EZMLM/IDX MANUAL, Version 0.01

(c) 1998, Fred Lindberg, lindberg@id.wustl.edu 20-JUN-1998

EZMAN-0.01, a manual for ezmlm and ezmlm-idx. It is intended for non-technical users. For technical information, please see documentation in the ezmlm-idx package, especially the FAQ and the man pages. Features available only with ezmlm-idx are marked(*). The manual has 3 chapters. Chapter 1 is a ezmlm/idx user's manual. It is intended for people (wanting to be) subscribed to ezmlm mailing lists. It describes, how to subscribe, unsubscribe, post, and access the list archive. Chapter 2 is an ezmlm/idx moderator's and list administrator's manual. It is an extension of chapter 1 for people who function as moderators or remote administrators of ezmlm mailing lists. Chapter 3 is a ezmlm/idx list owner's introduction to ezmlm. It is an extension of Chapters 1 and 2 for ezmlm list owners. It describes ezmlm list types, and what is possible with ezmlm/idx lists, and how to deal with common problems. The owner who wants to learn how to control the details of list customization should consult the documentation in the ezmlm and ezmlm-idx packages, especially the FAQ. The appendices give references to other sources of ezmlm information, a summary of ezmlm/idx commands, and a brief discussion of a user/host-wide interface that can be used to mimic other MLMs to make migration easier. Many thanks to Toshinori Maeno for helpful comments/suggestions.

Contents

Chapter 1

EZMLM/IDX USER'S MANUAL

1.1 Introduction.

This chapter helps the (potential) subscriber to ezmlm/idx lists to learn the basic functions of the mailing list manager. It contains information on how to start and stop subscribing to ezmlm/idx list, how to access the list archive, how to send messages to the list, and how to contact the list owner, if necessary.[1]

Ezmlm is a modern mailing list manager. Its purpose is to efficiently send a message to a large number of recipients with minimal delay. It allows automated additions and subtractions from the subscriber database. In addition, it may keep an archive of messages. It can also impose restrictions on what may be sent or retrieved and by whom. Some list manager keep a database of subscriber information and tailor the message specifically for each subscriber. Ezmlm sends the same message to all subscriber. *Digests*(*), ordered collections of messages, are handled as a separate list. The only information ezmlm keeps about you is your subscriber address, and if applicable, which messages that couldn't be delivered to you due to e.g. problems with your mail server.

Ezmlm is highly automated. Virtually everything can be done simply, rapidly, and without human intervention. Ezmlm encodes your address and the message number in the return address to automatically handle *bounces*, i.e. messages returned to sender because they couldn't be delivered to the subscriber. This can be used to find out the message number, as well as the address you're subscribed under (see below).

1.2 Sending commands to an ezmlm/idx mailing list.

Ezmlm has a different address for each command. In essence, the address is the list address *and* the command. This is straight forward to interpret and therefore very fast. It also makes it very easy to send commands from the command line without having to worry about the subject and the message body.

If you want to interact with the ezmlm mailing list `mailinglist@comp.org`, send an empty message

[1]Features marked "(*)" are available only if the list uses ezmlm-idx in addition to ezmlm. See A.1 for more information on ezmlm and ezmlm-idx.

(contents and subject are ignored) to `mailinglist-command@comp.org`, replacing "`mailinglist@comp.org`" with the list address, and "command" with the desired command. For some commands, you can add arguments. Here are a few examples, which will be discussed further below:

`mailinglist-subscribe@comp.org`

Subscribe me.

`mailinglist-subscribe-user=gol.com@comp.org`

Subscribe `user@gol.com`. Here the default subscriber address (the one you send from) is overridden by an argument to the *-subscribe* command. To subscribe any address of type `user.name@host.dom` just replace the "@" in the address with "=" and add it after a hyphen to the command. In this case `mailinglist-subscribe-user.name=host.dom@comp.org`. Of course, ezmlm will ask `user.name@host.dom` to confirm the request.

`mailinglist-unsubscribe@comp.org`

Unsubscribe me.

`mailinglist-unsubscribe-user=gol.com@comp.org`

Unsubscribe `user@gol.com`.

`mailinglist-query@comp.org(*)`

Tell me if my address is subscribed to the list or not.

`mailinglist-get.123@comp.org`

Get message 123 from the archive.

`mailinglist-get@comp.org(*)`

Get the latest message from the archive. If the list has a digest, the messages returned are the messages received since the latest digest.

`mailinglist-get.45_67@comp.org(*)`

Get messages 45-67 from the archive. No more than 100 messages can be returned per request.

`mailinglist-index@comp.org(*)`

Send subject and author name for the last 100-200 messages. No more than 2000 entries are returned per request.

`mailinglist-index.300_399@comp.org(*)`

Send subject and author of messages 300-399.

`mailinglist-thread.45@comp.org(*)`

Receive an ordered set of all messages that have the same subject as message 45. "Re:", etc, do not make a difference, so the returned set is the usually entire thread.

`mailinglist-info@comp.org(*)`

Send me some information about this list (the list owner decides how this command is supported).

`mailinglist-faq@comp.org(*)`

Send me the list of frequently-asked questions (and answers) for this list (the list owner decides how this command is supported).

1.3 Subscribing to an ezmlm mailing list.

As mentioned previously, ezmlm commands are put into the ADDRESS. To subscribe to `mailinglist@comp.org`, send an (empty) message to `mailinglist-subscribe@comp.org`. Ezmlm will send a confirmation request back to you. To confirm your subscription, just reply to the "Reply-To:" address. Usually, the "reply" function of your mail program will do this. When ezmlm receives your confirmation, it will add your address to the subscriber database and send you a message to tell you that you are a subscriber. If your mail program supports it, you may be able to just click on the address in the message.

The confirmation is used so that only you (and the list administrator) can add/remove your address from the list of subscribers.

Sometimes, you may want to subscribe an address other than the one you are sending from. Let's say you're sitting at `john@home.net` and that all your mail is forwarded here from `harold@work.com` (when you are not in the office). You'd like `harold@work.com` to be the subscriber (so that you get the messages at work). To do this, send mail to: `mailinglist-subscribe-harold=work.com@comp.org`. Ezmlm will send the confirmation request to `harold@work.com` which will be forwarded and reach you at `john@home.net`. When you reply, `harold@work.com` is added as a subscriber. The ezmlm verification mechanism tests the only relevant issue: That the person at the subscriber address really wants to be a subscriber. After the confirmation reply is received, ezmlm will send a message to the subscriber address to let the recipient know that s/he is a subscriber.

