
Index

\$_ macro, 114
\${auth_authen} macro, 114, 170
\${auth_author} macro, 114, 170
\${auth_ssf} macro, 114, 170
\${auth_type} macro, 114, 170
\${cert_issuer} macro, 114, 165
\${cert_subject} macro, 114, 165
\${cipher_bits} macro, 114, 165
\${cipher} macro, 114, 165
\${client_resolve} macro, 161
\${daemon_name} macro, 114, 160
\${if_addr} macro, 114, 160
\${if_name} macro, 114, 160
\${i} macro, 170
\${mail_addr} macro, 114, 170
\${mail_host} macro, 114, 170
\${mail_mailer} macro, 114, 170
\${msg_id} macro, 193
\${nbadrpts} macro, 193
\${rcpt_addr} macro, 114, 176
\${rcpt_host} macro, 114, 176
\${rcpt_mailer} macro, 114, 176
\${_} macro, 160
\${tls_version} macro, 114, 165
2yz SMTP return code, 71, 72–73

3yz return code, 72
4yz SMTP code, 120, 122
5yz SMTP reply code, 71, 119, 122
220 SMTP success code, 12–13, 78
419 baiting, 7
419 fraud, 7
421 SMTP error code, 122
550 SMTP error, 78
554 SMTP error, 13

A

<a command and web references, 27
Abort item, 101
Abort phase, 96
Abort section, 87
Aborting envelopes, 197–200
Accept decision, 88
Accept reply, 89–90, 92–95
Advance Fee fraud, 7
Advisory-oriented handler functions, 155
AF_INET, 159
AF_INET6, 159

INDEX

- Aliases
 - hosts, 49–50
 - rebuilding database, 54, 62
 - sendmail, 53–54
 - Aliases file, 62, 65
 - Allocated memory, freeing up, 236
 - alt.test newsgroup, 74
 - amper program, 270
 - amper() subroutine, 270–272, 275
 - Apache HTTP server, 58, 65
 - Architecture, 231–232
 - Archiving spam, 244–246
 - argv argument, 166, 172
 - argv array, 167, 172
 - Arrays
 - CONF type, 247
 - of pointers to strings, 166
 - Articles
 - Date: header, 72
 - Lines: header, 73
 - mandatory headers, 72–73
 - Message-Id: header, 73
 - posting, 70–71
 - Asterisk (*) special character, 33
 - Atkinson Caller-ID standard, 6
 - Attachments
 - base64-decoding, 285–286
 - base64-encoded, 24–25
 - binary, 24
 - MIME (Multipurpose Internet Mail Extensions), 24–25
 - MIME headers, 285
 - quoted-printable encoded, 24–25
 - Authentication, 170
 - autoconf, 226, 290
 - autoheader, 226
 - automake, 226
- B**
- Baby-sitting script, 215–217
 - Background, running Milters in, 217–219
 - backlog argument, 127
 - Bait machine, 11
 - choosing platform, 44–47
 - compiler choice, 45–46
 - configuring sendmail, 50–54
 - database support, 46–47
 - excluding non-email ports, 56–58
 - forwarding copies of good email to, 64–65
 - installing Milters library, 46
 - network connections, 46
 - posix threads, 44–45
 - rebooting, 58–59
 - scanning, 56–57
 - sendmail version, 45
 - setting up
 - DNS records, 47–50
 - logging, 54–56
 - Bank card information theft, 7–8
 - <base command, 31
 - Base64 encoding
 - decoding, 258–265
 - marking end of data, 261
 - base64decode() function, 261, 264
 - Base64-decoding attachments, 285–286
 - Base64-encoded data, 258
 - Base64-encoding
 - attachments, 24–25
 - Subject: headers, 17–18
 - base64total() subroutine, 262
 - ba.test newsgroup, 74
 - Bayesian filters, xv–xvi, 288–293
 - Berkeley database, 232, 237
 - bg() function, 217–219
 - Binary attachments, 24
 - Bitwise OR (|), 104
 - BL (Blackhole List) sites, xv
 - Blacklisting, 232
 - Blocked senders, 232
 - Body, 92, 176–177, 190
 - deleting, 191
 - Milter replies per chunk, 94
 - modifying, 191

- replacing, 143–145, 191
 - routines, 255–293
 - Body item, 101
 - Body section, 87
 - Bounce address, 5
 - Bounce email, 245
 - Bounce information, multiline, 124
 - Bounce messages, 5, 167
 - Bounce reply envelope sender address, 15
 - Bouncing email, 16, 235
 - Boundary, 256–258
 - BSD method used to halt Milster, 215
 - bsearch() C library function, 281, 292
 - Buffers
 - containing replacement body, 144
 - for incoming message, 12–13
 - length of, 144
 - Bulletproof multithreaded code, 214
- C**
- C language compiler, keeping up-to-date, 45–46
 - C language programs main() function, 99–100
 - Cable accounts and nonfixed IP addresses, 4
 - Camouflaging HTML body, 18–22
 - Certificates, 165
 - cf variable, 181
 - cf/cf directory, 50
 - Chapters source code, 290
 - char *addr; regular expression, 243
 - Character-entity encoding, 20–21
 - decoding, 269–276
 - keywords, 20–21, 270
 - literal #, 20–21, 270
 - web references, 22
 - Characters
 - character-entity encoding, 269–276
 - converting to hexadecimal ASCII
 - equivalent, 277–279
 - quoted-printable encoding, 265–269
 - URL-encoding, 21
 - chdir() function, 208–209
 - check_mail rule set, 51
 - check_rcpt rule set, 51
 - check_relay rule set, 51
 - Child, 218
 - Chunks (of body), 255
 - concatenating, 187
 - containing too many characters with high
 - bit set, 189–190
 - counting number of bytes in, 187
 - Milster replies per, 94
 - reviewing, 186–190
 - unsigned char* type, 187
 - writing to disk file, 256
 - Cleanup section, 87
 - cleanup() routine, 247–248, 250
 - Clickable link, 12
 - Close item, 101
 - CNAME records
 - adding, 49–50
 - infinite loops, 35
 - leading to other CNAME records, 35
 - URLs and, 35–36
 - Commands, case insensitive, 29
 - Comments
 - breaking up words with, 18–19
 - .forward file, 64
 - HTML, 18–20
 - intervening newlines, 19–20
 - spam aliases, 63
 - unbalanced angle brackets, 20
 - unknown HTML keyword in angle brackets
 - acting like, 19
 - URLs used as, 36–37
 - comments() function, 281, 284
 - Compilers, choosing, 45–46
 - Complex data, storing, 117
 - Concatenating chunks, 187
 - config_getitem() routine, 251–252
 - config_read() routine, 248–249
 - Configuration files
 - # comment character, 246
 - cleaning spaces around strings, 247–248
 - looking up values, 251–252

