



LISTSERV® 16.0

List Owner's Manual

www.lsoft.com



Information in this document is subject to change without notice. Companies, names, and data used in examples herein are fictitious unless otherwise noted. L-Soft does not endorse or approve the use of any of the product names or trademarks appearing in this document.

Permission is granted to copy this document, at no charge and in its entirety, if the copies are not used for commercial advantage, the source is cited, and the present copyright notice is included in all copies. Recipients of such copies are equally bound to abide by the present conditions. Prior written permission is required for any commercial use of this document, in whole or in part, and for any partial reproduction of the contents of this document exceeding 50 lines of up to 80 characters, or equivalent. The title page, table of contents, and index, if any, are not considered to be part of the document for the purposes of this copyright notice, and can be freely removed if present.

Copyright © 2009 L-Soft international, Inc.
All Rights Reserved Worldwide.

LISTSERV is a registered trademark licensed to L-Soft international, Inc.

ListPlex, CataList, and EASE are service marks of L-Soft international, Inc.

UNIX is a registered trademark of X/Open Company Limited.

AIX and IBM are registered trademarks of International Business Machines Corporation.

Alpha AXP, Ultrix, OpenVMS and VMS are trademarks of Digital Equipment Corporation.

OSF/1 is a registered trademark of Open Software Foundation, Inc.

Microsoft is a registered trademark and Windows, Windows NT and Windows 95 are trademarks of Microsoft Corporation.

HP is a registered trademark of Hewlett-Packard Company.

Sun is a registered trademark of Sun Microsystems, Inc.

IRIX is a trademark of Silicon Graphics, Inc.

PMDF is a registered trademark of Innosoft International.

Pentium and Pentium Pro are registered trademarks of Intel Corporation.

All other trademarks, both marked and not marked, are the property of their respective owners.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

Some portions licensed from IBM are available at <http://oss.software.ibm.com/icu4j/>

This product includes code licensed from RSA Security, Inc.

Manuals for LISTSERV are available in PDF and HTML format at:

<http://www.lsoft.com/manuals/index.html>

L-Soft invites comment on its manual. Please feel free to send your comments by email to manuals@lsoft.com

Last Updated: December 8, 2009

Table of Contents

Preface - About This Manual	xiii
Editorial Note - New Version Numbering	xiii
Unicode (UTF-8) Support, Phase 1	xiii
LISTSERV Command Syntax Conventions	xiv
Contacting L-Soft	xv
Section 1 Getting Started	1
Section 2 About Mailing Lists and LISTSERV	3
2.1 What is LISTSERV®?	4
2.2 Who are the people involved?	5
Section 3 Communicating with LISTSERV	7
3.1 Using the Web Interface to Communicate with LISTSERV	7
3.2 Using Email to Communicate with LISTSERV	7
3.3 Anatomy of a LISTSERV List	9
3.4 The “OK” Confirmation Method	10
Section 4 Mailing List Types	13
4.1 One-Way Announcement Lists	13
4.2 Two-Way Discussion Lists	15
4.3 Moderated Lists	17
Section 5 Introduction to the Web Interface	21
5.1 The Default Home Page	21
5.2 Logging In	21
5.3 Setting a LISTSERV Password	23
5.4 Changing the LISTSERV Password	23
5.5 Using the Toolbar	24
5.6 Setting Your Personal Preferences	24
5.6.1 General Preferences	25
5.6.2 Archive Preferences	27
5.6.3 Report Preferences	29
5.6.4 Newsletter Profiles	29
5.7 Editing a Page	31
5.8 Accessing CataList	32
5.8.1 Searching for a List	33
Section 6 List and Subscriber Management Using the Web Interface	35
6.1 Using the List Management Dashboard	35
6.2 List Configuration	37
6.2.1 List Configuration Using the Wizard	37
6.2.2 Manual List Configuration	54
6.2.3 Topics	56

6.2.4 Normal List vs. Super-List	57
6.2.5 Mail-Merge	58
6.2.6 Alphabetical Keyword Listing Wizard	59
6.3 List Configuration Task Wizard	59
6.3.1 Administrators	60
6.3.2 Security	61
6.3.3 Attachments	65
6.3.4 Probes	69
6.3.5 Digesting and Indexing	70
6.3.6 Subject Tags	73
6.3.7 CataList	75
6.3.8 Banners	77
6.3.9 Mail Templates	77
6.4 Subscriber Management	78
6.4.1 Adding a New Subscriber to the List	78
6.4.2 Examining or Deleting a Subscription	79
6.4.3 Reviewing List Members	83
6.4.4 Bulk Operations	83
6.5 Submitting LISTSERV Commands	85
Section 7 List Moderation Using the Web Interface	87
7.1 Accepting or Rejecting the Message	87
7.2 Viewing the Message	88
7.3 Subscriber Settings	88
Section 8 Generating List Management Reports in the Web Interface	89
8.1 List Reports	89
8.2 Subscriber Reports	89
8.3 List Activity Reports	91
Section 9 Customizing Mail and Web Templates Using the Web Interface	93
9.1 Forcing the Browser to Use a Specific Character Set	96
9.2 Using the HTML Mail Template Gallery	96
9.3 Using a Conditional Expression Evaluator	97
Section 10 Working with Archives in the Web Interface	99
10.1 The Server Archive Index	99
10.2 Viewing List Archives	101
10.3 The Archive Index	103
10.4 Viewing an Archived Message	104
10.5 Subscribing, Unsubscribing, or Updating a List from the Archives	106
10.6 UTF-8 Archive Searches	108
Section 11 Posting Messages to a List from the Web Interface	111
11.1 Composing the Message	111
11.1.1 Using the HTML Editor	116
11.2 Creating Customized HTML Newsletters	116
11.2.1 Selecting a Template	116
11.2.2 Defining the Content	117

11.2.3 Fine Tuning the Newsletter	120
11.2.4 Testing and Sending the Newsletter	123
Section 12 Starting a Mailing List Using Email	125
12.1 Avoiding Duplication	125
12.2 Skills Needed to Start and Maintain a LISTSERV Mailing List	126
12.3 Creating a Mailing List Manually	126
12.3.1 Naming Conventions	127
12.4 List Header Keywords	129
12.5 Sending Commands to LISTSERV	130
12.6 Defining Personal Passwords	130
12.7 Retrieving the List Configuration	131
12.7.1 Who can edit the list configuration?	131
12.7.2 Retrieving the List Configuration by Email	131
12.8 Editing the List Header	133
12.9 Defining List Owners	134
12.10 Storing the List on the Host Machine	135
12.11 Fixing Mistakes	135
12.12 Security Options	136
12.12.1 The VALIDATE= Keyword	136
12.12.2 Controlling Subscription Requests	137
12.12.3 Controlling the Service Area of the List	137
12.12.4 Controlling Who Reviews the List of Subscribers	138
12.12.5 Controlling Access to the Notebook Files	138
12.12.6 Controlling Who Can Post Mail to a List	139
12.12.7 The "OK" Confirmation Mechanism	140
12.12.8 Explicitly Cancelling "OK" Cookies	142
12.12.9 Restricting Subscriber Privileges	143
12.12.10 Restricting the Number of Postings Per User Per Day	143
12.13 Setting Up Lists for Specific Purposes	144
12.13.1 Public Discussion Lists	144
12.13.2 Private Discussion Lists	145
12.13.3 Edited Lists	145
12.13.4 Moderated Lists	147
12.13.5 Semi-Moderated Lists	149
12.13.6 Self-Moderated Lists	149
12.13.7 Private Edited/Moderated Lists	149
12.13.8 Auto-Responders	150
12.13.9 Announce-Only Lists	151
12.13.10 Restricted Subscription Lists with Automatically-Generated Questionnaire	152
12.13.11 Peered Lists	153
12.13.12 Super-Lists and Sub-Lists	156
12.13.13 Cloning Lists	158
12.14 List Passwords are now Obsolete	159
12.15 Allowing/Blocking MIME Attachments	160
12.16 Content Filtering	160
12.17 DomainKeys Message Signing	162

Section 13 Managing Subscriptions through Email	165
13.1 Adding and Deleting Subscribers to/from a List	165
13.1.1 Adding Users Whose Address and Read Name Exceed 80 Characters	165
13.1.2 X.400 and X.500 Addressing – Special Problems	166
13.1.3 Continuation Card Syntax	167
13.2 Finding Users Who Do Not Appear in the List	167
13.3 Converting Existing Lists from Other Systems to LISTSERV	168
13.3.1 Converting Mailing Lists	168
13.3.2 Converting Message Archives	168
13.4 Adding Subscribers to Lists in Bulk	170
13.5 Deleting Subscribers from Lists in Bulk	170
13.6 Using the QUIET Option with Commands	171
13.7 Dealing with Bounced Mail	171
13.7.1 What is a bounce, and what can typically cause one?	171
13.7.2 The Owner-Listname Address	172
13.7.3 What to do about several types of bounces	172
13.7.4 Redistribution and Forwarding	176
13.7.5 "Sender:", "From:", or "Reply-To:" Fields in Body Causes Bounce	176
13.7.6 LMail Error Codes	177
13.8 Delivery Error Handling Features	179
13.8.1 Auto-Delete Considerations for Holidays	179
13.9 Address Probing	180
13.9.1 Active Address Probing	180
13.9.2 Passive Address Probing	181
13.10 Subscription Confirmation	181
13.11 Subscription Renewal	182
13.12 Using the SERVE Command When a User is "Served Out"	183
Section 14 Setting Subscription Options for Subscribers Using Email	185
14.1 Reviewing Current Subscription Options with QUERY	185
14.2 Setting Personal Subscription Options for Subscribers	185
14.3 Subscription Options	186
14.3.1 Mail/NOMail	186
14.3.2 DIGest/NO DIGest	186
14.3.3 MIME/NO MIME	186
14.3.4 INdEx/NO INdEx	187
14.3.5 ACK/NOACK/MSGack	187
14.3.6 Options for Mail Headers of Incoming Postings	187
14.3.7 Putting the List Name into the Subject: Field	188
14.3.8 CONCEAL/NO CONCEAL	188
14.3.9 REPro/NO REPro	188
14.3.10 TOPICS	188
14.3.11 POST/NO POST	189
14.3.12 EDITOR/NO EDITOR	189
14.3.13 REVIEW/NO REVIEW	190
14.3.14 RENEW/NO RENEW	190

14.4 Setting Original Default Options with the Default-Options= Keyword	190
14.5 Setting the Misc-Options= Keyword	191
14.5.1 DISCARD_HTML	191
14.5.2 IETFHDR_SUBJECT_TAG	191
14.5.3 IGNORE_EMAIL_CASE RESPECT_EMAIL_CASE	192
14.5.4 KEEP_DKIM_SIGNATURE	192
14.5.5 KEEP_EXCHANGE_DATA	193
14.5.6 NO_DKIM_SIGNATURE	193
14.5.7 NO_RFC2369	193
14.5.8 NO_SPAM_CHECK	193
14.5.9 RESPECT_EMAIL_CASE	193
14.5.10 SUBJECTHDR_SEQUENCE	193
14.5.11 SUPPRESS_APPROVED_BY	194
Section 15 Moderating and Editing Lists through Email	195
15.1 List Charters, Welcome Files, and Administrative Updates	195
15.2 The Role of the List Owner as Moderator	195
15.3 The Role of the List Owner as Editor	196
15.4 Setting Up an Edited List	197
15.5 Submitting Subscriber Contributions to an Edited List	199
15.6 Message Approval with Send= Editor, Hold	199
15.7 Using List Topics	200
15.8 The <listname> WELCOME and <listname> FAREWELL Files	202
15.8.1 Creating and Storing the listname WELCOME and FAREWELL Files	202
15.8.2 Using the listname WELCOME File as a Moderation Tool	203
15.8.3 Using the listname FAREWELL File as a Feedback Tool	204
15.8.4 The Alternative to Using WELCOME and FAREWELL Files	204
15.9 Social Conventions (“Netiquette”)	204
15.9.1 Recognizing and Accepting Cultural and Linguistic Differences	204
15.9.2 Private Mail Should Dictate Private Responses	205
15.9.3 Flaming is (Usually) Inappropriate	205
15.9.4 Foul Language	205
15.9.5 Unsolicited Advertising and Chain Letters	205
15.9.6 Other Disruptive or Abusive Behavior	205
15.10 Spamming: “Don’t Feed the Trolls”	205
15.11 Considerations for Appropriate Use Policies	206
Section 16 Managing Archives using Email	207
16.1 List Archives	207
16.1.1 Setting Up Archive Notebooks	207
16.1.2 Indexing Available Archive Notebooks	207
16.1.3 Deleting Existing Archive Notebooks by Email	207
16.2 File Archives	208
16.2.1 What is the File Archive?	209
16.2.2 Starting a File Archive for your List	209
16.2.3 Filelist Maintenance (VM Systems Only)	209
16.2.4 The listname.CATALOG System on non-VM Systems	212

16.2.5 Storing Files on the Host Machine	214
16.2.6 Deleting Files from the Host Machine	215
16.2.7 Automatic File Distribution (AFD) and File Update Information (FUI)	216
16.2.8 File "Packages"	216
16.2.9 Where to Find More information on File Archives	217
Section 17 Using Email to Post a Message	219
Section 18 Working with Mail and Web Templates Using Email	221
18.1 Using LISTSERV Templates	221
18.2 Types of Templates	221
18.2.1 Mail Templates	221
18.2.2 Web Templates	222
18.3 The Default Charset for Mail Templates	223
18.4 Naming Conventions for Message Templates and Fragments	223
18.5 Accessing the Default Template Files	223
18.5.1 Requesting a Copy	223
18.6 Mail Template Format and Embedded Formatting Commands	224
18.6.1 Mail Template Format	224
18.6.2 Common Variable Substitutions	224
18.6.3 Template Commands	226
18.6.4 Conditional Processing	229
18.6.5 The .QUIF Command	231
18.6.6 Using 8-Bit Characters in Templates	231
18.7 Editing List-Level Default Templates	231
18.7.1 The INFO Template Form	232
18.7.2 DEFAULT MAILTPL Templates	233
18.7.3 Tips for Using Templates	234
18.8 Using the DAYSEQ(n) Function	235
18.8.1 Rotating Bottom Banner	235
18.8.2 Rotating FAQ via the PROBE1 Template and "Renewal= xx-Daily"	235
18.8.3 Calculating the Value for DAYSEQ()	236
18.9 Storing the <listname>.MAILTPL File on the Host Machine	237
18.10 DIGEST-H and INDEX-H Template Files	237
18.11 WWW Interface Templates and Template Forms	238
18.11.1 Web Forms (Static) Contained in DEFAULT MAILTPL	238
18.11.2 The WWW_ARCHIVE.MAILTPL File	238
18.11.3 The DEFAULT.WWWPTL File (Dynamic Templates)	239
18.11.4 The SITE.WWWTPL File	239
18.11.5 National Language Template Files (idiom.mailtpl)	239
18.11.6 Template Precedence	240
18.12 Serving Up Custom Web Pages for your List	241
18.12.1 A Practical Example: ADMIN_POST	241
18.13 Using a Conditional Expression Evaluator	242
Section 19 Solving Problems	243
19.1 Helping Subscribers Figure Out the Answers	243
19.2 Loop-Checking Can Cause Occasional Problems with Quoted Replies	243

19.3 User Can't Unsubscribe and/or Change Personal Options	245
19.4 Firewalls	245
19.5 LISTSERV Won't Store Your List	246
19.6 If I can't find the answer, where do I turn?	246
Section 20 Advertising Your Public Mailing Lists	247
20.1 List of Lists Maintained by LISTSERV	247
20.2 Adding HTML to a List Header for the CataList	247
20.2.1 Update latency	248
20.2.2 Inserting a Pointer to Another List	248
20.2.3 Restrictions on the Placement of Equal Signs	249
20.3 Defining Search Categories in a List Header for the CataList	249
20.3.1 Examples of Category Settings	250
20.4 Implementing the INFO <listname> Command	251
20.5 The Global List Exchange (GLX)	251
20.6 How NOT to Advertise a Mailing List	252
Appendix A: LISTSERV Commands	253
General Commands	253
List Subscription Commands	253
Other list-related commands	257
Informational commands	260
Commands Related to File Server and Web Functions	262
Other Advanced Commands	264
List Owner and File Owner Commands	267
File Management Commands (for file owners only)	267
List Management Functions	268
LISTSERV Maintainer Commands	272
Commands for LISTSERV on VM Servers Only	274
Sending commands to LISTSERV	275
Defining Personal Passwords	276
Appendix B: Sample Boilerplate Files	277
Subscription Requests Sent to the List	277
Sending Other Commands to the List or to the *-REQUEST Address for the List	277
Unsubscribed User Still Getting Mail	278
Quoted Replies Include Message Headers Causing them to Bounce	278
Delivery Error with Unknown User Account	279
Setting a User to DIGEST because of Bouncing Mail	279
A Sample "Your List has been Created" Boilerplate	280
Appendix C: Country Codes	281
Index	285

List of Figures



One-on-One Email Interaction	3
Email Lists Interaction	3
Path of LISTSERV Message	4
People Who Can Interact with LISTSERV	5
Example of an Email Sent to LISTSERV	8
Example of an Email Sent to the List	8
Example of a List Header	10
Example of an “OK” Confirmation Message	11
Login Screen	22
Registering a LISTSERV Password	23
Changing the LISTSERV Password	23
The Toolbar	24
The General Preferences Tab	25
The Archive Preferences Tab	27
The Report Preferences Tab	29
Setting Your Newsletter Profile Preferences	30
Edit Mode - Top Section	31
Edit Mode - Bottom Section	32
CataList	33
Searching for a List	34
List Management Dashboard - Top Half	36
List Management Dashboard - Bottom Half	37
The List Management Menu	37
The List Configuration Wizard - Descriptions Tab	39
The List Configuration Wizard - Access Control Tab	40
The List Configuration Wizard - Distribution Tab	45
The List Configuration Wizard - Error Handling Tab	46
The List Configuration Wizard - List Maintenance Tab	49
The List Configuration Wizard - Security Tab	51
The List Configuration Wizard - Subscription Tab	52
The List Configuration Wizard - Other Tab	54
Manual List Configuration	55
The Alphabetical Keyword Listing Wizard	59
Example of an HTML Digest	73
Hiding a Header from CataList	76
Subscriber Management	78
Adding a New Subscriber	78
Examining or Deleting a Subscription	79
Select Subscriber to Examine or Delete	79
Subscriber Management Screen	82
The Bulk Operations Tab	84

Clearing the Command Interface	85
Moderated List Table	87
Message View	88
The Server Archive Index	99
Previewing a List	100
List Archives Screen	101
Previewing Latest Messages on the List Archives Screen	102
Archives Index	104
The Archive Browsing Screen	105
Joining, Leaving, or Updating a List from the Archives	106
The Send Message Screen - Initial View	111
The Send Message Screen - Advanced Options View with Pending Messages	114
Drafts Table on the Send Message Screen	115
Selecting a Newsletter Template	117
Defining the Newsletter Content	118
Re-Defining Placeholder Colors	120
Fine Tuning the Newsletter - Advanced Options	122
Fine Tuning the Newsletter - HTML Editor	123
Testing and Sending the Newsletter	124
Sample Output of LISTS GLOBAL IBM	125
A Sample List Header File for a List Called MYLIST	133
The Edited List Header File	134
Defining List Owners in the List Header File Example	135
The Editor-Header for a List Set to Send= Editor, Hold	139
A Typical Command Confirmation Request	141
Example of the "Editor-Header" Prepend By Default	199
Sample WELCOME File	203
FAREWELL File Used as a Feedback Tool	204
Deleting an Archive Notebook File by Email	207
Sample Filelist Retrieved with (CTL Option)	210
Adding a File Descriptor to the Filelist	211
Default Content of the INFO Template form for DEFAULT.MAILTPL	232
Sample of an Edited INFO Template Form	233
Typical Contents of a DIGEST-H or INDEX-H File	237
Sample DIGEST Output for a List with a DIGEST-H File	238
Sample of an Error Message with Headers Included	244
Another Sample of an Error Message with Headers Included	245
A Correctly Formatted Message with Headers Included	245

List of Tables

Keywords for an Announcement Style List	14
Other Keywords to Consider for an Announcement Style List	14
Templates to Customize for an Announcement Style List	15
Keywords for a Discussion Style List	16
Other Keywords to Consider for a Discussion Style List	16
Templates to Customize for a Discussion Style List	17
Keywords for a Moderated Style List	18
Other Keywords to Consider for a Moderated Style List	19
Templates to Customize for a Moderated Style List	19
Templates to Customize	94
Formatting Commands for List Owners	226
Advanced Formatting Commands for List Owners	227
Sample Output of an <i>INDEX Listname</i> Command	258
Country Codes	281

Preface - About This Manual

Every effort has been made to ensure that this document is an accurate representation of the functionality of LISTSERV®. As with every software application, development continues after the documentation has gone to press so small inconsistencies may occur. We would appreciate any feedback on this manual. Send comments via email to:

MANUALS@LSOFT.COM

The following documentation conventions have been used in this manual:

- Menus, options, icons, fields, and text boxes on the screen will be bold (e.g. the **Help** icon).
- Clickable buttons will be bold and within brackets (e.g. the **[OK]** button).
- Clickable links will be bold and underlined (e.g. the **Edit** link).
- Directory names, commands, and examples of editing program files will appear in Courier New font.
- Some screen captures have been cropped and/or edited for emphasis or descriptive purposes. This symbol denotes an important note or warning.



- This symbol denotes optional advice that can help you save time.



- This symbol denotes a new feature for LISTSERV 16.0.

Editorial Note - New Version Numbering

With this release, L-Soft is aligning LISTSERV's version numbering with the rest of the e-mail industry. There have been 51 released versions of LISTSERV since 1986 – 15 major upgrades and 36 minor releases. Version 1.8e in the “traditional” numbering system corresponds to 14.0. The present update is version 16.0.

Because the old nomenclature is more familiar to our users, in this version of the documentation we will continue to refer to versions of LISTSERV inferior to version 14.4 by the old version system.

Unicode (UTF-8) Support, Phase 1



A major new feature in LISTSERV 16.0 is the implementation of the first phase of Unicode support. LISTSERV's Unicode support is based on UTF-8 encoding. The long-term goal is for everything in LISTSERV to be a UTF-8 encoded string unless explicitly tagged otherwise. Because this is not a simple implementation, and because going to UTF-8 will undoubtedly break some customer scripts and applications, the road to full support for UTF-8 will be an ongoing, careful journey.

The first phase of the implementation revolves around these areas: the template processor, the message (job response) code, archive searches, and individual commands with support for UTF-8 (for now, only the LIST command is affected). The next phases will continue to roll out with the upcoming versions of LISTSERV.

LISTSERV Command Syntax Conventions

Generally, parameters used in this document can consist of 1 to 8 characters from the following set:

A-Z 0-9 \$#@+_-:

Deviations from this include:

- *fformat* – Netdata, Card, Disk, Punch, LPunch, UUencode, XXencode, VMSdump, MIME/text, MIME/Appl, Mail.
- *full_name* – *first_name* [*middle_initial*] surname (not your email address). Must consist of at least two space-separated words, e.g., "John Doe".
- *listname* – name of an existing list
- *node* – Either the fully-qualified domain name (FQDN) of an Internet host or the BITNET nodeid or Internet hostname of a BITNET machine which has taken care of supplying an 'internet' tag in its BITEARN NODES entry.
- *host* – Generally the same as node, but normally refers specifically to the fully-qualified domain name (FQDN) of an Internet host rather than to a BITNET nodeid.
- *pw* – a password containing characters from the set: A-Z 0-9 \$#@_-?!|%
- *userid* – Any valid RFC822 network address not longer than 80 characters; if omitted, the 'hostname' part defaults to that of the command originator.
- *internet_address* – Similar to *userid*, but specifically refers to a complete RFC822 network address in *userid@fqdn* format. When we use this nomenclature a fully-qualified hostname is required.

Other deviations from the standard set will be noted along with the affected commands.

Also, the following conventions represent variable or optional parameters:

- *italic type* – Always indicates required parameter names that must be replaced by appropriate data when sending commands to LISTSERV.
- *< >* – Angle brackets may sometimes enclose required parameter names that must be replaced by appropriate data when sending commands to LISTSERV. Sometimes used for clarity when italic type is inappropriate.
- *[]* – Square brackets enclose optional parameters which, if used, must be replaced by appropriate data when sending commands to LISTSERV.

Contacting L-Soft

Support

L-Soft international recognizes that the information in this manual and the FAQ questions on our web site (<http://www.lsoft.com/lsv-faq.html> or <http://www.lsoft.com/manuals/owner-faq.html>) are not going to solve every problem you may face. We are always willing to help diagnose and correct problems you may be having with your licensed LISTSERV server.

L-Soft strongly recommends that, for support purposes, it is best to use the technical support "lifebuoy" link from the Server Administration Dashboard to initiate a support ticket. This will help you create an email message to the support group that contains all the necessary information about the site configuration, license and so forth without requiring you to find the individual files or issue information commands.

If LISTSERV is not running, of course, this will not be possible. In that case, please try to use the following procedure:

- Make the subject line of your report indicative of the problem. L-Soft receives a great deal of mail with the subject "Help!", which is not very helpful when we receive them.
- Include any appropriate log entries. LISTSERV keeps logs of everything it does, and without the log trace back, it is often impossible to determine what caused a given error.
- If you're running a Unix server and LISTSERV dumps core, please run the debugger on the core file, produce a trace back, and include the results.
- Always send a copy of your site configuration files (with the passwords x'ed out). See the [Site Manager's Operations Manual](#) for LISTSERV for the locations and names of the two site configuration files.
- Send along anything else that you think might be helpful in diagnosing the problem.

If the supporting documents (for instance, log files) are extremely large, please contact support first before sending everything through. The support group has alternative methods of handling large files that they will be happy to share with you.

If you are not currently an L-Soft customer and are running an evaluation version of our software, please send your trouble reports to the evaluation users' list, LTSERV@PEACH.EASE.LSOFT.COM.

If you are running LISTSERV Lite, please send your trouble reports to the LISTSERV Lite support mailing list, LTSERV-LITE@PEACH.EASE.LSOFT.COM. This includes users of the paid version of the software unless you have also purchased paid support.

If your LISTSERV Classic/Classic HPO server for VM, VMS, unix, or Windows has paid-up maintenance, you may send problems to SUPPORT@LSOFT.COM for a quick reply.

Sales

To reach our worldwide sales group, simply write to SALES@LSOFT.COM. You may also call 1-800-399-5449 (in the US and Canada) or +1 301-731-0440 (outside the US and Canada) to speak to our sales representatives.

Section 1 Getting Started

If you are a new list owner, working with your first list, here are the recommended steps for getting started:

1. Read through Section 3 [Communicating with LISTSERV](#) to gain a general understanding of the concepts of LISTSERV commands, header keywords, and communicating with LISTSERV.
2. Decide on the type of mailing list you want: one-way, two-way discussion, or moderated discussion. See Section 4 [Mailing List Types](#) for more information if you are not sure.
3. Request your list from the site administrator. Depending on the procedures and arrangements provided at the individual site, the site administrator may configure your list to your specifications, or may simply create the list from the built-in list template that is closest to what you requested. This leaves the fine-tuning to you. You might only be provided with a “bare-bones” list with only the Owner and Notebook keywords defined, leaving the bulk of the configuration to you.
4. Obtain a LISTSERV password on the list’s server. See Section 5.3 [Setting a LISTSERV Password](#).
5. Define or adjust the list configuration. Even if your site administrator configured the list to your exact specifications, understanding how it is set up will benefit you by saving time and effort as you become a more experienced list owner. Use the Web Interface wizards to guide you:
 - a. Go to the List Configuration Wizard, and check the keywords in each category and consider whether or how they apply to your list.
 - b. Go through the List Configuration Tasks, and consider whether any of them are of interest.
 - c. Alternatively, read through the [List Keyword Reference](#) document, concentrating especially on the keywords listed for your type of list in Section 4 [Mailing List Types](#), and edit the list header using email or the Web interface list header editor (for details on using the Web Interface, see Section 6.2 [List Configuration](#)). A few keywords require special privileges to modify, so you may need to get assistance from the site administrator.
6. Consider which administrative email messages, if any, you want to customize. Modify their templates (see Section 9 [Customizing Mail and Web Templates Using the Web Interface](#)). The descriptions in Section 4 [Mailing List Types](#) list the templates that you are most likely to be interested in customizing for each list type.
7. Test your list. Subscribe to your list from one or more email addresses that do not have any special privileges on the list (not listed as Owner, Editor, or Moderator). Send test messages to the list. Try sending from editor, owner, subscriber, and non-subscriber addresses. Signoff (leave the list) from at least one address. Practice

performing common tasks. Does the list behave the way you expect it to? Do the messages look the way you want? If not, adjust the list's configuration and templates accordingly. When you are done testing, remember to remove the test addresses and the archives of the test messages, if necessary.

8. If you have an initial set of addresses to start the list with, create a text file and upload it into the list or add the addresses one at a time (see Section 6.4.1 [Adding a New Subscriber to the List](#)). Do not forget to add yourself if you want to receive list messages.
9. If desired, publicize your list in whatever forum is appropriate (for example, your own Web page, CataList, or in your print advertising). For details, see Section 20 [Advertising Your Public Mailing Lists](#).
10. Start using your list. Enjoy!

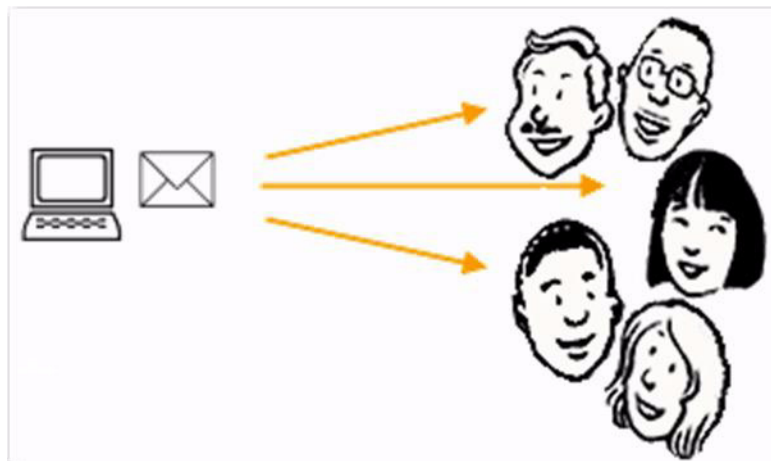
Section 2 About Mailing Lists and LISTSERV

The widespread use of electronic mail (email) has become an efficient, convenient, and economical way for people from all over the world to communicate. While traditional email is ideal for one-on-one interaction, email lists advance the concept a step further by disseminating a single message simultaneously to a group of people. Fast, global, and cost effective mass communication is easy with the use of email lists.

Figure 2-1 One-on-One Email Interaction



Figure 2-2 Email Lists Interaction



With email lists, companies and institutions can quickly and cost-effectively deliver thousands, even millions, of direct email messages simultaneously over the Internet. Furthermore, through database integration, messages can be personalized according to each recipient's demographic information and preferences. Newsletters and announcements can be conveniently distributed by email to patrons, customers, distributors, and resellers. Email discussion groups can be used for interaction among co-workers in special projects and workgroups, or to keep lines of communications open between individuals scattered throughout the world working towards a common goal. Individuals can form non-commercial discussion forums on just about any imaginable topic of interest, from hobbies to health support groups. The possibilities of how email lists can be used for communication among groups of people are virtually endless. See the Case Studies page on L-Soft's Web site for real-life examples:
<http://www.lsoft.com/customers/casestudies.asp>

2.1 What is LISTSERV®?

LISTSERV is an email list management software program. Email list management software makes it easy to administer email lists, performing in seconds the functions that would otherwise have to be done manually, possibly taking many hours or days to complete. Introduced in 1986 on BITNET, LISTSERV was the first software program to automate the administration of email lists. It is the most time-tested and reliable email list management product on the market today.



Note: LISTSERV is a registered trademark licensed to L-Soft international, Inc.

In order to use LISTSERV, the software must be installed on a server with a dedicated connection to the Internet. LISTSERV is available for a variety of server operating systems.

Once LISTSERV is installed on a server and running, it can:

- Handle list subscriptions and signoffs (leaving the list).
- Allow subscribers to alter their subscription settings, or owners to alter them on the subscriber's behalf.
- Distribute email to all list subscribers.
- Accumulate messages to be sent as periodic "digests".
- Save list messages in a searchable archive (sometimes referred to as "notebooks").
- Automatically handle communication between members and owners of lists.
- Report on the activity of mailing lists.
- Allow list owners to control the character and behavior of their list(s).
- Automatically process bounces (message delivery errors).
- Be used with an external database to send targeted, personalized e mail messages.

When a list is set up, an email address is created for the list, which is then used to send or "post" messages to the list. Once a message is sent, it is delivered to the central server where L-Soft's LISTSERV (the mailing list management software) resides. The software then completes the operation by automatically distributing the message to all subscribers on the list, in the manner indicated by their subscription options.

Figure 2-3 Path of LISTSERV Message



2.2 Who are the people involved?

There are five main classifications of people who can interact with LISTSERV. Each classification has a differing level of responsibility for its role and interaction with the software. Figure 1-4 below illustrates the hierarchy of these classifications. It also represents the relative number of people who fit into each classification.

Figure 2-4 People Who Can Interact with LISTSERV



The **LISTSERV maintainer** is the technical person in charge of the LISTSERV application on a particular server. The LISTSERV maintainer may not be in charge of the entire server, but should have privileged access to the computer and know where to get help if there is a problem with the hardware, software, or network connection. The LISTSERV maintainer:

- Oversees the administration of dozens or even hundreds of different lists, but is usually not involved with the content of individual lists.
- Has the responsibility for creating new lists and assigning list ownership.
- Is implicitly an owner of all lists – has access to all list owner functions, all archives, and log files.
- Allocates disk space and sets the path and folders to store information related to the lists.
- Provides assistance to owners of the lists on the server.
- Ensures that list owners abide by the site's appropriate use policies.
- Ensures that LISTSERV is up and functioning correctly, with a working connection to the Internet.

The **list owner** is the person formally in charge of the operation of a specific list. The list owner is usually knowledgeable in the field covered by the list. There can be more than one list owner for a list. The responsibility of the list owner is limited to the list itself, and does not include the computer running the list, its mail system, network lines, and other hardware specifics. List owners:

- Set and change the configuration of the list.
- Add, delete, and modify subscriptions to the list.
- Assist subscribers and potential subscribers.
- Set policies, procedures related to the list, and send administrative messages to the list.
- Run reports on list activity.

A **list moderator** may or may not be the list owner as well. There can be more than one list moderator for a list, and some lists have none. Moderators receive incoming messages sent to a moderated list and determine whether they should be distributed to the list. List moderators:

- Approve messages to be posted to the list.
- Stop inappropriate messages from being sent to the list.
- May edit the content of messages before sending them to the list.

List subscribers are people who have requested and have been granted permission to join a particular mailing list. Subscribers:

- Can leave a list.
- Can set the way they receive list postings or change the name or email address associated with their subscription, within the configuration limits set by the list owner.
- May be permitted to search and view the archives.
- May be permitted to send messages to the list, depending on the type of list.

Non-subscribers include everyone not subscribed to a list. Non-subscribers:

- Can make a request to LISTSERV to be subscribed to a particular list.
- Depending on the list configuration, they may or may not be allowed to send messages to a list or read its archives.

Section 3 Communicating with LISTSERV

LISTSERV was created before the World Wide Web was in general use. Back then, the primary way for most people to communicate with the LISTSERV program was by email, and this method is still widely used today. LISTSERV also includes a Web Interface to make communication easier and more intuitive. The Web interface is richly populated with on-screen information, tutorials, and wizards to help you use the program quickly and effectively.

When you want to perform actions using LISTSERV, such as adding a subscriber to a list, the action is based on a command LISTSERV receives by email or using the Web interface. Some commands are only available to LISTSERV maintainers and list owners, while other commands are available to subscribers and non-subscribers as well.

3.1 Using the Web Interface to Communicate with LISTSERV

Sending email messages to LISTSERV containing commands and posting messages to the list is sometimes confusing for people new to mailing lists. To simplify this process, the Web interface provides a centralized location for interaction with LISTSERV. As a list owner, you can use the Web interface to issue commands directly to LISTSERV. You can also use the Web interface to post messages to the list. For details on using the Web Interface to create and maintain your lists, see the following sections:

- Section 5 [Introduction to the Web Interface](#)
- Section 6 [List and Subscriber Management Using the Web Interface](#)
- Section 7 [List Moderation Using the Web Interface](#)
- Section 8 [Generating List Management Reports in the Web Interface](#)
- Section 9 [Customizing Mail and Web Templates Using the Web Interface](#)
- Section 10 [Working with Archives in the Web Interface](#)
- Section 11 [Posting Messages to a List from the Web Interface](#)

3.2 Using Email to Communicate with LISTSERV

There are two main email addresses that are used to work with LISTSERV lists. One is to communicate with the LISTSERV program — a “command address”, and one is used to post mail to the list — the “list address”. Any time the list configuration is changed or a subscription setting is changed, communication takes place with the LISTSERV program using the command address. Whenever a message is posted to the list, it is sent to the list address.

The command address will usually start with “LISTSERV@” and will be followed by the server name where the LISTSERV program has been installed. Messages sent to the command address should contain only commands to LISTSERV, one command per line. Lines containing non-commands will result in an error message being returned.



Tip: If sending commands from an email client that uses a “signature file”, make sure that the first line of the signature file is a line containing only “--” (two consecutive hyphens and a space) starting at the very start of the line. Otherwise, LISTSERV will

attempt to interpret the signature text as commands and return “Invalid command” errors. This is the standard for the first line of a signature.

Figure 3-1 Example of an Email Sent to LISTSERV

Date: Tue, 31 Dec 2002 10:35:35	<i>All email messages must have a date.</i>
From: beth_23@xyz.com	<i>This is the list owner's email address.</i>
Subject:	<i>The Subject is ignored by LISTSERV</i>
To: listserv@listserv.example.com	<i>The email address that communicates with the LISTSERV program always begins with "LISTSERV@" because you are contacting the LISTSERV program.</i>
GET womensclub(header	<i>A command (GET) is followed by specific parameters for that command. In this case, the name of the file to "get" followed by the options: a single parenthesis "(" and the word "header", indicating that LISTSERV should send only the header part of the list to the list owner's email address.</i>

The list address will always start with the name of the list followed by the address of the server where the list resides.

Figure 3-2 Example of an Email Sent to the List

Date: Tue, 31 Dec 2002 23:59:59	<i>All email messages must have a date.</i>
From: pres_woman@hotmail.com	<i>The sender must have permission to post.</i>
Subject: Next Meeting	<i>A posting should have a subject line.</i>
To: womensclub@listserv.example.com	<i>The email address that sends mail to the list of subscribers always starts with the name of the list.</i>
Dear Members,	
The next meeting of the women's club will be on January 27th at 8:00 PM. We will meet at the public library, 123 Oak Street.	<i>The body of the email message contains the information that is sent to the subscribers.</i>
The agenda is posted on our Web site at www.example.org. Don't forget to fill out your survey for the new year's activities.	
Your President	

For a list of LISTSERV commands, their descriptions, and how to issue them, see the [List Keyword Reference](#) document.



Email Command: You can also get a listing of commands by sending an email message to your LISTSERV server with the following command: INFO REFCARD

For more details on using email to create and maintain your lists, see the following sections:

- Section 12 [Starting a Mailing List Using Email](#)

- Section 13 [Managing Subscriptions through Email](#)
- Section 14 [Setting Subscription Options for Subscribers Using Email](#)
- Section 15 [Moderating and Editing Lists through Email](#)
- Section 16 [Managing Archives using Email](#)
- Section 17 [Using Email to Post a Message](#)
- Section 18 [Working with Mail and Web Templates Using Email](#)
- Section 19 [Solving Problems](#)

3.3 Anatomy of a LISTSERV List

A LISTSERV list consists of the list's configuration (header) and any number of subscriptions. Each subscription includes an email address, a name, and the subscription "settings" or "options". The list's configuration regulates list behavior that applies to the entire list, whereas subscription settings regulate behavior that may be different for each subscriber.

A list's configuration is defined by a set of "keywords" arranged in a format called a "List Header." The keywords contained in the list header have values associated with them. The values of the keywords determine the behavior of the list. For example, a list header that contains the keyword "Attachments" with the value set to "No" will not allow any messages that include attachments to be sent to the list. Keywords can be added, removed, or have their values changed in a list header by sending a command to LISTSERV to replace the header.

Keywords and their associated values set the behavior of the list. This behavior in turn sets the "style" of the list. For example, the keyword "Send" and its associated value determine who is allowed to post messages to the list. A list that has the keyword "Send" set to "Owner" allows only the owner of the list to post messages to the subscribers. This creates a one-way communication style for the list – the owner can send messages to the subscribers, but the subscribers cannot send messages to the list for distribution.

A list of this type might be used for newsletters or product announcements. A list with the keyword "Send" set to "Private" would allow the list owner and the subscribers to that list to send messages to the list, setting up a two-way communication style list. This type of list could be used for collaboration or discussion among the subscribers. See Section 4 [Mailing List Types](#) for descriptions of the three most common types of mailing lists.

There are more than 50 keywords that can set the behavior of mailing lists. Keywords can be added, removed, or have their values changed by using the Web interface or by sending an email message to the LISTSERV program using the command address.

Figure 3-3 Example of a List Header

<pre> PUT WC(HEADER PW=XXXXXX * * Women's Club * * .HH ON * * * Subscription= By_Owner * Ack= Yes * Confidential= Yes * Validate= Yes,Confirm * Reply-to= List,Respect * Send= Private * Errors-To= Owner * Owner= beth_23@xyz.com * Notebook= Yes * Notebook= E:\LISTS\WC * Notebook= Weekly,Private * * This template creates the * "classic" public discussion * list. Typically it will have * public archives, only * subscribers can post to the * list and replies * will go back to the list. * Only list owners may review * the list of subscribers. * * See Section 2 in the List * Owner's Manual for more * information. * * * .HH OFF * * This mailing list is for the * members of the Women's Club. </pre>	<p>The "PUT" command is used to replace the list header over email.</p> <p>The first non-blank line in the header is the title of the list.</p> <p>The "Hide Header" (.HH ON) directive is used so no one but the list owner will see the following header lines.</p> <p>Keywords in the list header and their respective settings or values determine the behavior of the list. The keyword is followed by an "=" (equal sign). Anything to the right of the "=" denotes the setting or value for that keyword.</p> <p>When a keyword has several values, they are separated by commas, or can be continued over several lines by repeating the same keyword.</p> <p>Any text that is not a keyword definition or a directive (line starting with a ".") is considered a comment and ignored by LISTSERV.</p> <p>A description of the template used to create a particular type of list may be present. This text is explanatory only. It is not used by LISTSERV, does not affect the behavior of the list in any way, and may be removed by the list owner if desired.</p> <p>When updating the list header using e mail, each line must start with an asterisk ("*").</p> <p>".HH OFF" turns the Hide Header directive off so that any part of the header that follows is viewable by the public</p> <p>A description of the list viewable by anyone has been added by the list owner.</p>
--	--

3.4 The "OK" Confirmation Method

For the sake of security, there are a number of actions for which LISTSERV requires confirmation before proceeding. In some cases, LISTSERV will accept a password-based validation. In other cases, email confirmation is required. When this happens, LISTSERV sends an email message with a subject line such as:

Subject: Command confirmation request (787EF897)

The string of "hexadecimal" numbers in parentheses ("787EF897" in the example) is called a "cookie" (sometimes referred to as a "confirmation code") and is different every time. The text of the message will look something like this:

Figure 3-4 Example of an "OK" Confirmation Message

```

Your command:

        quiet delete listname *@*

has been received. For security reasons, you are now required to reply to
this message, as explained below, to confirm the execution of your command.
Note that the security level of the list is under list owner control, and
that is the person you should contact if you have any complaint about secu-
rity procedures.

To confirm the execution of your command, simply point your browser to the
following URL:

    http://home.ease.lsoft.com/scripts/wa.exe?OK=787EF897&L=LISTNAME

Alternatively, if you have no WWW access, you can reply to the present mes-
sage and type "ok" (without the quotes) as the text of your message. Just
the word "ok" - do not retype the command. This procedure will work with any
mail program that fully conforms to the Internet standards for electronic
mail. If you receive an error message, try sending a new message to LIST-
SERV@HOME.EASE.LSOFT.COM (without using the "reply" function - this is
very important) and type "ok 787EF897" as the text of your message.

Finally, your command will be cancelled automatically if LISTSERV does not
receive your confirmation within 48h. After that time, you must start over
and resend the command to get a new confirmation code. If you change your
mind and decide that you do NOT want to confirm the command, simply discard
the present message and let the request expire on its own.

----- Original mail header -----
Return-Path: <yourname@EXAMPLE.COM>
Received: from mail.example.com by home.ease.lsoft.com (Windows v1.1b) with
SMTP id <9.FC988C9E@home.ease.lsoft.com>; Tue, 3 Dec 2002 17:16:06 -0500
Received: from yourbox by mail.example.com (Windows v1.1b) with SMTP id
<0.000007B9@mail.example.com>; Tue, 3 Dec 2002 17:16:06 -0500
From: "Your Name" <yourname@example.com>
To: listserv@home.ease.lsoft.com
Date: Tue, 3 Dec 2002 17:16:06 -0500
MIME-Version: 1.0
Content-type: text/plain; charset=US-ASCII
Content-transfer-encoding: 7BIT
Subject:

```

In the figure above, the owner sent an email command to delete all subscribers from a list that is configured such that it requires list-owner commands to be validated (Validate=Yes, Confirm). As the "delete" command did not include the owner's password, LISTSERV sent an email requesting confirmation of the command.

When you receive a command confirmation request, you must confirm within 48 hours (this time may be configured differently using the Confirm-Delay= keyword) in order for the command to be executed. There are two ways of confirming the command, using the Web interface or using email.

- To confirm with the Web interface, simply visit the link included in the message. The link sends the “OK” command to LISTSERV, providing the list name to which it applies and the cookie that was assigned to this command.
- To confirm with email, simply reply to the email message, keeping intact the Subject line (“Re:” or other prefixes may be added with no impact), in particular the cookie in parentheses at the end of the subject line. In the text of your reply, simply say “OK” (without the quotes) and nothing more.



Tip: If this method is not successful, then you can send a new message to the LISTSERV command address using the “OK xxxxxxxxx” command, where “xxxxxxx” is the cookie copied from the original confirmation request.

The cookie is the key to the “OK” confirmation method of validation. LISTSERV randomly generates a new cookie for each action that requires validation. Only someone with access to the email address to which LISTSERV sends the cookie knows what the cookie is. All of the privileges within LISTSERV are tied to an email address, so when LISTSERV needs to verify that a command was really initiated by the owner of a particular email address, it uses the “OK” confirmation method.



Caution: Never “OK” a cookie blindly. Make sure that it is for a command that you initiated or for a message that you want to have distributed to the list. Several cases of list “hijackings” or spam sent to well-secured lists have been traced back to a list owner or moderator absent-mindedly clicking an OK link that they should not have clicked.

Examples of actions that require an “OK confirmation”:

- Postings from a non-editor to a list with `Send= Editor, Hold`.
- Postings from a subscriber with the “REVIEW” subscription setting.
- Postings to a list with `Send=..., Confirm` (sender must confirm that the message came from them – strongly recommended for one-way lists, such as newsletters).
- Subscription requests for a list with `Subscription=Open, Confirm` or `Subscription=By_Owner, Confirm`. Subscription confirmations are strongly recommended on public lists, to prevent unwanted third-party subscriptions.
- Subscription and signoff requests made over the web without a login.
- `DELETE` or `ADD` commands sent over email without a password, on a list with `Validate=Yes, Confirm`.
- All subscription modifying commands, including `SET` and `SIGNOFF` commands from subscribers, on a list with `Validate=All, Confirm`.

As these examples illustrate, the list configuration controls most of the conditions under which confirmations are required. The list owner must decide on the right balance between security and convenience for the list. For more information, see the [List Keyword Reference](#) document.

Section 4 Mailing List Types

There are three main types or styles of mailing lists, each with its own advantages for specialized use. Setting specific keywords to certain values and then editing various templates creates different types of mailing lists. The style of list will best determine the particular keywords and their settings in your list header. Your LISTSERV maintainer will be able to create a list based on the style you want or need. When requesting a new list, make sure you tell your LISTSERV maintainer what style of list you want. The three major types of lists are:

- **One-Way Announcement** – The owner(s) and/or editor(s) of the list are the only people who are allowed to send messages to the subscribers. The communication flows one way – from the list administrators to the list subscribers. This type of list is primarily used for newsletters, product announcements, and dissemination of information that does not require feedback from the subscriber.
- **Two-Way Discussion** – The list administrators and the list subscribers are both permitted to post messages to the list. The communication flows two ways - between the administrators and the subscribers, and back and forth between different subscribers. This type of list is primarily used for discussion groups engaged in the exchange of ideas and information centered on a specific topic.
- **Moderated** – This type of discussion list has an editor or editors who review all incoming messages. The editor can then decide to allow the message to be sent to all the subscribers on the list, or not allow the message to be posted to the list. Moderated lists can be used to control the discussion by keeping off topic, inflammatory, or otherwise inappropriate messages from the reaching subscribers.

There are other types of lists, for example you can use a list as an “auto-responder”. The three types listed here are the most common.

4.1 One-Way Announcement Lists

One-way announcement lists or newsletters are types of lists where subscribers only receive information and do not interact with the other list members. This type of list is most commonly used for delivering news and media publications, company newsletters and any other types of announcements. For example, the “LISTSERV at Work” newsletter (see <http://www.lsoft.com/news/newsletter-us.asp>) is sent quarterly to keep the L-Soft community informed and up-to-date with the company's latest developments. One-way lists are similar to print newspaper subscriptions. They often have large numbers of subscriptions and people sign on and off the list as they wish. Set up an announcement list by using one of the following sets of keywords.

Table 4-1 Keywords for an Announcement Style List

Keyword and Suggested Value	Description
Send= Owner,Confirm	Only addresses listed in the Owner= keyword can post to the list. The sender is required to verify the post. Non-owner addresses attempting to post receive a rejection message.
Send= email-address,Confirm	Only the specified email address is allowed to post to the list. The sender is required to confirm the post. All other addresses attempting to post receive a rejection message.
Send= Editor,Hold,Confirm Editor= editor-address(es)	Only address listed in the Editor= keyword can post to the list without moderation. Editors are required to confirm their own postings. Any non-editor address attempting to post to the list will receive the MSG_POSTING_FORWARD_EDITOR message, and the posting will be forwarded to the first editor listed (or Moderator= addresses, if defined). The editor or moderator(s) must be careful not to OK these or they will be distributed.
Send= Editor,Hold,Confirm Editor= editor-address(es) Default-Options= NOPOST	Same as above, except that subscribers that attempt to post will receive a rejection message, and their post will not be forwarded to the editor or moderators. Non-subscriber posts will still be forwarded to the moderator, as above. If the editor addresses are subscribed, they must be set to "POST" before they will be allowed to post.

The following table contains some header keywords that you may want to consider setting in addition to those in the above table. Keywords appearing without a suggested value may have multiple settings and/or options to consider. For more information about assigning keyword values, see the [List Keyword Reference](#) document or the Keyword Wizard in the Web Interface.

Table 4-2 Other Keywords to Consider for an Announcement Style List

Keyword	Suggested Value
Auto-Delete=	Yes,Full-Auto (Set Max, Delay, and Probe values as appropriate for your list.)
Change-Log=	Yes
Confidential=	(Decide whether or not the existence of the list should be made public.)
Default-Options=	(Decide on the default subscription options for your list.)
Digest=	No (unless announcements are frequent)
Notebook=	(Decide whether or not to keep archives of posted messages, and decide who should have access to them.)

Keyword	Suggested Value
Reply-To=	Sender , Ignore
Subscription=	Open , Confirm
Validate	Yes , Confirm (Yes, Confirm is the minimum recommended protection. For stronger protection, see the List Keyword Reference document or the Keyword Wizard.)

You may also want to customize the following templates. For more information on customizing templates, see Section 9 [Customizing Mail and Web Templates Using the Web Interface](#).

Table 4-3 Templates to Customize for an Announcement Style List

Welcome Message	ADD1	TOP_BANNER_HTML
Farewell Message	DELETE1	BOTTOM_BANNER_HTML
\$SIGNUP	INFO	MSG_POSTING_REJECT_NOPOST
SIGNUP1	TOP_BANNER	MSG_POSTING_FORWARD_EDITOR
SIGNOFF2	BOTTOM_BANNER	MSG_POSTING_REJECT_NOTAUTH
You may also want to customize the following templates, unless you've disabled Auto-Delete or set Auto-Delete=Manual.		
AUTODEL	PROBE1	PROBE2

4.2 Two-Way Discussion Lists

Two-way discussion lists allow for interaction between group members. These open forums facilitate ongoing discussions among members. They are most commonly used for internal communication within an organization or group of people. For instance, many universities use secure "class lists" for interaction among students and teachers or to conduct remote teaching.

Unlike a one-way list, a discussion list not only allows, but also encourages interaction between members who are part of the group. Every member can write to the list, and in doing so, all members will receive a copy of the message. In this way, discussions can take place and views can be exchanged among a large number of people. Two-way discussion lists can be public or private depending on the way the list is configured. Public lists allow anyone to join and participate. Open public discussion lists tend to be large with many messages being posted (high volume). They can take up a lot of server space and they can be open to spamming, spoofing, and flaming if there are no security measures in place.

Closed or private two-way discussion groups can also be large and high volume, depending on the topic of the list, or they can be medium sized or very small. Subscriptions are monitored and are often subject to approval by a list administrator. Subscribers communicate amongst themselves on a specific topic. Incidents of

spamming and flaming are generally low because only subscribers can post to the list, and their subscription request has been reviewed. Set up a discussion list by using one of the following sets of keywords.

Table 4-4 Keywords for a Discussion Style List

Keyword and Suggest Value	Description
Send= Public	Anyone, even non-subscribers, can post to the list.
Send= Private	Only subscribed addresses may post to the list. Messages from an address that is not subscribed will be rejected, even if the person sending the message is subscribed under a different address.
Send= Public,Confirm,Non-Member	Anyone, even non-subscribers can post to the list, but non-subscribers are required to confirm their postings. This keeps most spammers off the list.

The following table contains some header keywords that you may want to consider setting in addition to those in the above table. Keywords appearing without a suggested value may have multiple settings and/or options to consider. For more information about assigning keyword values, see the [List Keyword Reference](#) document or the Keyword Wizard in the Web Interface.

Table 4-5 Other Keywords to Consider for a Discussion Style List

Keyword	Suggested Value
Attachments=	(Decide whether or not attachments should be allowed on posts to the list.)
Auto-Delete=	Yes,Full-Auto (Set Max, Delay, and Probe values as appropriate for you list.)
Change-Log=	Yes
Confidential=	(Decide whether or not the existence of the list should be made public.)
Daily-Threshold=	Default = 50 (Decide how many posts per day should be allowed on the list.)
Default-Options=	(Decide on the default subscription options.)
Digest=	(Decide whether or no to offer a digest form of the list.)
Moderator=	(List addresses that will moderate posts from subscribers set to REVIEW.)
Notebook=	(Decide whether or not to keep archives of posts, and decide who should have access to them.)
Reply-To=	(Decide where the replies should be directed to - the list, the sender, or both.)

Keyword	Suggested Value
Sizelim=	(Decide whether to place a limit on the size of post to the list.)
Subscription=	Open,Confirm
Validate=	Yes,Confirm (Yes,Confirm is the minimum recommended protection. For stronger protection see the List Keyword Reference document or the Keyword Wizard.

You may also want to customize the following templates. For more information on customizing templates, see Section 9 [Customizing Mail and Web Templates Using the Web Interface](#).

Table 4-6 Templates to Customize for a Discussion Style List

Welcome Message	SIGNOFF2	BOTTOM_BANNER
Farewell Message	ADD1	BOTTOM_BANNER_HTML
\$SIGNUP	DELETE1	TOP_BANNER
SIGNUP1	INFO	TOP_BANNER_HTML
You may also want to customize the following templates, unless you've disabled Auto-Delete or set Auto-Delete=Manual.		
AUTODEL1	PROBE1	PROBE2
If any subscribers are set to Review, then you may also want to customize the following templates.		
MSG_POSTING_FORWARD_EDITOR		
If using Confirm on the Send= keyword:		
MSG_POSTING_CONFIRM_SENDER		
If you are implementing restrictions on posting contents, then you may also want to customize one or more of the following templates.		
CONTENT_FILTER	MSG_POSTING_REJECT_CONTENTFILTER	MSG_POSTING_REJECT_BAD_ATTACHMENT
MSG_POSTING_REJECT_EXCEED_SIZELIM		

4.3 Moderated Lists

Similar to the two-way discussion list, a moderated list allows for the exchange of postings between subscribers and administrators, but an editor or moderator receives all incoming messages. The editor or moderator then decides to accept the message and post it to the list, or reject the message and not post it to the list. Lists can be set up with more than one moderator and they can take turns reviewing messages in a "round robin" fashion or all moderators can receive all messages.

Moderated lists can be of any size and subscriptions can be open or closed, depending on the type of list you want. Spamming and flaming are much less likely to happen on this type of list because all the postings are subject to approval by a person before they can be posted to the list.

Moderators receive and review postings sent to the list, and either approve them to be sent through the list or reject them to prevent them from being sent to the list. If there are multiple moderators, the list postings can be sent to each moderator in turn in a “round-robin” fashion or to all moderators. In the “round-robin” scenario, each message is sent to only one moderator for approval, sending one message to each moderator in the order listed in the list header before cycling back to the first moderator. The “round-robin” scenario allows a heavy load to be shared among several moderators. In the scenario where messages are all sent to all moderators, the first moderator to get to each message can approve or reject it. You would use this latter scenario if you want approvals to be made as soon as a moderator is available to review it.

The editors are those addresses that are allowed to send messages to the list without requiring moderation. The first address listed in the “Editor” keyword definition is known as the “primary” editor. If there is no “Moderator” defined, the primary editor is used as the sole moderator. The primary editor must always be an individual email address, but subsequent editors can be a list name enclosed in parentheses, meaning that all the subscribers on that list are allowed to post without being moderated. As a result, you can set up a separate list whose members are the editors of this one (and where subscriptions are closed). If you use the current list name, you have what is known as a “self-moderated” discussion list: subscribers can post without their messages being moderated, but posts from non-subscribers must be approved (see Section 4.2 [Two-Way Discussion Lists](#)). Set up a moderated list by using one of the following sets of keywords.

Table 4-7 Keywords for a Moderated Style List

Keyword and Suggest Value	Description
Send= Editor Editor= email-addresses	A moderated list, with all postings going to the primary editor. Instead of approving posts, the editor must resend the email message to the list, and may modify the message before resending it. Unless the editor is using an e-mail program that supports “Resent” headers (for example, Pine), the message will be distributed with the editor’s e-mail address in the From: line. Editors may post without requiring re-posting.
Send= Editor, Hold Editor= email-addresses Moderator= email-addresses	A moderated list, with postings going to one moderator at a time, in round-robin fashion, for approval. Editors may post without requiring approval.
Send= Editor, Hold Editor= email-addresses Moderator= ALL, email-addresses	A moderated list, with postings going to all moderators for approval. Any moderator may approve any post. Editors may post without requiring approval.
Send= Editor, Hold, Confirm Editor= email-addresses Moderator= ALL, email-addresses	A moderated list, with postings going to all moderators for approval. Any moderator may approve any post. Editors must confirm their own posts.

The following table contains some header keywords that you may want to consider setting in addition to those in the above table. Keywords appearing without a suggested value may have multiple settings and/or options to consider. For more information about assigning keyword values, see the [List Keyword Reference](#) document or the Keyword Wizard in the Web Interface.

Table 4-8 Other Keywords to Consider for a Moderated Style List

Keyword	Suggested Value
Auto-Delete=	Yes, Full-Auto (set Max, Delay, and Probe values as appropriate for you list)
Change-Log=	Yes
Confidential=	(Decide whether or not the existence of the list should be made public.)
Default-Options=	(Decide on the default subscription options.)
Digest=	No
Notebook=	(Decide whether or not to keep archives of posts, and decide who should have access to them.)
Reply-To=	Sender, Ignore
Subscription=	Open, Confirm
Validate	Yes, Confirm (Yes, Confirm is the minimum recommended protection. For stronger protection, see the List Keyword Reference document or the Keyword Wizard.0

You may also want to customize the following templates. For more information on customizing templates, see Section 9 [Customizing Mail and Web Templates Using the Web Interface](#).

Table 4-9 Templates to Customize for a Moderated Style List

Welcome Message	ADD1	BOTTOM_BANNER
Farewell Message	DELETE1	BOTTOM_BANNER_HTML
\$SIGNUP	INFO	TOP_BANNER
SIGNUP1	POST_EDITOR	TOP_BANNER_HTML
SIGNOFF2		
You may also want to customize the following templates, unless you've disabled Auto-Delete or set Auto-Delete=Manual.		
AUTODEL	PROBE1	PROBE2

Section 5 Introduction to the Web Interface

LISTSERV's Web Administration Interface makes LISTSERV administration significantly easier, allowing you to change many LISTSERV site configuration settings "on the fly" (although some changes may still require a restart of the server before they are recognized).



Note: Most sites will be able to upgrade to LISTSERV 16.0 without losing local web customizations, although this is not optimal and will not generally expose new features to your users. To assist you in customizing the new LISTSERV 16.0 Web Interface, L-Soft has produced a Customization Manual, which is available in PDF format at the following location: <http://www.lsoft.com/resources/manuals.asp>

The LISTSERV 16.0 Web Interface requires JavaScript to be enabled by default. However, those who prefer not to use JavaScript can set their navigation style to "Non-Script Navigation" in their personal preferences. For more information about preferences, see Section 5.6 [Setting Your Personal Preferences](#).

Virtually all list management operations can be accomplished via this Interface, which is tied into LISTSERV's own password manager for security.



Note: The Web Interface cannot be used to manage lists that are coded `Validate= Yes,Confirm,NoPW` or `Validate= All,Confirm,NoPW` because passwords are not accepted for validation in those cases.

5.1 The Default Home Page

The default home page for LISTSERV typically is reached by using the URL:

- On unix: `http://yourhost.domain/cgi-bin/wa`
- On VMS: `http://yourhost.domain/htbin/wa`
- On Windows: `http://yourhost.domain/scripts/wa.exe` or `http://yourhost.domain/cgi-bin/wa.exe`

Of course, this is not standardized; the location of the 'wa' script is determined by the value of `WWW_ARCHIVE_CGI` in LISTSERV's site configuration file. In any case, invoking 'wa' without any parameters returns the default home page.

5.2 Logging In

You can log into the Web Interface from any list's main web archive index page (assuming that this link has not been removed by the list owner; it exists in the `WWW_INDEX` mail template by default). The Interface may also be reached by a link from the default LISTSERV home page mentioned in the section above.

To access the Web Interface without a link, you point your web browser to the "wa" script. Typically the Interface is accessed as follows:

- On unix: `http://yourhost.domain/cgi-bin/wa`

- On VMS: `http://yourhost.domain/htbin/wa`
- On Windows: `http://yourhost.domain/scripts/wa.exe` or `http://yourhost.domain/cgi-bin/wa.exe`

and by default, users are directed to the main archives page for the server.

To get to the administrative pages, you will have to log in. If you already have a personal LISTSERV password, then you will simply log in with your existing userid/email address and password.

If you login with the "save my password in a cookie" method, LISTSERV will issue you a cookie that allows you to bypass this login screen (and incidentally to stay logged into the Interface for longer than 15 minutes without having to log in again when your session expires). This option is, however, only recommended for people who have physically secure machines (for instance, on your machine at home or in your office) or who are able to otherwise keep unauthorized users from logging in since LISTSERV cannot tell who is using the cookie. Specifically, if your browser does not support separate configurations or bookmark files for different users, you should not use the cookie method in a workplace environment.



Notes: There is a known bug in Netscape prior to version 4.0 that allows you to see the userid/email address and password typed into the text boxes if you back up to the login page using the "Back" arrow. The userid/email address you use here must be associated with the personal password you have from LISTSERV. If you have registered a password as `joe@unix.host.com` and try to log in here as `joe@host.com` with that password, LISTSERV will reject your login.

Figure 5-1 Login Screen

Login Required

Login Required

Please enter your email address and your LISTSERV password and click on the "Log In" button. If this is the first time you see this dialog, or if you have forgotten your password, you will need to [get a new LISTSERV password](#) first.

Email Address:

Password:



Tip: There is a new option for the **Login Cookie Expiration** setting on the General tab of your personal preferences so you can decide whether or not you want the login cookie to expire when you close the browser (end the session). This is highly desirable for those who wish to have increased security on shared computers. For details, see Section 5.6.1 [General Preferences](#).

5.3 Setting a LISTSERV Password

If you do not already have a personal LISTSERV password (set with the `PW ADD` command or via the Web Interface) or cannot remember your password, you need to define one now. If you choose to do this via the Web Interface, simply click the **Get a New LISTSERV Password** link. The Register LISTSERV Password screen opens.

Figure 5-2 Registering a LISTSERV Password



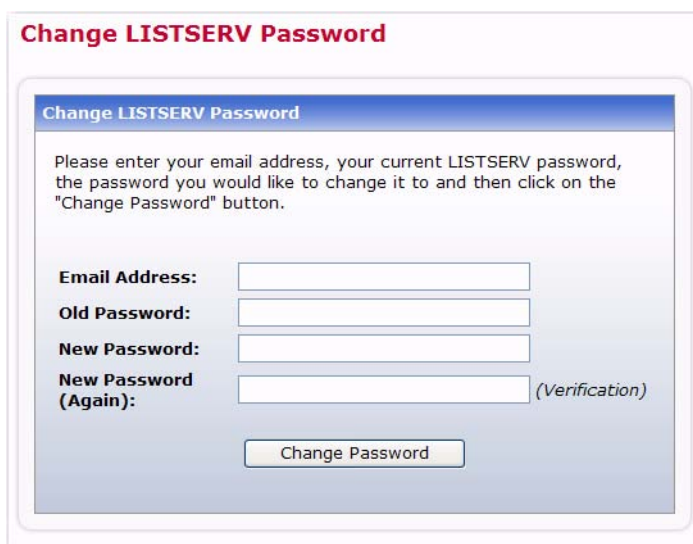
The screenshot shows a web form titled "Register LISTSERV Password" in a red header. Below the title is a blue bar with the text "Register LISTSERV Password". The main content area has a light blue background and contains the following text: "Please enter your email address and the desired password, then click on the 'Register Password' button. If you already had a LISTSERV password but cannot remember what it was, this procedure will automatically replace your existing password with the new one you will be entering below." Below this text are three input fields: "Email Address:", "Password:", and "Password (Again):". The "Password (Again):" field is followed by the text "(Verification)". At the bottom of the form is a button labeled "Register Password".

Enter your email and password. Confirm the password by entering it again, and then click the **[Register Password]** button. When your password registration is accepted, a confirmation email will be sent to you. You will have to activate your password by responding to the email (or clicking the link it contains).

5.4 Changing the LISTSERV Password

To change your LISTSERV password, click the **[Change Password]** button on the Login Screen. The Change LISTSERV Password screen opens.

Figure 5-3 Changing the LISTSERV Password



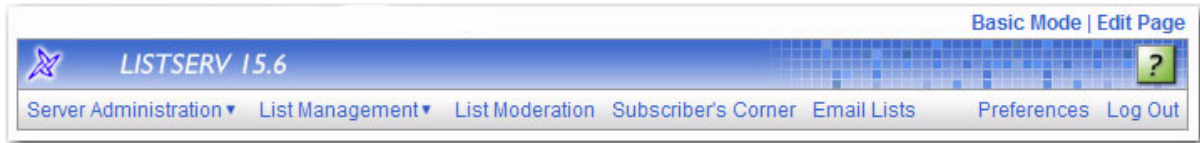
The screenshot shows a web form titled "Change LISTSERV Password" in a red header. Below the title is a blue bar with the text "Change LISTSERV Password". The main content area has a light blue background and contains the following text: "Please enter your email address, your current LISTSERV password, the password you would like to change it to and then click on the 'Change Password' button." Below this text are four input fields: "Email Address:", "Old Password:", "New Password:", and "New Password (Again):". The "New Password (Again):" field is followed by the text "(Verification)". At the bottom of the form is a button labeled "Change Password".

Enter your email address, your current LISTSERV password, the password you would like to change it to, and then click on the **[Change Password]** button.

5.5 Using the Toolbar

The Toolbar contains menus and icons that give you quick access to the different functions in the LISTSERV Web Interface.

Figure 5-4 The Toolbar



Note: If **Non-Script Navigation** is set in your preferences, then the toolbar will only consist of the main topics in LISTSERV. This means that when you click on a topic (i.e. **Server Administration**), the Web Interface refreshes and displays the page that was the first option on the Toolbar menu (used with **Pulldown Navigation**, which is described in Section 5.6 [Setting Your Personal Preferences](#)). For example, the first page for the Server Administration topic is the Server Administration Dashboard. At the top of the Dashboard, the remaining options for Server Administration will be displayed as links. Use these links to view these pages.

The **List Management** menu gives list owners access to the List Dashboard, list configuration, customization, and subscriber management. For more information on the features associated with this menu, see Section 6 [List and Subscriber Management Using the Web Interface](#).

The **List Moderation** option centralizes the moderation function, and it will show all messages needing moderation from the userid@host under which you are logged in. For more information on this feature, see Section 7 [List Moderation Using the Web Interface](#).

The **Email Lists** option takes you to the main list archives page. For more information on this feature, see Section 10 [Working with Archives in the Web Interface](#).

The **Preferences** option allows you to set personal preferences to control the appearance and default settings of the Web Interface. For more information on this feature, see Section 5.6 [Setting Your Personal Preferences](#).

The **Logout** option logs you out of the LISTSERV Web Interface.



Note: If LISTSERV is integrated with LISTSERV Maestro, then there will also be a **LISTSERV Maestro** menu on the Toolbar.



Help pages are accessed by clicking the **Help** icon at the right side of the Toolbar.

5.6 Setting Your Personal Preferences

LISTSERV allows you to set personal preferences to control the appearance and default settings of the Web Interface. Your personal preferences are saved with your email address and password, and they will be remembered each time you log in, even if you log in on a different computer.

To change your preferences, click on **Preferences** from the Toolbar, and then click on the General Preferences, Archive Preferences, or Report Preferences tab. Make your changes, and then click the **[Submit]** button.



Information for each option on the tabs can be found by clicking the **Help** icons associated with them.

5.6.1 General Preferences

The General Preference tab lets you set the appearance of the Web Interface; plus, if you are a server administrator, you can define your license and support expiration alerts.

Figure 5-5 The General Preferences Tab

Preference	Setting
Mode:	Expert Mode
Text Size:	Medium
Navigation Style:	Pulldown Navigation
Start Page:	LISTSERV Archives
License Expiration Alert:	45 Days
Support Expiration Alert:	45 Days
Owner Dashboard Changelogs:	After Prompt Only
Page Title Icons:	Shown
Login Cookie Expiration:	No Expiration
Cache Pages:	No

The following preferences can be set:

- **Mode** – Select how you want to view the Web Interface. The mode selected here will be displayed above the Toolbar.
 - If you choose **Tutorial**, then each web page is displayed with additional text that is meant to guide you through the options on the page. This is the best mode for beginners or for those occasional users.
 - If you choose **Basic**, then only the most common options are shown, but no tutorial text is displayed. This is the best mode for those users who are familiar with LISERV, but who are not experts and do not need to use the more advanced features. This is the default.

- If you choose **Expert**, then each web page is displayed with no tutorial text, similar to Basic mode, and certain pages may contain advanced options that are not available with the Basic mode. The Expert mode is available for those users who are very familiar with LISTSERV.
- **Text Size** – Select the size of the text for the Web Interface. Your options are **Small** (main body text is 11 pixels), **Medium** (main body text is 12 pixels), **Large** (main body text is 13 pixels) (default), or **Extra Large** (main body text is 16 pixels).
- **Navigation Style** – Select the toolbar style you prefer to work with.
 - If you choose **Pulldown Navigation**, then the toolbar will consist of javascript-powered drop-down menus. This is the default.
 - If you choose **Non-Script Navigation**, then the toolbar will only consist of the main topics in LISTSERV. Then, when you click on a topic (i.e. Server Administration), the Web Interface refreshes and displays the page that was the first option on the toolbar menu (used with Pulldown Navigation). For example, the first page for the Server Administration topic is the Server Administration Dashboard. At the top of the Dashboard, the remaining options for Server Administration will be displayed as links. Use these links to view these pages.



Note: If you choose **Non-Script Navigation**, all javascript throughout the Web Interface will be disabled.

- **Start Page** – Select the page that you want to act as your home page when you log in to the Web Interface. The default is **LISTSERV Archives**.
- **License Expiration Alert** – Select the number of days before your LISTSERV license expires that you want to be alerted on the Server Administration Dashboard. The default is **45 days**.
- **Support Expiration Alert** – Select the number of days before your support agreement expires that you want to be alerted on the Server Administration Dashboard. The default is **45 days**.
- **Owner Dashboard Changelogs** – Select whether you want any changelog data on the List Owner Dashboard to be loaded automatically every time you access the page or only after a prompt. Loading the changelog data automatically can make the page slower to load, especially if you have many lists or if they have large log files. The default is **After Prompt Only**.
- **Page Title Icons** – Select whether the page title icons should be shown or hidden. It might make sense to hide the icons on low screen resolutions since the conserved vertical space makes more of the page fit on the screen. The default is **Shown**.
- **Login Cookie Expiration** – Select how long you want your login cookie to be valid before it expires. The default is **No Expiration**, which means that the cookie will not expire until you explicitly log out. This means, for example, that you can close your browser and come back later and you will still be logged in. However, if you are working on a shared computer, for example, it might make sense to set the login cookie to expire for added security. After a time of inactivity, the cookie will no longer be valid and you will need to log in again to continue.

New

In addition, a new **Session** setting is available so you can decide whether or not you want the login cookie to expire when you close the browser (end the session). This is highly desirable for those who wish to have increased security on shared computers.



Note: Changing your cookie expiration setting does not change your existing login cookie. In order for the new login cookie setting to take effect, you need to log out and log in again.

New

- **Cache Pages** – Select whether or not you want recently visited pages of the Web Interface to be cached in your browser. The default is **No**, which provides extra security, especially if you are working on a shared computer.



Note: Without caching you will not be able to use the "Back" button in your browser to quickly return to past pages. Set this value to **Yes** if you prefer quick and easy back navigation and if this additional security is not critical to you.

5.6.2 Archive Preferences

The Archive Preference tab lets you define the default settings for the archive index and archive message viewer.

Figure 5-6 The Archive Preferences Tab

Preference	Setting
Hovering Descriptions:	Shown
Latest Messages on List Home Page:	Shown
Number of Messages on List Home Page:	3
Primary Sorting:	By Topic
Secondary Sorting:	Chronologically
Include Table of Contents:	Yes
Include Author's Name, Address:	Yes
Default View:	HTML Part
Archive Font:	Monospaced Font
Email Headers:	Normal
Archive Message Numbers:	No

These options are available for the archive index:



- **Hovering Descriptions** – Select whether you want hovering descriptions containing list descriptions and message abstracts to be displayed in the archives when you move your mouse over the list name or message. The default is **Shown**.



- **Latest Messages on List Home Page** – Select whether or not you want the latest messages to be shown on the list's home page. The default is **Shown**.



- **Number of Messages on List Home Page** – Select how many messages to show on the list's home page. The default is **3**. The maximum is 20.



Important: These three new preferences require javascript to function. Because of this, you must also have **Pulldown Navigation** selected as your **Navigation Style** on the General Preferences tab.

- **Primary Sorting** – Select how you want the archive index to be sorted. The options are **By Author**, **By Date**, or **By Topic**. The default is **By Topic**.
- **Secondary Sorting** – Select the order you want the archive index to be sorted after the messages have already been organized according to the **Primary Sorting** setting. The options are **Chronologically** or **Most Recent First**. The default is **Chronologically**.
- **Include Table of Contents** – Select whether or not you want to include a table of contents in your archive index. The default is **No**.
- **Include Author's Name, Address** – Select whether or not you want to include the author's name and email address in the archive index. The default is **Yes**.

These options are available for the archive message viewer:

- **Default View** – Select whether to view the **HTML** or **Text** part as the default for multi-part messages. The default is **HTML**.
- **Archive Font** – Select how you want the fonts to be displayed.
 - If you choose **Proportional Font**, then each character has a different width, depending on their size. For example, the letter w has a different width than the letter i.
 - If you choose **Monospace Font**, then each letter has the same width or "pitch". The most common pitch values are 10 or 12. For example, the letter w and the letter i would have the same pitch value; therefore, the word window would look like this using Courier font – window. Notice that the width allotted for the w and the i are the same. Other examples of monospace font are Prestige Elite and Monaco. Monospace font is also the name of a monospaced Unicode font.



Note: **Proportional Fonts** look more professional, but **Monospaced Fonts** are often better for data because the fixed width of each character makes column alignment easier. Printed documentation is usually done with proportional font.

- **Email Headers** – Select which headers to view in the archive message viewer.
 - If you select **Normal**, then only the Subject, Reply-To, From, Date, and Content-Type headers will be shown. This is the default.
 - If you select **Show All Headers**, then every header will be shown.

- **Archive Message Numbers** – Select whether you want individual messages or threads to be numbered in the archive index. The default is **No**.

5.6.3 Report Preferences

The Report Preference tab lets you define the default settings you want to use when running reports.

Figure 5-7 The Report Preferences Tab

The screenshot shows the 'Preferences' window with the 'Report Preferences' tab selected. The settings are as follows:

Preference	Setting
Changelog Period:	1 Day
Report Period:	6 Months
Lists and Subscribers Displayed:	50

The following preferences can be set:

- **Changelog Period** – Select the default date range of the Server Activity Report and the List Activity Report. The default is **1 day**.
- **Report Period** – Select the default date range of the Server Usage Report and the Anti-Virus, Anti-Spam Report. The default is **6 Months**.
- **List and Subscribers Displayed** – Enter the number of lists and subscribers you want to display per page when running any report that displays lists or subscribers. The default is **50**.



5.6.4 Newsletter Profiles

The Newsletter Profile tab in your Preferences allows you enter a profile for sending newsletters using the built-in HTML newsletter template system (see Section 11.1 [Creating Customized HTML Newsletters](#) for details on these templates).

Profile fields that have been defined will be automatically entered into HTML newsletter templates when the system is launched. Not all profile fields need to be defined. You will have the opportunity to change or fill in empty profile fields during the content development stage in the newsletter template system. Empty profile fields will be omitted from the final newsletter.

The following fields are available:

- **Newsletter Name** – Enter the name of your newsletter. This value will automatically replace the &*NEWSLETTER; placeholders in a newsletter template.

- **Physical Address** – Enter the physical address of your company or organization. This value will automatically replace the &*ADDRESS; placeholders in a newsletter template.
- **Phone Number** – Enter the phone number of your company or organization. This value will automatically replace the &*PHONE; placeholders in a newsletter template.
- **Contact Email** – Enter the contact email address of your company or organization. This value will automatically replace the &*CONTACT; placeholders in a newsletter template.

Figure 5-8 Setting Your Newsletter Profile Preferences

The screenshot shows the 'Preferences' window with the 'Newsletter Profile' tab selected. The window contains a table with two columns: 'Profile Field' and 'Profile Setting'. The fields and their settings are as follows:

Profile Field	Profile Setting
Newsletter Name:	<input type="text"/>
Physical Address:	<input type="text"/>
Phone Number:	<input type="text"/>
Contact Email:	<input type="text"/>
Website URL:	<input type="text"/>
Slogan:	<input type="text"/>
Logo:	<input type="text"/> <input type="button" value="Browse..."/> (Max 256 kB)
Footer:	<input type="text"/>
Legal:	<input type="text"/>
Delivery Test Addresses:	<input type="text"/>
HTML Editing:	Source Code Mode <input type="button" value="v"/>

There are 'Update' buttons at the top right and bottom right of the window.

- **Website URL** – Enter the Website URL of your company or organization. This value will automatically replace the &*WEBSITE; placeholders in a newsletter template. Since this field will be clickable in the newsletter, remember to enter the http:// or https:// part before the URL as applicable.
- **Slogan** – Enter a slogan for your company or organization. This value will automatically replace the &*SLOGAN; placeholders in a newsletter template.