Some ezmlm mailing lists may use subscription moderation(*). Subscription to these lists is identical to the process described above, *except* that after confirmation the request is passed on to the list moderator(s). You become a subscriber only after one of the moderators has approved your request. If you want to send any additional information to the moderator, do so in your confirmation message. Ezmlm will notify you once you are a subscriber.

1.4 Unsubscribing from an ezmlm mailing list.

You've guessed it already! To unsubscribe from `mailinglist@comp.org` just send a message to `mailinglist-unsubscribe@comp.org`, then reply to the confirmation request. Ezmlm will send you a message to let you know that you are no longer a subscriber. Ezmlm will also let you know if the address was not in the subscriber list. If so, you are probably subscribed under another address.

To find out your subscription address, look at the very first header of any message from the list. This should contain something like `mailinglist-return-2345-harold=work.com@comp.org` which tells you that the message number is "2345" and the subscriber address is `harold@work.com`. Now you remember that you subscribed a different address, and construct the appropriate command address to remove that address from the subscriber database: `mailinglist-unsubscribe-harold=work.com@comp.org`. Replying to the confirmation request will return the desired acknowledgement that the address has been removed from the subscriber database.

There is another way to find which one of many possibilities is your subscription address. Imagine you have addresses `fred@old.com`, `gollum@previous.edu`, and `webmaster@previous.edu` all forwarded to `f1234@paid.job.com`. You get the messages, but you don't know to which address they are sent. You could figure it out from the "Delivered-To:" and "Received:" headers. You can also send *-unsubscribe* requests for each potential subscription address and reply to the confirmation requests. The message sent by ezmlm in response to you *-unsubscribe* confirmations will tell you if you *-unsubscribe* attempt was successful.

With ezmlm-idx(*) you can also send mail to `mailinglist-query-fred=old.com@comp.org`, `mailinglist-query-gollum=previous.edu@comp.org`, and so on. For each message, you will receive a reply telling you if the address is subscribed or not. Construct a "*-unsubscribe*" message from the one that gave you a positive reply and you're off the list!

If you for some reason are not successful with these attempts you can as a last resort contact `mailinglist-owner@comp.org`. Please describe your problem and include a FORWARDED message from the list and a list of the possible addresses you might be subscribed under. The list owner will be able to help you get off the list, and can usually figure out your subscription address from the information sent.

Unsubscribe from subscription moderated lists works the same way as for normal lists. You never need approval to remove your address from the list.

When a list is set up to allow posts from subscribers only(*), a post from an address (`jonesj@softx.com`) may be rejected since this address is not a subscriber (even though mail to the subscriber `john@univ.edu` reaches you, ezmlm has no way of knowing this). The easiest way to deal with this is to unsubscribe `john@univ.edu` and subscribe `jonesj@softx.com`. If this is not possible/desirable, send the addresses in question with a note to `mailinglist-owner@comp.org`. The list owner can add your sender address (in this case `jonesj@softx.com`) to an extra address lists of non-subscribers allowed to post (and access the archive).

1.5 Sending messages to an ezmlm mailing list.

To send messages to an ezmlm list, you must know its name. If the list is called `mailinglist@comp.org`, just send a message to `mailinglist@comp.org`. Any message accepted at this address is sent to all list subscribers.

Due to the efficiency of ezmlm, you usually get back the message you sent to the list within a few minutes. Of course, this is true only is you are a subscriber.

Most ezmlm lists require that you specify a "Subject", a few words that give everyone an idea what your message is about. This is a good thing, since many subscribers automatically delete mailing list messages without subject. Conversely, a good subject causes those interested/able to help with your comment/problem to read the message. If you send a message without a subject, ezmlm will send it back to you with an error message.

ezmlm also rejects messages to the list if the subject is a single command word only, such as "subscribe" or "help". These are almost always misdirected commands from novice users. Instead of "flames" from many subscribers, the sender gets an informative error message. As you know, your should send mail to `mailinglist-unsubscribe@comp.org` to unsubscribe from the list `mailinglist@comp.org` and a message to `mailinglist-help@comp.org` to get help and command information.

The list owner may place additional restrictions on messages to the list. Ezmlm allows rejection of messages that are not from subscribers(*), that are too long/short(*), or that contain certain MIME parts(*). If your message is not acceptable by these criteria, ezmlm will send your message back to you and tell you why it failed. You can then correct the problem and try again. You can also mail `mailinglist-owner@comp.org` to reach the list owner and ask for assistance. Replace "mailinglist" and "comp.org" with the real list and host name, respectively. In all messages to the owner, please include a FORWARDED copy of the error/problem message (see below). *Note:* ezmlm lists handle almost everything themselves. Therefore, the list-owner may check the mail only infrequently. Use ezmlm if at all possible, and be patient when you wait for a reply from the owner.

In special cases, the list owner may set up the list to be message moderated(*). When you subscribe, you should be told if the list is moderated. Also, ezmlm adds a special "Delivered-To: moderator for ..." header to the messages. On a message moderated list, your message, instead of going directly to the list, is sent to one or more moderators. They can accept or reject, but not modify the message. If the message is accepted, it is sent to the list unmodified. If it is rejected, it is sent back to you, optionally with an explanation from the moderator. On moderated lists it may take a little longer for the message to reach all the subscribers, since it has to be read and approved by at least one moderator before being sent out to the subscribers.

Message moderation is very useful for e.g. announcement lists, or lists where for other reasons not all posts are accepted.

1.6 Replying to a message from an ezmlm mailing list.

Messages sent via an ezmlm mailing list come "From:" the original sender. For many lists, it is appropriate to discuss privately with the original sender, then post a follow-up message with e.g. a solution to the list. In this case, just "Reply" to the sender. Without this behavior two things may happen: First, a subscriber may intend to reply to the original sender only, but sends a message to the entire list. This can be very embarrassing. Second, misconfigured subscriber programs sometimes send "John is on vacation" messages back to the "From:" address. The ezmlm setup allows these to go to the original sender (a minor nuisance) rather than to the list (a major nuisance with the potential for mail loops if the subscriber software is severely misconfigured).

In other cases, you may want to send the reply to all subscribers. The easiest is to type in the list address by hand or to use the "Reply-to-all" function of your mail reader. If you do this, delete your address as well as the original sender's address so that the message goes to the list only. Since messages reach subscribers within minutes, it is usually superfluous to send a reply directly to the sender as well.

1.7 Accessing the ezmlm message archive.

By default, ezmlm mailing lists keep an archive of all the messages sent to the list. Some of them may have been removed by the list owner, but usually at least the last few month's message will be there. With ezmlm you can retrieve one message at a time with the *-get* command. With ezmlm-idx(*) you can access the archive in three different ways: You can get an "*-index*" listing message subjects and authors only. This information is sent as sets of 100 messages with up to 2000 entries per request. You can also "*-get*" a range of messages(*) (up to 100 per request) which will be sorted by subject ("Re:" and other "subject modifiers" are ignored) and time received. Finally, you can retrieve a set of messages or a "thread" containing a specific message. This usually gives you an ordered set starting with the first post, and then all replies in order received.