INDEX

- Configuration files *continued*
 - rereading, 220
 - routine actually reading, 248
 - running, 253
 - simplest form, 246
 - Configurations
 - dynamic, 246–256
 - static, 246
 - configure script, creation of, 226
 - configure.ac template file, 226
 - confINPUT_MAIL_FILTERS mc macro, 51
 - confMILTER_MACROS_CONNECT mc macro, 115
 - confMILTER_MACROS_ENVFROM mc macro, 115
 - confMILTER_MACROS_ENVRCPT mc macro, 115
 - confMILTER_MACROS_EOM mc macro, 115
 - confMILTER_MACROS_HELO mc macro, 115
 - Connect item, 100
 - Connect phase and Milter replies, 89–90
 - Connect section, 87
 - Connecting host
 - name of, 157
 - result of lookup of name, 161
 - Connection-context type, 154
 - Connection-oriented handler functions, 155
 - Connection-oriented resources, deallocating, 155
 - Connection-persistent information,
 - deallocating, 228
 - Connections
 - affecting, 155
 - behavior, 12–13
 - cipher suite used for, 165
 - cleanup, 200–203
 - deferring if host cannot be looked up, 164
 - defining context, 157
 - disconnecting by rejecting with tempfail, 121
 - initializing and timeout, 109
 - keeping track of, 161
 - listening for incoming, 126–127
 - logging, 161
 - number of envelopes processed during
 - connection, 202–203
 - total duration, 202–203
 - Milters rejecting, 89
 - rejecting, 122
 - reviewing, 156–161
 - skipping checks, 159
 - termination, 200–203
 - Connection-specific macros, 115
 - Content-Transfer-Encoding: header, 25
 - Content-Type: header, 256
 - Continue reply, 89–94, 96
 - cp pointer, 175, 189–190
 - Credit card information theft, 7–8
 - ctx context pointer, 113–114, 116–117, 121, 153, 155, 157, 166, 172, 178, 183, 194
 - Custom-added headers, 132
-
- ## D
- Daemons, 58
 - Data access routines, 113–127
 - DATA SMTP command, 120, 190
 - DATA phase, 133
 - Data portion
 - body, 190
 - headers, 190
 - reviewing, 176, 182–186
 - Databases, support for, 46–47
 - Date: header, 72, 131
 - &#ddd; expressions, 272
 - dealloc envelope() handler function, 229
 - Deallocating connection-oriented resources, 155
 - dealloc_connection() routine, 228
 - dealloc_envelope() routine, 228–229
 - deamper() routine, 274–275
 - Debugging
 - default level, 125
 - setting level, 124–126

- decimal() subroutine, 272–275
- Decoding
- base64 encoding, 258–265
 - character-entity encoding, 269–276
 - quoted-printable encoding, 265–269
 - URL-encoding, 277–279
- Default SMTP replies, 120
- Deferring envelopes, 245
- #define statement, 212
- delay_checks FEATURE, 51
- /dev/null file, 53, 65, 218
- df queue file, truncating, 144
- Dial-up accounts and nonfixed IP addresses, 4
- Dictionaries, 288–293
- dig program, 48
- Directories
- accepting core dumps, 208
 - defining with preprocessor #define directive, 209
 - for Milters, 215
- Discard reply, 89, 91–95
- Discarding envelopes, 245
- Disguising Subject: header, 16–18
- Distributed model, 232
- dn_expand() function, 224
- DNS (Domain Name Service)
- adding CNAME records, 49–50
 - sender identification, xvi
 - TXT record, 6
- DNS and BIND* (Albitz and Liu), 50
- DNS (Domain Name Services)-based services, xv–xvi
- DNS records
- domain *versus* subdomain, 47–48
 - setting up, 47–50
 - wildcard, 34–35
- dn_skipname() function, 223
- Domain Keys standard, 6
- Domain names, 47
- case insensitive, 29
 - registering for testing, 48
- Domain records, controlling, 15
- _domainkey domain, 6
- Domains
- adding new host, 47–48
 - enclosed in quotation marks, 29–30
 - versus* subdomains, 47–48
- D_REENTRANT, 112
- DSL (digital subscriber line) and nonfixed IP addresses, 4
- dsn argument, 122
- DSN reply code, changing, 121
- Dynamic configurations, 246–256
- ## E
- EHLO command
- arriving at unexpected times, 90
 - requiring before MAIL FROM: command, 161
 - reviewing, 161–165
 - sending site, 172
- EHLO/HELO phases and Milter replies, 89
- Email
- composed of multiple parts, 256–258
 - delivered using order specified by MX records, 14
 - detecting when received from MX servers, 221–225
 - dividing into small, well-defined units, xvi
 - false positives, 10
 - fictional persons created to receive, 61–63
 - filtering out unwanted, xv
 - graylisting, 242–244
 - literal "+" character inserted in user part, 75–77
 - maximum size, 247
 - policies for inbound and outbound, 232
 - postage, xvii
 - protecting good, 64–65
 - screening inbound and outbound, 234
 - significant spam rating, xvi
 - unsolicited, xiv
 - whitelisting, 241–242

