Majordomo and Mailing List Administration

About Majordomo

The Outpost 10F server makes use of the Majordomo mailing list and administration software. Majordomo is an automated system that facilitates email lists and email list management on the server. Through using majordomo, users are allowed to subscribe and unsubscribe to mailing lists, and to retrieve files from list archives.

There are two ways you can manage your OTF Majordomo mailing lists:

Cobalt Management System

The Cobalt site management system enables site managers (managers of admin.outpost10f.com for example) to create and manage majordomo mailing lists through a graphical user interface in a web browser. This method is clearly the easiest, however, the online interface does not provide access to all the commands and tools offered by Majordomo. To view more information on how to manage your mailing lists this way, visit the cobalt site management manual.

Email Commands

In order to access all commands and tools provided by Majordomo, you can interact with the Majordomo software by sending it commands in the body of email messages addressed to Majordomo@*hostname*.outpost10f.com where *hostname* is the OTF host that holds the list of interest.

The Configuration File

Each mailing list has a configuration file stored on the server that defines how the list operates. There are many options or *directives* in this file one can manipulate in order to change the nature of the mailing list. One can use email commands to request this file and replace it with a modified copy. An explanation of all the directives in the configuration file can also be found in this document.

Using Email Commands

As mentioned, you can interact with the Majordomo software by sending it commands in the body of mail messages addressed to Majordomo@*hostname*.outpost10f.com where *hostname* is the OTF host that holds the list of interest. Please do not put your commands on the subject line; Majordomo does not process commands in the subject line.

You may put multiple Majordomo commands in the same mail message. Put each command on a line by itself.

If you use a "signature block" at the end of your mail, Majordomo may mistakenly believe each line of your message is a command; you will then receive spurious error messages. To keep this from happening, either put a line starting with a hyphen ("-") before your signature, or put a line with just the word

end

on it in the same place. This will stop the Majordomo software from processing your signature as bad commands.

For example, if you wanted to read the information file for a mailing list called `unions` located on www.outpost10f.com, you could send the following email:

```
To: Majordomo@www.outpost10f.com
From: jim@outpost10f.com
Subject: List Information

info unions
```

Majordomo would then automatically reply to this email with the information you requested. (Note that the To:, From:, and Subject: lines are email headers; the only thing you would type in the body of your email is `info unions`).

The various user and list owner commands will now be discussed.

User Commands

General Commands

General commands can be used by anyone, whether or not they are subscribed to a list, though some lists may restrict these commands to subscribers:

`help`

> Request a help message listing commands and the version of Majordomo that's in use.

`lists`

> Request the list addresses and descriptions of lists served by Majordomo. Lists may be concealed from this request.

`info` ***`listname`***

> Request the list's information file. The information file contains a public description of what the list is for.

`who` ***`listname`***

> Request a list of subscribers' email addresses. On some lists, only subscribers may issue this request.

`end`

> Causes Majordomo to ignore the rest of the message. A line that starts with a dash (-) also stops Majordomo from reading the rest of the message.

Subscriber Commands

Subscriber commands are either used to subscribe to the list or are restricted to list subscribers:

`subscribe` ***`listname [address]`***

> Subscribe to mailing list *`listname`*. If *address* is given, subscribe that email address instead of the sender's address (which may require list owner approval).

`unsubscribe` ***`listname [address]`***

Unsubscribe from a list. If *address* is given, remove that email address from the list (which may require list owner approval).

which *[address]*

Request to know to which lists you are subscribed. If *address* is given, request to know to which lists *address* is subscribed.

Archive Commands

Archive commands let users search list archives and retrieve files. Note that as of this point in time archive support has not been enabled on the Outpost 10F server. These commands may be restricted to list subscribers:

index *listname*

Request an index of the files in *listname*'s archive.

get *listname file*

Request the file *file* from *listname*'s archive.