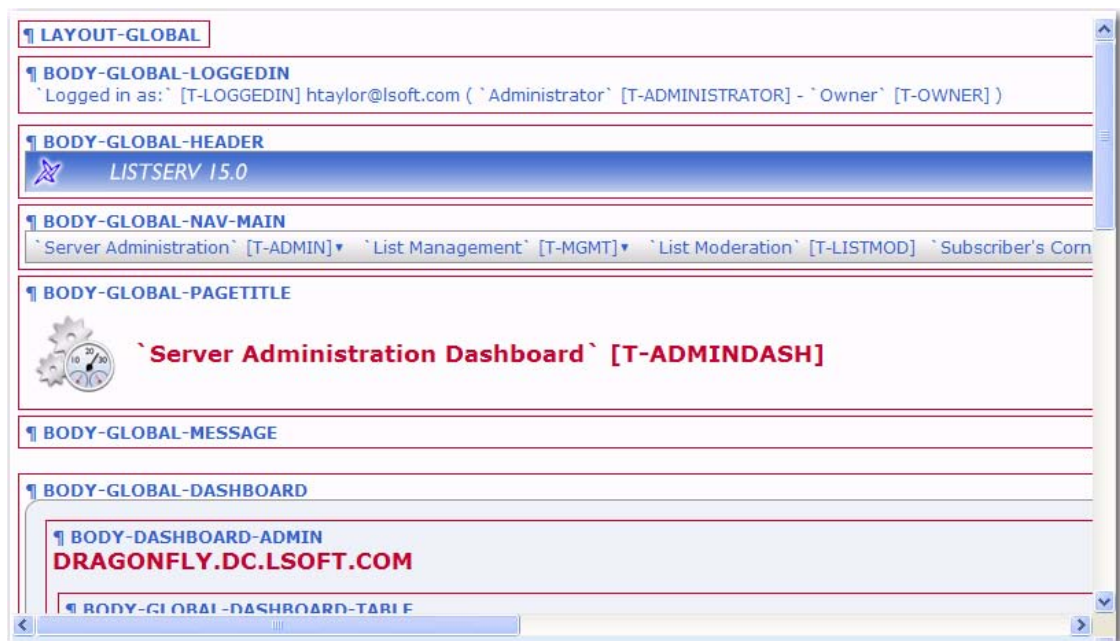
- **Logo** – Upload a logo for your company or organization. This image, which is saved on your LISTSERV server, will automatically replace the &*LOGO; placeholders in a newsletter template. In order for the image to be properly displayed in an HTML newsletter, the logo should be in .gif, .jpg, or .png format.
- **Footer** – Enter a short footer for your newsletter. This value will automatically replace the &*FOOTER; placeholders in a newsletter template.
- **Legal** – Enter any legal disclaimers, company registration numbers or similar information for your newsletter. This value will automatically replace the &*LEGAL; placeholders in a newsletter template.
- **Delivery Test Addresses** – Enter the list of email addresses to which newsletter delivery tests will be sent, one address per line.
- **HTML Editing** – Select the default HTML editing method. Source Code Mode will display the HTML code. HTML Editor Mode will use an HTML Editor.

5.7 Editing a Page

The Interactive Edit Mode is a tool to help server administrators and list owners customize the LISTSERV Web Interface. Only server administrators can make site-wide changes, while list owners can modify pages and templates for the lists that they own. When a logged in user has the privileges to edit a page, an **Edit Page** option appears in the top-right corner of the Web Interface. Click on this option to enable **Edit Mode**.

When enabled, the screen is split into two sections. The top section shows the current page divided into its component templates. By default, all LAYOUT and BODY template components are displayed as well as the names of all text variables. Each template is marked by a paragraph sign and followed by its name.

Figure 5-9 Edit Mode - Top Section



The bottom section contains the control panel.

Figure 5-10 Edit Mode - Bottom Section

To edit a template using the template editor, click on its name. The template editor will open in a new window. If you want to change a specific piece of text, make note of the name of the text variable, which is printed inside brackets following the text. Then, select it from the text variable drop-down menu at the bottom and click **[Load Text]**. A text area will appear containing the current text. Update the text, and then click **[Update Text]** to save the changes.

To expand the menu of variables to include all variables for a specific page, click the Show Variables drop-down menu, select the **Show All Text** option, and then click the **[Update]** button. (The **Show Displayed Text Only** option narrows the menu to include only the text that is currently displayed on screen.)



Important: Every template can be modified either for the entire site or for a specific list. When enabling Edit Mode, the templates that will be opened and updated are the ones for the list that is currently selected. For administrators, if no list is selected, then the site-wide templates will be modified. Before making any modifications, check carefully that the list for which you want to make changes is selected, especially if you own many lists or have server administrator privileges. Server administrators have the option to toggle between site-wide templates and templates for the selected list by clicking the **Modify for** drop-down menu and selecting **Load Text**.

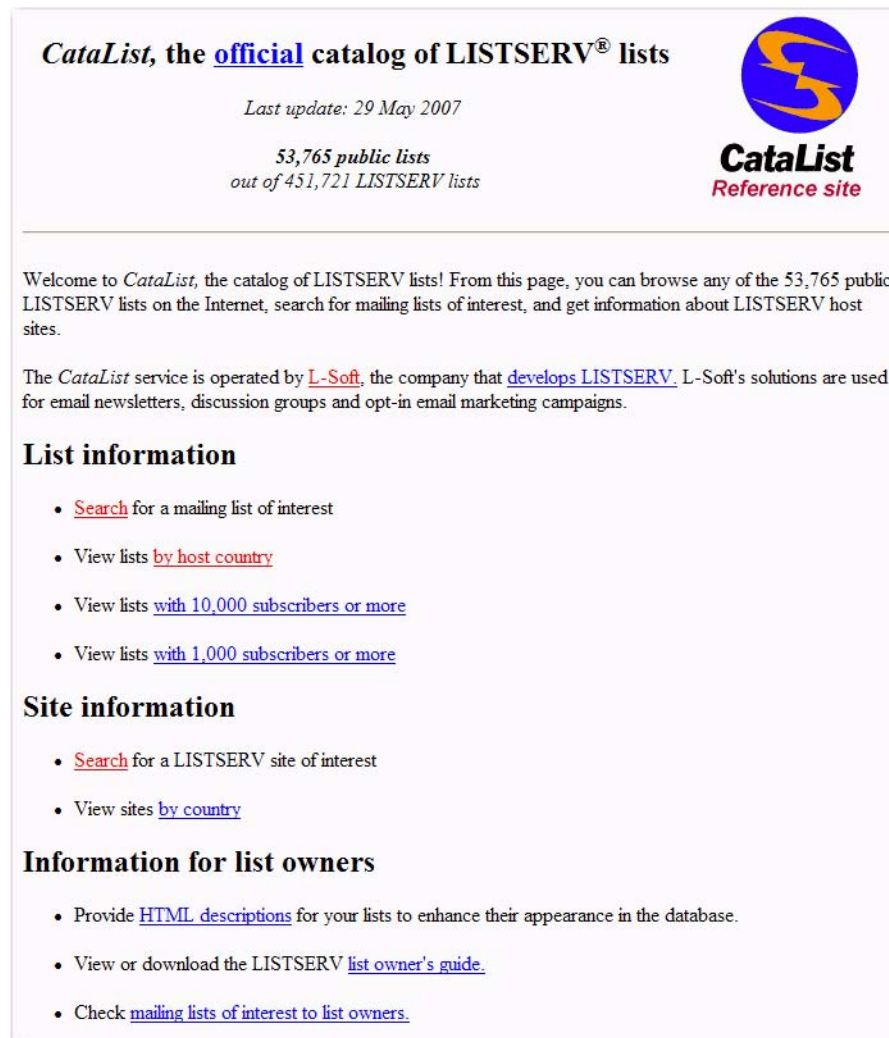
To exit the Edit Mode, simply click **Close Edit Mode**.

5.8 Accessing CataList



CataList, the catalog of LISTSERV lists, can be accessed at <http://www.lsoft.com/catalist.html> or, if you are logged into LISTSERV, by clicking on the **CataList** icon at the bottom of each screen. You can also access the CataList from the Server Archive Index (see Section 10.1 [The Server Archive Index](#) for more information.)

Figure 5-11 CataList



CataList lets you browse any of the 56,128 public LISTSERV lists on the Internet, search for mailing lists of interest, and get information about LISTSERV host sites.

While many of these public lists do screen new subscriptions, this is usually done to make sure that newcomers understand what the list is about and, perhaps more importantly, what it is not about. As long as you respect the list charter (set of rules for members), you can reasonably expect to be welcomed on any of the lists that you find in this database.

5.8.1 Searching for a List

You can search this database in a variety of ways to find a list of interest. You can browse through a set of lists by country, by host country, with 1,000 subscribers or more, or with 10,000 subscribers or more by clicking on the link that corresponds with each category. Or, you can perform a search to find a list with specific criteria.

To start your search, simply click on the **Search** link under **List Information** or **Site Information**.

Read the on screen instructions for search criteria, including the use of multiple search strings. Input your search criteria in the edit box. Check the appropriate checkboxes to

narrow or widen your search and then click the **[Start the search!]** button. The default home page will appear for the list.

If you have any questions about a specific list, direct them to the list owner. The list owner is always reachable by email at an address formatted like this:

listname-request@domainaddress

There are usually directions on how to subscribe to the list as well as a description of what the list is about on the home page.

Figure 5-12 Searching for a List

List search

This form allows you to search the database for lists whose name (XYZ-L), host name (LISTSERV.XYZ.EDU) and/or title ("Central America Discussion List") matches your search string. You can search for multiple topics by separating them with a comma. For instance, a search on "opera,classical" will return all the entries containing the word "opera" OR the word "classical". To search for a topic that actually contains a comma, type two commas in a row.

Look for:

In: ☒ List name ☐ Host name

☐ Lists with web archive

LISTSERV® site search

This form allows you to search the database for sites whose host name (LISTSERV.XYZ.EDU) and/or description ("Swedish University Network (SUNET), Stockholm, Sweden") matches your search string.

Look for:

In: ☒ Host name ☒ Site description

Section 6 List and Subscriber Management Using the Web Interface

The **List Management** menu on the Toolbar gives you a variety of features for working on and maintaining your lists and subscribers.

6.1 Using the List Management Dashboard

The List Management Dashboard is one of the screens that may appear when you log in. (This is determined by your preference settings.) If it does not appear, then you can open the dashboard by clicking **List Management**, and then **List Dashboard**.

The top part of the List Management Dashboard gives you the ability to control which lists are displayed:

- **Show Lists** – This case-insensitive parameter controls which lists are displayed. The default is to display all lists. For example, if you have a particular group of lists that you want to query that all contain the text "SALES" in the list name, for example, type "SALES" into the box and click **[Search]**.



Note: The search automatically expands "SALES" to "*SALES*". Wildcards are not otherwise supported.

- **Show Lists Owned By** – This parameter allows the site maintainer to narrow the selection of lists to those owned by a particular owner. Type an email address in userid@host format and click **[Search]**.



Note: Wildcards are allowed, so "joe*@" will return lists owned by joe@example.com, joely.brooks@example.org, and so forth. Addresses are not case sensitive, so "JOE*@" is equivalent to "joe*@".



Tip: The **Self Only** option is available for the server administrator, giving them the ability to modify the lists displayed so that only the listed they own are displayed.

The next part of the Dashboard is divided into two sections, providing information and reports about your technical support and lists.

Each section uses icons to indicate its status and available actions:



- **Green Shield with a Checkmark** – This icon means that you are current. Note that in the **Moderation** section this icon mean that there are no messages pending moderation.



- **Orange Diamond with an Exclamation Mark** – This icon means that something requires attention. Note that for the **Moderation** section, this icon means that there are messages pending moderation.



- **Life Buoy** – This icon is used if the Server Administrator has enabled technical support, making it easy and convenient to send requests to L-Soft support. Once you click on this icon, an email message opens. Enter any information describing your problem. Please be as detailed as possible.

The **Technical Support** section shows whether or not the Server Administrator has enabled technical support. If it is enabled, then the **Life Buoy** icon is shown, making it easy and convenient to send requests to L-Soft support. Once you click on this icon, an email message opens. Enter any information describing your problem. Please be as detailed as possible.

The **Moderation** section lists any messages that are awaiting moderation. The messages displayed here belong to a list for which you are listed as a moderator.



Note: This section is only displayed if you are a moderator on one or more lists. In addition, only two icons are used in this section. The green icon indicates that there are no messages pending moderation; the orange icon indicates that there are messages pending moderation.

Figure 6-1 List Management Dashboard - Top Half


List Management Dashboard

Select List:


Search Options

Dashboard for htaylor@lsoft.com


Technical Support

 Technical support has been enabled. If you encounter problems with your lists, you can contact the server administrator by clicking on the life buoy icon.

Once you click on this icon, an email message opens. Enter any information describing your problem. Please be as detailed as possible.



Moderation [Edit Table](#)

 There are currently no messages requiring moderation on any lists that you have selected for this report and for which you are listed as a moderator. Follow the "Edit Table" link to select the lists that you want included in this report.

The bottom part of the Dashboard contains a table that shows list configuration and list activity (changelog) data, which is a combination of the List Report and the List Activity reports. (Note that the list activity data is only visible if a list has changelogs enabled.) By default, the changelog data is not automatically calculated because of the time it takes to process the log files, especially if you have many lists or if they have large log files. To calculate the data, just click on one of the plus signs, [+].



Note: The calculated changelog data is now displayed using bar graphs for easier comparison.



Tip: If you would like the changelog numbers to be loaded automatically every time you access the page, you can change the **Owner Dashboard Changelogs** setting in the **Preferences** section.

To add or remove columns from the table, click the **Edit Table** option. This option is a great way to customize the information shown in the table, making sure only the information you want to see is visible. If you customize the table, then your changes will be saved in your preferences and will be automatically loaded every time you log in.



You can also sort the columns by simply clicking on the name of the column. Clicking on the column name once will sort from low to high or A to Z; clicking on the column name twice will reverse the sort order (i.e. from high to low or Z to A).

The **Lists Per Page** parameter controls how many lists will be displayed on a single page. The default is 10. If you want to break the list into 20 lists (for example), then simply enter "20" in the box and click **[Update]**.

The **Changelog Period** parameter lets you select the date range for the changelog columns in the report. The default is 1 day. If you want to change this period, choose a different option from the drop-down menu, such as 14 days, and then click **[Update]**.

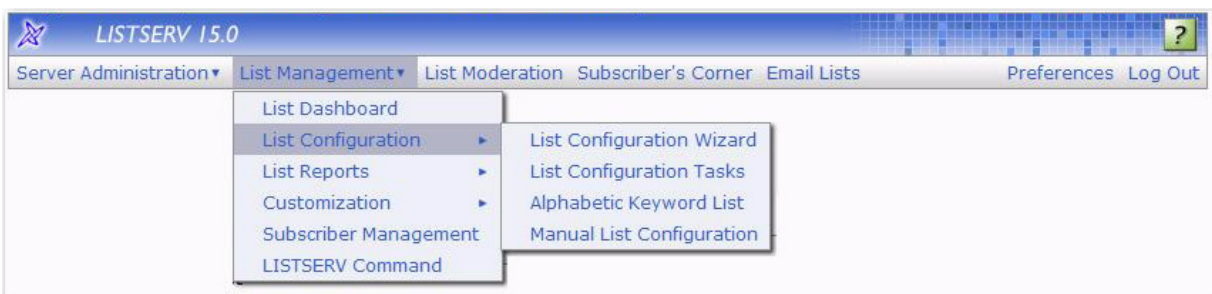
Figure 6-2 List Management Dashboard - Bottom Half

[Previous] [A~CC] [CO~D] [I~LIS] [LIT~P] [SE~STA] [STR~SV] [TES] [TES~UPD] [UPD] [Next]							Edit Table
List Name ▲	Subscribers	Send	Subscription	Log: Subscribe	Log: Signoff	Log: Post	
COMMUNITY [Configure]	1 [View]	Public [Edit]	Open [Edit]	42 <div><div></div></div>	41 <div><div></div></div>	40 <div><div></div></div>	
DISCUSSION [Configure]	2 [View]	Public [Edit]	Open [Edit]	93 <div><div></div></div>	87 <div><div></div></div>	32 <div><div></div></div>	
DRAGONFLY [Configure]	1 [View]	Private [Edit]	Closed [Edit]	0 <div><div></div></div>	1 <div><div></div></div>	7 <div><div></div></div>	
Lists per Page: <input type="text" value="3"/>				Changelog Period: <input type="text" value="30 Days"/>			<input type="button" value="Update"/>

6.2 List Configuration

Lists can be configured using a wizard, which guides you step-by-step through the configuration process, or manually.

Figure 6-3 The List Management Menu



6.2.1 List Configuration Using the Wizard

To open the List Configuration Wizard, click on the **List Management** menu, select **List Configuration**, and then select **List Configuration Wizard**.



Tip: To view help for any option in the List Configuration Wizard, simply click on the **Help** icon associated with it.

6.2.1.1 Descriptions

On the Descriptions Tab, enter the following information:

- **List Title** – A short description of the purpose of the list. The list title must fit on a single line and not exceed 40-50 characters. Choosing a descriptive title is important for public lists because it will be displayed when people search [CataList](#), the online catalog of LISTSERV lists. Potential subscribers should be able to determine the purpose of your list by reading the title. The list title is also used as the "name" part of the list's email address in the mail headers of messages distributed to the list. It is also used in the Web Interface and in some administrative messages.
- **List Description** – Enter a few lines of text containing a brief description of the purpose of the list. This description will be available to anyone who retrieves the public portions of the list header through a "REVIEW listname" command. The list description will also be displayed on the list's home page or archive index.

For the purpose of the Wizard, the list description is defined as any text following the last keyword definition or the last .HH OFF directive (whichever comes last) up to the start of the "HTML description" (if present). If you edit the list configuration header directly, you may enter comments throughout the header (comments are any text that is not a keyword definition or a directive). These will not affect the list description, unless they follow the last keyword definition or .HH OFF directive.
- **HTML Description** – (Optional) If provided, it will be used by [CataList](#), the online catalog of LISTSERV lists. If a text-based list description is not provided, then the HTML description will also be used on the list's home page or archive index. If a text-based list description and an HTML description are both provided, the text-based list description will be displayed by default. This default behavior can be changed by editing the list's OBJECT-A0-LISTDESC template.

To delete an existing list description, check **Delete List Description**, and click **[Submit]**.

To delete an existing HTML description, check **Delete HTML Description**, and click **[Submit]**.

Figure 6-4 The List Configuration Wizard - Descriptions Tab

List Configuration (LISTSERV-AT-WORK) [LISTSERV-AT-WORK Home](#)

Select List: LISTSERV-AT-WORK LISTSERV at Work Update

Descriptions Save

List settings available on this page: [List Title](#) | [List Description](#) | [HTML Description](#)

Keyword

List Title:
LISTSERV at Work

List Description:
LISTSERV at Work is L-Soft's quarterly newsletter, containing information, opinions, guidelines, insights and industry news. It is meant to keep customers and the L-Soft community informed and up-to-date with the company's latest developments.

[Delete List Description](#)

HTML Description:
<p>LISTSERV at Work is L-Soft's quarterly newsletter, containing information, opinions, guidelines, insights and industry news. It is meant to keep customers and the L-Soft community informed and up-to-date with the company's latest developments.</p>
<p align="center"></p>

[Delete HTML Description](#) [View](#)

[Show Only Currently Set Keywords](#) Save

Complete List Header [Edit Manually](#)

- * LISTSERV at Work
- * HH ON
- * Moderator= All,jkumpula@lsoft.se
- * Editor= jkumpula@lsoft.se
- * Notify= No
- * Auto-Delete= Yes,Full-Auto
- * Change-Log= Yes
- * Subscription= Closed
- * Owner= jkumpula@lsoft.se
- * Send= Editor,Hold,Confirm,Non-Member
- * Default-Options= NoPost
- * Ack= Yes
- * Review= Owner
- * Notebook= Yes,E:\LISTSERV\LISTS\LISTSERV-AT-WORK,Yearly,Private
- * Confidential= Service
- * HH OFF
- * LISTSERV at Work is L-Soft's quarterly newsletter, containing information, opinions, guidelines, insights and industry news. It is meant to keep customers and the L-Soft community informed and up-to-date with the company's latest developments.
- * <HTML>
- * <p>LISTSERV at Work is L-Soft's quarterly newsletter, containing information, opinions, guidelines, insights and industry news. It is meant to keep customers and the L-Soft community informed and up-to-date with the company's latest developments.</p>
- * <p align="center"></p>
- * </HTML>

To edit the list configuration manually, click **Edit Manually**. For more information, see Section 6.2.2 [Manual List Configuration](#).

6.2.1.2 Access Control

The Access Control Tab contains the keywords that define the addresses that have access to specific functions.

Figure 6-5 The List Configuration Wizard - Access Control Tab

List Configuration (LISTSERV-AT-WORK) [LISTSERV-AT-WORK Home](#)

Select List: LISTSERV-AT-WORK LISTSERV at Work Update

Access Control Settings Save

List settings available on this page: [Attachments](#) | [Review](#) | [Send](#)

Keyword	Setting
Attachments=	---
Allowed types:	<input type="text" value="image,message/rfc822,Store(1M)"/>
Filter:	<input type="checkbox"/>
Review=	Owner(s)
Specify:	<input type="text" value="OTHERLIST"/>

Summary of current "Send" setting:

- This is a moderated list.
- Posts from subscribers will be sent to moderation without requiring confirmation from the sender. Non-subscribers will be required to confirm their postings before the postings are sent to the moderators for approval. Editors will be required to confirm their own postings.
- Editor= jkumpula@isoft.se
- Moderator= All, jkumpula@isoft.se
- Postings submitted to moderation can be approved using the List Moderation Web interface or the email "OK" confirmation method.

Note: If you change the primary **Send=** setting, the header will be automatically updated and the page redisplayed with the valid options for the new setting you selected.

Send= Editor

Special:

Confirm: ☒

Confirm applies to: Non-Members and Editors

Hold: ☒

Semi-Moderated: ☐

NoMIME: ☐

[Show Only Currently Set Keywords](#) Save

The following keywords are available for definition, if applicable:

- **Attachments** – Use this keyword to control the posting of various types of MIME attachments (images, audio, etc.) to you list. It also includes the ability to control the posting of inline uuencoded files to your list on an “on/off” basis; “off” being the default if attachment control is enabled.
- For detailed instructions, see the online help by clicking the **Help** icon associated with **Attachments**.



Important: The ability of LISTSERV to filter or reject messages that contain MIME attachments is completely dependent on the ability of the poster's mail client to properly identify the MIME attachment when the mail is originally sent. Filtering/rejection is done based on the Content-Type headers found in the message – NOT by evaluation of the actual contents of the attachment. If, for instance, an executable binary (normally Content-Type: application/octet-stream) is sent by the client with a Content-Type of "text/plain", it will not be filtered or rejected by LISTSERV since text attachments are not covered by this keyword setting.

- **Files** – (NJE only; obsolete in other versions) Use this keyword to indicate whether NJE files can be sent to the list or not. The default value is **No**, which may prevent some non-RFC822 mailer users from posting to lists.



Note: This keyword has absolutely no effect under the non-NJE versions of LISTSERV. Specifically it will not prevent users from sending "attached" (MIME-encoded) files to lists. It is provided under all versions for backwards compatibility only (i.e., for lists being migrated from NJE servers). See the **Attachments** keyword for attachment blocking.

- **Filter** – Use this keyword to specify how the "filtered" address patterns are to be applied. Select one of the following:
 - **Also** – If you want the addresses or patterns entered below to be in addition to the "default" filters set up for the whole server (recommended).
 - **Only** – If you want to replace the default filters.
 - **Safe** – If you want to use the "safe" filter.



For detailed instructions, see the online help by clicking the **Help** icon associated with **Filter**.

- **Review** – Use this keyword to define the categories of users that are allowed to review the non-concealed Internet addresses and names of subscribers to the list. The default is **Private**.
- **Send** – Use this keyword to define the categories of users who can mail to the list. This can be used to place the list under the control of an editor. The default value is **Public**.



For detailed instructions, see the online help by clicking the **Help** icon associated with **Send**.

- **Stats** – (VM Only) Use this keyword to indicate whether or not statistics are to be maintained for the list. If yes, select which level of statistics is desired and who is able to retrieve the statistics reports. The default value is **Normal,Private**.



Note: This keyword is obsolete and has absolutely no effect on all ports of the software except for VM. On non-VM servers it is provided for backwards compatibility only (i.e., for lists being migrated from VM) in order that any existing **Stats** keyword setting in a migrated list header does not trigger a command parser error.

6.2.1.3 Distribution

The Distribution tab contains the keywords that determine the method of distribution for list message.

The following keywords are available for definition, if applicable:

- **Ack** – Use this keyword to define the default value of the "ACK/NOACK" distribution option for the corresponding list, that is the value assigned to new users when they subscribe to the list. This value can be altered by subscribers' "SET" command, but not by users who are not signed on to the list. This means that this option will always be in effect when distributing mail from people who are not on the distribution list.

If **Yes** is selected, then a short acknowledgment with statistical information on the mailing will be sent back to you.

If **No** is selected, then no acknowledgement will be sent. This is the default value.

- **Daily-Threshold** – Use this keyword to limit the number of postings that may be processed by the list in a calendar day (midnight to midnight, server time), and, with the addition of an optional second parameter, limits the number of postings that may be accepted from any individual user per calendar day (midnight to midnight in the server's local time zone).

The default is **50**. When the value of the first parameter is reached, the list is automatically placed on hold, and the list owner or LISTSERV maintainer must issue the FREE listname command.



Note: It may or may not be advisable to increase this parameter for higher-volume lists – individual list owners should study the issue carefully before increasing the daily threshold of their high-volume lists.

When the value of the optional second parameter is reached by an individual user, the user is told that their posting will not be processed and that they should resend it later if they still want it to be posted. The list itself is not held in this situation. The default is to have no such limit, in which case the second parameter is not defined. List owners and list editors are exempt from the individual daily limit. There is no command to reset the limit for an individual user, although the list owner may update the header to increase the value.

- **Digest** – Use this keyword to control the automatic digestification function allowing subscribers who do not have the time to read large numbers of messages as they arrive to subscribe to a digestified or indexed version of the list. The list owner decides whether digests are available or not, the frequency at which they are issued, and the day of week or time of day when the digest should be distributed.



For detailed instructions, see the online help by clicking the **Help** icon associated with **Digest**.



Note: UTF-8 encoded messages can now be displayed using the new plaintext digest format. For details, see Section 6.3.5.4 [Displaying UTF-8 Encoded Messages in the Body of a Digest](#).

- **Internet-Via** – (BITNET Only) Use this keyword to specify the host to use for routing Internet email. There is no default value. This parameter determines whether or not mail bound for Internet addresses is routed through a specific Internet gateway. In principle this keyword should never need to be set on non-BITNET hosts.
- **Mail-Via** – Use this keyword to specify whether or not to send messages out through the LISTSERV "DISTRIBUTE" network or directly through the local SMTP server. The default is **DISTRIBUTE**.



Warning: This keyword should generally only be used by the site administrator to troubleshoot delivery problems.

- **Newsgroups** – Use this keyword to define the RFC822 "Newsgroups:" header for a list. This field may be required by certain news gatewaying software and should only be defined if the list is gatewayed to Usenet and the gatewaying software requires it. The default is **None**.
- **NJE-Via** – (BITNET Only) Use this keyword to determine whether or not mail bound for NJE addresses is routed through a specific gateway. This keyword should never be set on non-BITNET hosts.
- **Prime** – Use this keyword to determine whether or not mail for the list is processed during "prime time", a value that is determined by the LISTSERV maintainer and is kept in the system configuration file. This keyword can be most useful in controlling the load on the machine running LISTSERV. The default is **Yes** (that is, list postings are distributed at the time they are received).



For detailed instructions, see the online help by clicking the **Help** icon associated with **Prime**.

- **Reply-To** – Use this keyword to indicate whether or not the "Reply-to:" tag supplied by the sender of the mail file, if present, is to be preserved or discarded and, if discarded or omitted, what should be placed in the new "Reply-to:" generated by the server. The default value is **List,Respect**.



Note: Some mailing systems are unable to process a "Reply-To:" field with multiple addresses correctly and may therefore disregard the **Reply-to= Both** option and treat it as **Reply-to= List**.



Warning: Setting this parameter guarantees only one thing – that LISTSERV will generate an appropriate RFC822 Reply-To: header in the mail it distributes to subscribers. There is unfortunately no guarantee that the mail transfer agent (MTA) or mail client on the receiving end will honor the Reply-To: header. This is because some mail clients, out-of-office robots, and Internet MTAs either simply do not recognize the existence of Reply-To: or do not implement it properly. Specifically, RFC2076 "Common Internet Message Headers" reports that the use of Reply-To: is "controversial", which is defined as: "The meaning and usage of this header is controversial, i.e. different implementors have chosen to implement the header in

different ways. Because of this, such headers should be handled with caution and understanding of the different possible interpretations." (RFC2076, page 4). While L-Soft recognizes that it is sometimes important to provide an explicit Reply-To: header to indicate a response path, L-Soft cannot be held responsible for problems arising from the inability of a remote server to properly process Reply-To: headers.



For detailed instructions, see the online help by clicking the **Help** icon associated with **Reply-To**.

- **Sender** – Use this keyword to define the value LISTSERV will place in the RFC822 "Sender:" field. The second parameter is optional, and is included to allow the specification of a second mailbox for use with IETF headers. The first value is used for non-IETF headers and is expected to contain the name and address of the list, or the keywords LIST or NONE. The second mailbox is used for IETF headers; if it is omitted, the generic "owner-listname" mailbox is substituted.



Important: Setting this value DOES NOT change the RFC822 "From:" header. Per standard, LISTSERV is not allowed to change the From: header, but must pass it through unchanged.

- **Sub-Lists** – Use this keyword to specify whether or not lists on this server are sublists of this list, creating a super-list.



For detailed instructions, see the online help by clicking the **Help** icon associated with **Sub-List** or see Section 6.2.4 [Normal List vs. Super-List](#).

- **Topics** – Use this keyword to define the topics for this list. List topics provide a way to run a mailing list (preferably moderated) where several sub-topics are being discussed in parallel, but some subscribers are only interested in a subset of the topics.



For detailed instructions, see the online help by clicking the **Help** icon associated with **Topics** or see Section 6.2.3 [Topics](#).

Figure 6-6 The List Configuration Wizard - Distribution Tab

List Configuration (LISTSERV-AT-WORK) LISTSERV-AT-WORK Home

Select List: LISTSERV-AT-WORK LISTSERV at Work Update

Distribution Settings Save

List settings available on this page: [Ack](#) | [Daily-Threshold](#) | [Digest](#) | [Reply-To](#) | [RSS-Abstract-Words](#)

Keyword	Setting
Ack=	Yes <input type="button" value="v"/>
Daily-Threshold=	<input type="text" value="100"/>
Per sender:	<input type="text" value="5"/>
Digest=	--- <input type="button" value="v"/>
Where:	--- <input type="button" value="v"/>
Specify:	<input type="text" value="E:\LISTSERV\LISTS\LISTSERV-AT-WORK"/>
Frequency:	--- <input type="button" value="v"/>
When:	<input type="text" value=""/>
Maximum size:	<input type="text" value="1000"/>
Bottom_Banner:	<input type="checkbox"/>
Reply-To=	--- <input type="button" value="v"/>
Specify address:	<input type="text" value="listserv-at-work-request@DRAGONFLY.DC.LSOFT.COM"/>
Existing Reply-To:	--- <input type="button" value="v"/>
RSS-Abstract-Words=	<input type="text" value="500,250"/>

[Show Only Currently Set Keywords](#) Save

6.2.1.4 Error Handling

The Error Handling tab contains the keywords that determine how LISTSERV deals with delivery errors. The following keywords are available for definition, if applicable:

- **Auto-Delete** – Use this keyword to define whether or not LISTSERV should automatically delete users whose account has expired or whose system has permanently disconnected.

For detailed instructions, see the online help by clicking the **Help** icon associated with **Auto-Delete**.

- **Errors-To** – Use this keyword to define the person or list that will receive rejection mail for the list. The default value is **Owners**.

- **Loopcheck** – Use this keyword to determine the type of loop checking performed by LISTSERV to avoid perpetuating mail loops. The default is **Full**. Loop checking is configured on a list by list basis only.



Warning: ALWAYS USE THIS KEYWORD WITH CAUTION! Misuse of this keyword can and will allow mailing loops onto your list!



For detailed instructions, see the online help by clicking the **Help** icon associated with **Loopcheck**.

- **Safe** – Use this keyword to control the email address LISTSERV places in the SMTP MAIL FROM: field, which is where standards-compliant mailers will return delivery errors. When set to **No**, then these errors are sent to the list address as before, hopefully to be intercepted by the loop detector and passed on to the list owner. When set to **Yes**, then the error address is set to 'owner-listname', and delivery errors sent to that address are passed on to the list owner without the risk of creating a mailing loop. The default is **Yes**.



Important: Setting this keyword to **Yes** does not guarantee that all errors will go to the 'owner-listname' mailbox. Unfortunately, there are many non-compliant mailers which will continue to send the error back to the list (usually because it is listed in the 'Reply-To:' or 'Sender:' field). Setting this keyword to **Yes** does significantly decrease the potential for mailing loops, but not enough to actually code "Loopcheck= No", unless you are sure that all your subscribers have compliant mailers.

Figure 6-7 The List Configuration Wizard - Error Handling Tab

List Configuration (LISTSERV-AT-WORK)

Select List: LISTSERV-AT-WORK LISTSERV at Work Update

Error Handling Settings Save

List settings available on this page: [Auto-Delete](#) | [Errors-To](#)

Keyword	Setting
Auto-Delete=	Yes
Type:	Full-Auto
Delay:	10 days
Max:	100 bounces
Probe:	10
Errors-To=	---
Specify:	barbara@example.com

[Show Only Currently Set Keywords](#) Save

6.2.1.5 List Maintenance

The List Maintenance tab contains the keywords that control list maintenance and moderation. The following keywords are available for definition, if applicable:

- **Editor** – Use this keyword to define the list editor(s). Enter the email addresses that are allowed to post to the list without moderation. The first address is the "primary" editor and the default moderator and must be a single email address. To give the "editor privilege" to all subscribers of a list (on this server) enter the listname in parentheses, for example: "(MYLIST)".

When used in conjunction with the "Send=Editor" option, it causes all mail sent to the list to be automatically forwarded to the first person listed in the **Editor** keyword, who will then determine whether to allow it to be distributed to the list. The editors are the only persons (with the list owners) who are allowed to mail directly to the list. ANY editor can send mail to the list while only the FIRST one will receive copies of mail sent to the list (but see also **Moderator**).



Important: The first editor **MUST** be an email address that goes to a person, not a file server, list server, mailer, or other automated email address. Specifying a program's mailbox as the primary editor could result in a mailing loop for which L-Soft international, Inc., could not be held responsible.



For detailed instructions, see the online help by clicking the **Help** icon associated with **Editor**.

- **Editor-Header** – Use this keyword to define the list editor(s). When used in conjunction with the "Send=Editor" option, it causes all mail sent to the list to be automatically forwarded to the first person listed in the **Editor** keyword, who will then determine whether to allow it to be distributed to the list. The editors are the only persons (with the list owners) who are allowed to mail directly to the list. ANY editor can send mail to the list while only the FIRST one will receive copies of mail sent to the list (but see also **Moderator**).



For detailed instructions, see the online help by clicking the **Help** icon associated with **Editor-Header**.

- **List-Address** – Use this keyword to determine how LISTSERV announces its list address in the header of messages delivered to the list: NJE vs. Internet address, short vs. long list name, etc. The default options (when neither **List-Address** or the site configuration `LIST_ADDRESS` parameter are defined) are long list name and Internet address. A corresponding `LIST_ADDRESS` configuration option may be added to the LISTSERV site configuration file.



Important: The only effect of the **List-Address** keyword is to change the way the list identifies itself in list postings, command replies, etc. It does not instruct the mail system to create forwarding entries to support the new name, nor does it establish the specified name as an alias for the list (use **List-ID** for this purpose). In general, list owners should not use this keyword without first consulting with the LISTSERV maintainer.

- **List-ID** – (VM Only) Use this keyword to define an alternate name (alias) for the list.



Note: List owners should not set a value for **List-ID** without first consulting with the LISTSERV maintainer, since it will be necessary to add appropriate system mailer aliases before the name specified in **List-ID** will work.

- **Moderator** – Use this keyword to define which editors of a moderated list receive postings for forwarding to the list. The default is the first editor as defined by the **Editor** keyword. If multiple moderators are defined, the load is spread across them.



For detailed instructions, see the online help by clicking the **Help** icon associated with **Moderator**.

- **New-List** – Use this keyword to define the name of the new list that this list has been migrated to.



Warning: This keyword can only be used on a list that has already been migrated and all other keywords except Owner removed from the list configuration. To prevent accidental deletion of all keywords, this Wizard will not remove the other keywords for you. The easiest way to do this is to go to the List Management page and use the Configuration button to edit the "list header" and remove all the keywords except Owner.

- **Notebook** – Use this keyword to indicate whether or not an automatic log of every piece of mail sent to the list is to be kept, and to define at which interval of time its file name must be changed and who is allowed to retrieve it from the server. The default values are **Notebook= No,A,Single,Private**. Only a postmaster may set this keyword.



For detailed instructions, see the online help by clicking the **Help** icon associated with **Notebook**.

- **Notebook-Header** – Use this keyword to determine whether or not individual messages in notebook archives are stored with full Internet header information or with "short" headers. The default is **Short**.
- **Notify** – Use this keyword to define whether the list owner (or the person indicated by "Notify=mon-address") is to receive notification of new subscriptions and deletions, etc. The default is **Yes**, meaning that non-quiet list owners will be notified.
- **Owner** – Use this keyword to define the person or list of persons who "own" the list. They are responsible for controlling access to the list and defining the list control keywords that are best suited to the purpose of the list.

This keyword is required in every list and there is no default value. Any combination of explicit network addresses and complex access-levels is acceptable, for example: Owner= BIG@BLUE,(STAFF-L),Owner(MAIN-L)




Note: The use of the "Quiet:" parameter causes all subsequently defined list owners to be excluded from receiving any delivery error messages or other administrative mail from LISTSERV.

- **Peers** – Use this keyword to define the list of all the servers in the world that are peer-linked to the list, either directly or via one or more other peer servers. This information is used by the various list management commands to determine the


"nearest" peer list to a given user. For example, when a SUBSCRIBE command is received from a user and it is determined that there is a nearer peer list for that email address, the subscription request is automatically forwarded to the appropriate LISTSERV.

- **Renewal** – Use this keyword to control whether or not subscribers are required to renew their subscriptions on a regular basis, and what the subscription period is. Multiple renewal times can be set. Each renewal time can be specified as an interval or a set date.

 For detailed instructions, see the online help by clicking the **Help** icon associated with **Renewal**.

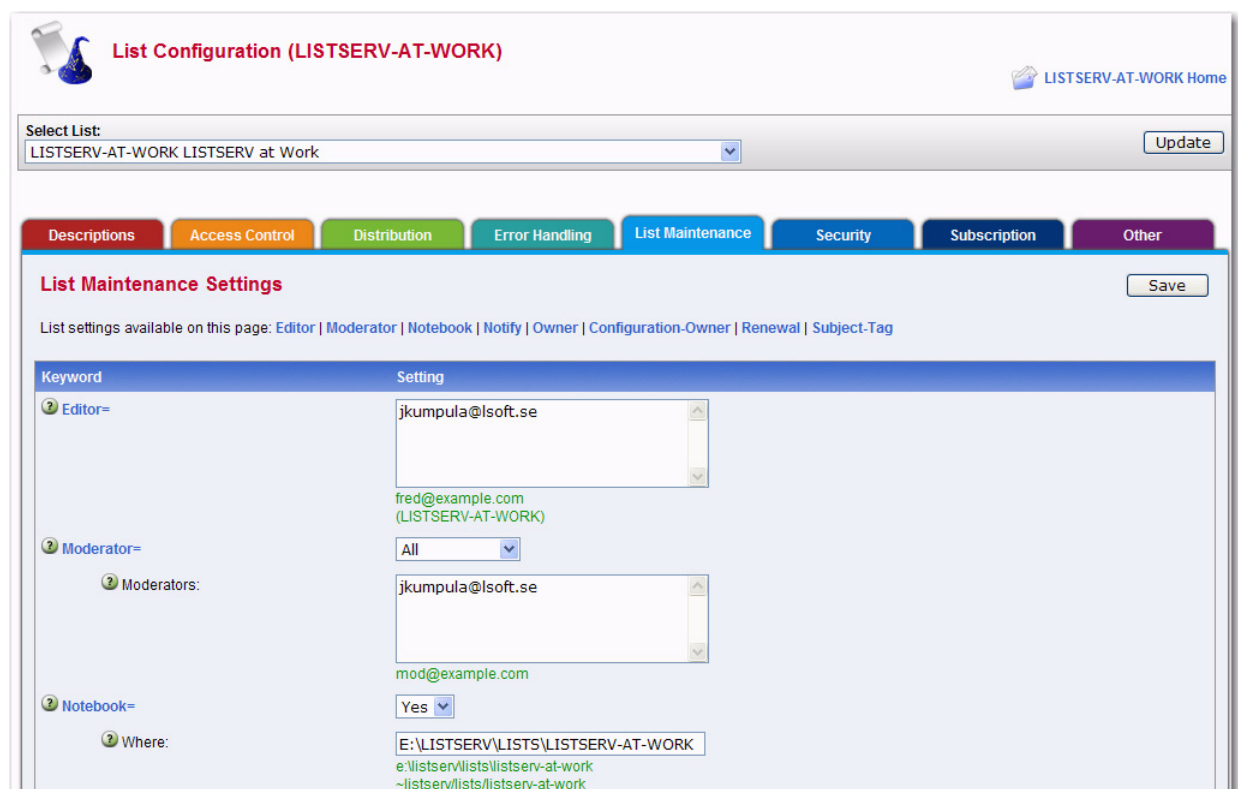
- **Sizelim** – Use this keyword to indicate the maximum size for acceptable messages to be posted to the list (larger messages will be rejected). A plain integer represents the size as a number of lines; add "K" or "M" after the integer to represent kilobytes or megabytes.






- **Subject-Tag** – Use this keyword to define the text to use in lieu of the listname in the subject line for subscriptions set to "SUBJECTHDR".

 For detailed instructions, see the online help by clicking the **Help** icon associated with **Subject-Tag**.

- **X-Tags** – Use this keyword to indicate whether or not "X-To:" and "X-cc:" tags are to be included in the output mail files to list recipients of the original mail file (other than the list userid), and how they should appear in the RFC822 header.

Figure 6-8 The List Configuration Wizard - List Maintenance Tab



Keyword	Setting
 Editor=	jkumpula@lsoft.se fred@example.com (LISTSERV-AT-WORK)
 Moderator=	All
 Moderators:	jkumpula@lsoft.se mod@example.com
 Notebook=	Yes
 Where:	E:\LISTSERV\LISTS\LISTSERV-AT-WORK e:\listserv\lists\listserv-at-work ~listserv\lists\listserv-at-work

6.2.1.6 Security

The Security tab contains the keywords that control list security. The following keywords are available for definition, if applicable:

- **Change-Log** – Use this keyword to indicate whether or not to maintain a subscription changelog. This is required if you want to be able to produce historical and statistical reports about subscription activities.



For detailed instructions, see the online help by clicking the **Help** icon associated with **Change-Log**.

- **Confidential** – Use this keyword to indicate whether or not the list should be hidden from users. A confidential list will not appear on the "Lists" command output. "**Confidential= No**" is the default value and indicates that the list is not confidential. "**Confidential=Service**" indicates that the list is to be hidden from users who are not in the list's service area (see "**Service=**" keyword) but not from other users. "**Confidential= Yes**" means that the list is unconditionally confidential.



Note: The local list of (public) lists can be retrieved only by those users who are considered local, per the setting of the server-wide LOCAL= variable in LISTSERV's site configuration file. All other users will be told that none of the lists on the server are visible via the LISTS command, and will be referred to the use of the LISTS GLOBAL search-text command or to the CataList. This is regardless of the setting of Confidential=.

- **Exit** – Use this keyword to define the name of the exit program to be used with this list. Only a Site Administrator may set this keyword.



For detailed instructions, see the online help by clicking the **Help** icon associated with **Exit**.


- **Local** – Use this keyword to define the nodes that are to be considered as 'local nodes' for service area checking. The LISTSERV machine is automatically considered as a 'local node' and does not have to appear in the list. Subscribers from any of the local nodes will receive separate pieces of mail with a single recipient in the "To:" field – in other words, they will never receive a grouped piece of mail as non-local recipients would if there are more than one recipient in their node. 'Node' is a generic term that means "anything after the '@' sign in the network address". For instance, "SEARN" and "SEARN.SUNET.SE" are both valid node names. By default, this keyword takes its value from the LOCAL variable in LISTSERV's site configuration file.
- **PW** – (Peered Lists Only) Use this keyword to define the list password. When sending the list back to the server, the password is prefixed to the list file for validation (see the **Validate** keyword for more specifics). The **PW** parameter is "invisible" once it is defined; that is, for security reasons, it does not appear either when the list is reviewed or when it is retrieved with a GET command by the list owner.



Note: LISTSERV generates a 16-character random password for lists at list creation time if this keyword is not explicitly defined, making such lists more secure from random

hackers. List owners are now encouraged to use personal passwords (defined with the `PW ADD` command, q.q.v.) in preference to list passwords for this reason. The one exception to this keyword's obsolescence is when you are creating peer lists. Peers must have the same **PW** keyword so you cannot use the LISTSERV-generated random password when creating peers.

- **Service** – Use this keyword to specify a limited service area for this list (leave blank to allow subscriptions from anywhere).

 For detailed instructions, see the online help by clicking the **Help** icon associated with **Service**.

- **Validate** – Use this keyword to determine what level of validation (if any) is performed for various LISTSERV commands that apply to individual lists. There are six different settings ranging from very basic to very strict. The two most common settings are **Yes** and **No**.


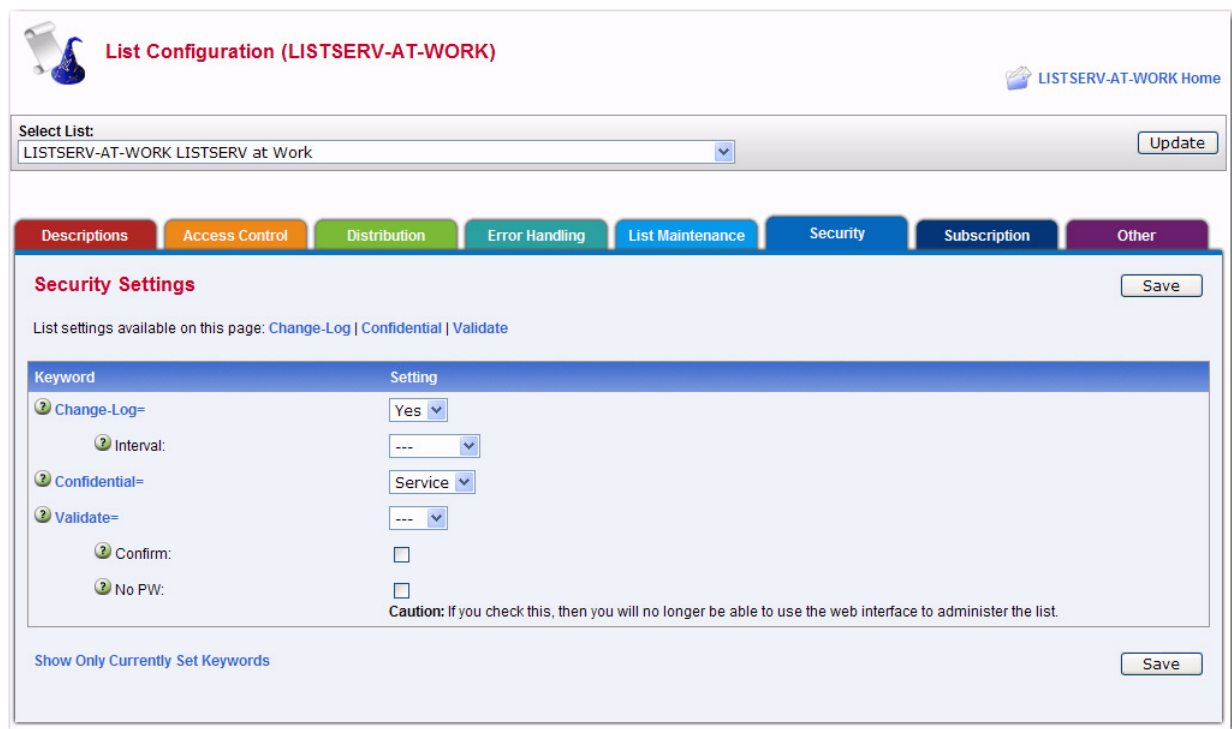
 For detailed instructions, see the online help by clicking the **Help** icon associated with **Validate**.

Figure 6-9 The List Configuration Wizard - Security Tab









List Configuration (LISTSERV-AT-WORK)

Select List: LISTSERV-AT-WORK LISTSERV at Work Update

Descriptions Access Control Distribution Error Handling List Maintenance **Security** Subscription Other

Security Settings Save

List settings available on this page: [Change-Log](#) | [Confidential](#) | [Validate](#)

Keyword	Setting
 Change-Log=	Yes ▼
 Interval:	--- ▼
 Confidential=	Service ▼
 Validate=	--- ▼
 Confirm:	<input type="checkbox"/>
 No PW:	<input type="checkbox"/>

Caution: If you check this, then you will no longer be able to use the web interface to administer the list.

[Show Only Currently Set Keywords](#) Save

6.2.1.7 Subscription

The Subscription tab contains the keywords that control subscription to the list. The following keywords are available for definition, if applicable:

- **Confirm-Delay** – Use this keyword to define the number of hours to hold subscription requests awaiting confirmation.





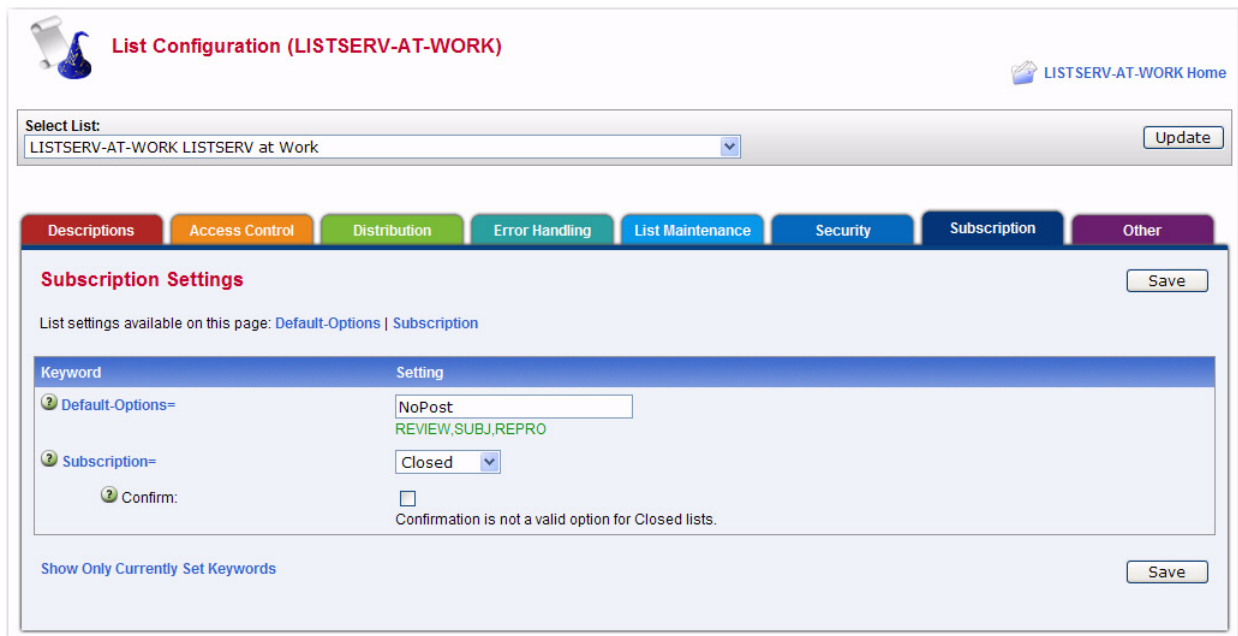
-  For detailed instructions, see the online help by clicking the **Help** icon associated with **Confirm-Delay**.
- **Default-Options** – Use this keyword to define the subscriber options that will be the default for all new subscribers. Separate options with commas.
-  For detailed instructions, see the online help by clicking the **Help** icon associated with **Default-Options**.
- **Default-Topics** – Use this keyword to specify which of the currently defined topics should be assigned to new subscribers by default.
-  For detailed instructions, see the online help by clicking the **Help** icon associated with **Default-Topics**.
- **Subscription** – Use this keyword to define whether or not new users are allowed to subscribe to the list, and if not, whether their subscription requests are to be forwarded to the list owner or not.
-  For detailed instructions, see the online help by clicking the **Help** icon associated with **Subscription**.

Figure 6-10 The List Configuration Wizard - Subscription Tab






List Configuration (LISTSERV-AT-WORK) LISTSERV-AT-WORK Home

Select List: LISTSERV-AT-WORK LISTSERV at Work Update

Descriptions
Access Control
Distribution
Error Handling
List Maintenance
Security
Subscription
Other

Subscription Settings Save

List settings available on this page: [Default-Options](#) | [Subscription](#)


Keyword	Setting
 Default-Options=	NoPost REVIEW,SUBJ,REPRO
 Subscription=	Closed ▼
 Confirm:	<input type="checkbox"/> Confirmation is not a valid option for Closed lists.

[Show Only Currently Set Keywords](#) Save

6.2.1.8 Other

The Other tab contains miscellaneous keywords that were not covered on the previous tabs. The following keywords are available for definition, if applicable:

- **Categories** – Use this keyword to define the categories to use for searches in [CataList](#).

 For detailed instructions, see the online help by clicking the **Help** icon associated with **Categories**.

- **DBMS** – (Non-VM Only) Use this keyword to specify whether or not this list is stored in a DBMS. This should only be set by the Site Administrator.



Warning: Do not set this keyword unless the LISTSERV site configuration is set up to work with a DBMS.



For detailed instructions, see the online help by clicking the **Help** icon associated with **DBMS**.

- **Indent** – Use this keyword to determine the minimum number of columns allowed for list addresses in response to the `REVIEW` command. The default is **40**.
- **Language** – Use this keyword to define the language in which information mail and messages are to be sent to subscribers of the list. The postmaster must have provided the required data file (called *idiom*.MAILTPL, where *idiom* is the name of the language specified by this keyword) to the server. The default is **English**, which uses `DEFAULT.MAILTPL`.



For detailed instructions, see the online help by clicking the **Help** icon associated with **Language**.

- **Limits** – (ISP Add-On Only) Use this keyword to define specific limits for a list. Currently only the number of subscribers can be limited.



Important: This keyword is available only with the ISP add-on and may only be added or changed by the LISTSERV Maintainer.

- **Long-Lines** – Use this keyword to enable or disable "long-lines" support.



Note: This keyword was added to maintain compatibility with LISTEARN and will be removed in a future version of LISTSERV. The default is **Long-Lines= Yes**. It is unlikely that this keyword will need to be set for any list.

- **Mail-Merge** – Use this keyword to specify whether or not mail-merge is allowed on this list. Mail-merge must be set by a postmaster as it requires certain site configuration settings. For more information about mail-merge, see Section 6.2.5 [Mail-Merge](#).




For detailed instructions, see the online help by clicking the **Help** icon associated with **Mail-Merge**.

- **Misc-Options** – This keyword is a catch-all for certain behavior-modifying options that are not otherwise covered by other, more specific keyword settings. Currently the only options available are as follows:
 - **NO_SPAM_CHECK** – Use this option to disable spam scans for a particular list and its associated xxx-request address. (This is only useful if the LISTSERV maintainer has enabled spam-scanning via the `SPAM_EXIT` feature.)
 - **SUPPRESS_APPROVED_BY** – Use this option to suppress RFC822 "Approved-By:" headers that would normally be generated by LISTSERV in messages posted through moderated lists.
 - **IGNORE_EMAIL_CASE** or **RESPECT_EMAIL_CASE** – These options are mutually-exclusive; only one can be defined at a time per list.

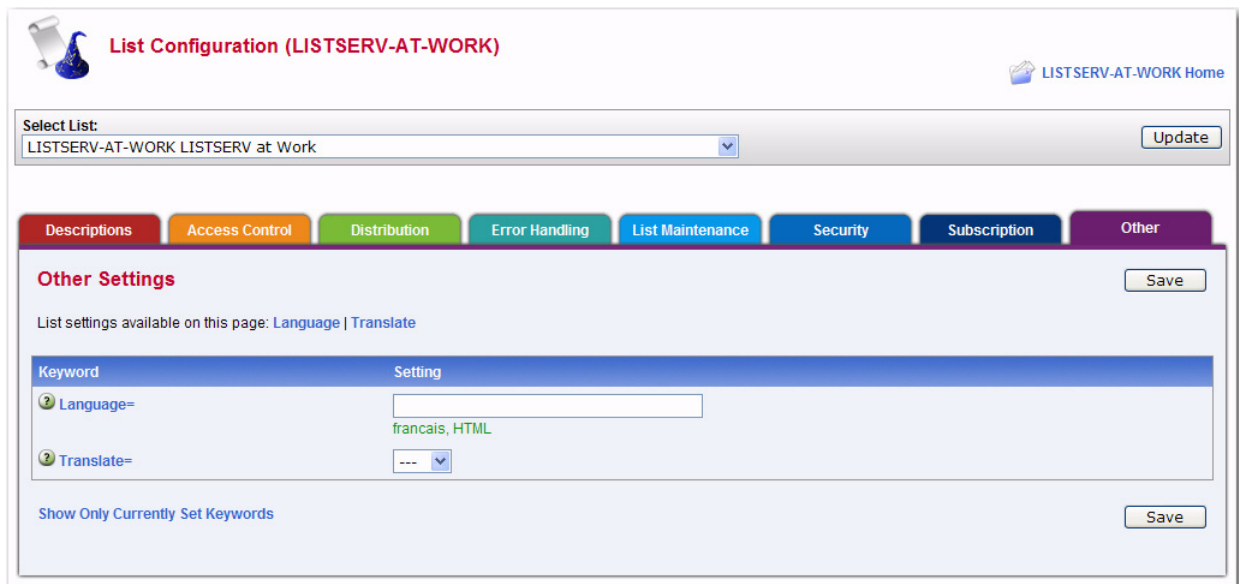
When set in a list header, **Misc-Options= IGNORE_EMAIL_CASE** causes the **ADD** command to ignore the case of the "local part" of list subscriber entries (that is, the part of the address that is to the left of the "@" sign). Although most modern mail clients are configured to ignore the case of the local-part, this behavior technically violates RFC821 which states that local-parts are considered case-sensitive.

When a list is set to **Misc-Options= RESPECT_EMAIL_CASE**, this tells **LISTSERV** to operate per RFC821 and treat address fields with differently-cased local parts as different addresses. The option is provided as an override to the site-level **IGNORE_EMAIL_CASE** configuration variable and does not need to be set to preserve the default unless the site setting has been changed to make **IGNORE_EMAIL_CASE** the default.

 For detailed instructions, see the online help by clicking the **Help** icon associated with **Misc-Options**.

- **Translate** – Use this keyword to determine whether or not **LISTSERV** keeps or removes control characters from files that it distributes. **Yes** removes control characters; **No** keeps them. The default setting is **Yes**.

Figure 6-11 The List Configuration Wizard - Other Tab





List Configuration (LISTSERV-AT-WORK)

Select List: LISTSERV-AT-WORK LISTSERV at Work Update

Descriptions Access Control Distribution Error Handling List Maintenance Security Subscription Other

Other Settings Save

List settings available on this page: [Language](#) | [Translate](#)

Keyword	Setting
 Language=	<input type="text" value="français, HTML"/>
 Translate=	<input type="text" value="---"/>

Show Only Currently Set Keywords Save

6.2.2 Manual List Configuration

To configure a list manually, click on the **List Management** menu, select **List Configuration**, and then select **Manual List Configuration**.

Figure 6-12 Manual List Configuration

List Configuration (TEST-HAT) [TEST-HAT Home](#)

Select List:
TEST-HAT Test List

TEST-HAT

List Header [Edit List Configuration with Wizard](#)

Test List

.HH ON
Subscription= Closed
Owner= htaylor@lsoft.com
Send= Editor, Hold, Confirm
Editor= htaylor@lsoft.com
Moderator= htaylor@lsoft.com
Ack= Yes
Review= Owners
Notebook= No
Change-log= No
Notify= No
Attachments= Yes
Confidential= Yes
.HH OFF

For those of you who want to configure the list manually, you can edit the list header in its "raw" state. This is only recommended for people who are very comfortable with the format of the LISTSERV list header and know the keywords and their parameters very well.

The list header appears in a multi-line text box that can be scrolled up and down. You simply type in the changes or added lines just as if you were using a regular text editor. When you are finished, click the **[Save]** button to submit the changes. If you want to start over, you can click the **[Revert]** button to reload the header information from the server.

When you submit your changes with the **[Save]** button, you will get the same kind of feedback from LISTSERV as you would if you sent a PUT operation by mail. The next screen will either say that the header of the list has been successfully updated, or it will indicate that it has found errors and that the header has not been stored. The feedback page also has a text box containing the header information you've just stored (or tried to store) so if you need to make further corrections to the header, you don't have to back up and start over.



The list header keywords and their parameters are documented in the [List Keyword Reference](#) document, in the online help, or (when using the configuration wizard) by clicking the **Help** icon for each keyword.

6.2.2.1 Hiding Parts of the Header

Parts of the header can be hidden from the public through the Web Interface using Manual List Configuration or by updating the header over email. To hide parts of the header, enter the "hide header" directive:

```
.HH ON
```

on a line by itself before the first header line you want hidden, and turn it off with the directive:

```
.HH OFF
```

on a line by itself after the last line you want hidden. You may have as many hidden and non-hidden blocks in the header as you wish. However, do not hide the first non-blank line, which represents the list's title.

6.2.3 Topics

List topics provide a way to run a mailing list (preferably moderated) where several sub-topics are being discussed in parallel, but some subscribers are only interested in a subset of the topics. For instance, a working group might have general discussions, decisions, and messages related to meetings. People who cannot attend the meetings can then opt out of last calls for hotel reservations and discussions about seafood restaurants, whereas people who have no time to follow the discussions can elect to get just the decisions. At any rate, such a compartmented list requires a certain discipline in order to be successful, as the posters must label their messages to indicate which topic(s) they belong to.

Through the **Topics** keyword on the Distribution tab of the List Configuration Wizard, the list owner can define up to 23 topics for the list. For example:

```
Topics= News,Benchmarks,Meetings,Beta-tests
```



Warning: Once set, you must never reorder the Topics keyword.

To save disk space, LISTSERV remembers which topics users have selected through their ordering in the **Topics** keyword. That is, "News" is "topic number 1" for LISTSERV, "Benchmarks" is "topic number 2", and so on. This means that you can change the name of a topic without requiring users to alter their subscriptions (for instance, you could decide that "Tests" is a better name than "Beta-tests" and just make the change). However, you must never change the order of the topics in the **Topics** keyword. If you want to remove a topic, replace it with a comma. For instance, to remove the "Meetings" topic, you would change the keyword to:

```
Topics= News,Benchmarks,,Beta-tests
```

Topic names can contain any character except space, colon and comma; the use of double quotes or equal signs is discouraged, as they require special attention when coding list header keywords. In addition, topic names may not start with a plus or minus sign, and the words ALL, NONE, RE, OTHER and OTHERS are reserved.

Posters label their messages through the subject field. LISTSERV first skips any possible sequence of 'Re:' keywords, and takes anything to the left of a colon as a list of topics,

separated by commas. The posting is considered to belong to all the topics listed before the colon. If none of these topics is valid for the list, it is classified in a special topic, "Other". If some of the topics are valid but others are undefined, the invalid ones are ignored. At any rate the subject field is left unchanged. Here is an example:

Subject: Benchmarks,News: Benchmarks for XYZ now available!

Messages that should be read by everyone can be posted to the special topic "All". Topic names can be shortened to any unambiguous abbreviation. In our example, "Be" is ambiguous because it could be either "Beta-tests" or "Benchmarks", but "Bench" is acceptable.

Subscribers select the topics they wish to receive with the SET command. The syntax is SET listname TOPICS: xxx where xxx can be:

- A list of all the topics the user wishes to receive. In that case these topics replace any other topics the user may have subscribed to before. For instance, after 'SET XYZ-L TOPICS: NEWS BENCH', the user will receive news and benchmarks, and nothing else.
- Updates to the list of topics the user currently receives. A plus sign indicates a topic that should be added, a minus sign requests the removal of a topic. For instance, 'SET XYZ-L TOPICS: +NEWS -BENCH' adds news and removes benchmarks. If a topic name is given without a + or - sign, + is assumed: 'SET XYZ-L TOPICS: +NEWS BENCH' adds news and benchmarks. The first topic name must have the plus sign to show that this is an addition, and not a replacement.
- A combination of the above, mostly useful to enable all but a few topics: 'SET XYZ-L TOPICS: ALL -MEETINGS'.

The colon after the keyword TOPICS: is optional, and TOPICS= is also accepted. Do not forget to include the special OTHER topic if you want to receive general discussions which were not labeled properly. On the other hand, if you only want to receive properly labeled messages you should not include it. ALL does include OTHER.



Important: Topics are active only when your subscription is set to MAIL. Digests are indexes always contain all the postings that were made because the same digest is prepared and sent to all the subscribers.

6.2.4 Normal List vs. Super-List

Using the **Sub-Lists** keyword on the Distribution tab of the List Configuration Wizard, the list maintainer can define a "super-list" (as in opposite of sub-list), that is, a "container" list that includes all the subscribers in a predefined set of sub-lists. This can be done recursively to any depth. For security reasons, only the maintainer can create a super-list. Concretely, the "Sub-lists=" keyword is protected from owner tampering in the same fashion as "Notebook=". The value is a comma separated list of all the sub-lists, which must all be on the same (local) machine. For instance:

Sub-lists= MYLIST-L,MYOTHERLIST-L

or, if you want to put each sublist on a separate line:

Sub-lists= MYLIST-L

Sub-lists= MYOTHERLIST-L

The default value for this keyword is null, e.g., to have no sublists.



Note: The super-list and all of its sublists must reside on the same LISTSERV server.

The only difference between a normal list and a super-list is what happens when you post to it. With the super-list, the membership of all the sub-lists is added (recursively) and duplicates are suppressed. Other than that, the super-list is a normal list with its own archives, access control, etc. You can even subscribe to it, and this is actually an important aspect of the operation of super-lists. If you are subscribed to the super-list itself, the subscription options used to deliver super-messages to you are taken from your subscription to the super-list, just like with any other list. All combinations are allowed, and in particular NOMAIL is allowed, meaning you don't want to get messages posted to the super-list. When you are subscribed to multiple sub-lists, on the other hand, things work differently:

NOMAIL subscriptions are ignored. You will get the super-message if you have an active (not NOMAIL) subscription to at least one sub-list. The idea is that the super-message must be equivalent to posting to all the sub-lists, without the duplicates. Since all it takes to get a message posted to all the sub-lists is a single non-NOMAIL subscription, this is how the super-list works. The only way not to get the super-messages is to subscribe to the super-list directly and set yourself to NOMAIL.

The DIGEST and INDEX options are ignored and internally converted to MAIL. The first reason is that, since in most cases the user will be on multiple sub-lists (otherwise you don't need a super-list in the first place), the only safe method to set subscription options for super-messages is by subscribing to the super-list so that there is no ambiguity. The second reason is that, in most cases, super-lists will be used for out of band administrative messages rather than for large volume discussions, so it is actually preferable to have the message sent directly. The third reason is that the super-list and sub-lists may not necessarily offer the same options (DIGEST and INDEX). In particular it is expected that many super-lists will not have archives. If you want a DIGEST or INDEX for the super-messages, you must subscribe to the super-list directly.

Topics, if defined, are evaluated on a per-list basis. That is, for every sub-list (and for the super-list), LISTSERV determines whether the topic of the message is one that you want to see. If not, it acts as if you were not subscribed to this particular list. Roughly speaking, this works very well if all the sub-lists have the same set of topics (or a well-defined set of common topics), and doesn't work well at all if every list has its own set of topics.

6.2.5 Mail-Merge

Advanced mail-merge features are available and can be accessed either by sending specially-formatted DISTRIBUTE jobs to LISTSERV or by using the web administration interface. The web interface is not a "wizard" but simply an interface that allows you to "cut and paste" a mail merge message and select different standardized groups of list subscribers to whom the message is to be sent.



Notes: LISTSERV's mail-merge functionality **REQUIRES** the use of LISTSERV's Embedded Mail Merge feature. For more information, see the EMM keyword in the [Site Configuration Keyword Reference](#) document. Mail-merge functions are documented fully in the [Advanced Topics Guide for LISTSERV](#).

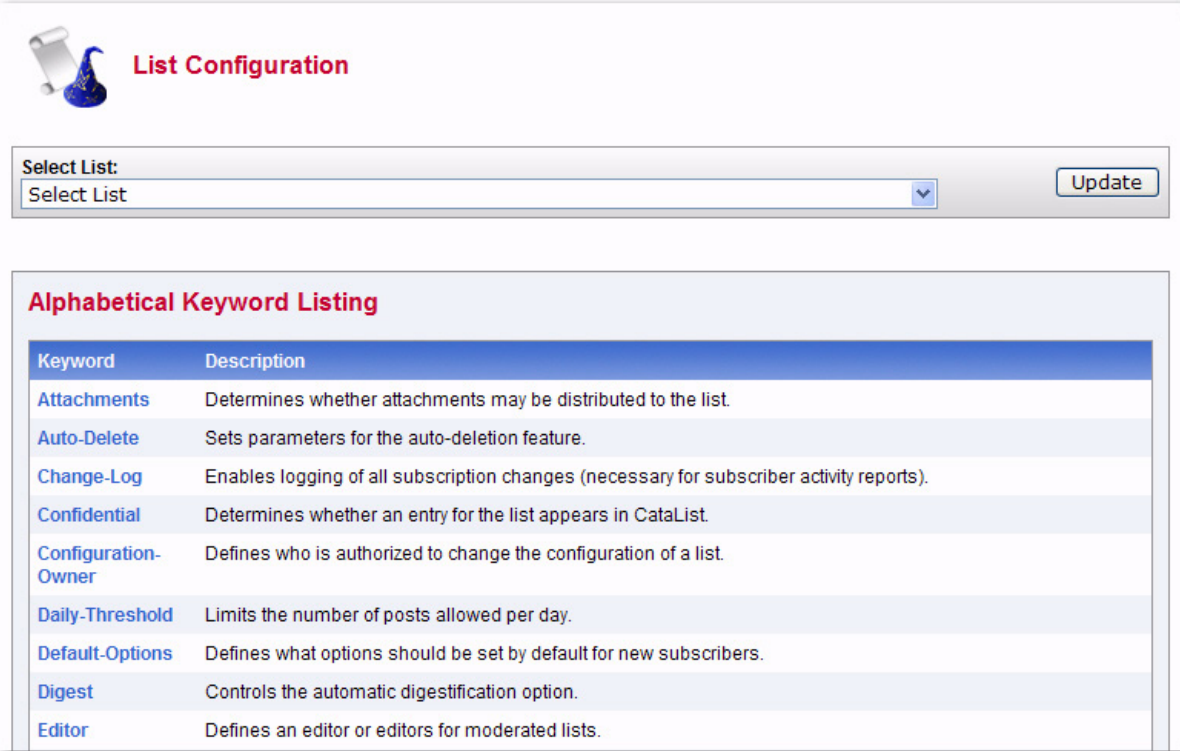
6.2.6 Alphabetical Keyword Listing Wizard

This wizard lists all of the list configuration keywords in alphabetical order and is useful if you know exactly which keyword you want to change. When you click on a keyword, a wizard for that particular keyword opens and shows only the parameters and details for that keyword.

To open the Alphabetica Keyword Listing Wizard, click on the **List Management** menu, select **List Configuration**, and then select **Alphabetical Keyword Listing**.

For details on using this wizard, please see the online help.

Figure 6-13 The Alphabetical Keyword Listing Wizard



Keyword	Description
Attachments	Determines whether attachments may be distributed to the list.
Auto-Delete	Sets parameters for the auto-deletion feature.
Change-Log	Enables logging of all subscription changes (necessary for subscriber activity reports).
Confidential	Determines whether an entry for the list appears in CataList.
Configuration-Owner	Defines who is authorized to change the configuration of a list.
Daily-Threshold	Limits the number of posts allowed per day.
Default-Options	Defines what options should be set by default for new subscribers.
Digest	Controls the automatic digestification option.
Editor	Defines an editor or editors for moderated lists.

6.3 List Configuration Task Wizard

The List Configuration Task Wizard guides you through providing digest and index versions of the list, in addition to the usual individual postings. To access this wizard, click on the **List Management** menu, select **List Configuration**, and then select **List Configuration Tasks**.

Make desired changes to each of the keywords on any of the tabs, and then press the **[Submit]** button. Alternatively, you can click on each keyword to access the Wizard pages for each keyword – these include lengthy explanations about the keywords.



Tip: To view help for any option in the List Configuration Task Wizard, simply click on the **Help** icon associated with it.

6.3.1 Administrators

The Administrators tab allows you to define the administrative roles for the list.

6.3.1.1 What is an Owner?

Owners are the primary administrators of the list. Email addresses with "owner" privileges may change the configuration and templates of the list, add and delete subscriptions, and change the settings on subscriptions.

Owners also receive email sent to the official list owner address. Unless otherwise specified, they receive notifications of subscriptions, signoffs, and error messages related to the operation of the list.

Some owner addresses may be designated as "quiet" owners. These addresses have all the owner privileges but do not receive any of the owner messages. Each list must have at least one non-quiet owner.

If you are replacing an owner, start by adding the new owner address. Next, make sure that the new address is able to function as an owner before removing the old address from the list of owners. If you make a mistake by first removing the old working address and then discovering the new address does not work, you may have to get the site administrator to fix your list for you.

6.3.1.2 What is an Editor?

On lists set to Send=Editor, editors are those addresses that are allowed to post directly to the list without moderation (that is, approval from a moderator).

On moderated lists where the Moderator keyword is not defined, the first editor listed in the Editor keyword acts as sole moderator.

On all lists, regardless of the value of the Send keyword, editors are not subject to limitations on the number of daily posts to the list, as set by the Daily-Threshold keyword.

The first editor listed in the Editor keyword must be an email address pointing to an individual. Other editors can point to lists on the same server including the current list; in that case, all the subscribers on the list have "editor" privileges. To enter the name of a list, you must enclose just the list name (not the list address) in parentheses.

If you do not specify the Editor keyword, the primary editor is the first listed Owner address.

6.3.1.3 What is a Moderator?

When a list is fully or partially moderated, all messages from "moderated" addresses are sent to the moderator(s) for approval. You can set up your list so that incoming messages go to each moderator in turn in a "round-robin" fashion, or so that all incoming messages go to all moderators.

If you do not specify the Moderator keyword, moderated messages get sent to the primary Editor address.

Messages are moderated in the following circumstances:

- The message is sent from a non-editor address on a list with Send=Editor.
- The message is sent from a subscription that is set to "REVIEW".
- The message is sent from a non-subscribed address on a list where Default-Options includes "REVIEW"

6.3.1.4 What is an Error?

Occasionally, problems occur on a list, and LISTSERV needs to know where to send error notifications. If you do not specify the Errors-to keyword, error messages will be sent to all non-quiet owners.

Listed below are some of the most common circumstances where error notifications might be sent to the address specified by Errors-to (or its default value):

- If the list is set to Auto-Delete=Yes, and there have been any delivery errors ("bounces") within the set time period, a "Daily Monitoring Report" is sent every morning.
- If the list is set to Auto-Delete=No, then all bounces are forwarded.
- If an email is received for the list from an address specified in the Filter keyword.
- If an email is received for the list containing mail headers pointing to the list (this may be an indication of a condition that would cause a mailing loop if the post were allowed to be delivered to the list).

6.3.1.5 What is a Notification?

Whenever someone subscribes or is added to the list, or someone signs off or is removed from the list, a notification is sent to the non-quiet owners (if Notify=Yes) or to the address(es) specified in the Notify keyword. To turn off notification of subscription activities, set Notify=No.

6.3.2 Security

The Security tab lets you set the security policies for your list.

LISTSERV's security options are wide ranging, from almost no protection (easiest to administer your list, but also most open to hacker attacks) to total protection requiring validation of each and every command sent to LISTSERV for your list. It is also possible to limit access to various aspects of your list, such as who can subscribe, who can review the list of subscribers, who can access the list archives, and whether the list is publicized in any way.

Security always requires a trade-off with convenience. Tightening up security requires you and your subscribers to go to extra trouble. LISTSERV provides many security options, but leaves the final choice up to the individual list owner. Think carefully in each case: How much does the need for convenience outweigh the need for security, or vice-versa?

6.3.2.1 Levels of Validation

The Validate keyword controls the level of command validation desired for your list. The default, Validate= No, requires password validation only for storing the list on the server. This is often sufficient for general needs. However, when a list is set this way, LISTSERV only compares the RFC822 "Sender:"/"From:" headers against the Owner= keyword(s) in the list header to determine whether or not the person ostensibly sending the commands has authority to do so. Otherwise, at this level, LISTSERV does not validate commands it receives for the list, assuming that the mail it receives is genuinely coming from a list owner. This level of validation does not protect the list from commands issued by hackers who have forged mail in the name of the list owner. If you run a list on a

controversial topic or just do not feel comfortable without at least some security, Validate= No is not the best option.

The next level is Validate= Yes. At this level, LISTSERV requires a password for all of its "protected" commands. This password is the sender's personal LISTSERV password as defined by the PW ADD command. The commands protected by this level are those that affect subscriptions or the operation of the list, such as DELETE or ADD. Users will also have to validate most commands that affect their subscriptions, but generally can do so using the "OK" mechanism rather than defining a personal password. Some user commands will be forwarded to the list owner for validation rather than accepting password validation from the user.

The next level is Validate= Yes,Confirm. At this level, LISTSERV will require validation with the "OK" mechanism (see below) by default, but will still accept passwords where appropriate. While the less-secure passwords are still accepted, this is considered a good compromise between list security and list owner and user convenience.

The next level is Validate= YES,Confirm,NoPW. At this level, LISTSERV will no longer accept passwords as validation for protected commands. The logic is that because of the way the "OK" mechanism is implemented, passwords are not as safe as "cookies". This is the recommended setting for lists that must be kept secure.

Two other levels are Validate= All,Confirm and Validate= All,Confirm,NoPW. These levels require "OK" validation for all commands that cause a change in state except for the PUT command. If NoPW is not specified, passwords are accepted where appropriate. With these levels, commands that do not cause a change in state (such as QUERY) do not require validation.

Lists which are set to either Validate= Yes,Confirm,NoPW or Validate= All,Confirm,NoPW may not be managed with the Web Interface, which is password-driven.

6.3.2.2 Subscription Options

You can control subscription requests by use of the Subscription= keyword. By default, this keyword is set to Subscription= By Owner, meaning that all subscription requests will be forwarded to the list owner for disposition. You can also refuse all subscription requests by setting Subscription= Closed.

To code a list for open subscriptions without list owner intervention, use Subscription= Open. If you would like to add protection against forged subscription requests or bad return mailing paths, then use Subscription= Open,Confirm. The latter will cause a subscription confirmation request to be sent to the prospective subscriber, which he or she must respond to using the "OK" confirmation mechanism.

6.3.2.3 Service Areas

Defining a service area is a method of restricting access to your list by hiding it from a LIST GLOBAL command and/or by limiting subscription requests to a defined set of host machines. Limiting access to certain lists can be highly desirable; for instance, keeping a student list for a class section at a university from being advertised or accessible by people all over the world. Without setting certain keywords appropriately, such a list would be visible to a LIST GLOBAL command.

If you wish to simply hide your list from a LIST command, but still allow anyone to subscribe to it if they know it is there, use the keyword Confidential= Yes. Users

subscribed to the list as well as the list owner(s) will be able to see the list if they issue a LIST command.

If you wish to hide your list from and refuse subscription requests from users outside the local area, you define two keywords:

```
Service= some.host.edu
Confidential= SERVICE
```

Service= can be set to a particular host machine, or a set of host machines or nodes. It can be set to Service= Local, meaning it will use either LISTSERV's global definition of which machines are Local, or the machines defined by the list keyword Local=. If you wish to set Service to Local, you should check with your LISTSERV site administrator to find out which nodes are considered local. If the global definition is not suitable, you can override it by defining the Local= keyword:

```
Local= bitnode1,some.host.edu,another.host.com
Service= Local
Confidential= Service
```

If there are many subdomains within your primary domain, you may wish to use the wildcard when defining the Local= or Service= keywords. For instance:

```
Service= HOST.COM,*.HOST.COM
```

defines the service area as "HOST.COM and all subdomains ending in .HOST.COM".