INDEX

- Email addresses
 - determining who may have sold, 76
 - encoded @ character, 79
 - expressing abstractly as possible, 79–80
 - innocent person's as bounce address, 5
 - JavaScript obscuring, 80
 - masking URLs, 31
 - as plain text outside mailto: command, 79
 - posting to newsgroup, 67–74
 - reader cutting and pasting, 79
 - setting up for spam, 65–67
 - showing ultimate recipient, 77–78
 - spam email, 38
 - Usenet, 68
 - verifying, 77–78
- Email fraud
 - bank or credit card information theft, 7–8
 - Nigerian fraud, 7
 - password theft, 8
 - viruses and worms, 8–9
- Email readers, 233
- Empty addresses, 167
- Encryption key length, 170
- End of envelope, 190–197
- End of headers section, 87
- End of message section, 87
- End users
 - internal customer as, 233–234
 - modeling, 233–234
 - outside world as, 234
- End-of-body semaphore, 186
- End-of-envelope cleanup, 192–193
- End-of-message phase timeout, 192
- End-of-message routine and Milter replies, 95–96
- Envelope recipients
 - accepting, 91, 154
 - adding recipient, 138–140
 - addresses, 141–142
 - delivery agent name, 176
 - maximum number, 247
 - number of, 239
 - possible replies, 92
 - processing, 117
 - recipient address, 176
 - rejecting, 172
 - relay host, 176
 - removing, 140–143
 - removing address from list, 142
 - rule sets and aliasing modifying, 142
- Envelope senders
 - accepting, 156
 - address, 5, 91, 165–166, 170
 - authentication, 170
 - bounce messages, 167
 - deferring, 169
 - delivery agent name, 170
 - discarding, 156, 169
 - error notification sent, 166
 - MTAs rejecting, 16
 - rejecting, 156, 169
 - relay host part, 170
 - reviewing, 90
 - saving address, 169–170
 - source of spam email, 6
- Envelope-handling functions, deallocating
 - resources, 156
- Envelopes, 155
 - aborting, 197–200
 - accepting, 156
 - arbitrary number of recipients, 171
 - DATA headers, 191
 - DATA portion, 143, 191
 - deferring, 245
 - direct access to raw information, 43
 - discarding, 156, 245
 - end of, 190–197
 - falsifying sender address, 15–16
 - headers, 143
 - identifying, 225
 - MAIL FROM: command, 91
 - MAIL FROM envelope sender, 191
 - private data, 239
 - RCPT TO envelope recipients, 191
 - rejecting, 122, 156, 180, 245

- Envelope-specific information, deallocating, 228
 - Envelope-specific macros, 115
 - Envelope-specific resources, deallocating, 169, 184, 188, 192
 - Envfrom item, 100
 - Envrcpt item, 100
 - Eoh item, 101
 - Eom item, 101
 - ep pointer, 189
 - errno variable, 108, 111
 - errno.h included file, 111
 - Error messages, name of program for use in, 68
 - Errors
 - recording, 218
 - smfi_setconn() routine, 107
 - /etc/aliases file
 - editing, 62–63
 - minimal, 53–54
 - /etc/inetd.conf file, minimizing, 56–58
 - /etc/init.d directory, 57, 213
 - /etc/init.d/apache file, 58
 - /etc/magic file
 - tests in, 286
 - usage, 284–288
 - /etc/mail directory, running Milters under, 209–210
 - /etc/mail/aliases file, 53–54
 - /etc/mail/local-host-names file, setting up, 52–53
 - /etc/mail/milters directory, 209
 - /etc/rc* files, 57
 - /etc/syslog.conf file, 55
 - Exception process, whitelisting, 242
 - exit(2), 148
 - EXPN command, 77–78
 - Exporting shell macros, 215
 - EX_SOFTWARE, 111
 - Extended SMTP commands, 166, 172
 - EX_UNAVAILABLE, 111
- F**
- Fake recipients
 - automatic addresses, 62
 - creation of, 61–63
 - names corresponding to real services, 62
 - non-user names, 62
 - UNIX administrative names, 62
 - Fallback hosts and mail, 13–14
 - False positives, 10
 - Falsifying envelope sender address, 15
 - Fatal (nonrecoverable) errors, 148
 - Feedback
 - human, 237–239
 - possible mechanisms, 237–240
 - file (for viewing local files), 29
 - file program, 284–285
 - simplified, 286
 - testing, 287
 - Files, identifying types by file contents, 284–288
 - Financial institutions, 8–9
 - finger program, 75, 81
 - Firewalls, 9, 234
 - Fixed IP addresses, 4
 - Flags and smfiDesc structure, 103–104
 - flags item, 100
 - fork() function, 218
 - .forward file, 64–65
 - FreeBSD
 - copying startup scripts, 213
 - /root/bin/roll shell script, 56
 - freehostent() function, 236
 - ftp (File Transfer Protocol), 29
 - ftp daemon, 57
 - ftp host name, 52
 - Fudgenews, 67
 - missing command-line switches, 69
 - opening connection to news posting host, 69
 - post() subroutine, 70–71
 - switches, 68
 - Functions and Milter phases, 102

INDEX

Fuzzy address matching, 242–244
 fuzzy() subroutine, 243

G

gethostbyname() function, 236
 getipnodebyname() C library function, 164,
 222, 236
 GETLONG, 224
 getpeername(3), 157
 GETSHORT, 224
 getuid function, 211
 g.msn.com websites, 33
 GNU autoconf suite, 226–227
 GoodMailSystems website, xvii
 Gorillas, 3, 4–5
 Graylisting, 242–244
 greetpause FEATURE, 13
 Grokking site, 26–37
 GROUP command, 72
 Group ID, 213
 Guerrillas, 3, 5–6

H

haddr argument, 159
 Handle signals, 219–221
 Handler functions, 116, 151–203

- advisory-oriented, 155
- belonging to connection, 113
- connection-oriented, 155
- message-oriented (envelope-oriented), 154
- recipient-oriented, 154
- smfi data access routines, 113–127
 - xxfi_ prefix, 102
 - xxfi_ prefix for names, 102

 header item, 100
 Header sender address, 5
 Headers, 92, 143, 176–177, 190

- adding, 129–132, 191

- appearing multiple times, 177
- case insensitive names, 181
- changing, 135–138
- count of, 133
- custom-added, 132
- end of, 182–186
- illegal values, 179
- index into list of existing, 133
- inserting in messages, 132–135
 - Milters, 93–94
- MIME, 178
- missing, 177
- modifying, 191
- multiple lines, 177–178
- name, 177
 - in form of string, 130, 133, 136
- name portion, 178
- ordering, 93
- prefixed with literal X-, 131
- recording presence, 182
- rejecting, 180–182
- removing, 135–138, 191
- reviewing, 176–182
- RFC standards, 130, 134
- trace-type, 136
- tracking of offset, 136
- user-added, 136
- value, 130–131, 134, 177
- value portion, 178