To access the archive, again remember that commands are put into the ADDRESS. Here are examples of the commands:

`mailinglist-get.123@comp.org`
: Get message 123.

`mailinglist-index@comp.org(*)`
: Get the subject and author information for the last 100-200 messages. This also tells you the latest message number.

`mailinglist-index.205@comp.org(*)`
: Get the index for messages 200-299.

`mailinglist-index.200_299@comp.org(*)`
: Get the index for messages 200-299.

`mailinglist-get@comp.org(*)`
: Get the messages accumulated since the latest digest (returns latest message if there are no digests for the list). The normal format is MIME with only the important headers. If you instead of the *-get* command use *-getr* you get the digest as a single long message without MIME. The *-getv* command gives you the MIME format, but now all the headers of the individual messages are returned.

`mailinglist-get.123_456@comp.org(*)`
: Get messages 123-456. Because you can only get up to 100 messages per request, only messages 123-222 will be send. Send new requests to get the remainder.

`mailinglist-thread.123@comp.org(*)`
: Get the thread containing message 123. To change the format, use the *-threadr* or *-threadv* format instead. These work just like their *-get* cousins (see above).

If you haven't been following the list for a while, or you just want to see the discussion of a specific question, the easiest is to start with *-index*(*). If not found, send further *-index*(*) requests going backwards in the archive. Once you've found one message (e.g. "667") in the thread use the *-thread*(*) command: `mailinglist-thread.667@comp.org` to get the entire set.

If you're a digest subscriber and find an interesting discussion, use *-thread* or the `mailinglist-get@comp.org` address to catch up with the latest messages in that thread or on the list. The *-thread*(*) command is also useful if you catch the tail end of a discussion and want to read earlier messages.

Some lists may be set up to allow archive access to subscribers only(*). If you are not a subscriber, you can simply subscribe. If you are a subscriber, but are denied archive access, you are subscribed under an address different from the one you are sending from. Easiest is to unsubscribe (see above) the current subscriber address and subscribe the address you're sending from. Alternatively, some mail programs (e.g. Mutt) allow you to change the SENDER address depending on where you are sending mail. Last, if this address difference is necessary, you may FORWARD a list message, and a note with the two addresses and a request to `mailinglist-owner@comp.org`. The owner will add your second address to a special address list of senders allowed to access the archive (and post) even though they are not subscribers. Remember to replace "`mailinglist@comp.org`" with the list address.

1.8 Getting ezmlm digests a few times per week rather than individual messages(*).

Ezmlm list may be set up with a digest function. This is basically a second mailing list. The difference is that this list does not accept messages but sends out "digests", ordered collections of messages sent to the main list. Usually, a "digest" is sent out every other day. This is very convenient for users who want to follow a list, but not participate in the discussion. A digest is sorted, so it's easy to read what interests you.

Simple ezmlm/idx lists test if the conditions for a digest are met whenever a message arrives. Thus, the digests do not arrive at a specific time, but rather approximately every two days, or more often if there is a lot of traffic.

The digest of a list "list" is called "list-digest", and the commands are the same as for the main list. Thus, to subscribe to `mailinglist-digest@comp.org` (the digest of `mailinglist@comp.org`) send a message to: `mailinglist-digest@comp.org`, or to subscribe the address `jonesj@softx.com` send to: `mailinglist-digest-jonesj=soft.com@comp.org`. Of course, ezmlm will send a confirmation request, and then a message notifying the user that s/he is a subscriber.

1.9 Unsubscribing from an ezmlm digest list.

You unsubscribe to the digest in the same manner as you unsubscribe from the main list, but using the digest list name. Thus, to unsubscribe your current address from `mailinglist-digest@comp.org` send mail to: `mailinglist-digest-unsubscribe@comp.org`. Again, you can unsubscribe `john@work.com` by instead mailing `mailinglist-digest-unsubscribe-john=work.com@comp.org`. Ezmlm confirmation works as usual. If subscription to the main list is moderated(*), subscription to the digest list is moderated as well.

1.10 Temporarily halting message delivery.

Just unsubscribe from the list or list-digest, then subscribe when you want mail to resume. If desired, get missed messages from the archive with *-get* or just subject/author info with *-index*(*).

Due to the fast replies from ezmlm this is easier and more efficient than having “dormant” subscribers.

1.11 Contacting the list-owner.

You can always contact the list owner by mailing to the “owner” address. For the list `mailinglist@comp.org` just send your message to `mailinglist-owner@comp.org`. Please state your problem and FORWARD any error message or a list message with your request. Since ezmlm can take care of virtually all request without human intervention, the list owner may read mail only infrequently. Please be patient and remember that in most cases you are asking the list owner to manually do what you could do yourself from your mail reader via ezmlm with the information in the error or list message.

Chapter 2

EZMLM/IDX MODERATOR'S AND ADMINISTRATOR'S MANUAL

2.1 Introduction.

This chapter describes how to be a moderator (message and subscription) and remote administrator for a mailing list run by ezmlm/idx[1] .

Ezmlm keeps subscriber lists secret. Only if configured (see below) will it allow the addresses to be listed, and then only to remote administrators. Similarly, moderator identities are never revealed to subscribers or to other moderators.

Ezmlm sends special moderator help in response to a *-help* request for a moderator. To receive this information for `mailinglist@comp.org`, send mail to `mailinglist-help@comp.org`.

2.2 Message moderation(*).

Ezmlm lists may be set up with message moderation. When ezmlm receives a message for the list, it saves it, sends a moderation request with the message enclosed to the moderator(s). The subject of moderation request is "MODERATE for ...".

If you would like to approve the message, just reply to the "Reply-To:" address of the moderation request. Usually, this is the default for the "reply" function of your mail program. Alternatively, click on the "accept" address or copy and paste it into the "To:" field of a new message.

To reject the message, reply to the "From:" address, click on the "reject" address, or copy and paste the "reject" address into the "To:" field of a new message. You can also just replace the "accept" in the accept address with "reject". Optionally, type a comment between two lines starting with "%%%". The sender of

[1]Features that require ezmlm-idx are marked "(*)"

the message will receive a message from ezmlm explaining that the message was rejected. If you typed a comment, it will be put into the rejection notice.

The third alternative is to ignore the request. If no moderator has replied within 5 days, the message is returned to the sender with an explanation.