6.3.2.4 Who May Review the List of Subscribers?

The ability to review the subscriber list may be restricted to either subscribers or to list owners. This is done by setting the Review= keyword appropriately.

To restrict reviews of the list to subscribers only, set Review= Private. This is the default.

To restrict reviews of the list to list owners only, set Review= Owners.

To allow anyone, including non-subscribers, to review the list, set Review= Public. This is not recommended unless your LISTSERV server is operating on an intranet.

You can also restrict reviews to users within the list's service area by setting Review= Service and defining the Service= keyword appropriately.

6.3.2.5 Who May Access the Archives?

Restricting access to the list's notebook archive files is similar to controlling who may review the list. This is accomplished by setting the fourth parameter of the Notebook= keyword to an appropriate value. For instance,

```
Notebook= Yes,e:\listserv\lists\mylist,Monthly,Public
```

defines a monthly notebook on LISTSERV's A disk that is accessible by anyone. By changing Public to Private, access to notebooks becomes available only to list subscribers. The same access-levels are available for this keyword as for Review=.

Note: The location (second) parameter of the Notebook= keyword may be changed only by the LISTSERV maintainer.

If enabled, notebook archives are private by default.

6.3.2.6 Who May Post to the List?

The Send= list header keyword is the basic control for who may post mail to the list. If the list allows non-subscribers to post, set Send= Public.

For a list that does not allow non-subscribers to post, set Send= Private. Send= Private is the default.

If you want a further level of security for Send= Private, you may set Send= Private,Confirm. This setting requires each poster to confirm (with the "OK" mechanism) that the posting actually came from them. This can help in cases where a hacker might be trying to "spoof" (forge) mail from an otherwise legitimate subscriber. It is not recommended to set this in normal circumstances.

For a list where all posts are forwarded to a moderator/editor, there are three settings:

Send= Editor forwards all postings to the list editor (see the Editor= and Moderator= keywords). This setting allows the editor to make changes before forwarding the message back to the list. In order for the edited message to retain the original sender in the From: header, the editor's email program must be capable of inserting "Resent-" header lines in forwarded mail. If the email program is not capable of this, all such posts forwarded to the list will appear to be coming from the editor, not the original sender. Editors should check with their system administrator to find out whether or not their email program inserts the "Resent-" headers.

Send= Editor,Hold forwards a copy of the posting to the editor but differs from Send= Editor in that LISTSERV holds the posting for a period of time (usually 7 days) until the editor confirms the message with the "OK" mechanism (see below). Unconfirmed messages simply expire and are discarded by LISTSERV, so there is no need to formally disapprove a posting. This method of message confirmation is well suited to lists where it is not often necessary to modify the text of a posting, and also is an excellent work around if the editor's email program does not generate "Resent-" headers in forwarded mail.

Below is a sample of the editor-header for a list set to Send= Editor,Hold:

```
Date: Tue, 5 Mar 2002 10:47:21 -0500
From: "L-Soft list server at DRAGONFLY.DC.LSOFT.COM (14.3)"
<LISTSERV@DRAGONFLY.DC.LSOFT.COM>
Subject: MYLIST: approval required (XXXXXXXX)
To: Joe ListOwner <joe@PRUNE.EXAMPLE.COM>
```

This message was originally submitted by jack@UNIX.FOO.COM to the MYLIST list at DRAGONFLY.DC.LSOFT.COM. You can approve it using the "OK" mechanism (click on the link below), ignore it, or repost an edited copy. The message will expire automatically and you do not need to do anything if you just want to discard it. Please refer to the list owner's guide if you are not familiar with the "OK" mechanism; these instructions are being kept purposefully short for your convenience in processing large numbers of messages.

```
http://DRAGONFLY.DC.LSOFT.COM/scripts/
wa.exe?OK=XXXXXXXX&L=MYLIST
```


Send= Editor, Hold, Confirm is identical to Send= Editor, Hold except that postings coming directly from an editor must be confirmed (with the "OK" mechanism) by the editor who sent the message. This is the recommended setting for any moderated list or announce-only list because it protects the list from hackers who might try to forge (spoof) mail from a legitimate editor address.

A fourth method, called "self-moderation", exists for lists where subscribers are allowed to post freely, but non-subscriber posts are always sent to an editor for approval. To enable self-moderation, set

```
Send= Editor
Editor= userid@host,(listname)
```

Ensure that "listname" is in parenthesis. Self-moderation will catch all posts from non-subscribers, including posts from subscribers who are posting from a different address. For instance, if the subscriber originally signed up as joe@foo.com but is posting from joe@unix1.foo.com, LISTSERV will treat his mail as non-subscriber mail. Self-moderation may require some slight changes in individual user subscriptions in order for it to work seamlessly.

6.3.2.7 Restricting Subscriber Privileges

Another security issue involves protecting the list from people who refuse to play by the rules. LISTSERV includes several different levels of privilege restriction for these users.

The REVIEW personal option setting. By issuing a SET listname REVIEW FOR userid@host command to LISTSERV, you can moderate postings at the individual subscriber level. Postings from subscribers set to REVIEW are passed on to the Editor(s) or Moderator(s) of the list. If neither of these keywords are defined for your list, the postings are passed on to the primary list owner. The person who receives the postings can then determine whether or not to approve them. Note that the subscriber always receives notification that his or her posting has been forwarded to a moderator for approval. This is to avoid the impression that the subscriber's posting has been lost before reaching LISTSERV.

The NOPOST personal option setting. By issuing a SET listname NOPOST FOR userid@host command to LISTSERV, you can prevent a subscriber from posting to the list entirely. LISTSERV will reject postings from these subscribers and will not pass them on to a moderator. As with the REVIEW setting, note that the subscriber always receives notification that his or her posting has been rejected.

The NOPOST and REVIEW options are mutually exclusive.

You may define NOPOST or REVIEW as the default for all new subscriptions by including it in the Default-Options keyword.

6.3.3 Attachments

The Attachments tab allows you to define filtering of email sent to the list based on the contents.

LISTSERV allows you to restrict what types of attachments are allowed to be distributed to your list, put a limit on the size of messages, or even reject or moderate messages based on the mail headers or text of the message.

6.3.3.1 What is an Attachment?

Most email programs allow users to "attach" files to email messages they send. These files may be word processing documents, image files, or something else.

Since email is designed to transport plain text messages (and not image or word processing files), before attachments are sent they must be encoded in some way. The encoded attachment is then received and decoded automatically by the recipient's email program so that it can then be opened normally.

Unfortunately, attachments tend to be large, and sometimes they may contain viruses. For these reasons, many mailing list owners want to restrict the kind of attachments that are allowed on their mailing lists (or forbid them altogether).

There are two common methods that are used to encode attachments, MIME and UUEncoded.

6.3.3.2 MIME Attachments

The first and most common method of encoding is called MIME, which stands for "Multipurpose Internet Mail Extensions".

One advantage of this method of encoding for attachments is that messages containing MIME attachments have to include a description of what kind of file each attachment contains, using the list of types of files given at <http://www.iana.org/assignments/media-types/>.

By selecting specific MIME types to allow as attachments, you can tell LISTSERV to permit distribution of messages containing images, but not allow messages containing other kinds of attachments, for example.

Unfortunately, sometimes MIME attachments are not encoded correctly by the sending email program, so that the description of what kind of file the attachment is is inaccurate. (It might say that the file is an image when actually it is a spreadsheet document, for instance). For this reason, unless you ban attachments entirely, you cannot guarantee that no unwanted attachments will get through to the list.

It is possible to have plain text MIME attachments, with no encoding or special treatment required. As they are harmless, LISTSERV will not reject messages containing only plain text attachments.

LISTSERV's handling of most MIME attachment types is controlled by the Attachments keyword.

6.3.3.3 UUEncoded Attachments

Another, less common, method of encoding attachments is called "UUEncode".

Messages containing UUEncoded attachments do not include information about what kind of file the attachment contains. As a result, your only options are to allow all UUEncoded attachments or forbid them all.

LISTSERV's handling of UUEncoded attachments is controlled by the Attachments keyword.

6.3.3.4 HTML Messages

One special type of MIME message is the HTML message. HTML (Hypertext Markup Language) is the language used to write Web pages. Many email programs allow users to display and compose messages using HTML. Using HTML to compose email

messages allows for more creative messages, including graphics and colorful text, which are not permitted in standard plain text messages.

Unfortunately, not all email programs are able to interpret HTML messages correctly, displaying instead the raw HTML code. Also, in recent years some computer viruses have spread through email using HTML messages. For these and other reasons, some list owners prefer not to allow HTML messages on their mailing lists.

LISTSERV's handling of HTML messages is controlled by the Language keyword.

6.3.3.5 Exchange and Outlook Attachments

Microsoft Outlook and Exchange use their own attachment type, application/ms-tnef. LISTSERV removes these attachments from messages it distributes unless specifically configured not to because they are disruptive to subscribers who use non-Microsoft email clients.

LISTSERV's handling of these attachments is controlled by the Language keyword.

6.3.3.6 Size Limits

One objection to messages containing attachments is that they tend to be large, and therefore may take a lot of time to download. For this reason, LISTSERV allows you to set a limit on the size of the messages it will distribute. This is controlled by the Sizelim keyword.

6.3.3.7 Content Filtering

The CONTENT_FILTER mail template form, if present, contains filtering rules, one rule per line, empty lines ignored. Each rule has the following format:

```
[prefix:] pattern
```

The prefix, if present, can be a mail header tag (for example "Subject:"); "Header:" to check the whole header; or "Text:" to search the message text. The latter is the default if no prefix is supplied, it is provided in case the pattern contains a colon in the first word. If there are multiple mail header tags with the specified name (such as "Received:"), each such tag is searched and it is enough for one of them to match the pattern. If the requested tag is not present in the header, there is no match. A text search will search every line of the first text/plain part in the message. If there is no text/plain part, there is no match.

Regular comparisons such as those described above are not case sensitive. Patterns are standard LISTSERV patterns: the asterisk is the wildcard character. If there is no asterisk in the pattern, it is replaced with "*pattern*".

The content filter also supports "exact match" comparisons, which are triggered by a double colon. For instance:

```
Subject::
```

There are two significant differences between exact and regular match:

You must supply your own wildcard characters in an exact match (if you want to use wildcards). A regular match will insert leading and trailing wildcards if none are found. Thus, an exact match is the only way to make a comparison without wildcards.

You can make an exact match for the empty string. Empty regular matches are ignored since they map to a wildcard comparison for **, which would be always true. This also makes it possible to apply an exact match to a message that does not contain a specified

header. For instance, if you want all messages to contain a (mythical) KABOOM: RFC822 header, with an exact match you can tell LISTSERV to perform one of the content-filtering actions if the the header is not present. This is not possible with a regular match.

Note however that you cannot differentiate a header with an empty KABOOM field from a header with no KABOOM field.

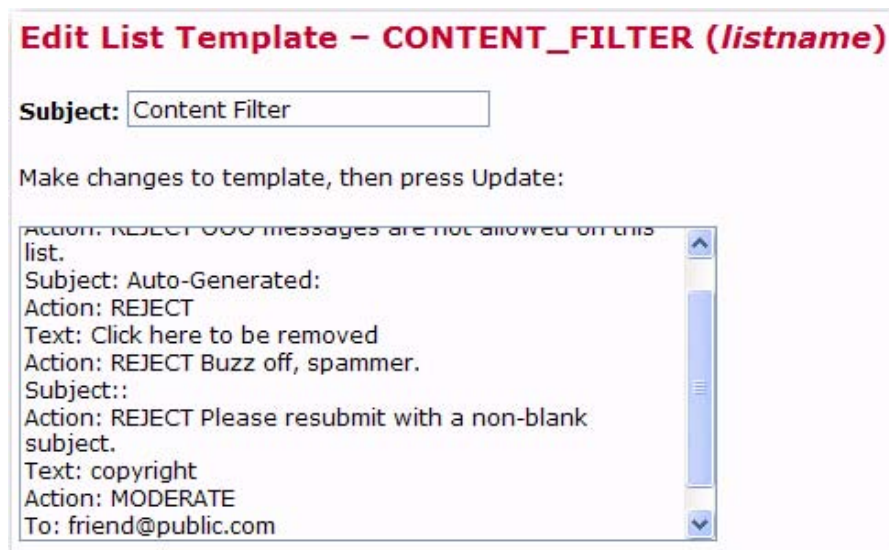
One of the most handy uses for the exact match syntax is to be able to write a rule to reject messages with blank subject lines. For instance:

```
Subject::
Action: REJECT Please resubmit your message with a non-blank subject.
```

Every rule can, optionally, be followed by an action rule. This has the following format:

```
Action: ALLOW
Action: REJECT reason
Action: DISCARD comment
Action: MODERATE
```

The available actions are the same for both regular and exact comparisons. For instance,



The default is "Action: REJECT" with no specified reason.

REJECT means that the message is rejected. The text of the rejection is fetched from the BAD_CONTENT mail template form, with the reason supplied as a variable called &COMMENT.

MODERATE means that the message is to be forwarded to the list editor to be manually approved or rejected.

DISCARD means that the message is to be discarded without further processing; any text following DISCARD is echoed to the LISTSERV log file.

ALLOW means that the message is allowed and all remaining rules are ignored. This could be used in moderated lists to allow certain posters to bypass certain filters, for instance:

Edit List Template – CONTENT_FILTER (listname)

Subject:

Make changes to template, then press Update:

Subject: Out of office
 Action: REJECT OOO messages are not allowed on this list.
 From: JOE@EXAMPLE.COM
 Action: ALLOW
 Text: Click here to be removed

In the example above, messages with Subject: lines containing "Out of office" are rejected. Messages containing the text "Click here to be removed" are also rejected UNLESS it came From: joe@example.com.



Note: A compilation of commonly used Internet message headers is available in [RFC 2076](#). The CONTENT_FILTER is a data template. None of the usual mail template substitutions or commands are valid.

6.3.4 Probes

The Probes tab allows you to configure probing and auto-deletion of bad addresses.

LISTSERV offers very flexible ways to automatically remove address that bounce (produce delivery errors) based on how many times they bounced or how long they have been bouncing. It also offers a mechanism to force subscribers to renew their subscription or be removed. Probing allows LISERV to send specially formatted messages so that bounces are handled even if the recipient's mail server is not sending standard bounces.

6.3.4.1 What is Probing?

Probing an address involves sending a unique message to that address (and that address only) to test if the address is valid or if the address generates bounces (messages that are returned to LISERV because the address is bad).

Probing is more reliable than other types of bounce collection because sometimes bounces from regular mailing list traffic do not include enough information to determine which subscription was responsible for the bounce.

When LISERV probes an address, it generally uses a unique address in its return path; therefore, if LISERV receives any mail to the probe address, it is certain that it is a bounce for the probed address.

6.3.4.2 What is Active Probing?

With active probing, a probe message is explicitly sent to the subscribers informing them that their addresses is being probed and usually instructing them to just discard the message. If a probe message bounces, depending on the mailing list's setup, the

address may be immediately removed or additional probe messages may be sent to make sure that the address is actually bad.

Generally, if active probes are enabled, then they are sent to subscribers whose subscriptions have been inactive for some time. This allows the list owner to ensure that there are not many old bad addresses set to NOMAIL subscribed to the mailing list.

The PROBE1 mail template contains the initial active probe message. Sometimes list owners will modify the message sent by the active probe mechanism to send information about the mailing list, such as a FAQ, a mailing list charter, or instructions on how to subscribe and sign off.

The follow-up message that may be sent if the initial probe fails can also be modified. It is contained in the PROBE2 template.

6.3.4.3 What is Passive Probing?

Passive probing is another option for testing subscriber email addresses. With passive probing, normal messages that are sent to the mailing list are used as the medium of the probe.

Essentially, when a message is sent to the mailing list, a certain percentage (determined by the Auto-Delete Probe setting) of the mailing list subscribers are sent individual copies of the message with customized return paths. These messages look just like regular mailing list messages, but any bounces generated by them go to the customized probe return addresses. This allows LISTSERV to automatically determine which address each bounce is for, even if the bounce is otherwise in a format that LISTSERV does not understand.

Depending on the setup of the Auto-Delete keyword, the bouncing address may be removed immediately or additional probes may be sent. As with active probes, any additional probe messages that are sent will contain the language in the PROBE2 template.

6.3.5 Digesting and Indexing

The Digest tab guides you through providing digest and index versions of the list, in addition to the usual individual postings.

Normally, LISTSERV sends messages out to subscribers as soon as it receives them, so that the subscribers receive the mailing list messages throughout the day. Some may prefer to get all of the messages at the same time, combined into a single piece of email. Such a collection of messages is called a digest.

Another option, similar to the digest, is for LISTSERV to send the subscriber a list of what messages have been distributed to the mailing list recently, along with information about when the message was posted, how big it is, and who sent it. This is referred to as an index.

6.3.5.1 What is a Digest?

Normally, LISTSERV sends mailing list messages out to subscribers as soon as it receives them, so that the subscribers get messages throughout the day.

Some may prefer to get all of the messages at the same time, combined into a single piece of email. Such a collection of messages is called a digest.

Depending on the volume of messages that go out over a mailing list, it may make sense to have the digest go out once a day, once a week, or once a month.

LISTSERV allows subscribers to get digests in three formats: HTML, MIME, and NOMIME NOHTML. Subscribers can individually choose the format that works best in their email clients. Each email client is different, so subscribers should experiment with the different digest styles to find the one they prefer.

6.3.5.2 What is an Index?

An index, similar to the digest, is another option for receiving one message that summarizes a collection of messages from LISERV. LISERV sends the subscriber a list of what messages have been distributed to the mailing list recently, along with information about when the message was posted, how big it is, and who sent it.

Indexes are only available for archived mailing lists that have digests enabled. Indexes are sent out at the same time as the digests.

Indexes are available in HTML and NOHTML formats. If HTML is used, the index includes a link to each message in the Archive Interface. If NOHTML is used, the index includes instructions on how to retrieve the messages the subscriber wants.

6.3.5.3 What is a Plain Text Digest?

Plain text is the simplest form of digest. All email programs should be able to read plain text digests without any difficulty. The basic form of this type of digest is given in RFC 1153.

At the beginning of a plain text digest there is a list of the subjects of the messages in the digest:

```
Subject: MYLIST Digest - 10 Jun 1996 to 11 Jun 1996
There are 5 messages totalling 50 lines in this issue.
Topics in this issue:
  1. Request for comments
  2. Another subject, another ruler (3)
  3. Project deadline on Thursday
```



Note: If the mailing list is used for discussions, typically there will be multiple messages for each subject.

After the topics list, the contents of the DIGEST-H template (if any) will be displayed:

```
This mailing list is sponsored by XYZ Industries. If you
would prefer to receive this mailing list as individual
messages instead of as a digest, send a "SET MYLIST MAIL
NODIGEST" to listserv@lists.example.com.
```

After the contents of the DIGEST-H template (if any), the messages will appear in the order LISERV received them, separated by lines consisting of 30 hyphens.

Since the digest is plain text, any HTML messages will appear uninterpreted (the raw HTML code will appear) and any attachments will appear in their encoded form.

Additionally, any special characters (smart quotes or accented letters, for example) may not display correctly in plain text digests.



6.3.5.4 Displaying UTF-8 Encoded Messages in the Body of a Digest

LISTSERV 16.0 debuts a new "plaintext" digest format that will properly display UTF-8-encoded messages in the body of the digest. The old "plaintext" digest was limited to 7-bit ASCII text, and as such, was only able to display UTF-8 messages as received, that is, in their non-human-readable UUencoded or Base-64-encoded form.

This new feature produces a digest that looks exactly like an RFC1153-compliant digest. In reality, it is a MIME message, and it can handle any character set. For each message in the digest, LISERV locates the most pertinent text part, decodes it, and uses that as the text of the message in question. The format is no longer formally RFC1153, but it looks as close to RFC1153 as possible while still being useful for non-ASCII messages.

1. All RFC822 headers are '=?' decoded, converted to UTF-8, and output in plain text.
2. All MIME messages are run through LISERV's MIME parser and reduced into a plain text part. This is converted to UTF-8 and is the final display "copy".
3. Any attachments and additional material other than the plain-text part are discarded. This includes additional plain-text parts.
4. Step 2 can fail if there is a MIME or encoding error, or if there is simply no plain text data in the message. The error that is thrown is controlled by one of two new mail templates, depending on the error: MSG_DIGEST_FRAGMENT_MIME_ERROR or MSG_DIGEST_FRAGMENT_NO_TEXT, both found in DEFAULT MAILTPL. The content of the appropriate template is then output.



Note: These are single-line templates; therefore, they are not suitable for a complex explanation of what happened. Any edits to these templates should be kept short and to the point.

5. If there is only one HTML part in the message, there is no plain text part and step 2 fails.
6. To make all this possible, the whole digest is charset=UTF-8.

Setting "Misc-Options= OLD_NOMIME_DIGEST" in the list header will force LISERV to revert to the original RFC1153 format, if for some reason this is required. Normally it should not be; however, even if you can guarantee that every posting is 7-bit ASCII text and nothing else, the new format has the downside of using Unicode. While this of course is identical to ASCII for pure ASCII input, the message may be marked as Unicode in people's mail clients.

6.3.5.5 What is an HTML Digest?

Recipients of an HTML digest who have email programs that are programmed to handle such digests will see an index of the day's messages followed by the contents of the DIGEST-H template (if any).

Clicking on a subject in the table of contents takes you down to the relevant message or messages in the index. Clicking on any of the messages will then take you to the message in question.

HTML digests include information about the content of each message (through MIME); therefore, each message should display normally.



Note: Since the digest includes all of the messages as MIME attachments, all of the links in the HTML digest index are of the form: "cid: content-id" (see RFC 2111 for more information about this type of URL). Unfortunately, some email clients, even some that otherwise support HTML, do not handle such references correctly. For this reason, some subscribers may not be able to use HTML digests.

Figure 6-14 Example of an HTML Digest

MYLIST Digest - 7 Feb 2002 to 8 Feb 2002 **Table of contents:**

- [Request for comments \(2\)](#)
- [Another subject, another ruler](#)
- 1. [Request for comments](#)
 - [Re: Request for comments \(02/08\)](#)
From: Fred Q. Public <fred@EXAMPLE.COM>
 - [Re: Request for comments \(02/08\)](#)
From: Abby L. Normal <abby@EXAMPLE.COM>
- 2. [Another subject, another ruler](#)
 - [Re: Another subject, another ruler \(02/08\)](#)
From: Frank Lee <frank@EXAMPLE.COM>

Browse the [MYLIST online archives](#).

6.3.5.6 What is a MIME Digest?

A MIME digest is midway in complexity between a plain text digest and an HTML digest. It contains a table of contents of the topics discussed in the digest:

```
There are 4 messages totalling 86 lines in this issue.
Topics of the day:
  1. Request for comments (3)
  2. Another subject, another ruler
```

Each message in the digest is then included as a MIME attachment. The subscribers access the messages as they would any other type of attachment. And, since MIME standards require that the type of content of each attachment is identified, all messages should appear normally without the sort of display problems that plain text digests have.

6.3.6 Subject Tags

The Subject Tags tab guides you through defining a Subject-Tag and setting your list members' subscription options to use the Subject-Tag.

Some mailing list subscribers like to have list mailings easily identified by a "subject tag", which is a string within square brackets added to the beginning of the "Subject" line in the email headers.

6.3.6.1 How do I make SUBJECTHDR a default option?

SUBJECTHDR may be abbreviated to SUBJ. If "SUBJECTHDR" or "SUBJ" is already part of your Default-Options setting, it already is the default. Otherwise, add SUBJ or SUBJECTHDR to the Default-Options text box. If the text box already has something in it, separate SUBJ from the other settings with a comma (",").

You cannot combine SUBJECTHDR with any of the other header options. If any other "header" setting is in Default-Options, it must be removed from the Default-Options string before adding SUBJECTHDR. The other mail header options are:

- FULLHDR (this is the default if you do not specify something different in Default-Options)
- SHORTHDR
- IETFHDR
- DUALHDR
- FULL822 (obsolete)
- SHORT822 (obsolete)

6.3.6.2 Why would a subscriber want a subject tag?

Some subscribers like list mail to be easily recognizable as soon as it is received. Others like to set filters in their email software to move list mail into a special folder based on the subject line.

In contrast, other subscribers prefer not to tie up valuable "real estate" in the subject line with extra text, and therefore do not like it.

The preference is often based on what email software the subscriber is using, but often it is just a matter of taste. That is why this is a setting that can be set differently for each subscriber.

Note that even if you set SUBJECTHDR as the default for all subscribers, any subscriber can change their own subscription to their preferred header setting.

6.3.6.3 Why specify a different Subject-Tag?

You do not have to set the Subject-Tag list keyword in order to make it available to your subscribers. If you do not provide a subject tag, the list name will be used for any subscription set to SUBJECTHDR.

The primary reason for setting the Subject-Tag keyword is to provide a short text string to be used instead of the list name. Some email client programs limit the number of characters they display in the subject line. Long list names in a subject tag may take up most or all of the space provided for display by these email clients.

6.3.6.4 Why would you make SUBJECTHDR the default?

When you specify a "default option" you are telling LISTSERV to use this setting for all new subscribers. Specifying a new default option will not change the options for current subscribers.

Set default options because you specifically want new subscribers to have those options, or you believe that most subscribers would want those options and you want to save them the trouble of setting options themselves.

When deciding whether to set SUBJECTHDR as a default option, you may want to consider the technical sophistication of your typical subscriber. People with little technical experience are not likely to experiment with their subscription settings. They are unlikely to know that SUBJECTHDR is available, unless they are on a list where it is the default. If they do find out about it, they are more likely to ask the list owner to set it for them than to set it themselves.

Set SUBJECTHDR as the default if you think it will be welcome by most of your future subscribers. They can always change it to something else if they do not like it.

6.3.6.5 Why set the SUBJECTHDR tag for all subscribers?

If your list already has subscribers, and you think that most of the current subscribers would prefer this setting, you might consider changing all subscriptions to this setting.

Before doing so, you may want to check your subscribers' current settings. Some subscribers may have specific reasons for wanting a different header setting. Their preferences may be based on the email client they are using to read the list mail.

It may be appropriate, depending on the nature of the list, to warn your subscribers if the appearance of the list messages is about to change.

If you decide you want to notify your subscribers of the change, consider how you want them notified. If you choose to have LISTSERV notify the subscribers, they will receive a standard message from LISTSERV, which they may or may not understand. Depending on the technical sophistication of your subscribers, you may want to change the settings "quietly" (without automatic notification) and then send a message of your own to the list explaining the change you just made.

6.3.7 CataList

The CataList tab shows all the options available for cataloguing your list in CataList.

Listing in CataList makes it easier for prospective subscribers to find your list. CataList is known for keeping an up-to-date record of lists. It is updated several times a day, so the listings are generally never more than a few hours out of date.

6.3.7.1 Why List in CataList?

Listing in CataList makes it easier for prospective subscribers to find your list. CataList is known for keeping an up-to-date record of lists. It is updated several times a day, so the listings are generally never more than a few hours out of date.

For an example of a CataList entry, see the [entry for the LSTOWN-L list](#).

6.3.7.2 Missing Listing in CataList

If you can not find the entry for your list in CataList, there are a number of possible causes:

- The list is set to Confidential=Yes or Confidential=Service. Confidential=Yes prevents the list from being publicized automatically anywhere. Confidential=Service prevents the list from being listed in CataList, but allows the list archives (if they exist) to be listed on the server's index page.
- The list is new or was changed to Confidential=No only recently. Wait a day to make sure you have allowed time for database updates, and then check again.

- The LISTSERV server is set to "standalone" runmode. Only "networked" LISTSERV servers benefit from CataList listings, spam and spoof warnings, and other advantages. If neither of the first two bullets applies, check if DRAGONFLY.DC.LSOFT.COM is in CataList. If not, you may want to ask your LISTSERV site administrator whether it would be possible to set up the server to run in "networked" or "tableless" runmode.

6.3.7.3 HTML Description in CataList

An HTML description of your list is optional. If provided, it will be used by CataList, the online catalog of LISTSERV lists. If a text-based list description is not provided, the HTML description will also be used on the list's home page or archive index. If a text-based list description and an HTML description are both provided, the text-based list description will be displayed by default. This default behavior can be changed by editing the list's OBJECT-A0-LISTDESC template.

To delete an existing HTML description, check the **Delete HTML Description** checkbox and click **[Submit]**.

6.3.7.4 Hide Header from CataList

The CataList entry for each list contains a link to "Take a look at the list's configuration", that is, the list header. Even if the list is not in CataList, it is possible for anyone to request a copy of your list header by sending the REVIEW command to LISTSERV. If there are parts of your list header that you consider to be sensitive information, you should hide these parts by using the .HH (hide header) directives.

You do so by adding .HH ON on a line by itself before the lines you want to hide, and .HH OFF on a line by itself after the lines you want to hide. You can also intersperse hidden lines and non-hidden lines.

If you want to hide the entire configuration, you can add .HH ON directly after the list title and .HH OFF directly before the list description.

Figure 6-15 Hiding a Header from CataList



6.3.8 Banners

The Banners tab guides you through changing all the banners that may be added to messages. LISTSERV provides several templates that can be inserted at the top and bottom of the individual messages that are sent to the list, as well as on digests and index mailings.

6.3.8.1 What is a Banner?

As a mailing list administrator, sometimes you want all messages distributed to a mailing list to include a specific piece of information. For instance, you might want to have instructions on how to sign off the mailing list at the bottom of each message, or to have a copyright notice at the top of the messages. LISTSERV's banner templates allow you to do this.

The top banner is typically used for information that is deemed to be critical and requires prominent placement, for example copyright notices.

The bottom banner is typically used for general information.

6.3.8.2 HTML vs. Text Banners

It is usually desirable to specify both a plain text and an HTML version of any banner you add to messages. That way the HTML version, including any formatting or markup you specified, will be added to any HTML messages distributed to the mailing list, and the plain text version will be added to any non-HTML messages. If you only specify a plain text banner, then LISTSERV will use it on both HTML and plain text messages. Since the plain text banner will not contain any HTML formatting, it may not look the way you want when added to an HTML message.

The HTML banners should contain HTML formatting tags so that they look "right" when added to an HTML message. The banners are included into the "body" of the HTML part of the mail message, and so should include only tags that are appropriate in the "body" part of an HTML document. For example, if you use a Web design tool to create the banner, you should open the HTML file that it produces in a text editor (Notepad, for example), and then cut and paste only the lines of HTML code that are between the <body> and </body> tags into the template. You may also want to add
 tags at the beginning and at the end of the template to make sure that there is always a line break between the banner and the rest of the email.

If you do not want to use a Web design tool, it is easy to make a very basic HTML banner by taking the text banner and adding some simple HTML tags. For example:

6.3.9 Mail Templates

The Mail Templates tab introduces you to a few mail templates that many list owners choose to customize. LISTSERV provides a large number of templates that control the contents of email messages automatically sent to subscribers (and others) by LISTSERV under various circumstances. This tab lists a few of the most popular ones.



For detailed instructions, see the online help by clicking the **Help** icon associated with the Mail Templates tab or click on the **Template Commands** or **Template Variables** tip at the top of the tab.

6.4 Subscriber Management

The Subscriber Management screen allows the list owner to examine or delete a subscription and add a new subscriber to the list.

To open the Subscriber Management screen, click on the **List Management** menu, and then select **Subscriber Management**.

Figure 6-16 Subscriber Management

The screenshot shows the 'Subscriber Management (LISTSERV-AT-WORK)' web interface. At the top, there is a logo and a link to 'LISTSERV-AT-WORK Home'. Below this is a 'Select List:' dropdown menu with 'LISTSERV-AT-WORK LISTSERV at Work' selected, and an 'Update' button. There are two tabs: 'Single Subscriber' (selected) and 'Bulk Operations'. The main content area is titled 'LISTSERV-AT-WORK' and contains two sections: 'Examine or Delete Subscription' and 'Add New Subscriber'. The 'Examine or Delete Subscription' section has a 'Name or Address:' input field with suggestions: 'henry@somewhere.com', 'Henry Brown', and 's*lvia'. Below the input field are 'Search in LISTSERV-AT-WORK' and 'Clear' buttons. The 'Add New Subscriber' section has an 'Email Address and Name:' input field with suggestions: 'henry@somewhere.com Henry Brown' and 'Henry Brown <henry@somewhere.com>'. Below the input field are two radio buttons: 'Send Email Notification' (selected) and 'Do Not Notify the User'. At the bottom of this section are 'Add to LISTSERV-AT-WORK' and 'Clear' buttons. At the very bottom of the interface, there is a link: 'Review List Members: In Browser | By Email'.

6.4.1 Adding a New Subscriber to the List

To add a new subscriber, click on the **List Management** menu, and then select **Subscriber Management**. The Subscriber Management screen opens. On the Single Subscriber tab, click the **Select List** drop-down menu to select the list you want to add the subscriber to. In the **Add New Subscriber** section, enter the email address and name of the new subscriber.

Figure 6-17 Adding a New Subscriber

This is a close-up screenshot of the 'Add New Subscriber' section. It shows the 'Email Address and Name:' input field with suggestions: 'henry@somewhere.com Henry Brown' and 'Henry Brown <henry@somewhere.com>'. Below the input field are two radio buttons: 'Send Email Notification' (selected) and 'Do Not Notify the User'. At the bottom of this section are 'Add to HAT-TEST' and 'Clear' buttons.

Then, select whether or not to send an email notification to this subscriber, and click the **[Add to List]** button.



Note: The full name of the subscriber is optional. If omitted, then the user will be added anonymously to the list.

6.4.2 Examining or Deleting a Subscription

To add a new subscriber, click on the **List Management** menu, and then select **Subscriber Management**. The Subscriber Management screen opens. From the Single Subscriber tab, you can view or delete a subscription. This works very much like the "SCAN" command. Simply enter your criteria in the text box and click **[Search in List]**.

Figure 6-18 Examining or Deleting a Subscription

If there is no match for your entry, then you will get back the same page but with a Scan: No match message at the top. If, on the other hand, your search is successful, one of two things will happen.

If there are multiple matches for your criteria, a screen will be displayed with a scrollable list box containing all of the matches

Figure 6-19 Select Subscriber to Examine or Delete

Next, simply choose the user you want to examine or delete and click on the appropriate button. If you did not find what you were looking for, you can press the **[New Search]** button to get a new search screen.

If there was only a single match to your query, then the preceding screen will be bypassed and you will go directly to the next screen, which is the Subscriber Management screen for the subscription. It displays the values of all the settings for that subscription, including the subscription date and name. From this screen, you can delete the subscription or change the name, the email address, or the subscription options associated with the subscription.

The following settings are available in the **Subscription Type** section:

- **Regular** – With a "regular" subscription, you receive individual postings immediately as they are processed by LISTSERV.

- **Digest (Traditional), Digest (MIME format), and Digest (HTML format)** – With a "digest" subscription, you receive larger messages (called "digests") at regular intervals, usually once per day or once per week. These "digests" are collections of individual list postings. Some lists are so active that they produce several digests per day.

Digests are a good compromise between reading everything as it is posted and feeling like the list is clogging your mailbox with a multitude of individual postings.

There are three digest formats: a "traditional", text-only format; a MIME format, which (with mail clients that understand MIME digests) "bursts" the individual messages out of the digest so that you can read them separately; and an HTML format, which requires an HTML mail clients.

- **Index (Traditional) or Index (HTML format)** – With an "index" subscription, you receive short "index" messages at regular intervals, usually once per day or once per week. These "indexes" show you what is being discussed on the list, without including the text of the individual postings. For each posting, the date, the author's name and address, the subject of the message, and the number of lines is listed. You can then download messages of interest from the server (the index contains instructions on how to do that).

An index subscription is ideal if you have a slow connection and only read a few hand-picked messages. The indexes are very short and you do not have to worry about long download times. The drawback of course is that you need to reconnect to retrieve messages of interest from the server.

You can choose to have the index sent to you in either a traditional format (plain text) or in HTML format with hyperlinks.

The following settings are available in the **Mail Header Style** section:

- **Normal LISTSERV-style header** – "Full" mail headers (normally the default), containing Internet routing information, MIME headers, and so forth. The ('To:') header contains the address of the list.
- **LISTSERV-style, with list name in subject** – "Full" mail headers (like the default) except that a "subject tag" is added to the subject line of mail coming from the list. If there is no subject tag defined in the list's configuration, the name of the list will be used. This can be very useful for sorting and filtering mail.
- **"Dual" (second header in mail body)** – Dual headers are regular short headers followed by a second header inside the message body. This second header shows what list the message is coming from ('Sender:'), the name and address of the person who posted it ('Poster:'), the poster's organization, if present, and the message subject. Dual headers are helpful if your mail client does not preserve the original return email address.
- **sendmail-style (advanced option)** – This option selects sendmail-style headers, i.e. an exact copy of the original, incoming mail header with the addition of a ('Received:') line and a ('Sender:') field. Some technical people prefer this type of header.

- **Normal LISTSERV-style (RFC 822 Compliant)** – (For Advanced Use Only) "Full" mail headers (like the default) except that the ('To:') header contains the recipient's email address instead of the list address.

The following settings are available in the **Acknowledgements** section:

- **No acknowledgements** – LISTSERV will not send any acknowledgement at all when you post to the list. This is probably not a good setting unless you really do not want any feedback from LISTSERV as to whether or not your posting was received and distributed.
- **Short message confirming receipt** – Typically, this is the default setting, although it can be overridden by the list owner. If you choose this setting, LISTSERV will send you a short message whenever you post to the list, confirming the distribution of your message and telling you how many people it was sent.
- **Receive copy of own postings** – Some people prefer this setting over the short acknowledgement message. It tells LISTSERV to send you a copy of your own postings so that you can see exactly how it appeared on the list (useful if you are behind an unreliable gateway or firewall).

The following settings are available in the **Miscellaneous** section:

- **Mail delivery disabled temporarily** – This option toggles the receipt of mail from the list. You may want to disable mail delivery if you will be away from your mail for an extended period of time.
- **Address concealed from REVIEW listing** – This option conceals you from the list of subscribers shown by the REVIEW command. Note that the list owner and the LISTSERV administrator can always get the complete list of subscribers, regardless of this setting. Nowadays, most lists are configured so that only the list owner can use the REVIEW command, but some lists still allow subscribers to get a listing of all the other participants.
- **User is exempt from renewal/probing** – The user will not receive renewal reminders, if enabled for the list. In addition, the user will be exempt from address probing, which is used to determine if the address is valid or if the address generates bounces.
- **User may bypass moderation** – The user may post to the list without having the message approved by the moderator.
- **All postings sent to the list owner for review** – All postings to the list will be sent to the list owner for review before it is officially posted on the list.
- **User may not post to the list** – This options simply means that the user can not post to the list.

Figure 6-20 Subscriber Management Screen

Subscriber Management (TEST1) TEST1 Home

View or Set Subscription Options

Notification Options:

☒ Send Email Notification
☐ Do Not Notify the User

Name:
Email Address:
Subscribed Since: 10 May 2007

Subscription Type

☒ Regular [NODIGEST]
☐ Digest (traditional) [NOMIME DIGEST]
☐ Digest (MIME format) [NOHTML MIME DIGEST]
☐ Digest (HTML format) [HTML DIGEST]
☐ Index (traditional) [NOHTML INDEX]
☐ Index (HTML format) [HTML INDEX]

Mail Header Style

☒ Normal LISTSERV-style header [FULLHDR]
☐ LISTSERV-style, with list name in subject [SUBJECTHDR]
☐ "Dual" (second header in mail body) [DUALHDR]
☐ sendmail-style [IETFHDR]
☐ Normal LISTSERV-style (RFC 822 Compliant) [FULL822]

Acknowledgements

☐ No acknowledgements [NOACK NOREPRO]
☒ Short message confirming receipt [ACK NOREPRO]
☐ Receive copy of own postings [NOACK REPRO]

Miscellaneous

☐ Mail delivery disabled temporarily [NOMAIL]
☐ Address concealed from REVIEW listing [CONCEAL]
☐ User is exempt from renewal/probing [NORENEW]
☐ User may bypass moderation [EDITOR]
☐ All postings sent to list owner for review [REVIEW]
☐ User may not post to list [NOPOST]

If you are deleting a subscription or changing/updating its options, the two options at the top of the screen allow you to choose whether or not a notification should be sent to the subscriber about the change or deletion. The two options when used with the **[Delete]** button are strictly equivalent to "DELETE listname userid@host" and "QUIET DELETE listname userid@host", respectively, and the other equivalent commands are formatted identically. **Send Email Notification** is the default.

The **[Delete From All Lists]** button is strictly equivalent to the command "DELETE * userid@host" and is used to delete the user from all lists on the local server (for site managers) or from all lists on the local server that are owned by the invoker (for list owners).

If you are making changes to the user's name field, address, or user options, use the **[Update]** button to commit the changes. If you make changes to both the options and the identification fields, user option settings are updated first, and then changes are made to the name and address fields.

Following either a **[Delete]** or an **[Update]** operation, the main Subscriber Management screen is displayed along with a message indicating the success or failure of your operations.

6.4.3 Reviewing List Members

At the bottom of the Single Subscriber tab, there is a section where you can review your list members using a browser or email. Clicking on either the **In Browser** or **By Email** option will open the Command Interface.



Tip: You can now see if a list has sub-lists when you are reviewing your subscribers in a browser.

The Command Interface is used for submitting LISTSERV commands that are not otherwise facilitated by the Web interface. For some commands, the response is automatically displayed by the Web interface. For others, a special command parameter must be used in order to display the response in the browser, otherwise the response is sent by email. Yet, other commands are only able to respond by email.

For more details of using the Command Interface, see Section 6.5 [Submitting LISTSERV Commands](#).

6.4.4 Bulk Operations

The Bulk Operations tab allows a list owner to upload an input file containing email addresses and (optionally) names, one address per line, and either add all the email addresses in the file to the list (optionally replacing the current subscribers) or remove them from the list.

To access, click **List Management**, and then select **Subscriber Management**. The Subscriber Management screen opens. Click on the Bulk Operations tab.

The input file is created on your own machine with an ASCII text editor. After clicking the **[Import]** button you will see a command response similar to the following:

- If the **Add the imported address to “List”; do not remove any subscribers** option is selected:
ADD: no error, 202 recipients added, no entry changed, no duplicate, none forwarded.
- If the **Remove all subscribers from “List”, and add the imported address** option is selected:
DELETE: 14 subscribers removed.
ADD: no error, 38 recipients added, no entry changed, no duplicate, none forwarded.
(If this option is selected, but no input file is specified, then you will only get the DELETE message.)

- If the **Remove the imported addresses from “List”**; do not add any subscribers option is selected:

DELETE: 93 subscribers removed.

- If the **Remove the imported addresses from all lists** option is selected:

DELETE: 243 subscribers removed.

DELETE: 109 subscribers removed.

Global deletion process complete, 352 entries removed.

- If you do not supply an upload file where required, or if your browser does not support the RFC1867 file upload extension, you get the following message:

Your browser did not upload any file during the transfer. Assuming you did fill in the file input box, the most likely cause is that your browser does not support the file upload extension (RFC1867).



Notes: Bulk operations are not enabled by default. The site manager must enable this functionality explicitly. If you get an error 2 when you click on the **[Import]** button, this means that the "upload" directory has not been created. If you get an error 13 when you click on the **[Import]** button, this means that the "upload" directory has been created but the CGI program user does not have write permission in that directory. In addition, the input file must be a plain text file (not a word processor document or spreadsheet) and must contain one address per line, optionally followed with a space (or TAB) and the subscriber's name. In addition, the subscribers being added or deleted will not be notified.

Figure 6-21 The Bulk Operations Tab

Subscriber Management (UPDATES-HDMail)

UPDATES-HDMail Home

Select List:
UPDATES-HDMail HDMail Product Updates Update

Single Subscriber **Bulk Operations**

UPDATES-HDMail

Caution: Some of the functions offered through this page will **remove all subscribers** from UPDATES-HDMail. Double-check your selection before submitting.

Function:

- ☒ Add the imported addresses to UPDATES-HDMail; do not remove any subscribers.
- ☐ Remove all subscribers from UPDATES-HDMail, and add the imported addresses (to remove all subscribers, select this option and omit the input file).
- ☐ Remove the imported addresses from UPDATES-HDMail; do not add any subscribers.
- ☐ Remove the imported addresses from all lists.

Input File: Browse... Import

6.5 Submitting LISTSERV Commands

The LISTSERV Command Interface is used for submitting LISTSERV commands that are not otherwise facilitated by the Web Interface.

For some commands, the response is automatically displayed by the Web Interface. For others, a special command parameter must be used in order to display the response in the browser, otherwise the response is sent by email. In addition, other commands are only able to respond by email.

To access the LISTSERV Command Interface, click on the **List Management** menu, and then select **LISTSERV Command**.

The Command Interface can only be used for single line commands. In particular, the PUT command will not work through the Web Interface. Multi-line commands must be sent by email.

A selection of frequently used commands is available at the bottom of the screen.



In addition, the LISTSERV Command Interface now has a **Clear Command** option, making it easy to clear a command so you can issue multiple commands in a sequence.

Figure 6-22 Clearing the Command Interface

LISTSERV Command Interface (DISCUSSION)

[DISCUSSION Home](#)

Select List:
 DISCUSSION L-Soft's Discussion List Update

Information about the DISCUSSION list is being mailed to you.

DISCUSSION

(Clear Command)
 New Command: INFO DISCUSSION Enter Command

Useful Commands:
[Release \(Server info\)](#) | [Show Stats](#) | [Show License](#) | [Show Points All](#) | [Show All Lists](#) | [Show Hardware](#) | [Served Off Addresses](#)

[Review DISCUSSION](#) | [Query DISCUSSION](#) | [Confirm DISCUSSION](#) | [Free DISCUSSION](#) | [Hold DISCUSSION](#) | [Count Subscribers](#) | [Count by Country](#) | [Count by Topic](#) | [Reindex DISCUSSION Archives](#)

Send over Email:
[Info about DISCUSSION](#) | [List of DISCUSSION Subscribers](#) | [User Commands](#) | [Owner Commands](#) | [Admin Commands](#) | [Keyword Reference](#)

Section 7 List Moderation Using the Web Interface

The List Moderation Interface allows list moderators to moderate a list through the Web Interface, rather than using email. To start moderating a list, click on **List Moderation** from the Toolbar. The List Moderation screen has a list selection area at the top. Click the drop-down menu to select the list you want to work with. Unlike the other list selection areas that show the lists for which the logged-in user is the owner, this one only shows the lists for which the user may act as moderator.

If you are moderator to more than one list, select the list you want to moderate. If there are no messages for you to moderate, the only action possible is to “refresh” using the **[Refresh]** button, which looks for messages to moderate in the selected list.

If there are messages to moderate, a table, like the one below, is displayed and contains all the messages that were sent to the logged-in email address for moderation.

Figure 7-1 Moderated List Table

MODERATED LIST

Subject	From	Date
<input type="checkbox"/> A sample message	"Non Subscriber" <nonmember@example.com>	Tue, 26 Mar 2002 18:56:08 -0500
<input type="checkbox"/> Moderated message from a subscriber	"List Subscriber" <member@example.com>	Wed, 27 Mar 2002 00:00:11 -0500

The first column contains check boxes, one per message. The second column shows the **Subject** of the message. The third column shows the **From** email address that sent the message. The fourth column shows the **Date** in the message (not necessarily when the message was received). When **All Moderators** is checked, two extra columns are displayed showing whether your account received the moderation request or other accounts did; if the `Moderator=` keyword is set to "All" with multiple moderators, both columns will be checked for all messages.

7.1 Accepting or Rejecting the Message

If you can tell at a glance which messages to approve or reject, or to change report criteria (using the search), simply check all the boxes corresponding to the messages you want to act upon.

Next, select the action you want to perform:

- **Approve** – Approve all messages whose boxes have been checked.
- **Reject** – Reject all messages whose boxes have been checked
- **Reject and Serve Off** – Reject all messages whose boxes have been checked and serve off the senders. This bans future postings from their email addresses.
- **Refresh** – Refresh the display without accepting or rejecting any messages.

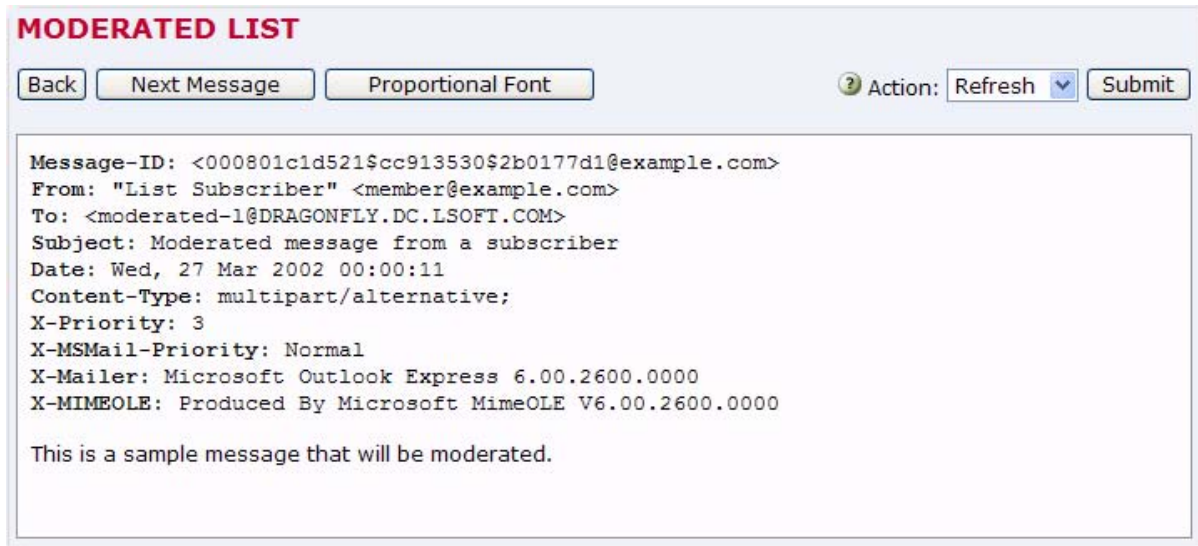
Click the **[Submit]** button to perform the selected action on the messages and/or re-display the screen using the selected criteria.

7.2 Viewing the Message

Sometimes, you cannot immediately tell whether a message should be accepted or rejected without reading the contents of the message. To open the message contents, click on the subject of the message to display a new view that shows only one message.

If the message is multi-part (the example includes an HTML part), the part displayed is determined by your preferences. The main mail headers of the message are also displayed. If you wish to approve or reject the message, select the appropriate action and then press the **[Submit]** button.

Figure 7-2 Message View



MODERATED LIST

Back Next Message Proportional Font Action: Refresh Submit

Message-ID: <000801c1d521\$cc913530\$2b0177d1@example.com>
 From: "List Subscriber" <member@example.com>
 To: <moderated-l@DRAGONFLY.DC.LSOFT.COM>
 Subject: Moderated message from a subscriber
 Date: Wed, 27 Mar 2002 00:00:11
 Content-Type: multipart/alternative;
 X-Priority: 3
 X-MSMail-Priority: Normal
 X-Mailer: Microsoft Outlook Express 6.00.2600.0000
 X-MIMEOLE: Produced By Microsoft MimeOLE V6.00.2600.0000

This is a sample message that will be moderated.

7.3 Subscriber Settings

If the logged-in email address is an owner of the list, and if the "From" address of the message is a subscriber to the list, then the email address in the main moderation view is a link and the **Change Subscriber Settings** option is shown on the message view. Both lead to the Interface for viewing and changing the subscriptions settings.

Both are convenient for lists that have `Default-Options=REVIEW`, with a policy of changing the subscription to "NOREVIEW" after the first on-topic posting to the list. The moderator/owner may look at the message, decide that the subscriber may be removed from moderation, change the subscription settings to "NOREVIEW", and then approve the message. Be sure to change the settings first because the message will no longer be displayed after it has been approved or rejected.

Similarly, if a subscriber has sent inappropriate material to the list, the moderator/owner may want to delete the subscription.



Note: After using the subscriber settings, always use the **[Back]** button to return to the List Moderation Interface and not the "Back" feature of the browser. The "Back" feature of the browser may cause past approve/reject commands to be repeated, and errors will be displayed, complaining that the "cookies" do not correspond to any existing messages awaiting moderation. This does not do any harm at all to your list, but may still be disconcerting.

Section 8 Generating List Management Reports in the Web Interface

The Web Interface makes it possible for those list owners with multiple lists to obtain reports on their lists and search these lists for specific characteristics.

To access the List Management Reports, click on the **List Management** menu, select **List Reports**, and then select either **List Reports**, **Subscriber Reports**, or **List Activity Reports**.

8.1 List Reports

To create a list report:

- Select the list or lists to generate the report for. By default, you will see all of the lists you own.



Tip: The List Report contains a “search” option. Use this to search for a specific list or group of lists.

- Next, select the columns you want to include in the report.



- Optionally select a **Report Format**. The default is a web-based table viewed **On Screen**. You can also get the results in a "Comma-separated values" (CSV) format that can then be saved from your web browser to a file on your computer, allowing you to import the data into any reporting software that supports CSV files. For the CSV format, you can select either **CSV Format (Subset)** or **CSV Format (All)**.



Note: For reports that span multiple pages, the **CSV Format (Subset)** option will only include data from the current screen in the report. For example, if a list contains 100 subscribers but only 50 subscribers are shown on screen at a time, then the CSV file will only contain the 50 subscribers shown on screen. Also, if you are using any narrowing options, only the data that fits the search criteria is included in the CSV file. The **CSV Format (All)** option, on the other hand, will include all the data in the CSV file regardless of the number of pages or any search or narrowing options.

- Finally, click **[Submit]**.



Note: Each column heading is a clickable link that will sort the column in ascending or descending order. A small arrow pointing up or down will appear indicating the order. Each list name in the report is a clickable link to the list configuration page for that particular list.

8.2 Subscriber Reports

To create a report, select the list or lists to generate the report for. By default, you will see all of the lists you own. Next, select the columns you want to include in the report. Finally, click **[Submit]**.

The report generated is not just a simple report. It also provides the means of manipulating the reported data. Once the list owner has generated a report, it is possible to change subscription settings or delete one or more subscribers. It is also possible to add subscribers to the list.

The Subscriber Report contains a “search” option. Use this to search for a specific subscriber.

The Subscriber Report also lets you add a subscriber to a list. To add a new subscriber to the list, type the user's email address followed by the full name. Then, choose whether or not to notify the user that he has been added and click on the **[Add Subscriber]** button.

The Subscriber Report also lets you change subscriber settings. Once the report has been generated, it becomes possible to change any subscriber's settings (except for the subscription date):

- Select the subscriber by checking the box to the left of the subscriber name. Check the **Invert** box to select every subscriber EXCEPT those that you have checked off.
- Make the change at the bottom of the column using the drop down boxes. All of the the selections for changing the current options will be listed. Highlight the new option by clicking it.
- Optionally select a **Report Format**. The default is a web-based table viewed **On Screen**. You can also get the results in a "Comma-separated values" (CSV) format that can then be saved from your web browser to a file on your computer, allowing you to import the data into any reporting software that supports CSV files. For the CSV format, you can select either **CSV Format (Subset)** or **CSV Format (All)**.



Note: For reports that span multiple pages, the **CSV Format (Subset)** option will only include data from the current screen in the report. For example, if a list contains 100 subscribers but only 50 subscribers are shown on screen at a time, then the CSV file will only contain the 50 subscribers shown on screen. Also, if you are using any narrowing options, only the data that fits the search criteria is included in the CSV file. The **CSV Format (All)** option, on the other hand, will include all the data in the CSV file regardless of the number of pages or any search or narrowing options.

- Click the **[Submit]** button to save your changes. If you would like your subscribers to receive an email notification that their options have been changed, check the **Send email notification of changes** box. Otherwise, no notification will be sent.

To view and make changes to the subscription options of individual subscribers, click on the subscriber name. If a name is not available, click on the **No Name Available** link. The Subscriber Management screen will open for the selected individual. Make any changes and then click **[Update]**.

To delete subscribers, check the boxes to the left of the subscriber names you want to remove, and then click the **[Delete Selected Subscribers]** button.

Each column heading is a clickable link that will sort the column in ascending or descending order. A small arrow pointing up or down will appear indicating the order.

8.3 List Activity Reports

Two types of activity reports can be generated using this Interface. The first report is a history report. This report simply displays the changelog records that match the selection criteria, one record per line, similar to reading the changelog yourself. The second report is a statistics report, which allows you to reduce the changelog records to numbers, based on several different criteria. This is where you can determine (for instance) how many postings were made to a given list on a range of specified dates, how many times a particular LISTSERV command was issued, and so forth.

You should generally choose only one report column on which to report statistics (Event or Email Address or Details) as each value of the text field is counted independently and sorted in alphabetical order, and rows could then include unrelated values and cause some confusion.

Changelog reports are only available for lists where the Change-Log listheader keyword is set to Yes, and only for the time span it was enabled. Enabling Changelogs in the list header will not suddenly make it possible to view past list activity. If Changelogs are enabled, but set to rotate over time (for example, monthly changelogs), the reports are only available for the current report.

To generate a changelog report:

- Select a list.
- Define the **Report Type** to create a historical or statistical report or both. If you check both report fields, you will get two reports. The historical report shows the actual changelog entries. The statistical report shows counts and averages.
- Define the **Report Entries** to determine which events to include in the report.
- Define the **Report Interval** to set the date range for the report.
- Optionally, you can also select the reporting interval for statistics reports. The default is to show totals for the entire reporting period.
- Optionally select a **Report Format**. The default is a web-based table viewed **On Screen**. You can also get the results in a "Comma-separated values" (CSV) format that can then be saved from your web browser to a file on your computer, allowing you to import the data into any reporting software that supports CSV files. For the CSV format, you can select either **CSV Format (Subset)** or **CSV Format (All)**.



Note: For reports that span multiple pages, the **CSV Format (Subset)** option will only include data from the current screen in the report. For example, if a list contains 100 subscribers but only 50 subscribers are shown on screen at a time, then the CSV file will only contain the 50 subscribers shown on screen. Also, if you are using any narrowing options, only the data that fits the search criteria is included in the CSV file. The **CSV Format (All)** option, on the other hand, will include all the data in the CSV file regardless of the number of pages or any search or narrowing options.

- Click the **[Submit]** button.

Section 9 Customizing Mail and Web Templates Using the Web Interface

The majority of Web Interface pages and administrative messages sent to subscribers can be customized. Customizing your list's Web pages and administrative messages creates an individual look and tone for your list. Subscribers will see Web pages and read administrative messages that are specific to your list, containing pertinent information and instructions rather than just generic pages and messages.

The template editor allows the site administrator and list owner to customize the majority of the Web Interface Pages and Administrative Messages sent out by LISTSERV. There are two types of templates you can customize – Web and Mail.

Web Templates (also referred to as Dynamic Web Templates) control the pages produced by the Web Interface. These pages are produced dynamically when they are accessed. What gets displayed by the browser depends on the circumstances and may change depending on who is accessing the Interface, which list they are accessing, the settings of that list, and so on.

Commands in Web Templates begin with a plus sign "+" and variables begin with an ampersand followed by a plus sign "&+".

Mail Templates control text produced by LISTSERV itself. Although generally categorized as mail templates, they actually fall into three different types:

- Mail Templates control the contents of email messages sent by LISTSERV. A mail template is a complete email message. Formatting commands are available, substitutions that make sense in the context of the specific message are available, and while other templates may be imbedded with the .IM command, the message is in and of itself ready for LISTSERV to send.
- Message Templates supply text that will ultimately be shown to the user. These messages may be included in a mail template; or they may be included in an email sent by LISTSERV in response to LISTSERV commands sent by email to the LISTSERV command address; or they may be returned to the Web Interface in response to commands sent through the Web Interface. Limited formatting is available.
- Message Fragments templates are the lowest level of mail templates. Fragments are pieces of text produced by LISTSERV as parts of other messages or emails. For example, list digests must follow a certain format dictated by the Internet RFCs. Therefore, it is not possible to provide a complete mail template for digests. However, some of the text within the digest is not mandated by the RFCs, and so LISTSERV provides some fragment templates to control these parts, for example MSG_DIGEST_FRAGMENT_DATERANGE1 to control the date range and MSG_DIGEST_FRAGMENT_PREAMBLE1 to control the text preceding the table of contents. Formatting commands are generally not available in fragments.



Notes: Mail-merge fields are now allowed in Mail templates.

When editing mail templates, LISTSERV will tag each mail template with the character set that was in use when the changes were made. Unedited mail templates will have ISO-8859-1 as the default character set.

Commands in Mail Templates begin with a period "." and variables begin with an ampersand "&".

To access the template editor, click on the **List Management** or **Server Administration** menu, select **Customization**, and then select either **Web Templates** or **Mail Templates**.

Once the template editor has opened, simply select the list and template you want to work with. The template editor also lets you create new template.

For more information about customization templates, see the [Customization Manual for LISTSERV 16.0](#) or click on the link(s) located in the **Tips** section at the bottom of the editor.

Many list owners prefer to customize some of the administrative messages sent to subscribers and the main Web page for the list archives. However, it is not necessary to customize any of the templates. Customizing templates can be complex for new users, so it is not covered in depth in this manual. The table below lists some of the templates you will most likely want to customize.

Table 9-1 Templates to Customize

Template	Description
Welcome Message	A template for a message sent to new subscribers, in addition to the standard "signup" message. By default, no welcome message is sent.
Farewell Message	A template for a message sent to subscribers leaving the list. By default, no farewell message is sent.
\$SIGNUP	A standard message sent to all new subscribers, usually with list instructions. By default, this template is included in both the SIGNUP1 and ADD1 templates.
SIGNUP1	A message sent to new subscribers who have signed on by themselves. By default, includes the \$SIGNUP template.
ADD1	A message sent to new subscribers who have been added to the list by the owner. By default, includes the \$SIGNUP template.
SIGHNOFF2	A message sent to users who are trying to sign off a list they are not signed on to or from an address that was not the one used to subscribe with.
DELETE1	This message is sent to a subscriber who has been removed from the list by a list administrator.
POST_EDITOR	A message sent to a user when their post has been forwarded to a moderator (Send= Editor or subscription is set to REVIEW).

Template	Description
INFO	A message containing information about the list. Sent to anyone sending the "REVIEW <i>listname</i> " command to LISTSERV. By default includes the non-hidden parts of the list header.
BOTTOM_BANNER	Adds a text message to the bottom of each message posted to the list. Usually used for disclaimers, tips, and/or signoff or other instructions. If you plan to send HTML messages through the list, you should also define a BOTTOM_BANNER_HTML template containing the HTML version of the BOTTOM_BANNER.
TOP_BANNER	Adds a text message to the top of each message posted to the list. Usually used for a copyright or short legal warning which absolutely has to be seen by each and every reader. If you plan to send HTML messages through the list, you should also define a TOP_BANNER_HTML template.
AUTODEL1	This message is sent to users who are automatically deleted from the list when delivery errors (bounces) to that address exceed the threshold set by the Auto-Delete= keyword (or its default setting).
WWW_INDEX	The main web page for the list archives. This template creates the <i>listname.html</i> file. This is a "static" web template. Requires knowledge of HTML to customize (see below).

Things to consider when editing templates:

- Deleting a template in the Web Interface template editor or removing it from the *listname.mailtpl* file (when customizing by email) will not prevent a message from being sent. It merely deletes the customizations and results in the default message being sent. The way to prevent a message from being sent is to customize the template by inserting the directive ".QQ" on a line by itself. Be careful, if you add ".QQ" to a template that is included in other templates, you will be canceling those templates as well.
- Changing the wording of most mail templates is fairly straightforward. Simply be aware of the following:
 - Variables begin with an ampersand ("&") and may end with a semicolon. You may remove variables from the text if you do not wish to customize that part of the text. For example, the ADD1 template contains the text "You have been added to the &LISTNAME mailing list (&TITLE) by &MBX(&INVOKER)". The "&LISTNAME", "&TITLE", and "&MBX(&INVOKER)" will be replaced by the list name, title, and the name and email address of the owner that added the subscription. This is easily changed to "You have been added to the &LISTNAME mailing list (&TITLE) by the list owner" if you do not wish to reveal the name and address of the owner.
 - If an ampersand ("&") appears in the template text (this includes any URLs with ampersands) it must be doubled ("&&") in order to keep LISTSERV from treating it like a variable.

- Lines starting with a dot (".") are template "directives". Do not make changes to these lines without understanding what they do.
- Changing the Web templates requires knowledge of HTML. You cannot edit the Web templates using an HTML editor – the templates contain variables and template directives of which HTML editors are ignorant.

9.1 Forcing the Browser to Use a Specific Character Set



The Global Skins (SKIN) template now contains an option to force the browser to use a specific character set for cases where the browser's auto-detect feature fails to use the correct character set. This variable, DEFAULTCHARSET, can be defined for the site as a whole or for particular lists that are known to use a specific character set. To set, simply change the default setting of 0 to the appropriate charset, such as ISO-8859-5.

To change this variable for all of your lists, click on the **Server Administration** menu, select **Customization**, and then select **Web Templates**. Search for the Global Skins template by entering either Global Skins or SKIN in the **Search Template** field, and then click **[Search]**. Edit the template by changing DEFAULTCHARSET to the desired character set.

To change this variable for a specific list, click on the **List Management** menu, select **Customization**, and then select **Web Templates**. Select the list you'd like to work with, and then search for the Global Skins template by entering either Global Skins or SKIN in the **Search Template** field and click **[Search]**. Edit the template by changing DEFAULTCHARSET to the desired character set.

9.2 Using the HTML Mail Template Gallery



The HTML Mail Template Gallery allows the list owner or the site administrator to select a template style for the list or the site as a whole.

To select a side-wide default template, click on the **Server Administration** menu, select **Customization**, and then select **Mail Templates**. The Mail Template Editor opens.

To select a default template for a list, click on the **List Management** menu, select **Customization**, and then select **Mail Templates**. The Mail Template Editor opens. Click the **Select List** drop-down menu, and then select the list to define the default template for. The Mail Template Editor refreshes with the HTML Mail Template Gallery displayed.

In the HTML Mail Template Gallery, select an HTML mail template by clicking the **[Select]** button associated with the template you want to use. The screen will refresh with the selected template displayed in the **Mail Template Style** box.

To edit a template, click the **Easel** icon. The Edit HTML Mail Template screen opens, which is where you can make any changes necessary. If you have made changes to a default template and would like to undo those changes, click the **Revert** icon associated with that template.

To create a new template based on a default template, simply edit that template and change the template name before clicking the **[Save]** button. The template will now be saved with the new name.

To create a new template from scratch, click on the **Create HTML Mail Template** option. The Create HTML Mail Template screen opens, which is where you will enter the content for the new template. See the special documentation on that screen to learn more about the format of the HTML mail templates.



Tips: When creating new templates, make sure that the template name starts with HTMLMAIL- and it will be automatically included in the list of available HTML mail templates within the editor.

If you are creating a newsletter template, you are required to have both an HTMLMAIL- and a TEXTMAIL- template (for the alternative text part). If you do not want the alternative text to be part of your newsletter you can always leave the TEXTMAIL- template blank and delete it on the Fine Tune tab when using the Newsletter Templates wizard.

To delete a template, click on the **Recycling Bin** icon associated with that template.

To refresh the mail template gallery, click the **Refresh Gallery** option.

9.3 Using a Conditional Expression Evaluator



The web interface CGI now supports LISTSERV's conditional expression evaluator for templates, made available as +IF. For instance, it is now possible to code the following in a template:

For wildcard string matches:

```
+IF HELLO =* HE*  
*do something here*  
+ENDIF
```

For comparing numbers, etc.:

```
+IF (5 > 6)  
*do something here*  
+ENDIF  
(+ENDIF is an alias for +EB; you can use either.)
```


Section 10 Working with Archives in the Web Interface

The Archives feature consists of all files other than notebook logs that have been stored on the LISTSERV host for your list.

10.1 The Server Archive Index

The Server Archive Index screen displays any list that has been marked for archiving and can be accessed by clicking **Email Lists** on the Toolbar.

Figure 10-1 The Server Archive Index

LISTSERV Archives

DRAGONFLY.DC.LSOFT.COM

This LISTSERV server is located at DRAGONFLY.DC.LSOFT.COM. Below you will find all lists that have been configured for public archiving. To access a specific list, simply click on the name of the list in the table. To find confidential or unlisted lists, type in the list name in the search box to the right. For lists that have been configured with an HTML description, you can get more information about the list by hovering the mouse over the list name.

[Previous] [B~M] [T~U]

List Name	List Title
TEMPLATES	HTML Template Test List (4 Subscribers)
UPDATES-HDMAIL	HDMail Product Updates (95 Subscribers)
UPDATES-LISTSERV	LISTSERV Product Updates (476 Subscribers)
UPDATES-LISTSERV-MAESTRO	LISTSERV Maestro Product Updates (320 Subscribers)

[Previous] [B~M] [T~U]

Lists per Page:

Access Unlisted Lists

Options

- [Log Out](#)
- [Change Password](#)
- [Search Archives](#)
- [Preferences](#)

Resources

- [About LISTSERV](#)
- [LISTSERV Documentation](#)



Note: The Server Archive Index can be set as the starting page when logging on to LISTSERV. For more information, see Section 5.6 [Setting Your Personal Preferences](#).

This screen offers you several options and resources:

- **Log In/Log Out** – Click to log in or log out of LISTSERV.

- **Get Password/Change Password** – Click to get a LISTSERV password or to change an existing password.
- **Search Archives** – Click to open the Archive Search page.
- **Preferences** – Click to open the Preferences page.
- **About LISTSERV** – Click to open the L-Soft website in your Internet browser. You will be taken directly to LISTSERV's product page.
- **LISTSERV Documentation** – Click to open the L-Soft website in your Internet browser. You will be taken directly to the Documentation page.
- **CataList Email List Search** – Click to open [CataList](#), the official catalog of LISTSERV lists. For more information, see Section 5.8 [Accessing CataList](#).

To view a quick snapshot of what the list is, simply hover your mouse over the list name. A popup of the list will appear.



Note: This setting can be turned on/off on the Archive Preference tab. For details, see Section 5.6.2 [Archive Preferences](#). In addition, this feature requires javascript.

Figure 10-2 Previewing a List

LISTSERV Archives

DRAGONFLY.DC.LSOFT.COM

This LISTSERV server is located at DRAGONFLY.DC.LSOFT.COM. Below you will find all lists that have been configured for public archiving. To access a specific list, simply click on the name of the list in the table. To find confidential or unlisted lists, type in the list name in the search box to the right. For lists that have been configured with an HTML description, you can get more information about the list by hovering the mouse over the list name.

[Previous] [B~M] [T~U]

List Name	List Title
TEMPLATES	HTML Template Test List (4 Subscribers)
UPDATES-HDMAIL	HDMail Product Updates (95 Subscribers)
UPDATES-LISTSERV	LISTSERV Product Updates
UPDATES	

[Previous] [B~M] [T~U]

2 Lists per Page

Receive announcements about new versions, level sets, patches and updates by email. Being subscribed to the alerts guarantees that you will receive all important LISTSERV product news.

Access Unlisted Lists

Options

[Log Out](#)

[Change Password](#)

[Search Archives](#)

[Preferences](#)

Resources

[About LISTSERV](#)

[LISTSERV Documentation](#)

DRAGONFLY.DC.L

To access the archives for a specific list, simply click on the list name in the table. If the list is not displayed, then use the **Access Unlisted Archives** field to search for it. The **Access Unlisted Archives** field allows you to access the archives of confidential lists that are not listed in the archive index. Simply type the name of the list in the text box and click **[Submit]**. This field will also work for non-confidential lists.

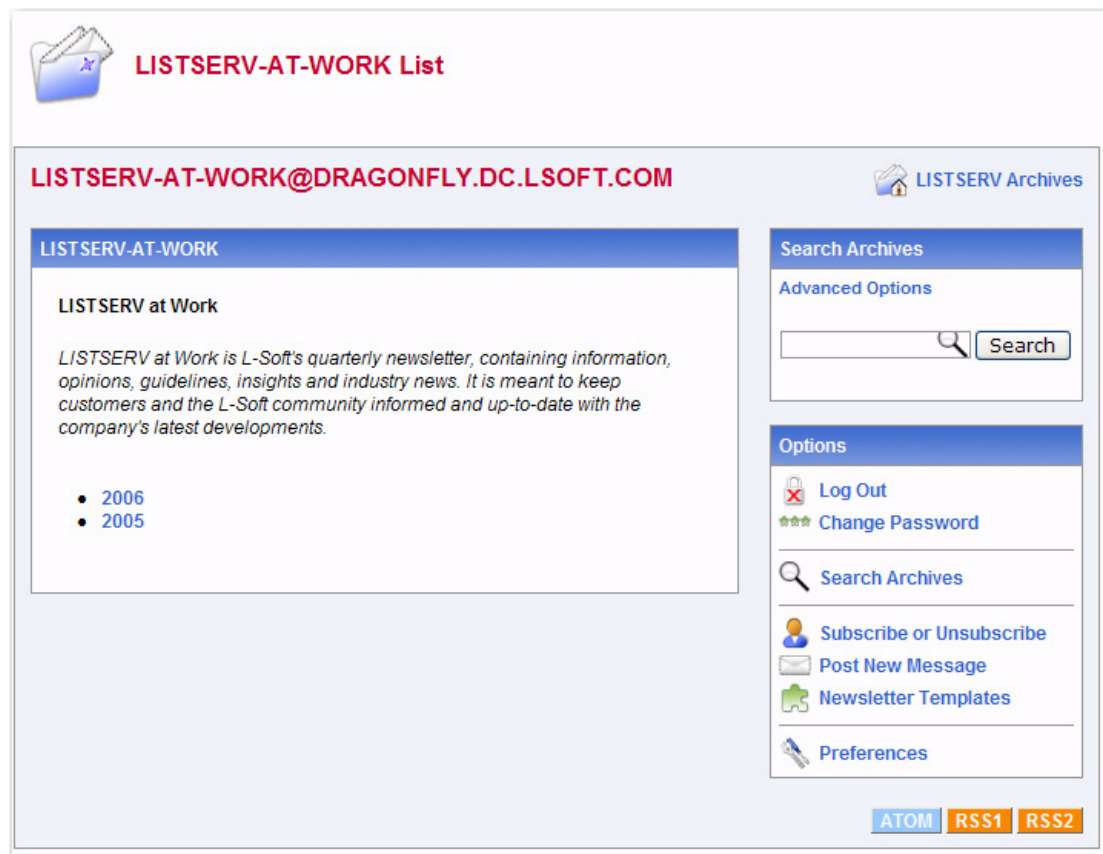
The Confidential list configuration keyword determines whether the list should be hidden from the archive index or not. Confidential= No is the default value and indicates that the list is not confidential. Confidential=Service indicates that the list is to be hidden from users who are not in the list's service area (see the Service keyword for more details) but not from other users. Confidential= Yes means that the list is unconditionally confidential. See the [List Keyword Reference](#) document for more information on these keyword.

After clicking on the list you want to view archives for, you are taken to the List Archives screen for that list. See the next section for information.

10.2 Viewing List Archives


The List Archive screen displays the archives for a list. Depending on how the list archive was set up, you can access the archives by week, month, or year.

Figure 10-3 List Archives Screen



This page also offers you several other options:

- **Log In/Log Out** – Click to log in or log out of LISTSERV.

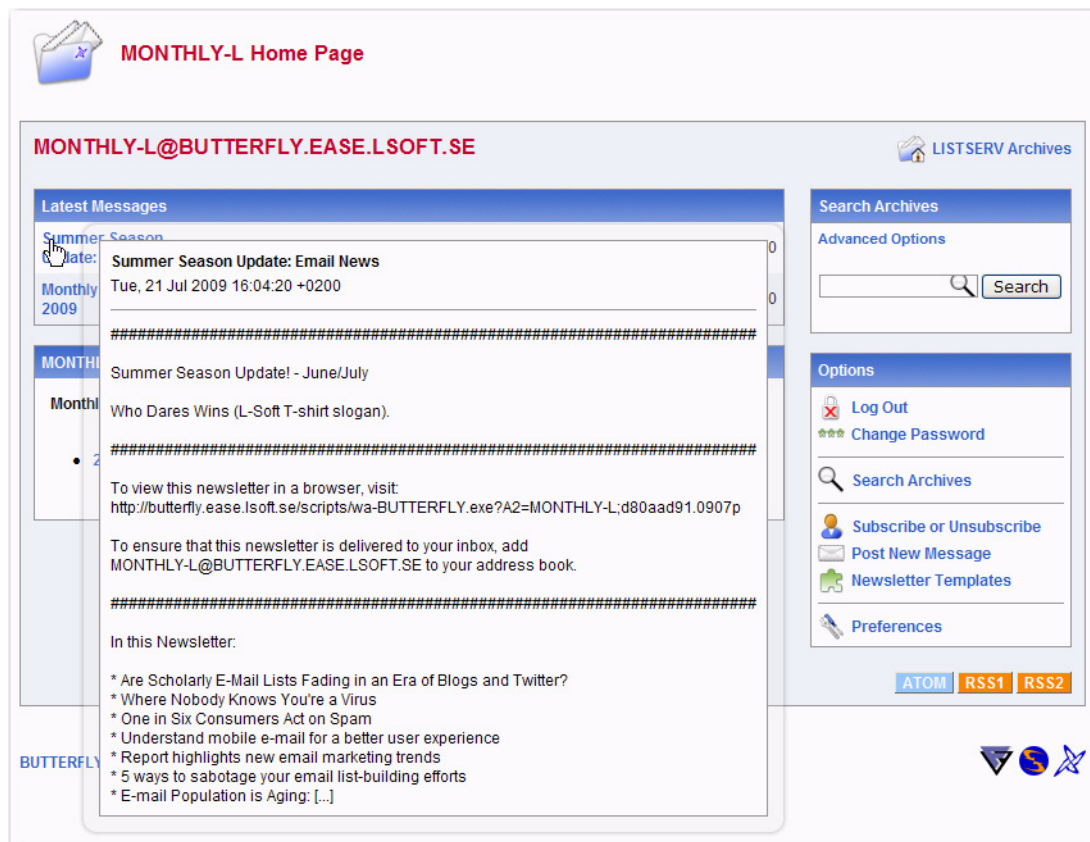
- **Get Password/Change Password** – Click to get a LISTSERV password or to change an existing password.
- **Subscribe or Unsubscribe** – Click to open the Subscribe or Unsubscribe page. For more information, see Section 10.5 [Subscribing, Unsubscribing, or Updating a List from the Archives](#).
- **Post New Message** – Click to compose and post a message to the list. For more information, see Section 11 [Posting Messages to a List from the Web Interface](#).
- **Search Archives** – Click to open the Archive Search page.
- **Preferences** – Click to open the Preferences page.
-  • **Newsletter Templates** – Click to create and send a template using the template library. For more information, see Section 11.1 [Creating Customized HTML Newsletters](#).



In addition, if the option to view the latest messages on the List Archives page is enabled, then a **Latest Messages** table is shown at the top of the screen. If you have hovering descriptions enabled as well, then you can hover your mouse over the subject line and preview the message. For details on enabling these options, see Section 5.6.2 [Archive Preferences](#).

To access a specific archive, simply click on it. You are taken to the Archive Index. See the next section for information.

Figure 10-4 Previewing Latest Messages on the List Archives Screen



10.3 The Archive Index

The Archive Index screen displays the message archives for a list.



If an RSS abstract has been defined for the message, you can view a quick snapshot of the message by hovering your mouse over the subject. A popup of the message will appear.



Note: This setting can be turned on/off on the Archive Preference tab. For details, see Section 5.6.2 [Archive Preferences](#). In addition, this feature requires javascript.

This page offers you several options:

- **Show Author** – Click to show who the message is from (the **From** column will be displayed in the table). If this information is already displayed, then this option is not enabled.
- **Hide Author** – Click to hide who the message is from (the **From** column will not be displayed in the table). If this information is already hidden, then this option is not enabled.
- **Show Table of Contents** – Click to show the table of contents. If this information is already displayed, then this option is not enabled.
- **Hide Table of Contents** – Click to hide the table of contents. If this information is already hidden, then this option is not enabled.
- **Subscribe or Unsubscribe** – Click to open the Subscribe or Unsubscribe page. For more information, see Section 10.5 [Subscribing, Unsubscribing, or Updating a List from the Archives](#).
- **Post New Message** – Click to compose and post a message to the list. For more information, see Section 11 [Posting Messages to a List from the Web Interface](#).
- **Search Archives** – Click to open the Archive Search page.
- **Preferences** – Click to open the Preferences page.
- **Newsletter Templates** – Click to create and send a template using the template librarys. For more information, see Section 11.1 [Creating Customized HTML Newsletters](#).