Headers section, 87
 HELO command

- arriving at unexpected times, 90
- requiring before MAIL FROM: command,
 161
- reviewing, 161–165
- skipping, 90

 helo item, 100
 HELO/EHLO section, 87
 HELO/EHLO SMTP command, 120
 hicount counter, 189
 Hijacked PCs, xv, 5, 234
 host argument, 162

- Host names, 157
 - accepting, 162
 - adding to spam database, 234–235
 - case insensitive, 29
 - looking up, 161
 - for MX records, 48
 - records, 223
 - for posting to Usenet, 68
 - random word masquerading, 34–35
 - string containing, 162
 - validity, 162
 - host.domain form, 32
 - host.domain part, expressed as IP number, 32
 - Hosts
 - adding to existing domain, 47–48
 - comparing IP numbers, 235–237
 - disguising name, 32
 - enclosed in quotation marks, 29–30
 - IP number of connecting, 157
 - MX records, 13, 223
 - names of, 175
 - redirecting site, 33–34
 - using other aliases, 49–50
 - HTML
 - bogus keywords, 283–284
 - camouflaging body, 18–22
 - character-entity encoding, 20–21
 - clickable link in code, 12
 - commands and URLs referenced case insensitive, 29
 - commands and web references, 27–28
 - comments, 18–20
 - declaring common keywords, 280
 - detecting non-HTML words, 281
 - documentation, 99
 - intervening newlines in comments, 19–20
 - keywords, 283
 - order of encoding, 22
 - unknown keywords acting like comments, 19
 - URL encoding, 21–22
 - valid keywords, 19
 - HTML comments
 - illegal form, 280
 - legal form, 279–280
 - stripping, 279–284
 - HTML documents and special characters, 20–21, 269–276
 - HTML (Hypertext Markup Language)-enabled email readers, 8
 - HTML-capable mail programs, 18–19
 - http (Hypertext Transport Protocol), 29
 - HTTP listener, 58
 - https (HTTP with Secure Sockets Layer, or SSL), 29
 - Human feedback, 237–239
- I**
- \$i macro, 114, 226
 - ident lookup, 160
 - identd, 75
 - Idle, Eric, 10
 - if clause, 117
 - IMAP (Internet Message Access protocol)
 - email readers, 233
 - include/milter.h file, 154
 - inet: prefix, 106–107, 112
 - inet6: prefix, 106–107
 - inetd daemon, 58, 81
 - inetd.conf file, commenting out lines, 58
 - INPUT_MAIL_FILTER mc command, 51
 - Installing Milter library, 46
 - Internal PCs, risks imposed by, 234
 - IP addresses
 - associated with receiving (listening) interface, 160
 - fixed, 4
 - nonfixed, 4–5
 - reverse look up, 157
 - IP numbers
 - assigning multiple to network interface, 15
 - comparing, 235–237

INDEX

- IP numbers *continued*
 - connecting host, 157
 - decimal or hexadecimal, 32
 - rejecting connections from, 221
 - spam-sending site, xv
 - used by machines without fixed IP numbers, xv
- ip pointer, 261–262
- IPv4 socket, 106–107
- IPv6 socket, 106–107
- ishtmlcmp() function, 281
- ISPs, 4–5
- isspace() C language library routine, 259
- Items and zero-length string, 250

- J**
- JavaScript, obscuring email address, 80
- JavaScript.Encode URLs, 37
- Jones, Terry, 10

- K**
- Keystroke logging, 8–9
- Keywords
 - character-entity encoding, 20–21
 - valid HTML, 19
- Kill (Ctrl+C) keyboard shortcut, 112

- L**
- l items, 112
- Large ISPs
 - policies, 5
 - spam, 4–5
 - TXT record, 6
 - whitelisting, 241
- LDAP (Lightweight Directory Access Protocol), 232
- Leftmost comparison, 251
- len argument, 187
- libmilter directory, 46
- libmilter RPM (Redhat Package Manager), 45
- libmilter/docs documentation, 151–152
- libmilter.h file, 102
- libmilter/mfapi.h included file, 111
- Library routines, reporting errors to, 124–126
- Lines: header, 73
- Linux
 - copying startup scripts, 213
 - method used to halt Milter, 215
 - /usr/sbin/logrotate file, 56
- listen(3), 126
- Listeners
 - daemons as, 58
 - eliminating unwanted, 57
- Listening connection, establishing, 109
- Listening daemon, name of, 160
- listen(2) queue, 126–127
- listen(3) queue, 127
- Literal character-entity encoding, 20–21
- lnsl, 112
- loadwords() routine, 288–289
- local: prefix, defining, 105–106
- local3 logging facility, 55
- localhost loopback interface, 112
- local-host-names file, 52–53
- Log files. limiting size, 56
- logadm program, 56
- Logging
 - connections, 161
 - defensive programming, 226
 - facilities available for nonsystem programs, 55
 - Milters, 225–226
 - number of envelopes processed during connection, 202–203
 - overview, 54–55
 - queue identifiers, 225
 - recording every connection, 161
 - rotating logs, 56