List Owner Commands

List owner commands let the list owner control the configuration of the list. They require the list password:

config *listname list-password*

Request a copy of the list's configuration file. The configuration file is what regulates the entire list and contains a handful of variables that you might find useful. A description of these variables can be found in Appendix I.

newconfig *listname list-password*

Replace the list's configuration file with a new file. The new file must appear below the `newconfig` command and be followed by a line containing EOF.

newintro *listname list-password*

Replace the list's intro file with a new one. The intro is what a new subscriber is automatically sent by Majordomo. It serves the purpose of welcoming the new member and perhaps setting out some groundrules. The new file must appear below the command and be followed by a line containing EOF.

newinfo *listname list-password*

Replace the list's info file with a new one. The info file contains information about the mailing list and tells those interested what it is for. The new file must appear below the command and be followed by a line containing EOF.

writeconfig *listname list-password*

Rewrite the list's configuration file to include all the comments and variables in the distributed configuration file. Useful when upgrading from earlier versions of Majordomo.

password *listname current-backup-password new-backup-password*

Change the backup list password.

mkdigest *listname list-password*

Issue a digest immediately for a digested list. Digests are currently not enabled on the Outpost 10F server.

approve *list-password* subscribe *listname email-address*

Subscribe an email address to the list.

approve *list-password* unsubscribe *listname email-address*

Unsubscribe an email address from the list.

approve *list-password* who *listname*

Get a list of subscribers to a list, even if the who request is disabled for that list.

Config File Directives

As mentioned, list owners configure their lists by requesting a copy of the list's configuration file, *listname.config*, editing the file, and mailing it back to Majordomo.

The configuration file is nearly all valid Perl. Lines in the configuration file that begin with a pound sign (#) are comments. Other lines are called *directives* and define the value of list options. Most directives take string values, numbers, or filenames, and are in the format *option = value*. Others take lists of strings or regular expressions and are defined like this:

```
Option << END
Values are listed here, one per line or this might be a paragraph value.  To include a
blank line, use a line with a single hyphen:

-

Normally, multiple spaces     like this are treated as a single space.  If it's important to
preserve them, start the line with a hyphen:

-1    2    3    4    5

To get a single hyphen at the beginning of a line, double it:

-- is a hyphen

the definition endes with the line END
END
```

All of the directives used in the configuration file will now be discussed. Note however that only a few of them will be of value at OTF. These include:

- announcements
- advertise
- noadvertise
- description
- message_fronter
- message_footer
- message_header
- moderate
- moderator
- mungedomain
- reply_to
- restrict_post
- subject_prefix
- subscribe_policy

- unsubscribe_policy
- welcome

admin_passwd

The administrative password must accompany administrative commands from the list owner. Each list has two administrative passwords. One is stored in the file *listname.passwd*. The other is given in the configuration file by defining admin_passwd. admin_passwd defaults to *listname.admin* and should definitely be changed.

administrivia

If administrivia is set to yes, suspected administrative requests sent to the list address are forward to the list owner; setting it to no disables request filtering. The default value is yes. The variables $admin_headers and $admin_body in *majordomo.cf* control which patters are considered to be administrative requests.

advertise and noadvertise

The advertise and noadvertise options control the visibility of the list in response to a lists request. These options take arrays of regular expressions as values, like this:

```
advertise << END
/@.*.myhost.com/
/sarah@friendly.org/
END
noadvertise <<END
END
```

Each line in the list is a regular expression, enclosed by slashes. Lists are always visible to their subscribers. For nonsubscribers, if the email address of the sender of a lists request matches a regular expression in noadvertise, the list doesn't appear in Majordomo's response. If not, the list appears if the sender's address does match a regular expression in advertise. Either or both of these options may be left undefined by including no regular expressions between the << END and END. Undefined options are effectively disabled.