To sort the table, you can click on one of the column headers (**Subject**, **From**, or **Date**). For example, if you click on **Date** once, then the messages will be sorted in chronological order – the oldest message is listed first and the most recent message is listed last. If you click on **Date** again, then the messages will be reversed chronologically – the most recent message is listed first and the oldest message is listed last. The small white triangle will let you know which column header the table is sorted by and in what order.



Notes: The sort order of the table also determines the Table of Contents. If you sort by **Subject**, then the Table of Contents will be sorted by subject. However, when the table is sorted using the **From** column, the Table of Contents changes to a List of Contributors that displays who the message is from.

The default sorting for the table is set using the **Primary Sorting** and **Secondary Sorting** options on the Archive Preferences tab under **Preferences**. For more information, see Section 5.6 [Setting Your Personal Preferences](#).



If you are a server administrator or an owner of the list, then a **Recycling Bin** icon is displayed next to each message. Clicking on this icon allows you to delete the entire message from the archives.



Before the message is deleted, a confirmation page is displayed, giving you a chance to review or cancel your selection. Once deleted, the message is permanently erased and cannot be recovered.

To view a specific message, click on its subject in the Table of Contents (if visible) or in the table. The message is displayed in the Archive Browsing screen. See the next section for more information.

Figure 10-5 Archives Index

LISTSERV-AT-WORK Archives

2006

View: [Show Author](#) | [Hide Author](#)
[Show Table of Contents](#) | [Hide Table of Contents](#)

Subject	From	Date	Size	Delete
LISTSERV at Work - Issue 1, 2006	L-Soft Newsletter <editor@LSOFT.COM>	Thu, 20 Apr 2006 10:00:00 -0500	641 lines	
LISTSERV at Work - Issue 2, 2006 (Anniversary Issue)	L-Soft Newsletter <editor@LSOFT.COM>	Thu, 15 Jun 2006 10:00:00 -0500	660 lines	
LISTSERV at Work - Issue 3, 2006	L-Soft Newsletter <editor@LSOFT.COM>	Tue, 24 Oct 2006 10:00:00 -0500	614 lines	

[Delete All](#)

Search Archives
 Advanced Options

Options
[Log Out](#)
[Change Password](#)
[Search Archives](#)
[Subscribe or Unsubscribe](#)
[Post New Message](#)
[Newsletter Templates](#)
[Preferences](#)

Archives
 2006
 2005

[ATOM](#) [RSS1](#) [RSS2](#)

10.4 Viewing an Archived Message

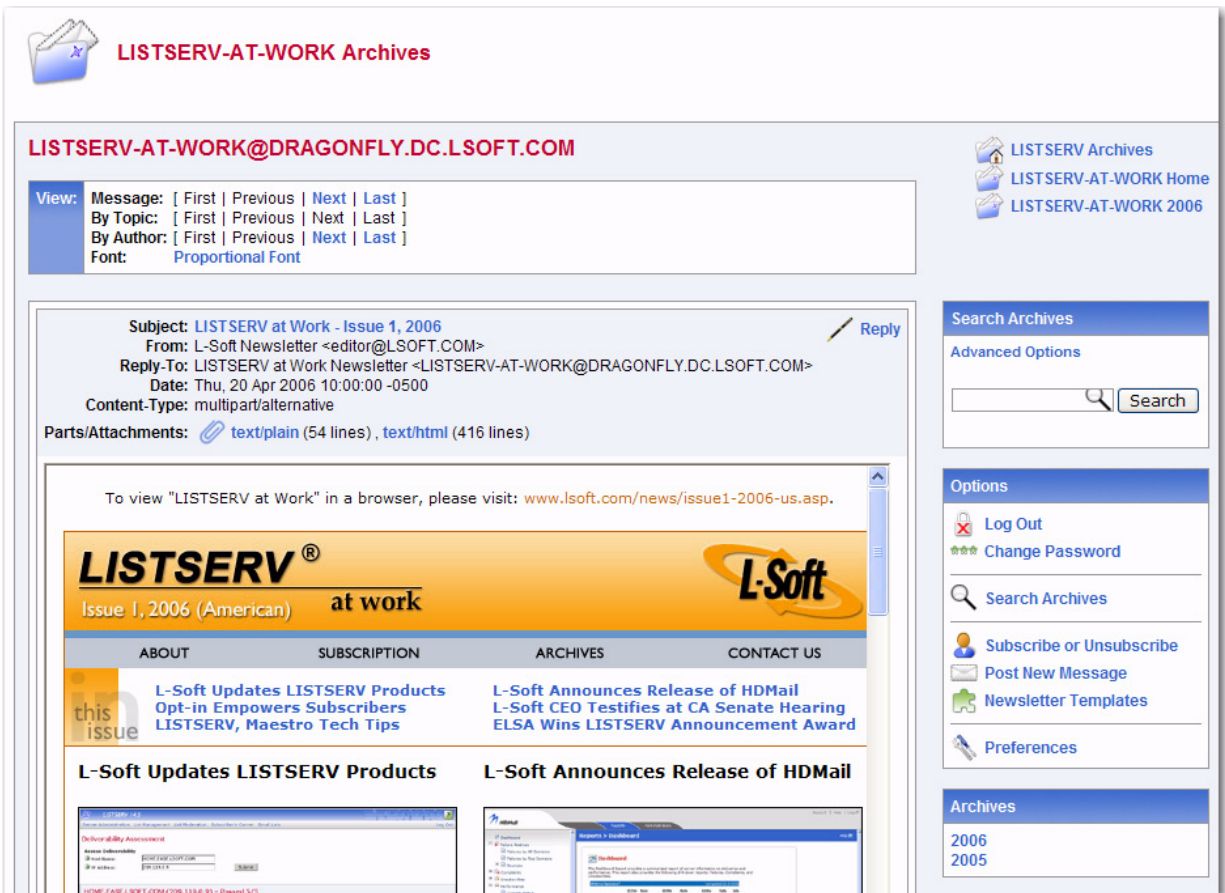
The Archive Browsing screen displays the message viewer, letting you view all of the messages in the current archive.

You can view your messages using several methods:

- **Next Message** – Click to view the next message.
- **Previous Message** – Click to view the previous message.
- **Next in Topic** – Click to view the next message in the topic.
- **Previous in Topic** – Click to view the previous message in the topic.

- **Next by Author** – Click to view the next message by the same author.
- **Previous by Author** – Click to view the previous message by the same author.
- **Proportional Font** – Click to view the message using proportional font.
- **Monospaced Font** – Click to view the message using monospace font.

Figure 10-6 The Archive Browsing Screen



This page also offers you several other options:

- **Subscribe or Unsubscribe** – Click to open the Subscribe or Unsubscribe page. For more information, see Section 10.5 [Subscribing, Unsubscribing, or Updating a List from the Archives](#).
- **Reply** – Click to compose and post a reply to the current message.
- **Post New Message** – Click to compose and post a message to the list. For more information, see Section 11 [Posting Messages to a List from the Web Interface](#).
- **Search Archives** – Click to open the Archive Search page.
- **Preferences** – Click to open the Preferences page.
- **Newsletter Templates** – Click to create and send a template using the template librarys. For more information, see Section 11.1 [Creating Customized HTML Newsletters](#).

New



Note: If the message you are viewing contains additional parts or attachments, then the **Parts/Attachments** row will be displayed after the email header. To view the part or attachment, simply click on it. For multi-part messages, the default view (**HTML** or **Text**) is set using the **Default View** option on the Archive Preferences tab under **Preferences**. For more information, see Section 5.6 [Setting Your Personal Preferences](#).

10.5 Subscribing, Unsubscribing, or Updating a List from the Archives

The List Archives screen, the Archive Index screen, and the Archive Browsing screen contain the option for subscribing to or unsubscribing from a list. This **Subscribe or Unsubscribe** option brings you to the Subscribe or Unsubscribe screen, which lets you subscribe or unsubscribe to a mailing list and to change your subscription settings.

To unsubscribe from a list, simply click the **[Unsubscribe to List]** button.

To update or subscribe to a list, select the settings you want to use for your subscription and then click either **[Update Options]** or **[Subscribe to List]**. See the information below for details on the available settings.

Figure 10-7 Joining, Leaving, or Updating a List from the Archives

Subscribe or Unsubscribe to the LISTSERV-AT-WORK List

LISTSERV-AT-WORK@DRAGONFLY.DC.LSOFT.COM

You are currently **subscribed** to the LISTSERV-AT-WORK list. From this screen, you can update your subscription options or unsubscribe.

List Name: [Switch to Another List](#)

Name:

Email Address:

Subscribed Since: 30 Jan 2009

Subscription Type

- ☒ Regular [NODIGEST]
- ☐ Digest (traditional) [NOMIME DIGEST]
- ☐ Digest (MIME format) [NOHTML MIME DIGEST]
- ☐ Digest (HTML format) [HTML DIGEST]
- ☐ Index (traditional) [NOHTML INDEX]
- ☐ Index (HTML format) [HTML INDEX]

Mail Header Style

- ☒ Normal LISTSERV-style header [FULLHDR]
- ☐ LISTSERV-style, with list name in subject [SUBJECTHDR]
- ☐ "Dual" (second header in mail body) [DUALHDR]
- ☐ sendmail-style [IETFHDR]
- ☐ Normal LISTSERV-style (RFC 822 Compliant) [FULL822]

Acknowledgements

- ☐ No acknowledgements [NOACK NOREPRO]
- ☒ Short message confirming receipt [ACK NOREPRO]
- ☐ Receive copy of own postings [NOACK REPRO]

Miscellaneous

- ☐ Mail delivery disabled temporarily [NOMAIL]
- ☐ Address concealed from REVIEW listing [CONCEAL]
- ☐ User may not post to list [NOPOST]

[Update Options](#) [Unsubscribe to LISTSERV-AT-WORK](#)

The following settings are available in the **Subscription Type** section:

- **Regular** – With a "regular" subscription, you receive individual postings immediately as they are processed by LISTSERV.
- **Digest (Traditional), Digest (MIME format), and Digest (HTML format)** – With a "digest" subscription, you receive larger messages (called "digests") at regular intervals, usually once per day or once per week. These "digests" are collections of individual list postings. Some lists are so active that they produce several digests per day.

Digests are a good compromise between reading everything as it is posted and feeling like the list is clogging your mailbox with a multitude of individual postings.

There are three digest formats: a "traditional", text-only format; a MIME format, which (with mail clients that understand MIME digests) "bursts" the individual messages out of the digest so that you can read them separately; and an HTML format, which requires an HTML mail clients.

- **Index (Traditional) or Index (HTML format)** – With an "index" subscription, you receive short "index" messages at regular intervals, usually once per day or once per week. These "indexes" show you what is being discussed on the list, without including the text of the individual postings. For each posting, the date, the author's name and address, the subject of the message, and the number of lines is listed. You can then download messages of interest from the server (the index contains instructions on how to do that).

An index subscription is ideal if you have a slow connection and only read a few hand-picked messages. The indexes are very short and you do not have to worry about long download times. The drawback of course is that you need to reconnect to retrieve messages of interest from the server.

You can choose to have the index sent to you in either a traditional format (plain text) or in HTML format with hyperlinks.

The following settings are available in the **Mail Header Style** section:

- **Normal LISTSERV-style header** – "Full" mail headers (normally the default), containing Internet routing information, MIME headers, and so forth. The ('To:') header contains the address of the list.
- **LISTSERV-style, with list name in subject** – "Full" mail headers (like the default) except that a "subject tag" is added to the subject line of mail coming from the list. If there is no subject tag defined in the list's configuration, the name of the list will be used. This can be very useful for sorting and filtering mail.
- **"Dual" (second header in mail body)** – Dual headers are regular short headers followed by a second header inside the message body. This second header shows what list the message is coming from ('Sender:'), the name and address of the person who posted it ('Poster:'), the poster's organization, if present, and the message subject. Dual headers are helpful if your mail client does not preserve the original return email address.
- **sendmail-style (advanced option)** – This option selects sendmail-style headers, i.e. an exact copy of the original, incoming mail header with the addition of a

('Received:') line and a ('Sender:') field. Some technical people prefer this type of header.

- **Normal LISTSERV-style (RFC 822 Compliant)** – (For Advanced Use Only) "Full" mail headers (like the default) except that the ('To:') header contains the recipient's email address instead of the list address.

The following settings are available in the **Acknowledgements** section:

- **No acknowledgements** – LISTSERV will not send any acknowledgement at all when you post to the list. This is probably not a good setting unless you really do not want any feedback from LISTSERV as to whether or not your posting was received and distributed.
- **Short message confirming receipt** – Typically, this is the default setting, although it can be overridden by the list owner. If you choose this setting, LISTSERV will send you a short message whenever you post to the list, confirming the distribution of your message and telling you how many people it was sent.
- **Receive copy of own postings** – Some people prefer this setting over the short acknowledgement message. It tells LISTSERV to send you a copy of your own postings so that you can see exactly how it appeared on the list (useful if you are behind an unreliable gateway or firewall).

The following settings are available in the **Miscellaneous** section:

- **Mail delivery disabled temporarily** – This option toggles the receipt of mail from the list. You may want to disable mail delivery if you will be away from your mail for an extended period of time.
- **Address concealed from REVIEW listing** – This option conceals you from the list of subscribers shown by the REVIEW command. Note that the list owner and the LISTSERV administrator can always get the complete list of subscribers, regardless of this setting. Nowadays, most lists are configured so that only the list owner can use the REVIEW command, but some lists still allow subscribers to get a listing of all the other participants.

For lists with topics enabled, then the **Topics** section is visible (if the topics section is not visible, topics are not enabled). This section allows you to subscribe or unsubscribe to all, some, or none of the available topics.

10.6 UTF-8 Archive Searches

LISTSERV 16.0 now has UTF-8 archive searches. The **new UTF-8 archive indexer** is now the default indexer.



Important: All archives will be automatically reindexed the first time LISTSERV 16.0 starts. Please be aware that this process may take awhile.

The new UTF-8 archive indexer works as follows:

- The message is parsed and all plain-text parts are decoded, converted to UTF-8 and indexed. If there is no plain-text part, the first text part (presumably HTML) is indexed.

- This means that you can now search base64 messages.
- Data from the mail header is converted to UTF-8 before being indexed. Only the subject and sender address are indexed, so "date" will no longer be found in every message.
- There is still no high-level concept of character set, but that doesn't matter since everything is converted to UTF-8 before being indexed.
- The SEARCH command knows to convert input parameters to UTF-8 before calling the search engine, when the index is in Unicode. If you search for åäö, you will find it regardless of the charset in which it was typed.

An index must be either entirely in ASCII or entirely in Unicode. When appending to an existing index, the corresponding indexer is chosen regardless of configuration. To switch to Unicode, you have to re-index. There is no way around that - your old ASCII index doesn't have the entries for base64 messages and so on.

When creating a new index, the ASCII format is automatically chosen.

Unicode indexes have a *FLAGS* line in the DBNAMES file. This is the only way to tell them apart because the index format hasn't had significant changes. In addition, do not be surprised if the index is smaller than before. For many lists, this will be the case. Lastly, take note that searches may be faster than they were before these changes.

10.6.0.1 Known Issues

1. The changes made broke the search function with ASCII indexes in the following rare scenario:
 - You are using ISO-8859-1 and making a case-insensitive search containing national characters.
 - You have entered the search string in lower case.
 - You are searching the subject only, not the message itself (meaning that you are requesting a subject-only search, using the subject input box). If you search for the subject text in the main input box, it will work.
 - You are not running HPO.
 - The data you supplied is present but in the wrong case.
2. The new code has the following known problems with Unicode indexes only:
 - Due to having both old and new search engine active in the product and not wanting to duplicate thousands of lines of code, there may be cases where the search engine returns entries that should not have matched. This will only happen with non-ASCII characters.
 - Case matches may not be 100% reliable until the case-support phase of the unicode support feature is complete.

WA may have problems displaying search results correctly on the LCMD page.

Section 11 Posting Messages to a List from the Web Interface

After you have created or subscribed to a list you will eventually want to participate in the discussion, assuming that you have joined a list where subscribers have permission to post (on some lists you are not allowed to post mail back to the other subscribers). For more information on the different types of mailing lists, see Section 4 [Mailing List Types](#).

11.1 Composing the Message

The archives for a list give you the option to post a message from the List Archive screen, the Archive Index, and the Archive Browsing screen. For more information on these screens, see Section 10.2 [Viewing List Archives](#), Section 10.3 [The Archive Index](#), and Section 10.4 [Viewing an Archived Message](#). To access the list archives, click **Email Lists** on the Toolbar. To compose and send a message to the list from the archives, click the **Post New Message** option on any of the archive screens. The Send Message screen opens, which is where you compose your plain text, HTML, or multi-part messages. If the list allows attachments, then you can add those as well.

Figure 11-1 The Send Message Screen - Initial View

Send Message (LISTSERV-AT-WORK)

LISTSERV-AT-WORK@DRAGONFLY.DC.LSOFT.COM

NEW: Newsletter Templates

LISTSERV Archives

LISTSERV-AT-WORK Home

Your Name:

Subject:

Content-Type: Plain Text

Show Advanced Show Tests

Message:

Send copy of message to: htaylor@Lsoft.com ☐

Save Draft Send Message



Tip: LISTSERV comes with ready-to-use newsletter templates that let you create professional-looking HTML newsletters with or without HTML coding experience. You can access these templates by clicking the **Newsletter Templates** option. For details, see Section 11.2 [Creating Customized HTML Newsletters](#).

Fill out each of the data entry fields, and then click the **[Send]** button.

The following fields are available:

- **Your Name** – Enter your name as you wish it to appear in the **From** field of the email header of the message. The email address that will be used in the **From** field is the one that is logged into LISTSERV and may not be changed.
- **Topic** – Select the topic, if any, that you want to apply to your message. This option will only be shown if topics are enabled for the list.
- **Subject** – Enter the subject line for the email message posted to the list.
- **Content-Type** – Use the drop-down menu to select the format of your email message. You may select **Plain Text** or **HTML**.
 - **Plain Text** – To add a plain text message, type directly into the text box or copy and then paste your text into the text box.
 - **HTML** – To create an HTML message, type directly into the text box or copy and then paste your HTML code into the text box. If you have the advanced options displayed, then you can also upload the HTML content (see below).

If you click the **[Show Advanced]** Button, you can also use the Upload Applet. To use this applet, click the **[Upload HTML]** button. The applet will start and you can browse for your HTML file and then automatically upload it along with any inline image files. If your HTML file contains links to remote image files, the applet will check them for errors. After the upload is complete, your message will be displayed as HTML code in the text box. Any inline files will be displayed as attachments.



If you have Javascript enabled, then you can click on the **[HTML Editor Mode]** button to open the HTML Editor. For details on using this editor, see Section 11.1.1 [Using the HTML Editor](#).

- **Add Part** – Click the **[Add Part]** button to compose a multi-part message. Multi-part messages usually have a plain text part and an HTML part. To build a multi-part message, select the content type of the first part of your message. Next, click the **[Add Part]** button to open a second text box. Add the content using one of the methods described above. To delete an added part, click the **[Delete]** button associated with that part.
- **Show Advanced** – Use the **[Show Advanced]** button to display the following advanced options:



- **Schedule Delivery** – Select whether you want the message to be delivered immediately or at a specific time and date. The input format of the **Delivery Date** field is **yyyy-mm-dd**. The input format of the **Delivery Time** field is **hh:mm:ss**, using the 24-hour clock. The time zone used is that of the LISTSERV server. If you enter a delivery date or time that has already passed, then the message will be delivered immediately.



If you select to schedule the delivery for a specific date and time, then after the message is sent, and you re-visit the Message Posting Interface (Send

Message screen), a **Pending Messages** table will be shown at the bottom of the screen. This table displays all messages that have yet to be delivered.

- **Upload HTML** – Use this button to start an applet that will allow you to browse for your HTML file and then automatically upload it along with any inline image files. If your HTML file contains links to remote image files, the applet will check them for errors. After the upload is complete, your message will be displayed as HTML code in the text box. Any inline files will be displayed as attachments.
- **Charset** – Use this drop-down menu to select the charset (character set) of the message. If your message requires a charset that is different from the default charset for LISTSERV (UTF-8), then you can use the drop-down menu to select the matching one before you type or paste in your message. This selection will not translate one charset to another. It is used to sync the charset of the original message with LISTSERV so that it can be encoded correctly for sending.



- **RSS Abstract** – Many lists allow subscribers to read postings through RSS feeds. LISTSERV automatically creates RSS abstract from the text part of the message. By default, LISTSERV uses a maximum of 500 and a minimum of 250 words for an implicit RSS abstract. These default numbers can be changed either for the entire site or on a list-by-list basis. However, this screen also allows you to create explicit RSS abstracts, which can be HTML or text of any length. To provide an explicit RSS abstract, simply enter text here in the **RSS Abstract** text box. To use your text or HTML part in its entirety for the RSS abstract, click the **Use as RSS Abstract** icon next to the that particular section; the content will be automatically copied into the abstract text area.



- **Attachments** – Use the **[Browse...]** button to find your file on a local drive. Click the **[Attach]** button to attach your file to your outgoing message. The attachment will be listed in a table beneath the **Attachments** field. To delete an attachment, click the **[Delete]** button associated with that item in the table.



Notes: Each list can have its own rules for accepting or rejecting messages with attachments. The **[Browse...]** and **[Attach]** buttons will function regardless of individual list settings. As a result, your message may be rejected or stripped of its attachment by the list. Be sure to check with the list owner if you have any questions.

Figure 11-2 The Send Message Screen - Advanced Options View with Pending Messages

Send Message (HAT-TEST)

HAT-TEST@BUTTERFLY.EASE.LSOFT.SE

Your Name:

Subject:

Content-Type:

Schedule Delivery: ☒ Deliver Immediately ☐ Deliver at Following Time

Delivery Date:

Delivery Time:

Charset:

Attachments:

RSS Abstract:

Tests:

Message:

Send copy of message to: htaylor@lsoft.com ☐

Pending Messages	From	Posted Date	Scheduled Date
Test	Holly Taylor <htaylor@LSOFT.COM>	Tue, 1 Dec 2009 16:20:49 +0100	Wed, 2 Dec 2009 18:00:00 +0100
Weekly Updates	Holly Taylor <htaylor@LSOFT.COM>	Tue, 1 Dec 2009 16:27:00 +0100	Wed, 9 Dec 2009 14:01:00 +0100



- **Show Tests** – Use the **[Show Tests]** button to display the following test options:
 - **Content Analysis** – If your LISTSERV license includes technical support, then you can use this button to check your message and evaluate the likelihood that it will be deemed to be spam by recipient mail systems. LISTSERV will connect to a hosted service at L-Soft, which runs the email message through SpamAssassin. The results will be shown on the screen. If necessary, you can then make adjustments and run the test again.

- **Send Test to Self** – When constructing an email message and before sending it to your list, you can use this button to send a test message to yourself to make sure that the results are satisfactory. LISTSERV will send a copy of the message to the email address that you used to log in to the Web interface. This kind of test delivery can be particularly useful when creating HTML newsletters or multi-part messages. Email clients vary in their degree of HTML support. Sending a test message allows you to double-check that the message is rendered as you intended. If necessary, you can make adjustments to your message before posting it to the list.



- **Quote Original Message** – When replying to a message that has been previously posted to a list, then the **Quote Original Message** icon is available below the message box. Clicking on this icon will place a quote of the original message in your reply.



- **Save Draft** – If you have started working on an email message or newsletter and want to continue at a later time, you can click the **[Save Draft]** button to save a draft of the message. The subject line, the text part, and the HTML part (if applicable) are all stored. However, advanced options, such as attachments and RSS abstracts are not stored with the draft. The drafts, which are identified by their subject lines and shown with a timestamp, will be listed in a table on the right side of the screen. When you return to the Send Message screen, you can just click on the subject line to retrieve the draft. As a safety measure, LISTSERV automatically saves and updates drafts every time the message posting interface is refreshed. To delete existing drafts, click on the **Recycling Bin** icon next to the draft that you want to erase. Existing drafts are deleted automatically when they are posted to the list.



Note: Drafts are automatically deleted after 30 days.

Figure 11-3 Drafts Table on the Send Message Screen

Send Message (LISTSERV-AT-WORK)

LISTSERV-AT-WORK@DRAGONFLY.DC.LSOFT.COM

Your Name:

Subject:

Content-Type: Plain Text

Message:

☐ Send copy of message to: htaylor@Lsoft.com

Message Drafts

Announcement	2009-03-20 09:44:59	
Newsletter	2009-03-20 09:45:51	

[NEW: Newsletter Templates](#)
[LISTSERV Archives](#)
[LISTSERV-AT-WORK Home](#)

11.1.1 Using the HTML Editor



If you've selected **HTML** from the **Content Type** drop-down menu and you have JavaScript enabled, then the **[HTML Editor Mode]** button will be available on the Send Message screen. This button opens a WYSIWYG-style editor in place of the HTML source code text box. If you have already entered HTML code into the text box before you launch the editor, the editor will parse and rewrite the existing HTML code, which may introduce changes to the code. To return to the HTML source code text box, click the **[Source Code Mode]** button.



To preview your HTML newsletter, click on the **Preview** icon on the HTML Editor toolbar. The message will open in a new window, letting you preview the finished product.

11.2 Creating Customized HTML Newsletters



You can now create professional-looking HTML newsletters with or without HTML coding experience. LISTSERV comes with ready-to-use newsletter templates that contain placeholders that, when selected in the template designer, will let you fill in predefined areas with your own text and/or images.

To access the Newsletter Template wizard, go to the List Archives screen, the Archives Index screen, or the Archive Browsing screen for the list you want to post the newsletter to, and then click on the **Newsletter Template** option.

11.2.1 Selecting a Template

To select a newsletter template, click the **[Select]** button associated with the template you want to use as your content. The Select Template screen will refresh with the selected template displayed in the **Template Selected** box.



To edit a template, click the **Easel** icon. The Edit Newsletter Template screen opens, which is where you can make any changes necessary.



Notes: If you decide to edit an existing, default newsletter template on this screen, then you will no longer be able to edit colors on the Content Definition screen. Any color modifications should instead be done on this screen.

You can also edit the default templates using the Web Template Editor. For details, see Section 9.2 [Using the HTML Mail Template Gallery](#).



If you've made changes to a default template, and you'd like to undo those changes, click the **Revert** icon associated with that template.

To create a new template based on a default template, simply edit that template.

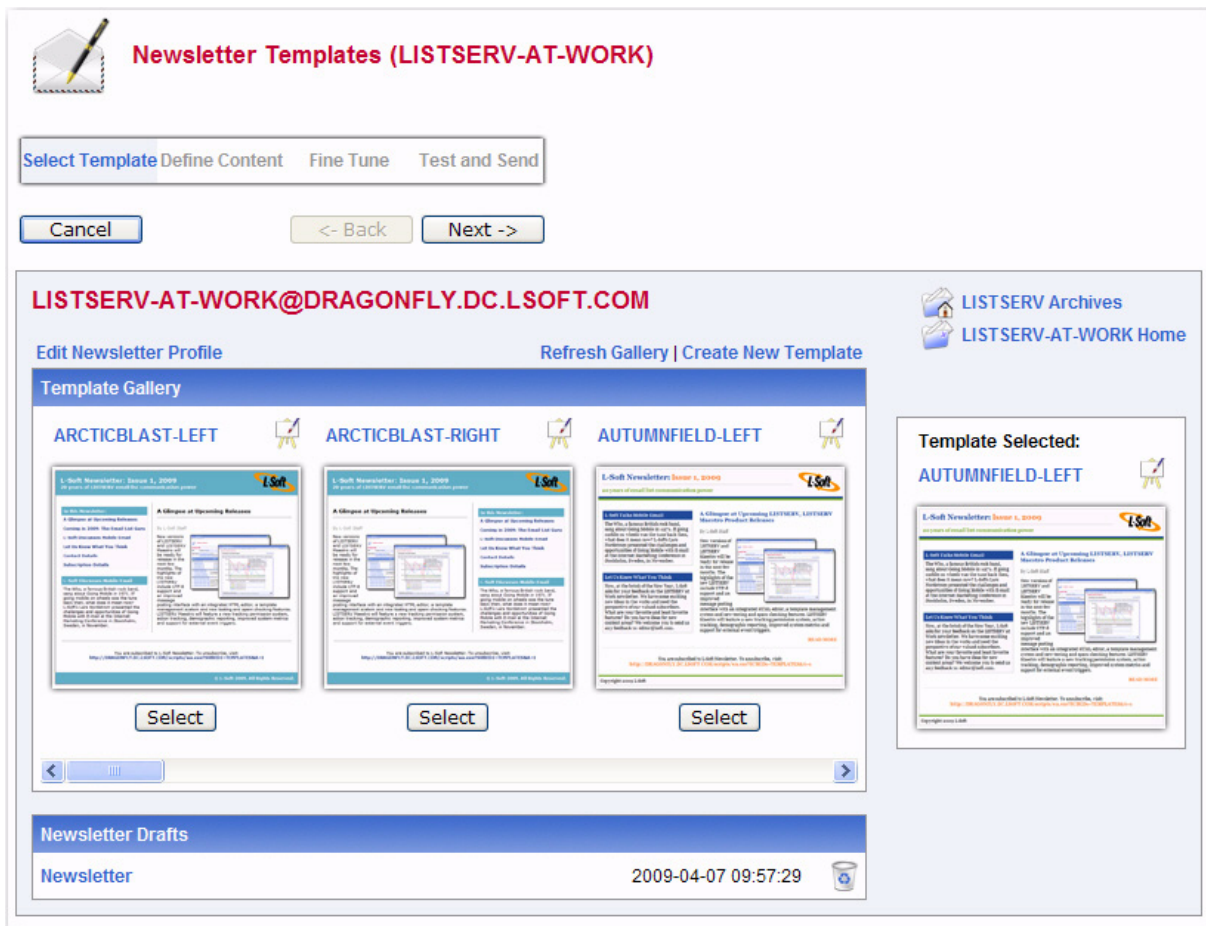
To create an all new template, click on the **Create New Template** option. The Create New Template screen opens, which is where you enter the content for the template.



To delete template, click on the **Recycling Bin** icon associated with that template.

To refresh the newsletter template gallery, click the **Refresh Gallery** option.

Figure 11-4 Selecting a Newsletter Template



To edit the profile for all of the newsletters, click on the **Edit Newsletter Profile** option. The Edit Newsletter Profile screen opens, which is where you can enter any changes you'd like to make to your newsletter profile. These changes will be applied to the entire newsletter template gallery and will be associated with your user account only.



Note: The newsletter's profile fields can also be set in your preferences. For details, see Section 5.6.4 [Newsletter Profiles](#).



The **Newsletter Drafts** section at the bottom of the screen contains any previously saved drafts. To continue working on an existing draft, click on the draft's subject. To delete the draft, click on the **Recycling Bin** icon associated with that draft.



Note: Drafts are automatically deleted after 30 days.

Once you've selected the newsletter template you want to work with, click the **[Next]** button to continue.

To cancel the selection of the newsletter template and refresh the screen, click the **[Cancel]** button.

11.2.2 Defining the Content

The Define Content screen is where you "fill in the blanks". The templates were designed to make defining newsletters quick and easy; the majority of the options, settings, and

coding are already in place. The only work left for you is to give the newsletter a customized feel by filling in special template placeholders.

Figure 11-5 Defining the Newsletter Content

The screenshot shows the 'Newsletter Templates (LISTSERV-AT-WORK)' interface. At the top, there are tabs for 'Select Template', 'Define Content' (which is active), 'Fine Tune', and 'Test and Send'. Below these are 'Cancel', '<- Back', and 'Next ->' buttons. The main header displays the email address 'LISTSERV-AT-WORK@DRAGONFLY.DC.LSOFT.COM' and links to 'LISTSERV Archives' and 'LISTSERV-AT-WORK Home'. The form includes fields for 'Your Name' (Joe Smith), 'Subject' (Holiday Sale), 'Template' (WATERFALL: Default HTML Mail Template 'Waterfall'), 'News Items' (2), and 'Feature Items' (2), with an 'Update' button. A 'Placeholders:' list on the left includes items like WEBSITE, LOGO, ISSUE, NEWSLETTER, HEADLINE1, BYLINE1, IMAGE1, TEXT1, URL1, HEADLINE2, BYLINE2, IMAGE2, TEXT2, URL2, FOOTER, FEATUREHEAD1, FEATURETEXT1, FEATUREHEAD2, FEATURETEXT2, ADDRESS, PHONE, CONTACT, and LEGAL. The 'Preview:' window shows a sample newsletter layout with placeholders like &*NEWSLETTER;, &*HEADLINE1;, &*FEATUREHEAD1;, and &*FEATURETEXT1;. It also includes a 'Contents:' section with a prompt to click on a placeholder to the left. At the bottom right of the preview are 'Revert' and 'Save' buttons. On the far right, a 'Newsletter Drafts' table lists drafts with dates and times.

The following fields are available at the top of the screen:

- **Your Name** – This field is automatically filled with the email address associated with your user account. To change this, simply delete the email address and enter your name as you wish it to appear in the **From** field of the email header of the newsletter. The email address that will be used in the **From** field is the one that is logged into LISTSERV and may not be changed.
- **Subject** – The subject line for the newsletter.
- **Template** – If you want to change the template you are currently using, click this drop-down menu and select a new template from the drop-down list.
- **News Items** – Enter the number of news items you want to have in your newsletter. This will determine the amount of news item placeholders you will need to define. The maximum number allowed is 99.

- **Feature Items** – If available, enter the number of feature items you want to have in your newsletter. This will determine the amount of feature item placeholders you will need to define. The maximum number allowed is 99.

To view the template in HTML format, click **HTML Part**. To view the text alternative, click **Text Alternative**.

In the placeholders list, any placeholder without a value supplied is marked with a red x; these placeholders will require your attention. Placeholders where a value has already been supplied are marked with a green checkmark.



Note: If you leave a placeholder blank, then this content will not appear in the newsletter; however, there are some exceptions such as headers and text since they are expected to be defined. If you want to remove a header for a news item or feature item, simply adjust the value for those items in the appropriate field above.

The type and amount of placeholders available for definition will vary depending on the type of template and the number of news and/or feature items you have entered. The following placeholders are always available for definition and can also be defined in your newsletter profile:

&*NEWSLETTER; – This placeholder is used for the name of your newsletter.

&*ADDRESS; – This placeholder is used for the address of your organization.

&*PHONE; – This placeholder is used for the phone number of your organization.

&*CONTACT; – This placeholder is used for the contact email address of your organization.

&*WEBSITE; – This placeholder is used for a Website URL of your organization.

&*SLOGAN; – This placeholder is used for a slogan for your organization.

&*LOGO; – This placeholder is used for a logo of your organization.

&*FOOTER; – This placeholder is used for a short footer for your newsletter.

&*LEGAL; – This placeholder is used for any legal disclaimers, company registration numbers, or similar information for your newsletter.

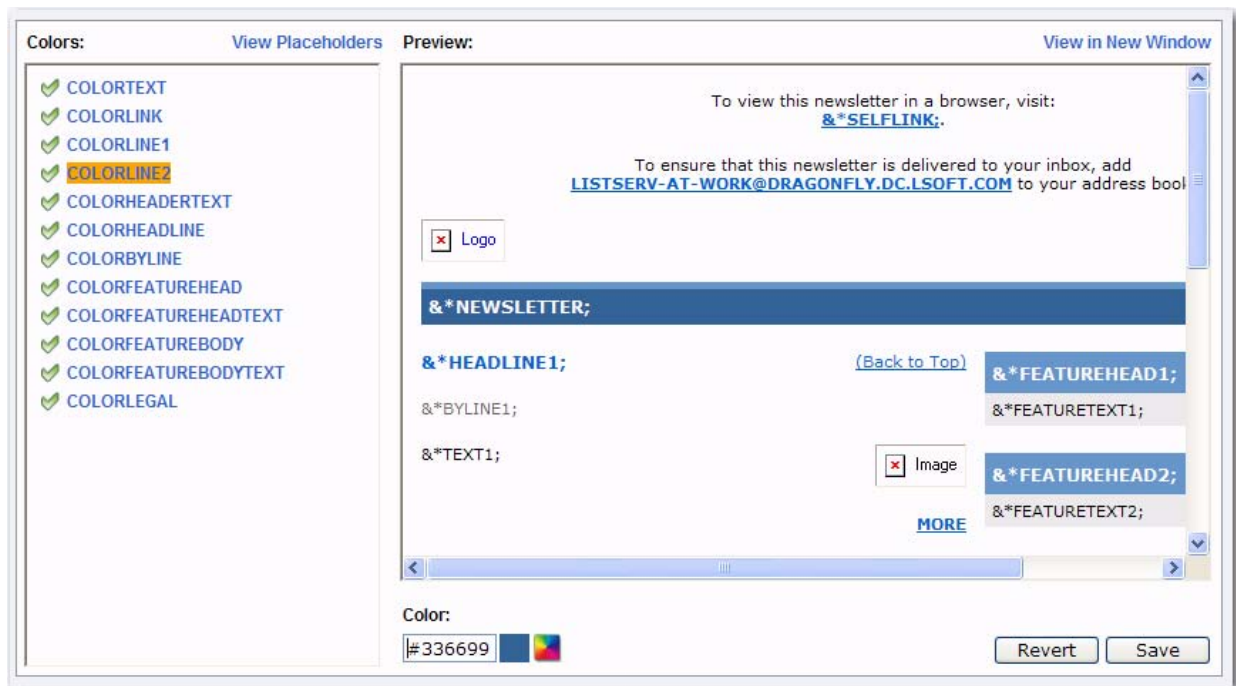
The **&*SELFLINK;** placeholder will be automatically replaced with a link to view the newsletter in a browser. This placeholder is only available for lists with archives.

The placeholder that is currently selected in the placeholder list is highlighted both in the placeholder list and in the template preview pane. You can also move the mouse over the preview pane to see and select placeholders.

To define the selected placeholder, enter its value in the **Contents** text box below the template preview pane. Click **[Update]** to fill the placeholder.

To re-define the colors used for the placeholders, click on the **View Colors** options. The placeholders list is replaced with the placeholder colors list. Select the placeholder color you want to re-define, and then enter the RGB color value in the **Color** field. To view a color palette to help you select the color, click on the **Color Picker** icon, and then click on the color you want to use. The color for that particular placeholder will be updated.

Figure 11-6 Re-Defining Placeholder Colors



To return to the placeholders list, click **View Placeholders**.

To preview the newsletter, click the **View in New Window** option.

Once you are done filling out the placeholders, click **[Save]**. This makes your updates and changes permanent, and places a copy of the newsletter in the **Newsletter Drafts** section.



Note: Drafts are automatically deleted after 30 days.

To undo your changes, click the **[Revert]** button. The template will revert back to the previously saved version of the template (or the default version if no saved version is available).

Once you are finished defining the content of the template, click **[Next]** to continue.

To cancel the definition of the newsletter and return to the Select Template screen, click the **[Cancel]** button. If you have a saved draft version of your newsletter and you click **[Cancel]**, then the saved version will still be available in the **Newsletter Drafts** section.

To return to the Select Template screen without canceling your work, click **[Back]**.

11.2.3 Fine Tuning the Newsletter

For those who are familiar with HTML, LISTSERV gives you the ability to expand your limits and design profession-looking HTML newsletters that go above and beyond the everyday standards.



Note: Any changes you make on this tab will be lost if you use the **[Back]** button.

The following fields are available at the top of the screen:

- **Your Name** – This field is automatically filled with the email address associated with your user account. To change this, simply delete the email address and enter your

name as you wish it to appear in the **From** field of the email header of the newsletter. The email address that will be used in the **From** field is the one that is logged into LISTSERV and may not be changed.

- **Subject** – The subject line for the newsletter.
- **Content-Type** – This drop-down menu is read-only unless the advanced options are activated (see below). If available, you can re-define the format of the newsletter.

To view and define the advanced options, click the **[Show Advanced]** button. The following options are available:

- **Charset** – Use this drop-down menu to select the charset (character set) of the newsletter. If your newsletter requires a charset that is different from the default charset for LISTSERV (UTF-8), then you can use the drop-down menu to select the matching one before you type or paste in your content. This selection will not translate one charset to another. It is used to sync the charset of the original newsletter with LISTSERV so that it can be encoded correctly for sending.
- **RSS Abstract** – Many lists allow subscribers to read postings through RSS feeds. LISTSERV automatically creates RSS abstract from the text part of the message. By default, LISTSERV uses a maximum of 500 and a minimum of 250 words for an implicit RSS abstract. These default numbers can be changed either for the entire site or on a list-by-list basis. However, this option also allows you to create explicit RSS abstracts, which can be HTML or text of any length. To provide an explicit RSS abstract, simply enter text here in the **RSS Abstract** text box. To use your text or HTML part in its entirety for the RSS abstract, click the **Use as RSS Abstract** icon next to either the **Plain Text Content-Type** or **HTML Content-Type** section; the content will be automatically copied into the abstract text area.
- **Attachments** – Use the **[Browse...]** button to find your file on a local drive. Click the **[Attach]** button to attach your file to your outgoing message. The attachment will be listed in a table beneath the **Attachments** field. To delete an attachment, simply click the **[Delete]** button associated with that particular item in this table.



Notes: Each list can have its own rules for accepting or rejecting messages with attachments. The **[Browse...]** and **[Attach]** buttons will function regardless of individual list settings. As a result, your newsletter may be rejected or stripped of its attachment by the list. Be sure to check with the list owner if you have any questions.

Figure 11-7 Fine Tuning the Newsletter - Advanced Options

In the **Plain Text Content-Type** section, you can edit the plain text for the newsletter. Any changes you make in this section are not automatically made in the HTML section. Therefore, if you want these changes in the HTML section as well, you will need to manually enter them.

In the **HTML Content-Type** section, you can edit the HTML part of the newsletter. Any changes you make in this section are not automatically made in the Plain Text section. Therefore, if you want these changes in the Plain Text section as well, you will need to manually enter them.

If you have JavaScript enabled, then the **[HTML Editor Mode]** button will be available. This button opens a WYSIWYG-style editor in place of the HTML source code text box.



Note: If you have already entered HTML code into the text box before you launch the editor, the editor will parse and rewrite the existing HTML code, which may introduce changes to the code. To return to the HTML source code text box, click the **[Source Code Mode]** button.



To preview your HTML newsletter, click on the **Preview** icon on the HTML Editor toolbar. The message will open in a new window, letting you preview the finished product.

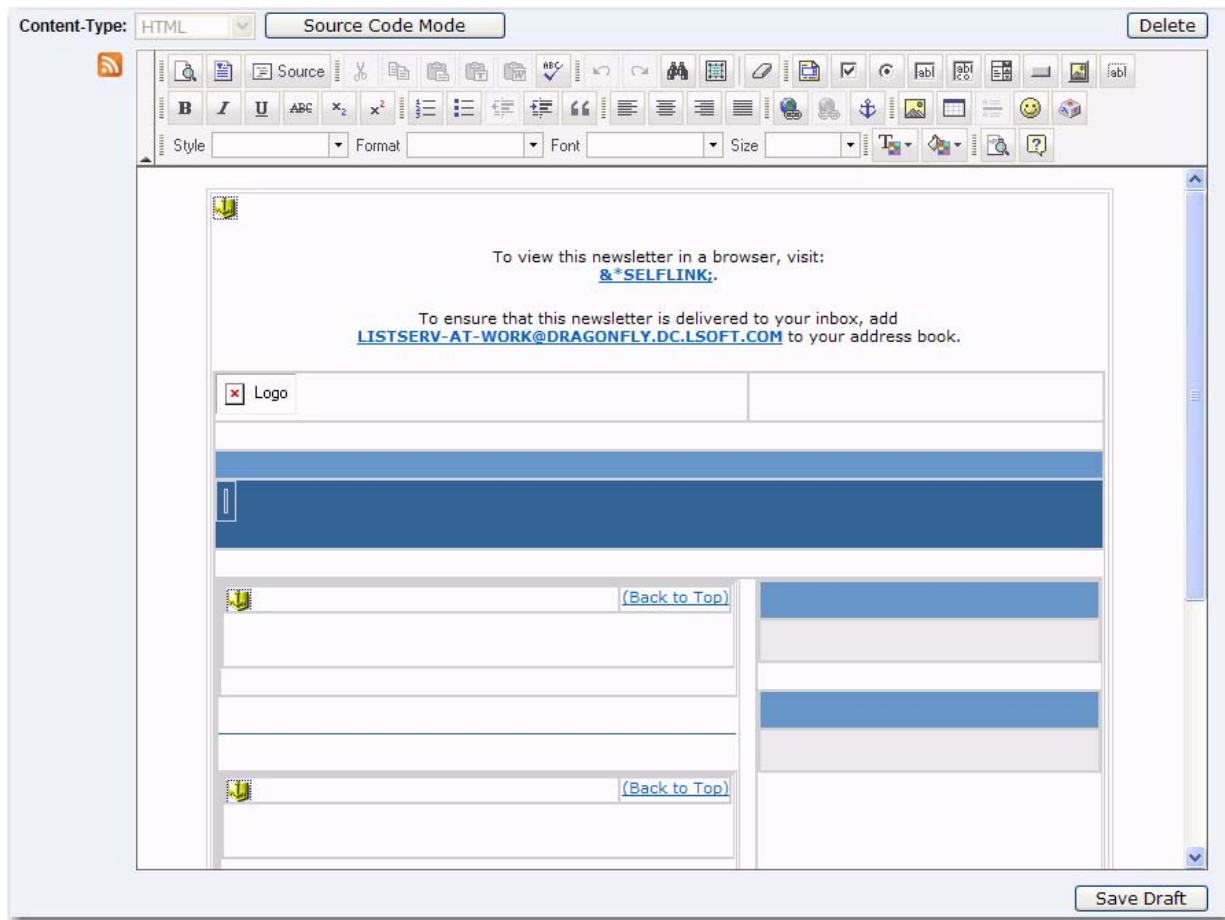
If you have started working on a newsletter and want to continue at a later time, you can click the **[Save Draft]** button to save a draft of the message. The subject line, the text part, and the HTML part are all stored. The drafts, which are identified by their subject lines and shown with a timestamp, will be listed in the **Newsletter Drafts** section. When you return to the Newsletter Template wizard, you can just click on the subject line to retrieve the draft. To delete existing drafts, just click on the **Recycling Bin** icon next to the draft that you want to delete. Existing drafts are deleted automatically when you have posted them to the list.



Important: The draft you save on the Fine Tune tab will not be the same draft you may have saved on a previous tab. Therefore, you may want to change the subject to reflect the version you are saving.

Note: Drafts are automatically deleted after 30 days.

Figure 11-8 Fine Tuning the Newsletter - HTML Editor



11.2.4 Testing and Sending the Newsletter

When constructing a newsletter and before sending it to a list, you can send a test message to make sure that the results are satisfactory. This kind of test delivery can be particularly useful when creating HTML newsletters because email clients vary in their degree of HTML support. Sending a test message allows you to double-check that the newsletter is rendered as you intended. If necessary, you can make adjustments to your newsletter before posting it to the list.

The following options are available on this screen:

- **Your Name** – This field is automatically filled with the email address associated with your user account. To change this, simply delete the email address and enter your name as you wish it to appear in the **From** field of the email header of the newsletter. The email address that will be used in the **From** field is the one that is logged into LISTSERV and may not be changed.
- **Subject** – The subject line for the newsletter.
- **Schedule Delivery** – Select whether you want the newsletter to be delivered immediately or at a specific time and date. The input format of the **Delivery Date** field is **yyyy-mm-dd**. The input format of the **Delivery Time** field is **hh:mm:ss**, using the 24-hour clock. The time zone used is that of the LISTSERV server. If you

enter a delivery date or time that has already passed, then the newsletter will be delivered immediately.

- **Test Addresses** – Enter the email addresses of those who will be receiving the test. Make sure to include your email address so you can view the newsletter as well. If you had entered delivery test email addresses in your Newsletter Profile, then these addresses will appear here.
- **Content Analysis** – If your LISTSERV license includes technical support, then you can use this button to check your message and evaluate the likelihood that it will be deemed to be spam by recipient mail systems. LISTSERV will connect to a hosted service at L-Soft, which runs the email message through SpamAssassin. The results will be shown on the screen. If necessary, you can then make adjustments and run the test again.
- **Send Tests** – Use this button to send a test message to the test addresses.
- **Back** – Click this button if you want to navigate back and make changes.
- **Send** – Click this button to send the newsletter once you are done testing it and are satisfied with its content and appearance.



Note: When you test a newsletter that has `&*SELFINK;`, then this link will not work in the test message because the message has yet to be archived.

Figure 11-9 Testing and Sending the Newsletter

The screenshot shows the 'Newsletter Templates (LISTSERV-AT-WORK)' web interface. At the top, there are four tabs: 'Select Template', 'Define Content', 'Fine Tune', and 'Test and Send'. Below the tabs are buttons for 'Cancel', '<- Back', and 'Send ->'. The main content area has a header 'LISTSERV-AT-WORK@DRAGONFLY.DC.LSOFT.COM' and links for 'LISTSERV Archives' and 'LISTSERV-AT-WORK Home'. The form includes fields for 'Your Name' (Joe Smith) and 'Subject' (Semi-Annual Sale). Under 'Schedule Delivery', there are radio buttons for 'Deliver Immediately' (selected) and 'Deliver at Following Time'. There are also fields for 'Delivery Date' (yyyy-mm-dd) and 'Delivery Time' (hh:mm:ss). A 'Test Addresses' text area contains 'sample@test.com'. At the bottom, there are buttons for 'Content Analysis' and 'Send Tests'. On the right side, there is a 'Newsletter Drafts' section showing a draft titled 'Semi-Annual Sale' dated '2009-10-09 09:49:35' with a trash icon.

Section 12 Starting a Mailing List Using Email

This section (and much of the following sections) assumes that you are administering your list by email, which is the "traditional" method of LISTSERV list administration. LISTSERV also includes a web-based administration interface for lists (described in Section 5 [Introduction to the Web Interface](#)). Lists that are coded "Validate= Yes,Confirm,NoPW" or "Validate= All,Confirm,NoPW" must be managed by mail, since the Web Administration Interface is secured by passwords and these settings reject password validation, instead requiring validation by the "OK" confirmation method. If you use the web-based tools to manage your list, you should still skim this section and refer to it for more advanced list management information.

12.1 Avoiding Duplication¹

Before you start your list, it pays to do a careful search in several places to find out if you are duplicating an already-existing list, or if the name you are considering is already in use for a list on a differing subject. The first place to check is the "CataList" service maintained by LISTSERV itself. This service lists all public lists running on LISTSERV servers worldwide. Point your Web browser of choice at the URL <http://www.lsoft.com/CataList.html> to access CataList. If you don't have a web browser, you can alternately send the command `LISTS GLOBAL search_string` in the body of mail to `LISTSERV@LISTSERV.NET` (or to LISTSERV at any host site). You will receive a mail message in return containing a list of all lists known to LISTSERV where either the name of the list or the short list description contains your search string. For instance, `LISTS GLOBAL IBM` would result in the following being returned to you:

Figure 12-1 Sample Output of `LISTS GLOBAL IBM`

```
12 Dec 2001 13:35
(search string: IBM)
Copyright 2001 L-Soft international, Inc.

L-Soft international, Inc. owns the copyright to this compilation of Inter-
net mailing lists and hereby grants you the right to copy the enclosed
information for the sole purpose of identifying, locating and subscribing
to mailing lists of interest. Any other usages are strictly prohibited.

*****
* To subscribe, send mail to LISTSERV@LISTSERV.NET with the following *
* command in the text (not the subject) of your message: *
*
*          SUBSCRIBE listname *
*
* Replace 'listname' with the name in the first column of the table. *
*****

Network-wide ID  Full address and list description
-----
AIX-L           AIX-L@NEW-LISTS.PRINCETON.EDU
                IBM AIX Discussion List
```

1. Parts of this section was adapted from *Some Lists of Lists*, compiled by Marty Hoag.

(Quite a few more lists were deleted for brevity)

You might want to make your search more specific, as this particular search locates every list that has IBM somewhere in its title. For instance, if you wanted to start a list on some aspect of the IBM 370, you might do better to search for IBM 370.

Alternative searches you can do include:

- Check the Usenet newsgroups `news.announce.newusers` and `news.lists`, if they are available to you via your local news feed.
- Use one of the World Wide Web search engines such as Alta Vista or Google to search for matches to the name you want to use.

12.2 Skills Needed to Start and Maintain a LISTSERV Mailing List

You should already be familiar with your mailing system and text editor. Otherwise, there are no special skills required. It is the goal of this manual to give you what you need to know about LISTSERV user commands, privileged LISTSERV owner commands, and how to read and interpret Internet-style mail headers¹. LISTSERV itself is designed to operate in an identical manner no matter which operating system it is running under. Thus the fact that LISTSERV is running under VM, VMS, some flavor of Unix, or Windows NT should not be a concern to the list owner, who may not even know which version of LISTSERV his lists are running on.

Additionally, we have made an attempt to give you a basic "list owner's course" in anticipation of some of the issues you may encounter in the course of moderating a list.

12.3 Creating a Mailing List Manually

If you are looking for a site to host a list, consider the following:

- First, find out if your computing center maintains a LISTSERV host.
- If not, you might consider a commercial LISTSERV site. There are a number of such sites, including L-Soft's own EASESM service. You can get more information on EASESM at <http://www.lsoft.com/products/ease.asp>.



Note: Many sites (predominantly, but not necessarily limited to, those in .EDU domains) will not host commercial or potentially-controversial lists because of internal policies regarding appropriate use of their computing facilities. In such a case, your only option may be to seek a commercial LISTSERV site.

Physically creating the list is the task of the LISTSERV maintainer (sometimes referred to as the "LISTSERV postmaster") at a given LISTSERV host site². Specific procedures for requesting a list startup vary from institution to institution. It is usually best to contact the computing center at the site for more information.

1. Standard Internet-style mail headers are described in [RFC822](#) and [RFC2822](#).

2. Note that the "LISTSERV postmaster" is not identical to the regular POSTMASTER address at a host site. The term "LISTSERV postmaster" is a canonical term from early in LISTSERV's history and refers only to the person who is the actual LISTSERV maintainer at the host site. The term has falling into disuse and its use is discouraged because of the potential confusion it may cause.

Because most list owners do not have the appropriate permissions to create lists, instructions on how to physically create lists are not included in this manual. If you are a LISTSERV maintainer, you can find these instructions in the Installation Guide that came with the software, or in the Site Manager's Operations Manual for LISTSERV.

12.3.1 Naming Conventions

When naming a list, there are a few conventions and restrictions that you should keep in mind.

12.3.1.1 The "-L" Convention

The "-L" convention isn't required, but it can help people to realize that the mail is coming from a mailing list rather than from a real person. The people we are referring to here are people who run Internet mail systems, who may see a great deal of mail coming from a single host and begin to wonder why. If it comes from a userid that ends in a "-L", they will be more likely to recognize it as list mail.

12.3.1.2 Reserved Names

You may not create lists whose names match the following wildcards:

```
owner-*
*-request
*-search-request
*-server
*-signoff-request
*-subscribe-request
*-unsubscribe-request
```

For instance, lists cannot be made with names like "owner-loyalty", "linux-server", and "donation-request". While it is physically possible to create a list with a name that matches one of the above wildcards, attempts to send mail to the list (for example, a list called "linux-server") will result in an error, logged as follows in the LISTSERV log:

```
4 Dec 2001 11:47:02 -> Invalid list (LINUX), generating bounce.
```

These "pseudo-mailboxes" have a special meaning to LISTSERV, which has internal rules that govern how mail sent to these addresses is handled. See the [Site Manager's Operations Manual for LISTSERV](#) for more information on what happens to mail sent to these special addresses.

12.3.1.3 Reserved Characters

In general, you want to avoid "special" characters such as the ones above the number keys on your keyboard. For example, don't use:

- ! can be confused for "bang-path" addressing, for example, UUCP
- @ a reserved character
- # can cause problems with some mail software that uses it for addressing
- \$ may have a special meaning to the unix shell
- % another addressing character that could cause problems

- & sometimes reserved by non-unix systems (specifically on NT, it has a special meaning to the shell). However, please note that use of this character in the name of a list or in a sendmail alias for a list will cause LISTSERV on unix to choke. Note that it is possible under unix to create a list with a "&" character in the name quite easily, and it is also possible to create a sendmail alias with a "&" character in the alias. That does not mean it will work.
- * the wildcard character
- () generally reserved and can't be used in file names
- + should be avoided because recent versions of sendmail deliver mail addressed to "user+whatever@somedomain" to "user@somedomain." Whether or not this is an intelligent thing to do on sendmail's part is left as an exercise for the user, but it can affect mail being sent to a list with a "+" character in the listname.
- / reserved and can't be used in file names
- . Although on some systems it is physically possible to create lists with a dot character in the name, LISTSERV will not accept this nomenclature. The only place a dot can or should be used is before the word "LIST" in the PUT command; for example, PUT MYLIST-L.LIST is equivalent to PUT MYLIST-L LIST.
- " not allowed

It is best if you avoid the use of special characters altogether and stick exclusively to the letters A-Z, numbers 0-9, and the underscore and hyphen characters when naming lists. Note that the "_" (underscore) character may cause problems with some non-compliant receiving systems. Also note that the space character (ASCII 0x20) is illegal in a list name, and L-Soft recommends that, although apostrophes (aka "single-quotes", ASCII 0x27) are valid in an RFC822 username, they should not be used in list names since some mail programs may not accept them.

If you have any question about the validity of a particular name, you can of course refer to [RFC822](#) or the updated [RFC2822](#) for the Internet standards for e-mail addressing.

12.3.1.4 Maximum Length of the List Name

The length of the list name (that is, the name of the list file and thus the "official" name of the list) is restricted as follows:

- VM: 8 characters
- Non-VM: unlimited (but see below)

If you need a longer list name for a list running on a VM server, then you should use the List-ID= keyword (see the [List Keyword Reference](#) document).



Note: L-Soft recommends using names of 32 characters or less whenever possible as they provide for correct alignment of the results returned by certain commands. Very long (for example, program-generated) list names are likely to conflict with mail system limits and L-Soft recommends other solutions to the problem of dynamically generated lists. As a rule, list names in excess of 70 characters are likely to result in mail delivery problems.

12.3.1.5 Making the List Name User-Friendly

While you can (within limits) name a LISTSERV mailing list just about anything you want, you will probably want to follow a couple of simple guidelines:

1. Keep the name simple.
2. Keep the name as short as possible without causing confusion.

No doubt you could name a list MY-LIST-FOR-MATH-STUDIES, but who wants to type that? Conversely, MLFMS-L wouldn't mean much to Joe Random User. Somewhere in the middle is a reasonable compromise, for example, MATH-STUDIES (or even just MATH-S).

12.4 List Header Keywords

How a LISTSERV mailing list performs its tasks is defined by its header keywords. There are several different categories of keywords, each of which is discussed below in general terms. We will discuss these keywords in detail in subsequent sections, and a complete alphabetical listing of list header keywords, including default settings and all options available, is provided in the [List Keyword Reference](#) document.

- **Access Control Keywords** – These keywords designate the level of "openness" for a list. They determine who can post to the list, who can review the list of subscribers, and whether or not the list is open to general subscription.
- **Distribution Keywords** – This group has to do with how LISTSERV distributes postings to subscribers, including whether or not acknowledgments are sent back to posters, how many postings may go through the list daily, whether or not the list is available in digest form and whether it is available to USENET through a gateway. These keywords also determine whether or not list topics are enabled, and how LISTSERV will configure outgoing postings for replies.
- **Error Handling Keywords** – Included under this group are the keywords controlling automatic deletion, loop-checking, and to whom error messages are sent for disposition when received by LISTSERV.
- **List Maintenance and Moderation Keywords** – A fairly large group of keywords having to do with how the list is operated, including definitions for the list owner, list editor, and the list archive notebook; whether or not (and who) to notify when users subscribe and sign off; how often subscriptions must be renewed, and so forth. These are perhaps the most basic keywords that can be set for a given list, and one of them ("Owner=") must be set for a list to operate.
- **Security Keywords** – These keywords control who can "see" the list (that is, whether or not the list appears in the List of Lists for a given user, based on the user's host site), whether or not the list is protected by a password, and the level of security necessary for changes to the list itself. The "Exit=" keyword is also contained in this group.
- **Subscription Keywords** – These control whether or not the list is open to general subscriptions, whether or not a mailing path confirmation is required, and what user options are set by default upon subscription.

- **Other Keywords** – These control other aspects of list management that are not generally changed from their defaults, and which do not fit readily into the categories listed above.

12.5 Sending Commands to LISTSERV

In the following sections, you will see numerous references to "sending commands to LISTSERV". All LISTSERV commands are sent to the server either by email or via the web administration interface described in Section 5 [Introduction to the Web Interface](#). For mailed commands, this means that you must create a new mail message using whatever command this requires for your mail client (click on "New message" or its equivalent for most mail clients) addressed to the LISTSERV address. Let's say for the sake of argument that the list you want to subscribe to (or are currently subscribed to) is running on a server called `LISTSERV.MYCORP.COM`. In order to send a command to that server, you would create a new message and address it to `LISTSERV@LISTSERV.MYCORP.COM`, and place the command(s) in the body (not the subject) of the message.

Depending on how you have security set up for your lists, some or all commands may require that you validate them with a personal LISTSERV password.

12.6 Defining Personal Passwords

The passwords recognized by LISTSERV for various operations (assuming that the `NOPW` parameter is not used with the `"Validate="` keyword) are of two distinct types:

- **Personal Passwords** – LISTSERV can store a personal password in its signup files corresponding to your userid. This password not only can be used for list maintenance operations, but also protects your FUI (file update information) and AFD (automatic file distribution) subscriptions (if available on your server) and must be used to store your archive files, if any, on the server.
- **List Passwords** – List passwords are obsolete except in only one special case (peered lists). We mention them here only because users upgrading from earlier versions will be aware of their existence. You should define and use a personal password for all protected operations.

To add a personal password, send mail to LISTSERV with the command

```
PW ADD newpassword
```

in the body of the message. LISTSERV will request a confirmation via the "OK" mechanism (see above) before it adds the password.

If you want to remove your password altogether, send the command

```
PW RESET
```

This command will also require confirmation.

And finally, if you simply want to change your personal password, send the command

```
PW CHANGE newpassword [PW=oldpassword]
```

If you do not include the old password in the command (e.g., you've forgotten it), LISTSERV will request an "OK" confirmation. Otherwise, it will act on the command without need for further confirmation (unless, of course, the oldpassword provided is incorrect).

Personal passwords may also be defined via the web administration interface.

12.7 Retrieving the List Configuration



Warning: Never attempt to hand-edit a production list file in place and restart the server. The GET and PUT operations are the only supported methods. Particularly under unix and Windows, LISTSERV will not always accept the hand-edited list file because some editors will insert control characters or CR-LF combinations that LISTSERV cannot parse. Under VM or VMS, it is always possible that hand-editing the list will introduce some sequence that will cause an operational error. L-Soft suggests that this method be used sparingly, if at all, and does not support it.

12.7.1 Who can edit the list configuration?

In general, any email address specified explicitly as a list owner (in the Owner= list keyword setting) may retrieve, edit, and store the list configuration (also known as the "list header").

If the list header keyword Configuration-Owner= is set, then only those list owners specified in both Owner= and Configuration-Owner= may retrieve, edit, and store the list configuration. All other owners (that is, those not specified in Configuration-Owner=) may perform other list maintenance duties, such as adding/deleting subscribers or modifying their subscription options. The Configuration-Owner= keyword is "protected" and can be changed only by someone with LISTSERV site maintainer privileges.

The LISTSERV site maintainer has "super-owner" privileges and may edit all list configurations on the server regardless of the setting of Owner= or Configuration-Owner=.

12.7.2 Retrieving the List Configuration by Email

Once your list has been created by the LISTSERV maintainer, you can have a copy of the list sent to you for editing purposes. Simply issue the command

```
GET listname (HEADER
```

to LISTSERV. This will cause the server to mail you a copy of the list header only (without the subscriber list).



Note: You can retrieve the entire list, subscribers and all, by omitting the (HEADER switch. However, L-Soft strongly discourages getting the entire list at any time. This is because you do not need the entire list file if all you want to do is to change list header keyword settings. Also, since LISTSERV has well-documented commands available to manage user subscriptions, you should never attempt to hand-edit a list file in order to add or delete subscribers. Therefore there should normally be no reason to issue the `GET listname` command without the (HEADER switch.

The GET command automatically locks the list so that no changes can be made to the operating copy on the server until you do one of two things:

- Issue the `UNLOCK listname` command (if you decide no changes are needed).
- Send the list back to the server with the `PUT` command.

Leaving the list locked also prevents new subscribers from signing up. It is therefore not advisable to leave the list locked for long periods of time. This necessitates remembering to issue the `UNLOCK` command if you decide not to make any changes.

It is possible to request that `LISTSERV` not lock the list when it is sent to you. This is accomplished by adding the `(NOLOCK` switch to the `GET` command. You can use `(NOLOCK` and `(HEADER` together as in the following example:

```
GET listname (HEADER NOLOCK
```



Note: The `"` (" switch character is used only once.

Caution: It is not advisable to use the `(NOLOCK` switch in at least two cases:

Don't use the `(NOLOCK` switch if you are not the sole owner of the list. This prevents conflicting GETs and PUTs by different list owners. For instance, Owner(A) GETs the list without locking it. Owner(B) then also GETs the list. The owners make differing changes to the list header. Owner(B) PUTs his changes back first. Owner(A) then PUTs his changes back, erasing every change Owner(B) made. If Owner(A) had not used the `(NOLOCK` switch, Owner(B) would not have been able to GET a copy of the list until Owner(A) either unlocked the list or PUT his copy back. (Owner(B) could also unlock the list himself, but it would be advisable to ask Owner(A) if he was finished editing the list header before doing so.)

Don't use the `(NOLOCK` switch if you get the entire list rather than just the header. You will erase all subscriptions for users who subscribed between the time you GET the list and PUT the list back. It is easier to deal with questions as to why they got the "listname has been locked since time by list-owner" message than to explain why they got a subscription confirmation and now aren't getting list mail.



Note: A PUT command containing new subscribers added "on the fly" will result in only the header of the list being updated and a warning being generated that says if you really wanted to PUT the entire list, subscribers and all, that you should use the `PUTALL` command.

`LISTSERV` maintainers should note one further caution: It is considered extremely inadvisable to "hand-edit" subscriber lists, as columns at the far right of each subscriber's entry contain list control codes corresponding to the subscriber's personal option settings. The only case in which it might be appropriate to "hand-edit" would be to delete a user entirely, and then only if all attempts to delete the user via the `DELETE` command fail. For instance, X.400 or X.500 addresses can cause `DELETE` to fail because of their use of the `/` character. You can use wildcards to delete these subscriptions:

```
DELETE XYZ-L *ADMD=ABC*PRMD=DEF*@X400.SOMEHOST.COM
```

You can also enclose the address in double quotes:

```
DELETE XYZ-L "/ADMD=ABC/PRMD=DEF/...../@X400.SOMEHOST.COM"
```

Finally, depending on your list configuration, you may have to use a password or respond to a confirmation request in order to GET your list header. The syntax for using a password with the `GET` command is

```
GET listname (options PW=password
```

For instance,

```
GET MYLIST-L (HEADER NOLOCK PW=MYPASSWORD
```

12.8 Editing the List Header

Once the LISTSERV maintainer has notified you that the basic list has been created, you can send a GET command to the server to make any modifications necessary, as explained above. For instance,

```
GET MYLIST (HEADER PW=MYPASSWD
```

might cause LISTSERV to send you the following list header file:

Figure 12-2 A Sample List Header File for a List Called MYLIST

```
PUT MYLIST.LIST PW=XXXXXXXX
* The Descriptive Title of My List
*
* Owner= NATHAN@EXAMPLE.COM (Nathan Brindle)
* Notebook= Yes,E:\LISTS\MYLIST-L,Monthly,Public
* Errors-To= Owner                      Send= Public
* Subscription= Open,Confirm  Ack= Yes          Confidential= No
* Validate= No                Notify= No        Reply-to= List,Respect
* Review= Public              Default-Options= NoFiles,NoRepro
*
* This list installed on 96/11/02, running under L-Soft's LISTSERV
* for Windows NT.
*
* Comment lines...
*
```

You can now physically edit this file in a couple of different ways, depending on what tools you have on your workstation:

- Cut and paste the header from your mail program into an ASCII text editor such as vi, emacs, or Notepad. Edit the various keyword values and set the password in the PUT command appropriately. Finally, cut and paste from your editor into a new mail message addressed to LISTSERV and send the message.
- Cut and paste from your mail program into the body of a new mail message addressed to LISTSERV and do your editing in the mail program.
- If you use the 'mail' or 'mailx' programs under unix, you can save the header into a text file (for instance, named myheader.txt) and mail it back to LISTSERV with the command line syntax

```
mail listserv@myhost.com < myheader.txt
```

In Figure 2-3, we've made some changes to the list header and it is ready to be included in a mail message and sent back to LISTSERV. Note that the PUT command has been modified to include your personal password (see Section 12.6 [Defining Personal Passwords](#) for instructions on how to obtain a personal password).

Figure 12-3 The Edited List Header File

```

PUT MYLIST.LIST PW=MYPASSWD
* The Descriptive Title of My List
*
* Owner= NATHAN@EXAMPLE.COM (Nathan Brindle)
* Owner= Quiet:,nathan@linus.example.com,ncbnet@linus.example.com
* Notebook= Yes,A,Monthly,Public          Auto-Delete= Yes,Full-Auto
* Errors-To= ncbnet@linus.example.com      Subscription= Open,Confirm
* Ack= Yes                               Confidential= No          Notify= No
* Validate= Yes,Confirm
* Reply-to= List,Respect      Review= Public          Send= Public
* Default-Options= NoFiles,NoRepro
*
* This list installed on 96/11/02, running under L-Soft's LISTSERV
* for Windows NT.
*
* Comment lines...
*

```

If LISTSERV responds to your PUT operation with error messages, bear in mind that the most common problems are:

1. You sent the header back from an account which is not defined as a list owner
2. You used the wrong password or didn't specify a password in the PUT command
3. Your mail program indents paragraphs by default, such that each header line does not start in column 1 and LISTSERV does not recognize your message as a list header
4. You sent the header file back as an attachment rather than as plain text in the body of the message.
5. Your mail client wrapped the lines of the header so that LISTSERV received header lines that did not begin with "*" and attempted to treat them as subscriber addresses. In this case you can probably solve the problem by increasing the right margin setting in your mail client so that messages at least 80 columns wide do not get wrapped.

12.9 Defining List Owners

List owners should be persons who will undertake the responsibility of managing the list in all of its aspects. A list owner may be a moderator; a list owner may be called upon to determine why a user can't unsubscribe from the list, or to handle delivery errors, or to fix other problems that may arise.

The primary list owner (the first owner defined) has special responsibilities as well. This owner is considered the Editor and the primary Moderator for lists that have Send= Editor but do not have Editor= or Moderator= defined. This owner receives all error messages when Errors-To= is set to "Owner". In short, the primary list owner is generally the person who is ultimately responsible for the workings of the list.

Secondary list owners fall into two categories:

- Non-Quiet list owners receive mail sent to the listname-request address, and will receive error messages if Errors-To= Owners.

- Quiet list owners will never receive delivery errors or other administrative mail from LISTSERV.

Here is a sample list header excerpt for a list with all three types of list owners defined:

Figure 12-4 Defining List Owners in the List Header File Example

```
* Owner= NATHAN@EXAMPLE.COM (Nathan Brindle)
* Owner= nathan@linus.example.com
* Owner= Quiet:
* Owner= ncbnet@linus.example.com,cheng@linus.example.com
```



Notes: All list owners defined after the `* Owner= Quiet:` line will be quiet list owners.

You can define multiple owners on a single line by separating them with a comma. If you put "Quiet:" on a line with list owner userids, you must place a comma after "Quiet:", e.g.

```
* Owner=
Quiet:,ncbnet@linus.example.com,cheng@linus.example.com
```

There must *always* be at least one non-quiet list owner. Otherwise LISTSERV sends all error messages and other administrative mail to the LISTSERV maintainer by default.

12.10 Storing the List on the Host Machine

When you are ready to store your list back on the host, include the list file in a mail message to LISTSERV. Ensure that the `PW=XXXXXXXX` command is in the first line of the mail body. Change `XXXXXXXX` to the personal password you have previously defined with the `PW ADD` command (see Section 12.6 [Defining Personal Passwords](#)). Then, send the message.

If LISTSERV has trouble processing the edited list file, it will return a discrepancy report to you with each error noted. If the errors are categorized as "warnings only," LISTSERV will go ahead and store the list. However, if any one error is categorized as a serious error, the list will not be stored and the old version will be retained.



Caution: If you are using a mail client that allows "attachments" to mail, do not "attach" the list file to your mail message. It must be in plain text with the `PUT` line at the top. LISTSERV will not translate encoded attachments.

12.11 Fixing Mistakes

LISTSERV always backs up the current list file before it stores a new copy. Should you discover that you have made a mistake (for instance, you have deleted all users by storing a header and adding users "on the fly"), it is possible to retrieve the previous copy of the list by issuing a `GET listname` (OLD command to the host server. You must then add the `PUTALL listname LIST PW=XXXXXXXX` command to the top of the file and store it. (`PUTALL` is used in this case since you are storing the entire list, not just the list header.)

12.12 Security Options

LISTSERV's security options are wide ranging, from almost no protection (easiest to administer a list, but also most open to hacker attacks) to total protection requiring validation of each and every command sent to LISTSERV for the list. It is also possible to limit access to various aspects of the list, such as who can subscribe, who can review the list of subscribers, and who can access the list archives. The list can be hidden from the LIST command, either at the global level or from all requests, including those from users on LISTSERV's local machine, or from a definable range in between.



Note: LISTSERV does not set any file system permissions for any files. LISTSERV's security features are meant to allow and deny access to LISTSERV's various files depending on how various keywords are set, but in order to make your system completely safe, you must ensure that you have also set file and directory permissions appropriately for your operating system. For instance, LISTSERV may deny GET access to certain list archive files to non-subscribers, but if you have not set the appropriate file system permissions at the operating system level, you may have left open a window for someone to reach these files via anonymous ftp.

12.12.1 The VALIDATE= Keyword

The `VALIDATE=` keyword controls the level of command validation desired for the list. The default, `VALIDATE= NO`, requires password validation only for storing the list on the server. This is often sufficient for general needs. However, when a list is set this way, LISTSERV only compares the RFC822 "Sender:"/"From:" headers against the `Owner=` keyword(s) in the list header to determine whether or not the person ostensibly sending the commands has authority to do so. Otherwise at this level LISTSERV does not validate commands it receives for the list, under the assumption that the mail it receives is genuinely coming from a list owner. This level of validation does not protect the list from commands issued by hackers who have forged mail in the name of the list owner. If you run a list on a controversial topic or just don't feel comfortable without at least some security, `VALIDATE= NO` is probably not for you.

The next level is `VALIDATE= YES`. At this level, LISTSERV requires a password for all of its "protected" commands. This password is the sender's personal LISTSERV password as defined by the `PW ADD` command. The commands protected by this level are those that affect subscriptions or the operation of the list, for example, `DELETE` or `ADD`. Users will also have to validate most commands that affect their subscriptions, but generally can do so using the "OK" mechanism rather than defining a personal password. Note that some user commands will be forwarded to the list owner for validation rather than accepting password validation from the user.

The next level is `VALIDATE= YES, CONFIRM`. At this level, LISTSERV will require validation with the "OK" mechanism (see below) by default, but will still accept passwords where appropriate. While the less-secure passwords are still accepted, this is considered a good compromise between list security and list owner and user convenience.

The next level is `VALIDATE= YES, CONFIRM, NOPW`. At this level, LISTSERV will no longer accept passwords as validation for protected commands. The logic is that because of the way the "OK" mechanism is implemented, passwords are not as safe as "magic cookies". This is the recommended setting for lists that must be kept secure.

Two other levels are `VALIDATE= ALL,CONFIRM` and `VALIDATE= ALL,CONFIRM,NOPW`. These levels require "OK" validation for all commands that cause a change in state except for the `PUT` command. If `NOPW` is not specified, passwords are accepted where appropriate. With these levels, commands that do not cause a change in state (e.g., `QUERY` and other strictly-informational commands) do not require validation.



Note: `LISTSERV` requests coming from the local system via `CP MSG` or `CP SMSG` on VM systems or via `LCMD` on VMS or Unix systems never require validation, as they cannot be forged.

Lists which are set to either `Validate= Yes,Confirm,NoPW` or `Validate= All,Confirm,NoPW` may not be managed via the web administration interface, which is password-driven.

See the [List Keyword Reference](#) document for complete information on the `VALIDATE=` keyword.

12.12.2 Controlling Subscription Requests

Subscription requests are controlled by use of the `SUBSCRIPTION=` keyword. By default, this keyword is set to `SUBSCRIPTION= BY_OWNER`, meaning that all subscription requests will be forwarded to the list owner for disposition. Subscription requests can be refused completely by setting `SUBSCRIPTION= CLOSED`.

To code a list for open subscriptions without list owner intervention, set `SUBSCRIPTION= OPEN`. If it is desired to add protection against forged subscription requests or bad return mailing paths, code `SUBSCRIPTION= OPEN,CONFIRM`. The latter will cause a subscription confirmation request to be sent to the prospective subscriber, which he or she must respond to using the "OK" confirmation mechanism.

In order to restrict subscriptions to persons in a specific service area, see the next section.

12.12.3 Controlling the Service Area of the List

The `Service=` keyword is not available in `LISTSERV Lite`.

It may be desirable to restrict access to a list to people in a small area. For instance, you probably would not want a list for students in a class section at a university to be advertised or accessible by people all over the world. However, without setting certain keywords appropriately, such a list will be visible to a `LISTS GLOBAL` command.

The local list of (public) lists can be retrieved only by those users who are considered local, per the setting of the server-wide `LOCAL=` variable in `LISTSERV`'s site configuration file. All other users will be told that none of the lists on the server are visible via the `LISTS` command, and will be referred to the use of the `LISTS GLOBAL search-text` command or to the `CataList`. This is regardless of the setting of `Confidential=` as outlined below.

To simply hide a list from a `LISTS` command, but still allow people to subscribe to it if they know it is there, use the keyword `Confidential= YES`. Note that users subscribed to the list as well as the list owner(s) will be able to see the list if they issue a `LISTS` command. In addition, all other non-subscribers, including users on the local machine, will not be able to determine that the list exists via a `LISTS` command.

To hide a list from and refuse subscription requests from users outside the local area, you define two keywords:

```
* Service= bitnode1,bitnode2,some.host.edu
* Confidential= SERVICE
```

`Service=` can also be set to `Service= LOCAL`, meaning it will use either LISTSERV's global definition of which machines are `LOCAL`, or the machines defined by the list keyword `Local=`. The LISTSERV maintainer should define hosts and nodes that are considered local with the server-wide `LOCAL=` variable in the site configuration file. If the global definition is not suitable, it can be overridden by defining the `Local=` list header keyword:

```
* LOCAL= bitnode1,bitnode2,some.host.edu,another.host.co
m* SERVICE= LOCAL
* CONFIDENTIAL= SERVICE
```

If there are many subdomains within your primary domain, it may be preferable to use the wildcard when defining the `LOCAL` or `SERVICE` list header keywords. For instance:

```
* SERVICE= *.HOST.COM
```

defines the service area for a specific list as "all subdomains ending in `.HOST.COM`".



Note: Defining a service area for a list controls only from which domains subscription requests may be accepted. It does not control who may post to the list. Depending on local circumstances, it may be desirable to set lists with controlled service areas to `Confidential= Service`.

12.12.4 Controlling Who Reviews the List of Subscribers

For whatever reason, it may be desirable to restrict the ability to review the subscriber list either to subscribers or to list owners. This is done by setting the `REVIEW=` keyword appropriately.

To allow anyone, including non-subscribers, to review the list, set `REVIEW= PUBLIC`.

To restrict reviews of the list to subscribers only, set `REVIEW= PRIVATE`. This is the default.

To restrict reviews of the list to list owners only, set `REVIEW= OWNERS`.

Reviews can also be restricted to users within the list's service area by setting `REVIEW= SERVICE`, and defining the `SERVICE=` keyword appropriately (see the preceding section).