- sendmail, 128–129, 225
 - setting up, 54–56
 - setting up local#, 55–56
 - Solaris, 225
 - total connection duration, 202–203
 - logmilter Milter, 289
 - Logs, rotating, 56
 - Lost productivity, 9
- M**
- m4 Build file, 45
 - Macros
 - adding to default list, 160–161
 - connection-specific, 115
 - defining, 111, 258–259
 - end-of-file (or end of buffer), 259
 - illegal input character or white space characters, 259
 - needed, 111
 - envelope-specific, 115
 - fetching values, 114–115
 - name whose value is looked up, 114–115
 - passing sendmail macros to Milter, 115
 - persisting, 114
 - xxfi handler function return values, 153
 - xxfi_connect() handler function, 160–161
 - xxfi_envfrom() handler function, 170
 - xxfi_envrcpt() handler function, 176
 - xxfi_eom() handler function, 193
 - xxfi_helo() handler function, 165
 - magic() routine, 286–287
 - Mail Abuse website, xvi
 - mail facility, 54
 - Mail fallback hosts, 13–14
 - MAIL FROM: command, 51, 86, 87, 120, 154–155
 - calling Milters, 85
 - ESMTP (Extended SMTP) arguments, 91
 - Milter replies, 90–91
 - reviewing, 165–171
 - mail host name, 52
 - mailto: command, 78–79
 - searching for, 65
 - main() function, 99, 252
 - arguments, 111
 - changing default socket time out, 100
 - minimal, 110–112
 - routing, 97
 - Makefile, 51, 226
 - Makefile.am template file, 226
 - malloc, 119
 - Masking signals, 220
 - Masking web addresses, 78–80
 - maxrcpts item, 247
 - maxsize item, 247
 - mc configuration file
 - adding
 - macros to default list, 160
 - Milter support, 51
 - adding macros, 193
 - delay_checks FEATURE, 51
 - editing, 50–51
 - naming Milters, 101–102
 - smfi_setconn() routine, 107
 - mc macros, 115
 - Memory
 - allocating for strings, 122
 - freeing allocated, 118
 - Memory leaks, avoiding, 227–229
 - Message-Id: header, 73, 185, 193
 - Message-oriented (envelope-oriented) handler functions, 154
 - Messages
 - aborting, 96
 - accepting, 135, 154
 - adding header, 129–132
 - to be logged, 54
 - body, 92
 - bouncing, 5, 235
 - changing, 121
 - data portion, 178
 - discarding, 135

INDEX

- Messages *continued*
 - headers, 92, 132–135
 - lacking Message-Id: header, 185
 - large chunks of random text, 23–24
 - left with no recipients, 141
 - multiple Milters reviewing, 131–132, 135
 - quarantining, 146–148, 191
 - rejecting, 135, 185
 - reviewing data portion, 176, 182–186
- Microsoft Windows, 285
- MI_FAILURE value, 103, 109, 119, 121–123, 127, 130, 132–136, 139, 141, 143–144, 146
- Milter header file, 99
- MILTER macro, 215
- MILTERARGS macro, 215
- MILTERDIR macro, 215
- MilterEmailAddress variable, 240
- milter.init script, 214–216
- MILTERKILL macro, 215
- Milter-library, 97
 - declaring Milter phases, 100
 - installing, 46
 - overview, 97–99
 - registering smfiDesc structure with, 112
 - routines, 97–98
 - smfi_prefix, 97, 99
 - version, 102
 - xxfi_prefix, 99
- MILTERRUN macro, 215
- Milters, 85
 - abort phase, 96
 - aborting, 214
 - adding support in sendmail, 50–52
 - architecture, 231–232
 - baby-sitting script sleep time, 215
 - beginning execution, 100
 - capabilities, 103–104
 - command-line arguments, 215
 - communicating with sendmail, 104
 - configuration file, 209, 232
 - considering portability early, 226–227
 - database, 209
 - debugging level, 124–126
 - declaring phases acceptable or ignorable, 100
 - default wait, 109
 - defining
 - macros, 111
 - name, 215
 - directory for, 215
 - distributed model, 232
 - dynamic configurations, 246–256
 - email address stored in variable, 240
 - failing, 103, 214
 - functions for phases, 102
 - headers, 93–94
 - immediate abort, 219
 - interweaving calls to many, 86–87
 - killing, 112
 - learning from human input, 238–239
 - libraries needed, 112
 - listening, 109
 - logging, 225–226
 - macros for passing sendmail macros to, 115
 - main() function, 99–100
 - method used to halt, 215
 - multiple reviewing message, 131–132, 135
 - multithreaded, 113, 214
 - name of, 101–102
 - non-root user, 211–213
 - order called, 132, 135, 138
 - orderly shutdown, 219
 - phrases accepted or ignored, 100–103
 - port numbers listening on, 106
 - post-connection cleanup, 96
 - preventing from running as root, 211
 - private variables, 100
 - process phases, 87
 - queue identifiers, 225
 - quitting, 148–149
 - real user ID, 211
 - as recipient, 240
 - registering with library, 88
 - regular-expression rules, 244

- rejecting connection, 89
- rejecting SMTP command, 119–120
- replies
 - for Connect and Ehlo/Helo, 89–90
 - at end-of-message routine, 95–96
 - to MAIL FROM: command, 90–91
 - per chunk, 94
 - to RCPT TO: command, 91–92
- required initialization elements, 99–100
- return values from multiple, 88
- reviewing recipients, 91–92
- role of, 85–86
- running
 - in background, 110, 217–219
 - under /etc/mail directory, 209–210
 - in foreground for testing, 219
 - by root, 105
 - in /usr/local directory, 210
- sendmail
 - point of view, 86–87
 - supporting, 45
- SMTP DATA replies, 92–94
- sockets, 100
- source code examples, 289–290
- starting, 213–217
- startup script, 213–217
- static configurations, 246
- status or startup files, 209
- stopping, 213–217
- syslog records, 102
- T parameter, 191
- tempfailing SMTP command, 119–120
- time before restarting, 215
- timeout on amount of time, 145
- UNIX domain socket, 210
- updating knowledge, 232
- use of multiple, 85
- user ID, 210–213
- waiting for connection from sendmail, 112
- where to run, 208–210
- milfers directory, 289–290
- MILTERSEMAPHORE macro, 215
- MIME (Multipurpose Internet Mail Extensions)
 - attachments, 24–25
 - Content-Type: header, 256
 - headers, 178
 - headers and attachments, 285
- MIME-encoded boundaries, parsing, 256–258
- MIME-encoded messages, 187
- Missing headers, 177
- MI_SUCCESS value, 119, 121, 125, 130, 133, 136, 139, 141, 144, 146
- Monty Python's Flying Circus*, 10
- msg argument, 122–123
- MTAMARK (Marking Mail Transfer Agents) x
 - standard, 6
- MTAs (mail transfer agents), xiii, 6
 - multiline reports, 124
 - rejecting envelope sender, 16
- Multiline replies, 123–124
- Multipart messages, 256–258
- Multithreaded mode, launching, 109–110
- Multithreaded operation, 112
- Multithreaded program
 - deallocating resources, 96
 - signals, 220
- MX host, spam email sent directly to highest-numbered, 222
- MX records, 13
 - adding, 48–49
 - controlling domain records, 15
 - extracting host name associated with, 224
 - looking up, 48–49
 - printing, 225
 - trapping IP number subterfuge, 14
- MX (mail exchange) servers
 - anticipating, 221–225
 - deferring envelopes, 245
 - detecting when mail received from, 221–225
 - looking up, 222–224
 - relaying spam through, 13–15
 - unable to run spam filters on, 222