To minimize mail to busy moderators, ezmlm does not confirm moderator actions. More than one moderator may reply. The message will not be duplicated. The first valid reply to be received will decide the fate of the message. Ezmlm will send an error message only if if a later reply requests an action different from the one already taken. Thus, if you decide to accept the message, but another moderator has already rejected it, ezmlm will notify you. If instead the message had been previously accepted, ezmlm would log your request, but not send you a notification.

If a message is send *by* a moderator, it is sent for approval to that moderator only. The approval is necessary since anyone can claim to be a moderator, but only the moderator will be able to receive the moderation request. This feature is very useful for announcement lists where any moderator can make an announcement without bothering the others, while at the same time non-moderators cannot post without permission.

2.3 Remote administration(*).

A remote administrator is an address that can add/remove addresses from the subscriber list. To subscribe `john@home.net` to the list `mailinglist@comp.org`, send mail to `mailinglist-subscribe-john=home.net@comp.org`. Normally, `john@home.net` will be asked to confirm. However, if you are a remote administrator, *you* will receive the confirmation request. Thus, `john@home.net` does not need to participate. When you've successfully added the address, `john@home.net` will receive a message acknowledging his subscription.

You can unsubscribe addresses (*-unsubscribe*) or determine if an address is a subscriber (*-query*). You may also *-list* subscriber addresses list and/or *-edit* list text files, such as the one sent in reply to the *-help* command (see below). You can use all these functions to help users subscribe or unsubscribe to the list.

2.4 Adding subscriber aliases(*).

Ezmlm lists may be set up to only allow subscribers to send messages to the list. This is less secure than moderation, but still keeps most "garbage" off the list. Occasionally, a user may wish to send messages from an address other than the subscription address. As a remote administrator, you can add the user's alias to a special "allow" database. To add `john@home.net` as an alias to the `mailinglist@comp.org`, send mail to `mailinglist-allow-subscribe-john=home.net@comp.org`. *-unsubscribe* and other commands work the same way. The messages ezmlm sends talk about the `mailinglist-allow@comp.org` mailing list, but of course you know that this is just a figure of speech.

On lists that do not have subscription moderation, users can add themselves to the "allow" database in the same way. This is not documented in the USER'S manual.

Archive access may also be restricted to subscribers. Like subscribers of the list or the digest list, addresses in the "allow" database are allowed to access the archive.

2.5 Preventing an address from posting(*).

Just as you may want to allow some non-subscriber addresses to send messages to the list, you may want to (temporarily) prevent an address from posting. This is easily defeated, but very useful if some defective mail robot or other mailing list sends messages to the list address. The system works just like the "allow" list, but is called "deny". To prevent `robot@druga.su` from posting to `mailinglist@comp.org`, send mail to `mailinglist-deny-subscribe-robot=druga.su@comp.org` and reply to the confirmation request, respectively.

You can do this only if you are a remote administrator and the list is set up to support this option. You can use *-unsubscribe*, *-query*, and other commands with this address collection, just as for the normal subscriber database. Normal users cannot access the database. This database does not affect archive access.

2.6 Listing subscribers(*).

The list may be configured to allow remote administrators to obtain a list of subscriber addresses. If `mailinglist@comp.org` is set up with this option and you are a remote administrator, mail to `mailinglist-list@comp.org` will get you the list. If you are sending from another account, you can use `mailinglist-remote=admin.net@comp.org` to get the list to `remote@admin.net`. Of course, this will work only if `remote@admin.net` is a remote administrator.

Subscriber lists for the "digest", "deny" and "allow" databases are obtained in a similar manner by a message to `mailinglist-digest-list@comp.org`, `mailinglist-deny-list@comp.org`, or `mailinglist-allow-list@comp.org`.

2.7 Subscription moderation(*).

If the list is set up for subscription moderation, a moderator has to approve the user's subscription request. The process starts with a user *-subscribe* request, followed by the normal user confirmation. After this, ezmlm sends a second confirmation request to the moderator(s). If any moderator replies to this request, the user is accepted as a subscriber.

You may send the user a message to request further information. However, this will reveal your address/identity to the user.

Users may unsubscribe at any time without moderator assistance.

2.8 Editing ezmlm administrative messages(*).

If configured, remote administrators may edit (via E-mail) ezmlm text files, such as the one sent in reply to the *-help* command. A list of editable files, their use, and instructions for editing may be obtained with the *-edit* command. Thus, for the list `mailinglist@comp.org`, send a message to `mailinglist-edit@comp.org` and follow the instructions. With or without ezmlm-idx, the messages can be modified locally with a text editor.

Chapter 3

EZMLM/IDX OWNER'S INTRODUCTION

3.1 Introduction.

This chapter tells you how to deal with common problems and introduces you to the power and technical aspects of ezmlm/idx. If you want to customize your list, please use the man pages and FAQ included in the ezmlm and ezmlm-idx packages (see A.1).

As the owner of an ezmlm list you will appreciate the high degree of speed and automation of ezmlm. Even with large numbers of subscribers, the parallelism of ezmlm/qmail results in rapid delivery of the message to the subscribers. This makes it easy to have discussions among list members. With slower list managers the discussion is often limited to a small set of users who send mail on the topic to each other, "Cc:ing" the message to the list. When the other subscribers receive the message, the discussion is way past this point[1] .

Ezmlm *-subscribe* and *-unsubscribe* commands result in a handshake which essentially eliminates the possibility of people other than the user adding or removing the user's address from the list.

For the list `mailinglist@comp.org`, mail to the `mailinglist-help@comp.org` address returns a message with all the relevant ezmlm commands for subscription and message archive related functions. Archive access is simple and powerful.

Together these features make it very easy to set up and manage an ezmlm mailing list. The rare messages to the list-owner address (`mailinglist-owner@comp.org`, in the example) are usually due to subscribers who have subscribed under one address, and now use another. Even they can (un)subscribe without owner intervention, but some subscribers may be unable/unwilling to read the ezmlm help message and follow the instructions therein.

[1]Features available only if ezmlm-idx is used are marked "(*)".

3.2 Using auxiliary address databases(*).

In addition to the regular subscriber address databases for the list and list-digest, ezmlm may use up to three other address databases. One of these the "allow" database is housed in **DIR/extra/**[2] . In case posts or access to the archive is restricted to subscribers, addresses present in **DIR/extra/** are also accepted. If the subscriber `jonesj@softx.com` sometimes posts from `john@home.net`, you can just add the second address to **DIR/extra/** to allow this. The previous chapter describes how to do this via the "allow" command extension (see 2.4). If you have shell access, you can simply:

```
% ezmlm-sub ~/listDIR/extra john@home.net
```

To list all the addresses in **DIR/extra/**:

```
% ezmlm-list ~/listDIR/extra
```

Similarly, you can use ezmlm-unsub(1) and ezmlm-list(1) to remove and list addresses, respectively.