12.12.5 Controlling Access to the Notebook Files

Restricting access to the list's notebook archive files is similar to controlling who may review the list. It is accomplished by setting the fourth parameter of the `NOTEBOOK=` keyword to an appropriate value. For instance,

```
* NOTEBOOK= Yes,A,Monthly,Public
```

defines a monthly notebook on LISTSERV's A disk that is accessible by anyone. Change `Public` to `Private` if you wish only subscribers to be able to access the notebooks. The same access-levels are available for this keyword as for `REVIEW=`. (See the [List Keyword Reference](#) document for a discussion of access-levels.)

It is possible to define "Service=" in terms of IP address blocks in order to limit access to list archive notebooks. See "Service=" in the [List Keyword Reference](#) document for details.

If enabled, notebook archives are private by default.

12.12.6 Controlling Who Can Post Mail to a List

The `Send=` list header keyword is the basic control for who may post mail to the list. If the list allows non-subscribers to post, set `Send= Public`. (This is the default.)

For a list that does not allow non-subscribers to post, set `Send= Private`.

For a list where all posts should be forwarded to a moderator/editor, there are two settings:

- `Send= Editor` forwards all postings to the list editor (see the `Editor=` and `Moderator=` keywords). This setting allows the editor to make changes before forwarding the message back to the list. Note that your mail program must be capable of inserting "Resent-" header lines in your forwarded mail—if it is not capable of this, all such posts forwarded to the list will appear to be coming from the editor. Check with your system administrator if you are not sure whether or not your mail program inserts the "Resent-" headers.
- `Send= Editor, Hold` forwards a copy of the posting to the editor but differs from `Send= Editor` in that `LISTSERV` holds the posting for a period of time (usually 7 days) until the editor confirms the message with the "OK" mechanism (see below). Unconfirmed messages simply expire and are flushed by `LISTSERV`, so there is no need to formally disapprove a posting. This method of message confirmation is well-suited to lists where it is not often necessary to modify the text of a posting, and also is an excellent work around if the editor's mail program does not generate "Resent-" headers in forwarded mail.

Below is a sample of the editor-header for a list set to `Send= Editor, Hold`:

Figure 12-5 The Editor-Header for a List Set to `Send= Editor, Hold`

```
Date:      Tue, 4 Aug 1998 10:47:21 -0500
From: "L-Soft list server at PEACH.EASE.LSOFT.COM (1.8d)"
        <LISTSERV@PEACH.EASE.LSOFT.COM>
Subject:    B5-L: approval required (9723A0DD)
To: Joe ListOwner <joe@PRUNE.EXAMPLE.COM>

This message was originally submitted by jack@UNIX.FOO.COM to the B5-L list
at PEACH.EASE.LSOFT.COM. You can approve it using the "OK" mechanism, ignore
it, or repost an edited copy. The message will expire automatically and you
do not need to do anything if you just want to discard it. Please refer to
the list owner's guide if you are not familiar with the "OK" mechanism;
these instructions are being kept purposefully short for your
convenience in processing large numbers of messages.

----- Original message (ID=9723A0DD) (13 lines) -----
```

- `Send= Editor, Hold, Confirm` is identical to `Send= Editor, Hold` except that postings coming directly from an editor must be confirmed (with the "OK" mechanism) by the editor who sent the message. *This is the recommended setting*

for any moderated list or announce-only list as it protects the list from hackers who might try to forge mail from a legitimate editor address.

- A final method (called "self-moderation") exists for lists where subscribers should be allowed to post freely, but non-subscriber posts should always be sent to an editor for approval. To enable self-moderation, set

```
Send= Editor          (or Send= Editor, Hold)
Editor= userid@host, (listname)
```

Ensure that "listname" is in parenthesis. Note that self-moderation will catch all posts from non-subscribers—including posts from subscribers who are posting from a different address. For instance, if the subscriber originally signed up as joe@foo.com but is posting from joe@unix1.foo.com, LISTSERV will treat his mail as non-subscriber mail. Self-moderation may require some slight changes in individual user subscriptions in order for it to work seamlessly. See also the Default-Options= list header keyword description.

12.12.7 The "OK" Confirmation Mechanism

Depending on the setting of the Validate= list header keyword, certain LISTSERV commands have always required a password for execution. However, with a recognition that mail can be forged ("spoofed") by just about anyone on the Internet today, L-Soft introduced a "magic cookie" method of command validation that is considered much more secure than passwords.

In essence, the "magic cookie" method requires that the sender of the command must confirm his command via a reply containing only the text "OK". (This is actually simplistic; see below.) If mail is spoofed from the list owner's user id, the command confirmation request will always be sent to the list owner's user id, thus preventing the spoofer from confirming the command. Moreover, the "cookie" itself (an eight-digit hexadecimal number) is registered to the "From:" user id of the original command. A typical command confirmation request looks like this:

Figure 12-6 A Typical Command Confirmation Request

```

Date:      Wed, 5 Aug 1998 09:50:06 -0400
From:      "L-Soft list server at LISTSERV.EXAMPLE.COM (1.8d)"
           <LISTSERV@LISTSERV.EXAMPLE.COM>
Subject:   Command confirmation request (5C019D91)
To:        joe_user@EXAMPLE.COM

Your command:

                PW REP XXXXXXXXX

requires confirmation. To confirm the execution of your command, simply
point your browser to the following URL:
    http://listserv.example.com/scripts/wa.exe?OK=5C019D91

Alternatively, if you have no WWW access, you can reply to the present
message and type "ok" (without the quotes) as the text of your message. Just
the word "ok" - do not retype the command. This procedure will work with any
mail program that fully conforms to the Internet standards for electronic
mail. If you receive an error message, try sending a new message to
LISTSERV@LISTSERV.EXAMPLE.COM (without using the "reply" function - this
is very important) and type "ok 5C019D91" as the text of
your message.

Finally, your command will be cancelled automatically if LISTSERV does not
receive your confirmation within 48h. After that time, you must start over
and resend the command to get a new confirmation code. If you change your mind
and decide that you do NOT want to confirm the command, simply discard the
present message and let the request expire on its own.

```

The general method of replying to a command confirmation request is to use the web browser confirmation method outlined in the confirmation request.

If you prefer, you can use the old method of responding by mail:

- REPLY to the command confirmation request with the text "ok" in the body of the reply. (Non-case-sensitive) LISTSERV reads the "cookie" from the subject line and if it corresponds to a held job, the job is released and processed.

If this does not work, it is possible that the Subject: line was corrupted in transit and you may need to try the following:

- SEND a new message to LISTSERV with the text "ok xxxxxxxx" (where xxxxxxxx is the command confirmation number from the original confirmation request) in the body of the reply.

It is also possible to confirm multiple command confirmation requests with a single message (for instance, if you have Send= Editor, Hold and have a number of requests to be responded to). This eliminates multiple "Message approved" mails from LISTSERV. However, make sure that you send the confirmations in a new mail message rather than replying to one of them. (See the "bracketed OK" syntax mentioned below.)

You can send the "OK" from any address, which helps when the address field of your mail gets changed somewhere along the line. For instance if you are logged into the web administration interface as joe@example.com and issue a command that requires mail confirmation, LISTSERV will send the request to joe@example.com (as expected). If your mail system expands joe@example.com to Joe_Doakes@mail.example.com, the "OK" will still succeed and Joe_Doakes@mail.example.com will get a message that says


```
> ok
Confirming:
> QUIET DELETE * jane@example.com
[reply sent to joe@EXAMPLE.COM]
```

while as a protection against "spoofed" commands the actual command response will be sent to joe@example.com like this:

```
jane@EXAMPLE.COM has been removed from the TEST list. No notification
has been sent.
```

```
Global deletion process complete, one entry removed.
```

The "OK" confirmation mechanism also has the following features:

- An "OK" without an argument (confirmation number) flushes the job stream so any text following an "OK" on a line by itself will not be seen by the LISTSERV command processor.
- Bracketed "OK" functionality. This feature allows you to send multiple commands for which LISTSERV will request only a single "OK" (where normally you would expect to have to "OK" each individual command). The syntax is as follows:

```
OK BEGIN
command1
command2
...
command3
OK END
```

- A command confirmation ("OK") may now be sent by clicking on a web URL provided in the command confirmation request (mailed "OK"s are still perfectly acceptable, of course).



Note: In a "bracketed OK" the aggregate length of the data stream (that is, the total number of characters in the command lines falling between OK BEGIN and OK END) MUST be less than 32K characters. In practice you should use bracketed OKs for limited numbers of commands only, say no more than 10-12 at a time. In particular, if you have many ADD or DELETE commands to send, it is far more efficient (and strongly preferred) to use the bulk ADD and bulk DELETE syntaxes described in Section 13.4 [Adding Subscribers to Lists in Bulk](#) and Section 13.5 [Deleting Subscribers from Lists in Bulk](#).

12.12.8 Explicitly Cancelling "OK" Cookies

It is possible to explicitly cancel an OK confirmation cookie. The command is simply

```
OK CANCEL xxxxxxxxx
```

(for instance, "OK CANCEL 8F2E8F4B"), and if the cookie is valid, LISTSERV will respond "Confirmation code 8F2E8F4B cancelled." If the cookie is not valid (e.g. has expired, has already been cancelled, or is simply incorrect), LISTSERV will send its standard message telling you in part that "The confirmation code 8F2E8F4B does not correspond to any pending command."

12.12.9 Restricting Subscriber Privileges

Another security issue involves protecting the list from people who refuse to play by the rules. LISTSERV includes several different levels of privilege restriction for these users; some are available for list owners without the intervention of the LISTSERV maintainer.

1. The REVIEW personal option setting. By issuing a `SET listname REVIEW FOR userid@host` command to LISTSERV, you can moderate postings at the individual subscriber level. Postings from subscribers set to REVIEW are passed on to the Editor(s) or Moderator(s) of the list, or, if neither of these keywords are defined for your list, the postings are passed on to the primary list owner. At this point, the person who receives the postings can determine whether or not to approve them. Note that the subscriber always receives notification that his or her posting has been forwarded to a moderator for approval. This is to avoid the impression that the subscriber's posting has been lost before reaching LISTSERV.
2. The NOPOST personal option setting. By issuing a `SET listname NOPOST FOR userid@host` command to LISTSERV, you can prevent a subscriber from posting to the list entirely. LISTSERV will reject postings from these subscribers and will not pass them on to a moderator. As with the REVIEW setting, note that the subscriber always receives notification that his or her posting has been rejected.
3. The FILTER= list header keyword. You can filter individual users from subscribing and/or posting to your list by adding them to the Filter= list header keyword. For instance, if you have a list called MACTALK-L and you want to discourage redistribution lists from using the same name as your list, you can add

```
* Filter= Also,MACTALK-L@*
```

See the [List Keyword Reference](#) document for more information on the Filter= syntax.

12.12.10 Restricting the Number of Postings Per User Per Day

You can control the maximum number of postings per day per subscriber on a list-by-list basis by setting the optional second parameter of the "Daily-Threshold=" list header keyword. The default is to have no such daily limit per user.

If set, when the per-subscriber threshold is reached, the subscriber is told that his message cannot be processed because he has reached the limit for today, and that he should repost his message at a later time. The counter for this limit resets to zero at midnight for all lists.

This limit is waived for the list owner(s) and any list editors/moderators.

If you want to set this limit, note that an overall daily threshold must be set for the list in the first parameter of the keyword. If no "Daily-Threshold=" keyword is already present in your list header, the default is "Daily-Threshold= 50". Thus, to leave the default value in force and to add a daily limit of 5 postings per day per user, you would code:

```
* Daily-Threshold= 50,5
```

For more information, see the [List Keyword Reference](#) document.

12.13 Setting Up Lists for Specific Purposes

You can create certain types of lists from standard templates via the web administration interface. See Section 5 [Introduction to the Web Interface](#) for information on how to access the web administration interface.

12.13.1 Public Discussion Lists

Public discussion lists have always been the "classic" type of LISTSERV mailing list. Such lists are available to discuss just about everything imaginable. In the last few years it has become desirable to secure mailing lists against random spamming and mail bombing, but no discussion of different types of lists would really be complete without talking about this kind of list.

Typically, a public discussion list is wide-open (although some things, like the ability to review the subscribership, may be restricted). Anyone can subscribe (with a confirmation to verify the mailing path), anyone can post, anyone can read the messages in the archives, and security is set fairly low. Very large lists (hundreds or even thousands of users with hundreds of postings every week) may likely be set up this way as it is a "low-maintenance" way to run a list (and most spams tend to be caught by LISTSERV's anti-spamming filters anyway). For instance, you might have

```
* My public discussion list (MYLIST-L)
* Subscription= Open,Confirm
* Ack= Yes
* Confidential= No
* Validate= No
* Reply-to= List,Respect
* Review= Owners          Send= Public          Errors-To= Owner
* Owner= joe@example.com
* Notebook= Yes,E:\LISTS\MYLIST-L,Weekly,Public
```

For more security, you might want to code

```
* Validate= Yes,Confirm
```

and if you want to cut down on the amount of "me-too"ism on the list, you could set

```
* Reply-to= Sender,Respect
```

to force the default Reply-To: header to point back to the original poster instead of to the list. Note that the ",Respect" option means that if a user sends mail to the list that contains a "Reply-To:" header pointing back to the list (unlikely that this may be), LISTSERV will "respect" that header and use it. If you absolutely do not want this to be possible, you should code the following instead:

```
* Reply-to= Sender,Ignore
```



Caution: "Reply-To:" are not universally honored!

Note: There is one major caveat with regard to the use of the Reply-To= list header keyword. Setting this parameter guarantees only one thing -- that LISTSERV will generate an appropriate RFC822 Reply-To: header in the mail it distributes to subscribers. THERE IS UNFORTUNATELY NO GUARANTEE THAT THE MAIL TRANSFER AGENT (MTA) OR MAIL CLIENT ON THE RECEIVING END WILL HONOR THE Reply-To: HEADER. This is because some mail clients, out-of-office robots, and Internet MTAs either simply do not recognize the existence of Reply-To:

or do not implement it properly. Specifically RFC2076 "Common Internet Message Headers" reports that the use of Reply-To: is "controversial", that is, "The meaning and usage of this header is controversial, meaning that different implementors have chosen to implement the header in different ways. Because of this, such headers should be handled with caution and understanding of the different possible interpretations." (RFC2076, page 4). While L-Soft recognizes that it is sometimes important to provide an explicit Reply-To: header to indicate a response path, L-Soft cannot and will not be held responsible for problems arising from the inability of a remote server to properly process Reply-To: headers.

12.13.2 Private Discussion Lists

Private discussion lists are similar to public discussion lists, but with varying restrictions on who may subscribe, who may post and who may view the archives. Such lists are relatively safe from random spamming since typically only a subscriber can post (but note that a spammer spoofing mail from a subscriber's address will probably be successful unless first caught by the anti-spamming filters). For instance:

```
* My private discussion list (PRIVATE-L)
* Subscription= By_Owner
* Ack= Yes
* Confidential= Service
* Validate= No
* Reply-to= List,Respect
* Review= Owners
* Send= Private
* Errors-To= Owner
* Owner= joe@example.com
* Notebook= Yes,E:\LISTS\PRIVATE-L,Weekly,Public
```

is a low-security private discussion list where subscriptions requests are passed on to the list owner(s) for review, only subscribers may post, and only subscribers may view the list archives. Here again, for more security you might want to set "Validate= Yes,Confirm", and of course you can have replies go to the original poster rather than to the list with "Reply-To= Sender,Respect" (with the same caveats as noted in Section 12.13.1 [Public Discussion Lists](#)).

12.13.3 Edited Lists

An edited list is one that requires a human editor to approve messages sent to the list. Some list software and most USENET newsgroups refer to this as "moderation", but to avoid confusion between two types of moderated LISTSERV lists, the present example will be referred to as an "edited" list.

Examples of edited lists range from refereed electronic journals to lists where the list owner simply wishes to exercise control over which postings are allowed to go to the list.

To set up a basic edited list, simply add

```
* Send= Editor
* Editor= someuser@somehost.com
```

to the basic list header. Note that the primary Editor= specification (that is, the first editor defined by an Editor= keyword for the list) must be a human person who will be

able to act on postings sent to him or her for approval. You may not use an access-level specification (such as "Owner") when defining the primary editor for a list.

Please note that L-Soft recommends setting "Send= Editor,Confirm" so as to add a level of security against malicious users forging mail from an "Editor=" address to get around your moderation settings, or against badly-configured "vacation" programs that simply reflect the message back to the list in a manner that makes it appear that the mail is coming from the editor's address. The "Confirm" option causes LISTSERV to request an "OK" confirmation from an editor when it receives mail claiming to be from that editor.

You can define multiple editors, but only the first editor will receive postings for approval. Anyone defined as an editor may post directly to the list without further intervention.

Multiple editors can be defined on separate Editor= lines or can be grouped several on a line, for example,

```
* Editor= someuser@somehost.com,anotheruser@anotherhost.com
* Editor= yetanotheruser@his.host.com
```

To approve postings with the above configuration, the editor simply forwards (or "resends", or "bounces"--the terminology is unclear between various mail programs) the posting back to the list address after making any desired changes to the content. This should be done with a mail program that supports "Resent-" fields. If "Resent-" fields are not found by LISTSERV in the headers of the approved posting, then the posting will appear as coming from the editor's address rather than from the original poster. If your mail program does not support "Resent-" fields, you should use the "Send= Editor,Hold" option and approve messages with the "OK" mechanism described below.

If you do not need to physically edit the content of your users' posts (for instance, to remove anything considered "off-topic" or to remove included mail headers and so forth), you can code

```
* Send= Editor,Hold
```

The "Hold" parameter causes LISTSERV to send you a copy of the posting along with a "command confirmation request". To approve the posting, you simply reply to the confirmation request with "ok".

For security purposes, you can code

```
* Send= Editor,Confirm
```

which will cause LISTSERV to request a command confirmation ("ok") from the editor sending the approved posting back to the list. This makes it impossible for an outside user to "spoof" mail from an Editor address.

Naturally, you can also code

```
* Send= Editor,Hold,Confirm
```

Finally, please note that the NOPOST subscriber option will take precedence over Editor=, if set for someone defined as an editor. This means that if you have "Default-Options= NOPOST" for your list and you add an editor as a subscriber, you will have to manually reset the editor to POST (with "SET listname POST FOR userid@host") before things will work properly. You will know that this is necessary if your editor can successfully approve postings but is then told that he or she cannot post to the list.

12.13.4 Moderated Lists



Note: The `Moderator=` keyword is disabled in LISTSERV Lite.

A moderated list is similar to an edited list, but for LISTSERV's purposes it refers to a list that uses the `Moderator=` list header keyword to "load-share" posting approvals among several editors. It is set up similarly to an edited list, as follows:

```
* Send= Editor,Confirm
* Editor= someuser@somehost.com
* Moderator= someuser@somehost.com,anotheruser@anotherhost.com
* Moderator= yetanotheruser@his.host.com
```

This list will "load-share" the approval process between the three moderators, who will each receive one-third of the postings for approval. Note that a primary editor should still be defined.

If it is desired to have one editor handle more than a single share of the approvals, you simply define the editor more than once in `Moderator=`. For instance,

```
* Send= Editor,Confirm
* Editor= someuser@somehost.com
* Moderator= someuser@somehost.com,anotheruser@anotherhost.com
* Moderator= someuser@somehost.com,yetanotheruser@his.host.com
```

would cause every other posting to be forwarded to `someuser@somehost.com` for approval.

If the parameter "All" is coded at the beginning of the list of moderators, LISTSERV will send copies of all postings to all moderators, any of whom may approve the message. An example of this would be

```
* Moderator= All,kent@net.police.net,joe@bar.edu
```

Please note that something like

```
* Moderator= kent@net.police.net,All,joe@bar.edu,alex@reges.com
```

is not valid. "All" must appear at the beginning of the list of moderators.

Assuming "`Send= Editor, Hold`", once a message is approved by one of the moderators, any other moderator attempting to approve the same message will be told that the message cannot be found and has probably expired (since the cookie for that message will be gone).

If the message body is edited in any way before it is approved (i.e., by forwarding an edited copy back to the list), and more than one moderator is involved, duplicates are possible. Thus it is important that the moderators of any list set up this way pay close attention to whether or not the posting has already been approved by another moderator. Note carefully that this means if the "All" parameter is used in "`Moderator=`" with "`Send= Editor`" (that is, without the "Hold" parameter), again a separate synchronization method will have to be used to prevent duplicates, as two moderators are unlikely to make exactly the same edits to the message. Even if LISTSERV were able to identify the two submissions as being the same message, it would not know which to choose over the other.

The "Hold" and "Confirm" options for "`Send=`" can also be used with these examples, if desired. L-Soft recommends that "Confirm" be used by default.



Note: The NOPOST subscriber option will take precedence over both Editor= and Moderator=, if set for someone so defined. This means that if you have "Default-Options= NOPOST" for your list and you add an editor or a moderator as a subscriber, you will have to manually reset the editor to POST (with "SET listname POST FOR userid@host") before things will work properly. You will know that this is necessary if your editor or moderator can successfully approve postings but is then told that he or she cannot post to the list.



Note for moderation "OK" requests and MIME attachment display: In versions previous up to LISTSERV 1.8e, an OK confirmation request for a message coming to a moderated list displayed the message to be approved in its "raw" format; that is, there was no attempt made to display/decode MIME attachments that might be present in the message to be approved. LISTSERV now addresses the problem by including a copy of the first text/plain part (if one exists in the message) for the purpose of quick screening. The following restrictions apply:

- 1.) This is only done for MIME messages (even simple single-part ones, but they must have MIME headers).
- 2.) The text part in question is sent pretty much 'as is', that is, as an extra text/plain part in the message, with all the options and encoding and what not supplied in the original message. The reason is quite simply that it would be a lot of work and, in some extreme cases (incompatible code page, etc.), completely impossible, to embed it into the first text/plain part with the LISTSERV message. The drawback is that some mail agents might conceivably only show the first part until you take some kind of clicking action.

It is important to understand that only the first text/plain part is extracted in this fashion. The goal was to make it easier to approve or reject simple text messages, not to build a factory around a simple problem. The ENTIRE message is available at an extra click.

Where security is a concern, it is important to review the ENTIRE original message and not just the plain text part. There could be an obscene GIF or another text part or a text/html part not matching the contents of the text/plain part or whatever. This is why, again, you are given the ENTIRE original message.

List owners using certain email clients (specifically Pine, which handles attachments in a secondary viewing area) may find the new format difficult to use. If preferred, the pre-1.8e behavior may be reverted to by specifying "NOMIME" in the Send= list header keyword; for instance,

* Send= Editor, Hold, NoMIME

12.13.5 Semi-Moderated Lists

"Semi-moderation" was developed some years ago after a great debate on whether or not an "urgent" message should be allowed to be posted to an edited list without having to go through the approval process. Although this option is still available, it can be misused by anyone who knows about it, and is therefore not generally recommended for use. However, should this feature be deemed necessary, it is activated by setting

```
* Send= Editor,Semi-Moderated
```

Then, any subscriber needing to send an "urgent" message to the list simply types "Urgent:" in the subject line of their mail, followed by the subject of the message. Messages that do not have the "Urgent:" subject are forwarded to the list editor for approval as usual.

In order to minimize the chance of spam slipping through without editorial approval, messages with an "Urgent:" subject originating from non-subscribers will be rejected.

12.13.6 Self-Moderated Lists

So-called "self-moderated" lists were invented in 1993 or 1994 when the current epidemic of spamming was beginning to get cranked up and before the "spam filter" was developed by L-Soft. With the spam filter in operation, self-moderation is not as much of an issue anymore, but some lists still run this way.

Self-moderation takes advantage of the ability to make an access-level a secondary list editor, and is implemented as follows:

```
* Send= Editor,Confirm
* Editor= someone@someplace.com,(listname)
```

(The "Hold" and "Confirm" parameters for "Send=" may naturally be used if required. L-Soft recommends that "Confirm" be used by default.)

Usually, one of the list owners is the primary editor (here "someone@someplace.com") and the specification of (*listname*) makes all of the subscribers of the *listname* list editors, and thus eligible to send messages directly to the list without editor intervention. Postings from non-subscribers (e.g., spammers) are deflected to the primary owner for his or her disposition.

There is one caveat to this kind of list. If a user subscribes to the list, and later his mail address changes (for instance, the hostname changes slightly but mail sent to the old address is automatically forwarded to the new address), any postings from him to the list from the new address will be forwarded to the editor because the new address is not subscribed to the list. Thus there is a certain amount of list-owner overhead on this kind of list in keeping track of users whose addresses have changed and modifying the subscriber list to reflect those changes. The "CHANGE" command can be of help in this regard.

12.13.7 Private Edited/Moderated Lists

This type of edited or moderated list allows subscribed users to post with editor or moderator intervention, but rejects postings received from non-subscribers with a note to the poster stating that they are not allowed to post.

Using the same header you would create for an private discussion list (see Section 12.13.2 [Private Discussion Lists](#)), simply add the following line to the header:

```
* Default-Options= REVIEW
```

You should also add `Editor=` and (optionally) `Moderator=` keyword settings to the list. At least one editor must be defined to handle the message approval chores, otherwise the first listed list owner will receive the messages for approval.

The following rules apply:

- For brand-new lists or existing lists which have no subscribers, all subscribers added to the list after this option is set will be set to REVIEW, and nothing further needs to be done.
- For existing lists with existing subscribers, you will need to set the existing subscribers to the REVIEW option manually, that is, with the command

```
QUIET SET listname REVIEW FOR *@*
```

- New subscribers who sign up or are added after you add the `Default-Options=` keyword setting will automatically be set to the REVIEW option.
- Finally, the list editor will also be set to REVIEW if he is subscribed to the list under this scenario. This can be important if the list editor wants to approve even his own postings (for instance, to help avoid someone spoofing mail to the list from his address). If the list editor does not require this "suspenders and belt" level of security, he can simply set himself to NOREVIEW.

12.13.8 Auto-Responders

Since LISTSERV Lite does not support list-level mail templates, this functionality is effectively not available in LISTSERV Lite.

An "auto-responder" is a type of list that simply responds with a set message whenever it receives mail from someone. This kind of list can be useful for things like service messages or upgrade availability, or even to simply send back a standardized message to a user who has sent mail to a "support" address.

A simple auto-responder header might look like this:

```
* Auto-responder for service messages
* Owner= someone@someplace.com
* Send= Public      Notebook= No      Subscription= Closed
```

In other words, it can be very simple, since you probably don't want notebook archives for this kind of auto-responder, you don't want people to subscribe to the list as it isn't really a mailing list, and so forth. To make the auto-response message for this list, you'd then create a `listname.MAILTPL` file that includes a `POSTACK1` template, like the following:

```
>>> POSTACK1 Service Message for &MYNAMES
&MYNAMES will be down Sunday from 0200 EST until 0500 EST for backups
and upgrades. For more information contact LSTMAINT@&MYHOST.
```

This particular template would inform the user that LISTSERV would be down (&MYNAMES translates to `LISTSERV@NODE` where `NODE` is the value of `NODE=` in the system configuration file) and to send questions to `LSTMAINT@` the local host. In order to change the service message, it would be necessary to change the `POSTACK1` template.

12.13.9 Announce-Only Lists

An "announce-only" list would be used to distribute a newsletter or other timely information where responses to the list are neither expected nor desired. A typical announce-only list header might look like this:

```
* The FOO Product Announcement List
* Owner= foo@myhost.com
* Owner= Quiet:
* Owner= anotheruser@myhost.com
* Owner= yetanotheruser@myhost.com
* Editor= foo@myhost.com
* Editor= anotheruser@myhost.com
* Editor= yetanotheruser@myhost.com
* Notebook= No
* Errors-To= Owner
* Subscription= Open,Confirm
* Validate= No
* Review= Owners
* Send= Editor,Confirm
* Reply-To= foo@myhost.com,Ignore
* Sender= None
```

This list is set up so that generally any response to postings will go back to `foo@myhost.com`, which might be a special account set up specifically to handle such things, or a mail alias pointing to another account. The newsletter can be posted by `foo`, or `anotheruser`, or `yetanotheruser`, all of whom are editors, but the likelihood is that it would be posted from the `foo` userid so that the From: line would read "From: `foo@myhost.com`".

L-Soft strongly recommends that all announce-only lists use the "Send= Editor,Confirm" or "Send=Editor,Hold,Confirm" setting. The ",Confirm" parameter tells LISTSERV to require a confirmation for any posting sent by a user defined as an Editor=. This is important for two reasons:

- **Security** – This setting tells LISTSERV to request confirmation from the Editor for all postings it receives that purport to be from that Editor. This prevents hackers from forging mail under an Editor's address, because any forgeries will require that the Editor in question approve them before they go to the list.
- **Loop Protection** – Certain broken mailers can and will bounce mail back to your list in a "reflected" manner, that is, the bounce will appear to be a legitimate posting from the Editor to the list instead of looking like an error. This is different from a forgery attempt because (it is assumed) the mailer on the other end is not doing this with malicious intent. Requiring the editor confirmation will stop these potential loop-generating messages from getting through to the list.

To stop a posting from going to the list under this scenario, simply don't OK it and delete the confirmation request message.

12.13.10 Restricted Subscription Lists with Automatically-Generated Questionnaire

Since LISTSERV Lite does not support list-level mail templates, this functionality is effectively not available in LISTSERV Lite.

Sometimes it is desired to send out a little questionnaire before approving a subscription to a list with a very narrowly-defined topic or to lists created for members of specific organizations. By setting "Subscription= By_Owner", you can of course force all potential subscriptions to require list owner approval. In the "old days", if you wanted more information before you approved the subscription request, you had to manually send a questionnaire out to the user and wait for him or her to return it to you.

By setting "Subscription= By_Owner" and adding two simple template forms to your listname.MAILTPL (as explained in Section 18 [Working with Mail and Web Templates Using Email](#)), you can now have LISTSERV send your questionnaire out automatically, as soon as the subscription request is received.

The first template form you need to add to *listname*.MAILTPL is called SUB_OWNER, and in this case it would typically look like this:

```
>>> SUB_OWNER &LISTNAME: &WHOM requested to join
.TO &WHOM
A copy of the &LISTNAME membership questionnaire has been sent to you.
Please read it carefully and follow the instructions to complete it and
return it to the list owners.
```

The .TO &WHOM directive is required so that the message is sent to the subscriber rather than to the list owner. If you want the non-quiet list owners to receive a copy of this message (which is admittedly unlikely), you can simply add CC: &OWNERS to the end of the .TO line, for example,

```
.TO &WHOM CC: &OWNERS
```

Or, if you want to cc: a specific user such as joe@unix1.example.com, use

```
.TO &WHOM CC: joe@unix1.example.com
```



Note: You cannot format the SUB_OWNER template; it all comes out as one long paragraph without formatting no matter what you do, because it is a "linear" template. But you should modify it from the default to let people know that they will receive a questionnaire to be filled out and returned.

The second template form you need to add to *listname*.MAILTPL is called ADDREQ1 and it can be as simple or as detailed as you want. All of the available template formatting commands can be used in ADDREQ1. For instance:

```
>>> ADDREQ1 &LISTNAME Membership Survey
.RE OWNERS
.TO &WHOM
.CE &LISTNAME Membership Survey
NOTE: Please make sure when you send this back that it goes to the
address &LISTNAME-Request@&MYHOST. Thanks.
```

This is a standard questionnaire required for all prospective subscribers to &LISTNAME. Blah blah blah...

In this case, you want the message to go to the subscriber, with a Reply-To: header pointing back to the (non-quiet) list owners. The first line indicating the return address is added for those users with mail clients that don't recognize Reply-To: headers.

You can also put a pre-formatted ADD job into the questionnaire to simplify your job when the questionnaire comes back. For instance,

```
.fo off
```

For List Owner's Use Only -- Be sure to include with your Reply

```
-----  
// JOB  
  ADD &LISTNAME &WHOM &USERNAME  
// EOJ  
-----
```

```
.fo on
```

For more detailed information on mail templates, see Section 18 [Working with Mail and Web Templates Using Email](#).

12.13.11 Peered Lists

This functionality is not available in LISTSERV Lite.

Occasionally the need to split a very large list may arise. This was more common when LISTSERV ran only on BITNET, whereas the TCP/IP version of LISTSERV is not limited by BITNET constraints. However, because of the fact that subscribers may be scattered all over the world, in rare cases it can make sense to split (or "peer") a list and share the mail load among two or more LISTSERV servers. Peering also makes it possible to have list archives located in more than one place; for example, a list might be peered between a European host and a North American host, making it possible for subscribers on each continent to retrieve archives from the nearer host.

Although there is no problem about peering to another L-Soft LISTSERV list, linking to a non-L-Soft mailing list manager is not supported and can and will cause serious problems (including mailing loops) for which L-Soft international, Inc. could not be held responsible.

12.13.11.1 Linking Two or More LISTSERV Mailing Lists

Please observe the following points:

- All lists should have a Peers= keyword setting that includes all of the other peers in the group as its arguments. For example, consider a peer group containing ListA, ListB, and ListC. ListA must have "Peers= ListB@its.host.com,ListC@its.host.com", whereas ListB must have "Peers= ListA@its.host.com,ListC@its.host.com" and finally ListC must have "Peers= ListA@its.host.com,ListB@its.host.com".

For lists running on LISTSERV for VM, setting the Peers= keyword makes it possible to EXPLODE them for better network efficiency. (Because peering is not widely used today, it is unlikely that the EXPLODE command will be ported to other platforms.)

- All lists must have the same list-level password, set with the PW= list header keyword. If this point is ignored, messages approved on one peer will not be accepted by the other peer and an error message will be generated, i.e.,

The approval request code received together with your posting for the MYLIST-L list is incorrect. For a peered list, this may be a normal condition. The approval protocol is not guaranteed to work among peer chains with pre-1.8b servers, and will also fail if the peers have a different password. For a non-peered list, the only likely explanation is a failure in the mail system or a recent change in mail system version or configuration. At any rate, please resubmit your message and go through the approval procedure a second time, and contact theLISTSERVadministratoriftheproblem persists.

----- Rejected message (73 lines) -----

This means that you must explicitly set the PW= list header keyword for each peer and not use the password LISTSERV generates automatically at list creation time. (This is the only case in which it is important and indeed required to manually set PW= for a list.)

- Each peer must be subscribed to at least one other peer, and the "real name" field for the subscription MUST be set to "Peer Distribution List".

12.13.11.2 Moving Users From One (Peer) Server to Another

You should be aware of the fact that a MOVE operation is not just an ADD to the new server and a DELETE to the current one. This would effectively transfer the person from the old server to the new one but his distribution options would be lost in the process. Besides, you should make sure that the user does not lose any mail in the process. The proper course of action to be taken when people are moved from one list to the other is the following:

- Send mail to the list telling people that a new peer server is being linked to the list, and that some subscribers will be moved to it.
- If the prerequisites for using the MOVE command are met, you should use either individual MOVE commands (in the case that there are very few users to move) or a batch-MOVE command with associated DDname (see the LISTJOB MEMO guide for more information on commands-jobs) to move the users. You may want to use the QUIET option to suppress notification if there are a lot of users to move.



Warning: The MOVE command should not be used to move peer list servers. See the MOVE command description for more details.

If you cannot use the MOVE command, you should try one of the following two methods:

- For each user to be moved, issue the following commands in the following order:
 - Query *listname* FOR *userid@host* (old server), write down the options.
 - QUIET ADD *listname* *userid@host* *full_name*
 - QUIET SET *listname* *options* FOR *userid@host*
 - Wait until you get confirmation for the two previous commands

- `QUIET DELETE listname userid@host` (old server)
- If there are a lot of users to move, the following method is preferred:
 - `GET listname` (old server)
 - `GET listname` (new server)
 - If you are using VM XEDIT, then receive both files and use the XEDIT "PUT" and "GET" commands to move users from one list to the other. You must preserve the contents of columns 81-100 across the move.
 - If you are using another text editor, then make sure that the editor you are using does not "imbed" control codes such as line breaks, tabs or word-wrapping characters into the text when you edit it. Use the cut and paste controls to copy lines in their entirety. You must preserve the contents of columns 81-100 across the move. Imbedded control codes and/or word wrap will generate errors when the list is stored back on the server.
 - Store the two lists back on their respective servers.

12.13.11.3 Special Commands For Peered Lists Only

ADDHere `listname userid@host <full_name> <PW=list_password>`

The ADDHERE command is strictly identical to ADD, with the exception that the placement of the user is not checked against the list of peer servers; in other words, the specified user is added to the local list without any further verification. (By comparison, the ADD command causes LISTSERV to check automatically to see if there is no better-suited peer list for the specified user.)

EXPLODE `listname <F=fformat> [VM only]`

The EXPLODE command provides a means whereby a list can be automatically analyzed by LISTSERV to optimize the placement of its recipients over the various peer servers hosting the list. It requires a "Peers=" keyword to be defined in the list header (see the [List Keyword Reference](#) document). Non-BITNET userids will be exploded according to the network address of the corresponding gateway (as per the SERVICE NAMES file), or ignored if the gateway could not be identified. LISTSERV will create a commands-job file containing the necessary MOVE command to transfer all the users which were found to be (possibly) mis-allocated to the peer server which is nearest to them. This file will then be sent to you so that you can review it before sending it back to the server for execution.

`MOVE listname userid@host <TO> newhost <PW=list_password>`
`DD=ddname listid@newhost [VM only]`

The MOVE command allows list owners to easily move users from one peer server to another. It will move the complete user entry from the source server to the destination one, including full name as it appears in the specified list and all list distribution options. The MOVE operation will be done in such a way that no mail can possibly be lost by the target while the MOVE operation is in progress (duplicate mail might be received for a short duration, however). Notification will be sent to the target user unless the QUIET option was used.

If the source and destination list names are identical, only the destination node ('newhost') needs be specified. Otherwise, the full network address ('listid@newhost') must be specified.

The `MOVE` command requires both source and destination lists to have the same password. Since each server will have to send a password to the other to validate the (special) `ADD/DELETE` commands it is sending to the other, it has potentially a way to trap the password specified by the server, thus thwarting any attempt at inventing a protocol to allow use of this command on lists which have a different password. Besides, no `MOVE` operation will be accepted on lists which do not have a password at all, because for technical reasons it would allow unauthorized users to easily add someone to a list (since there would be no password validation).

The `MOVE` command is the proper way to effect a move operation. You should not use any other command/set of commands unless you cannot use `MOVE`. **THE MOVE COMMAND SHOULD NOT BE USED TO MOVE DISTRIBUTION LISTS!!!** Since a `MOVE` is basically an `ADD + DELETE`, with the latter being done only **AFTER** the `ADD` is completed, moving a distribution list address with the `MOVE` command can cause a duplicate link to be defined for a short period of time. This could result in a transient mailing loop, which could become permanent if the size of the looping mailfiles is less than the size of the inter-servers "DELETE" command jobfile, and the RSCS priority of the latter has been altered.

12.13.12 Super-Lists and Sub-Lists

This functionality is not available in LISTSERV Lite.

It is possible to define a "super-list" (as in opposite of sub-list), that is a "container" list that includes all the subscribers in a predefined set of sub-lists. This can be done recursively to any depth. Site maintainers can create super-lists consisting of any of the lists on the server, regardless of who owns them. A non-maintainer list owner may convert a standard list that they own into a super-list, but they may only add other lists that they own to the `Sub-Lists=` setting. Attempts by non-maintainer list owners to add non-owned lists to their super-list will result in an error when the list configuration is stored, and the configuration will be left unchanged.

The value is a comma separated list of all the sub-lists, which must all be on the same (local) machine. For instance:

```
* Sub-lists= MYLIST-L,MYOTHERLIST-L
```

The default value for this keyword is null, that is, to have no sublists. In addition, the super-list and all of its sublists must reside on the same LISTSERV server.

The only difference between a normal list and a super-list is what happens when you post to it. With the super-list, the membership of all the sub-lists is added (recursively) and duplicates are suppressed. Other than that, the super-list is a normal list with its own archives, access control, etc. You can even subscribe to it, and this is actually an important aspect of the operation of super-lists. If you are subscribed to the super-list itself, the subscription options used to deliver super-messages to you are taken from your subscription to the super-list, just like with any other list. All combinations are allowed, and in particular `NOMAIL` is allowed, meaning you don't want to get messages posted to the super-list. When you are subscribed to multiple sub-lists, on the other hand, things work differently:

- NOMAIL subscriptions are ignored. You will get the super-message if you have an active (not NOMAIL) subscription to at least one sub-list. The idea is that the super-message must be equivalent to posting to all the sub-lists, without the duplicates. Since all it takes to get a message posted to all the sub-lists is a single non-NOMAIL subscription, this is how the super-list works. The only way not to get the super-messages is to subscribe to the super-list directly and set yourself to NOMAIL.
- The DIGEST and INDEX options are ignored and internally converted to MAIL. The first reason is that, since in most cases the user will be on multiple sub-lists (otherwise you don't need a super-list in the first place), the only safe method to set subscription options for super-messages is by subscribing to the super-list so that there is no ambiguity. The second reason is that, in most cases, super-lists will be used for out of band administrative messages rather than for large volume discussions, so it is actually preferable to have the message sent directly. The third reason is that the super-list and sub-lists may not necessarily offer the same options (DIGEST and INDEX). In particular it is expected that many super-lists will not have archives. If you want a DIGEST or INDEX for the super-messages, you must subscribe to the super-list directly.

Topics, if defined, are evaluated on a per-list basis. That is, for every sub-list (and for the super-list), LISTSERV determines whether the topic of the message is one that you want to see. If not, it acts as if you were not subscribed to this particular list. Roughly speaking, this works very well if all the sub-lists have the same set of topics (or a well-defined set of common topics), and doesn't work well at all if every list has its own set of topics.

Postings to a super-list are always archived in the super-list's notebooks (if enabled), and never in the notebooks of the sub-lists. This is because by its nature a posting to the super-list is not equivalent to cross-posting a message to all of the sub-lists. Rather, LISTSERV recurses into the sub-lists and generates an "on the fly" listing of all of the users on the super-list and the sub-lists (this is how it avoids duplicates, among other things) and then treats this "on the fly" listing as if it were the subscriber list of the super-list itself. You will note that a super-list posting is always identified as coming from the super-list, regardless of whether a given user is subscribed to the super-list or to one or more of the sub-lists.



Note: A REVIEW command sent for the super-list will not recurse into the sub-lists pointed to by the super-list. If you have a super-list called SUPER and you send a REVIEW SUPER command, LISTSERV will respond with only the people who are subscribed directly to SUPER. The only way to find out what users are covered by the super-list is to send REVIEW commands for the super-list and all of its sub-lists.

The REPRO subscriber option is recursive and will be honored for users who are subscribed to the sub-lists.

Access to the super-list's notebook archives is not automatically recursive. If you want sub-list subscribers to be able to access the archives of the super-list (but don't want the sub-list subscribers to have to subscribe to the super-list), then you must configure the Notebook= keyword for the super-list so that it contains references to each of the sublists. For example, say we have a super-list called SUPER and two sub-lists called SUB-A and SUB-B. We want the subscribers of both SUB-A and SUB-B to be able to read the archives of SUPER (since postings to SUPER won't be archived in SUB-A or SUB-B),

but we don't want people who aren't subscribed to any of the three lists to be able to access the archives. So we set

```
* Notebook= Yes,C:\LISTS\SUPER,Monthly,Private,(SUB-A),(SUB-B)
```

and anyone subscribed to the SUPER list or to the SUB-A or SUB-B lists can access the SUPER archives.

If you have many sub-lists, you can specify multiple Notebook= lines, for example,

```
* Notebook= Yes,C:\LISTS\SUPER,Monthly,Private,(SUB-A),(SUB-B)
```

```
* Notebook= (SUB-C),(SUB-D),(SUB-E),(SUB-F)
```

LISTSERV will read these two (or more) Notebook= lines and concatenate the values.

12.13.13 Cloning Lists

Some sites may have a need for many lists that are essentially identical. For instance, a series of class section lists for a university department may have the same owner, allow the same class of users to subscribe, and so forth. LISTSERV makes it possible to maintain large collections of lists by "including" keywords from an external file.

For instance, consider a mathematics course with ten sections. Each section should have its own list (for instance, called M101-001, M101-002, and so forth), but the lists will otherwise be identical. The LISTSERV maintainer simply creates a text file (in this case called M101 KEYWORDS) containing the keyword definitions that will be shared by the lists, as follows:

```
PUT M101 KEYWORDS PW=createpw
* Owner= mathwhiz@someuni.edu (Professor J. Random User)
* Owner= Quiet:
* Owner= gradasst@someuni.edu (Joe Doakes, Graduate Assistant)
* Notebook= Yes,/home/listserv/archives/m101,Monthly,Private
* Auto-Delete= No
* Errors-To= gradasst@someuni.edu
* Subscription= Closed
* Notify= Yes          Confidential= Yes          Validate= Yes,Confirm,NoPW
* Reply-to= List,Ignore  Review= Owners          Send= Private
* Default-Options= Repro
```

Next, the LISTSERV maintainer stores this file in the usual way, by first making a filelist or catalog entry for it (as outlined in Section 16 [Managing Archives using Email](#)) and then storing it with a PUT operation. Generally the GET and PUT FACs for this file should specify that the list owner(s) should be able to retrieve and store it. The file must be stored in LISTSERV's A directory (the same directory that contains the *.LIST files).



Note: It is also possible to create this file directly in LISTSERV's A directory with a text editor; if you do so, make sure that you do not include the PUT command shown above. You should still make the filelist or catalog entry for the file so that the list owners can retrieve and store it.

Next, the LISTSERV maintainer creates and stores a skeleton list header for each of the section lists. The first section list (M101-001) is illustrated below:

```
PUT M101-001 LIST PW=createpw
* Math 101 Section 001 Mailing List
* .IK M101
```

The `.IK` command tells LISTSERV that whenever it uses this list, it should read the keyword definitions from the file `M101 KEYWORDS` (note carefully that the syntax is `".IK M101"`, not `".IK M101 KEYWORDS"`). Now, whenever the professor in charge of the class wants to make a change to all of the M101 lists (for instance, he has a new graduate assistant), he simply `GETs` the file `M101 KEYWORDS`, makes the changes, and `PUTs` the file back, instead of having to `GET` separate headers for each list and make the changes to all of them individually.



Notes: On some servers it may be necessary to stop and restart LISTSERV (or do a `GET+PUT` of all of the list headers involved) to make changes to the `KEYWORDS` file appear. This is because LISTSERV may have the `KEYWORDS` file and/or the list headers that use it cached at the time you modify it.

In order to see the complete list header, send a `REVIEW listname` command. The response to a `GET` will be only the skeleton header with the `.IK` command. If `GET` did not work this way, you would not be able to change or remove the `.IK` command line once you set it.

The sample `KEYWORDS` file above includes a `Notebook=` keyword. This will cause the notelogs for all of the lists that use this `KEYWORDS` file to be written in the same directory, per the example, `/home/listserv/archives/m101`. This means that in that directory you would have notelogs for the M101-001 list, the M101-002 list, and so forth (depending of course on what lists use the example M101 `KEYWORDS` file). If this behavior is not desired, simply don't put a `Notebook=` keyword in the `KEYWORDS` file, and define it in the list header for the cloned list instead, either before or after the `.IK` directive.

For the web archive interface, note carefully that if you do use the same directory for all of the cloned lists' notelogs, you will still have to make separate web archive directories for each list under your `WWW_ARCHIVE_DIR` directory if you intend to serve the archives via the web interface. In other words, the web interface doesn't care where you keep a list's notelogs as long as it has a directory specified under `WWW_ARCHIVE_DIR` for it to write the list's web archive indexes into. So while all of your notelogs may go into `/home/listserv/archives/m101`, regardless of the name of the cloned list, you still need to make (for example) `/usr/local/etc/httpd/htfiles/archives/m101-001` and so forth in order to serve the notelogs on the web.

12.14 List Passwords are now Obsolete

When creating the list, a random password is assigned for security if the LISTSERV maintainer does not define one explicitly. List passwords are no longer necessary in all but one special situation; they are simply another line of defense, and it is much more

secure to allow LISTSERV to randomly assign the list password and for you to use a personal password to validate your LISTSERV commands. See Section 12.9 [Defining List Owners](#) to learn how to create a personal password.

The only situation in which a list password MUST be defined explicitly in a list header is in the case of peered lists, where the PW= list header keyword must be set to the same value on all peers.

12.15 Allowing/Blocking MIME Attachments

LISTSERV includes a MIME attachment-filtering feature which is configured at the list level by setting the Attachments= list header keyword. The new keyword allows three distinct modes:

- Allow all MIME attachments, no filtering or blocking
- Reject MIME attachments with notice to the poster
- Filter MIME attachments out of messages transparently

In addition, you can configure specific MIME types to reject or filter while allowing other types through (for instance, you can block executable files but allow images or word processing files based on their MIME type).

For information on the various settings, please see the section on the Attachments= keyword in the [List Keyword Reference](#) document.

12.16 Content Filtering

This feature is not available in LISTSERV Lite.

This feature is intended primarily to filter out-of-office messages and the like. It is not intended as a profanity filter. Attempts to configure it to filter profanity will most likely prove to be futile in the long run and are not recommended by L-Soft.

The CONTENT_FILTER mail template form, if present, contains filtering rules, one rule per line, empty lines ignored. Each rule has the following format:

```
[prefix:] pattern
```

The prefix, if present, can be a mail header tag (e.g. "Subject:"); "Header:" to check the whole header; or "Text:" to search the message text. The latter is the default if no prefix is supplied, it is provided in case the pattern contains a colon in the first word. If there are multiple mail header tags with the specified name (e.g. "Received:"), each such tag is searched and it is enough for one of them to match the pattern. If the requested tag is not present in the header, there is (surprise!) no match. A text search will search every line of the first text/plain part in the message. If there is no text/plain part, there is no match. Again, this is designed to filter read receipts, loops, chain letters, spam, you name it. There was no attempt on the developers' part to make this a profanity filter, and future versions will not be "enhanced" to make futile attempts at (for instance) decoding Word documents to look for obscene words.

Regular comparisons such as those described above are not case sensitive. Patterns are standard LISTSERV patterns, that is, the asterisk is the wildcard character. If there is no asterisk in the pattern, it is replaced with "*pattern*" much like the SCAN command.



Documented Restriction: You cannot match literal asterisk characters in a string as there is no way to escape them. Any asterisk in a pattern will always be evaluated as a wildcard.

The content filter also supports "exact match" comparisons, which are triggered by a double colon. For instance:

Subject::

There are two significant differences between exact and regular match:

- You must supply your own wildcard characters in an exact match (if you want to use wildcards, that is). A regular match will insert leading and trailing wildcards if none are found. Thus, an exact match is the only way to make a comparison without wildcards.
- You can make an exact match for the empty string. Empty regular matches are ignored since they map to a wildcard comparison for **, which would be always true. This also makes it possible to apply an exact match to a message that does not contain a specified header. For instance, if you want all messages to contain a (mythical) KABOOM: RFC822 header, with an exact match you can tell LISTSERV to perform one of the content-filtering actions if the the header is not present. This is not possible with a regular match.



Note: You cannot differentiate a header with an empty KABOOM field from a header with no KABOOM field.

One of the most handy uses for the exact match syntax is to be able to write a rule to reject messages with blank subject lines. For instance:

Subject::

Action: REJECT Please resubmit your message with a non-blank subject.

Every rule can, optionally, be followed by an action rule. This has the following format:

Action: ALLOW

Action: REJECT reason

Action: DISCARD comment

Action: MODERATE

(The available actions are the same for both regular and exact comparisons.) For instance,

>>> CONTENT_FILTER

Subject: Out of office

Action: REJECT OOO messages are not allowed on this list.

Subject: Auto-Generated:

Action: REJECT

Text: Click here to be removed

Action: REJECT Buzz off, spammer.

Subject::

Action: REJECT Please resubmit with a non-blank subject.

Subject: copyright

Action: MODERATE

```
To: friend@public.com
Action: DISCARD This guy is a spammer
```

The default is "Action: REJECT" with no specified reason. REJECT means that the message is rejected. MODERATE means that the message is to be forwarded to the list editor to be manually approved or rejected. DISCARD means that the message is to be dropped on the floor without further processing; any text following DISCARD is echoed to the LISTSERV console (and is thus logged).

ALLOW means that the message is allowed and all remaining rules are ignored. This could be used in moderated lists to allow the list moderator to bypass certain filters, for instance:

```
>>> CONTENT_FILTER
Subject: Out of office
Action: REJECT OOO messages are not allowed on this list.
From: JOE@EXAMPLE.COM
Action: ALLOW
Text: Click here to be removed
Action: REJECT Buzz off, spammer.
```

In the example above, messages with Subject: lines containing "Out of office" are rejected. Messages containing the text "Click here to be removed" are also rejected UNLESS they come from joe@example.com.

The text of the rejection is fetched from the BAD_CONTENT mail template form, with the reason supplied as a variable called &COMMENT. The rejection message looks like this:

```
Date: Tue, 4 Dec 2001 22:03:42 -0500
From: "L-Soft list server at LISTSERV.EXAMPLE.COM (1.8e)"
      <LISTSERV@LISTSERV.EXAMPLE.COM>
Subject: Rejected posting to TEST@LISTSERV.EXAMPLE.COM
To: Joe User <joe@EXAMPLE.COM>
```

Your posting to the TEST list has been rejected by the content filter.
OOO messages are not allowed on this list.

followed by the text of the posting including all mail headers. (In this case the body of the message contained the text "out of office" and the rule above was applied.)

A default site-wide CONTENT_FILTER template form may be defined in \$SITE\$.MAILTPL for use by lists whose owners do not prefer to provide their own custom versions in their listname.MAILTPL files.

12.17 DomainKeys Message Signing

This feature is not available in LISTSERV Lite.

DomainKeys message signing is available to sites running LISTSERV Classic or LISTSERV Classic HPO. Current LISTSERV maintenance is also required. For more information on how to configure LISTSERV for DomainKeys support, please contact your site administrator.

Assuming that it is available for your use, DomainKeys support for lists is enabled by default. This means that all list postings and administrative messages related to a list will be signed to assert that they actually originated from your LISTSERV server.

If for some reason you wish to disable DomainKeys message signing for a given list, you can do so by adding

* Misc-Options: NO_DKIM_SIGNATURE

to your list header. Or, if you prefer to disable it server-wide by default, you can add NO_DKIM_SIGNATURE to the DEFAULT_MISC_OPTIONS site configuration variable setting.

Incoming DomainKeys or DKIM signatures submitted to a mailing list will be removed unless "Misc-Options= KEEP_DKIM_SIGNATURE" is set in the list configuration. This is necessary because these signatures almost never match after the message has been processed. The worst thing that could possibly happen to your deliverability is a DomainKeys signature that does not match and causes the message to be flagged as suspicious.

The KEEP_DKIM_SIGNATURE option is experimental and not meant for general use. As DomainKeys is specified today, signatures DO NOT survive posting to mailing lists (LISTSERV or otherwise), so LISTSERV removes them by default to avoid triggering alerts for subscribers on systems that have implemented the client side of DomainKeys. The DKIM specification may be more robust in this respect, but even DKIM signatures will probably not survive when posted through a mailing list. Use the KEEP_DKIM_SIGNATURE option at your own risk.

Section 13 Managing Subscriptions through Email

13.1 Adding and Deleting Subscribers to/from a List

A list owner may add and delete subscribers manually. The command syntax is:

```
ADD listname netaddress full_name
DELEte listname netaddress
```

In a perfect world, subscribers would understand intuitively how to subscribe and unsubscribe from mailing lists. Unfortunately, this is not always the case. Depending on an individual's style of list management, a list owner may choose to add or delete subscribers to the list manually, or send the potential subscriber instructions on how it is done. (See Appendix B: [Sample Boilerplate Files](#) for sample "boilerplate" instruction files that can be modified to suit local purposes.) And for lists coded `Subscription=By_Owner` or `Subscription= Closed`, it is of course necessary to use the `ADD` command to subscribe a user.

If the list is set to confirm mailing paths for new subscriptions (`Subscription=Open,Confirm`), it is probably wisest to use the latter option, since if a subscriber is added manually to a list, the confirmation process is bypassed.

Full_name should contain at least two discrete words, but it is also possible to add users without knowing the value for *full_name*. Simply use an asterisk ("`*`") character. If the user is already subscribed to another list on the same host, `LISTSERV` will pick up the value for *full_name* from its signup files. Examples are:

```
RIGHT: ADD DOAKES-L joe@example.com Joe Doakes
```

```
RIGHT: ADD DOAKES-L joe@example.com *
```

```
WRONG: ADD DOAKES-L joe@example.com Joe
```

```
WRONG: ADD DOAKES-L joe@example.com Joe-Doakes
```

When adding users, `ADD` will also accept a full RFC822 address that you can cut and paste from the "From:" line of a message. Be sure that you remove the "From:" part of the line. For example, the "From:" line

```
From: Joe User <JoeUser@example.com>
```

becomes an `ADD` command as follows:

```
ADD MYLIST-L Joe User <JoeUser@example.com>
```

13.1.1 Adding Users Whose Address and Read Name Exceed 80 Characters

Although this is not as much of a problem as it used to be, it can happen with any system which allows users to have a very long "local part" (i.e., the part to the left of the "`@`") in their userid, or with users on systems that just have very long names, such as some of the hosts in the .US domain generally have. For instance, you might try to send the following `ADD` to `LISTSERV`:


```
QUIET ADD MYLIST someone.with.a.real.long.userid.that.wraps@hishost.com
His Name
```

"His Name" wraps to the next line. If you send this to LISTSERV, LISTSERV treats the two lines as separate commands even though you did not hit RETURN after the user's address, and it responds:

```
> QUIET ADD MYLIST someone.with.a.real.long.use-
rid.that.wraps@hishost.com someone.with.a.real.long.use-
rid.that.wraps@HISHOST.COM is not yet in the signup file. Please
specify the full name of that person, as in "ADD MYLIST JOE@XYZ.EDU Joe
H. Smith".
> His Name
Unknown command - "HIS". Try HELP.
```

To avoid this problem, set up your ADD command with a "continuation card" as follows:

```
// QUIET ADD MYLIST someone.with.a.real.long.userid.that.wraps@hishost.com,
His Name
```

Typically this was a problem primarily with the X.400 and X.500 addressing schemes, which are not as prevalent as they used to be.

13.1.2 X.400 and X.500 Addressing – Special Problems

X.400 and X.500 addressing schemes can cause problems for the list owner who is trying to add or delete one. These addressing schemes use the "/" character to separate address elements, but to LISTSERV, "/" is a special character and you would not be able to add or delete one of these addresses by simply cutting and pasting it into an ADD or DELETE command.

For instance, you might have an address like:

```
/G=Joe/S=Randomuser/OU=403402ABD/O=SOME.CORP/@LANGATE.SOME.HOST.COM
```

In order to either add or delete this address, there are two issues:

- The address may wrap to the next line once you add the DELETE listname command, and LISTSERV will not accept it.
- The address contains characters that LISTSERV will reject as illegal (the "/" character).

For adds, to get around both of these issues, you must use a LISTSERV JOB syntax as follows:

```
ADD MYLIST-L DD=X500 PW=MYPASSWORD
//X500 DD *
/G=Joe/S=Randomuser/OU=403402ABD/O=SOME.CORP/@LANGATE.SOME.HOST.COM
/*
```

Any other method will trigger an RFC822 parser error because of the "/" characters in the address. (A user who is subscribing from an address like this will have no trouble; it is only when the list owner uses the ADD command that this difficulty surfaces.)

For deletions, to get around both of these issues, the wildcard character ("*") can be used. You may not need the entire address in order to delete it, so you might just use

```
DELETE MYLIST *G=JOE*S=RANDOMUSER*@LANGATE.SOME.HOST.COM
```

which solves both the line wrap problem and the illegal character problem at the same time.

You can also use double-quotes around the address if it contains illegal characters, and a "continuation card" (see Section 13.1.3 [Continuation Card Syntax](#)) if the address is too long to fit on one line:

```
// DELETE MYLIST,
"/G=Joe/S=Randomuser/OU=403402ABD/O=SOME.CORP/@LANGATE.SOME.HOST.COM"
```

13.1.3 Continuation Card Syntax

The basic syntax of a continuation card is

```
//<space><beginning of command><space><comma>
<continuation of command><space><comma>
<continuation of command>....
```

for example,

```
// QUIET ADD MYLIST someone.with.a.real.long.userid.that.wraps@hishost.com,
His Name PW=mypassword
```

or, for instance, for a large GETPOST job,

```
// GETPOST MYLIST 10769-10770 10772 11079 11086 11095 11099-11100 11104,
11111 11115 11118 11121 11124 11131 11144 11147 11153 11158 11166 11168
```

Without going into a lot of detail, the "//<space>" at the beginning of the command causes LISTSERV to look for a comma at the end of the first line and, if it finds the comma, to add anything following the comma on the second line to the end of the first line. Be sure to put a space before the comma at the end of the first line, as LISTSERV will not add the space for you.

For more information, see the section on LISTSERV's Command Jobs Language Interface (CJLI) in the [Advanced Topics Manual](#) for LISTSERV.

13.2 Finding Users Who Do Not Appear in the List

Sometimes the list owner will get a message from a subscriber who says, in essence, "I keep trying to (unsubscribe/change to digest/etc.) and LISTSERV says I'm not subscribed. Can you help?" This requires some detective work.

There are a couple of strategies for figuring out what is wrong. List owners should first use the powerful SCAN command to search for a pattern anywhere in the subscriber list. The syntax is:

```
SCAN listname search-text
```

For instance, "SCAN TEST-L Nathan" might return:

```
> scan test-l Nathan
Nathan Brindle <nbrindle@INDYCMS.IUPUI.EDU>
Somebody Else <nathan@BAZ.NET>
Jonathan Smith <jsmith@FOO.BAR.COM>
SCAN: 3 matches.
```



Note: SCAN is not case-sensitive. "Nathan", "NATHAN", and "nathan" all return the same results.

Searches with SCAN should start out simple and become more complex as needed. For instance, if there are only three people in the list with the string "NATHAN" as part of their subscription record, it will be unlikely that you will need to make the search any more

complex. If you are looking for "SMITH", however, it may be necessary to further qualify your search string, say to look for "JOE SMITH". Another reason it is important to begin with a simple search string is that your user may not be subscribed under the exact address the error is returning to you. For instance, say you don't have the user's id, but you have a host name. You can search for all occurrences of the host name, but note that the search:

```
SCAN TEST-L MAIL.FOO.BAR.COM
```

will not find the user `jsmith@foo.bar.com`. If you run the following search:

```
SCAN TEST-L BAR.COM
```

however, you will find Mr. Smith's subscription.

Another possibility is that the subscriber may be using more than one address to work with his subscription. For instance, say the user's complaint to you came from `JOE@SUN6.SOMEUNI.EDU`. Looking at the list, you find a subscription for `JOE@SUN8.SOMEUNI.EDU`. LISTSERV has no way to know that `JOE@SUN6` is the same person as `JOE@SUN8`, even though Joe and you know they are. The solution to Joe's problem above is for you to delete his SUN8 subscription and add his SUN6 address. Then Joe needs to be sure that he uses SUN6 in the future, if not for reading mail, then at least for managing his own subscription.

Another strategy would be to submit a wildcard QUERY to the list. The drawback to this method is that it might require multiple tries to find the subscription, depending on the complexity of the wildcard query.



Note: Not only can this sort of problem arise from a subscriber using more than one workstation to read mail, but it can also arise when a particular site changes its domain configuration, forwards mail from the old addressing scheme to the new addressing scheme, and doesn't inform its users of the change. In these cases, users often don't realize there is a problem until they try to unsubscribe or change personal options, because the change has been transparent to them.

13.3 Converting Existing Lists from Other Systems to LISTSERV

13.3.1 Converting Mailing Lists

Currently there are no supported conversion programs that will take (for instance) a Majordomo or ListProc mailing list and convert it to LISTSERV format. However it should be possible to extract the address list from the non-LISTSERV list and use a bulk add operation (see Section 13.4 [Adding Subscribers to Lists in Bulk](#)) to populate your new LISTSERV list.

13.3.2 Converting Message Archives

Existing list archive notebooks will probably not be in LISTSERV format (a modified VM MAILBOOK format), but rather, in the standard unix mailbox format. Again there are no supported programs to convert such archives to the LISTSERV format, but the basic format is as follows:

Message separator
Body of Message

Message separator**Body of Message**

The MAILBOOK message separator is a line of 73 "=" characters (ASCII &H3D). Each archive notebook file must start with a message separator as the first line, e.g.:

```
=====
Date:      Tue, 3 Mar 1998 10:36:55 -0500
Sender:    Test list <TEST@LISTSERV.EXAMPLE.COM>
From:      Nathan Brindle <nathan@example.com>
Subject:    Test
Mime-Version: 1.0
Content-Type: text/plain; charset="us-ascii"
```

First test message

```
=====
Date:      Tue, 3 Mar 1998 10:39:11 -0500
Sender:    Test list <TEST@LISTSERV.EXAMPLE.COM>
From:      Nathan Brindle <nathan@example.com>
Subject:    Test2
Mime-Version: 1.0
Content-Type: text/plain; charset="us-ascii"
```

Second test message

and the notebooks must be named in a standard LISTSERV format, e.g., TEST.LOG9803A, so that LISTSERV will see them and include them in the output of the INDEX listname command.



Note: For unix mailbox-formatted archives, you must remove the first line of each message, which begins with "From" followed by a space (this is the standard unix mailbox delimiter). Below is an excerpt from a unix mailbox for illustration purposes:

```
From owner-test@listserv.example.com Mon Dec 29 15:17:20 1997
Return-Path: <root@listserv.example.com>
Received: from localhost (root@localhost) by listserv.example.com
(8.8.3/8.8.3) 0
Date: Mon, 29 Dec 1997 15:17:20 -0500 (EST)
From: root <root@listserv.example.com>
To: test@listserv.example.com
Subject: Test
Message-ID: <Pine.LNX.3.95.971229151703.711A-100000@listserv.example.com>
MIME-Version: 1.0
Content-Type: TEXT/PLAIN; charset=US-ASCII
Status: RO
X-Status:
```

This is a sample message from Majordomo in unix mailbox format.

Root

```
From owner-test@listserv.example.com Mon Dec 29 15:23:51 1997
Return-Path: <root@listserv.example.com>
Received: from localhost (root@localhost) by listserv.example.com
(8.8.3/8.8.3) 0
Date: Mon, 29 Dec 1997 15:23:51 -0500 (EST)
....
```

Each of the lines beginning with From owner-test@listserv.example.com is the delimiter separating one message from the next.

13.4 Adding Subscribers to Lists in Bulk

If you are moving a list from a non-LISTSERV site, you can quickly and easily convert the existing subscriber list to the LISTSERV format by following these instructions:

1. Have the LISTSERV maintainer at your new site create the new list header and install it on the machine.
2. Create an add job as follows. The QUIET and IMPORT command words are optional; omit the square brackets if you use them. The "full name" field is optional as long as you use the IMPORT option; otherwise you must either specify "*" (for an anonymous subscription) or a full name consisting of at least two separate words.

```
[QUIET] ADD listname DD=ddname [IMPORT] PW=yourpassword
//ddname DD *
userid1@host1.com [*|full name]
userid2@host2.com [*|full name]
...more users, one per line...
useridn@hostn.com [*|full name]
/*
```

For example (what fun:),

```
ADD B5-L DD=MYDD IMPORT PW=BLAHBLAH
//MYDD DD *
SHERIDAN@BABYLON5.MIL John Sheridan
IVANOVA@BABYLON5.MIL Susan Ivanova
LONDO@CENTAURI.PRIME.GOV Londo Mollari
GKAR@KAARI.NARN.GOV Citizen G'Kar
/*
```

If you are importing from an existing non-LISTSERV list, you should remove any lines from the original list that do not actually identify subscriber addresses. If you are converting to LISTSERV from ListProc, note that LISTSERV will not convert ListProc user options to their LISTSERV equivalents; you must take a line like

```
user1@somehost.com POSTPONE NEWLIST NO user's name
```

and reduce it at least to

```
user1@somehost.com user's name
```

Otherwise, the ListProc options will become part of the full name field.

3. Send the job to LISTSERV.

The IMPORT option implies a QUIET ADD (in other words, you do not need to specify QUIET if you use IMPORT) and otherwise vastly speeds up the ADD process by loosening syntax checking and omitting success messages. If you do not use the IMPORT option and do not specify QUIET, the users you bulk add will receive the normal SIGNUP message and/or WELCOME file as usual.

13.5 Deleting Subscribers from Lists in Bulk

If you have a large number of users to delete at one time, you can use a bulk delete syntax that is similar to the bulk ADD documented above. However please note that there

is no "IMPORT"-type option for this feature, and as usual for the `DELETE` command you specify only the user's address in the data DD.

There is, however, a `BRIEF` option that can be specified which is good when you don't want a long list of "userid@host has been deleted from list xxxx" messages, one for each user deleted. Use of the `BRIEF` option tells `LISTSERV` to return only a count of the users that were deleted.

Once again you construct a `LISTSERV JOB` framework as follows and then send it to `LISTSERV`:

```
[QUIET] DELeTe listname DD=ddname [BRIEF] PW=yourpassword
//ddname DD *
userid1@host1.com
userid2@host2.com
...
useridn@hostn.com
/*
```

You will probably want to use the `QUIET` modifier when doing a bulk delete, in order to suppress the notification message to the users being deleted.

13.6 Using the QUIET Option with Commands

Prepending the command word "QUIET" before any `LISTSERV` command that you issue on behalf of a subscriber causes `LISTSERV` to suppress any notification to the subscriber of the changes you have made. This is particularly helpful when deleting subscribers whose accounts have expired and when setting subscribers with full mailboxes to `NOMAIL`, as it will help avoid another error message from the host when the notification message bounces. It is also helpful when adding subscriptions to the list that should not receive any welcome mail, such as redistribution lists and `USENET` newsgroups.

Examples of the usage of `QUIET` include:

```
QUIET ADD EXCEL-L comp.spreadsheets.excel@netnews.somenode.edu
QUIET DELETE EXCEL-L Bouncemeister@somenode.edu
```

13.7 Dealing with Bounced Mail

13.7.1 What is a bounce, and what can typically cause one?

A bounce is simply an undeliverable e-mail message. The term "bounce" is used to describe it because normally the system that discovers the delivery error "bounces" a copy of the message back to you with some sort of delivery error message. Sometimes these messages are easy to decipher – "No such user at foo.bar.com" – but uncomfortably often they are not that easy. Certain systems, as noted above, kindly format error notifications in a format that `LISTSERV` can understand, and if your list is configured for auto-deletion, these bounces will be the least of your worries – in fact, they will not be worrisome at all.

13.7.2 The Owner-Listname Address

If you receive bounces processed through LISTSERV you will note that they normally say something like the following at the top:

```
The enclosed message has been identified as a delivery error for the MYLIST-L
list because it was sent to 'owner-mylist-l@LISTSERV.EXAMPLE.COM'.
```

```
----- Message in error -----
```

What this message means is simply that LISTSERV has received mail sent to the *owner-listname* mailbox for your list. Mail sent to this special address is automatically forwarded by LISTSERV to the address(es) you have defined in the `Errors-To=` list header keyword. The little "error-header" shown above is prepended to the actual error message to let you know that this is an error for your list (rather than unceremoniously dumping it into your mailbox and making you wonder "why did I get this?", since some delivery errors aren't specific about what list or even what user they are for). So whenever you get mail saying it was found in the *owner-listname* mailbox, it means that it is an error that you need to deal with for the list referred to by *listname*.

If you find that you have users trying to contact you (as list owner) at the *owner-listname* address, you should tell them that the correct generic address for contacting the list owner(s) is *listname-request*, not *owner-listname*. Mail sent to the *listname-request* address will be sent to all non-quiet list owners and furthermore will be automatically responded to with the `REQACK1` mail template form from your *listname*.MAILTPL file (or its default from `DEFAULT.MAILTPL`; see Section 18 [Working with Mail and Web Templates Using Email](#)) while mail to *owner-listname* will not be responded to at all unless you do so explicitly. The nice thing about having people use the *listname-request* address is that you can store your list's FAQ (if you have one) in the `REQACK1` mail template form and probably not have to answer all of the questions you get as list owner--like "how do I subscribe?" and "how do I sign off?".

13.7.3 What to do about several types of bounces



Note: It is not the intent of (nor would it be reasonably possible for) this manual to document each and every kind of delivery error that you may ever see as a list owner. Unfortunately, and completely outside the control of anyone at L-Soft, new types of cryptic, difficult to understand, and totally misleading errors appear all the time. If you run across something not otherwise documented here, the best place to ask for help is the LSTOWN-L mailing list (see Section 19.6 [If I can't find the answer, where do I turn?](#)).

That being said, here are a few of the typical mail errors you will have to deal with as a list owner. Newer, so-called "Notary" format error codes are documented in RFC1893, which can be found at the WWW URL

<http://www.ietf.org/rfc/rfc1893.txt?number=1893>

- No such user at host, user unknown (or "notary" format error number 5.1.1)

Most of the time, this is authoritative and indicates that the user's access has been curtailed for some reason (graduation, no longer employed, etc.). A quiet delete (syntax: "QUIET DELETE listname userid@host") is in order unless you have reason to believe that the message is not authoritative. Variations on this message include "Recipient unknown" and "Ambiguous address: userid". The latter doesn't

really mean the user doesn't exist, but it's almost as bad, and many list owners choose to classify it as "no such user".

Microsoft Exchange servers send back the following message for an unknown user:

```
Joe@EXCHANGE.EXAMPLE.COM on Wed, 4 Mar 1998 13:31:50 -0600
    The recipient name is not recognized
    MSEXCH:IMS:Example Corp:EXAMPLE:EXCOMEXCH 0 (000C05A6)
Unknown Recipient
```

- No such host, host unknown (or "notary" format error number 5.1.2)

This is sometimes authoritative and sometimes not. If a host goes down or a gateway fails, often this message is returned by an intermediate host or gateway. If the user is bouncing a great deal of mail from a high-volume list, it is probably best to set the user to NOMAIL (syntax: "SET *listname* NOMAIL FOR *userid@host*") rather than to summarily delete him. This way, the error messages stop, the user is sent an automatic message telling him his personal options have been changed by the list owner, and the user doesn't have to go through the subscription process again if the problem has been solved in the interim.

The problem is that some hosts go down on a regular basis and this error makes it impossible to tell if the host in question is gone forever or gone until the local sysadmin reboots his machine. After a while, you will begin to recognize the transient hosts and may elect to ignore them. If you choose to set the user to NOMAIL, you should send a message to the user just in case the system has come back up, and you should keep some sort of record of the users you've set this way so you can follow up later with another message.

- No MX or A records for host (or "notary" format error 5.4.4)

Similar to "no such host". This means that the Domain Name Service (DNS) can't find any routing information for *host* but has found at least one reference to it. This generally indicates a DNS configuration error and may or may not be transient.

- Transient failure: cannot deliver for n days

A host is experiencing periodic failures, and the gateway or intermediate host has not been able to deliver the message for n days. Usually the host will attempt redelivery. Usually there is nothing wrong with the user address, so it is a list owner decision as to whether it is worth waiting out the transient failure or going ahead and setting the user to NOMAIL. Unfortunately, by the time you get this message, the failure is n days in the past, the "transient failure" is very probably over, and you are likely to receive further error messages for n more days until the intermediate host's queue is exhausted.

- Mailbox full, quota exceeded (or "notary" format error number 4.2.2)

This usually happens on systems with tiny user mailbox space, but it can happen on any system if a user subscribes to too many lists or goes on an extended vacation without setting lists to NOMAIL. The best solution is to set the user to NOMAIL yourself. Variations on this message include VMS's "file extend failed writing to [disk.user]MAIL.MAI".

- Unknown mailer error x

This is a favorite Unix sendmail configuration bounce. NOMAIL or DELETE, according to your preference. Since it is a configuration problem, it is usually transient. One system sent the following under an "unknown mailer error 1" heading:

```
binmail: /usr/spool/mail/userid: too big to accept new
messages.
    It's size is 205735 bytes which is 935 bytes over
quota.
mail: cannot open dead.letter
554 <userid@node>... unknown mailer error 1
```

This is apparently a "mailbox full" error, as "userid's" mail spool is "over quota". It is also possible that it means your message would put the user over quota by 935 bytes. Either way, there isn't enough space in the user's mailbox to store your message (in this case, it was a daily digest). Note that "unknown mailer error x" does not always mean the user's mailbox is full – what it always means is that sendmail cannot identify the cause of the error.

- Bounced, but sent successfully

This error comes from cc:Mail systems and is extremely misleading. It claims that the mail bounced to one address, but was sent successfully to another.

```
While talking to smtp.ccabc.com:
>> DATA
<< 554 I/O error to mailbox
554 MILLERT@smtp.ccabc.com... Service unavailable
----- Recipients of this delivery -----
Bounced, cannot deliver:
    MILLERT@smtp.ccabc.com
Sent successfully:
    <MILLERT@ABC.COM>
```

What this means (assuming that the mail hasn't gone through a redistribution list or a mirror site) is that you have a user MILLERT@ABC.COM on your list, and the server accepted the mail for that address successfully. However, that address actually maps to a different internal address (in this case MILLERT@smtp.ccabc.com) and for whatever reason, the server can't forward the mail on. This is the equivalent of a "user unknown" error for MILLERT@ABC.COM.

- Too many hops (or "notary" format error number 5.4.6)

Means that the message has transited through too many intermediate mail systems (1 transit = 1 hop). Most of the time this will be due to a temporary looping condition on the user's end (despite the "permanent" 5.4.6 error). For instance, the following Internet routing headers indicate a loop between three different mail machines (starting from the bottom and working back to the top):

```
Received: from un1.sample.com (root@un1.sample.com [200.9.212.3])
    by un7.sample.com (8.8.7/8.8.7) with ESMTP id RAA22765
    for <user@example.com.ar>; Wed, 4 Mar 1998 17:17:10 -0300
Received: from ull.sample.com (root@ull.sample.com [200.0.224.2])
```

```

by un1.sample.com (8.8.7/8.8.7) with ESMTP id RAA27352
for <user@example.com.ar>; Wed, 4 Mar 1998 17:16:00 -0300
Received: from un7.sample.com (un7.sample.com [200.9.212.4])
by ull.sample.com (8.8.8/8.8.8) with ESMTP id RAA13496
for <user@example.com.ar>; Wed, 4 Mar 1998 17:15:40 -0300 (GMT-3)
Received: from un1.sample.com (root@un1.sample.com [200.9.212.3])
by un7.sample.com (8.8.7/8.8.7) with ESMTP id RAA22034
for <user@example.com.ar>; Wed, 4 Mar 1998 17:15:27 -0300
Received: from grape.EASE.LSOFT.COM (grape.ease.lsoft.com [206.241.12.34])
by un1.sample.com (8.8.7/8.8.7) with ESMTP id RAA25235
for <user@example.com.ar>; Wed, 4 Mar 1998 17:08:43 -0300

```

The problem here appears to be that the mailers at sample.com are MX (mail exchanger) sites for example.com.ar, but that they can't decide which one of them should hold onto the mail until it can be delivered to example.com.ar. So it looped through 7 iterations until the un7 machine finally decided that enough was enough (including the passage through LISTSERV it had taken 26 hops and un7 was set to accept a maximum of 25 hops) and generated an error.

You may occasionally see a "too many hops" message that isn't a loop. Usually the non-looping variant is due to the recipient being many hops away from the mail originator and the maximum hop count being set too low on the recipient's machine. Many older sendmail installations, for instance, will accept only 10-15 hops before they reject the message. With today's Internet a setting of 30-40 is probably much more reasonable.

A particularly annoying error you may have to deal with (unlikely these days as this kind of addressing is becoming quite obsolete) comes from Banyan networks and is of the form:

```
LLONG@StarShip@Dora: Mailbox full
```

Obviously this is not a properly-configured address (at least, not as far as LISTSERV is concerned), and if you SCAN or QUERY the list for it, you will get a negative response. If, however, you SCAN the list for LLONG, you may find a user such as:

```

> scan test-l LLONG
Bill Smith <LLONG%StarShip%Dora@BOONDOCK.TERTIUS.COM>
SCAN: 1 match.

```

This user can now be set to NOMAIL and the errors will stop after the Banyan host has emptied its queue. If you do not find the user on the first SCAN, try using another part of the address as your search text.



Notes: A user may have his mail forwarded from the account that is actually subscribed to an account on another machine where he reads his mail. If the second machine is bouncing the mail, it may not be immediately apparent from the bounce messages that the mail is actually being forwarded. It is important to check for variants of the userid in the bounce message as it may be related to the userid that is actually subscribed to the list.

There are many forms of error messages. Many mail systems do not conform to Internet "standards" (some of them even return non-English error messages!) and LISTSERV's auto-deletion feature will not always catch their bounces.

13.7.4 Redistribution and Forwarding

Perhaps the worst type of bounce is one that comes from a user who is "hiding" behind an account that redistributes mail (a "redistribution list"), or a user whose Internet address has changed slightly but who is still subscribed to your list under his original address.