INDEX

mx() function, 222
 testing, 208, 224
mysql, 232

N

name argument, 178
Name item, 100
Named pipes, 104–106
Named sockets, 105–106
Network connections and bait machine, 46
newaliases command, 54
newaliases program, running, 65
Newline characters, 289–290
News server
 acknowledging posting, 73
 allowing posting, 72
 host sending greeting, 71
Newsgroups
 to post to, 68
 posting to, 67
 validating existence, 72
Nigerian fraud, 7
nmap program, 56–57
Non-root user, 211–213
NOQUEUEID string, 226
Nwords global variable, 289
nwords() function, 290–291

O

okaymail user, 65
~/oksenders file, 244
Old MTA addresses, 167, 173
op pointer, 261–262
Operating systems
 posix threads, 44
 thread-safe C language library, 44–45
Organized crime and Nigerian fraud, 7
ourmilt.run script, 216–217

P

Parent, 218
Parsing MIME-encoded boundaries, 256–258
Passing state, 255
Passwords, 8–9
Paul Graham Spam website, xvi
Pdata structure, 119
pdatap pointer, 119
percenthex() subroutine, 277–279
Phone numbers
 detection of, 38
 whitelisting, 242
Phonemes, 23
Platform, choosing for bait machine, 44–47
Plus addressing, 75–77
Pointers, storing single, 119
POP (Post Office Protocol) email readers,
 233
Portability, 226–227
Ports
 excluding non-email, 56–58
 list of numbers, 57
 unnecessary services listening, 57
Posix threads, 44–45
POST command, 72
Postage, xv, xvii
Post-connection cleanup, 96
Posting
 articles, 70–71
 to Usenet news groups, 67–74
post() subroutine, 70–71
Preventive measures
 EXPN command, 77–78
 telling users about plus addressing, 75–77
Printer hosts name, 175
printf() statements, 219
Printing MX records, 225
priv pointer, 170
priv variable, 185
Private data, 239
 allocating memory to pointer, 117

- fetching, 118–119
 - registering, 116–118
 - Private variables, 100
 - priv->qid variable, 226
 - procmail program, 242
 - Programs run as root, 211
 - Protecting good email, 64–65
 - Protocols
 - default, 31
 - enclosed in quotation marks, 29–30
 - identifying in URL, 29–30
 - not actually present with each URL, 31
 - Pthreads. *See* posix threads
 - pthread_sigmask() library routine, 221
- Q**
- qpdecode() function, 266–269
 - Quarantine reply, 96
 - Quarantining messages, 146–148, 191
 - Queue identifiers, 225–226
 - Queued messages
 - not seen by sendmail, 146–148
 - sendmail identifier, 170
 - QUIT command, 74
 - Quitting Milters, 148–149
 - Quoted-printable encoded attachments, 24–25
 - Quoted-printable encoding, decoding, 265–269
- R**
- Random text, 23–24
 - RCPT TO: command, 117, 120
 - calling Milters, 85
 - Milter replies to, 91–92
 - reviewing, 171–176
 - RCTP TO: command, 87
 - rd.yahoo.com website, 33
 - README file, 46
 - Real user IDs, 211
 - Rebooting bait machine, 58–59
 - Received: headers, 131, 177, 182
 - Receiving (listening) interface, 160
 - Recipient address @ character, 175
 - Recipient-oriented handler functions, 154
 - Recipients
 - accepting, 154
 - counting number of, 116–117
 - number of bad, 193
 - rejecting, 122
 - Recording errors, 218
 - Redirect servers, 33–34
 - Redirecting site, 33–34
 - regerror() function, 244
 - regexec() C library routine, 243
 - Registering private data, 116–118
 - Regular-expression evaluation, 243
 - Reject decision, 88
 - Reject reply, 89, 91–95
 - Rejecting
 - connections from IP numbers, 221
 - envelopes, 245
 - spam, 244–246
 - Relaying through MX (mail exchange), 13–15
 - Resources
 - deallocating, 96
 - connection-oriented, 155
 - envelope-handling functions, 156
 - failure to deallocate temporary, 118
 - res_query() function, 223
 - return keyword, 112
 - Return values from multiple milters, 88
 - Reverse DNS, 6
 - Reverse lookup of IP address, 157
 - Reviewing connections, 156–161
 - Reviewing SMTP HELO/EHLO, 161–165
 - RFC1413 validation, 160
 - Risks with internal PC customers, 234
 - root user
 - delivering mail for, 53
 - executing programs, 211