In the example above, the list is only visible to requests which come from *.myhost.com* addresses or from *sarah@friendly.org*. To make a list totally invisible, define noadvertise like this:

```
noadvertise << END
/./
END
```

announcements

If announcements is set to yes, the list maintainer receives notification whenever someone subscribes or unsubscribes from the list. If announcements is set to no, no notifications are sent.

approve_passwd

If a list is moderated, its moderator approves messages by resending them to the list with an Approved: *password* header. approve_passwd sets the password for message approval. It defaults to *listname.pass* and should be changed.

comments

The comments option is a place to put comments into the configuration file that are preserved even when the file is rewritten. Its value is a paragraph:

```
Comments << END
This configuration file is maintained by jerry
END
```

date_info and date_intro

If date_info is set to yes, the date that the list's info file was last updated with the newinfo command is placed at the top of the info file. If set to no, the date isn't placed in the file (though it's still included in the response to an info request). Placing the date in the file is handy if the file might be looked at in some other way than an info request. date_info defaults to yes. date_intro performs the same function for the list's intro file.

debug

Setting debug to yes prevents the *resend* program from actually distributing list messages; instead it sends the list owner a message containing the commands that it would have executed. The default is no.

description

description is a single string that describes the mailing list in Majordomo's response to the lists request.

digest_volume and digest_issue

If the mailing list is digested, digest_volume gives the next volume number; digest_issue is the next issue number. digest_issue is automatically updated as each digest is produced. digest_volume must be updated manually when a new volume should be issued. These directives are currently not of any value on the OTF server as digesting is not enabled.

digest_maxdays and digest_maxlines

These options control how many days can pass between digests and how long (in lines) a digest can grow before being distributed. See also maxlength, which controls digest size in characters. These directives are currently not of any value on the OTF server as digesting is not enabled.

digest_name

If the mailing list is digested, the digest's subject line contains the value of digest_name, followed by the volume and issue numbers. It defaults to the name of the mailing list. This directive is currently not of any value on the OTF server as digesting is not enabled.

digest_work_dir

You can define digest_work_dir to direct Majordomo to build digests in a particular directory; if undefined, Majordomo uses the *listname* subdirectory in the *digests* directory. Generally, you shouldn't define this. This directive is currently not of any value on the OTF server as digesting is not enabled.

get_access

This option controls who can use the `get` command to retrieve a file from the list's archive. Three values are possible: `open` allows anyone to issue a `get` command, `list` only allows list members, and `closed` prevents anyone from using the command. Actually, the list's owners can override `closed` access by using the `approve` command. For example:

```
Approve listname password get filename
```

A *command_access* option is available for the commands `get`, `index`, `info`, `intro`, `which`, and `who`. This directive is currently not of any value on the OTF server as archiving is not enabled.

maxlength

Sets the maximum number of characters a message may contain in order to be distributed without approval. It's also the size at which digests are automatically distributed if the list is digested. The default value is 40000.

message_fronter, message_footer, and message_headers

These options allow you define a paragraph of text that is inserted in the headers (`message_headers`), at the beginning of the body (`message_fronter`), or at the end of the body (`message_footer`) of each message or digest distributed to the list. In addition to static test, certain special strings are automatically expanded by majordomo in useful ways:

`$LIST`
: Expands into the name of the mailing list. Only undigested lists may use `$LIST`.

$SENDER
: Expands into the sender's address from the message's From header. Only undigested lists may use $SENDER.

$VERSION

Expands into the Majordomo version number. Only undigested lists may use $VERSION.

SUBJECTS

Expands into a table of contests for a digest. It may only be used in the message_fronter of a digested list.

Majordomo discards anything after two blank lines in these options, so be sure to leave only single blank lines between paragraphs.

moderate

When moderate is set to yes, the list is moderated. All messages sent to moderated lists are forwarded to the moderator(s) for approval. Default is no.

moderator

If moderator is set to an email address, messages requiring approval are bounced to that address, rather than the list owner.

mungedomain

Majordomo allows users to subscribe themselves at addresses that differ from their From address, but requires list-owner approval in the most common configuration (see the description of subscribe_policy below). If mungedomain is set to yes, users whose mail comes from *user@host.domain.xxx* may subscribe the address *user@domain.xxx* without approval. Because such situations are increasingly common, this can be a valuable time saver. Its value defaults to no.

precedence

The precedence option, if defined, sets the value for the Precedence header for outgoing messages. It defaults to bulk, which prevents vacation programs from responding to the list, but also instructs *sendmail* to discard messages that bounce rather than send them back to Majordomo for processing.

purge_received

If `purge_received` is set to `yes`, *resend* removes all Received headers before distributing messages to the list. This can produce headers that are shorter and more uniform, and makes it easier to conceal the identity of message senders, which may or may not be desirable. It also defeats *sendmail*'s built-in loop detection feature that rejects messages with too many Received headers.