Another address database is **DIR/blacklist/**. If the list was configured with the ezmlm-make(1) "-k" switch, posts from addresses in this database are rejected. This is easy to circumvent, but very handy if a subscriber creates a mail loop (ezmlm/qmail has excellent loop detection, but some subscriber vacation programs generate a new message to the list for every message received and ezmlm has no way of knowing that this is not a legitimate post). For such a subscriber address, you can just temporarily add it to **DIR/blacklist/**, which is some cases is preferable to simply unsubscribing the user. If this database is configured, remote administrators can access it via the "deny" command extension (see 2.5).

The final database is **DIR/mod/** in which moderator and remote administrator addresses are stored. Remote administrators may themselves (un)subscribe any address. Remote administration (see 2.3) is configured with the ezmlm-make(1) "-r" switch. The same addresses may also be subscription moderators. If configured ("-s" switch) a user can subscribe only if one or more of the moderators approve. Messages to the list can also be moderated. This is configured with the "-m" switch. Again, the moderators are stored in **DIR/mod/**, although it is possible to configure the list so that message and subscription moderators are different sets of addresses (see A.1).

To add a moderator/administrator:

```
ezmlm-sub ~/DIR/mod john@home.net
```

ezmlm-list(1) and ezmlm-unsubscribe(1) are used to list and remove moderators. Normally, the moderator database cannot be accessed remotely. However, remote access can be arranged (see A.1).

3.3 How ezmlm list setup is controlled.

Ezmlm lists are set up with a one line ezmlm-make(1) command. Existing lists can be modified using the ezmlm-make "-e"(*) switch, without loss of archive, subscriber database, or other state information.

[2] File names are shown in **boldface**. Directories end with a slash. "**DIR/**" is the list directory.

The operation of ezmlm-make is controlled by ezmlmrc(*). To customize ezmlm list setup on a host-wide or user (virtual domain)-wide basis, copy this file from the ezmlm binary directory to **/etc/ezmlmrc** of **~user/.ezmlmrc** and edit it. (See A.1.)

3.4 Basic list types.

ezmlm offers virtually infinite possibilities for customization. The more common types of lists can be created and configured with a single ezmlm-make command. For more complicated lists and a discussion of how ezmlm works, see A.1).

3.4.1 Public vs. non-public lists.

Normal mailing lists are *public*, i.e. anyone can subscribe/unsubscribe, and access the list archive. Occasionally, you may want to set up a non-public list, where (un)subscribe actions are possible only from the command line or remotely by the administrator(s). This is controlled by the ezmlm-make(1) "-P" switch.

3.4.2 Adding a digest list to your ezmlm list(*).

The digest is a collection of messages, produced about every two days. If more than 30 messages or 64 kbytes of "message body" have arrived since the latest digest, the digest is sent earlier. All these criteria can be changed (see A.1). Also, lists can be set up to produce digests at a specific time each day, every other day, etc see A.1).

Some users do not get involved in discussions on the list but rather want to read messages on a specific topic. For these users, a digest with sorted messages is much preferable to receiving the messages one at a time. These users would subscribe to the digest list, rather than to the list itself.

To set up a digest list with the list, all you need to do is to use the ezmlm-make "-d" switch. The digest of the list `list@host` is always called `list-digest@host`. Thus for the list `mailinglist@comp.org`, the digest list would be `mailinglist-digest@comp.org`.

The digest can be sent in several formats. The default, MIME with removal of "non-essential" headers is virtually always the best. A "format-specifier" can be used with the ezmlm-get(1) "-f" switch to override this. "v" returns MIME with all headers, and "r" a rfc1153-like non-MIME format (see A.1). You can use **DIR/digheaders** to specify the message headers shown for archive excerpts when using the "m" (default) format.

3.5 The ezmlm archive.

Ezmlm mailing lists are by default set up to keep an archive of all the messages sent to the list. This archive is used to create digests. Users can also access messages in the archive, using the *-get*, *-thread*(*), and *-index*(*) addresses (see above). If the users would like to receive messages in a non-default format, they may do so by adding a *format specifier*(*) to the command address name. Thus, to receive messages 45-65 from `mailinglist@comp.org` in the "v" (virgin - MIME with all headers) format, the user would send a message to `mailinglist-getv45-65@comp.org`.

3.6 Restricting list usage.

Ezmlm mailing lists can be set up with many restrictions, both for posts and archive access. Through customization of ezmlm, there are many variations. The more common ones are accessible through ezmlm-make(1) switches when the list is created or modified (see A.1).

3.6.1 Placing restrictions on messages to the list.

By default, ezmlm rejects all messages that have an empty subject or a subject consisting of a command name (such as "subscribe") only. The lists will also reject messages that do not have the list address in the "To:" or "Cc:" header(*). This removes the majority of spam without affecting normal posts. In addition, lists may be configured to reject messages below or above a certain size(*), measured on the message body. Ezmlm can also reject messages containing certain MIME content types(*). It may make sense to reject messages consisting only of "text/rtf" on lists with users on multiple platforms, many of which would be unable to read the message. Similarly, restricting the maximal size of messages to e.g. 20000 bytes may save inexperienced users from quoting entire digests or sending large programs or such as an attachment. All these restrictions are handled by ezmlm-reject(1) in the **DIR/editor** file in the list directory (see A.1).

3.6.2 Placing restrictions on subscribers(*).

To set up lists with subscription moderation so that each *-subscribe* request must be approved by a moderator, use the ezmlm-make(1) "-s" switch.

Lists set up with the ezmlm-make(1) "-u" switch allow posts only from subscribers of the list or list-digest. Use of the "-m" switch makes all posts moderated (see below). A combination of these switches ("-um") has a special meaning: posts from subscribers go directly to the list, whereas posts from others are sent out for moderation. Also, messages that are ignored (neither accepted nor rejected) by the moderators, usually result in an informative error message to the sender after about 5 days. However, for "-um" lists, such posts are ignored. This gives the moderator the choice of accepting the message, rejecting it with the accompanying notification of the sender, and silently ignoring it (most appropriate for e.g. junkmail/spam).

3.6.3 Placing restrictions on archive access(*).

For some lists, you may not want non-subscribers to be able to read archived messages. This restriction can be placed on a list with the ezmlm-make(1) "-g" switch. Since ezmlm always sends archive excerpts to the sender of the request this ensures that only persons able to read mail to a subscriber address can receive archive excerpts. All such persons can already read the list traffic.