Redistribution lists typically (but not always) take some form of your list's name (such as "xxxxx-L-REDIST@foo.bar.com"), and thus their subscriptions tend to be easy to find. What is difficult is that you have no way of knowing which users (or how many users) are hidden behind this interface, nor any way of knowing what their userids are.

Forwarded accounts generally fall into one of two categories – those where the user has forwarded his own mail from one account to another rather than changing his subscription, and those where the user's system name has changed and the old address is still valid but is forwarding mail to the new address without the user being aware of it.

Let's say that suddenly you are bombarded with delivery errors for someuser@baz.net. Your immediate reaction is to set this person to NOMAIL or (in some cases) to delete him/her altogether. You therefore send `set xxxxx-L nomail for someuser@baz.net` to LISTSERV. LISTSERV responds: "No subscription for someuser@baz.net in list XXXXX-L."

In a best-case scenario, you can query the list for `*@*.baz.net` and find either a user like someuser@glork.baz.net (the address has changed and the local sysadmins didn't inform the user) or a redistribution-list account like xxxxx-L@baz.net. These are easily-fixed redistribution bounces. In the first case, you delete the user and let him or her resubscribe. In the second case, you can try sending a message to owner-xxxxx-l@baz.net with a cc: to postmaster@baz.net and inform them of the problem. If it persists, you could send a further message informing them that you are suspending the redistribution list's subscription until such time as they tell you the problem on their end is fixed, and simply set xxxxx-l@baz.net to NOMAIL.

The worst-case scenario is as follows: baz.net may be bouncing the mail to you, but there may not be a single subscription for baz.net in your list. Here's where you have to do some careful sleuthing. First, run a wildcard query such as `QUERY xxxxx-l FOR *@*baz*` or `QUERY xxxxx-l FOR *baz*@*`. The former will find users at baz.com, for instance, where baz.net is a synonym for baz.com. The latter query may seem somewhat strange, but it's possible that the mail is being routed through a gateway and the actual subscription is for xxxxx-l%baz.net@cunyvms.cuny.edu or similar.

13.7.5 "Sender:", "From:", or "Reply-To:" Fields in Body Causes Bounce

Sometimes you will receive bounces from LISTSERV with a error header like this:

The enclosed message, found in the VISBAS-L mailbox and shown under the spool ID 19630445 in the system log, has been identified as a possible delivery error notice for the following reason: "Sender:", "From:" or "Reply-To:" field pointing to the list has been found in mail body.

Sometimes this is a legitimate bounce from a mail system that isn't compliant with Internet standards for mail, and the reason the "Sender:", "From:", and/or "Reply-To:" headers are significant is because if this mail were to be allowed through to the list it could very possibly start a loop with the non-compliant mail server. Normally this is a good thing; however, an unfortunate side-effect of the loop-checking code that catches this kind of bounce means that LISTSERV may treat replies to list mail from some mail clients as if they are delivery errors. LISTSERV has no way to know the difference

between a bounce and a legitimate message that just happens to have unquoted included headers so it takes the conservative route and bounces it to the list owner as a "possible" delivery error. This way the list owner can (if he or she wants to) return the message to the user in question and ask them to either quote out or delete the headers from their replies.

In any case this is specifically known to be a problem with Pegasus Mail and some incarnations of the Microsoft Exchange Client, but there are probably other mail programs that do the same thing. The problem arises when the user's mail client includes the "Sender:", "From:", or "Reply-To:" fields that point back to the list itself (for instance, the above error was for VISBAS-L@PEACH.EASE.LSOFT.COM) in the quoted material and doesn't quote them correctly--that is to say, without a quoting character, or with a space between the quoting character and the included text. For instance, a reply from Pegasus with quoted material might include the following lines:

User's reply,...

```
> Date: Tue, 31 Dec 1996 17:00:00 -0700
> Reply-to: Visual Basic List <VISBAS-L@PEACH.EASE.LSOFT.COM>
> From: Joe User <JOE@UNIX.FOO.COM>
> Subject: Re: 97 Style ToolBars
> To: VISBAS-L@PEACH.EASE.LSOFT.COM
```

The quoted lines below the user's reply would trigger LISTSERV's loop detection functions because there is a space between the ">" character and the "Reply-To:" and the "From:" headers.

The correct, netiquette-approved method of quoting these headers is to delete them entirely from the body of your message. Quoting is generally done for reasons of context and message headers are not needed for context. (Pegasus actually lets you toggle this on and off via the "Advanced options for replies" dialog. Other clients don't seem to have this function.) Note that Eudora quotes messages with no space between the ">" character and the quoted text, so this is not an issue with Eudora.

If necessary, subscribers using Pegasus can change the quoting character (at least they can in some versions of Pegasus; the author has not tested current versions) by editing their copy of PMAIL.INI and changing the value in

[General]

...

Commenting string = >

Normally this variable contains "> ", that is, ">" followed by a space character. If you remove the space, Pegasus quotes "properly" and this is no longer a problem. Other mail clients may or may not have similar configuration settings. As always, users should consult the documentation for their mail program before making changes to its configuration.

(Also, see Section 19.2 [Loop-Checking Can Cause Occasional Problems with Quoted Replies](#)).

13.7.6 LMail Error Codes

LMail is an L-Soft mailer product for VM mainframes. When it receives error mail from remote hosts it translates the error into a standard format recognized by LISTSERV so

that LISTSERV can take action as necessary. From time to time you may see such errors in your mailbox; they look like this:

```
-----
Date:      Fri, 8 Jan 2000 20:04:50 +0100
Reply-To:  Postmaster@SEARN.SUNET.SE
From:      RFC822 mailer (LMail release 1.2d/1.8d) <MAILER@SEARN.EXAMPLE.SE>
Subject:   Undelivered mail
To:        ERIC@SEARN.EXAMPLE.SE
cc:        Postmaster@SEARN.SUNET.SE
X-Report-Type: Nondelivery; boundary="> Error description:"

An error was detected while processing the enclosed message. A list of the
affected recipients follows. This list is in a special format that allows
software like LISTSERV to automatically take action on incorrect addresses; you
can safely ignore the numeric codes.

--> Error description:
Error-For:  JACK@SEARN.SUNET.SE
Alias:      JACK@SEARN.BITNET
Error-Code: 3
Error-Text: No such local user.
Error-For:  JOE@SEARN.SUNET.SE
Alias:      JOE@SEARN.BITNET
Error-Code: 3
Error-Text: No such local user.
Error-End:  Two errors reported.

----- Rejected message (5 lines) -----
Date:      Fri, 8 Jan 1993 20:04:47 +0100
From:      Eric Thomas <ERIC@SEARN.BITNET>
To:        JACK@SEARN.BITNET, JOE@SEARN.BITNET
Testing.
```

The LMail codes stand for the following:

- 0 is used for all the weird errors that can't easily be classified, and LISTSERV takes no action on these codes by definition. This includes errors such as "invalid device name" or "device full" which have meaning only for the postmaster of the host that bounced the message.
- 1 means "don't know that/don't know how to get there" (as opposed to "can't get there right now"), which is used when the host can't be found (e.g., "host unknown").
- 2 means "configuration error". This means that LMail has detected an error in its routing tables which prevents it from delivering the mail. It does not necessarily mean that the address is bad.
- 3 is "no such local user".
- 4 is "not allowed to mail to this user".

The difference between 3 and 4 is that a 3 indicates there is no way to successfully send mail to the user, whereas 4 indicates the user cannot receive mail from the address your message came from. LMail uses code 4 when a local LMail user directs it to reject all mail coming from mailing lists, but to let private mail through. Typically you will not see very many 4 codes.

- 5 means "mailbox full", "quota exceeded", and so on.

LISTSERV itself takes action on error codes 1, 3, and 4, and forwards anything else to the list owner.

13.8 Delivery Error Handling Features

LISTSERV supports several levels of automatic deletion based on error messages passed back to it in LMail format by certain remote systems. While auto-delete will not solve all of your bouncing mail problems, it has the potential to take care of most "permanent" errors (including "no such user" and "no such host"). However, note that auto-delete ignores "temporary" errors such as "host unreachable for 3 days", "system error", "disk quota exceeded", and so forth, such that users whose accounts generate "temporary" errors are not summarily deleted from the list.

By default, LISTSERV mailing lists generate a report which lets the list owner know what userids are causing problems, rather than deleting users at the first error LISTSERV understands. If the Delay() and Max() parameters are set to non-zero values for a list coded "Auto-Delete= Yes", LISTSERV will not take immediate action on mail delivery errors. You will receive an "auto-deletion monitoring report" daily to show you which subscribers are bouncing mail, what the error is, when it started, when the last error arrived, and how many errors have been received for the subscriber in total. By default, LISTSERV will wait 4 days (or for a maximum of 100 error messages per individual user) before deleting a subscriber.

If you code "Delay(0)", LISTSERV will not wait to take action, but will delete the subscriber at the first error LISTSERV understands. Note carefully that LISTSERV will not generate a daily error monitoring report when Delay(0) is used.

The default value is "Auto-Delete= Yes,Semi-Auto,Delay(4),Max(100)" for lists coded "Validate= No" and "Auto-Delete= No" for all other lists.

Under LISTSERV Lite, Auto-Delete= is available but deletes on the first bounce (e.g., "Delay(0),Max(1)") regardless of the Delay() and Max() settings.

Implementation of the "Auto-Delete=" keyword is discussed in detail in the [List Keyword Reference](#) document under "Error Handling Keywords."

13.8.1 Auto-Delete Considerations for Holidays

Making a big increase to the DELAY threshold to provide more leniency during a holiday may not be a good idea. While it will indeed disable the monitor for the duration of the holiday, switching back to the normal threshold when you return will cause the monitor to delete all the users that had been bouncing during the holidays. In general, you should avoid making temporary changes to the DELAY threshold, because it takes the monitor a while to adapt to the new settings.

The best way to relax the rules during a long holiday is to leave the DELAY threshold unchanged but switch the monitor to passive mode ("Auto-Delete= Yes,Manual"). Nobody will be deleted over the holidays, but the monitor's cycle will not be perturbed. When you return, you should wait about a week before switching back to automatic mode. This is because, after a long holiday such as Christmas, it usually takes about 2 working days for system administrators to solve all problems. In some cases, the problems will have caused bounces to remain undelivered. So, by fixing the problems, the system administrators may actually send a flood of new bounces corresponding to problems that have now been solved. Unfortunately, since the monitor only receives NON-delivery reports, it has no way to know that these problems have in fact been

solved. As a rule of thumb, you will note that your daily delivery error reports are much longer than usual over the vacation. When you return, you should wait until they are back to their normal size before switching back to automatic mode.

13.9 Address Probing

This functionality is not available in LISTSERV Lite.

There are two levels of automatic address probing available in LISTSERV.

13.9.1 Active Address Probing

This functionality is not available in LISTSERV Lite.

Active address probing is available for two reasons: first, to enhance subscription renewal functionality so that no "CONFIRM *listname*" response is required from subscribers in order to stay subscribed, and second, to enhance the ability of the auto-deletion feature to handle bounces that can't be parsed into something LISTSERV can recognize.

"Renewal= . . . ,Probe" activates this enhanced bounce processing feature, whereby subscribers are probed at subscription renewal time using the PROBE1 mail template. The "Probe" option makes subscription renewal passive rather than reactive; no "CONFIRM *listname*" response is needed from the user. In fact, the desired response from the user is to discard the message and do nothing, making the process very simple. LISTSERV also probes addresses that return mail delivery errors, and probe messages have a special signature in the return address that allows LISTSERV to uniquely identify any bouncing address, without having to understand the bounce itself.

If the probe bounces, LISTSERV first sends the PROBE2 template with a copy of the bounce, to show the user (if the account actually works in spite of the bounce) what garbage his mail system is sending people. LISTSERV then schedules a new probe for the next day, or deletes the user immediately, depending on the auto-delete policy. Every failure triggers a new daily probe until the user gets deleted or the problem gets fixed. The user can also save his subscription manually by sending a CONFIRM *listname* command (this is explained in PROBE2). This doesn't solve the underlying problem, so eventually the user should get tired of confirming in an emergency and notify his system administrators that the system is generating bounces saying (for instance) "Your message was registered at the MORONICUS mail gateway. Press F1 for more information" that cause the problem in the first place.

When used together with "Auto-Delete= . . . ,Full-Auto", the probe option deletes all delivery errors from bounced probes, even if LISTSERV can't understand the error. This means the list owner never ever has to see a single bounce from a probed address. The list, however, is kept clean because bad addresses are always detected. In fact, the biggest risk is that the users of the MORONICUS mail gateway will be deleted even though they do get their mail.

This being said, note carefully that all errors bounced by non-compliant mail hosts to the wrong address, and non-probe errors that are sent to the owner-*listname* address but are not in a format that LISTSERV can parse, will still show up in your error mailbox. If the bounce goes to the wrong address, LISTSERV never sees it and cannot probe it. If the error goes to the correct address (owner-*listname*) but isn't specific enough for LISTSERV to understand, while LISTSERV will be able to see it, it still won't be able to probe it. Finally, in some cases where the error is so vague (or constructed in a

complicated manner that defies LISTSERV's attempts to parse it) the error may be passed to the LISTSERV postmaster, instead of to the list owner, for disposition, even if it was correctly returned to the owner-listname address.

Yet even with these restrictions, the author saw an error queue of 1300 errors/day shrink to under 50 errors/day by applying the ",Probe" parameter to seven high-volume lists, which in his opinion was much more acceptable.

13.9.2 Passive Address Probing

This functionality is not available in LISTSERV Lite.

In effect, passive probing is very similar to active probing, but it is not tied to subscription renewal. Passive probing is enabled by default for small lists (e.g., <1K subscribers) but not for large ones due to the fact that passive probing does cost additional resources and large lists are often used for one-shot mailings where it is simply not effective to use those resources to probe addresses that will not be used a second time.

Passive probing operates by turning a certain percentage of your regular list messages into transparent probes that look like a normal message but also double as a probe, rather than sending out the explicit PROBE1 template as in active probing. You enable (or tune) passive probing by adding a ",Probe(xx)" parameter to the Auto-Delete= keyword setting. For instance,

```
Auto-Delete= Yes,Full-Auto,Probe(30)
```

where "30" is the number of days to wait between probes for any given user (the default is Probe(30). Subscribers with working mail systems will not see any difference, subscribers with flaky mail systems will occasionally receive a message showing that their mail bounced and saying that they should report the problem to their ISP, and of course plain bad addresses will go away.

In order to disable passive probing you set the probe parameter to 0, i.e.,

```
Auto-Delete= Yes,Full-Auto,Probe(0)
```

If you have users who for whatever reason should not be probed, then you can deactivate passive probing (and any other renewal you have set for the list) with the SET userid@host NORENEW command.

13.10 Subscription Confirmation

For lists coded "Subscription= Open", you can require confirmation on all new subscription requests, thus ensuring that LISTSERV has a clear mailing path back to the subscriber. In the past, a user could send a subscription for an open subscription list to LISTSERV, which upon acceptance would immediately start sending the user list mail. If the user was located behind a "broken" or one-way gateway, this produced immediate bounced mail until the list owner noticed and deleted the subscription. Note that requiring confirmation at the time of subscription does not guarantee that the clear mailing path will continue to exist permanently.

"Subscription= Open,Confirm" causes LISTSERV to send a Command Confirmation Request to the potential subscriber before actually adding the user to the list. The subscriber is requested to reply to the request by sending a validation "cookie" back to LISTSERV (this "cookie" being the hexadecimal number pulled from the subject line).

The Command Confirmation Request, while straightforward, has the potential to cause confusion if users do not read carefully the instructions that make up the request. LISTSERV expects confirmation codes to be sent in a specific way because some mail gateways add lines to the header of the message that LISTSERV doesn't understand. If a user forwards the request back to LISTSERV, or creates a new mail message to send the 'cookie' back, it usually will not work correctly. The sequence should thus be as follows:

1. SEND the subscription request to LISTSERV.
2. REPLY to the confirmation request ('ok')
3. SEND the confirmation code (if necessary) ('ok 23CBD8', for example)
4. Send mail to the list owner (not the list) if the subscription request fails after step 3.



Note: If a list owner adds a user manually, the confirmation process is bypassed.

13.11 Subscription Renewal

You can code subscription renewal into your lists. This is one method to keep lists "pruned down" and avoid having large lists that are actually distributing mail to only a fraction of the users. For instance, you may have a number of subscriptions set to NOMAIL for one reason or another. NOMAIL user(a) may have forgotten that he has a subscription; user(b) may have set NOMAIL instead of unsubscribing; user(c) may no longer exist because she graduated or no longer works for the service provider; you may have set user(d) to NOMAIL because of recurrent mail delivery errors. Requiring a periodic confirmation of subscriptions is therefore a reasonable course of action for large, non-private lists.

Subscription renewal is disabled by default. If you do not want subscription renewal, or if you wish to turn it off, simply do not include a "Renewal=" keyword in your list header.

To add subscription renewal, you add the following keyword to the header of your list:

```
* Renewal= interval
      - or -
* Renewal= interval,Delay(number)
      - or -
* Renewal= interval,Delay(number),Probe
```

where interval is a period of time such as Weekly, Yearly, 6-monthly, or something similar, and Delay(number) is an integer corresponding to how many days LISTSERV will wait for the renewal confirmation to arrive. (See "Renewal=" in the [List Keyword Reference](#) document for more information on renewal and delay periods; see Section 4.9 [Address Probing](#) for more information on the "Probe" parameter. You can have multiple interval parameters; again, see the entry for "Renewal=" in the [List Keyword Reference](#) document for details).

The confirmation request mailing asks the subscriber to send the command CONFIRM listname back to LISTSERV. If the subscriber does not do so within a certain length of time, LISTSERV automatically deletes the subscription. The default delay time is 7 days. If you wish to use the default delay time, it is not necessary to code ",Delay(7)" into your Renewal parameters.



Note: You may wish to increase the delay time to accommodate users whose subscriptions expire over holidays (such as the Christmas/New Year's week) in order to avoid accidental deletions. Also, be aware that confused subscribers can and will send the CONFIRM command back to the list, rather than to LISTSERV. LISTSERV's default filter will catch these commands and forward them to the userid(s) defined by the "Errors-To=" keyword.

It is possible to waive subscription renewal for certain users (such as list owners, editors, redistribution lists, etc.). In order to do this, simply issue the command

[QUIET] SET listname NORENEW FOR net-address

to LISTSERV. It is most advisable to do this in the case of redistribution lists, as they broadcast the renewal notice to their users, who a) cannot renew the subscription and b) become very confused when they see the notice, often sending "what does this mean?" mail to the list.

You can also issue the CONFIRM command for a subscriber:

[QUIET] CONFIRM listname FOR net-address



Note: "Active" users of the list (that is, people who post regularly to the list) will never be required to renew their subscriptions, nor (if subscription "probing" is enabled) will they ever be sent the passive subscription probe. LISTSERV presumes that such users have valid addresses and does not require a renewal confirmation from them.

13.12 Using the SERVE Command When a User is "Served Out"

If a user sends more than 50 consecutive invalid commands to LISTSERV, LISTSERV automatically serves that user off so that further commands from that user will be ignored. Should a user become served off in this fashion, it is possible for the list owner or any other user to issue a `SERVE net-address` command to restore that user's access. As with all other LISTSERV commands, the SERVE command is sent to LISTSERV.

While served off, the user will be unable to set personal options and will be unable to subscribe or unsubscribe to lists on that server. Note that a user will likely be served off of one particular LISTSERV site but not others, and also that the user may not even realize that he has been served off (in spite of the fact that LISTSERV sends notification to the user to that effect).



Note: The `SERVE` command will not restore service to users who have been manually served off by the LISTSERV maintainer.

Section 14 Setting Subscription Options for Subscribers Using Email

14.1 Reviewing Current Subscription Options with QUERY

The syntax is similar to the subscriber's method of reviewing his options, except that the list owner must specify for whom the options are being checked.

```
Query listname FOR userid@host
```

It is possible to use wildcards in the subscriber address. For instance,

```
Q LSTOWN-L FOR J*@UBVM*
```

will return option listings for subscribers such as JIMJ@UBVM, JOHN@UBVMS.CC.BUFFALO.EDU, etc. This can be handy if you are searching the list for someone whose subscription address differs from the address you are given in an error report (see the examples, above, in "Dealing with bounced mail").

Using the WITH qualifier, you can also query a list for users who have a specific option set. For instance, you might want to know which users are set to NOMAIL. Send the command

```
Q listname WITH NOMAIL FOR *@*
```

and LISTSERV will return a list of those users. It is also possible to query a list for multiple options:

```
Q listname with DIGEST CONCEAL FOR *@*
```

will return a list of those subscribers who have set their subscription to DIGEST and also to CONCEAL.

You can also query users by the list topics they are subscribed to. For instance:

```
Q listname WITH TOPICS: ADMIN FORUM FOR *@*
```

shows all members subscribed to both the ADMIN and FORUM topics.

```
Q listname WITH TOPICS: -ADMIN FOR *@*
```

shows all members who are not subscribed to the ADMIN topic.

```
Q listname WITH TOPICS: ADMIN -TEST FOR *@*
```

shows all members who are subscribed to the ADMIN topic but not to the TEST topic.

14.2 Setting Personal Subscription Options for Subscribers

Again, the syntax is similar to the subscriber's method.

```
SET listname option FOR userid@host
```

— or —

```
QUIET SET listname option FOR userid@host
```

14.3 Subscription Options

14.3.1 Mail/NOMail

Setting this option to `Mail` indicates that the subscriber will receive mail from the list. `NOMail` is the complementary command that stops mail but leaves the user subscribed to the list. (`NOMail` is often a good compromise for users who are leaving the office for vacation or on extended business trips, and who don't want a full mailbox on their return.) The format of the messages received is controlled by the `DIGEST/INDEX/NODIGEST/NOINDEX` options (see the following sections).

14.3.2 DIGest/NODIGest

Causes the subscriber to receive one posting per digest cycle (typically daily) rather than individual messages as they are processed by `LISTSERV`.

The `MAIL/NOMAIL` option is isolated from `DIGEST/INDEX`. The `MAIL/NOMAIL` option controls whether messages should be delivered, and the `DIGEST/INDEX/NODIGEST/NOINDEX` option controls the format in which messages should be delivered. Thus, switching to `NOMAIL` and back to `MAIL` now preserves the digest/index/normal delivery setting. To provide as much compatibility with the old syntax as possible, the four options operate as follows:

- `DIGEST` – Enable digest delivery mode (which negates `INDEX`), enable mail delivery.
- `INDEX` – Enable index delivery mode (which negates `DIGEST`), enable mail delivery.
- `NOMAIL` – Disable mail delivery.
- `MAIL` – Restore mail delivery. Note that If mail delivery was already enabled, the `MAIL` option negates `INDEX/DIGEST`. Thus, a user going from `NOMAIL` to `MAIL` will keep his previous delivery options, whereas a user going from `DIGEST` or `INDEX` to `MAIL` will in fact deactivate index/digest mode.

To revert from digest/index subscription mode to normal delivery, you can use either the `MAIL` option as before, or the `NODIGEST/NOINDEX` option.

The command `SET listname MAIL` sent by a user who is set to `DIGEST` but not also set to `NOMAIL` will cause the user to be set to `NODIGEST` (the behavior is identical for users set to `INDEX` but not to `NOMAIL`). `SET listname MAIL` sent by users set to `DIGEST/NOMAIL` or `INDEX/NOMAIL` will simply remove the `NOMAIL` setting and leave the user set to `DIGEST` or `INDEX` as the case may be.



Note: In extreme cases, subscribers using the `DIGEST` option may receive more than one digest per cycle if the digest limit is reached before the end of the cycle.

14.3.3 MIME/NOMIME

Toggles MIME functions on and off. Currently this is only useful if the user has a mail client that supports MIME digests. Users who send their `SUBSCRIBE` command using a MIME-compliant agent will have this option set automatically unless "`Default-Options= NOMIME`" is specified for the list.

In future versions, this toggle may control other MIME functions.

14.3.4 INDEX/NOINDEX

Causes the subscriber to receive one posting per digest cycle containing only an index of subject topics for all messages during that cycle. See the section on DIGEST (above) for further information.

14.3.5 ACK/NOACK/MSGACK

These three command words control the level of acknowledgment the subscriber receives when posting to the list. ACK causes LISTSERV to send a short confirmation message to the subscriber when the post has been received and distributed. NOACK disables the confirmation feature for the subscriber (although BITNET subscribers will receive a short interactive message on their terminal). For BITNET subscribers, MSGACK provides the same information as ACK via interactive messages.

14.3.6 Options for Mail Headers of Incoming Postings

By specifying one of the following command words, the subscriber can control the amount of mail header information prepended to list mail. The syntax is `SET listname headertype`, where *headertype* is one of the following:

- FULLhdr – "Full" mail headers (default) (formerly FULLBSMTP)
- SHORThdr – Short headers (formerly SHORTBSMTP)
- IETFhdr – Internet-style headers
- DUALhdr – Dual headers, useful with PC or Mac mail programs
- FULL822 – "Full" RFC822 mail headers
- SHORT822 – Short RFC822 mail headers
- SUBJecthdr – "Full" RFC822 mail headers (like the default), except that LISTSERV adds the list's default subject tag to the subject line of mail coming from the list. To turn this off, simply set another mail header option.



Note: Do not use FULL822 or SHORT822 unless debugging specific problems or unless directed by L-Soft. Use of these options can seriously slow performance as they force LISTSERV to generate a separate "envelope" for each user so set. FULL822 and SHORT822 are obsolete but remain available for compatibility with certain older BITNET mailers still in use.

Quite a few non-technical users are relying on non-RFC822 user interfaces for reading their mail. Quite often these user interfaces are user-friendly, quality implementations of a proprietary mail protocol which the users are proficient with, but which happens not to lend itself to bidirectional mapping to RFC822. The users may have a good reason for using this particular program, and they complain that it is not always clear what list the postings come from, or who posted them. Other users have very primitive mail programs which do not preserve the original RFC822 header and may not even have a "message subject" concept. The user knows which list the message came from, but not who posted it, making private replies impossible.

The DUALHDR (minimum abbreviation: DUAL) header option is provided to help solve this problem. Dual headers are regular short (SHORThdr) headers followed by a second header inside the message body. This second header shows what list the message is

coming from ('Sender:'), the name and address of the person who posted it ('Poster:'), the poster's organization, if present, and the message subject. The date is not shown because even the most primitive mail programs appear to supply a usable message date.

The `SUBJECTHDR` (minimum abbreviation: `SUBJ`) header option is provided for users who want to see a "tag" in the subject line of their incoming list mail that indicates where the mail is coming from (e.g., to activate a filter in their mail program to drop the message into a specified notebook). If you have `SHORT` headers (or any other header option) set, setting your option to `SUBJECThdr` will automatically change you to `FULLHDR`, as subject tags require full headers. Also, subject tags are not generated for messages sent without a RFC822 "Subject:" header.

Generally, users will be well-served by the `FULL` header option, which is the default.



Documented Restriction: The use of the `SHORTHDR` or `DUALHDR` options will break messages that depend on MIME encoding, because these options strip the RFC822 headers that identify the message as a MIME message. `SHORTHDR` and `DUALHDR` were designed for the non-MIME mail clients which prevailed in `LISTSERV`'s early history. As most mail clients today support MIME, the use of these options is now deprecated.

14.3.7 Putting the List Name into the Subject: Field

To do this, use the `SUBJECThdr` personal option as explained in the previous section. To set this option by default for new subscribers, include it in the `Default-Options=` keyword setting for your list (see Section 14.4 [Setting Original Default Options with the Default-Options= Keyword](#)). To set it for existing subscribers, use the `SET` command.

14.3.8 CONCEAL/NOCONCEAL

Occasionally, a subscriber may not want his presence to be known to someone else making a casual `REVIEW` of the list. Subscribers may choose to "hide" their subscription from the `REVIEW` command by using the `CONCEAL` command. Conversely, a subscriber may choose to remove this restriction by issuing the `NOCONCEAL` command.



Note: The list owner can always obtain a list of all subscribers, both concealed and unconcealed, by issuing the `GET listname (NOLOCK)` command, or by issuing a `QUERY listname WITH CONCEAL FOR *@*` command.

14.3.9 REPro/NOREPro

This option controls whether or not the subscriber will get a copy of his or her own posts back from the list after they are processed. Generally, if a subscriber's mail program is configured to file copies of the subscriber's outgoing mail, or if the subscriber has one of the acknowledgment options (`ACK/MSGack`) enabled, this option should be set to `NOREPro`. If, on the other hand, the subscriber is set to `NOACK` and doesn't keep a copy of outgoing mail, this option should probably be set to `REPro`.

14.3.10 TOPICS

Topics are not available in `LISTSERV Lite`.

If list topics are enabled, this option allows the subscriber to specify which topics he or she will receive. The syntax of a `SET TOPICS` statement is significantly different from that

of the other options. See Section 15.7 [Using List Topics](#) for more information on this syntax.

14.3.11 POST/NOPOST

This option may be set only by list owners or the LISTSERV maintainer. It is not available in LISTSERV Lite.

A subscriber set to NOPOST may not post to the list. NOPOST gives the individual list owner the ability to serve out abusive or obnoxious posters without having to add such users to the list's "Filter=" setting. Subscribers set to NOPOST will still receive list mail – they just won't be able to post mail to the list.

The list owner or LISTSERV maintainer may issue the

```
SET listname POST FOR userid@host
```

command to reverse a previously-set NOPOST.



Note: For peered lists, NOPOST must be set globally or a user can bypass the setting by simply posting to another peer. Thus, you must add the user manually to the other peers and then set the user to NOMAIL as well as NOPOST on the peers.

Setting NOPOST for a user cancels any previous EDITOR or REVIEW setting for that user.



Note: List editors who are set to NOPOST will be able to approve messages but will then be told they cannot post to the list. The NOPOST subscriber option does override any Editor= or Moderator= definition in the list header, so be sure that your editors and moderators are set to POST.

14.3.12 EDITOR/NOEDITOR

This option may be set only by list owners or the LISTSERV maintainer, and is effective only on moderated lists. It is not available in LISTSERV Lite.

A subscriber set to EDITOR on an edited/moderated list may post directly to the list without a moderator's intervention. It is virtually identical to adding the subscriber's address to the "Editor=" keyword, but easier to manage. The only difference between the EDITOR option and the "Editor=" keyword, other than not being visible in the list header, is that the "Editor=" keyword also defines a (seldom used) access level class that can then be used in keywords such as "Review=". Thus, one could have a list with "Review= Editor", indicating that only the users listed in the "Editor=" keyword are allowed to review the list. The EDITOR option does not confer this privilege.



Note: The EDITOR option is only meaningful on moderated lists.

The list owner or LISTSERV maintainer may issue the

```
SET listname NOEDITOR FOR userid@host
```

command to reverse a previously-set EDITOR.

Setting EDITOR for a user cancels any previous NOPOST or REVIEW setting for that user.

14.3.13 REVIEW/NOREVIEW

This option may be set only by list owners or the LISTSERV maintainer. It is not available in LISTSERV Lite.

When a subscriber is set to REVIEW, all postings from that subscriber are forwarded to the list editor or list owner for approval. Approval for these postings is always via the OK mechanism – there is no need to forward the posting to the list, simply reply to the approval confirmation with "OK".

Note that if a list is unmoderated, it is still possible to direct REVIEW postings to a specific person by adding an "Editor=" or "Moderator=" keyword to the list header.

The list owner or LISTSERV maintainer may issue the

```
SET listname NOREVIEW FOR userid@host
```

command to reverse a previously-set REVIEW.

Setting REVIEW for a user cancels any previous NOPOST or EDITOR setting for that user.

14.3.14 RENEW/NORENEW

This option may be set only by list owners or the LISTSERV maintainer.

Enables or disables subscription renewal confirmation on an individual subscriber basis. Setting a subscription to NORENEW is particularly useful for exempting list owners, redistribution lists, and other subscriptions which should not or must not receive the confirmation request message from the renewal process.

The list owner or LISTSERV maintainer may issue the

```
SET listname RENEW FOR userid@host
```

command to reverse a previously-set NORENEW.

14.4 Setting Original Default Options with the Default-Options= Keyword

The list owner may specify original defaults for many subscriber options by using the "Default-Options=" keyword. This keyword takes regular SET options as its parameters. Examples include:

- * Default-Options= DIGEST,NOREPRO,NOACK
- * Default-Options= REPRO,NONE

You may have more than one "Default-Options=" line in your header, as needed.



Notes: Any default topics are set with the "Default-Topics=" keyword. See the [List Keyword Reference](#) document for details on this keyword.

Your existing subscribers are not affected by any change to the Default-Options= keyword. This keyword sets initial options only for people who subscribe after it is defined. If you want to update your existing subscribers to the Default-Options settings, you must use the SET command with a wildcard (i.e., FOR *@*) to do so.

14.5 Setting the Misc-Options= Keyword

The `Misc-Options=` keyword is a catch-all for certain behavior-modifying options that are not otherwise covered by other, more specific keyword settings. Currently the only options available are as follows:

- `DISCARD_HTML`
- `IETFHDR_SUBJECT_TAG`
- `IGNORE_EMAIL_CASE` | `RESPECT_EMAIL_CASE`
- `KEEP_DKIM_SIGNATURE`
- `KEEP_EXCHANGE_DATA`
- `NO_DKIM_SIGNATURE`
- `NO_RFC2369`
- `NO_SPAM_CHECK`
- `RESPECT_EMAIL_CASE`
- `SUBJECTHDR_SEQUENCE`
- `SUPPRESS_APPROVED_BY`



Notes: For details on other list keywords, see the [List Keyword Reference](#) document.

14.5.1 DISCARD_HTML

Replaces "Language= NoHTML", which is deprecated starting with LISTSERV 15.5.

Tells LISTSERV to strip any HTML attachments from postings (while retaining HTML tags sent in the body of plain text messages).

The actual function of this setting is to remove the attachment that contains the HTML mail from the message. It does not remove HTML tags from plain text messages. This means that setting this option will not suppress HTML in messages sent from (for instance) Eudora Pro 3.x (since Eudora Pro 3.x does not send the HTML message as a MIME attachment with a plain text alternative).

If an HTML message does not contain a plain text alternative, and this option is set, the message will be rejected. LISTSERV 1.8d and earlier would allow such messages through.

14.5.2 IETFHDR_SUBJECT_TAG

Add subject tags to IETF headers (that is, for users who are set to the IETFHDR personal option). However, as this can be considered a violation of the standard for IETF-style headers, it can be prevented site-wide by the site administrator if desired.

Adding "`Misc-Options= IETFHDR_SUBJECT_TAG`" to the list header causes the IETFHDR option to always include subject tags. This is a per-list setting; in other words, either all subscribers to a list who are set to the IETFHDR option get the subject-tag, or no. The design consideration is that, at present, it is prohibitively expensive to switch the existing header types into flags as the number of combinations grows significantly.

14.5.3 IGNORE_EMAIL_CASE | RESPECT_EMAIL_CASE

These options are mutually-exclusive; only one can be defined at a time per list.

When set in a list header, "Misc-Options= IGNORE_EMAIL_CASE" causes the ADD command to ignore the case of the "local part" of list subscriber entries (that is, the part of the address that is to the left of the "@" sign). Although most modern mail clients are configured to ignore the case of the local-part, this behavior technically violates RFC821 which states that local-parts are considered case-sensitive.

If an entry whose "local part" differs only in case is found in the list during an ADD operation (for instance, JOE@EXAMPLE.COM vs. joe@EXAMPLE.COM), that entry will be assumed to be the entry that was sought, and the address field will be updated to the new case (that is, "JOE@" will be changed to "joe@"). No other change will be made to the entry unless there is a change in the name field, in which case the name field will also be updated.

If there is no change in the address field associated with the entry, no change will be made to the entry (again, unless the name field changes, in which case the entry will be updated).

In either case, when this option is set, a new entry with a different case will NOT be added.

Note the following caveats:

1. Pre-existing duplicates are not automatically removed from lists when this option is set.
2. Because ADD updates the case of entries, it is possible to end up with multiple entries that have exactly the same case.
3. The only real way to de-dupe a given address is to DELETE and then re-ADD it.

Other than this, existing duplicate entries work exactly as they did before the option was enabled. Commands that do not add new entries ignore the option.

And finally, it should be carefully noted that the PUT command also ignores the option.

When a list is set to "Misc-Options= RESPECT_EMAIL_CASE", this tells LISTSERV to operate per RFC821 and treat address fields with differently-cased local parts as different addresses. The option is provided as an override to the site-level IGNORE_EMAIL_CASE configuration variable and does not need to be set to preserve the default unless the site setting has been changed to make IGNORE_EMAIL_CASE the default.

14.5.4 KEEP_DKIM_SIGNATURE

Incoming DomainKeys signatures in messages submitted to a mailing list will be suppressed unless "Misc-Options= KEEP_DKIM_SIGNATURE" is set in the list configuration.

The KEEP_DKIM_SIGNATURE option is experimental and not meant for general use. As DomainKeys is specified today, signatures DO NOT survive posting to mailing lists (LISTSERV or otherwise), so LISTSERV removes them by default to avoid triggering alerts for subscribers on systems that have implemented the client side of DomainKeys.

14.5.5 KEEP_EXCHANGE_DATA

Replaces "Language= Exchange", which is deprecated starting with LISTSERV 15.5.

Tells LISTSERV to keep Microsoft Exchange attachments in postings (the default is to remove them). Note that this affects "application/ms-tnef" attachments only-- LISTSERV does not currently strip WINMAIL.DAT attachments.

14.5.6 NO_DKIM_SIGNATURE

"Misc-Options= NO_DKIM_SIGNATURE" is available at the list level to override LISTSERV's default DomainKeys message signing if desired.

14.5.7 NO_RFC2369

RFC2369 support is activated by default and supplies all of the headers specified in the standard except "List-Post:", which L-Soft considers to be redundant.

In compliance with RFC2369, LISTSERV discards any pre-existing List-xxx tags.

RFC2369 compliance can be disabled using:

```
* Misc-Options= NO_RFC2369
```

and this can also be specified in the site-wide DEFAULT_MISC_OPTIONS variable. When RFC2369 support is disabled, you get the old behavior; that is, the tags are neither added nor removed.

14.5.8 NO_SPAM_CHECK

Use this option to disable spam scans for a particular list and its associated xxx-request address. (This is only useful if the LISTSERV maintainer has enabled spam-scanning via the SPAM_EXIT feature.)

14.5.9 RESPECT_EMAIL_CASE

See above at [IGNORE_EMAIL_CASE | RESPECT_EMAIL_CASE](#).

14.5.10 SUBJECTHDR_SEQUENCE

It is possible for each list posting to have a sequence number attributed to it, which can be seen by subscribers who are set to the SUBJECTHDR personal option. This new feature is enabled by adding "Misc-Options= SUBJECTHDR_SEQUENCE" to the list header. Site administrators can enable it server-wide by adding the value SUBJECTHDR_SEQUENCE to DEFAULT_MISC_OPTIONS in the site configuration. The format of the new subject tag is

```
[listname - number]
```

For example:

```
Subject: [TEST - 256] Test of SUBJECTHDR_SEQUENCE
```

where 'number' is a sequence number, starting from 1 and increasing indefinitely for all practical purposes (the counter will accommodate over 2 billion postings per list).

"Subject-Tag=" is still used to change the first item. In order to allow server administrators to set a server-wide default for the new feature, it was not possible to implement the sequence number feature as an extension to "Subject-Tag=".



Note: Both the new [listname - number] and the traditional [listname] tags are removed from the subject on incoming messages. This happens whether or not the option is set, because people might be replying to old messages before the option was changed.

LISTSERV tries very hard never to skip a sequence number. The only plausible scenario in which this could ever happen would be when the MTA (not LISERV) fails to deliver the message. However, there are several cases where a sequence number will be reused:

- If the site administrator deletes or temporarily renames PERMVAR.FILE in an attempt to solve an unrelated problem. Deleting PERMVAR.FILE is not supported as a troubleshooting method.
- If the list is migrated to a new host.
- If there is a crash, disk full error, etc., when updating the counter.

14.5.11 SUPPRESS_APPROVED_BY

Use this option to suppress RFC822 "Approved-By:" headers that would normally be generated by LISERV in messages posted through moderated lists.

Section 15 Moderating and Editing Lists through Email

Much of this section is subjective, based on personal experiences during nearly two decades of list ownership, and may not necessarily match your own philosophy of "the way things ought to be." The following sections are offered as one way to run a list, and the author does not mean to assert that the one way offered is the only way. As we seem to say so often, "your mileage may vary."

15.1 List Charters, Welcome Files, and Administrative Updates

One of the most important things you can do as a list owner is make it clear from the outset what policies are in place and will be enforced if it becomes necessary. Due to a potential for controversy, for instance, some lists may require a formal "list charter" by which all subscribers must agree to abide before they are allowed to subscribe. Other lists may be able to get by with a simple welcome file (see below) that spells out basic netiquette, policies on "flaming" and commercial posts, and anything else that seems appropriate (such as how to get in touch with the list owner in an emergency, where the list archives are located, etc.).

It is particularly important on open subscription lists that you make a concerted effort to remind your subscribers on a regular basis of the policies you have set for your list, as well as any other information they need in order to make best use of your list. If you have a great deal of subscriber turnover, it may be necessary to do this every few weeks. You may decide to put together a quarterly or semi-yearly post for more stable lists. Ensure that the subject line is indicative of what the administrative posting is so that there is no question as to whether or not you posted it (even if subscribers don't read it). (Note that you can use the PROBE1 mail template form and automatic address probing to do this.)

15.2 The Role of the List Owner as Moderator

By default, the list owner becomes a moderator of sorts, even if the list in question is neither edited nor officially moderated. This means that, as a list owner, you must be prepared to maintain order if it becomes necessary. At the same time, you must moderate yourself so that you do not alienate users and cause your list and/or host institution to suffer as a result. Thankfully, mailing lists have generally enjoyed relative peace and quiet over the years in comparison to newsgroups, but mailing lists have unique problems of their own.

Lists dedicated to controversial subjects are more likely to become arenas for "flame wars" between subscribers with hard-held and differing opinions than those dedicated to the discussion of popular software packages, but this does not mean that the latter are immune any more than it means that the former are constantly plagued by flames. The example set by you as list owner and as a participating subscriber to the list is perhaps the most important factor in whether or not your list becomes a site known for strife and controversy. In other words, if you appear not to care about whether or not discussion is on topic and/or civilized, no one else will, either. Yet if you become a policeman – the other end of the spectrum – no one will want to subscribe or participate for fear of your wrath. Either way, your list is unlikely to last very long.

The middle ground is, as in most things, the place to be when administering a list. Some call this "firm but fair," letting things go pretty much as they will but stepping in with a wry or gently chiding remark from time to time when exchanges get heated. And they will! Software discussion lists are particularly bad about this when new subscribers ask "frequently-asked questions" (FAQs) and veteran subscribers respond in exasperated fashion with "RTFM!" (Read The Fine Manual) and similar nasty retorts. Good list owner practice at this point is likely to be a good-natured reminder from you that flames belong in private mail, pointing out that new subscribers have no way of knowing that the particular questions they ask have been asked (and answered!) "n" random times before, and possibly adding a link to the list's archives (if they are available on the web) or instruction on how to use the SEARCH command or the web-based archive search feature to look for answers before asking.

Finally, if your mailing list has an international audience, you must be careful to account for language problems and cultural differences. You will need to decide which languages are allowed or not allowed on the list; this should be mentioned shortly in the list abstract or welcome message. Unless the list is specific to one country or is explicitly for discussion in a specific language, the official language will probably be English. As your list grows, some subscribers may object to this decision, arguing that people who have trouble expressing themselves in English should be allowed to use their own language, with the understanding that many people will be unable to understand what they are saying. As the list owner, it will be your call. Usually, the best compromise is to start a separate list for discussions in the new language. However, you must be careful in wording your decision. In multi-lingual cultures, it is usually considered a courtesy to use the other person's language. It is certainly considered rude for people to demand that everyone else should speak their language. Thus, if your native language is English, you will be in a delicate position. To avoid a flame war, you will want to make sure that your decision does not come out as a unilateral demand. Politely suggesting a separate list, and tolerating an occasional non-English posting when the poster genuinely cannot speak English, is often the best course of action.

Another possible source of flame wars is unintended rudeness. It is easy to forget that non native speakers are making an effort every time they post something to the list. People will make mistakes, sometimes appearing rude when they did not mean to, simply because they used the wrong word. Another cause of apparent rudeness is cultural difference. Things which are perfectly normal in one culture can be insulting in another. For instance, ad hominem attacks are perfectly acceptable in some countries. Conversely, referring to other people by their first name ("As Peter said in his last message,...") can be downright insulting in some cultures, where anything short of the full title is at best condescending. But, of course, in other countries the use of the full title is considered sarcastic... There is no middle ground here, because there are too many conflicting cultures and too many languages. The only way to successful cross-cultural communication is through the tolerance of other people's cultural habits, in return for their tolerance of yours.

15.3 The Role of the List Owner as Editor

Edited lists are generally used for the purpose of "full moderation" or for refereed electronic journals or the like, for which random postings from subscribers and/or non-subscribers may not be welcome for general distribution. This places the list owner and any editors in the position of being full-time monitors of what is and is not allowed to go through to the list.

A word of warning to potential list editors: Rules on the Internet are not set in stone. Some people will insist on their right to post without what they will term "censorship" by the list editor. Some will become upset to the point of threatening to report you to your local computing center administrators for abridging their freedom of speech, or (in the U.S.) even threatening to sue you/your institution for an abridgment of their First Amendment rights. It is therefore important to you that you keep a "paper trail" of such complaints in the event that threats become reality and you are asked about them. This common practice in the business world should be common practice in list ownership as well.

Freedom of speech and copyright issues on the Internet have not yet been tested in the courts as of this writing. These are both areas in which list editors and list owners must tread carefully. Always document any problems you may have in these areas.

15.4 Setting Up an Edited List

The "Moderator=" keyword and moderation "load-sharing" are not available in LISTSERV Lite.



Note: L-Soft currently recommends that edited lists be coded with the ",Confirm" parameter to the "Send=" keyword, in other words:

```
* Send= Editor,Confirm
      - or -
* Send= Editor,Hold,Confirm
```

This will help prevent malicious users from forging mail from an editor address in order to get around your moderation settings, by telling LISTSERV to require an "OK" confirmation whenever it receives a posting from an editor address. The "OK" request goes to the editor address, so the forger is stymied.

Some vacation programs and broken mailers have recently been "reflecting" mail back to lists in such a way that the mail appears to be coming from the editor's address (and the mail therefore gets through). Setting the ",Confirm" option will stop this phenomenon as well.

When confirmation is required for Editor postings, please note that the confirmation request always goes to the Editor who posted, even if you have moderation "load-sharing" configured as noted below. Moderation "load-sharing" applies to postings from general users only.

Should you decide that an edited list is the way to go for your particular situation, you need only add the following lines to your list header file:

```
* Send= Editor
* Editor= userid@some.host.edu
```

where "userid@some.host.edu" should be replaced with the network address of the person who will be handling submissions to your list.

There can be multiple editors as well (and multiple Editor= lines, if desirable), and they do not have to be list owners:

```
* Send= Editor
* Editor= alex@reges.org,joe@foo.bar.edu
* Editor= tony@tiger.com
```

Normally, LISTSERV forwards submissions only to the first editor defined by the "Editor=" keyword. In the case above, all submissions would go to the primary list owner.



Note: The first editor CANNOT be an access-level; e.g., you cannot use the notation "Editor= Owner" to define the first editor. LISTSERV requires that the primary editor of a list must be the e-mail address of a real person.

This does not apply to second and subsequent editors. For instance, in order to allow subscribers to post directly but have non-subscriber posts sent to an editor for approval, you can code something like:

```
* Send= Editor
* Editor= alex@reges.org,(MYLIST-L)
```

On a high-volume list, LISTSERV allows you to share the editing load via the "Moderator=" keyword. By default, this keyword is set to the same value as the first editor defined by "Editor=". When you define more network addresses with the "Moderator=" keyword, LISTSERV sends submissions to each moderator in sequence. The difference between the "Editor=" and "Moderator=" keywords lies in the fact that while any editor can post directly to the list, only moderators receive the forwarded submissions from non-editors.

Here is an example of a list with both Editor= and Moderator= keywords defined:

```
* Send= Editor
* Editor= joe@foo.bar.edu,tony@tiger.com,kent@net.police.net
* Moderator= kent@net.police.net,joe@foo.bar.edu
```

This list will "load-share" the editing duties between Kent and Joe. Tony is able to post directly to the list, but will not receive forwarded subscriber posts for editing.



Note: Whereas an Editor is not required to be a Moderator, a Moderator should always be listed as an Editor. LISTSERV currently compares the contents of the "Editor=" and "Moderator=" keywords and consolidates the two sets of parameters if necessary, but coding lists this way is not considered good practice and the "compare/consolidate" feature may be removed in a future upgrade.

If the parameter "All" is coded before any moderator addresses, LISTSERV will send copies of all postings to all moderators, any of whom may approve the message. An example of this would be

```
* Moderator= All,kent@net.police.net,joe@foo.bar.edu
```

Assuming "Send= Editor, Hold", once a message is approved by one of the moderators, any other moderator attempting to approve the same message will be told that an identical message has already been posted to the list.

If "Send= Editor" (e.g., without "Hold"), then if a note is appended or prepended to the edited post, or if the body of the post itself is edited (that is to say, if the body of the

approved message is changed), duplicates are possible. Thus, it is important that the moderators of any list set up this way pay close attention to whether or not the posting has already been approved by another moderator.

15.5 Submitting Subscriber Contributions to an Edited List

By default, LISTSERV forwards subscriber contributions to the Moderator/Editor with the following paragraph prepended to the message body:

Figure 15-1 Example of the "Editor-Header" Prepended By Default

```
This message was originally submitted by JOE@FOO.BAR.COM to the
ACCESS-L list at PEACH.EASE.LSOFT.COM. If you simply forward it
back to the list, using a mail command that generates "Resent-
" fields (ask your local user support or consult the
documentation of your mail program if in doubt), it will be
distributed and the explanations you are now reading will be
removed automatically. If on the other hand you edit the
contributions you receive into a digest, you will have to remove
this paragraph manually. Finally, you should be able to contact
the author of this message by using the normal "reply" function
of your mail program.

----- Message requiring your approval (25 lines) -----
----- [message body]
```

If you leave this paragraph prepended to the message, LISTSERV will strip it off when it processes the message and to all intents and purposes the message will appear to have come directly from the original sender. Warning: If your mail program or client does not generate "Resent-" fields, the forwarded postings will appear to be coming from you rather than from the original sender. See Section 15.6 [Message Approval with Send=Editor, Hold](#) for an alternative if your mail program does not generate these fields.



Note: If you leave the editor-header paragraph on the message, make sure that your mail client or mail server does not insert quoting characters (e.g., ">") at the beginning of all of the lines in the message when you use the forwarding function of your mail program. If this happens then the editor-header will not be stripped from the message.

When you are ready to edit and/or submit the contribution to the list, simply use the "Forward" function of your mail client. You can make any changes you feel are appropriate to the message body, but be sure to read Sections 15.2 [The Role of the List Owner as Moderator](#) and 15.3 [The Role of the List Owner as Editor](#) before deciding.

15.6 Message Approval with Send= Editor, Hold

LISTSERV includes an optional mechanism allowing you to simply "ok" messages which are then posted with all the correct headers. This option is targeted mainly at list moderators who just approve/reject messages, as opposed to people who actually edit the content of messages. The option is also a good choice if you have a mail client that does not insert "Resent-" header lines into forwarded mail.

To activate this feature, code your list `"Send= Editor, Hold"` and be sure that you have defined at least one editor who will be in charge of approving the messages. A copy of the message on "hold" is sent to the editor with minimal instructions (in order to avoid adding a long message before the text needing approval each time).

To approve a message forwarded to you with `"Send= Editor, Hold"`, simply reply to the approval request and type "OK" as the body of your reply. LISTSERV will normally pick up the confirmation request number from the subject line. If there is a problem, LISTSERV may ask you to resend the approval confirmation along with the number. For instance,

OK 6A943D3C

If the message has been in the spool longer than the time-out period (LISTSERV holds these jobs for a minimum of 7 days), you will receive notification that the confirmation number does not match any queued job. If you need to increase the time-out period, you can set a value for the "Confirm-Delay=" list header keyword that is greater than 168h, but please read the section on "Confirm-Delay=" in the [List Keyword Reference](#) document before doing so.

If you do not want the message to be forwarded on to the list, you need not do anything. The message will expire automatically at the end of the time-out period and will be deleted from the queue.

15.7 Using List Topics

List topics are not available in LISTSERV Lite.

List topics provide powerful "sub-list" capabilities to a list. When properly set up and used, topics give subscribers the ability to receive list postings in a selective manner, based on the beginning of the "Subject:" line of the mail header. It is important to note the following points about topics:

- Topics are best employed on moderated lists. This makes it possible to review the "Subject:" header line to make sure that it conforms to one or more of the topics defined for the list before you forward the post to the list. Not only does this help catch simple errors (such as misspellings of the topic), but it also allows the moderator to add a topic into the subject line if one is not already there.
- If you employ topics on unmoderated lists, your subscribers must be well-educated in their use. Otherwise, there is no point in using them. Messages that do not conform to a specified topic are lumped into the reserved topic "Other" and are distributed only to subscribers who have explicitly defined "Other" as a topic they wish to receive. Therefore, some subscribers will receive the message and some won't, and it is problematic as to whether the message will actually reach the entire audience for which it is intended.
- It is important to note that topics are active only when the subscriber's subscription is set to MAIL. All messages posted to the list, regardless of topic, are included in the digest and/or index for the list (if available) because the same digest/index is prepared and sent to all the digest/index subscribers. Similarly, all messages posted to the list are archived in the list's notebook logs (if available), making it possible for subscribers to retrieve postings in topics they are not set to receive normally.

The basic keyword syntax for defining list topics in the list header file is:

```
* Topics= topic1,topic2,...topic21
```

And the basic syntax used to set topics for users once they have been defined is:

```
SET listname TOPICS: xxx yyy zzz for userid@host
```

where *xxx*, *yyy*, and *zzz* can be:

- A list of all the topics the subscriber wishes to receive. In that case these topics replace any other topics the subscriber may have subscribed to before. For instance, after 'SET XYZ-L TOPICS: NEWS BENCH', the subscriber will receive news and benchmarks, and nothing else.
- Updates to the list of topics the subscriber currently receives. A plus sign indicates a topic that should be added, a minus sign requests the removal of a topic. For instance, "SET XYZ-L TOPICS: +NEWS -BENCH" adds news and removes benchmarks. If a topic name is given without a + or - sign, + is assumed; "SET XYZ-L TOPICS: +NEWS BENCH" adds news and benchmarks. The first topic name must have the plus sign to show that this is an addition, and not a replacement.
- A combination of the above, mostly useful to enable all but a few topics: "SET XYZ-L TOPICS: ALL -MEETINGS".

The colon after the keyword TOPICS: is optional, and TOPICS= is also accepted. The subscriber should not forget to include the special OTHER topic if you want to receive general discussions which were not labeled properly. On the other hand, if the subscriber only wants to receive properly labeled messages it should not be included. ALL does include OTHER.

With the "Default-Topics=" keyword, you can also set default topics for users that will be effective as soon as they subscribe to the list. For instance,

```
* Default-Topics= NEWS,BENCH,OTHER
```

would set the new user to receive topics NEWS, BENCHmarks, and any messages that are incorrectly labeled.

You may get a listing of topics with the number of subscribers who have them set by issuing the command

```
REVIEW listname Short TOPics
```

(if you do not specify Short then the topic listing follows the list of subscribers in the review output). The following is a sample output (assuming you actually have topics enabled; if topics are not enabled then the TOPics option is ignored):

* Topic	Subscribers
* -----	-----
* Apps	1,411
* Backup	1,330
* Beta	951
* Bugs	1,416
* Comm	1,395
* Desktop	1,407
* Hardware	1,401
* Install	1,373

* Internet	1,002
* Network	1,399
* Wish	1,336
*	
* "Other" topic	1,294
* Digest/index subscribers	1,384

See the [List Keyword Reference](#) document under the Topics= keyword for more information on setting up and using list topics.

15.8 The <listname> WELCOME and <listname> FAREWELL Files

When a user subscribes and signs off of a list, LISTSERV looks for list owner-supplied files called *listname.WELCOME* and *listname.FAREWELL*, respectively. If found, it sends the user a copy of the appropriate file in addition to its own administrative message. The WELCOME and FAREWELL files allow the list owner to send a more personal message to the user that can help set the tone for how the list is used. The WELCOME file may contain information about the list charter and netiquette rules, or be simply a message welcoming the user to the list. The FAREWELL file can be used to gather feedback about how the list is serving users.

15.8.1 Creating and Storing the *listname* WELCOME and FAREWELL Files

The procedure differs slightly on VM systems, but the following will work for unix, VMS and Windows systems:

1. Create the file. If you place a "Subject:" line at the top of the document, i.e., as the first line, LISTSERV will pick that line up and use it as the RFC822 "Subject:" header line. Otherwise, LISTSERV places a generic subject line in the mail message.
2. If the file contains special characters (i.e., non-7-bit ASCII characters) and you want to specify a character set for LISTSERV to include in the headers of the message, place a line such as:

Character-Set: ISO-8859-7

at the top of the message (or directly following the "Subject:" line if one is configured). The value "ISO-8859-7" is used here as an example only and should be replaced with the appropriate character set descriptor. If the file does not contain any non-7-bit ASCII characters, this line will be ignored.

3. Be sure that you have defined a "personal password" to LISTSERV with the PW ADD command before you PUT the welcome file. If you have done this but can't remember the password, send LISTSERV a PW RESET command. You will then be able to add a new password with the PW ADD command.
4. Send the file to LISTSERV with a PUT listname WELCOME PW=XXXXXXXXX command at the top of the file, just as if you were putting the list itself. Replace XXXXXXXXX with your personal password.

The variation for VM systems is that the LISTSERV maintainer will have to create a fileid for the file before you can PUT it on the server. Contact the LISTSERV maintainer before trying to store your WELCOME and/or FAREWELL files.

Here is the format of a very simple WELCOME file. (Note that the FAREWELL file is created and stored in an identical manner.)

Figure 15-2 Sample WELCOME File

```
PUT SONGTALK WELCOME PW=XXXXXXXXX
Subject: Welcome to Songtalk!
Welcome to Songtalk, the list for Songwriters talking about
their work.
Your list owner is Susan Lowell (susan@example.com).
```

15.8.2 Using the *listname* WELCOME File as a Moderation Tool

The WELCOME file should contain information geared toward orienting the new subscriber to several areas. The outline of a suggested WELCOME file follows:

- The revision date for the WELCOME file.
- A heading including the short and long names of the list, along with the name and network address of the primary list owner (or the list owner who handles subscription issues/problems).
- Any warnings about the list that you want people to see immediately. These might include
 - a notice regarding the volume of mail subscribers can expect from the list
 - any newsgroups that echo the list
 - ftp sites for the list
 - where to send LISTSERV commands
 - where to find more in-depth information about the list (if you do a quarterly administrative posting or have a FAQ, where can it be found?)
- A short abstract of what the list is all about. This might be a duplicate of the description you send to NEW-LIST.
- The author includes the following paragraph at this point:

Users new to the use of L-Soft's LISTSERV are encouraged to read the online files LISTSERV REFCARD and LISTSERV GENINTRO, which can be obtained by sending the following commands in the body of a mail message to LISTSERV@LISTSERV.NET:

```
INFO REFCARD
INFO GENINTRO
```

- Any guidelines for use of the list, including the list charter if you have one.
- Information about the notebook archives and how to retrieve them.
- Other list-specific information that might be important to new users.

Naturally, list owners should write WELCOME files based on their own experience of what is needed. A WELCOME file should not be static – review it once in a while to ensure that it continues to meet the needs of new subscribers.

15.8.3 Using the *listname* FAREWELL File as a Feedback Tool

The following FAREWELL file was anciently used on the ACCESS-L list on PEACH.EASE.LSOFT.COM, and was intended as a tool to get feedback from users. When it was in use ACCESS-L's list owner typically received 3-5 responses to this message each week.

Figure 15-3 FAREWELL File Used as a Feedback Tool

```
Subject: Your ACCESS-L Signoff Request
I'm sorry to see that you're leaving ACCESS-L.  If there is
anything you believe ACCESS-L should have offered but didn't,
or there are any other suggestions you may have for the list,
please feel free to write directly to me.

Sincerely,
Nathan Brindle <nathan@indycms.iupui.edu>
ACCESS-L List Owner
```

15.8.4 The Alternative to Using WELCOME and FAREWELL Files

It is possible to modify LISTSERV's default mail template so that only one message is sent to users when they subscribe and unsubscribe, rather than an administrative message from LISTSERV and a WELCOME or FAREWELL file from the list owner. See Section 18 [Working with Mail and Web Templates Using Email](#) for the details on modifying the default mail templates.

However, it is likely that the average list owner will prefer to use the WELCOME and FAREWELL files over changing the more-complicated templates. Thus both avenues are provided, and may be used depending on the list owner's level of comfort.

15.9 Social Conventions (“Netiquette”)

Like so many other things, network users tend to expend a great deal of virtual gunpowder about the subject of etiquette on the network (otherwise known as netiquette). Part of the culture of the network is built on the fact that an individual user can put forward any face he or she cares to present. Thus, over time, the network has evolved various sets of rules that attempt to govern conduct. To avoid taking up a great deal of space arguing the merits of differing systems of netiquette, the following general pointers that should be accepted by most users are offered for the convenience of the list owner.

15.9.1 Recognizing and Accepting Cultural and Linguistic Differences

The Internet is international, and while English is generally accepted as the common language of the network, list owners and list subscribers cannot afford to take the position that everyone on the Internet understands English well. In a medium that is invariably connected to language, special understanding is required to deal with questions or statements from people for whom English is not the primary tongue. Often today (at least in the US) a person's first sustained interaction with others on an international basis is via the Internet. It is imperative that this interaction be on the highest level of cordiality and respect from the outset in order for all concerned to benefit.

Additionally, care should be taken when using local idiom and slang. A common word or phrase used by Americans in everyday speech, for instance, might be taken as profanity or insult by those in other English-speaking countries, and may not be understood at all by non-native speakers of English. When a list has a high international readership, it is probably best to avoid non-standard English so as to provide the clearest and least-objectionable exchange of ideas.

15.9.2 Private Mail Should Dictate Private Responses

If someone on a mailing list has sent a private message to you (i.e., not to the list at large) and you have lost that person's address but want to respond, do not post private mail to the list. The REVIEW command will give you a copy of the list membership that you can search for the person's address. If this approach does not work, contact the local postmaster or the list owner for help.

15.9.3 Flaming is (Usually) Inappropriate

Flames (insults) belong in private mail, if they belong in mail at all. Discussions will often result in disagreements. Rebuttals to another person's opinions or beliefs should always be made in a rational, logical and mature manner, whether they are made publicly or privately. What is a flame can range from the obvious (ranting and raving, abusive comments, etc.) to the not-so-obvious (comments about how many "newbies" seem to be on the list these days, "RTFM!" exhortations, etc.).

15.9.4 Foul Language

Subscribers should refrain from abusive or derogatory language that might be considered questionable by even the most liberal and open-minded of networkers. If you wouldn't say it in front of your mother, don't say it in electronic mail.

15.9.5 Unsolicited Advertising and Chain Letters

Most of these are contrary to appropriate use policies governing the use of the poster's Internet access provider. Not only that, they are annoying and (in the case of chain letters) often illegal. See Section 6.10 on the subject of "spamming" for more details.

15.9.6 Other Disruptive or Abusive Behavior

Self-explanatory. It is rarely possible to catalog all forms of anti-social network behavior. Be sure that you as a list owner cover as many bases as you think necessary when promulgating a code of netiquette for your list. Then, be sure to adhere to it yourself.

15.10 Spamming: "Don't Feed the Trolls"

"Spamming" is a network term invented to describe the act of cross-posting the same message to as many newsgroups and/or mailing lists as possible, whether or not the message is germane to the stated topic of the newsgroups or mailing lists that are being targeted. A "spam" is defined therefore as either (1) a specific act of spamming, such as the original so-called "Green Card Spam", or (2) the message that actually comes to your list as a result of someone initiating a specific act of spamming ("The message you just

saw was a spam, and it should be ignored"). Spams are fairly easy to recognize at a quick glance; they often have "To:" fields directed to large numbers of lists, usually in alphabetical order.

If a spam gets through to your list, it will probably engender sarcastic replies (often with the spam quoted in its entirety) – and if your list is coded "Reply-To= List", they will likely come back to the list. It is therefore imperative that you make subscribers aware that when a spam occurs:

- The person responsible for the spam is probably not subscribed to the list, and any response back to the list will not reach that person.
- An appropriate response to a spam is to forward a single copy of the spam to the person in charge of the site from which the spam originated ("POSTMASTER", "ROOT", etc.) pointing out that the spammer is probably violating his site's appropriate use policies.
- It is inappropriate to attempt to flood the spammer's mailbox with network mail in response. This is probably in violation of your network's appropriate use policies, and it just wastes bandwidth.

Perhaps the best policy an individual subscriber can adopt toward spammers is simply to ignore them and allow list owners and newsgroup moderators to take care of the problem. If this does not work and subscribers send their complaints to the list anyway, it might be a good idea to moderate the list for a few days until the furor dies down.

LISTSERV attempts to detect "spams" using a variety of proprietary methods. When LISTSERV decides that a message is a spam, it locks out the user for 48 hours, worldwide in the case of registered LISTSERV networked servers. While locked the user is still able to use LISTSERV normally and to post to mailing lists, but all messages will be forwarded to the list owners for human verification. The user is informed that this has happened but is not informed of which lists caught the message and which didn't, denying him any idea how successful he has been.

L-Soft will not document how LISTSERV decides a message is a spam because the point has been reached where a number of authors are writing and selling books detailing how to avoid such precautions. If L-Soft were to document its methods, the next editions of these books would simply include updated instructions on how to bypass them.

If you are interested in a discussion of the phenomenon of SPAM, you can join the SPAM-L mailing list on LISTSERV@PEACH.EASE.LSOFT.COM.

15.11 Considerations for Appropriate Use Policies

As a list owner, it is important that you take into consideration any appropriate use policies that might apply to your list. For instance, if your list is hosted by an educational site that has a policy restricting mail with commercial content from being sent out by its users, your list will technically be in violation of that policy if it distributes mail from users advertising commercial services. You would be well advised to request a copy of the appropriate use policy (if any) from your host site and make sure that your subscribers are aware of it by including pertinent sections in your WELCOME file and/or your administrative postings.

Host sites are not the only entities that might have appropriate use policies. The network your host is a part of may have such policies as well.

Section 16 Managing Archives using Email

16.1 List Archives

The list archive consists of all of the notebook logs for your list. (If your list is coded "Notebook= No", then it does not have a list archive, of course.) Users can find out what notebook logs are available for a specific list by sending the command `INDEX listname` to the appropriate LISTSERV host.

16.1.1 Setting Up Archive Notebooks

If your list is coded "Notebook= No", you should consult your LISTSERV maintainer to change the keyword to create list archive notebooks. The LISTSERV maintainer is the only one who can change this setting and specify where the notebook should be kept. Also, depending on local policies, you may or may not be allowed to archive your list, or keep more than a few months' or weeks' worth of archives available at a given time.

16.1.2 Indexing Available Archive Notebooks

To find out what archive notebooks are available for your list, simply send the `INDEX listname` command to LISTSERV.

16.1.3 Deleting Existing Archive Notebooks by Email

If your list is set to `Notebook=Yes`, you may occasionally need to delete archive notebook files to conserve disk space or to prevent access to out-of-date information.

First, you may need to find out what archive notebook files are available for your list. Send the following command:

```
INDEX listname
```

LISTSERV will then send a list of all the files associated with your list. For example:

Figure 16-1 Deleting an Archive Notebook File by Email

```
*
* Archive files for the MYLIST list at LISTSERV.EXAMPLE.COM
* (monthly logs)
*
* filename filetype      GET PUT size (bytes) date      time
* -----
MYLIST LOG0209          LOG OWN      3,198 2002-09-28 18:45:25
MYLIST LOG0210          LOG OWN    50,837 2002-10-11 16:29:09
MYLIST LOG0211          LOG OWN    34,639 2002-11-30 19:49:34
MYLIST LOG0212          LOG OWN    95,669 2002-12-26 09:30:16
```

If you wish to retain a copy of the archive file, you may request a copy of it with the `GET` command before you delete it. For instance:

```
GET MYLIST LOG0209
```


To delete an archive notebook file, simply execute a `PUT` operation for the notebook in question without sending any other text along with the `PUT` command line (turn off automatic signatures, and do not use a mail service that automatically adds text to every message sent). For instance, send an email message with the following text as the only content in the body of the message:

```
PUT MYLIST LOG0209 PW=yourpassword
```

This command without any other additional text would delete MYLIST LOG0209 (the file containing the September 2002 archives of the MYLIST list) from the server.

If you provide a digest for your list, you should not delete the current archive file.

Two important issues:

- This command **MUST** be issued by e-mail. It cannot be issued via the "Execute a LISTSERV command" facility of the Web Interface.
- You **MUST** turn off your signature file (if one is enabled in your mail client) in order to successfully delete files. If you do not, LISTSERV will store your signature file in place of the file you are trying to delete instead of deleting the file.

16.2 File Archives

There are three file server systems currently in use by various versions of LISTSERV:

- The VM (mainframe) version of LISTSERV continues to support the "traditional" file server system. While it is very powerful, this file server system dates back to 1986 and suffers from a few annoying limitations. In addition, it is written in a non portable language. This will be replaced eventually with the "new" file server system, currently under development.
- The non-VM versions of LISTSERV 1.8d enhanced further the new file server system introduced in non-VM 1.8c, which included most of the functionality of the "traditional" file system. Notably, GIVE and file "packages" became available. Most end user commands continue to work as before. However, there is no guarantee that the internal data files manipulated by the file server functions will remain as before. Note that SITE.CATALOG files from versions 1.8a through 1.8c are still supported and will not need to be changed in order to work with 1.8d and later.
- The non-VM versions of LISTSERV 1.8a and 1.8b supported a "temporary" file server system, to provide an interim solution while the new system was being developed. This temporary system supports only a subset of the functions of the traditional system. This system is no longer supported by L-Soft as it has been superseded by the new non-VM file server referenced above.

In general, the three systems are compatible, with the understanding that the temporary system does not include all the possible options. However, the mechanism for registering files (defining them to the file server system) is different.

Since the first and third systems will eventually be replaced by the second system, rather than providing an exhaustive section detailing all filelist aspects from the management side, we have provided only a basic overview of the two systems currently in the field, with pointers to where further information may be obtained.

16.2.1 What is the File Archive?

The file archive consists of all files other than notebook logs that have been stored on the LISTSERV host for your list. Users can find out what files are available for a specific list by sending the command `INDEX listname` to the appropriate LISTSERV host.

16.2.2 Starting a File Archive for your List

16.2.2.1 On VM Systems ONLY

With the traditional system (running on the VM servers), the LISTSERV maintainer creates files called "xxxx FILELIST", which contain definitions for all the files belonging to a particular archive. These FILELIST files must be created by the LISTSERV maintainer at the site before they can be edited by the list owner.¹

16.2.2.2 On Workstation and PC Systems

The LISTSERV maintainer creates a definition for your `listname.CATALOG` in a system-global file called `SITE.CATALOG`. The list owner then follows the instructions in Section 16.2.4 [The listname.CATALOG System on non-VM Systems](#) to register files and store them on the server.



Note: The instructions in Section 16.2.3 [Filelist Maintenance \(VM Systems Only\)](#) and the instructions in Section 16.2.4 [The listname.CATALOG System on non-VM Systems](#) are not interchangeable. If you are not sure which system your list is running on, then you can send the command `RELEASE` to the server to find out.

16.2.3 Filelist Maintenance (VM Systems Only)

If your list is running on LISTSERV under unix, Windows, or VMS, please skip this section as it does not pertain in any way to your implementation of LISTSERV.

Maintaining the filelist for your archive is not difficult. It requires only that you have a working knowledge of VM XEDIT (or your local system's editor) and understand how to send files via email.

16.2.3.1 Retrieving the Filelist

To retrieve your filelist in an editable format, send the command

```
GET listname FILELIST PW=XXXXXXXX (CTL
```

to the LISTSERV host where the filelist is stored. The `(CTL` switch causes LISTSERV to lock the filelist until you store it again or explicitly unlock it with an `UNLOCK listname FILELIST` command. (If you don't want to lock the filelist, use `(CTL NOLOCK` instead.) If your mail account is not located on the same host as LISTSERV, you will need to provide your personal password (same as your password for getting and putting your lists).

A filelist retrieved with the `(CTL` option does not look like the filelist you get with an `INDEX` command. A sample `(CTL` option filelist appears below:

1. If you are interested in the mechanics of starting a VM-type filelist, the best reference is "Setting Up the LISTSERV File Server--A Beginner's Guide" by Ben Chi (bec@albany.edu). This publication is available from LISTSERV@ALBANY.EDU as FSV GUIDE.

Figure 16-2 Sample Filelist Retrieved with (CTL Option

* Files associated with MYLIST and available to subscribers:								
		rec			last - change			
* filename	filetype	GET	PUT	-fm	lrecl	nrecs	date	time Remarks
* -----								
MYLIST	POLICY	ALL	OWN	V	79	45	94/03/16	12:04:23 Mission Statement
MYLIST	BOOKLIST	ALL	OWN	V	79	177	94/04/19	16:24:57 Books of interest
MYLIST	QUARTER	ALL	OWN	V	73	113	95/03/11	08:57:04 Quarterly posting
* Listowner's files (not public)								
MYLIST	FAREWELL	OWN	OWN	V	78	9	95/03/11	08:53:41 Goodbye memo
MYLIST	WELCOME	OWN	OWN	V	73	105	95/03/11	09:14:38 Hello memo



Note: The filelist does not include the comment lines you would normally see at the top of an INDEX filelist; nor does it include any notebook archives. LISTSERV creates these lines dynamically at the time the INDEX command is received from a user. If the filelist you have retrieved has any of this kind of material in it, either (a) you have not retrieved the filelist correctly, or (b) you or someone else has stored the filelist previously with this material included. If you did a GET with (CTL, you should be able to remove these extraneous lines by simply deleting them.

If you do an INDEX of your archive and it has (for instance) two sets of comment lines or duplicate notebook archive listings, then you should GET the filelist with (CTL and edit out the offending lines. While the extra lines will not affect the operation of the file server, they are a source of potential confusion for your users.

16.2.3.2 Adding File Descriptors to the Filelist

"Adding a file to a filelist" is not exactly accurate terminology, although it is a widely-used phrase. Adding files to file archives is a two-step process. First, add a file descriptor to the appropriate filelist and store the filelist on the server. Second, store the file itself on the server.

To add a file descriptor, start a line with a space and then type in your file's name, access codes, five dots (periods) and a short description, each separated by a space. For example:

```
MYLIST FAQ ALL OWN . . . . . Frequently-Asked Questions for
MYLIST
```



Note: The line must begin with a space. Also, you must place five dots separated by spaces between the PUT file access code (here it is OWN) and the short description. These dots are place holders for the record format (recfm), longest record length (lrecl), number of records (nrecs), and the date and time of the last update. If these dots are not present, LISTSERV will return an error message when you try to store the filelist.

You will note that the line you have just added does not look like the other lines in the filelist. Ignore the "pretty" formatting. LISTSERV will reformat the information for you. After adding the line, your filelist should look like this:

Figure 16-3 Adding a File Descriptor to the Filelist

* Files associated with MYLIST and available to subscribers:									
		rec			last - change				
* filename	filetype	GET	PUT	-fm	lrecl	nrecs	date	time	Remarks
* -----									
MYLIST	POLICY	ALL	OWN	V	79	45	94/03/16	12:04:23	Mission Statement
MYLIST	BOOKLIST	ALL	OWN	V	79	177	94/04/19	16:24:57	Books of interest
MYLIST	QUARTER	ALL	OWN	V	73	113	95/03/11	08:57:04	Quarterly posting
MYLIST	FAQ	ALL	OWN	Frequently-Asked Questions for MYLIST
* Listowner's files (not public)									
MYLIST	FAREWELL	OWN	OWN	V	78	9	95/03/11	08:53:41	Goodbye memo
MYLIST	WELCOME	OWN	OWN	V	73	105	95/03/11	09:14:38	Hello memo



Note: You can add comment lines to the filelist by placing an asterisk in the left-most column instead of a space. Comment lines can act as indexes, descriptions, or pointers to other resources.

Once you are finished adding file descriptors, save the filelist to disk.

16.2.3.3 File Access Codes (FAC) for User Access

FACs define which users have access to files in the file archive. The FAC for `GET` indicates who may retrieve the files, and the FAC for `PUT` indicates who may store the files on the server.



Note: Some special FACs exist for "superusers" such as the `LISTSERV` maintainer(s) and the `LISTSERV` Master Coordinator, who may `GET` and `PUT` any file regardless of its `GET/PUT` permissions.

The basic FAC codes that are always available for the VM server are:

- `ALL` – Universal access.
- `PRV` – Only members of the associated mailing list have access.
- `OWN` – Only the owners of the associated mailing list have access.



Notes: The FAC codes `PRV` and `OWN` work only on the VM filelist system. They do not work on the non-VM catalog system. See Section 16.2.4 [The listname.CATALOG System on non-VM Systems](#) if you are configuring the non-VM systems.

This assumes the name of the filelist is identical to the name of the associated mailing list – for instance, `MYLIST@FOO.BAR.EDU` would have a `MYLIST` `LIST` file and a `MYLIST` `FILELIST` file. Ask your `LISTSERV` maintainer for assistance if this is not the case or if you need special FACs added for special user access to files.

16.2.3.4 Deleting File Descriptors from the Filelist



Note: Before you delete file descriptors from the filelist, you should delete the files themselves from `LISTSERV`'s archive disk. See Section 16.2.6 [Deleting Files from the Host Machine](#) for instructions.

If this step is not followed, `LISTSERV` may not be able to find the file you want to delete after you edit the filelist and store it.

16.2.3.5 Storing the Filelist

1. Create a mail message to `LISTSERV` at the appropriate host. (Sending a filelist to `LISTSERV@LISTSERV.NET` will not work. The filelist must be sent to the host it resides on.)
2. Include the filelist file as plain text in the body of the mail message. Do not attach it with MIME or another encoding scheme, as `LISTSERV` does not translate encoded messages.
3. Make sure that your mail client does not automatically add a signature file to the bottom of your mail. If it does, your signature file will be treated as part of the filelist and will be stored along with it.
4. At the top of the filelist, add a single line as follows:

```
PUT filename FILELIST PW=XXXXXXXX
```

where `XXXXXXXX` is your personal password for `LISTSERV` on that host. Note that this is similar to the `PUT` command used when storing the list file.

5. Send the filelist to `LISTSERV`.

Once `LISTSERV` acknowledges the receipt and storage of the filelist, you can send the files that correspond to the file descriptors in your filelist. See Section 16.2.5 [Storing Files on the Host Machine](#) for instructions.

16.2.4 The listname.CATALOG System on non-VM Systems

List-level file catalogs are not available in `LISTSERV LITE`; you must register files in the `SITE.CATALOG` file per the instructions in the installation guide.

`LISTSERV` running on non-VM systems has a file archive registration system similar to (but differing in important respects from) the old VM `FILELIST` system.

This "sub-catalog" enhancement allows the `LISTSERV` administrator to delegate file management authority in a controlled and secure manner. Multiple list owners can be given the authority to maintain their own sub-catalog in a predefined directory.

From a list owner's point of view, the procedure works as follows:

1. Ask the `LISTSERV` administrator to create the sub-catalog for your list. You will need to provide the email addresses of the person(s) who will be in charge of managing it ("catalog owners").
2. The catalog owners use the `GET` and `PUT` commands to update their catalog and register new files in their directory. Each file has the usual `GET` and `PUT` file access codes, allowing the catalog owners to further delegate the management of individual files to third parties ("file owners").
3. The file owners manage the files in question using the `GET` and `PUT` commands. Authorized users can retrieve the files using the `GET` command.

If your list is being migrated from VM to one of the non-VM versions of `LISTSERV`, please note that it is not necessary to create entries in your sub-catalog for `WELCOME`, `FAREWELL`, and `MAILTPL` files. If entries for these files are not created, they simply do not appear in the output of an `INDEX` command. However, if desired, you can force them to appear by defining them in your sub-catalog.

16.2.4.1 Updating the Sub-Catalog

Once the sub-catalog is created, the catalog owner(s) can register new files using the following procedure (in this example, it will be assumed that the sub-catalog is called MY.CATALOG):

1. Send a GET MY.CATALOG command to LISTSERV (or, if the catalog is brand new, start from an empty file).
2. Register new file(s) in the catalog (see below).
3. Use the PUT MY.CATALOG PW=XXXXXX command to store the updated catalog.