INDEX

- root user *continued*
 - preventing Milters from running as, 211
 - programs run as, 211
 - /root/bin/roll shell script, 56
 - Rotating logs, 56
 - Routers, 9, 234
 - Routines
 - body, 255–293
 - decoding
 - base64 encoding, 258–265
 - character-entity encoding, 269–276
 - quoted-printable encoding, 265–269
 - URL-encoding, 277–279
 - /etc/magic file usage, 284–288
 - parsing MIME-encoded boundaries, 256–258
 - passing state, 255
 - stripping HTML comments, 279–284
 - Rule sets
 - disposing of message, 96
 - rejecting connection, 89
 - runas() function, 211–213
- S**
- Sanity process and whitelisting, 242
 - <script command, 37
 - Semaphore file, 217
 - Sender identification, xvi
 - Sending site
 - connection cleanup, 200–203
 - EHLO command, 172
 - sendmail
 - 220 greeting, 12–13, 90
 - 220 SMTP code, 156
 - adding Milter support, 50–52
 - aliases, 53–54
 - buffer for incoming message, 12–13
 - configuring, 50–54
 - connection request from sending host, 89
 - getpeername(3), 157
 - greetpause FEATURE, 13
 - header added by Milter, 131
 - host and RFC1413 validation, 160
 - interweaving calls to many Milters, 86–87
 - killing and restarting, 52
 - log records, 225
 - logging, 128–129
 - mail facility, 54
 - mc macros, 115
 - minimal aliases file, 53
 - as MTA (mail transfer agent), xiii
 - multiple Milter programs, 85
 - plus addressing, 75–77
 - point of view on Milters, 86–87
 - queried files not seen by, 146–148
 - rejecting envelope recipient, 172
 - reverse lookup of IP address, 157
 - setting up local-host-names file, 52–53
 - SMART_HOST option, 4
 - source directory, 50
 - version supporting Milters, 45
 - where and how to deliver email, 64
 - sendmail*, 3rd edition (Costales and Allman), xiv
 - sendmail configuration file
 - Milter.LogLevel option, 128–129, 131, 134, 136, 138, 140, 142, 145, 147
 - order Milters called, 132, 135, 138
 - sendmail Cookbook* (Hunt), xiv
 - sendmail macros
 - fetching values, 114–115
 - xxfi_connect() handler function, 160–161
 - xxfi_envfrom() handler function, 170
 - xxfi_envrcpt() handler function, 176
 - xxfi_helo() handler function, 165
 - sendmail Milters, xv, 43
 - sendmail Performance Tuning* (Christenson), xiv
 - sendmail website, 45
 - sendmail.cf file, 52
 - sendmail.mc file, 50–52
 - Services
 - screening URLs, xvi–xvii
 - unnecessary listening on ports, 57
 - setsid() function, 218

- sfstatat type, 153, 157–158, 166, 168, 171, 174, 178, 180, 183–184, 186, 194–195, 197, 201
- Shell macros, 214–215
- Shutting down in stages, 148
- sig() function, 220
- SIGHUP signal, 219–221
- SIGINT signal, 219–221
- sigmarkreadconf() function, 220
- sigmarkrereadconf() function, 220–221
- Signals, 219–220
- Signature detectors, attempting to fool, 23–24
- SIGPIPE signal, 219–221
- SIGTERM signal, 219–221
- SIGUSR1 signal, 219–221
- SIGUSR2 signal, 219
- sizeof (3) integer, 117
- Sleepycat DB, 46–47
- slocal program, 64, 242
- slowmilt open source, 290
- Small businesses and whitelisting, 241
- SMART_HOST option, 4
- smfi data access routines, 113–127
- smfi modifier routines, 127–149
- smfi routines, 97, 151
- smfi_addheader() routine, 98, 129–130, 191, 196
 - ctx connection-context pointer, 130
- smfi_addrcpt() routine, 98, 138–140
- smfi_chgheader() routine, 98, 135–138, 191
- smfi_delrcpt() routine, 98, 140–141, 240
- smfiDesc structure, 100–103, 111–112, 158, 162–163, 167–168, 173, 179, 183, 187, 194, 198, 201
 - declaring xxfi_ functions, 155
 - flags, 103–104
 - global or local, 101
 - items, 100–101
 - position of xxfi_connect() handler function, 158
 - registering, 100
 - with milter-library, 112
 - with smfi_register() function, 103
- SMFIF_ADDHDRS flag, 104, 129, 132
- SMFIF_ADDRRCPT flag, 104, 138–139
- SMFIF_CHGBODY flag, 104, 143
- SMFIF_CHGHDRS flag, 104, 135
- SMFIF_DELRCPT flag, 104, 140–141
- SMFIF_QUARANTINE flag, 104, 146–147
- SMFIF_ADDHDRS flag, 104, 129, 132
- SMFIF_ADDRRCPT flag, 104, 138–139
- SMFIF_CHGBODY flag, 104, 143
- SMFIF_CHGHDRS flag, 104, 135
- SMFIF_DELRCPT flag, 104, 140–141
- SMFIF_NONE flag, 104
- SMFIF_QUARANTINE flag, 104
- smfi_getpriv() routine, 98, 113, 117–119, 170, 192, 199, 203, 239
- smfi_getsymval() routine, 98, 112–115, 165, 170, 176, 193, 226
- smfi_insheder() routine, 98, 132–135
- smfilter structure, 103
- smfi_main() routine, 98, 100, 109–110, 112
- smfi_opensocket() routine, 98, 104–109, 112
- smfi_prefix, 97, 99
- smfi_progress() routine, 98, 145–146, 191, 192
- smfi_quarantine() routine, 98, 146–147, 191
- smfi_register() routine, 97–98, 100–103, 102, 112, 158, 162, 167, 173, 179, 183, 187, 194, 198, 201
- smfi_replacebody() routine, 98, 143–144, 191
- SMFIS_ACCEPT return value, 153, 158–159, 163, 168, 174, 180, 184, 188, 195
- SMFIS_CONTINUE return value, 153, 159, 164, 169, 174, 180, 184, 188, 195, 240
- SMFIS_DISCARD return value, 153, 159, 164, 169, 174, 180, 184, 188, 195
- smfi_section() routine, 105–107
- smfi_setbacklog() routine, 98, 113, 126
- smfi_setconn() routine, 98, 104–109, 112
- smfi_setdbg() routine, 98, 113, 124
- smfi_setmlreply() routine, 98, 113, 123–124
- smfi_setpriv() routine, 98, 113, 116–118, 193, 203, 228
- smfi_setreply() routine, 113, 119–122
- smfi_settimeout() routine, 98, 109, 112