The **DIR/extra/** directory holding the "allow" addresses can be useful to give archive access to addresses not in the list or list-digest subscriber databases. Addresses in this database are treated as subscribers, both for archive access and posts, but of course they will not receive list messages.

You can also make the list non-public. This can be done by removing **DIR/public** or by using the ezmlm-make "-P" switch (see A.1). This results in a list where only remote administrators can access the archive. You can eliminate the archive altogether, but it may be required for other purposes (such as digest generation).

3.7 Modifying messages sent by ezmlm.

All these additions can be made default for all lists by modifying **ezmlmrc** (see A.1).

3.7.1 Adding a trailer to each message(*).

You can add small "trailer" to each message. This usually would contain information on how to unsubscribe from the list, or where the list *html* archive is kept. To add a trailer, simply put the text in the file **text/trailer** in the list directory. You can use the ezmlm-make(1) "-t" switch to set up the list with a simple trailer that gives "unsubscribe" instructions.

3.7.2 Removing certain headers from messages.

DIR/headerremove contains headers that are removed from all messages before they are sent to the subscribers. Some of these must be removed in order for the list to function. You can add others (see A.1).

3.7.3 Adding headers to list messages.

Headers in **DIR/headeradd** are added to posts before they are sent to subscribers. They are also added to the archived version of the message.

Ezmlm can add a *sequence number header*(*) showing the message number. To achieve this, just place the desired header name (e.g. "X-Seq:") in the file **DIR/sequence**.

3.7.4 Stripping certain MIME parts from multipart messages(*).

Ezmlm can strip out MIME parts from multipart MIME messages. Some mail clients use "multipart/alternative" to send both plain text and html versions of the same message. On many lists this is considered annoying and wasteful (messages are usually 2-3 times as large). You can have ezmlm remove the html part from such messages by adding the line "text/html" to **DIR/mimeremove**.

3.7.5 Adding a subject prefix to messages(*).

Ezmlm can also add a "prefix" to the subject line of each message that does not already contain it. This was common on old lists, when mail readers were unable to sort mail, except by subject. It still popular for historical reasons, even though it violates mail standards. Use this only if you feel that it is necessary, and for novice users where "look" is more important than correctness. If you do use this make it the list name and short, e.g. "[mailinglist]" (well, that's from the example, in real life this is too long and too long a name for a mailinglist as well. `ml@comp.org` with "`ml`" would be better). The prefix with the list name can be configured with the ezmlm-make(1) "-f" switch. Alternatively, just place the text in the file **DIR/prefix**.

For a real thrill, you can use "[ml-#]". Here, "#" will be replaced by the message number, labelling all messages with the message number of the first message in the thread. This is not only a mail standard violation, but will inconvenience many users and html archiving programs. Still, it may be useful and help novice users use the archive efficiently (mail `ml-thread-#@comp.org`) where "#" is the number in the prefix of the message to get the entire thread from the archive). This works also with partially rfc2047-encoded subjects. (See the ezmlm-send(1) man page for more info.)

3.8 Allow remote administrators to list subscriber addresses(*).

Many old mailing list managers allow users to list the addresses of all subscribers. This used to be OK, but has lately become a means for junkmailers/spammers to acquire large collections of addresses. Consequently, ezmlm does not allow this. Still, it may be useful for the list owner to be able to get such a list to help a user unsubscribe or to remotely backup the subscriber list. Ezmlm will supply such a list if the list was set up with the ezmlm-make(1) "-r" and "-l" switches *and* the recipient of the list is a "remote administrator" (see 2.3).

The remote administrator right apply to all address databases of the list (subscribers of list and list-digest, "allow" and "deny"), except the moderator addresses. Moderators' addresses cannot be accessed remotely.

3.9 Allowing remote administrators to edit ezmlm text files(*).

Use the ezmlm-make(1) "-n" switch to allow remote editing of text files to remote administrators. Editing is allowed for all *existing* text files. Thus, in order to allow editing of the **DIR/text/info** file (sent in response to the *-info* command), you must at least have created such a file from the shell command line, or via ezmlmrc (default for new lists). See A.1 for more information.

A list of editable files is contained in the **DIR/text/edit-list** file. This file itself for the list `mailinglist@comp.org` is editable via `mailinglist-edit.edit_list@comp.org`. This file has no effect on which files can be edited, but it useful since it contains the instructions returned in reply to the *-edit* command. Hyphens in file names need to be replaced by underscores in the edit commands. See A.1 for more information.

3.10 Ezmlm handling of undeliverable/bounced messages.

Qmail/mail systems are fault tolerant by virtue of a queuing mechanism where a message is removed from the queue only when the recipient mail transfer agent (MTA) has acknowledged receipt of the message. With this, the responsibility for the message has been transferred from the sending MTA to the receiving MTA. For messages that remain in the queue, the sending MTA will retry the transmission. After a set period of time, the message will be bounced, i.e. returned as "undeliverable". If the recipient does not exist (which may be caused by a misconfiguration of e.g. the domain naming service) the message is returned immediately as "recipient unknown".

Ezmlm uses the subscriber address and message number encoded within the sender address to keep track of which message have "bounced" for which recipient. Since messages can bounce for a variety of reasons, ezmlm doesn't remove a subscriber from the list after a single bounce. Rather, it waits about 12 days after the first bounce, then sends a "warning" message to the subscriber. The warning message lists the numbers of the messages missed. The user can retrieve these from the archive. If this warning message bounces as well, ezmlm waits another 12 days, then sends a "probe". First when this message bounces is the subscriber removed from the subscriber database. This mechanism guarantees that a subscriber won't be removed due to a temporary error, while assuring that all non-existent or permanently failing addresses will be removed eventually.

3.11 Helping users unsubscribe.

Some users never learn and don't want to spend the time to figure out how to unsubscribe. As a remote administrator(*), you can unsubscribe them entirely on your own (see 2.3). You can also do this from the command line. Alternatively, you can as a normal user send a *-unsubscribe* request for the subscriber address and let the subscriber confirm. Another alternative is to tell the user to send a message to `listname-help@listhost` and ask the user to follow the instructions, which are very explicit.

Other users have attempted to unsubscribe, but failed. They usually will tell you what they've tried, and forward a list message to you. Most likely, they are subscribed with an address other than their normal sender address. If you're lucky, the subscriber address is in the users E-mail to you. If not, you can ask the user to give you a list of possible addresses. You can then send mail to `listname-list@listhost`(*) (if enabled) and use "Find" or "grep" to search through the list for parts of the users addresses. If remote subscriber listing is not available, try:

```
% ezmlm-list DIR | grep -i 'part_of_user_address'
```

Then send an unsubscribe for the address(es). The user will usually receive a confirmation request (if you found the correct address), and can reply to it to get off the list.