Alternatively, if the catalog owner has an account on the LISTSERV host system and write access to the directory associated with the sub-catalog, the file can be edited directly. Note however that, in that case, the LISTSERV-ISP quota system will be inoperative as it has no control over disk accesses which do not go through LISTSERV itself.

The format of sub-catalogs is similar to that of SITE.CATALOG:

MY.FILE	my.file	ALL JOE@XYZ.COM
(1)	(2)	(3) (4)



Notes: (1) This defines the name of the file as seen by LISTSERV users. That is, the command to retrieve the file will be GET MY.FILE.

(2) This defines the name of the actual disk file where the contents of MY.FILE will be stored. Normally, you should specify the same as (1), or just an equal sign (LISTSERV will then substitute the name you provided for (1)). However, in some cases you may want to make a particular file available under multiple names. This can be done by registering multiple files (i.e. multiple values for (1)), and using the same (2) value every time.

(3) This file access code determines who can order the file through a GET command. The following file access codes are available:

ALL – universal access.

PRIVATE (xxx) – Only members of the xxx list have access.

OWNER (xxx) – Only the owners of the xxx list have access.

SERVICE (xxx) – Only users in the service area of the xxx list have access.

NOTEBOOK (xxx) – Same access as the archives of the xxx list.

user@host – The user in question is granted access.

Except for ALL, which must occur on its own, multiple file access code entries can be specified, separated by a comma with no intervening space. For instance:

```
MY.FILE C:\FILES\XYZ\MY.FILE JOE@XYZ.EDU,JACK@XYZ.EDU,PRIVATE(XYZ-L) CTL
```

defines a file that Joe, Jack and the subscribers of the XYZ-L list can order via the GET command, but that only the LISTSERV administrator can update.

(4) This file access code determines who can update the file with the PUT command.

See note (3), above, for more information on FAC codes.

(2) defaults to the value of (1), and (3) and (4) default to the GET and PUT access codes of the sub-catalog itself, respectively. So, in most cases a sub-catalog entry will be as simple as:

```
MY.FILE
```

Additionally, comment lines (starting with an asterisk) or blank lines can be interspersed with file definitions. These comments will be echoed when the sub-catalog is indexed (see below), in sequence with the file definitions. For instance, your catalog could read:

```
*
* Files for the XYZ sub-project
*
XYZ.AGENDA
XYZ.BUDGET
XYZ.PROPOSAL-1
XYZ.PROPOSAL-2
```

16.2.4.2 Indexing the Sub-Catalog

If MY.CATALOG is defined as:

```
MY.CATALOG      /home/lists/xyz/my.catalog      xxx JOE@XYZ.COM
```

then any user who matches the 'xxx' file access code is authorized to issue an INDEX MY command to get a formatted version of the catalog. For compatibility with older versions of LISTSERV, GET MY.FILELIST will produce the same results. If there is a mailing list called MY, a list of the archive files will be appended automatically.

16.2.5 Storing Files on the Host Machine

To store a file on any LISTSERV host, first ensure that it has been registered with an entry in a filelist or catalog. Then follow these instructions:

- Be sure that you have defined a "personal password" to LISTSERV with the PW ADD command before you PUT the new or edited file. If you have done this but can't remember the password, send a PW RESET command to LISTSERV, then a new PW ADD command.
- Edit your file and save it. Add a single line at the top of the file as follows (square brackets indicate optional parameters):

```
PUT filename extension [filelist|catalogname] PW=XXXXXXXX
```

(This line will not appear to people who GET the file from LISTSERV.) Replace XXXXXXXX with your personal password. If you specify the filelist or catalog name, do not put the square brackets around the name.

There are a couple of issues that need to be noted here:

- If the file you are going to store is registered in the sitewide catalog or filelist, do not specify the name of the catalog or filelist.
- If the file you are going to store is registered in a sub-catalog or filelist other than the sitewide one, you may have to specify the name of the sub-catalog or filelist in order to be able to store the file. This is because it is entirely possible that two lower-level filelists or catalogs may have files registered with the same name (for instance, README TXT). If LISTSERV has two sub-catalogs registered (for instance, MYLIST CATALOG and HISLIST CATALOG) that both have a file called README TXT registered, then a PUT README TXT command will tell LISTSERV to try and store the file in the first catalog it comes to in the hierarchy. If MYLIST CATALOG is registered before HISLIST CATALOG in SITE CATALOG, LISTSERV will try to store the file as if it belonged to MYLIST (which we assume is what you want). However, if HISLIST CATALOG is registered before MYLIST CATALOG (and many sites like to keep things in alphabetical order, so this is a most likely scenario), LISTSERV will try to store the file as if it belonged to HISLIST, and you will get an error stating that you aren't allowed to store the file.
- Be sure that the file has been registered with an entry in a filelist or the site catalog.
- Send the mail message to LISTSERV.

16.2.6 Deleting Files from the Host Machine

To delete a registered file on any LISTSERV host:

1. Be sure that you have defined a "personal password" to LISTSERV with the PW ADD command before you PUT the delete job. If you have done this but can't remember the password, send a PW RESET command to LISTSERV, then a new PW ADD command.
2. Create a new mail message addressed to LISTSERV. Add a single line at the top of the message as follows:

```
PUT filename extension [filelist|catalogname] PW=XXXXXXXX
```

(Replace XXXXXXXXX with your personal password.) The same issues noted in Section 16.2.5 [Storing Files on the Host Machine](#) regarding the filelist/catalog name are operative here.
3. Send the mail message to LISTSERV.
4. LISTSERV will tell you that the file has been successfully deleted.
5. For VM Systems ONLY: GET the *listname* FILELIST for your list and delete the line for the file you've just deleted. PUT the *listname* FILELIST back on the server.
6. For Workstation and PC Systems ONLY: Get the *listname*.CATALOG for your list and delete the line for the file you've just deleted. PUT the *listname*.CATALOG back on the server. (This is not necessarily required since under non-VM, if the physical file does not exist, LISTSERV will not include it in the output of an INDEX command. This is primarily a housekeeping measure.)



Note: Number 5 and number 6 are not necessary when you are deleting notebook archives. LISTSERV generates the notebook archives index "on the fly" when needed.

16.2.7 Automatic File Distribution (AFD) and File Update Information (FUI)

If your list is running on LISTSERV under unix, Windows, or VMS, please skip the rest of this section as it does not currently pertain in any way to your implementation of LISTSERV.

AFD and FUI have not yet been ported to the workstation and PC environments. However, this feature is supported on VM and will be supported in the near future on the other platforms.

These two features are similar in their command syntax, but do different things. AFD provides a method whereby users may subscribe to specific files, which will be sent to them any time the files are updated. For instance, if you have a FAQ file that is updated monthly, a user could send an AFD subscription to that FAQ file and LISTSERV would send it to the user every time you updated and stored the FAQ.

FUI, on the other hand, is a method whereby a user subscribes to a file but receives only a notification that the file has been updated. The user can then GET the file at his own discretion.

AFD and FUI can be password-protected to protect users from network hackers who might forge mail from the user subscribing him to large or frequently-updated files. If a password is not provided in an AFD or FUI ADD command, LISTSERV warns the user that it would be a good idea to password protect the subscription.

16.2.8 File "Packages"

You can define a group of files as a "package" that can be retrieved by users with a single GET command. First, ensure that all the files in the package are defined in the appropriate filelist and stored on the server as detailed above.

Next, create a file descriptor in the appropriate filelist or catalog for a file called *filename* \$PACKAGE (or *filename*. \$PACKAGE for non-VM), where filename is the name you have chosen for the group of files. Be sure that the filetype or extension is \$PACKAGE, with a leading \$ sign, and store your filelist.

Now create the actual filename \$PACKAGE file. At the top of the file you can insert comment lines beginning with asterisks, e.g.:

```
* MYLIST $PACKAGE
* Packing list for MYLIST PACKAGE
*
* You can make other comments here, such as
* the contact email address.
*
* filename filetype filelist
*=====
```

Following these comment lines, you insert lines for each of the files contained in the package. There are two ways to format entries in your \$PACKAGE file:

- A "compatibility" mode that works on all platforms, and which is identical to the original method used on VM (and which VM servers still must use). In the compatibility mode the basic format for the entries is

```
filename filetype filelist <optional_comments>
```

for example,

```
MYLIST    $PACKAGE MYLIST The packing list
INTEREST  FILE      MYLIST Interest groups
NETIQUET  FILE      MYLIST How to behave
ANOTHER   FILE      MYLIST No comment
```

- In the second (new) mode for non-VM servers only, the entries are formatted like this:

```
filename.extension <optional_comments>
```

for example,

```
MYLIST.$PACKAGE The packing list
INTEREST.FILE   Interest groups
NETIQUET.FILE   How to behave
ANOTHER.FILE    No comment
```



Note: Anything that is not the name of a file in the package must be commented out with an asterisk in the leftmost column of the line. It is possible to create a package file without any comment lines at all, but this is not preferable in practice. Often users will get the package file itself just to see what is in it. You should include a reference to the package file itself so that the user will get a copy of the "packing list" to check against the files he receives from LISTSERV.

The final step is to send the package file to LISTSERV like any other file.

Now users can do one of two things:

1. They may get the entire package of files sent to them by sending LISTSERV the command `GET filename PACKAGE` (without the \$ sign); or
2. They may request that LISTSERV send only the package file itself by sending LISTSERV the command `GET filename $PACKAGE` (with the \$ sign).

Packages may be subscribed to with the AFD and FUI commands (VM only).

16.2.9 Where to Find More information on File Archives

LISTSERV maintainers can also find more information in the [Site Manager's Operations Manual](#) for LISTSERV.

Section 17 Using Email to Post a Message

To send a message to the list using your email client program, simply compose a message and address it to the list address in your email client. The list address is the one that sends mail to the list moderator, if there is one, or directly to all the other subscribers. List addresses are formatted with the name of the list followed by the domain address (e.g. listname@someserver.com).

Section 18 Working with Mail and Web Templates Using Email

The majority of Web interface Web pages and administrative messages sent to subscribers can be customized. Customizing your list's Web pages and administrative messages creates an individual look and tone for your list. Subscribers will see Web pages and read administrative messages that are specific to your list, containing pertinent information and instructions rather than just generic pages and messages.

18.1 Using LISTSERV Templates

Templates are used to generate some of the mail LISTSERV sends to users in response to commands it receives. Among these are the "You are now subscribed..." message, the message sent to users when LISTSERV cannot find a subscription for them in a specified list, and others. Note that certain administrative mail (for instance, the response to the STATS and RELEASE commands) is hard-coded into LISTSERV and cannot be changed.

Other templates are used to generate the HTML code used by the web archive and administration interfaces.

18.2 Types of Templates

The template editor allows the site administrator and list owner to customize the majority of the Web Interface Pages and Administrative Messages sent out by LISTSERV. There are two types of templates you can customize – Web and Mail.

18.2.1 Mail Templates

Mail Templates control text produced by LISTSERV itself. Although generally categorized as mail templates, they actually fall into three different types: Mail Templates, Message Templates, and Message Fragments.



Note: Mail-merge fields are now allowed in Mail templates.

18.2.1.1 Mail Templates

A mail template is a complete mail message. All commands are available, substitutions that make sense in the context of the specific message are available, the message may be formatted, and while other templates may be imbedded with the .IM command, the message is in and of itself ready for LISTSERV to send.

18.2.1.2 Message Templates

The next level down is a message template, i.e. a template that by default does not send any mail but supplies text that will ultimately be shown to the user - not always by e-mail, it could be over the web interface for instance.

Typically, message templates that are sent by mail will be "bundled" into a single message. However, a given message template can be forced to send an individual unbundled mail by using the new .SM command.

In a message template, you have the following restrictions:

- The commands .TO/.CC/.RE/.CS are not allowed unless you use .SM.
- Normally there will be limited access to formatting unless stated otherwise in the template. By "limited" is meant that you will, at a minimum, have the capability to break the text into separate paragraphs and to use .FO ON/OFF, although leading, trailing and duplicate blank lines will be removed.
- Unless stated otherwise, .QQ will suppress the message. Note that this does not mean that other related messages are also suppressed, especially if they have already been sent. .QQ only suppresses the template you are currently reading.

18.2.1.3 Message Fragments

The lowest level is a message fragment. It could be a list of months in French, or a common sequence of words that is put in a template not just to make other templates more readable, but to improve consistency (see &MSGREF). This fragment is used in another messages and is not itself a message, hence the following restrictions:

- .SM is not supported in message fragments (and thus other commands related to sending mail, for instance, .TO/.CC, etc. are not supported, either). If .SM is specified in a message fragment, an error similar to the following is raised in the web interface:

>>> Mail template "MSG_SUBSCRIBE_FRAGMENT_OPTCHANGED1" does not support .SM

and no email is actually sent. In the example case, .SM would have to be specified in the message template MSG_SUBSCRIBE_SUCCESS in order for it to succeed.
- .SJ is ignored in message fragments The .SJ command should be issued in the template that sends the actual message, not in the fragment.
- Formatting (.FO ON/OFF) is not allowed. Everything will be internally merged into a single line (this is similar to what was previously called a "linear" template).
- .QQ is not supported in a message fragment.

18.2.2 Web Templates

Web Templates (also referred to as Dynamic Web Templates) control the pages produced by the Web Interface. These pages are produced dynamically when they are accessed. What gets displayed by the browser depends on the circumstances and may change depending on who is accessing the Interface, which list they are accessing, the settings of that list, and so on.

Commands in Web Templates begin with a plus sign "+" and variables begin with an ampersand followed by a plus sign "&+".

18.3 The Default Charset for Mail Templates



When editing mail templates, LISTSERV will tag each mail template with the character set that was in use when the changes were made.

Unedited mail templates will have ISO-8859-1 as the default character set.

18.4 Naming Conventions for Message Templates and Fragments

The following naming convention for message templates and message fragments has been introduced:

`MSG_command_action_name`

- 'command' is the name of the command. For instance, the name of a template that controls list postings starts with "MSG_POSTING_".
- 'action' is some kind of general action or category of action or the like. Every template with the same command and action SHOULD have the same calling conventions, in terms of what is allowed or not allowed, how the results are used, what global variables are available. Of course, each individual template may also have one or more variables that are specific to it. For instance, when a virus is detected, you will have the name of the virus, which is not available when "Sizelim=" is exceeded. If 'action' is something_FRAGMENT, the message is a fragment.
- 'name' is a unique identifier.

18.5 Accessing the Default Template Files

18.5.1 Requesting a Copy

LISTSERV stores its default mail template information in a file called DEFAULT MAILTPL, which can be requested by list owners and LISTSERV maintainers from LISTSERV with the GET command, just like any other file. The LISTSERV maintainer will find this file in LISTSERV's "A" directory (usually `~listserv/home/default.mailtpl` on unix, `LISTSERV\MAIN\DEFAULT.MAILTPL` on Windows systems, and `LISTSERV_ROOT:[MAIN]DEFAULT.MAILTPL` under VMS).



Note: DEFAULT MAILTPL contains the (obsolete) static web interface template forms.

LISTSERV stores its default dynamic web interface template forms in a file called DEFAULT WWWTPL, which can be retrieved in a manner identical to that for DEFAULT MAILTPL.



Note: It is considered unwise (and it is not supported) to modify the contents of DEFAULT MAILTPL or DEFAULT WWWTPL themselves, as these files will be overwritten by upgrades. It is possible to make sitewide changes that will not be overwritten without disturbing either of these files.

18.6 Mail Template Format and Embedded Formatting Commands

18.6.1 Mail Template Format

Each individual template form starts with a form name and subject line, such as:

```
>>> EXAMPLE1 This is the subject line
```

The following instructions are reminders for the form name and subject line:

- The template form starts with the line containing the form name and subject, and ends with the next line starting with '>>>', or at the end of the file.
- The subject line may contain substitutions (such as "&LISTNAME: &WHOM requested to join").
- Ensure that there is a blank space (ASCII 0x20) between '>>>' and the name of the form, or LISTSERV will not recognize the form.
- The names of the template forms must be typed in UPPER CASE.

A template form contains text and, optionally, formatting/editing commands, which start with a period in column 1. All other lines are treated as normal text: sequences starting with an & sign are substituted, then lines are joined together to form a paragraph, which is finally formatted like with any non-WYSIWYG text processor. You can suspend formatting with `.FO OFF` and resume it with `.FO ON`; when formatting is suspended, LISTSERV no longer joins lines to form a paragraph, but simply writes one line of text to the message for each line read from the template form. This makes it possible to include tables or a text-mode logo, but can create seriously imbalanced text if substitutions are used. For instance, a typical &WHOM substitution can range from a dozen characters to 60 or more, even though it only takes up 5 characters on your screen when you enter it.

18.6.2 Common Variable Substitutions

The following substitutions are always available:

- &DATE – Long-style date (04 Jan 1998)
- &TIME – hh:mm:ss
- &WEEKDAY – Three-letter day of the week, in English
- &MYNAMES – The substitution you will use most of the time when you need to refer to LISTSERV. For Internet-only or BITNET-only servers, this will display LISTSERV's only e-mail address. For servers with both Internet and BITNET connectivity, it will say "LISTSERV@hostname (or LISTSERV@nodeid.BITNET)".
- &MYSELF – LISTSERV's address, in the form LISTSERV@XYZ.EDU or, if no Internet hostname is available, LISTSERV@XYZVM1.BITNET.
- &MYNODE – LISTSERV's BITNET nodeid, without the '.BITNET', or its Internet hostname if no NJE address is available.
- &MYHOST – LISTSERV's Internet hostname or, if none is available, its NJE address (with '.BITNET').

- `&MBX(addr)` – Looks up the specified address in LISTSERV's signup file and displays "name <addr>" if a name is available, or just the original address otherwise. This is typically used to give the name of the command originator or target, along with his e-mail address: `&MBX(&WHOM)` or `&MBX(&INVOKER)`. Please note however that `&WHOM` and `&INVOKER` are not always available in every template. The "addr" parameter is always required; `&MBX` by itself is syntactically invalid.
- `&RELEASE` – LISTSERV's release number (e.g., "16.0").
- `&OSTYPE` – The operating system under which LISTSERV is running.
- `&OSNAME` – The full operating system name including the version number, e.g., "VM/ESA 1.2.3", "Windows NT 4.0", "Linux 2.0.27", "SunOS 5.4", etc.
- `&HARDWARE` – The type of machine LISTSERV is running on, e.g., "Pentium (512M)".

The following substitutions are also available for templates related to mailing lists:

- `&LISTNAME` – Either the short or long name of the list based on the value of "List-Address=" and/or its system default. By default the long ("List-ID=") name is used if present.
- `&TITLE` – Title of the list, or empty string.
- `&KWD(kwd)` – Value of the specified keyword for the list. You do not need to specify the name of the list - it is implicit. You need not put quotes around the keyword names either, although quotes will be accepted if present. Optionally, you can specify a second numeric argument to extract just one of the terms of a list header keyword; for instance, if the list header contains "Notebook=Yes,L1,Monthly,Private", `&KWD(NOTEBOOK,4)` has the value "Private". A third argument, also optional, specifies the default value for the keyword in case it was not initialized. It is meant to be used for conditional formatting in the default templates and list owners should not worry about it.
- `&LITE` – Has the value 1 when running the LISTSERV Lite product, and 0 otherwise. This variable can be used to write generic templates that account for the differences between the two products.
- `&LITEFE` – Has the value 1 when running the Free Edition of LISTSERV Lite. Similar to but distinct from `&LITE`.
- `&ISODATE` – Returns today's date in ISO format, i.e., yyyy-mm-dd.
- `&DAYSEQ(n)` – Used to create FAQ templates with rotating topics. May also be used to create bottom banners with rotating text (e.g., for lists with multiple commercial sponsors who get "ad space" in the banner on a rotating basis).

In addition, many template forms have their own specific substitutions, meaningful only in their specific context. For instance, a message informing a user that he was added to a mailing list may have an `&INVOKER` substitution for the address of the person who issued the ADD command. This is not meaningful for a template form intended to inform a user that he must confirm his subscription to a list within 10 days, so it is not generally available. If you attempt to use a substitution which is not available, the template processor writes an error message to the mail message it is generating, but sends it

anyway, in the hope that the recipient will be able to figure out the meaning of the message in spite of the error.



Important: If you need to include a sentence with an ampersand character, you will have to double it to bypass the substitution process, as in "XYZ &&co."

The mail template processor also supports HTML-like variable closure, in addition to the traditional LISTSERV closure (both methods are supported concurrently; there is no need to select one over the other). For example:

- **Traditional** – For more information, please send mail to &EMAIL or call &PHONE.
- **HTML** – For more information, please send mail to &EMAIL; or call &PHONE;.

Previously, HTML writers who used HTML closure conventions would not get the expected results. This change makes it easier for webmasters to get the desired results the first time.

18.6.3 Template Commands

Any line starting with a period in column 1 is processed as a formatting command. Note that neither substitutions nor formatting commands are case sensitive. The table below lists the formatting commands that list owners may need to use.

Table 18-1 Formatting Commands for List Owners

Command	Description
. *	Comment: anything on this line is simply ignored. This is useful for recording changes to template files when there are multiple owners. Just add a comment line with the date and your initials every time you make a change, for the benefit of the other owners.
.CC OFF	Removes all "cc:" message recipients. You can also add message recipients by specifying a series of e-mail addresses after the .CC statement, as in .CC JOE@XYZ.EDU. PC mail users should note that in this context "cc:" is a RFC822 term that stands for "carbon copy". RFC822 messages may have "cc:" recipients in addition to their "primary" recipients. There is no real technical difference between the two, the "cc:" indicator just denotes a message that is being sent for your information. Some administrative messages sent to list owners are copied to the user for their information, and vice-versa; this behavior can be disabled by adding a .CC OFF statement to the template.
.CE text	Centers the text you specify (just the text you typed on the same line as the .CE command). This can be useful to highlight the syntax of a command.
.FO OFF	Turns off formatting: one template line = one line in the final message.
.FO ON	You can resume formatting with .FO ON or .FO RAGGed. (.FO RAGGed requires LISTSERV 1.8e-2002a or later, that is, build date of 31 October 2002 or later.)
.FO RAGGed	Changes right-justified text formatting to left justified text formatting. You can resume right-justified formatting with .FO ON. (.FO RAGGed requires LISTSERV 1.8e-2002a or later, that is, build date of 31 October 2002 or later.)

Command	Description
.QQ	Cancels the message. LISTSERV stops reading the template form and does not send anything. This is useful if you want to completely remove a particular message; note however that this can be confusing with certain commands, as LISTSERV may say "Notification is being sent to the list owners" when in fact nothing will be sent because of the .QQ command in the template form.
.QU	Ends processing of the current template as if you had reached the end, but without cancelling the message. The main purpose is to avoid multi-level nested .BB/.EB conditional blocks (see below) that are hard to keep track of.
.RE OWNERS	Adds a 'Reply-To:' field pointing to the list owners in the header of the generated message. Use this command when you think users are likely to want to reply with a question. You can also use .RE POSTMASTER to direct replies to the LISTSERV administrator, if this is more appropriate.
.TO	Replaces the default recipients of a message with the value specified. For instance, if you use the ADDREQ1 template form to send new subscribers a questionnaire, application form or similar material, you will need to add a '.TO &WHOM' instruction to your modified template form, as by default the user will not receive a copy.

A number of more advanced commands are available to list owners with more sophisticated needs and some programming experience. If you encounter one of these commands in a template, you will probably want to leave it alone.

Table 18-2 Advanced Formatting Commands for List Owners

Command	Description
.ASIS text	<p>Tells LISTSERV to leave the text immediately following the .ASIS directive alone, that is, don't convert "<" and ">" characters into HTML &lt; and &gt; when creating pages. This is specifically for use in HTML templates where it is important not to convert parts of a URL reference. For instance,</p> <pre>.ASIS Click here.</pre> <p>As with the .CE directive, the text you intend to affect with the .ASIS directive must not wrap. The .ASIS directive will only work on text it finds on the same physical line into which it is coded.</p>
.BB cond	<p>Begin conditional block.</p> <p>Related commands are .EB, .ELSE, and .QU . See Section 19.5.4 Conditional Processing for usage.</p> <p>See also .QUIF .</p>

Command	Description
<code>.CS text</code>	<p>Define a (non standard) character set for the template in question, i.e.,</p> <p><code>.CS ISO-8559-7</code></p> <p>This setting is ignored if the template does not actually contain special characters (for instance, if the template is written in 7-bit ASCII). Otherwise the appropriate headers are created for the message in question when it is sent out.</p>
<code>.DD ddname</code>	Copies the contents of the specified DD into the message. This is meaningful only if a DD has been set up by LISTSERV for this purpose. As a rule of thumb, you should either leave these statements unchanged or remove them.
<code>.EB</code>	End conditional block (see <code>.BB</code>).
<code>.ELSE</code>	Conditional ELSE directive (see <code>.BB</code>).
<code>.IM name</code>	<p>Imbeds (inserts) another template form at this point in the message. This is used to avoid duplicating large pieces of text which are mostly identical, such as the templates for "you have been added to list X by Y" and "your subscription to list X has been accepted".</p> <p>As noted below, LISTSERV will not pick up an "imbedded" template form from <code>\$\$SITE\$.MAILTPL</code>. If you wish to include an "imbedded" template form (e.g., <code>\$\$SIGNUP</code>) in <code>\$\$SITE\$.MAILTPL</code>, you must also include the template form that calls it with the <code>.IM</code> command.</p>
<code>.QU</code>	Stop (i.e., QUIT) processing of the current template as if you had reached the end, but without cancelling the message. The main purpose is to avoid multi-level nested <code>.BB/.EB</code> conditional blocks that are hard to keep track of.
<code>.QUIF</code>	<p>(QUIT IF) Similar to <code>.QU</code>, stop processing the current template without cancelling the message, based on the result of an inline conditional comparison. The full syntax is <code>.QUIF var operator value</code></p> <p>For instance, <code>.QUIF &RC = 0</code></p> <p>This single statement is strictly equivalent to the block</p> <pre>.BB &RC = 0 .QU .EB</pre>
<code>.SE var text</code>	Defines or redefines a substitution variable. This is convenient for storing temporary (text) expression results which need to be used several times. Even standard variables such as <code>&LISTNAME</code> can be redefined - at your own risk. You must enclose the text expression in single quotes if you want leading or trailing blanks.

Command	Description
<code>.SJ</code>	<p>Set the subject line for the message. <code>.SJ</code> does two things:</p> <ol style="list-style-type: none"> 1. It overwrites the heading originally obtained from the '>>>' header line starting the template. A mail template uses this heading as the "Subject:" field. A message template, on the other hand, does not use this heading for anything, but it is still changed internally. See <code>.SM</code> below. 2. It leaves an explicit indication to the template invoker that it is desired to change the subject line. For a mail template, this has already happened. For a message template, on the other hand, the invoker is strongly urged to take reasonable steps to make it happen. "Reasonable" is defined by the invoker, not by the customer. <p><code>.SJ</code> is not available for message fragments. The subject line should be set in the template that creates the actual message.</p> <p>One important caveat when not using <code>.SM</code>: Most templates are going to be for command replies. A command reply may be followed by another reply to the same command (e.g. multi-user ADD), or to another command. All these replies are bundled into the same e-mail message sent in answer to the command(s), again unless you use <code>.SM</code>. And this message can only have one subject by definition.</p>
<code>.SM</code>	<p>Send a copy of the message via email, if echoed to the web interface.</p> <p><code>.SM</code> is not available in message fragments and is only available in message templates which do not contain a comment to the effect that <code>.SM</code> is not supported for that particular message template.</p>
<code>.TY text</code>	<p>Types one line of text on the LISTSERV console log. This can be useful to the LISTSERV maintainer for debugging, and also to record information in the console log.</p>

18.6.4 Conditional Processing

LISTSERV mail templates can be programmed with an if/then/else logic that evaluates available data and performs the appropriate task depending on the outcome of the evaluation.

A conditional block begins with the command `.BB` (begin block) and ends with `.EB` (end block). In the simplest case, the boolean expression following `.BB` is evaluated and, if false, all the text between the `.BB` and `.EB` delimiters is skipped. For instance, from \$SIGNUP:

```
.bb &DEFOPT ^= ''
```

Following instructions from the list owner, your subscription options have been set to "&DEFOPT" rather than the usual LISTSERV defaults. For more information about subscription options, send a "QUERY &LISTNAME" command to &MYNAMES.

```
.eb
```

.ELSE may be specified in cases where an alternate text is required, as in this more complicated example (also from \$SIGNUP):

```
.bb ((&x ^= POSTMASTER) and (&x ^= OWNER)) and (&x ^= OWNERS)
Please note that it is presently possible for
.bb &x = PUBLIC
anybody
.else
other people
.eb
to determine that you are signed up to the list through the use of the
"REVIEW" command, which returns the e-mail address and name of all the
subscribers. If you do not want your name to be visible, just issue a
"SET &LISTNAME CONCEAL" command.
```

```
.eb
```

Additionally, it is possible to simply exit a conditional at an arbitrary point by using a .QU (QUit) command. For instance, the MSG_POSTING_REJECT_BAD_ATTACHMENT message template is used both to reject unwanted attachment types as well as viruses. In order to avoid processing the entire template (which contains other text that is not germane to a virus rejection), .QU is used to simply exit processing and send the message immediately.

```
.bb &VIRUS = 1
.* A virus was detected in the message. Note that this is only possible
.* for messages sent to a list, LISTSERV does not execute attachments so
.* messages sent to LISTSERV are not checked for viruses.
Your posting to the &LISTNAME list has been rejected because it contains
.bb &VIRUS_NAME ^= ''
the '&VIRUS_NAME;'
.else
a
.eb
.bb &VIRUS_FILENAME ^= ''
virus in attachment '&VIRUS_FILENAME;'.
.else
virus.
.eb
You are strongly advised to check your computer for viruses as soon
as possible!
.qu
.eb
.* text for rejecting attachments follows
```

Notes: Conditional blocks may nest to an arbitrary depth.



The expression evaluator is recursive but not very sophisticated; the restriction you are most likely to encounter is that all sub-expressions have to be enclosed in parentheses if you are using boolean operators. That is, ".BB &X = 3" is valid but ".BB &X = 3 and &Y = 4" is not.

String literals do not require quoting unless they contain blanks, but quotes are

accepted if supplied.

Comparison operators are = <> ^= IN and NOT IN (the last two look for a word in a blank-separated list of options, such as a keyword value). These operators are not case-sensitive; == and ^= are available when case must be respected. Boolean operators are AND and OR.

A command line in a conditional block must be contained on one physical line and may not wrap, so be careful when sending MAILTPL files back to LISTSERV that you do not accidentally wrap long .BB lines.

The operators =* and ^=* are available to perform wildcard matches in conditional blocks. For instance JOHN_DOE@UNIX.EXAMPLE.COM =* J*DOE@*EXAMPLE.COM is a true statement. The wildcard specification is on the right-hand side whereas the actual text (or variable) you are evaluating is on the left.

18.6.5 The .QUIF Command

.QUIF (QUit IF) allows the invoker to stop processing the template if a certain condition is met, but without having to define a full-blown conditional block. For instance, the .IM command used to imbed one template into another returns a success code in the template variable &RC. If imbedding succeeds, &RC is set to 0, and something like the following is possible:

```
>>> MSG_POSTING_REJECT_BAD_ATTACHMENT Received an attachment not allowed
by "Attachments="
.* Use legacy BAD_ATTACHMENT template if present
.im *BAD_ATTACHMENT
.quif &rc = 0
.* at this point template processing stops and the message is sent if
&RC = 0
.* otherwise processing would continue with any following text.
"quif &rc = 0" is strictly equivalent to the block
.bb &rc = 0
.qu
.eb
```

18.6.6 Using 8-Bit Characters in Templates

If you include 8-bit characters (e.g., accented or national language characters) in templates, LISTSERV will automatically encode the templates on-the-fly using MIME quoted-printable encoding. While there is no guarantee that every mail program will be able to properly display 8-bit characters, those mail programs that do understand MIME quoted-printable encoding should have no trouble doing so.

18.7 Editing List-Level Default Templates

Please note that list-level mail templates are not available in LISTSERV Lite.

It is strongly recommended that the web interface customization tools be used to customize the "look and feel" of list templates. However, some administrators may wish to use the old method. If so, then simply make a copy of DEFAULT.MAILTPL on your local machine and name it listname.MAILTPL. Keep the original DEFAULT.MAILTPL around in case you make a mistake and need to start over.

At this point, you could theoretically store the listname.MAILTPL back on the LISTSERV host. However, without making any changes that would be somewhat pointless. At the very least you should edit the INFO template form before storing the template. Note also that you need only store the sections of the template that you have changed. For instance, if you edit the INFO template form but leave the rest of the template untouched, you can delete the rest of the template and store the INFO template form alone as listname.MAILTPL. The benefit to this approach is that any administrative changes to the rest of the default template are automatically applicable to your list as soon as they are made, rather than requiring that you edit your mail template individually to reflect such changes. If using the manual-editing procedure, L-Soft recommends that this approach be followed as the default.

18.7.1 The INFO Template Form

The INFO template form is LISTSERV's response to the command INFO listname. By default, it contains the following:

Figure 18-1 Default Content of the INFO Template form for DEFAULT.MAILTPL

```
>>> INFO Information about the &LISTNAME list
There is no information file for the &LISTNAME list. Here is a copy of the
list "header", which usually contains a short description of the purpose of
the list, although its main purpose is to define various list configuration
options, also called "keywords". If you have any question about the
&LISTNAME list, write to the list owners at the generic address:

.ce &LISTNAME-Request@&MYHOST

.dd &LISTHDR
```



Note: the replaceable parameters &LISTNAME and &MYHOST. Don't change &MYHOST; LISTSERV replaces it with the correct value for the name of the host site. &LISTNAME automatically inserts the name of the list. It's probably best to use &LISTNAME to refer to the list throughout the document rather than to replace it with something like "MYLIST-L". This ensures that the template form will be consistent with the default and will be simpler to debug should a problem arise. Also, in the event the name of the list changes, it will be unnecessary to edit the template form (although it would have to be renamed to match the new name of the list, of course).

Should it be desirable to replace the default INFO template form with information about the list, it is probably best to remove the .dd &LISTHDR line. This line instructs LISTSERV to read in the header of the list and add it to the response in lieu of any other data about the list. Many list owners add descriptive comment lines to their list headers, thus this default.

Here is a minimally-edited sample INFO template form for a list called MONKEYS.¹

Figure 18-2 Sample of an Edited INFO Template Form

```
>>> INFO Information about the &LISTNAME list
&LISTNAME is an open, unmoderated discussion list featuring monkeys. Things
such as how to care for a pet monkey, monkey diseases, monkey lore, endan-
gered species of monkeys, and monkey psychology are likely to be discussed.
The list is NOT intended for discussion of Darwinism and/or theories of
evolution.

If you have any question about the &LISTNAME list, write to
the list owners at the generic address:

.ce &LISTNAME-Request@&MYHOST
```

18.7.2 DEFAULT MAILTPL Templates



Note: We no longer provide a complete list of template forms in this section as there are simply too many of them. Templates are listed by name and description in the customization section of the web interface.

The following are template forms that can be defined, but which are not present by default in DEFAULT.MAILTPL. If you want to define them for a particular list, the easiest way to do so is via the web interface.

- **POSTACK1** (optional) – When present, this message is sent in reply to any message posted to the list. This is very useful for creating "infobots", or just for returning a standard acknowledgement to contributors. The **&SUBJECT** variable contains the subject of the original message, and naturally the usual substitutions (**&LISTNAME**, **&DATE**, **&TIME**) are available.
- **TOP_BANNER**, **BOTTOM_BANNER** (optional) – When these template forms are present, their contents are automatically inserted at the top (respectively bottom) of each and every message posted to the list. Typically, the top banner would be used for a copyright or short legal warning which absolutely has to be seen by each and every reader. The bottom banner could contain instructions for signing off the list, a disclaimer, an acknowledgement of a sponsor's contribution, a "tip of the week", etc.



Documented Restriction: The use in banners of substitutions which do not yield a constant result (e.g., **&TIME**) will defeat the duplicate mail detection part of **LISTSERV**'s loop-checking heuristics in any case where a subscriber is forwarding all mail back to the list. L-Soft advises that such substitutions never be used in a **TOP_BANNER** or **BOTTOM_BANNER**.

LISTSERV does not attempt to remove bottom banners from individual messages in digests, for instance, which have been included as quoted material in responses.

- **TOP_BANNER_HTML**, **BOTTOM_BANNER_HTML** (optional) – When these template forms are present, they will be used "as is" for HTML message parts. If absent, the regular banner is used for HTML, probably with less than 100% satisfaction.

1. Thanks to Marty Hoag of NEW-LIST.



Documented Restriction: The use in banners of substitutions which do not yield a constant result (e.g., &TIME) will defeat the duplicate mail detection part of LISTSERV's loop-checking heuristics in any case where a subscriber is forwarding all mail back to the list. L-Soft advises that such substitutions never be used in a TOP_BANNER_HTML or BOTTOM_BANNER_HTML.

- `CONTENT_FILTER` (optional) – When present, provides LISTSERV with a ruleset for message content filtering that can be configured at the list level. See Section 12.16 [Content Filtering](#) for more information on how to use content filtering.

18.7.3 Tips for Using Templates

- Many list owners require prospective subscribers to fill in a little questionnaire before being added to the list, or to explicitly state that they have read the list charter and agree to follow all rules or be removed from the list. The most convenient method, for both list owner and subscriber, is to have the SUBSCRIBE command return a copy of the questionnaire (or list charter, etc.), and not forward the request to the owner. The user answers the questions and returns them directly to the list owner, who then adds the subscriber manually. Naturally, it is more convenient for the user if this information arrives in a separate message, with a 'Reply-To:' field pointing to the list owner's address. Thus, you should not use the SUB_OWNER template form for this purpose, because it is a linear template form and does not give you any control over the 'Reply-To:' field. The SUB_OWNER template form could be modified to read "A copy of the list charter is being sent to you, please read it carefully and follow the instructions to confirm your acceptance of our terms and conditions." The list charter would then be sent separately, through the ADDREQ1 template form. You would use the .RE OWNERS command to instruct LISTSERV to point the 'Reply-To:' field to the list owners, and .TO &WHOM to change the destination from list owner to subscriber. If you want to receive a copy of the message, you can use .TO &WHOM cc: xxx@yyy.
- When writing template forms, it is a good idea to use substitutions (&XXXX) for information which may change in the future. In particular, it is not uncommon for lists to have to be moved from one host to another, and this will be a lot easier if the template forms use substitutions for the list address and list host. The &LISTADDR substitution translates the full address of the list (XYZ-L@XYZ.COM), whereas &LISTNAME is just the name (XYZ-L). For references to the server and host, use &MYHOST for the Internet hostname, &MYSELF for the server address (normally LISTSERV@&MYHOST), and &OWNER for the xxx-request mailbox address. These substitutions are "universal" and can be used in all template forms. For instance, if you decide to make a bottom banner with instructions for leaving the list, the text could read: "To leave the list, send a SINGOFF &LISTNAME command to &MYSELF or, if you experience difficulties, write to &OWNER."

18.8 Using the DAYSEQ(n) Function

The DAYSEQ(n) function is quite powerful. This function allows the list owner to code template forms (such as the PROBE1 or BOTTOM_BANNER messages) that change or "rotate" automatically.

The DAYSEQ(n) function is invoked in a .BB - .EB conditional block, and n corresponds to the number of days in the rotation period, i.e., to the number of variations that you want to make to the text of the message. &DAYSEQ(n) returns a number from 1 to n which increases by 1 every day, with no special regard for weekends. That is, if the rotation period is to last for a week, you code DAYSEQ(7). If the rotation period is 15 days, you code DAYSEQ(15). Two examples follow:

18.8.1 Rotating Bottom Banner

To create a rotating bottom banner, follow this example. A list has three commercial sponsors, each of whom are provided with an advertisement every three days. (Note that this doesn't take weekends into account; in this example, if company A is featured in the banner on Monday, it will be featured again on Thursday and then again on Sunday. However, in the following week it will be featured on Wednesday, Saturday, and Tuesday, so it will actually get rather good coverage.) Our BOTTOM_BANNER template form would look like this:

```
>>> BOTTOM_BANNER
.BB &DAYSEQ(3) = 1
Today's copy of the &LISTNAME newsletter has been brought to you by Company A.
.EB
.BB &DAYSEQ(3) = 2
Today's copy of the &LISTNAME newsletter has been brought to you by Company B.
.EB
.BB &DAYSEQ(3) = 3
Today's copy of the &LISTNAME newsletter has been brought to you by Company C.
.EB
```

If a company needs to get a higher percentage of "air" time than another, you can simply assign it more than one of the possible n values of &DAYSEQ(n). For instance, if you have two companies but one should get twice as many days of "air" time, you might code something like this:

```
>>> BOTTOM_BANNER
.BB (&DAYSEQ(3) = 1) OR (&DAYSEQ(3) = 3)
Today's copy of the &LISTNAME newsletter has been brought to you by Company A.
.EB
.BB &DAYSEQ(3) = 2
Today's copy of the &LISTNAME newsletter has been brought to you by Company B.
.EB
```

This would cause Company A's message to appear on days 1 and 3 of the rotation period and Company B's message to appear on day 2 only.

18.8.2 Rotating FAQ via the PROBE1 Template and "Renewal= xx-Daily"

Subscription renewal can be coded with daily granularity (however, please note that it is and remains inadvisable to use renewal intervals of less than a week). If you further code subscription probing into the "Renewal=" keyword with the ",Probe" parameter, you open

up the possibility of turning the standard PROBE1 template form into a periodic FAQ. Here's how:

We'll assume to start that you will code "Renewal= 15-Daily,Probe" in your list header. (You can experiment with other numbers, but since we have two messages and will be using &DAYSEQ(2), we need an odd renewal period.) We'll also assume that you want to send two versions of your FAQ each month; the first, a complete FAQ document, and the second, an abbreviated "reminder" version that just contains information about how to sign off, how to post to the list, and so forth. The basic algorithm is therefore:

When &DAYSEQ(2) = 1, send the full FAQ.

When &DAYSEQ(2) = 2, as it will 15 days later, send the abbreviated FAQ.

Your PROBE1 template form would thus look like this:

```
>>> PROBE1 Periodic FAQ posting for &LISTNAME
&WEEKDAY, &DATE &TIME
.BB &DAYSEQ(2) = 1
```

This is the complete FAQ for &LISTNAME. Please read it and keep a copy for future reference. A FAQ document for &LISTNAME is distributed every 15 days, the full FAQ alternating with a shorter "reminder" FAQ.

```
<body of the full FAQ document>
```

```
.EB
```

```
.BB &DAYSEQ(2) = 2
```

This is the abbreviated FAQ for &LISTNAME. Please read it and keep a copy for future reference. A FAQ document for &LISTNAME is distributed every 15 days, the full FAQ alternating with a shorter "reminder" FAQ.

```
<body of the abbreviated FAQ document>
```

```
.EB
```

18.8.3 Calculating the Value for DAYSEQ()

When you first start using a rotating banner with the &DAYSEQ variable, the &DAYSEQ(n)= 1 period begins based on the number of days elapsed since a baseline. On VM (and in REXX generally) you can calculate today's value easily with:

```
/* */
say Date('B') + 1
```

If you do not have access to a REXX interpreter, Date('B') is described as "the number of complete days (that is, not including the current day) since and including the base date, 1 Jan 0001, in the format 'dddddd' (no leading zeros or blanks)."¹ It also is equal to the C language expression `time(0)/86400 + 719162` or, for OpenVMS users, to the Smithsonian base date plus 678575.

For example, for Friday 22 Oct 2004, the value of `Date('B') + 1` is 731876. This value increases by one every day at midnight.

1. Cowlshaw, Michael: The REXX Language: A Practical Approach to Programming, 2nd ed., p.92. Englewood Cliffs, NJ: Prentiss-Hall, Inc., 1990.

18.9 Storing the <listname>.MAILTPL File on the Host Machine

The procedure differs slightly on VM systems, but the following will work for unix, VMS and Windows systems:

1. Get a copy of DEFAULT.MAILTPL and edit it.
2. Be sure that you have defined a "personal password" to LISTSERV with the PW ADD command before you PUT the template file. If you have done this but can't remember the password, send a PW RESET command to LISTSERV, then a new PW ADD command.
3. Send the file to LISTSERV with a PUT *listname* MAILTPL PW=XXXXXXXXX command at the top of the file, just as if you were storing the list itself. Replace XXXXXXXXXX with your personal password.

The variation for VM systems is that the LISTSERV maintainer will have to create a fileid for the file before it can be PUT on the server and seen by LISTSERV.



Note: The LISTSERV maintainer can create and edit these files in place with any standard text editor. Changes made to template files in this way are available to LISTSERV immediately after they are saved.

18.10 DIGEST-H and INDEX-H Template Files

Two other template files that are available pertain to the automatic "digestification" feature. You may create and store files called listname DIGEST-H and listname INDEX-H. These files define custom digest headers and custom index headers, respectively. The DIGEST-H and INDEX-H files are plain text files, like the WELCOME and FAREWELL files, and the instructions for storing them on the server are identical. Note that, as with the WELCOME and FAREWELL files, you cannot use the template formatting commands and replaceable parameters discussed above.

Figure 18-3 Typical Contents of a DIGEST-H or INDEX-H File

```
The MYLIST list is sponsored by ABig Corporation.

See http://www.abig.com for information on ABig Corporation's products.
```

The contents of DIGEST-H and INDEX-H are appended to the digest or index, respectively, immediately following the list of topics.



Notes: The INDEX-H output would be similar to the figure below, following the list of postings.

You can't add a digest or index "footer" because according to the standard, anything after the end of the digest text is supposed to be discarded.

Figure 18-4 Sample DIGEST Output for a List with a DIGEST-H File

```

Date: Tue, 11 Jun 2001 11:52:41 -0500
From: Automatic digest processor <LISTSERV@MYHOST.COM>
Reply-To: My test list <MYLIST@MYHOST.COM>
To: Recipients of MYLIST digests <MYLIST@MYHOST.COM>
Subject: MYLIST Digest - 10 Jun 2001 to 11 Jun 2001

There is one message totalling 10 lines in this issue.

Topics in this issue:

    1. Testing 125...3 sir!

The MYLIST list is sponsored by ABig Corporation.

See http://www.abig.com for information on ABig Corporation's products.

```

18.11 WWW Interface Templates and Template Forms

The following describes the available template files and their respective template forms for the WWW archive and administration interface. L-Soft does not advise modifying these templates unless you know exactly what you are doing. If you modify the templates it is strongly recommended that you keep copies of the originals in a safe location for fall-back. In practice, this means that you should NEVER edit the default template files themselves.

18.11.1 Web Forms (Static) Contained in DEFAULT MAILTPL



Note: We no longer provide a complete list of template forms in this section as there are simply too many of them. Templates are listed by name and description in the customization section of the web interface.

18.11.2 The WWW_ARCHIVE.MAILTPL File

In LISTSERV 14.3, this file became obsolete. LISTSERV will migrate existing templates from WWW_ARCHIVE MAILTPL to SITE MAILTPL, unless one of the following conditions arises:

- A conflicting template (same name, different contents) is already in SITE MAILTPL.
- The copy of the template in WWW_ARCHIVE matches the first template in the system search order.

If there are no conflicts, WWW_ARCHIVE MAILTPL is then renamed to WWW_ARCHIVE OLDTPL. If a conflict is detected, LISTSERV will not attempt to determine which version of the template is "correct", but rather will log something like the following:

```

7 Oct 2004 10:57:05 Migrating templates from WWW_ARCHIVE to SITE...
- $TEST_TEMPLATE: conflicting version found in SITE
7 Oct 2004 10:57:05 Conflicts detected, finish migration manually

```

It is then incumbent on the site maintainer to harmonize the differing versions of any templates reported to be in conflict.

18.11.3 The DEFAULT.WWWPTL File (Dynamic Templates)

The DEFAULT WWWPTL file contains the default templates for the parts of the WWW archive interface that are not defined in DEFAULT MAILTPL. Unless you have specific issues that need to be resolved (such as a national language preference or a need to point certain links to non-standard locations), it is strongly recommended NOT to edit this file. One reason for this is that DEFAULT WWWPTL will be overwritten by a software update. The safest approach to customizing the look and feel of your site is to use the customization features built into the web interface, which save your customizations in a different place.

When customizing these templates, there are two fundamental differences between them and the templates in DEFAULT MAILTPL:

1. Any substitution variable that you use (for instance, &LISTNAME) must be escaped with a "+" symbol between the ampersand and the name of the variable, thus: &+LISTNAME. (Note that, as with the regular mail template forms, not all substitution variables are available in every HTML template form.)
2. Any dot-formatting command you use (for instance, .CC , .BB , etc.) must have a "+" symbol rather than the dot, thus: +CC +BB



Note: We no longer provide a complete list of template forms in this section as there are simply too many of them. Templates are listed by name and description in the customization section of the web interface.

18.11.4 The SITE.WWWTPL File

The preferred method of editing site.wwwtpl is to use the web interface's built-in customization features to edit templates. L-Soft no longer recommends editing site.wwwtpl by hand.

However, if desired, you can override the default.wwwtpl file by providing a customized site.wwwtpl file in LISTSERV's A directory.

The site.wwwtpl file is the file in which edited web templates are placed. This prevents your site-wide definitions being overwritten in an upgrade (i.e., when default.wwwtpl will normally be overwritten). If a given template is found in site.wwwtpl, that version of the template takes precedence over the one found in default.wwwtpl. This means that you don't have to duplicate every template form in default.wwwtpl, just the ones you don't want overwritten by an upgrade.



Note: For list-level templates, site.wwwtpl will itself be overridden by definitions in any listname.wwwtpl files you have installed.

18.11.5 National Language Template Files (idiom.mailtpl)

National language templates can be written and used with LISTSERV (L-Soft does not provide them). The use of such templates is optional and governed by two settings:

- **Site-wide** – The DEFAULT_LANGUAGE= site configuration variable allows you to set the site-wide national language template for use by all lists on the server. By default this variable is unset and DEFAULT MAILTPL is used.
- **List-level** – The Language= list header keyword can be used to specify a national language template to be used for a particular list, for instance a Spanish-language list on an otherwise English-language server.

To create a national language template, you simply copy DEFAULT MAILTPL to *idiom* MAILTPL, where *idiom* is the name of the language, and translate it as desired. You also use *idiom* to specify the value for DEFAULT_LANGUAGE= and/or Language=. For a given language you can specify anything you want for *idiom*; in other words LISTSERV does not care if you call the file FRANCAIS MAILTPL or FRENCH MAILTPL, but if you do call it FRENCH MAILTPL you must specify FRENCH as the *idiom*, and likewise, if you call it FRANCAIS MAILTPL you must specify FRANCAIS as the *idiom*. LISTSERV has no information about what a given language is called, it simply looks for a MAILTPL file for the *idiom* data supplied.

If you are going to translate the web interface template forms into *idiom* as well, you will need to copy DEFAULT WWWTPL to *idiom* MAILTPL and again, translate as desired. This requires that you add a special template form to WWW_ARCHIVE MAILTPL, so that the 'wa' CGI script will know what to look for, as follows:

```
>>> LANGUAGE
idiom
```

where *idiom* is, of course, the same value we've been talking about above. For instance for a FRANCAIS *idiom* you'd use

```
>>> LANGUAGE
FRANCAIS
```

See Section 18.2.1 [Mail Templates](#) regarding the use of 8-bit characters in template forms.

18.11.6 Template Precedence

For template forms found in DEFAULT MAILTPL, the following precedence is used when LISTSERV searches for a given template form:

```
listname MAILTPL
idiom MAILTPL
WWW_ARCHIVE MAILTPL1
DEFAULT MAILTPL
```

That is to say, if LISTSERV needs a copy of the ADD1 mail template form, it will look first in the *listname*.mailtpl file for the list in question. If no such file exists, or if ADD1 is not present in *listname*.mailtpl, LISTSERV will look in *idiom*.MAILTPL (if Language= or DEFAULT_LANGUAGE= is set to *idiom*). Again, if the ADD1 form is not present in *idiom*.mailtpl, or if *idiom*.mailtpl does not exist, LISTSERV will then look in default.mailtpl (*www_archive*.mailtpl is skipped because ADD1 is not a web template form) and pull out the default ADD1 template form.

For template forms found in DEFAULT WWWTPL the precedence is:

```
listname WWWTPL
idiom WWWTPL
SITE WWWTPL
DEFAULT WWWTPL
```

1. WWW_ARCHIVE MAILTPL is searched only for web-related template forms and is bypassed for mail template forms. For instance WWW_ARCHIVE MAILTPL will not be searched for ADD1 but will be searched for WWW_INDEX.

The same sequence of events applies as for the MAILTPL files, except that SITE WWWTPL is never skipped (all template forms in the WWWTPL files are web forms).

18.12 Serving Up Custom Web Pages for your List

This feature is not available in LISTSERV Lite.

Originally in order to serve up custom or special web pages for a list it was necessary to construct those pages as HTML files and place them either into the /archives directory or link them from somewhere else. This was sometimes impossible for list owners who had no administrative access to the server's web directories or who had no other place from which to serve web pages.

It is possible to add ad-hoc web page templates by creating new (non-standard) template forms and enabling their display by setting a special variable value, SHOWTPL_ALLOWED, in the template form. For instance, one could set up a page with special administrative information, the list charter, netiquette information, or the like, and serve it and maintain it directly from the LISTSERV web template interface without need for any other access to the server.

18.12.1 A Practical Example: ADMIN_POST

The author used to serve up an administrative posting via FTP back in the days when his lists lived on a server that had FTP access to the archive notebooks. When FTP access to the server was cut off due to security concerns, he had to find another way to serve the information via the web. Here is how it was done:

(The following example assumes that you have the 1.8e web administration interface installed. New template forms cannot be created this way in previous versions.)

First, log into the web administration interface. Choose the list for which you will be making a new page, and then click the **[Templates]** button to enter the mail and web template editing area. Since the template you will be creating is a web template, click the **[Switch to WWW templates]** button to change modes.

Next, type the name of the new template form into the box provided, and click **[Create]**. The Edit List Template screen opens with the command response

```
The ADMIN_POST form has been successfully stored in the TEST
template library.
```

In the **Description** box, type a description of the template, for example, "Administrative information page".

In the large text box provided for the template text, first type the following line:

```
+SE SHOWTPL_ALLOWED 1
```

This line tells LISTSERV that it is allowed to serve the page on the web. If the line is not found, the template will not be available.

Following this line you can start adding your HTML. However note carefully that you cannot override the default headers and footers that have already been defined by other template forms in the library. You can start with a <title></title> block but it will be followed by the pre-defined header and then by your HTML.

After adding your HTML, click **[Update]**, and the template form will be stored. The URL for the page you are defining in this example will be `http://your_server_hostname/path_to_wa?SHOWTPL=ADMIN_POST&L=listname` (the parameters for 'wa' are case-

sensitive and must be sent in upper case). For instance, the author's version of the ADMIN_POST template form can be viewed at http://peach.ease.lsoft.com/scripts/wa.exe?SHOWTPL=ADMIN_POST&L=VISBAS-L.

18.13 Using a Conditional Expression Evaluator



The web interface CGI now supports LISTSERV's conditional expression evaluator for templates, made available as +IF. For instance, it is now possible to code the following in a template:

For wildcard string matches:

```
+IF HELLO =* HE*  
*do something here*  
+ENDIF
```

For comparing numbers, etc.:

```
+IF (5 > 6)  
*do something here*  
+ENDIF  
(+ENDIF is an alias for +EB; you can use either.)
```


Section 19 Solving Problems

19.1 Helping Subscribers Figure Out the Answers

Depending on your own preferences, some requests from subscribers for operations that they can perform for themselves can be fulfilled by you as the list owner or by the subscribers with some coaching from you. While it is a negative approach, the list owner can never assume that the subscriber reads or saves the materials sent to him at the time of subscription. Therefore, you will have to deal on a regular basis with users who ask how to unsubscribe, or how to get archive files, or how to set their subscription to DIGEST or NOMAIL.

Often these requests for help are posted directly to the list. The proactive approach to this problem is to do one or both of two things:

- Respond to the list with the answer so that all can benefit
- Respond privately to the subscriber with the answer if it has been posted repeatedly

If a user asks a question about a topic that has been discussed previously, you might suggest in a tactful way that the answer can be found in the archives. If your host server supports the LISTSERV database functions, you might even include a sample DATABASE JOB that the user can "clip and send" to LISTSERV.

Often it is tempting to simply "get things over with" and take care of the user's request in many cases – the user wants to be set to NOMAIL because he's going on vacation, the user wants off the list, etc. – but while this solves the short-term problem, it doesn't teach the user anything. Naturally it takes more time to be a coach than it does to be the all-powerful list administrator, but the goodwill you can create by being proactive rather than reactive outweighs the convenience of simply sending the command yourself. You will find that many subscribers appreciate the fact that someone takes the time to explain the complexities of LISTSERV to them.

In order to cut down on the time it takes to respond in "coaching" situations, many list owners prepare "boilerplate" files with the answers to common questions that they can simply "cut and paste" into return mail. (Several such "boilerplate" files are included in Appendix C.)

19.2 Loop-Checking Can Cause Occasional Problems with Quoted Replies

(See Section 13.7.5 "[Sender:](#)", "[From:](#)", or "[Reply-To:](#)" Fields in Body Causes Bounce for additional information.)

By default, LISTSERV's internal loop-checking routines look for anything in the body of a mail message that looks like a header line – specifically anything that looks like a "To:", "Sender:", or "Reply-To:" header line. If it finds anything like this, LISTSERV intercepts the message and sends it to the list owner (or the person(s) designated by the "Errors-To=" keyword) as an error.

Often a user who replies to list mail includes all or part of the message he is replying to as part of his reply ("quoting"). While this is a questionable practice to begin with, unfortunately a number of popular mail programs make it worse by including the quoted

message in its entirety (including header lines) in the body of the reply. For instance, the following message ended up in the author's error mailbox:

Figure 19-1 Sample of an Error Message with Headers Included

```
The enclosed message, found in the ACCESS-L mailbox and shown under the spool
ID 6305 in the system log, has been identified as a possible delivery error
notice for the following reason: "Sender:", "From:" or "Reply-To:" field
pointing to the list has been found in mail body.
----- Message in error (42 lines) -----
-----
Received: by access.mbnet.mb.ca id AA05697
(5.67b/IDA-1.4.4 for Microsoft Access Database Discussion List
<ACCESS-L@peach.ease.lsoft.com>); Wed, 1 Mar 1995 10:26:29 -0600
Date: Wed, 1 Mar 1995 10:26:29 -0600
From: xxxxxx xxxxxxxxxx <xxx@MBNET.MB.CA>
Message-Id: <199503011626.AA05697@access.mbnet.mb.ca>
To: Microsoft Access Database Discussion List
Message-Id: <199503011626.AA05697@access.mbnet.mb.ca>
To: Microsoft Access Database Discussion List
<ACCESS-L@PEACH.EASE.LSOFT.COM>
Subject: Re: Re: Foxpro listserv address
X-Mailer: AIR Mail 3.X (SPRY, Inc.)

<---- Begin Included Message <---->
Date: Thu, 23 Feb 1995 01:17:36 -0500
From: xxxxxxxx@xxx.com
Sender: Microsoft Access Database Discussion List
<ACCESS-L@peach.ease.lsoft.com>
Subject: Re: Foxpro listserv address
To: Microsoft Access Database Discussion List
<ACCESS-L@peach.ease.lsoft.com>

>BTW, I don't know why she is still on Foxpro, I thought they went out into
>the desert??

<---- End Included Message <---->
(subscriber's reply deleted)
```

The problem with this reply was two-fold, from a list owner's standpoint. First (a netiquette issue), the sender didn't bother to remove unnecessary header lines from his reply. If properly formatted, however, this would not normally cause an error.

Second, the mail software he was using didn't include ">" characters at the beginning of every line of the included message. Had it done so, the message would have passed through LISTSERV unhindered.

One variation on this error is mail software that quotes messages by adding the ">" character followed by a space for esthetic reasons. For instance, using the above error as an example:

Figure 19-2 Another Sample of an Error Message with Headers Included

```
> Date: Thu, 23 Feb 1995 01:17:36 -0500
> From: xxxxxxxx@xxx.com
> Sender: Microsoft Access Database Discussion List
> <ACCESS-L@peach.ease.lsoft.com>
> Subject: Re: Foxpro listserv address
> To: Microsoft Access Database Discussion List
> <ACCESS-L@peach.ease.lsoft.com>
> BTW, I don't know why she is still on Foxpro, I thought they went out
> into the desert??
```

This won't work either. Generally this is a client configuration problem and it can be fixed by setting the quoting character in the client's configuration file.

On the other hand, the following quote would have worked:

Figure 19-3 A Correctly Formatted Message with Headers Included

```
>Date: Thu, 23 Feb 1995 01:17:36 -0500
>From: xxxxxxxx@xxx.com
>Sender: Microsoft Access Database Discussion List
> <ACCESS-L@peach.ease.lsoft.com>
>Subject: Re: Foxpro listserv address
>To: Microsoft Access Database Discussion List
> <ACCESS-L@peach.ease.lsoft.com>
>BTW, I don't know why she is still on Foxpro, I thought they went out
>into the desert??
```

The ultimate solution to the problem is to warn subscribers to limit their quoting to a minimum, and in any case to be sure to delete anything that looks like a header line in the body of their reply.

19.3 User Can't Unsubscribe and/or Change Personal Options

See Section 13.2 [Finding Users Who Do Not Appear in the List](#) for details.

19.4 Firewalls

Firewalls on the Internet are set up for essentially the same reason firewalls are designed into buildings and automobiles – to keep dangerous things (in this case, hackers, viruses, and similar undesirable intruders) from getting in and wreaking havoc with sensitive data. Unfortunately, they don't always keep people from behind them from sending mail out, and this can cause problems when users from such sites attempt to subscribe to lists.

If your list is set to confirm all subscriptions with the "magic cookie" method ("Subscription= Open,Confirm"), you will receive an error message any time a user from a firewalled site attempts to subscribe, since the "cookie" confirmation message will bounce off the firewall. If your list is not set to confirm subscriptions, the same user will be able to subscribe to your list but all mail sent to him will bounce.

Some firewalls reportedly can recognize "friendly" LISTSERV mail and let it through, but because of security considerations, it is unlikely that this problem will ever completely go away. Thankfully it does not seem to be a major cause of mailing list errors.

19.5 LISTSERV Won't Store Your List

LISTSERV expects list files to be delivered to it without any formatting characters (excluding, of course, the carriage return-line feed at the end of each line). This can cause a problem if you try to store the entire list (header and subscribers) using a mail client that inserts line-wrap characters into text longer than 80 columns. Specifically, one client that does this is Pine; others that can cause problem are cc:Mail and just about any Windows POP client.

There are a couple of ways to get around this problem.

1. Don't get the entire list if all you're going to do is edit the header. Use the `GET listname` (HEADER syntax to get the header only, and use the `ADD` and `DELETE` commands to manipulate the subscriber list. This is the preferred method.
2. If you have to get the entire list, e.g., in order to delete a subscriber manually, use a client that does not wrap text (or turn off line wrap if possible). If you are on a unix system that has mailx installed, you can store a list from the command line with the command syntax

```
mailx listserv@host < listfile
```



Note: L-Soft does not recommend hand-editing the subscriber list; it is preferable to use wildcards to delete problem addresses, and using an editor to do this should always be the last resort.

3. If all else fails, you can use a public-domain utility called LB64 to convert the list file into a base-64 command JOB that LISTSERV will understand. This utility is generally available from the VM LISTSERV sites; send a `GET LB64 C` command to `LISTSERV@LISTSERV.NET` if you can't find it anywhere else.



Note: This is an unsupported utility. You will need to compile it with a C compiler (not supplied). The utility is primarily for users on unix systems, although with two minor modifications it can also be used on 32-bit Windows systems.

19.6 If I can't find the answer, where do I turn?

Two LISTSERV lists exist for list owner and LISTSERV maintainer questions.

- LSTSRV-L is the LISTSERV give-and-take forum. Its primary mission is to provide assistance to LISTSERV maintainers, but it can also be of interest to list owners who desire a more in-depth knowledge of the workings of the system. To subscribe to LSTSRV-L, send your subscription request to `LISTSERV@LISTSERV.NET`.
- LSTOWN-L is the LISTSERV list owners' discussion list, where list owners can get assistance on list maintenance and other aspects of list ownership. To subscribe to LSTOWN-L, send your subscription request to `LISTSERV@LISTSERV.NET`.

Section 20 Advertising Your Public Mailing Lists

20.1 List of Lists Maintained by LISTSERV

LISTSERV automatically produces a List of Lists that may be reviewed by users anywhere on the Internet in one of two ways:

- L-Soft's CataList service at <http://www.lsoft.com/CataList.html>.
- The LISTS GLOBAL searchtext command. This list of lists is made up of one-line entries containing the short listname and the descriptive title of the list (up to about 60 characters in length). A sample of the List of Lists format can be seen in Section 12 [Starting a Mailing List Using Email](#).



Note: It is possible to code a descriptive title in your list header that is more than 40 columns long, but the List of Lists will include only the first 40 columns of that title. Therefore, it is important from this respect to be sure that the descriptive title of your list is succinct and to the point.

20.2 Adding HTML to a List Header for the CataList

L-Soft's CataList service (<http://www.lsoft.com/CataList.html>) allows users to search the global list of public LISTSERV lists via the World Wide Web. Adding an HTML description to a list is easy, and can do a lot to enhance the appearance of a list in the database. All you have to do is update your list header and add the text of your choice. Here is an example:

```
* The coffee lovers' list
*
* Review= Public      Subscription= Open      Send= Public
* Notify= Yes        Reply-to= List,Respect
* Notebook= Yes,L,Monthly,Public
*
* Owner= claudia@espresso.xyz.it (Claudia Serafino)
*
* <HTML>
* COFFEE-LOVERS is an open list for, well, coffee lovers! Our
* motto is: <cite>"Instant - just say no!"</cite>
* That's pretty much our whole charter, although there are a
* few other <a href="http://www.coffee.org/charter.html">
* rules</a> that you may want to read before joining. For
* instance, we don't allow flame wars about decaf: if you like it,
* well, it's your body after all.
*
* <p>The list is maintained by
* <a href="http://www.coffee.org/claudia.html">Claudia
* Serafino</a> (that's me!) and you will find all sorts of
* useful info about coffee on my home page.
* </HTML>
*
```

In other words, you just insert your HTML text in the list header and bracket it with `<HTML>` and `</HTML>` tags (these tags tell the web interface where the HTML text begins and ends – they are not actually sent to the web browser). There are three simple rules that you must follow when inserting your HTML data:

1. The `<HTML>` and `</HTML>` tags must appear on a separate line, as shown in the example above. You cannot have anything else on that line and, in particular, you cannot mix keyword definitions with HTML data.
2. The HTML data you are providing is embedded into the document shown by the web interface when users query your list. Because you are given some space between two horizontal rules on an existing page, rather than a whole new page, you should not include tags that affect the whole document, like for instance `<TITLE>`.
3. While this procedure is compatible with all versions of LISTSERV, there are a few restrictions on the placement of equal signs within your HTML text with versions that do not have any specific support for the `<HTML>` and `</HTML>` markers. In practice, you can ignore this rule unless you get an error message while storing your list.

When reformatting your list header description for HTML, bear in mind that the text will not always be viewed using a web browser. It is best to keep the formatting as clear as possible and minimize the usage of HTML tags, since there are still many people without WWW access. For instance, do not hesitate to use white space between paragraphs for clarity.

20.2.1 Update latency

Barring network outages, a list header update takes a maximum of 24h to be reflected in the distributed LISTS database. Database updates are usually scheduled to be broadcast at night, so the changes take place overnight. Once the LISTS database has been updated, it can take a maximum of 24h for the frozen copy of the database used by the web interface to be updated. In most cases, both the LISTS database and its frozen copy on the web server will be updated overnight. However, if the site hosting your lists is several time zones west of the site hosting the web server, and if that server only updates itself once a day, you may have to wait two days for your update to be reflected.

20.2.2 Inserting a Pointer to Another List

Sometimes it may be useful to link a number of related lists together so that the viewer can quickly examine all the lists without having to go back to the search screen and retyping the names you are providing. You can do this using the special HTML sequence:

```
<!--#listref listname@hostname-->
```

This sequence is internally translated to an `<a>` tag with a URL that will bring up information about the list you indicated. You must then provide a suitable caption and a closing `` tag. Example:

```
Don't forget to take a look at  
<!--#listref COFFEE-L@COFFEE.ORG-->  
the coffee list!</a>
```

20.2.3 Restrictions on the Placement of Equal Signs

While all versions of LISTSERV are supported, servers which have no specific support for the `<HTML>` and `</HTML>` tags will process your HTML data as an ordinary list header line and attempt to determine whether it contains a list header keyword or descriptive text. The exact algorithms vary from one version to another, but in general the parser looks for a single word followed by an equal sign. With HTML text, it is possible (if unlikely) to generate such patterns. Here is an example:

```
*
* Sample list with problem pattern
*
* <HTML>
* For more information on the list, just check <a
* href="http://www.xyz.edu/mypage.html">my home page.</a>
* </HTML>
*
```

In that case, you can just reorder the HTML data so that the equal sign does not appear in this position. Alternatively, if the equal sign was meant to be actually displayed as an equal sign (as opposed to being part of some HTML tag), you can use the HTML escape sequence `=` instead.

20.3 Defining Search Categories in a List Header for the CataList



Note: The complete list of search categories may not yet be available when LISTSERV is released. Note also that during the "pilot" phase of categories implementation, all categories will be "open", and you can define search categories for your list as long as the categories you define are in compliance with the rules for defining categories. When the "production" phase begins, only categories defined below as "open" will be open, and if a list is created or modified without a "Categories=" keyword, LISTSERV will issue a warning (but will go ahead and store the list without it).

Another feature of the CataList service discussed in the preceding section is the ability to search for lists based on topic categories. For instance, a user might be looking for lists that discuss various aspects of opera. The same user might want to search not just for lists that discuss opera in general, but great operatic tenors in particular.

In order to implement search categories for your list, you use the new "Categories=" list header keyword, in conjunction with the list of categories that can be found at the CataList site. The URL for the category list is <http://www.lsoft.com/listcat.html>.

If you do not have a web browser, you can issue the command

```
GET LISTCAT FILE
```

to LISTSERV@LISTSERV.NET or any LISTSERV server running version 1.8c or higher to have a list of categories mailed to you.

A typical category listing is in two parts. The first part is the category title itself (this is what you code in the "Categories=" keyword). The second part is an optional description of what the category covers. For instance:

```
Category:SubCategory:MinorCategory Description of this category
```

There are two types of categories that you need to be aware of.

- **Open Categories** – These categories have a description indicating that they are open and can be added to. Taking our example of great operatic tenors above, you might see the following category listed:

```
Arts:Music:Opera:Singers Operatic Singers (Open)
```

You notice that there are further subcategories like

```
Arts:Music:Opera:Singers:Te_Kanawa_Kiri
```

```
Arts:Music:Opera:Singers:Caruso_Enrico
```

and so forth, but (gasp!) no category for your favorite tenor, Luciano Pavarotti! And your list is PAVAROTTI-L. Not to worry, however. Because the category of "Singers" is open, you can simply code:

```
* Categories= Arts:Music:Opera:Singers:Pavarotti_Luciano
```

and LISTSERV will accept the new subcategory "Pavarotti_Luciano".



Notes: When you create a new category, it will not show up until the central categories list has been updated.

There are two "root level" open categories, `Misc` and `Local`. The `Misc` category is world-searchable. If, however, you code a `Local` category, it will only be searchable from the search engine running on the server hosting your list.

- **Closed Categories** – These are categories that cannot be added to. In other words, if you see a category like:

```
Computers:Internet:Mailing_List_Managers:LISTSERV:Manuals:
List_Owners_Manual List Owner's Manual for LISTSERV
```

whose description does not indicate that it is open, then you cannot add new categories after the last term. If you try to create a new subcategory under a closed category, you will receive an error message when you PUT your list header, and your updated header will not be stored.

20.3.1 Examples of Category Settings

Categories are defined by the new "Categories=" list header keyword. Each category string's subcategories are internally delimited with colon (":") characters. Each separate category string is separated from the others with commas. If your "Categories=" keyword setting gets too long to fit on one line, simply define multiple "Categories=" keywords. Note that spaces are not allowed in categories; therefore

```
* Categories= Arts:Music:Opera:Singers:Luciano Pavarotti
```

is not legal, but

```
* Categories= Arts:Music:Opera:Singers:Luciano_Pavarotti
```

is.

A simple category setting would be:

```
* Categories= Arts:Music:Opera:Singers
```

and if someone searched on that category, they would find our list. But we saw above that we can create a new category if we are running a list dedicated to Luciano Pavarotti. So instead, we might code

```
* Categories= Arts:Music:Opera:Singers:Pavarotti_Luciano
```

If, however, we're running a list for the Three Tenors, we might want to code:

```
* Categories= Arts:Music:Opera:Singers:Pavarotti_Luciano
```

```
* Categories= Arts:Music:Opera:Singers:Domingo_Placido
```

```
* Categories= Arts:Music:Opera:Singers:Carreras_Jose
```

Or even:

```
* Categories= Arts:Music:Opera:Singers:Three_Tenors
```

depending on our preference.

If you code a sub-category that does not exist in a "closed" upper-level category, LISTSERV will respond with an error message that will list the legal sub-categories that you can use.

20.4 Implementing the INFO <listname> Command

This functionality is not available in LISTSERV Lite.

Section 18 [Working with Mail and Web Templates Using Email](#) includes details on how to include an informative paragraph in the information mail template file for your list. When a user sends the command INFO listname to your server, LISTSERV responds with either:

- The default response, which simply sends a copy of the list header to the user; or
- The customized paragraph included in the *listname*.MAILTPL file.

If *listname*.MAILTPL does not exist, the default response is sent. Also note that the user may send the INFO listname command to any L-Soft LISTSERV host (including the Global List Exchange discussed below), which will forward the request to the appropriate server.

20.5 The Global List Exchange (GLX)

The Global List Exchange, or GLX, is a central clearinghouse for LISTSERV subscriptions and List of List requests. For instance, If a user knows the name of a list but not the name of the host server, GLX simplifies the process by giving the user a single address where all subscription requests for lists running on L-Soft's LISTSERV can be sent.

By adding the GLX address in all advertisements for your list, you help other list owners as well as yourself by making it simple for users to subscribe to any list. Additionally, if for some reason a user is unable to contact your server directly, the GLX gives him an alternate subscription method.

The GLX address is LISTSERV@LISTSERV.NET.

20.6 How NOT to Advertise a Mailing List

It is generally considered a breach of netiquette to invade the privacy of other lists with a broadcast announcement that your list is up and running. The only time when this might be acceptable is when your list addresses a concern of people already subscribed to another list. If you feel it necessary to post an announcement on someone else's list, it is good manners to first send private mail to the owner of that list and ask his or her permission to do so. (The same policy applies to USENET newsgroups, though it may be more difficult to find out who the moderator is.)

It is certainly a breach of netiquette (and many networks' appropriate use policies) to blindly post multiple copies of your announcements to multiple lists. This kind of behavior is termed a "spam", something about which you may read more in Section 15 [Moderating and Editing Lists through Email](#). This kind of announcement is guaranteed to reap a good deal of bad will and may well result in the revocation of your network privileges.

Appendix A: LISTSERV Commands

This section is divided into five parts. The first three correspond to those commands available for use by the general user, list owners and file owners, and the LISTSERV maintainer. The last two describe how to send commands to LISTSERV and how to register LISTSERV passwords. Non-privileged users can send commands by mail or by interactive commands. (Note that interactive commands can only be sent if a two-way NJE or MSGD connection exists.) Privileged users can send commands by mail, interactive commands (subject to the same restriction previously noted) or via the console (VM) or the LCMD utility (non-VM).

Unless otherwise noted, commands are listed in alphabetical order, with the minimum acceptable abbreviation in capital letters. Angle brackets are used to indicate optional parameters. All commands which return a file accept an optional 'F=format' keyword (without the quotes) that lets you select the format in which you want the file sent; the default format is normally appropriate in all cases. Some esoteric, historical or seldom-used commands and options have been omitted.



Note: Some commands are not available on all platforms; these commands are marked appropriately.

Continuation cards (see the [Advanced Topics Manual for LISTSERV](#) regarding LISTSERV's Command Jobs Language) can be used to split long commands (for instance, ADD commands for users with long X.500 addresses) into two or more 80-character cards. In that case you must insert "// " (two slashes followed by a space) before the command text and a comma at the end of each line of the command so that CJLI considers it as a control card and performs the required concatenation. For instance,

```
// QUIET ADD MYLIST someone.with.a.real.long.userid.that.wraps@hishost.com ,  
His Name
```

or, for instance, for a large GETPOST job,

```
// GETPOST MYLIST 10769-10770 10772 11079 11086 11095 11099-11100 11104 ,  
11111 11115 11118 11121 11124 11131 11144 11147 11153 11158 11166 11168
```

Be sure to put a space before the comma at the end of the first line, as LISTSERV will not add the space for you.

General Commands

List Subscription Commands

The following commands are listed from most to least important.

SUBscribe listname [full_name | ANONYMOUS] [WITH options]

The SUBscribe command is LISTSERV's basic command, issued by users to join mailing lists. This command can also be used to change one's "full_name" field in LISTSERV's SIGNUP database (simply reissue the command with the changed name). Note that the full_name is not required if the user has previously signed up to lists on the same LISTSERV server, or if the user has previously registered in LISTSERV's SIGNUP database by using the REGISTER (q.q.v.) command.

The following syntax:

```
SUBSCRIBE listname ANONYMOUS
```

indicates that the user wishes to join the list anonymously, that is, without specifying a name. The CONCEAL subscription option is automatically set, granting the subscriber the maximal level of protection available.

The following additional syntax:

```
SUBSCRIBE listname full_name WITH option1 option2 ...
```

allows you to "preset" subscription options at subscribe time. For instance, you might want to subscribe to MYLIST-L in order to be able to search its archives, but don't want to receive postings. You would use the command

```
SUBSCRIBE MYLIST-L Joe User WITH NOMAIL
```

Or you might want to receive individual postings with the SUBJecthdr option and receive copies of your own postings instead of the standard acknowledgement that your message was distributed to the list:

```
SUBSCRIBE MYLIST-L Joe User WITH SUBJecthdr REPRO NOACK
```

Also,

```
QUIET SUBSCRIBE listname full_name WITH option1 option2 ...
```

suppresses the command response normally sent by LISYSERV that looks like this:

Confirming:

```
> SUBSCRIBE MYLIST-L Joe User
```

```
You have been added to the MYLIST-L list.
```

JOIN listname [full_name | ANONYMOUS] [WITH options]

JOIN is a synonym for SUBscribe.

SIGNOFF listname|*|* [(NETWIDE)]

The SIGNOFF command allows the user to cancel his or her subscription to lists.

SIGNOFF requires a qualifying parameter, as follows:

listname	Sign off of the specified list
*	Sign off of all lists on that server
* (NETWIDE	Sign off of all lists in the LISYSERV network

The "*" (NETWIDE" parameter causes the LISYSERV server to forward a copy of the signoff request to all other registered LISYSERV servers. This is a good option for a user who is changing service providers or otherwise losing a specific address that will not be forwarded. Please note that this parameter will not remove the user from non-LISYSERV lists or from LISYSERV lists running on non-registered sites. It will also not work if the server to which you issue it is running in STANDALONE runmode.

LISYSERV will attempt to sign off the address it finds in the RFC822 "From:" line and will not "fuzzy match" for "similar" addresses. The single exception is when the hostname part of the address is aliased to another hostname in LISYSERV's centrally-maintained ALIASES NAMES file.

UNSUBscribe listname|*|* [(NETWIDE)]

UNSUBscribe is a synonym for SIGNOFF.

CHANGE listname|* newaddr

This form of the **CHANGE** command can be used by any subscriber. It must be sent from the currently-subscribed address and results in an OK confirmation request being sent back to that address. This cookie then **MUST** be confirmed by the currently-subscribed address, exactly as it was entered, or the command will fail. This is the only case where a **LISTSERV** cookie must be confirmed by a specific address. Note that this assumes that the user still has login access to both addresses, or at least the ability to send mail from the old address.

SET listname option1 [option2 ...]

