though not recommended in a *clustered* environment (see section 2.7 about clusters), to use the `.forward` file to automatically deliver your mail from the system mail spool area directly into your mail area, whether or not you are reading mail, or whether or not you are even logged in! This is called *unattended autoincorporation*. It has a number of attractive features:

- since mail doesn't really stop in the system spool area, but rather is delivered directly to your `inbox` file, you are no longer restricted to reading and receiving mail on one particular node of a cluster
- reduces the startup time (and de-iconizing time) of **exmh** because it doesn't have to "catch up" on lots of spooled messages
- mail can be automatically filed to specified folders as it is being delivered (otherwise only possible for **exmh**, not for the line-mode **mh**)

However, unattended autoincorporation also has a number of drawbacks and caveats which make it a less attractive option:

- in a clustered environment where several separate nodes (hence several separate mail spool areas) see the same mail destination `inbox` file, file locking can become a serious problem. It can happen that a deadlock results in *no* mail being delivered while an overload of processes attempt to deliver mail in vain. This problem is especially serious for people who receive a lot of mail, and is exacerbated by the use of **rcvtty** (mail notification; see section F.1.2).
- **exmh** does not update the visual display as new mail gets autoincorporated. Therefore, while you may have saved the time of incorporating at startup, you will need to manually Rescan all Folders (under the More... button in the middle window) in order to see that you have new mail (and in which folders) when you are running **exmh**.

Because of the detrimental effect that unattended autoincorporation can have on a system, and the fact that when it breaks, mail is not delivered, **we do not recommend its use in most circumstances**. In particular, we believe that it should not be used in clustered environments. It is documented here for completeness only, and should not be taken as an endorsement.

To configure unattended autoincorporation on a node:

1) create a `.maildelivery` file describing your mail sorting preferences (see section F.2.6)

2) remove the line of your `.forward` (generally the first line) which contains something similar to `\{username}@{node}.fnal.gov`

3) replace it with `"| /usr/local/products/mh/current/lib/slocal -user {username}"` (include the double quotes but not the braces)

where `{username}` is your login id.

F.5 Unattended Autoincorporation in AFS Environment

Proper ACL settings (AFS directory permissions, discussed in section 7.6.2) must be made for autoincorporation in **MH**. The easiest way to set your mail folder ACLs is to run the utility **fixmailperms** from your `$HOME` directory. This allows the mail folders that you've designated

1. Some OS versions and **sendmail** versions support comments in `.forward`. You can test to see if yours does. If it does, you can leave the vacation line in the file, and just comment it out with a pound sign (#) when you don't need it.

for receiving mail to work with **slocal**, and it sets *all* your mail folders to be readable only by your account and the system administrator's. **fixmailperms** is part of the **UPS** product **mailtools**. To see a brief explanation, enter:

`% fixmailperms -help`

The command is entered as:

`% fixmailperms [`*folder_1 folder_2* **. . .** *folder_n***`]`**

The folder specifications are taken as relative to your `Mail` subdirectory. You need to specify all the folders into which you plan to automatically incorporate mail, and only those folders.

If you do not specify a minimum of one folder when you execute **fixmailperms**, unattended autoincorporation will not work.

For example, if you have a simple setup in which all incoming mail gets sent to your `inbox` folder, you'd enter:

`% fixmailperms inbox`

If this is satisfactory to you, skip to the next section. If instead you choose to set your ACLs individually (and not set your folders private), these are the ACLs you will need:

`system:anyuser rl`	for `$HOME`
`system:anyuser l`	for `Mail`
`system:anyuser li`	for `Mail/{folder}`[1]

Go to the directory whose ACL you want to change, and use the command syntax:

`% fs setacl . system:anyuser` ***xy***

where ***xy*** is **`rl`**, **`l`**, or **`li`**, depending on the directory. Run **`fs listacl`** to see the current setting.

1. This folder should be your *inbox* or equivalent, e.g., `Mail/inbox`. You may need to set this for several folders, depending on the complexity of your autoincorporation set up.