3.12 Helping subscribers post and access the archive on subscriber-only lists(*).

Subscriber-only lists allow posts and archive access only to subscribers of the main list or the digest list. If a subscriber's posts or archive access attempts are rejected, the subscriber is working with a sender address that is not the one s/he is subscribed under. Usually, the user's sender address will be in the E-mail to you. Ezmlm-idx keeps a database of "extra" addresses which are treated like subscribers for the purpose of subscriber-only restrictions. To add an address, send mail to `listname-allow-subscribe-user=userhost@listhost`. If you're a remote administrator the confirmation request will come to you, otherwise to `user@userhost`. Replying to the request will add the address. The user should now be able to post and access the archive. To remove an address, user the same procedure, but with the *-unsubscribe* command. Users can manipulate the database themselves, but this may be too complicated for all but advanced users.

3.13 Stopping mail loops.

Qmail and ezmlm have built-in advanced mail loop prevention. However, some autoresponders are misconfigured and create a new message to the list for every message received. Since this is a completely new message, ezmlm has no way of knowing that it is not a post. This is very rare, but may happen. The symptom is that a "John is on vacation" messages will arrive at the list, go out to subscribers, only to be followed by another "John is on vacation" message, and so on.

To stop this, look at the sender of the "vacation" message. Unsubscribe the address.

Even rarer is that there is a mail loop somewhere on the Internet, and that it sends of messages to your list. Unsubscribing doesn't help. Your only recourse is to block the sender of the message from posting. Easiest is to use the "blacklist"(*) function with the ezmlm-make(1) "-k" switch, then add the sender address:

```
% ezmlm-sub DIR/blacklist badname@badhost
```

or if configured, send mail as a remote administrator to `listname-deny-subscribe-badname=badhost@listhost`. You could also add a line:

```
|/bin/echo -- "$SENDER" |/usr/bin/grep -iv 'badname@badhost' || exit 99
```

to **DIR/editor** of the list.

3.14 Dealing with SPAM.

Ezmlm/idx lists by default(*) reject messages that do not have the list address in the "To:" or "Cc:" headers. Most "bulk E-mail" does not pass this test. If you still receive too much junk mail to your list, you can set up your list as a subscriber-only(*) list. This creates a small overhead of users who try to post from addresses that are not their subscriber address. See above on how to deal with this.

Since the message sender is easy to fake, this is not secure. A determined SPAMMER can subscribe to the list or just send mail "From:" a subscriber address. The next level of countermeasure is to have a message moderated(*) (and possibly subscription moderated(*)) list. The overhead is higher - someone has to approve each message - but it is much more secure. Ezmlm-idx has support for multiple moderators making it easy to set up a list with fast and efficient moderation.

3.15 Dealing with archiving servers.

Several sites on the Internet subscribe to mailing lists, archive the messages, and make them publicly available. This may not be desirable, as the this often exposes the addresses of the subscribers. The subscribers, while happy to discuss their work on a peer list, may not appreciate messages from the general public asking for follow-ups to messages posted to the list.

Ezmlm lists, by default(*), are set up with a "X-No-Archive: yes" header. For older lists or lists using ezmlm alone, just add this header to **DIR/headeradd**. Well-behaved archiving servers respect this. If you want to maintain full control, the only recourse is to make the list subscription moderated(*). Of course, the messages are no more "secure" than the least reliable subscriber.

3.16 Making an existing list message or subscription moderated(*).

Like other list changes, you can use ezmlm-make with the "-e" edit switch together with other desired switches and arguments. An existing list can be made message moderated with:

```
% ezmlm-make -em DIR
```

To add remote administration, add the "-r" switch. To add subscription moderation, add the "-s" switch.

3.17 Backing up and removing parts of the archive.

The archive is a set of files, one per message, kept in the **archive** subdirectory of the list directory. The directory **archive/0/** contains messages 1-99, **archive/1/** messages 100-199, etc. The files within the directories are named by the last 2 digits of the message number, "01-99" for **archive/0/** and "00-99" for others. In addition, each set of messages is indexed(*) in the file **index** stored with the corresponding message files.

With time, the archive grows and it may become necessary to remove some messages. When the list is at message "12301" one may wish to remove messages 1-9999 from the archive. Easiest is to remove messages as entire subdirectories. For the above, do:

```
% cd ~/
% tar -cvf listarch1_9999.tar DIR/archive/? DIR/archive/1?
```

```
% gzip listarch1_9999.tar
% rm -rf ~/DIR/archive/? ~/DIR/archive/1?
```

The “dir” here is the list directory. (You may be able to combine the “gzip” action with the “tar” command by adding the “z” switch.)

Save **listarch1_9999.tar.gz** in a safe place. (Usually, you would choose a better name for the backup “tar” file). Nothing else needs to be done. If instead you remove individual messages and use ezmlm-idx, it is useful, but not necessary to do:

```
% ezmlm-idx ~/dir
```

to re-index the archive. (For a large archive on a slow machine, this command may take a few minutes.)

3.18 Restoring parts of the archive.

To restore the messages saved in the example above to the archive, simply:

```
% cd ~/
% gunzip listarch1_9999.tar.gz
% tar -xvf listarcg1_9999.tar.gz
% ezmlm-idx ~/dir
```

Re-indexing the archive (as shown) is recommended, since the format for the **index** file may have been changed if ezmlm was upgraded since the backup.

Appendix A

More information.

A.1 More about ezmlm and ezmlm-idx.

Ezmlm is a mailing list manager written by Dan J. Bernstein. Ezmlm-idx is an extension to ezmlm written by Frederik P. Lindberg. Ezmlm has the essential features of a mailing list manager. Ezmlm adds useful, but non-essential features like digests, list moderation, etc. The list owner will have made the trade off of features vs. increased size and complexity. To test if your list has ezmlm-idx installed, send a message to `list-get@host` where "list@host" is the name of the list. A list with ezmlm alone will send you a help message saying that your message wasn't sent to one of its command addresses. A list with ezmlm-idx will return one or more messages from the archive, or a message telling you that the list is not archived. Throughout this manual, features available only with ezmlm-idx are marked with "(*)".