`reply_to`

The `reply_to` option, if defined, sets the value for the Reply-To header of outgoing messages. To set the Reply-To header to match the From header, set this option to `$SENDER`. Digested lists with undigested counterparts use this option to direct replies to the disgested list back to the undigested list. This option is undefined by default, which results in no Reply-To header on outgoing messages. It is recommended for OTF purposes, that this directive be set to the mailing list itself so that replies are sent back to everyone instead of the sender.

`resend_host`

`resend_host` sets a hostname that is appended to addresses used by *resend*. Alternatively, *resend* can be called with a –h *host* argument. This option is undefined by default.

`restrict_post`

The `restrict_post` option may be defined as a list of files (separated by spaces or colons) that are checked for addresses of approved message senders. Only senders whose addresses match a line in one of the listed files may send messages to the list. Because remote list owners generally can't access these files, however, this option has limited utility, and its comment in the *config* file says that it will be replaced in the future. If you wish to use this feature, please send an email to Maxwell@outpost10f.com.

`sender`

The `sender` option sets the SMTP From address and Sender header for outgoing messages. *@hostname* is appended, so only a local alias name is used. `sender` defaults to `owner-`*`listname`*.

strip

Most people's From headers include more than just their email address. Often their full name or other comments appear in the From: header. If the `strip` option is set to `yes`, these comments are stripped off when new subscriber addresses are added to the list's subscriber file. The option was originally added because the *smail* MTA objected to comment sin subscriber files. `strip` defaults to `yes`.

subject_prefix

If defined, the `subject_prefix` is prepended to the subject line of outgoing list messages unless it already appears in the message subject. The `subject_prefix` can only be a singe word. The expansion strings used in `message_footer`, `$LIST`, `$SENDER`, and `$VERSION` are also available; setting `subject_prefix` to `$LIST` is often valuable.

subscribe_policy

A particularly important configuration option, `subscribe_policy` controls who may subscribe to the list without the approval of the list owner. It can take on one of these values:

auto

Means that anyone can subscribe to the list without approval. Further, third-party subscriptions are allowed; anyone can ask to subscribe an address different than that in their From header.

auto+confirm

Works like `auto`, but Majordomo sends an authentication number to the subscriber, which he must send back to Majordomo to complete the subscription. This prevents people from subscribing others unwittingly and protects the list from addresses with typos in them.

open

Allows anyone to subscribe themselves to the list without approval. Attempts to subscribe other addresses require approval.

open+confirm
: Works like open, but also requires confirmation.

closed
: Requires list owner approval for every subscription request.

closed+confirm
: Works like closed, but also requires confirmation.

taboo_body and taboo_headers

These options help protect lists against unwanted mailings. If a message's headers match any regular expression listed in taboo_headers, or its body matches any regular expression listed in taboo_body, the message is sent to the list owner or moderator for approval. This provides great flexibility in keeping junk mail, abusive posters, or users at rogue sites from disrupting your list.

unsubscribe_policy

unsubscribe_policy controls who may unsubscribe from the list without the approval of the list owner. Its values are similar to those of subscribe_policy:

auto
: Means that anyone can unsubscribe from the list without approval. Further, third-party unsubscriptions are allowed; anyone can ask to unsubscribe an address different than in their From header. This is probably too liberal for most mailing lists.

Open
: Allows anyone to unsubscribe themselves from the list without approval. Attempts to unsubscribe other addresses require approval. This is the default.

closed

Requires list-owner approval for every unsubscripion. This is rarely used.

`welcome`

If the `welcome` option is set to no, welcome messages aren't sent to new subscribers. The default is `yes`.

Unused Options

The options `archive_dir`, `digest_archive`, `digest_rm_footer`, and `digest_rm_fronter` are included in the configuration file but are not operational and shouldn't be defined.