Allows the user to change his or her subscription options without administrative intervention. The options available to be changed are as follows:

- **ACK** – A mail message acknowledging the receipt and distribution of the user's posting is sent back to the user.
- **NOACK** – No posting acknowledgement is sent. In general, this setting should only be used if the user has also set himself to **REPRO**, as it is desirable in most cases that some indication of whether or not the posting was received by **LISTSERV** be sent.
- **MSGack** – (Obsolete) An interactive message is sent to acknowledge receipt and distribution. Note that this works only if both the machine running **LISTSERV** and the user's machine have **NJE** connectivity (e.g., **BITNET**). If **NJE** connectivity is not available on both ends, this option is effectively the same as **NOACK**.
- **CONCEAL** – Allows the user to be concealed from the **REVIEW** command. Note that the list owner or **LISTSERV** maintainer can always get the complete list of subscribers, regardless of this setting.
- **NOCONCEAL** – "Unhides" the user.
- **Files/NOFiles** – (Obsolete) These options toggle the receipt of non-mail files from the list. Note that this is **NJE**-specific, and thus obsolete for systems without **NJE** connectivity, but retained for compatibility.
- **Mail/NOMail** – These options toggle the receipt of mail from the list. Users who will be away from their mail for an extended period of time may prefer to simply turn the mail off rather than to unsubscribe, particularly if subscription to the list is restricted in some way.



Note: For backward compatibility, the command **SET listname MAIL** sent by a user who is set to **DIGEST** but not also set to **NOMAIL** will cause the user to be set to **NODIGEST** (the behavior is identical for users set to **INDEX** but not to **NOMAIL**). **SET listname MAIL** sent by users set to **DIGEST/NOMAIL** or **INDEX/NOMAIL** will simply remove the **NOMAIL** setting and leave the user set to **DIGEST** or **INDEX** as the case may be.

- **DIGests/INDEX/NODIGests/NOINDEX** – These options change the format in which list mail is received by the subscriber. **DIGEST** turns on digest mode, in which the subscriber receives a digest of postings at set times dependent on how the "Digest=" keyword of the list is set. **INDEX** turns on index mode, in which the subscriber receives a daily listing of subjects posted to the list, from which he or she

may order postings of interest. NODIGEST and NOINDEX toggle the mode back to individual postings sent as received by LISERV. Note that these options are interrelated; setting one will negate another.

- REPro/NOREPro – Causes LISERV to send you a copy of your own postings as they are distributed. Some users may prefer this behavior to the ACK option (see above).
- MIME/NOMIME – Toggles MIME options on and off. Currently only digests are available in MIME format. If DIGEST mode is set, the user will receive a MIME digest instead of the regular plain-text digest. Note that you must have a mail client that supports MIME digests (Pegasus is one that does) or this setting will do you little good. This option is automatically set at subscribe time for users who send their subscription command using a MIME-compliant agent, unless "Default-Options=NOMIME" is specified for the list.
- HTML/NOHTML – Toggle the HTML function for digests and indexes on and off.
- TOPICS: ALL | [+/-]topicname – For lists with topics enabled (see the Topics= list header keyword), subscribe or unsubscribe to topics. For instance, if a list has topics SUPPORT and CHAT, a user could subscribe to CHAT by sending SET TOPICS +CHAT. Or the user could unsubscribe to SUPPORT by sending SET TOPICS -SUPPORT. Finally, the user can subscribe to all available topics by sending SET TOPICS ALL.

Options for mail headers of incoming postings (choose one):

- FULLhdr – "Full" mail headers, (default) containing all of the routing information.
- IETFhdr – Internet-style headers.
- SHORThdr – (Obsolete) Short headers (only basic information about the message - Date, From, Subject, To -- is preserved). Setting SHORThdr will break MIME-encoded messages, so it should be used only on lists where MIME and HTML messages are not allowed.
- DUALhdr – Dual short headers, useful with older mail programs which do not preserve the RFC822 return email address. Same caveat as with SHORThdr.
- SUBJecthdr – "Full" mail headers (like the default) except that setting this option tells LISERV to add the list's default subject tag to the subject line of mail coming from the list. (See the listing in the [List Keyword Reference](#) document for "Subject-Tag=" for more information.) Note that if the user is set to SHORThdr (or any other header option other than FULLhdr), LISERV will automatically switch the user to FULLhdr, as subject tags require full headers. A subject tag is generated (for subscribers with the SUBJecthdr option set) even if the original message had no "Subject:" header. To turn the subject tagging off, the user simply sends a new SET command with any of the other header options (e.g., SET listname FULLhdr) and the SUBJecthdr option is reset.
- FULL822 – Essentially the same as "full" mail headers, but with the important difference that the recipient's email address is specified in the "To:" line rather than the address of the list. "FULL822" headers should be used with extreme caution, as

they cause LISTSERV to create a separate mail envelope with a single RFC821 RCPT TO: for each address so set. This behavior can significantly affect the performance of both LISTSERV and of your external mail system.

- `SHORT822` – Essentially the same as "short" mail headers, with the same caveats as noted for `FULL822`.



Note: `FULL822` and `SHORT822` headers should only be used if a specific problem indicates that they might solve the problem. One possible use would be to determine which subscriber from a specific site is causing the site to throw back delivery errors if that site does not specify which RCPT TO: is generating the error. These headers should never be used by default.



Documented Restriction: The use of the `SHORTHDR` or `DUALHDR` options will break messages that depend on MIME encoding, because these options strip the RFC822 headers that identify the message as a MIME message. `SHORTHDR` and `DUALHDR` were designed for the non-MIME mail clients which prevailed in LISTSERV's early history. As most mail clients today support MIME, the use of these options is now deprecated.

`CONFIRM listname1 [listname2]...]]`

The `CONFIRM` command should be issued when LISTSERV requests it. A request for `CONFIRM` should not be confused with a "command confirmation request" which requires an "OK" response. The `CONFIRM` command is used in two cases:

- When the list in question requires periodic subscription renewals (controlled by the `Renewal= keyword`). In this case, the amount of time between the request for confirmation and termination of the subscription is controlled by the `Delay()` parameter of the `Renewal= keyword`; the default is seven days.
- When LISTSERV's automatic address probing function fails and you receive a message to that effect. The response time is controlled by the settings of the `Auto-Delete= keyword` for the list in question.

Other list-related commands

`INDEX [listname]`

The `INDEX` command sent to LISTSERV without further qualification sends back the contents of the "root" level archive filelist on VM systems (LISTSERV FILELIST) or archive catalog on non-VM systems (SITE.CATALOG plus the contents of SYSTEM.CATALOG).

If the `INDEX` command is sent with the name of a list (e.g., `INDEX MYLIST`) or the name of a special filelist or catalog file (e.g., `INDEX TOOLS`, if `TOOLS FILELIST` on VM or `TOOLS.CATALOG` on non-VM exists), LISTSERV sends back the contents of the specified filelist or catalog. Several possibilities exist:

- For mailing lists without an associated filelist or catalog, LISTSERV creates an index "on the fly" containing entries for the accumulated notebook archives for that list. If notebook archives are not enabled for the list, LISTSERV will respond, "This server does not have any file by the name '*listname.filelist*'."

- For mailing lists with an associated filelist or catalog, LISTSERV will append the "on the fly" index of notebook archives to the entries in the associated filelist or catalog. For instance, for a list called MYLIST with associated catalog MYLIST.CATALOG, INDEX MYLIST might return:

Table 21-1 Sample Output of an INDEX Listname Command

```

*
* MYLIST FILELIST from LISTSERV@LISTSERV.MYCORP.COM
*
* ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
*
* The GET/PUT authorization codes shown with each file entry describe
* who is authorized to GET or PUT the file:
*
*     ALL = Everybody
*     CTL = LISTSERV administrators
*     OWN = List owners
*     PRV = Private, ie list members
*     LMC = LISTSERV master coordinator
*     N/A = Not applicable - file is internally maintained by LISTSERV
*     MSC = Miscellaneous - contact administrator for more information
*
* ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
*
*
* Information files for MYLIST
*
* filename      filetype      GET PUT size (bytes) date      time
* -----
* MYLIST        FAQ            ALL MSC      22,528 1996-02-09 21:30:10
* MYLIST        WELCOME        ALL MSC       279 1998-02-02 09:59:44
* MYLIST        FAREWELL        ALL MSC       92 1998-02-05 11:06:14
*
*
* Archive files for the MYLIST list at LISTSERV.MYCORP.COM
* (monthly logs)
*
* filename      filetype      GET PUT size (bytes) date      time
* -----
* MYLIST        LOG9603        LOG OWN      8,668 1998-05-27 15:29:57
* MYLIST        LOG9605        LOG OWN      7,865 1998-06-29 08:43:26
* MYLIST        LOG9606        LOG OWN     17,298 1998-07-23 12:46:20

```

- Lastly, for catalogs or filelists without an associated list, the INDEX command returns only the entries in the catalog or filelist, since there are no associated list archives to be indexed.

Under VM, instead of the size in bytes, three separate VM-specific columns are used. Please note the following definitions for them:

- rec -fm – Indicates whether the file is in a fixed or variable record format.

- `lrecl` – Logical record length. For a file with fixed record format (F), this is the length of each record. For a file with variable record format (V), this is the maximum record length.
- `nrecs` – Number of records (lines) in the file.

Lists [option]

Access the global list of lists maintained by LISTSERV. If no options are specified, then LISTSERV returns only local lists, one line per list. The available options are:

- `Detailed` – All local lists, complete with full header information.
- `Global xyz` – Only those whose name or title contains 'xyz'.
- `SUMmary [host]` – Membership summary for all lists on specified host, or the host to which the command is sent if no host is specified.
- `SUMmary ALL` – Membership summary for all hosts (long output, send request via mail).
- `SUMmary TOTAL` – Membership totals only.

"Lists Global" without a search string, which returns the entire list of lists, may no longer be issued by general users. General users should use the "Lists Global /xyz" format to search the list of lists, or (preferably) use L-Soft's CataList service at <http://www.lsoft.com/catalist.html>.

"Lists SUMmary", when issued to an unregistered host or to a host running in STANDALONE mode will generate the response "No information available yet - please try again later." because the file required for this function does not exist.

Query listname

Query your subscription options for a particular list (use the SET command to change them). Using the "*" wildcard in place of the name of a single list queries subscription options on all lists on the server.

REGister full_name | OFF

Register's the user's full name field in LISTSERV's SIGNUP files, or changes the current value of that field. When a user's name is registered, he or she can omit the full name field from subsequent SUBscribe requests to that server. Sending "REGISTER OFF" to LISTSERV deletes the user's entry from the SIGNUP file.

REView listname [(options)]

Get information about a list, assuming the list header keyword "Review=" is set appropriately. For general users, REVIEW does not return address information about subscribers who are set to CONCEAL. The options are:

- `BY sort_field` – Sort list in a certain order:
 - `Country` – Sorts by country of origin (ISO country codes).
 - `Date` – Sorts by subscription date (newest to oldest).
 - `Name` – Sorts by user name (last, then first).
 - `NODEid` – Sorts by hostname/nodeid
 - `Userid` – Sorts by userid

- `BY (sort_field1 sort_field2)` – You can specify more than one sort field if enclosed in parentheses. For instance, `BY (NODE NAME)`.
- `Countries` – Synonym of `BY COUNTRY`.
- `Topics` – Adds a breakdown of subscribers per topic (if `Topics=` is defined in the list header) at the end of the subscriber list. If you just want the breakdown, use `REVIEW listname SHORT TOPICS`. This does not show topics by individual subscribers (see the `QUERY` command instead). If `Topics=` is not enabled for a given list then this option is ignored.
- `LOCal` – Don't forward request to peers. This is only useful if the list is peered; normally it should not be necessary to issue this option.
- `Msg` – Send reply via interactive messages (BITNET users only).
- `NOHeader` – Don't send list header, just send the subscriber list.
- `Short` – Don't list subscribers, just send the header and the membership summary for the list.



Note: You can get a quick read of the number of subscribers on the list by sending the command `REVIEW listname SHORT NOHEADER`.

List owners and site maintainers may also use the additional option:

- `ALL` – List concealed members (who will show up with "[concealed]" next to their entry) as well as non-concealed members. (NOT available to general users even if `Review= Public`.)

SCAN listname text

Scan a list's membership for a name or address. Helpful if a user attempts to send a `SET` command or an `UNSUB` command and is informed that their address is not subscribed to the list. At the non-privileged level, this command will show all non-concealed entries that match the search text, assuming that the list is set to "Review= Public".

Stats listname [(options)]

The following command is available on VM servers only.

Get statistics about a list. This command is VM-specific, and was originally intended to return BITNET data. The single option is:

- `LOCal` – Doesn't forward to peers.

Informational commands

Help

Obtain a list of commonly-used LISTSERV commands. Also explains how to get the comprehensive reference card and tells who the (non-hidden) server manager(s) are.

Info [topic|listname]

Order a LISTSERV manual, or get a list of available ones (if no topic was specified); or get information about a list. For `Info listname`, the text in the INFO template form of `listname.MAILTPL` is used; however, if `listname.MAILTPL` does not exist or does not contain an INFO template form, the INFO template form of `DEFAULT.MAILTPL` is used.

Query File fn ft [filelist] [(options)]

(Available only on VM) Get date/time of last update of a file, and GET/PUT file access code. The single option is:

- **Flags** – Get additional technical data (useful when reporting problems to experts).

RELEASE

Find out who maintains the server, the version and build date of the software, and the version of LISTSERV's network data files.

SHOW [function]

Display information as follows:

- **ALIAS** *node1* [*node2* [...]] – BITNET nodeid to Internet hostname mapping.
- **DISTRIBUTE** – Statistics about DISTRIBUTE.
- **HARDWARE** or **HW** – Hardware information; what kind of machine is LISTSERV running on?
- **LICENSE** – License/capacity information and software build date.
- **LINKS** [*node1* [*node2* [...]]] – Network links at the BITNET node(s) in question.
- **NADS** [*node1* [*node2* [...]]] – Addresses LISTSERV recognizes as node administrators for the specified site(s).
- **NODEntry** [*node1* [*node2* [...]]] – BITEARN NODES entry for the specified node(s).
- **NODEntry** *node1* /*abc**/*xyz* – Just the ':xyz.' tag and all tags whose name starts with '*abc*'.
- **POINTS** [ALL | *list1* [*list2*...]] – Graduated (LISTSERV Classic) license point information. This information can help you plan orderly expansion of your site if you are running with a graduated LISTSERV Classic license. Under Lite this command shows Classic point usage.
- **STATs** – Usage statistics for the server (this is the default option).
- **VERSION** – Same output as **RELEASE** command.

If no function is specified, the output is per **SHOW STATs**.

The following options are available for VM servers only:

- **BITEARN** – Statistics about the BITEARN NODES file.
- **DPATHs** *host1* [*host2* [...]] – DISTRIBUTE path from that server to specified host(s).
- **DPATHs** * – Full DISTRIBUTE path tree.
- **FIXes** – List of fixes installed on the server (non-VM see **SHOW LICENSE**).
- **NETwork** – Statistics about the NJE network.
- **PATHs** *snode node1* [*node2* [...]] – BITNET path between '*snode*' and the specified node(s).

Commands Related to File Server and Web Functions

GET

- VM Syntax

```
GET fn ft [filelist] [(options) [F=fformat] [SPLIT=integer]
```

- Non-VM Syntax

```
GET fn ft [catalogname] [[F=fformat] [SPLIT=integer]]
```

Order the specified file or package from LISTSERV. The single option is for VM servers only and is:

- PROLOGtext xxxx – Specify a 'prolog text' to be inserted on top of the file

Examples:

```
GET MYFILE TEXT
```

```
GET MYFILE TEXT MYLIST-L
```

Typically the filelist name or catalog name is the same as the name of the list to which the files belong. A password (PW=xxxxx at the end of the command) is required to retrieve any file that does not have a GET FAC of ALL. For more information on FAC (File Access Control) codes, see the [Site Manager's Manual for LISTSERV](#).

Do not use dots (periods) in the file specification when specifying the filelist or catalog name, as this will result in an error. For instance

```
GET MYFILE.TEXT MYLIST-L
```

will result in an error, whereas

```
GET MYFILE TEXT MYLIST-L
```

will not.

To control the format in which LISTSERV returns the file(s) to you, you can specify the *F=fformat* parameter. Supported formats are Netdata, Card, Disk, Punch, LPunch, UUencode, XXencode, VMSdump, MIME/text, MIME/ Appl, Mail.

To split very large files into manageable chunks, you can specify the *SPLIT=integer* parameter. The integer value is the size you want the chunks to be generated, in kilobytes. For instance if you were ordering a 2MB notebook log and wanted to break it into 100KB chunks, you would specify *SPLIT=100*. This is handy for people whose mail systems place a limit on the size of an individual mail message that may be received by a given user.

The following syntax is accepted for **TCPGUI parameters for Change-Logs**:

```
GET listname CHANGELOG (MSG [COLUMNS(colspec1 colfilter1  
[colspec2 colfilter2[...]])
```

The design goal was to provide access via the TCPGUI (and thus the web interface) for change-log data. Further information can be found in the [Advanced Topics Guide for LISTSERV](#).

GIVE

- VM Syntax: `GIVE fn ft [filelist] [TO] userid@host`

- Non-VM Syntax:

```
GIVE fn.ft [TO] userid@host
```

or

```
GIVE fn ft catalogname [TO] userid@host
```

Sends a file stored in a LISTSERV file archive to someone else. For instance, you may want to send LISTSERV REFCARD to a new user. Rather than retrieving LISTSERV REFCARD and then forwarding it to the user, you simply issue a GIVE command to tell LISTSERV to send it directly. Note that the token "TO" is optional.

Examples:

- For LISTSERV running under VM:

```
GIVE LISTSERV REFCARD joenewuser@hishost.com
```

```
GIVE LISTSERV REFCARD TO joenewuser@hishost.com
```

```
GIVE README TEXT MYLIST-L joenewuser@hishost.com
```

```
GIVE README TEXT MYLIST-L TO joenewuser@hishost.com
```

- For LISTSERV running on non-VM hosts there are two syntaxes, depending on whether or not you need to specify a catalog name for the file in question. Note that the only real difference is whether or not you are required to specify a dot between the filename and the extension.

```
GIVE LISTSERV.REFCARD joenewuser@hishost.com
```

```
GIVE LISTSERV.REFCARD TO joenewuser@hishost.com
```

```
GIVE README TXT MYLIST-L joenewuser@hishost.com
```

```
GIVE README TXT MYLIST-L TO joenewuser@hishost.com
```

INDEX [filelist|catalog]

Same as GET xxxx FILELIST. If no filelist is specified, the default is LISTSERV FILELIST (on non-VM, SITE CATALOG is returned as LISTSERV FILELIST in this case).

PW function

Define/change a "personal password" for protecting AFD/FUI subscriptions, authenticating PUT commands, and so on.

- `ADD firstpw` – Define a password for the first time, or after a PW RESET. Requires confirmation via the "OK" confirmation method.
- `CHange newpw [PW=oldpw]` – Change your existing password. If you do not include your old password for authentication, LISTSERV will require confirmation via the "OK" confirmation method.
- `REP password` – Starting with 1.8d, this function is a hybrid of "ADD" and "CHange". If a password does not exist for the user, one will be added. If a password does exist for the user, it will be changed (with confirmation required via the "OK" confirmation method). "REP" was added primarily for use by the web archive and administration interface but can be used in e-mailed PW commands as well.

- **RESET** – Reset (delete) your password. This function always requires confirmation via the "OK" confirmation method.

SENDme

Same as GET.

AFD

Available on VM servers only.

Automatic File Distribution. The functions are as follows:

- **ADD** fn ft [filelist [prolog]] – Add file or generic entry to your AFD list.
- **DELeTe** fn ft [filelist] – Delete file(s) from your AFD list (wildcards are supported).
- **List** – Displays your AFD list.

For node administrators:

- **FOR** user **ADD/DEL/LIST** etc – Perform requested function on behalf of a user you have control over (wildcards are supported for DEL and LIST).

FUI

Available on VM servers only.

File Update Information: same syntax as AFD, except that FUI ADD accepts no 'prolog text'.

Other Advanced Commands

DISTtribute type source dest [options]



Note: Starting with 1.8d, the ability to send DISTRIBUTE jobs is limited to LISTSERV Maintainers by default, and requires a password. This section is retained for compatibility with 1.8c and earlier, and for 1.8d and later servers which have the DISTRIBUTE security feature turned off.

Distribute a file or a mail message to a list of users (see the Advanced Topics Guide for LISTSERV for more details on the syntax). The various parameters are, briefly:

Type:

- **MAIL** – Data is a mail message, and recipients are defined by '<dest>'.
- **MAIL-MERGE** – Data is a mail-merge message. See the [Advanced Topics Guide for LISTSERV](#) for specifics.
- **FILE** – Data is not mail, recipients are defined by '<dest>'.
- **POST** – (non-VM only) Same as MAIL except that the message is pre-approved. See the [Advanced Topics Guide for LISTSERV](#) for specifics.
- **RFC822** – Data is mail and recipients are defined by the RFC822 'To:/'cc:' fields.

Source:

- **DD=ddname** – Name of DD holding the data to distribute (default: 'DD=DATA').

Dest:

- `<TO> user1 <user2 <...>>` – List of recipients.
- `<TO> DD=ddname` – Use a DD called ddname for the destination addresses, one recipient per line.

Options for the general user:

- `ACK=NOne/MAIL/MSG` – Acknowledgement level (default: `ACK=NONE`).
- `CANON=YES` – 'TO' list in 'canonical' form (uid1 host1 uid2 host2...).
- `DEBUG=YES` – Do not actually perform the distribution; returns debug path information.
- `INFORM=MAIL` – Send file delivery message to recipients via mail.
- `TRACE=YES` – Same as `DEBUG=YES`, but file is actually distributed.
- `AV=YES[, FORCE]` – Check the message for viruses. See the [Advanced Topics Guide for LISERV](#) for specifics.
- `DKIM=NO/YES` – Sign the message with a DomainKeys signature (default: `DKIM=NO`) See the [Site Manager's Manual for LISERV](#) for specifics.

Options requiring privileges:

- `FROM=user` – File originator
- `FROM=DD=ddname` – One line: 'address name's
- `PRE-APPROVED=YES` – Pre-approve message (with `DISTRIBUTE POST` only)

GETPost listname post_number [post_number [...]] [NOMIME]

GETPost is used after receiving the output of a `SEARCh` command to retrieve the postings you want from the `SEARCh` output. For instance, if you want postings numbered 1730, 1731, 1732, and 1840 from the MYLIST list, send the command

```
GETPost MYLIST 1730-1732 1840
```

GETPost is analogous to the VM database command `PRINT`.

In previous versions, the GETPost command returned messages that contained MIME attachments in their "raw" form, which could not be extracted automatically by MIME-aware mail clients. Customers who wished to use list notebooks to archive word-processing documents (for instance) found this to be a problem. From LISERV 1.8e, attachments returned in messages by way of the GETPost command will now display as inline clickable links in the individual messages.

Users of certain email clients (specifically Pine, which handles attachments in a secondary viewing area) may find the new format difficult to use. If preferred, the pre-1.8e behavior may be reverted to by specifying "NOMIME" as the last parameter of the GETPost command.

Search

For lists running on VM servers, see also below at `DATABASE`.

The Search command syntax is similar to that of the `SEARCH/SELECT` commands in the "old" database functions. A very basic Search command for list MYLIST would look like this:

Search *search_string* IN MYLIST

You can also restrict your search by date, sender, or other criteria, for example,

Search *search_string* IN MYLIST SINCE 96/01/01

Search *search_string* IN MYLIST WHERE SENDER CONTAINS ERIC

The specific syntax is outlined in LISTDB MEMO (available from LISERV with the command "INFO DATABASE") and in the [Advanced Topics Guide for LISERV](#).



Note: The "new" Search command does not require a CJLI job framework to operate; simply send the Search command in the body of an email message to the appropriate server. LISERV will respond with an index of the postings matching your criteria and instructions on how to use the GETPost command to retrieve the posts you want.

SERVE user

Restore service to a user whose access to LISERV has been disabled. This generally occurs when a user has sent 51 incorrect commands in a row to LISERV, which LISERV interprets as a possible mail loop. (Note also that certain mail packages that send "Read:/Not Read:" notifications back to LISERV will trigger this scenario after 51 iterations. The best solution would be for the user to disable receipt notifications.) The user in question cannot restore his or her own service; this command must be issued from another userid. Note that if the user has been manually served out by privileged user (a LISERV maintainer), the SERVE command must be issued by a similarly-privileged user (who must also be a LISERV maintainer, although naturally the same user who issued the SERVE OFF command can issue the SERVE command).



Note: The THANKS command will not reset the serve-off counter (so vacation messages or auto-replies that contain a sentence starting with something like "Thanks for writing" will not defeat the system and users sending them will eventually be served off instead of continuing to loop ad infinitum).

THANKS

You can send this command to check to see if the server is alive. If it is, the server politely responds, "You're welcome!".

DATABASE function

This command is only available on VM servers:

Access LISERV database(s). The functions are explained in detail in the version of LISTDB MEMO available from VM servers, but the basic syntax is:

- Search DD=*ddname* <ECHO=NO> – Perform database search (see the VM version of LISTDB MEMO for more information on this)
- List – Get a list of databases available from that server
- REFRESH *dbname* – Refresh database index, if suitably privileged

Dbase

This command is only available on VM servers:

Same as DATABASE

List Owner and File Owner Commands

File Management Commands (for file owners only)

`PUT fn ft <filelist <NODIST>>`

Update a file you own. Options are:

- `<PW=password>` – Supply your password for command authentication.

The following options are VM-specific and will not work on the non-VM servers.

The "NODIST" option prevents AFD and FUI distributions when the file is updated. Other available VM only options include:

- `<CKDATE=NO>` – Accept request even if the current version of the file is more recent than the version you sent
- `<DATE=yyymmddhhmmss>` – Set file date/time.
- `<RECFM=F <LRECL=nnn>>` – Select fixed-format file (not to be used for text files).
- `<REPLY-TO=userid>` – Send reply to another user.
- `<REPLY-TO=NONE>` – Don't send any reply.
- `<REPLY-VIA=MSG>` – Request reply via interactive messages, not mail (Requires NJE connectivity).
- `<"parameters">` – Special parameters passed to FAVE routine, if any.

Standard parameters supported for all files:

- `TITLE=file title` – Change file "title" in filelist entry.

AFD/FUI

This command is only available on VM servers.

Automatic File Distribution privileged commands. In addition to the AFD/FUI functions listed above, a file owner may use the following function:

- `GET fn ft <filelist>` – Get a list of people subscribed to a file you own.

GET fn FILELIST <(options>

This command is only available on VM servers.

Special options for filelists:

- `CTL` – Return filelist in a format suitable for editing and storing back.
- `NOLOCK` – Don't lock filelist (use in conjunction with CTL).

REFRESH filelist <(options>

This command is only available on VM servers.

Refresh a filelist you own. The single option is:

- `NOFLAG` – Don't flag files which have changed since last time as updated (for AFD/FUI).

UNLOCK fn FILELIST

This command is only available on VM servers.

Unlock filelist after a GET with the CTL option if you decide not to update it after all

List Management Functions

Commands that support the QUIET keyword are marked with an asterisk (*).

ADD(*) listname user [full_name]

ADD(*) listname DD=ddname [IMPORT [PRELOAD]]

The first syntax is used to add an individual user to one of your lists, or update his name field. Note that you can substitute an asterisk (*) for *full_name* and LISTSERV will substitute "<No name available>" in the list.

The second syntax is used for bulk ADD operations where a dataset (DD=ddname) is used add multiple users, one address/name pair per line. For bulk operations you may also use the IMPORT option, which implies a QUIET ADD (in other words you do not need to specify QUIET if you use IMPORT) and otherwise vastly speeds up the ADD process by loosening syntax checking and omitting success messages. The IMPORT PRELOAD option is used to direct LISTSERV to preload the existing e-mail keys in memory before starting the transaction, which speeds the operation up considerably. This option is used primarily with DBMS lists to speed up bulk adds. PRELOAD is not necessary for traditional LISTSERV lists and does not normally lead to a significant performance improvement. However, when importing a new list (no existing subscribers), it does reduce CPU usage somewhat.

For a bulk ADD operation, the users are defined in a separate dataset beginning on the line following the ADD command. For instance,

```
ADD listname DD=ddname IMPORT
//ddname DD *
userid@host.com User Name
userid2@host.com User2 Name
... more address/name pairs, one per line ...
/*
```

Please see the [Site Manager's Manual for LISTSERV](#) for specific instructions for bulk ADD operations.

ADDHere(*)

Same as ADD, but means "add the user on this peer, do not forward this request to a (possibly) closer peer". For non-peered lists, is functionally identical to ADD.

CHANGE(*) listname | * newaddr

CHANGE(*) listname | * oldaddr | pattern newaddr | *@newhost

The first form can be used by any subscriber and results in a cookie being sent to the new address. This cookie MUST be confirmed by the new address, exactly as it was entered, or the command will fail. This is the only case where a LISTSERV cookie must be confirmed by a specific address.

The list owner form does not use cookies but simply applies the standard "Validate=" rules (as for a DELETE command). You can specify a wildcard pattern for the old address and *@newhost for the new address to rename certain addresses to a new hostname. The CHANGE1 template is sent unless you specify QUIET.

Change log entries are made (CHANGE oldaddr newaddr) and there is a CHG_REQ exit point which allows you to reject the operation.

DELeTe(*) listname user [(options)]

DELeTe(*) listname DD=ddname [BRIEF]

The first syntax is used to remove a single user from one of your lists, or from all local lists if listname is '*'. The available options are:

- Global – Forward request to all peers
- Local – Don't try to forward request to closest peer if not found locally
- TEST – Do not actually perform any deletion (useful to test wildcard patterns)

The second syntax is used for bulk DELETE operations (similar to a bulk ADD operation). See the [Site Manager's Manual for LISTSERV](#) for details. The single available option is:

- BRIEF – Good for deleting wildcard patterns (such as *@*) when you don't want a "userid@host has been deleted from list xxxx" for each user deleted. Returns instead only a count of the users that were deleted.

EXPLODE listname <(options)>

This command is only available on VM servers.

Examine list and suggest better placement of recipients, returning a ready-to-submit MOVE job.

- BESTpeers n – Suggest the N best possible peers to add.
- Detailed – More detailed analysis.
- FOR node – Perform analysis as though local node were 'node'.
- PREFER node – Preferred peer in case of tie (equidistant peers).
- SERVICE – Check to see that service areas are respected.
- With(node1 <node2 <...>>) – Perform analysis as though specified nodes ran a peer.
- WITHOut(node1 <node2 <...>>) – Opposite effect.

FREE listname <(options)>

Release a held list. The single option is:

- Global – Forward request to all peers.

GET listname <(options)>

Get a copy of a list in a form suitable for editing and storing the list and lock it so that other list owners can't modify it until you store it back (or until you or they issue an UNLOCK command). The options are:

- Global – Forward request to all peers.
- HEADer – Send just the header; on the way back, only the header will be updated. This is the recommended way to modify your list header.
- NOLOCK – Do not lock the list.

- OLD – Recover the "old" copy of the list (before the last PUT).

HOLD listname <(options)>

Hold a list, preventing new postings from being processed until a FREE command is sent. The single option is:

- Global – Forward request to all peers

Lists [option]

Additional options available for list owners and moderators:

- OWNed – Returns a list of local lists owned by the invoker.
- MODerated – Returns a list of local lists that are moderated by the invoker.

MOVE(*) listname user <TO> node

Move a subscriber to another peer. Do NOT use this command to move users from one list host site to another during migration. It is strictly for moving subscribers from one peer to another peer.

- listname DD=ddname – Move several subscribers to various peers

PUT listname LIST

Update a list from the file returned by a GET command. This is the standard "PUT command" or "list PUT" referred to throughout this document.

Use of the PUT command to store a list header with new subscriber data at the bottom (e.g., an attempt to add subscribers "on the fly") will result in only the header of the list being stored, and in the generation of the following warning:



Warning: New subscriber data was found in the replacement list you sent, possibly due to the use of a signature file with an unusual separator line. If you really meant to update the subscriber data, please resend your request with the word "PUT" replaced with "PUTALL". For now, only the list header will be updated.

PUTALL listname LIST

This command allows you to PUT an entire list file, that is, the list header followed by the list of subscribers.



Documented Restriction: PUTALL does not work with DBMS lists; only the header information is replaced. Subscriber information in the DBMS table is not changed. For DBMS lists where the subscriber information needs to be replaced *in toto*, either the DBMS should be manipulated with your regular DBMS tools or you should use ADD IMPORT.

Query listname <WITH options> FOR user

Query the subscription options of another user (wildcards are supported).

- * <WITH options> FOR user – Searches all the lists you own for the specified user(s) with the specified option(s).

REINDEX [options] list1 [list2 [...]]



Documented Restriction: Because it is resource-intensive, the REINDEX command is considered a "maintenance" command as opposed to an "everyday" command, and

should not be issued in production environments outside of regularly-scheduled maintenance windows.

The REINDEX maintenance command rebuilds the WWW indexes (default), database indexes, or both for one or more lists. This command requires list owner privileges.

The wildcard * or the literal ALL is supported to mean "all lists". List owners will be able to rebuild indexes only for their own lists. LISERV maintainers may rebuild indexes for all lists on the server.

The available option is:

DATABASE WWW BOTH IMMEDIATE



Notes: By default, web indexes will be rebuilt in the background on LISERV Classic HPO unless you specify the IMMEDIATE option. In additions, there is no background reindexing support for LISERV Classic non-HPO.

The main reason to use REINDEX is to retroactively add RSS abstracts to an existing list. It may also be useful if an archive file is edited by accident or if there is a corrupt file or the like. LISERV Classic non-HPO sites with one or more very large and active list(s) may also wish to run this command when upgrading to HPO.

On non-VM systems where there are no web indexes, normally there is no reason at all to rebuild your database indexes, other than in cases of disk corruption or in an upgrade to HPO coupled with the presence of a list with very large archives.

When rebuilding web indexes, BOTH the HTML files AND the INDxxxx files are reindexed. While there may be legitimate reasons for wanting to rebuild the HTML files (template changes, etc), it is NOT recommended to use the REINDEX command to accomplish that, as the overhead involved in rebuilding the INDxxxx files is prohibitive if they do not actually need rebuilding.

With LISERV Classic there is no background support, so REINDEX * should be scheduled for "quiet" periods if there are many lists and/or lists with large archives.

While a particular list has its indexes rebuilt, WA browsing accesses will generally fail and/or LISERV will throw "listname.html" not found errors. There is no good way to avoid that since the goal is to recreate the entire index structure from scratch, and during that process files will be deleted and not be available until they are rebuilt.

By default, database indexes are reset and will be rebuilt the next time they are needed; this is a fast operation. Under HPO, the IMMEDIATE option may be used to cause an immediate rebuild of all database indexes if absolutely necessary, but remember that the immediate rebuild will block the server from doing anything else until it completes.

However, this can be a useful command when migrating an entire server or perhaps just a single list during off hours and it is desired to pre-index everything so that the server is snappy when traffic increases in the morning.

SET(*) listname options <FOR user>

Alter the subscription options for other users (wildcards are supported when setting options for another user or set of users).

Additional options for list owners:

- NORENEW/RENEW – Waive subscription confirmation for this user.
- NOPOST/POST – Prevent user from posting to list.

- EDITor/NOEDITor – User may post without going through moderator.
- REView/NOREView – Postings from user go to list owner or moderator even if user is otherwise allowed to post.

Stats listname (RESET

This command is only available on VM servers.

Resets (BITNET) statistics for the list.

UNLOCK listname

Unlock a list after a GET, if you decide not to update it after all, or unlock a list if it has been locked by another list owner or by the LISTSERV maintainer. Note that if you are not the person who originally locked the list, it is considered good practice to ask the person who originally locked the list whether or not they are done with the list before you unlock it.

LISTSERV Maintainer Commands

All LISTSERV maintainer commands require a password for validation when issued by email. Commands issued by TELL or SEND from the local host or via the LCMD utility do not require password validation. (Commands issued by LCMDX do require password validation. LCMDX, the LISTSERV TCPGUI demonstration program, is not the same as the LCMD utility shipped with LISTSERV.)



DEBUG FLAGS [update1 [update2 [...]]]

Each update has the format:

[+|-]flag

where 'flag' can be either the name of a debug flag or its corresponding hex value. A plus sign turns on the flag, minus turns it off, no sign sets the specified flag and clears all others. The command updates the session debug flags as requested, then displays a summary:

```
* Debug flags for this session: 00000022
*
* 00000001 TRACE_DIST      [Trace DISTRIBUTE processing - OFF]
* 00000002 TRACE_MIME      Trace MIME parser
* 00000004 TRACE_LISTS     [Trace (a few) list-related functions - OFF]
* 00000008 TRACE_SPAM      [Trace spam filter calls - OFF]
* 00000010 TRACE_EMM       [Trace Embedded Mail-Merge processor - OFF]
* 00000020 TRACE_DEV       Temporary ad hoc tracing for development use
* 00000040 TRACE_FSAV      [Trace FSAV calls - OFF]
* 00000080 TRACE_LDAP_CALLS [Trace LDAP library calls - OFF]
* 00000100 TRACE_LDAP_DATA [Trace data obtained from LDAP - OFF]
* 08000000 HOLD_DISTBG     [Do not process background DISTRIBUTE jobs - OFF]
* 10000000 HOLD_XB64       [Hold X-B64 jobs instead of processing them - OFF]
* 20000000 KEEP_JOBFILES   [Keep successfully processed job files - OFF]
* 40000000 TRACE_TCPGUI    [Additional TCPGUI tracing - OFF]
```

Flags that have not yet been assigned a function will display as RESERVED in the summary and do nothing. The TRACE_DEV flag is used by L-Soft Development to trace various aspects of whatever is being worked on at the time. While TRACE_DEV may be enabled by site maintainers (and in certain cases the support department may request that TRACE_DEV be enabled for debugging purposes), the results will vary depending on the build (and will not be publicly documented for that reason).

Any changes made using the **DEBUG FLAGS** command are for the current session only (they will be reset when **LISTSERV** is restarted). For certain purposes it can make sense to have some flags permanently turned on, and the **DEBUG_FLAGS** site configuration variable remains available for those special cases.

FOR user command

Execute a command on behalf of another user (**LISTSERV** maintainers only). Note that this command is provided for debugging purposes only -- it provides a method for a **LISTSERV** maintainer to send commands "from" the specified user. It is not recommended to use this command syntax in production, for instance to issue **SET** or **SUBSCRIBE** or **UNSUBSCRIBE** commands on a user's behalf. For instance, the **LISTSERV** maintainer should use, respectively, the "SET listname options FOR userid@host", "ADD listname userid@host", or "DELeTe listname userid@host" syntaxes in preference to the "FOR userid@host command" syntax.

Lists [option]

- **Global** – All known lists, one line per list, sent as a (large!) file. Only **LISTSERV** maintainers may request this list, as it has become a favorite pastime of Internet mailbombers to issue **LIST GLOBAL** commands on behalf of users whose mailboxes they wish to bomb. You should direct users who request "the whole list of lists" to L-Soft's CataList service at <http://www.lsoft.com/catalist.html>.

Additional options available for site maintainers are:

- **OWNED BY internet_address** – Returns a list of local lists owned by the userid@host specified. Wildcards are acceptable.
- **MODerated BY internet_address** – Returns a list of local lists moderated by the userid@host specified. Wildcards are acceptable.

NODESGEN [WTONLY]

Regenerate all **LISTSERV** network tables, or just compile the links weight file (debugging command). This happens automatically when **LISTSERV** is rebooted if a new **BITEARN NODES** file is found. Otherwise you should issue a **NODESGEN** whenever you update **BITEARN NODES**.

PUT listname LIST

Create a new list. Requires the **CREATEPW** for validation when issued from a remote node. You may specify initial subscribers, one per line, following the list header when creating a list. See also the **PUTALL** command in the [Site Manager's Manual for **LISTSERV**](#).

PWC function

Password file management:

- **ADD user newpw** – Define a password for the specified user.
- **DELeTe user** – Delete password for that user.
- **Query user** – Query the password of the specified user.

REGister name|OFF FOR user

Set or delete a user's **SIGNUP FILE** entry.

SERVE user OFF [DROP] | LIST

SERVE user OFF permanently suspends access from an abusive user or gateway (restore service with *SERVE user*).

Adding "DROP" (for example, *SERVE user OFF DROP*) to the command is identical to *SERVE user OFF* except that the postmaster will not receive any notification messages from LISTSERV when/if the user continues to try to post.

Issuing a *SERVE LIST* command causes LISTSERV to return a list of all users who are currently served off or who are spam-quarantined.

For instance,

```
> serve list
JOE@EXAMPLE.COM          DROP 2003-08-20 15:51:20 by nathan@EXAMPLE.COM
FOOBAR@EXAMPLE.EDU       HARD 2003-04-07 14:55:29 by NATHAN@EXAMPLE.COM
BLAB@FOO.EXAMPLE.COM     SOFT 2004-09-14 10:53:18
SPAMMER@SPAMDOMAIN.COM   SPAM 2003-08-20 15:50:55

4 matching entries.
```

SHUTDOWN [REBOOT|REIPL]

Stop the server, and (optionally) restart it immediately (by specifying either *REBOOT* or *REIPL* -- the two options are synonymous). This is available on all platforms.

STOP

Same as *SHUTDOWN*.

Commands for LISTSERV on VM Servers Only**CMS *command_text***

Issue a CMS command and get the last 20 lines of response sent back to you, the rest being available from the console log.

CP *command_text*

Issue a CP command and get up to 8k of response data sent to you (the rest is lost).

DATABASE *function*

Control operation of databases:

- *DISABLE* – Disable interactive database access, without shutting down existing sessions
- *ENABLE* – Re-enable interactive access
- *SHUTDOWN* – Shut down all interactive database sessions, and disable interactive access

INSTALL *function*

Software update procedure (LISTSERV-NJE only):

- *CLEANUP shipment* – Remove an installed shipment from the log
- *CLEANUP BEFORE dd mmm yy* – Remove all shipments installed before that date

- `PASSWORD shipment PW=instpw` – Confirm installation of a shipment, when requested by LISTSERV
- `RELOAD shipment` – Attempt to reload a shipment which failed due to a disk full condition
- `STATUS` – Get a list of installed "shipments"

OFFLINE

Suspend processing of reader files and disable the GET command

ONLINE

Cancel OFFLINE condition

`PUTC fn ft <fm|cuu|dirid> <RECFM=F LRECL=nnn>`

Update a CMS file on one of LISTSERV's R/W mini-disks; note that this is similar to SENDFILE + RECEIVE or LINK + COPYFILE and should NOT be used to update file-server files

`SENDFile fn ft <fm|cuu|dirid>`

Request the server to send you a file from one of its disks

SF

Same as SENDFILE

`SHOW <function>`

In addition to the standard SHOW functions available on other servers, VM servers support the following functions:

- `BENCHmarks` – CPU/disk/paging benchmarks.
- `EXECLoad` – Statistics about EXECLOADED REXX files.
- `LSVFILER` – Statistics about LSVFILER file cache.
- `PREXX` – Statistics about PREXX functions usage.
- `STORage` – Information about available disk space and virtual storage.



Note: Some debugging commands and options have been omitted.

Sending commands to LISTSERV

You will see numerous references to "sending commands to LISTSERV" in this and other L-Soft manuals. All LISTSERV commands are sent to the server either by email or via the Web Interface described in Section 6.5 [Submitting LISTSERV Commands](#). For mailed commands, this means that you must create a new mail message using whatever command this requires for your mail client (click on "New message" or its equivalent for most mail clients) addressed to the LISTSERV address. Let's say for the sake of argument that the list you are managing is running on a server called `LISTSERV.MYCORP.COM`. In order to send a command to that server, you would create a new message and address it to `LISTSERV@LISTSERV.MYCORP.COM`, and place the command(s) in the body (not the subject) of the message.

Depending on how you have security set up for your lists, some or all commands may require that you validate them with a personal LISTSERV password.

Defining Personal Passwords

The passwords recognized by LISERV for various operations (assuming that the NOPW parameter is not used with the "Validate=" keyword) are of two distinct types:

- **Personal Passwords** – LISERV can store a personal password in its signup files corresponding to your userid. This password not only can be used for list maintenance operations, but also protects your FUI (file update information) and AFD (automatic file distribution) subscriptions (if available on your server) and must be used to store your archive files, if any, on the server.
- **List Passwords** – List passwords are technically obsolete except in one particular case (peered lists, which require each peer to have the same password). We mention them here only because users upgrading from earlier versions will be aware of their existence. You should define and use a personal password for all protected operations.

To add a personal password, send mail to LISERV with the command

`PW ADD newpassword`

in the body of the message. LISERV will request a confirmation via the "OK" mechanism (see above) before it adds the password.

If you want to remove your password altogether, send the command

`PW RESET`

This command will also require confirmation.

And finally, if you simply want to change your personal password, send the command

`PW CHANGE newpassword [PW=oldpassword]`

If you do not include the old password in the command (e.g., you've forgotten it), LISERV will request an "OK" confirmation. Otherwise, it will act on the command without need for further confirmation (unless, of course, the *oldpassword* provided is incorrect).

Personal passwords may also be defined via the web administration interface at login time.

Appendix B: Sample Boilerplate Files

So-called "boilerplate" files are handy for list owners who find themselves answering the same questions over and over again. Usually these questions refer to basic LISTSERV usage. You can save yourself a lot of time by keeping files on-line such as the ones below to cut and paste into replies. Feel free to edit these to suit your own tastes (or compose your own!).

(Be sure to insert the appropriate list names and LISTSERV hosts as required.)

Subscription Requests Sent to the List

LISTSERV subscription requests need to be sent to the LISTSERV address rather than to the list itself. You do this by sending mail to `LISTSERV@host` with the command

```
SUB listname Your Name
```

as the body of the message. If you are unfamiliar with LISTSERV and its associated commands, I suggest that you add the commands

```
INFO GENINTRO  
INFO REFCARD
```

as additional lines of your message. LISTSERV will then send you a file containing a General Introduction to Revised LISTSERV that will give you some instruction on the service and a Quick Reference Card of the various commands.

Thanks for your interest. If you have trouble subscribing with this method, please let me know and I will attempt to help.

If you have `Subscription= Open,Confirm` you might want to add the following:

Because LISTSERV verifies mailing paths for new subscribers (a process not implemented when the list administrator adds people manually), it is preferred that users subscribe themselves by the method outlined above.

Sending Other Commands to the List or to the *-REQUEST Address for the List

On Sun, 20 Mar 1994 22:44:25 -0800 (PST) you said:

```
>"INFO REFCARD"
```

*You need to redirect LISTSERV commands like the above (minus the double quotes by the way), to <listserv@host>. The *-request type addresses are for reaching the person that run the list.*

[another version:]

You've sent mail that appears intended for a mailing list to one of the addresses used to reach the list owner. That is, rather than sending your mail to `listname@host` you've sent the note to `OWNER-listname@host` or `listname-REQUEST@host`. Please re-send the appended note to the list address if you haven't done so already.

----- original message follows:

Unsubscribed User Still Getting Mail

Use this one after you have done an exhaustive search of the list and determined that the person simply isn't on the list. Typically the user is subscribed to a redistribution list and doesn't realize it.

Unfortunately I can't unsubscribe you from listname because you aren't subscribed to listname@host. I have run a check to see if you might be subscribed under a slightly different network address and have not found anything.

There are a few possibilities you should look into. Are you getting a digest? Are you perhaps getting a redistributed copy of postings, possibly from a redistribution list? If you look at the mail headers, and there is an indication that you may be getting the postings from another source, you will have to ask the people that run the other source to remove you from their list.

Use this one if the user unsubscribed successfully, BUT they are still getting list mail.

I've done a search of listname for a possible duplicate subscription for you and have not found anything. It's possible that the mail you are receiving was actually sent from listname before your unsubscribe request was processed. Depending on the routing, it could take anywhere from 24 to 48 hours for all such messages to get through the network, so please be patient.

Quoted Replies Include Message Headers Causing them to Bounce

When quoted replies from a user's mail client includes message headers in the mail body, the reply will be bounced back to the list owner.

If you forward such messages to the list, or back to the sender, you can add the following at the beginning. I ran across this one in the CBAY-L mailing list archives, and edited it slightly.

This message was sent to me from LISTSERV instead of the list. The original message included the entire message being replied to, including the mail headers. These headers in the body pointed to the list itself. LISTSERV has mail-loop avoidance code and when it sees headers that it thinks it generated itself, it bounces the message to the list owner. If your mail client does this, please remember to delete such "included header lines" from the body of your list replies.

-----original message-----

D.6.

Add the following to ask a postmaster for help on a bounced address you've set to NOMAIL; don't forget to include a cc: to the bounced address.

Postmaster(s),

Can you shed any light on the following error message? Please let me know what you find as I have removed the e-mail address from the mailing list in question and would like to restore service as soon as is feasible.

Thanks.

Aside to user: Should this note reach you (meaning that the mail delivery problems have been resolved), you can re-enable your mail service by sending mail to listserv@host with the following command: SET listname MAIL

Delivery Error with Unknown User Account

If you get a delivery error that doesn't specify which user account is causing the bounce, then use the following:

Postmaster,

I received the appended mail delivery report from your system and need help isolating the e-mail address that is causing the error. That is, there are multiple recipients from your system on the list but the delivery error doesn't explicitly mention any of the users on the list. I'm including a list of subscribers from your system. If any of them are no longer valid, or aren't usable address for some other reason, please let me know.

---- list of e-mail address on the indicated list follows:

Setting a User to DIGEST because of Bouncing Mail

If you've set a user to DIGEST because of bouncing mail, and the user is asking why he/she is now getting the digest, then use the following:

I received a mail delivery error for your address and issued a

SET listname DIGEST

on your behalf to minimize the number of bounce messages. I also sent a copy of the error I received to your postmaster (or the postmaster of the mail gateway that generated the error), asking for help. And since such delivery problems are often transient, I CC'd a copy of that note to your address, and included instructions for turning your mail back on. Apparently I didn't hear anything from your postmaster, or he/she said not to turn your mail back on until the problem was resolved. If they had responded and said the problem was resolved, I would have set you back to MAIL.

The other possibility is that I received a mail message indicating that there was some temporary problem with your account. In that case, for example if you had exceeded your disk quota and couldn't receive any new mail, I would not have bothered your postmaster. I have a different form letter that I send when that happens. Again it explains what has occurred and includes instructions for re-enabling your mailing list subscription. But I only send that one to the address the list member. Either way, whatever was wrong has been corrected, and you'd probably like to start receiving mail again. So, here's how you can restore your mail service. If you have any problems doing so, please let me know and I'll help. But since I don't know which of the three mail service options you had chosen before, I can't do it for you without guessing. You can re-enable your mail service by sending mail to listserv@host with one of the following commands

SET listname MAIL

SET listname DIGEST (if you want digest-format mail)

SET listname INDEX (if you want digest-index-format mail)

*in the *body* of the mail message. Please note that these settings are mutually exclusive, you can't choose more than one.*

A Sample "Your List has been Created" Boilerplate

Mailing List Setup Confirmation

I have created the XXXXX-L list on LISTSERV.MYHOST.COM per your setup sheet.

If you are new to LISTSERV, you will probably want to download L-Soft's Quick Start manual for list owners. Simply point your web browser to

<http://www.lsoft.com/manuals/QS-index.html>

and view online or choose the version appropriate for your word processor or viewer.

Formal documentation of list owner commands and other list ownership issues can be found in the List Owner's Manual, which is available at the URL

<http://www.lsoft.com/manuals/ownerindex.html>

Per your list service agreement, support for your list is handled through a mailing list, LIST-SUPPORT@LISTSERV.MYHOST.COM. You have been added to that list. Please direct all support questions to the LIST-SUPPORT list.

You may also be interested in subscribing to the LSTOWN-L mailing list for LISTSERV list owners. To do so, send a mail message to LISTSERV@LISTSERV.NET with the command

SUB LSTOWN-L Your Name

in the body (not the subject) of the message. There are a number of extremely experienced LISTSERV list owners subscribed there who are more than willing to share their expertise. Don't hesitate to ask for help.

You now need to instruct LISTSERV to add personal passwords for the list owner account(s). These passwords are used to validate privileged commands (such as the PUT command for storing your list "header" on the server after making changes to it). This is done by sending mail from each account to LISTSERV@LISTSERV.MYHOST.COM with the command

PW ADD password

(again, "password" is whatever you want it to be) in the body of the message. LISTSERV will request confirmation of this operation; simply reply to the confirmation request with the word "ok".

Adding these passwords will considerably lessen the chance that someone will "spoof" mail from you to make changes on your list. It is very unlikely that this will happen, but it never hurts to be cautious.

Sincerely,

Joe Smith

LISTSERV Maintainer

Appendix C: Country Codes

This list is sorted by country code, not by country. Remember that country codes do not always correspond to a person's physical location; for instance, a German CompuServe user would be counted as a USA user simply because his or her address ends with .COM. Similarly, there are several Canadian universities with .EDU suffixes; and so forth.

Table 23-1 Country Codes

COM	USA (company)	GH	Ghana	NU	Niue
EDU	USA (education)	GI	Gibraltar	NZ	New Zealand (Aotearoa)
GOV	USA (government)	GL	Greenland	OM	Oman
MIL	USA (military)	GM	Gambia	PA	Panama
NET	USA (network)	GN	Guinea	PE	Peru
ORG	USA (organization)	GP	Guadeloupe	PF	French Polynesia
AC	Ascension Island	GQ	Equatorial Guinea	PG	Papua New Guinea
AD	Andorra	GR	Greece	PH	Philippines
AE	United Arab Emirates	GS	South Georgia and South Sandwich Islands	PK	Pakistan
AF	Afghanistan	GT	Guatemala	PL	Poland
AG	Antigua and Barbuda	GU	Guam	PM	St. Pierre and Miquelon
AI	Anguilla	GW	Guinea-Bissau	PN	Pitcairn
AL	Albania	GY	Guyana	PR	Puerto Rico
AM	Armenia	HK	Hong Kong	PS	Palestinian Territory, Occupied
AN	Netherlands Antilles	HM	Heard and McDonalds Islands	PT	Portugal
AO	Angola	HN	Honduras	PW	Palau
AQ	Antarctica	HR	Croatia (Hrvatska)	PY	Paraguay
AR	Argentinian	HT	Haiti	QA	Qatar
AS	American Samoa	HU	Hungary		
AT	Austria	ID	Indonesia		
AU	Australia				

AW	Aruba	IE	Ireland	RE	Reunion
AX	Aland Islands	IL	Israel	RO	Romania
AZ	Azerbaijan	IM	Isle of Man	RS	Serbia
		IN	India	RU	Russian Federation
BA	Bosnia and Herzegovina	IO	British Indian Ocean Territory	RW	Rwanda
BB	Barbados	IQ	Iraq		
BD	Bangladesh	IR	Iran	SA	Saudi Arabia
BE	Belgium	IS	Iceland	SB	Solomon Islands
BF	Burkina Faso	IT	Italy	SC	Seychelles
BG	Bulgaria			SD	Sudan
BH	Bahrain	JE	Jersey	SE	Sweden
BI	Burundi	JM	Jamaica	SG	Singapore
BJ	Benin	JO	Jordan	SH	St. Helena
BM	Bermuda	JP	Japan	SI	Slovenia
BN	Brunei Darussalam			SJ	Svalbard and Jan Mayen Islands
		KE	Kenya	SK	Slovak Republic
BO	Bolivia	KG	Kyrgyzstan	SL	Sierra Leon
BR	Brazil	KH	Cambodia	SM	San Marino
BS	Bahamas	KI	Kiribati	SN	Senegal
BT	Bhutan	KM	Comoros	SO	Somalia
BV	Bouvet Island	KN	Saint Kitts and Nevis	SR	Suriname
BW	Botswana	KP	Korea (North)	ST	Sao Tome and Principe
BY	Belarus	KR	Korea (South)	SU	USSR (former)
BZ	Belize	KW	Kuwait	SV	El Salvador
		KY	Cayman Islands	SY	Syria
CA	Canada	KZ	Kazakhstan	SZ	Swaziland
CC	Cocos (Kneeling) Islands				
CD	Congo, Democratic Republic				
CF	Central African Republic	LA	Laos	TC	Turks and Caicos Islands
CG	Congo	LB	Lebanon	TD	Chad

CH	Switzerland	LC	Saint Lucia	TF	French Southern Territories
CI	Cote D'Ivory (Ivory Coast)	LI	Liechtenstein	TG	Togo
CK	Cook Islands	LK	Sri Lanka	TH	Thailand
CL	Chile	LR	Liberia	TJ	Tajikistan
CM	Cameroon	LS	Lesotho	TK	Tokelau
CN	China	LT	Lithuania	TM	Turkmenistan
CO	Columbia	LU	Luxembourg	TN	Tunisia
CR	Costa Rica	LV	Latvia	TO	Tonga
CS	Czechoslovakia (former)	LY	Libya	TP	East Timor
CU	Cuba			TR	Turkey
CV	Cape Verde	MA	Morocco	TT	Trinidad and Tobago
CX	Christmas Island	MC	Monaco	TV	Tuvalu
CY	Cyprus	MD	Moldova	TW	Taiwan
CZ	Czech Republic	MG	Madagascar	TZ	Tanzania
		MH	Marhsall Islands		
DE	Germany	MK	F.Y.R.O.M. (Macedonia)	UA	Ukraine
DJ	Djibouti	ML	Mali	UG	Uganda
DK	Denmark	MM	Myanmar	UK	United Kingdom
DM	Dominica	MN	Mongolia	UM	US Minor Outlying Islands
DO	Dominican Republic	MO	Macau	US	United States
DZ	Algeria	MP	Northern Mariana Islands	UY	Uruguay
		MQ	Martinique	UZ	Uzbekistan
EC	Ecuador	MR	Mauritania		
EE	Estonia	MS	Montserrat	VA	Vatican City State (Holy See)
EG	Egypt	MT	Malta	VC	Saint Vincent and the Grenadines
EH	Western Sahara	MU	Mauritius	VE	Venezuela
ER	Eritrea	MV	Maldives	VG	British Virgin Islands
ES	Spain	MW	Malawi	VI	Virgin Islands (U.S.)
ET	Ethiopia	MX	Mexico	VN	Viet Nam
		MY	Malaysia	VU	Vanuatu

FI	Finland	MZ	Mozambique		
FJ	Fiji			WF	Wallis and Futuna Islands
FK	Falkland Islands (Malvinas)	NA	Namibia	WS	Samoa
FM	Micronesia	NC	New Caledonia		
FO	Faroe Islands	NE	Niger	YE	Yemen
FR	France	NF	Norfolk Island	YT	Mayotte
FX	France, Metropolitan	NG	Nigeria	YU	Yugoslavia (former)
		NI	Nicaragua		
GA	Gabon	NL	Netherlands	ZA	South Africa
GB	Great Britian (UK)	NO	Norway	ZM	Zambia
GD	Grenada	NP	Nepal	ZR	Zaire (see CD)
GE	Georgia	NR	Nauru	ZW	Zimbabwe
GF	French Guiana	NT	Neutral Zone		

Index

A

- Address Probing [180](#)
 - active [180](#)
 - passive [181](#)
- Addressing Schemes
 - X.400 [166](#)
 - X.500 [166](#)
- Administrator Roles
 - editor [60](#)
 - moderator [60](#)
 - owner [60](#)
- Advertising
 - breaching "netiquette" [252](#)
 - unsolicited [205](#)
 - your mailing lists [247](#)
- Alerts
 - license expiration alert preference [26](#)
 - support expiration alert preference [26](#)
- Alphabetical Keyword Listing Wizard [59](#)
- Announcement-Only Lists [13](#)
 - creating [151](#)
- Archive Index [103](#)
- Archive Notebooks
 - deleting existing [207](#)
 - indexing [207](#)
 - managing [207](#)
 - setting up [207](#)
- Archive Preferences [27](#)
 - default view [28](#)
 - email headers [28](#)
 - enabling the Latest Messages on List Home Page option [28](#)
 - font [28](#)
 - hovering descriptions [28](#)
 - including the author's name and address [28](#)
 - including the table of contents [28](#)
 - message numbers [29](#)
 - primary sorting [28](#)
 - secondary sorting [28](#)
 - setting the Number of Messages on List Home Page [28](#)
- Archives
 - posting a message to the list [102](#), [103](#), [105](#)
 - subscribing to a list [102](#), [103](#), [105](#), [106](#)
 - unsubscribing from a list [102](#), [103](#), [105](#), [106](#)
 - updating a list [102](#), [103](#), [105](#), [106](#)

- UTF-8 searches [108](#)
- viewing an archived message [104](#)
- viewing in the Web Interface [99](#)
- viewing list archives [101](#)
- viewing the Archive Index [103](#)
- viewing the Server Archive Index [99](#)

Attachments

- content filtering [67](#)
- defining types for distribution on your list [65](#)
- Exchange and Outlook Attachments [67](#)
- HTML messages [66](#)
- limits [67](#)
- MIME Attachments [66](#)
- UUEncoded Attachments [66](#)

Automatic File Distribution (AFD) [216](#)

Auto-Responder Lists

- creating [150](#)

B

Banners [77](#)

- HTML [77](#)
- text [77](#)

Boilerplate Files

- samples [277](#)

Bounced Mail [171](#)

- forwarding [176](#)
- LMail error codes [177](#)
- redistribution [176](#)

Bulk Operations

- using the Web Interface [83](#)

C

CataList

- accessing [32](#), [100](#)
- adding an HTML description to your list [76](#)
- adding HTML to a list header [247](#)
- defining categories using the list header keyword [52](#)
- defining search categories in a list header [249](#)
- hiding your list header [76](#)
- reasons for not finding a list [75](#)
- setting options for cataloguing your list [75](#)
- using to find a list [33](#)

Categories

- defining using the list header keyword [52](#)

ChangeLog

- generating a report for lists [91](#)

- Character Set
 - default for mail templates [94](#), [223](#)
 - forcing the browser to use a specific character set [96](#)
 - setting for a message [113](#)
 - Cloning
 - lists [158](#)
 - Command Confirmation Request [140](#)
 - cancelling the OK cookies [142](#)
 - Command Interface
 - using to submit LISTSERV commands [85](#)
 - Commands [253](#)
 - ACK [187](#)
 - advanced commands [264](#)
 - cancelling the OK cookies [142](#)
 - CONCEAL [188](#)
 - confirmation request [140](#)
 - defining passwords [276](#)
 - DIGest [186](#)
 - EDITOR [189](#)
 - file management [267](#)
 - File Server [262](#)
 - for templates [226](#)
 - implementing the INFO command [251](#)
 - INDEX [187](#)
 - informational [260](#)
 - List Management [268](#)
 - list related [257](#)
 - List Subscription [253](#)
 - LISTSERV Maintainer [272](#)
 - Mail [186](#)
 - MIME [186](#)
 - MSGask [187](#)
 - NOACK [187](#)
 - NOCONCEAL [188](#)
 - NODIGest [186](#)
 - NOEDITOR [189](#)
 - NOINDEX [187](#)
 - NOMail [186](#)
 - NOMIME [186](#)
 - NOPOST [189](#)
 - NORENEW [190](#)
 - NOREPro [188](#)
 - NOREVIEW [190](#)
 - OK [10](#)
 - POST [189](#)
 - RENEW [190](#)
 - REPro [188](#)
 - REVIEW [190](#)
 - sending to LISTSERV [130](#), [275](#)
 - syntax conventions [xiv](#)
 - using the LISTSERV Command Interface [85](#)
 - using the QUIET option [171](#)
 - using the SERVE command when a user is "served out" [183](#)
 - Web Functions [262](#)
 - Considerations
 - auto-delete for holidays [179](#)
 - Content Filtering [160](#)
 - Content Templates
 - accessing from the HTML editor [116](#)
 - using in a message [116](#)
 - Continuation Card Syntax [167](#)
 - Converting
 - existing lists from other systems [168](#)
 - message archives [168](#)
 - Cookies
 - cancelling the OK confirmation cookies [142](#)
 - login cookie expiration preference [26](#)
 - Country Codes [281](#)
- D**
- Dashboards
 - for List Management [35](#)
 - DBMS
 - list header keyword [53](#)
 - Default Home Page [21](#)
 - Delivery Errors
 - handling [179](#)
 - Digests [70](#)
 - displaying UTF-8 encoded messages [72](#)
 - list header keyword [42](#)
 - with HTML [72](#)
 - with MIME [73](#)
 - with plain text [71](#)
 - Discussion Lists [15](#)
 - DomainKeys [162](#)
 - Dynamic Templates [239](#)
- E**
- Edited Lists
 - creating [145](#)
 - creating Private Edited Lists [149](#)
 - setting up [197](#)
 - submitting subscriber contributions [199](#)
 - Editing
 - a web page [31](#)
 - Editor-Header [47](#)
 - Email
 - using to communicate with LISTSERV [7](#)
- F**
- FAREWELL Files [202](#)
 - creating [202](#)
 - storing [202](#)
 - using an alternative [204](#)

- using as a feedback tool 204
- File Access Codes (FAC) 211
- File Archives 207
 - PC Systems 209
 - VM Systems 209
 - workstations 209
- File Packages 216
- File Update Information (FUI) 216
- Filelist
 - adding file descriptors 210
 - deleting file descriptors 211
 - File Access Codes for user access 211
 - maintenance for VM systems 209
 - retrieving 209
 - storing 212
- Files
 - deleting from the host machine 215
 - storing on the host machine 214
- Firewalls 245
- G**
- General Preferences 25
 - license expiration alert 26
 - login cookie expiration 26
 - mode 25
 - basic 25
 - expert 26
 - tutorial 25
 - navigation style 26
 - non-script 26
 - pulldown 26
 - owner dashboard changelogs 26
 - page caching 27
 - page title icons 26
 - setting the default home page 26
 - support expiration alert 26
 - text size 26
- Global List Exchange (GLX) 251
- H**
- Host Machine
 - deleting files from 215
 - storing a list on 135
 - storing files on 214
 - storing the listname.MAILTPL file 237
- HTML
 - creating mail templates 96
 - creating newsletters 116
 - customizing mail templates 96
 - customizing newsletters 116
- HTML Editor 116
- HTML Mail Template Gallery 96
- I**
- Indexing 71
- INFO Command
 - implementing 251
- J**
- K**
- L**
- List Activity Reports
 - generating 91
 - generating a changelog report 91
- List Archives 207
 - viewing in the Web Interface 101
- List Configuration 37
 - controlling access 39
 - controlling maintenance 47
 - controlling security 50
 - controlling subscriptions to the list 51
 - describing the list 38
 - determining distribution 42
 - handling errors 45
 - manually using the Web Interface 54
 - retrieving 131
 - by email 131
 - using the Alphabetical Keyword Listing wizard 59
 - using the Wizard 37
- List Configuration Task Wizard 59
 - administrators 60
 - attachments 65
 - banners 77
 - CataList 75
 - digesting 70
 - indexing 70
 - mail templates 77
 - probes 69
 - security 61
 - subject tags 73
- List Editor
 - using the list header keyword to define 47
- List Header
 - adding HTML for the CataList 247
 - defining search categories for the CataList 249
 - editing using email 133
 - hiding from CataList 76
 - hiding parts of 56
- List Header Keywords 129
 - ACK 42
 - Attachments 40
 - Auto-Delete 45
 - Categories 52

- Change-Log 50
- command confirmation 140
- Confidential 50
- Confirm-Delay 51
- Daily-Threshold 42
- DBMS 53
- Default-Options 52
- Default-Options= keyword 190
- Default-Topics 52
- Digest 42
- Editor 47
- Errors-To 45
- Exit 50
- Files 41
- Filter 41
- Indent 53
- Internet-Via 43
- Language 53
- Limits 53
- List-ID 47
- Local 50
- Long-Lines 53
- Mail-Merge 53
- Mail-Via 43
- Misc-Options 53
 - IGNORE_EMAIL_CASE 53
 - NO_SPAM_CHECK 53
 - RESPECT_EMAIL_CASE 53
 - SUPPRESS_APPROVED_BY 53
- Misc-Options= keyword 191
- Moderator 48
- New-List 48
- Newsgroups 43
- Notebook 48, 63
- NOTEBOOK= 138
- Notebook-Header 48
- Notify 48
- Owner 48
- Peers 48
- Prime 43
- PW 50
- Renewal 49
- Reply-To 43
- Review 41, 63
- REVIEW= 138
- Safe 46
- Send 41, 64
- SEND= 139
- Sender 44
- Service 51
- SERVICE= 137
- Sizelim 49
- Stats 41
- Subject-Tag 49
- Sub-Lists 44, 57
- Subscription 52
- SUBSCRIPTION= 137
- Topics 44, 56
- Translate 54
- Validate 51, 61
- VALIDATE= 136
- X-Tags 49
- List Management
 - list configuration
 - manually using the Web Interface 54
 - using the Alphabetical Keyword Listing wizard 59
 - using the List Configuration Wizard 37
 - list configuration task wizard 59
 - list configuration using the Web Interface 37
 - using the Dashboard 35
 - using the Web Interface 35
- List Management Reports
 - see Reports 89
- List Moderation
 - accepting the message 87
 - changing subscriber settings using the Web Interface 88
 - reject and serve off message 87
 - rejecting the message 87
 - using the Web Interface 87
 - viewing the message in the Web Interface 88
- List Moderator
 - definition 6
 - using the list header keyword to define 48
- List of Lists 247
- List Owner
 - as Editor 196
 - as Moderator 195
 - defining by email 134
 - definition 6
 - using the list header keyword to define 48
- List Reports
 - generating 89
- List Security 61
 - controlling subscription options 62
 - controlling who can access the archives 63
 - controlling who can post to the list 64
 - defining service areas 62
 - levels of validation 61
 - restricting subscriber privileges 65
- List Subscribers
 - definition 6

- List Subscriptions
 - set whether or not users can subscribe 52
 - setting the renewal times using the list header keyword 49
- List Topics 56
 - using 200
 - using the list header keyword to define 44
 - using the list header keyword to define the default 52
- List-Address 47
- Lists
 - about 3
 - adding a new subscriber using the Web Interface 78
 - adding an HTML description for CataList 76
 - adding subscribers 165
 - adding subscribers in bulk 170
 - anatomy of 9
 - avoiding duplication 125
 - cloning 158
 - controlling subscription requests 137
 - controlling the service area 137
 - controlling who can post to a list 139
 - controlling who reviews the list of subscribers 138
 - converting existing lists from other systems 168
 - creating Announcement-Only Lists 151
 - creating Auto-Responder Lists 150
 - creating Edited Lists 145
 - creating manually using email 126
 - creating Moderated Lists 147, 149
 - creating Peered Lists 153
 - creating Private Discussion Lists 145
 - creating Private Edited lists 149
 - creating Public Discussion Lists 144
 - creating Restricted Subscription Lists with auto-generated questionnaire 152
 - creating Self-Moderated Lists 149
 - creating Semi-Moderated Lists 149
 - creating split lists 153
 - creating Sub-Lists 156
 - creating Super Lists 156
 - creating using email 125
 - deleting a subscription using the Web Interface 79
 - deleting subscribers 165
 - deleting subscribers in bulk 170
 - editing 195
 - examining a subscription using the Web Interface 79
 - finding users that do not appear in the list 167
 - getting started 1
 - hiding the list header from CataList 76
 - linking to another list 248
 - linking two or more 153
 - message approval with Send=Editor, Hold 199
 - moderating 195
 - moving users from one (peer) server to another 154
 - naming conventions 127
 - "L" convention 127
 - maximum length of name 128
 - reserved characters 127
 - reserved names 127
 - user-friendly 129
 - normal lists vs. super lists 57
 - Peered Lists
 - special commands 155
 - reviewing subscribers 83
 - searching for a list using CataList 33
 - serving up custom web pages 241
 - storing on a host machine 135
 - types 13
 - moderated 17
 - one-way announcement 13
 - two-way discussion 15
 - X.400 addressing schemes 166
 - X.500 addressing schemes 166
- LISTSERV
 - about 4
 - communicating with 7
 - using email to communicate with 7
 - using the Web Interface 7
- LISTSERV Maintainer
 - definition 5
- LMail Error Codes 177
- Logging In 21
- Loopcheck
 - keyword 46
- Loop-Checking
 - problems associated with 243
- M**
 - Mail Templates 221
 - default charset 94, 223
 - Mail-Merge 58
 - list header keyword 53
 - Message Approval 199
 - Message Archives 168
 - Message Fragments 222
 - Message Templates 221

Messages

- adding an attachment 113
- adding as RSS Abstract 113
- composing
 - HTML 112, 116
 - plain text 112
 - using the Web Interface 111
- list moderation
 - accepting 87
 - reject and serve off 87
 - rejecting 87
- posting to a list
 - using email 219
 - using the Web Interface 102, 103, 105, 111
- quoting original 115
- saving as a draft 115
- scheduling delivery 112
- sending a test copy to yourself 115
- setting charset 113
- spam checking 114
- uploading HTML 112
- using Content Templates 116
- using the HTML editor 116

MIME Attachments

- allowing 160
- blocking 160

Moderated Lists 17

- creating 147
- creating Private Moderated Lists 149

N**Newsletter Profiles 29**

- contact email 30
- delivery test addresses 31
- footer 31
- HTML editing 31
- legal 31
- logo 31
- newsletter name 29
- phone number 30
- physical address 30
- slogan 30
- website URL 30

Newsletter Templates

- accessing 102, 103, 105
- creating 116
- customizing 116
- selecting 116
- setting the newsletter profile 117

Newsletters

- creating customized HTML versions 116

NJE-Via 43**Non-Subscribers**

- definition 6

Notebook Archives

- controlling access 138

O**OK Confirmation Method**

- using 10

One-Way Announcement Lists 13**P****Page Caching**

- preferences 27

Passwords

- changing for the Web Interface 23
- defining using commands 276
- defining using email 130
- setting for the Web Interface 23

Peered Lists

- creating 153
- linking two or more 153
- moving users from one server to another 154
- special commands 155

Pending Messages

- viewing 112

Placeholders

- using in a newsletter 117

Preferences

- setting 24
 - archive 27
 - general 25
 - newsletter profiles 29
 - report 29

Prime Time

- setting to delivery during 43

Private Discussion Lists

- creating 145

Private Edited Lists

- creating 149

Probing 69

- active 69
- passive 70

Public Discussion Lists

- creating 144

Q**QUIET Option**

- using with commands 171

R**Report Preferences 29**

- changelog period 29
- number of lists displayed 29

- number of subscribers displayed [29](#)
- report period [29](#)
- Reports
 - ChangeLog
 - for lists [91](#)
 - List Management
 - accessing [89](#)
 - List Activity Reports [91](#)
 - List Reports [89](#)
 - Subscriber Reports [89](#)
- Restricted Subscriptions Lists
 - creating with auto-generated questionnaire [152](#)
- RSS Abstract
 - adding to a message [113](#)
- S**
- Search Categories
 - defining in a list header for the Catalist [249](#)
 - settings [250](#)
- Security
 - cancelling the OK cookies [142](#)
 - command confirmation [140](#)
 - controlling access to the notebook files [138](#)
 - controlling subscription requests [137](#)
 - controlling the service area of a list [137](#)
 - controlling who can post to a list [139](#)
 - controlling who reviews the list of subscribers [138](#)
 - options [136](#)
 - restricting subscriber privileges [143](#)
 - restricting the number of postings per day [143](#)
 - validation level [136](#)
- Self-Moderated Lists
 - creating [149](#)
- Semi-Moderated Lists
 - creating [149](#)
- Send Message Screen
 - using [111](#)
 - using the HTML editor [116](#)
- Server Archive Index [99](#)
- Social Conventions
 - accepting cultural and linguistic differences [204](#)
 - inappropriate language [205](#)
 - netiquette [204](#)
 - privacy [205](#)
 - recognizing cultural and linguistic differences [204](#)
- Spam
 - checking messages [114](#)
- Spamming [205](#)
- Split Lists
 - creating [153](#)
 - linking two or more [153](#)
 - moving users from one server to another [154](#)
- Sub-Catalog
 - indexing [214](#)
 - updating [213](#)
- Subject Tags [73](#)
- Sub-Lists
 - creating [156](#)
- Subscriber Management
 - adding a new subscriber using the Web Interface [78](#)
 - bulk operations using the Web Interface [83](#)
 - deleting a subscription using the Web Interface [79](#)
 - examining a subscription using the Web Interface [79](#)
 - using the Web Interface [35, 78](#)
- Subscriber Reports
 - generating [89](#)
- Subscribers
 - adding to a list [165](#)
 - adding to a list in bulk [170](#)
 - deleting from a list [165](#)
 - deleting from a list in bulk [170](#)
 - restricting privileges [143](#)
 - restricting the number of postings per day [143](#)
 - reviewing [83](#)
 - setting personal subscription options [185](#)
 - setting subscription options [185](#)
- Subscription Options
 - ACK [187](#)
 - CONCEAL [188](#)
 - Default-Options= keyword [190](#)
 - DIGest [186](#)
 - EDITOR [189](#)
 - for mail headers of incoming postings [187](#)
 - INDEX [187](#)
 - Mail [186](#)
 - MIME [186](#)
 - Misc-Options= keyword [191](#)
 - MSGack [187](#)
 - NOACK [187](#)
 - NOCONCEAL [188](#)
 - NODIGest [186](#)
 - NOEDITOR [189](#)
 - NOINDEX [187](#)

- NOMail [186](#)
 - NOMIME [186](#)
 - NOPOST [189](#)
 - NORENEW [190](#)
 - NOREPro [188](#)
 - NOREVIEW [190](#)
 - POST [189](#)
 - QUERY [185](#)
 - RENEW [190](#)
 - REPro [188](#)
 - REVIEW [190](#)
 - setting [185](#)
 - setting personal options [185](#)
 - SUBJecthdr [188](#)
 - Subscriptions
 - confirmations [181](#)
 - managing [165](#)
 - renewing [182](#)
 - Super Lists [57](#)
 - creating [156](#)
- T**
- Templates
 - accessing [223](#)
 - calculating the value for DAYSEQ [236](#)
 - commands [226](#)
 - common variable substitutions [224](#)
 - condition processing [229](#)
 - conditional expression evaluator [97](#), [242](#)
 - creating [221](#)
 - creating new HTML mail templates [96](#)
 - creating newsletter templates [116](#)
 - customizing newsletter templates [116](#)
 - customizing using the Web Interface
 - mail templates [93](#)
 - web templates [93](#)
 - DEFAULT MAILTPL [233](#)
 - DEFAULT.WWWPTL file [239](#)
 - DIGEST-H file [237](#)
 - dynamic [239](#)
 - editing [221](#)
 - editing list-level default templates [231](#)
 - for the WWW Interface [238](#)
 - forcing the browser to use a specific character set [96](#)
 - forms [238](#)
 - getting copies of the Default Template File [223](#)
 - INDEX-H file [237](#)
 - INFO template form [232](#)
 - mail template formats [224](#)
 - modifying default HTML mail templates [96](#)
 - naming conventions [223](#)
 - national language template files [239](#)
 - newsletter [102](#), [103](#), [105](#)
 - precedence [240](#)
 - rotating the bottom banner [235](#)
 - serving up custom web pages for your list [241](#)
 - SITE.WWWTPL file [239](#)
 - storing the listname.MAILTPL file on the Host Machine [237](#)
 - subscription renewal [235](#)
 - tips for using [234](#)
 - types
 - Mail [93](#), [221](#)
 - Message [93](#), [221](#)
 - Message Fragments [93](#), [222](#)
 - Web [93](#), [222](#)
 - using 8-bit characters [231](#)
 - using the DAYSEQ(n) function [235](#)

Toolbar [24](#)
 - Email Lists [24](#)
 - Help icon [24](#)
 - List Management [24](#)
 - List Moderation [24](#)
 - LISTSERV Maestro [24](#)
 - Logout [24](#)
 - Preferences [24](#)

Topics [188](#)
 - using the list header keyword to define [44](#)
 - using the list header keyword to define the default [52](#)

Two-Way Discussion Lists [15](#)

U

Unicode Support [xiii](#)
 - displaying UTF-8 encoded messages in the body of a digest [72](#)
 - UTF-8 archive searches [108](#)

UTF-8 Archive Searches [108](#)

V

W

Web Interface
 - changing passwords [23](#)
 - default home page [21](#)
 - intro [21](#)
 - list configuration [37](#)
 - list configuration task wizard [59](#)
 - List Management Dashboard [35](#)
 - logging in [21](#)
 - manual list configuration [54](#)
 - mode preference [25](#)
 - posting a message [111](#)
 - setting the password [23](#)
 - setting your personal preferences [24](#)

- Toolbar [24](#)
 - using to communicate with LISTSERV [7](#)
 - viewing the Archives [99](#)
- Web Pages
 - editing [31](#)
 - serving up custom pages for your list [241](#)
- Web Templates [222](#)
- WELCOME Files [202](#)
 - creating [202](#)
 - storing [202](#)
 - using an alternative [204](#)
 - using as a moderation tool [203](#)
- WWW Interface Templates [238](#)

X

Y

Z