For more technical information about ezmlm, it's options, and error resolution, please read the FAQ (`<http://www.ezmlm.org/>`) and the ezmlm man pages. A text version of the FAQ is included in the ezmlm-idx distribution. The latest FAQ in various formats can be found at `<ftp://ftp.id.wustl.edu/pub/patches/>` or `<ftp://ftp.ezmlm.org/pub/patches/>`. The FAQ also contains information on where to obtain ezmlm, where to find mailing lists about ezmlm, etc. While it is called a "FAQ", the document will with time become the ezmlm/idx technical manual. The man pages are always installed with ezmlm. To read the man page about ezmlm, do:

```
% man ezmlm
```

To read the man page for ezmlm-make, do:

```
% man ezmlm-manage
```

Also, take a look at Dan J. Bernstein's pages on ezmlm and other programs he has written at `<http://www.pobox.com/~djb/>` and the qmail home page at `<http://www.qmail.org/>`.

A.2 More about this document.

This document is updated for ezmlm-0.53 and ezmlm-idx-0.31. It is available at <ftp://ftp.id.wustl.edu/pub/patches/ezman/> as "ezman-x.xx.format.*tar*.gz". "x.xx" is the version, "format" is the format, e.g. "ps", and ".tar" is present if the document consists of several files.

This document comes with absolutely no warranties. If there are errors or omissions in this manual, please send corrections/comments to the author. The manual is intended to be complete for users and moderators/administrators. For list owners, it is intended more as a summary and introduction to the FAQ.

A.2.1 Copyright and translations.

I reserve the right to sell a printed version of this document in the original English, or in any other language. You may distribute it in any other way, even for money, and of course you may give away a printed version.

If you make modifications, you must retain the original copyright notice and identify clearly the sections you've modified. I'd prefer if you notified me of errors/omissions so that I can improve this document myself.

If you translate this document, you own the translation. You may distribute your translation in any way, including printed and for sale. However, I reserve the right to make and distribute an independent translation.

These restritions are intended to encourage translation, while making it possible to use this text in e.g. a printed book for sale, and publish such a book in any language. The remainder of this notice is intended to make it as easy as possible to disseminate this document in the wonderful world of free software.

Appendix B

Ezmlm Commands

B.1 Commands available to users.

To use a command on the mailing list `ml@host.org`, send a message to `ml-command@host.org`. For commands affecting an address database, you can add "-target=targethost.dom" after "command" to apply the command to `target@targethost.dom` instead of to the address your sending from. This works for all commands related to subscriber databases, *-faq*(*), *-help*, and *-info*(*). It does not work for archive retrieval commands. For the archive access commands, you may instead add arguments consisting of a message number or range of message numbers after the command[1] .

-faq(*)

Send a list of frequently asked questions and answers specific for this list.

-get

Get a message or range of messages from the archive. Without argument it returns the messages since the latest digest. To retrieve a range of messages, use `list-get.123_234@host`(*). At most 100 messages are returned. Without ezmlm-idx, individual messages can still be retrieved by sending mail to `list-get.123@host`.

-help

Returns general command help for the list.

-index(*)

Returns a subject/author index for the messages. Without arguments, the entries for at least the latest 100 messages is returned. At most 2000 entries are returned per request.

-info(*)

Return information about this list.

[1]Features available only if ezmlm-idx is used are marked "(*)".

-query(*)

Tells you if your address (or the target address) is a subscriber or not. It's always returned to the address in question, so it cannot be used to determine the subscriber status of others. You can also use `list-query-user=userhost@listhost`, where "list@listhost" is the address of the list and "user@userhost" is the address that should be tested. The reply is sent to that address, so you can't use this to test the subscriber status of addresses that you do not control.

-subscribe

Add the target (your) address to the subscriber list. The address will receive a confirmation request.

-thread(*)

Return the thread containing the message number given as the argument. The argument is required.

-unsubscribe

Remove the target (your) address from the subscriber list. The address will receive a confirmation request. You can also use `list-unsubscribe-user=userhost@listhost`, where "list@listhost" is the address of the list and "user@userhost" is the address that should be removed from the subscriber database3.

B.2 Additional commands that may be available to remote administrators(*)

These commands are only available to remote administrators. They need to be explicitly enabled, i.e. they are disabled by default, even for remote administrators. Anyone can send these commands, but they are allowed only if the recipient is a remote administrator. Thus, if `address@addhost` sends a message to `mailinglist-list-mod=modhost@comp.org`, ezmlm will reply with a subscriber list to `mod@modhost` if this address is a remote administrator, and if the list is configured for remote subscriber listing.

-edit(*)

Edit ezmlm text files via E-mail.

-list(*)

Retrieve a list of subscriber addresses.

B.3 Command extensions for ezmlm address databases(*)

Ezmlm commands that manipulate address collections can be used on addresses other than the list subscribers. To apply a command to another collection, send mail to `ml-extension-command@host.org` or `ml-extension-command-target=targethost.dom@host.org`.

(none)

`ml-command@host.org` affects subscribers of the main list.

digest(*)

Affects subscribers of the digest list.

allow(*)

Affects addresses that are allowed to post or retrieve messages from the archive without necessarily being subscribers. By default, anyone may post to the list or retrieve archived messages, so this address database is without effect. It is very useful when posts and access is allowed to subscribers only, since some users may send from addresses other than their subscriber addresses.

deny(*)

These addresses are prevented from posting. Accessible only to remote administrators an only if configured (by default, this address list is not used). It is usually used to decrease the impact or errant mail robots rather than to attempt to prevent certain addresses from posting. These addresses never affect archive access.

Appendix C

Emulating other mailing list managers(*).

C.1 Global ezmlm commands (majordomo, listserv, listproc).

These mailing list managers use a central address. The command is in the message body. The ezmlm-reject(1) (*) program can be configured to answer request at a standard address of a computer. Usually, "`listserv@host`" or "`majordomo@host`" will be used. Here, ezmlm-request understands most basic commands of other mailing list managers, as long as the functionality is relevant to ezmlm.

Ezmlm will interpret the first line of the message body as a command (to emulate other mailing list managers). You need to send one message per command. The command "*lists*" will give you a list of available mailing lists with a short description. The command "*which*" will tell you which lists you are subscribed to. This will work **only** for list configured by the system administrator. Naturally, you can use the "ezmlm way" to put the commands into the address with a message to e.g. `listserv-lists@host`.

You can send basic commands for most mailing list managers to this address. Ezmlm will translate them into command messages to the individual lists. If only the local name of the list is specified, ezmlm will find the host part in its configured list. For instance "*subscribe fishing*" will subscribe you to the `fishing@wildlife.org` list, assuming that that list was in the reply to the "*lists*" command.

Please see the ezmlm FAQ and ezmlm-request(1) man page for more information on how to set up and use this interface.

C.2 Commands at the `list-request` address (smartlist).

If the ezmlm-make "-q" switch is used to set up/edit the list, ezmlm-make configures ezmlm-request(1) to answer the `list-request@host` address. If a subject is given, it will be interpreted as a command. See FAQ and the ezmlm-request(1) man page for more info.