INDEX

- SMFIS_REJECT return value, 153, 159, 163, 168, 174, 180, 184, 188, 195
- SMFIS_TEMPFAIL return value, 153, 159, 163, 168, 174, 180, 184, 188, 195
- smfi_stop() routine, 98, 148–149
- SMFI_VERSION literal expression, 102
- SMFI_VERSION macro, 111
- SMTP (Simple Mail Transfer Protocol)
 - changing reply, 119–122
 - DATA portion, 190
 - EHLO command, 90
 - envelopes, 15
 - EXPN command, 77–78
 - HELO command, 90
 - MAIL FROM: command, 90
 - MAIL FROM part, 5
 - Milter replies with data, 92–94
 - modifying messages, 127–149
 - reply code, 122
 - reviewing HELO/EHLO, 161–165
- smtp argument, 121
- SMTP commands
 - extended, 166, 172
 - rejecting, 119–120
 - tempfailing, 119–120
- Sockets
 - changing default time out, 100
 - opening, 107–109
 - setting up, 112
 - smfi_setconn() routine, 107
- Solaris
 - baby-sitting script, 217
 - copying startup scripts, 213
 - lnsl switch, 68
 - log records, 225
 - logadm program, 56
 - method used to halt Milter, 215
- Source code Milter examples, 289–290
- SPAM, xiv
- Spam, xiv
 - adding URL's host to database, 234–235
 - aliases and comments, 63
 - archiving, 244–246
 - attempting to fool signature detectors, 23–24
 - bouncing, 16
 - camouflaging HTML body, 18–22
 - clickable web reference (URL), 26
 - connection behavior, 12–13
 - constantly changing and evolving, 12
 - disguising Subject: header, 16–18
 - disposing of, 234
 - email addresses, 38
 - evolution of, 3
 - exponential growth, 10
 - falsifying envelope sender address, 15
 - filtering, xvi, 234
 - full-blown war against, 3
 - grokking site, 26–37
 - human view of, 11
 - internally structured, 12
 - ISPs, 4–5
 - large ISPs, 4–5
 - lost productivity, 9
 - maintaining history, 234–237
 - method to contact spammer, 26
 - passing through, 244–246
 - phone numbers, 38
 - possible feedback mechanisms, 237–240
 - rejecting, 244–246
 - relaying through MX (mail exchange), 13–15
 - religiously or politically motivated, 38
 - selling something, 38
 - setting up addresses to be gathered, 65–67
 - speeding up process, 12
 - tracking source, 76
 - unnecessary encoding, 24–25
 - what to do with, 232
- Spam address-gathering software, 77–78
- Spam detection software
 - envelope sender, 16
 - phonemes, 23
 - “Spam Song,” 10

- Spam suppression
 - cost of, 9–10
 - software, 10
 - Spam-blocking software recognizing spam, 11
 - Spammers
 - hijacked PCs, 5
 - method for recipient to contract, 26
 - thinking like, 38–39
 - Spam-screening software, 19
 - Spam-sending sites
 - false envelope sender, 16
 - IP number, xv
 - Special characters
 - HTML documents, 20–21, 269–276
 - redirect servers, 33–34
 - SPF (Sender Policy Framework) standard, 6
 - Spyware, 8
 - SQL (Structured Query Language) database, 232
 - src directory, 290
 - _srv._smtp.perm domain, 6
 - start argument, 216
 - Startup script, 213–217
 - State, passing, 255
 - stat() function, 289
 - Static configurations, 246
 - Status information, 155
 - stdin, 218
 - stdio.h included file, 111
 - stdout, 218
 - stop argument, 216
 - strerror() function, 108, 111, 121
 - String constants, 276
 - string.h included file, 111
 - Strings
 - allocating memory for, 122
 - cleaning spaces around, 247–248
 - longer than 980 characters, 122–123
 - Stripping HTML comments, 279–284
 - strtol() C library function, 268
 - struct keyword, 101
 - struct priv_struct type, 170
 - Structure, 101
 - Subdomains, 47–48
 - Subject: headers, 181
 - base64-encoding, 17–18
 - disguising, 16–18
 - submit.cf file, 52
 - Syntax error, 122–123
 - sysexit.h included file, 111, 208
 - syslog(3), 203
 - syslog records, 102
 - syslogd daemon, restarting, 56
 - syslog() function, 218, 226
 - System password database, 212
- T**
- Target file, 55
 - TCP sockets, 104, 106–107, 112
 - TCP/IP (Transmission Control Protocol/Internet Protocol), 12–13
 - Telnet to known web server on port 80, 66
 - Tempfail decision, 88
 - Tempfail reply, 89, 91–95
 - Template include file, 226
 - Test machine. *See* bait machine
 - test Militer, 290
 - Text, computing value for, 23–24
 - Theft
 - of bank or credit card information, 7–8
 - of passwords, 8
 - Thinking like spammers, 38–39
 - Threads, 148, 220
 - time(3) C library routine, 203
 - TLS encryption key length, 165
 - TLS/SSL (Transaction Layer/Secure Socket Layer) version, 165
 - T_MX type, 224
 - Tomlinson, Fred, 10
 - Trace-type headers, 136
 - Trapping signals, 220
 - ttl (time to live), 224

INDEX

TXT record, 6
Type, 224

U

umask, 105
Underlying database whitelisting, 241
Units, xvi
UNIX, 284–285
unix: prefix, 105–106
UNIX domain socket, 108, 210
UNIX System Administration Handbook, 3rd edition (Nemeth, Snyder, Seebass, and Hein), xiv
Unnecessary encoding, 24–25
unsigned char* type, 187
Unsolicited email, xiv
Unwanted headers, 182
URL detection, xvi–xvii
URL-encoding, 21–22
 decoding, 277–279
URLs (Uniform Resource Locators)
 case insensitive, 29
 CNAME records and, 35–36
 decoding, 22
 email addresses masking, 31
 encoding, 21–22
 encountering @, 32
 hand-screening, xv
 host.domain form, 32
 hostname random word masquerading, 34–35
 identifying protocol, 29–30
 JavaScript.Encode, 37
 quotation marks when pasting, 29–30
 recording host names in database, 26
 services that screen, xvi–xvii
 used as comments, 36–37
Usenet
 commercial postings, xiv
 email addresses, 68

 plus addressing, 77
 posting to news groups, 67–74
 spam risks, 77

User IDs

 avoiding use of, 210
 Milters, 210–213
 nonzero, 213
 real, 211
 resetting, 213
 value as, 212

User names, 8–9

 user variable, 212

User-added headers, 136

Users

 discovering if logged in, 81
 modeling, 233–234
 outside world as, 234
 telling about plus addressing, 75–77

 /usr/local directory, 210

 /usr/local/etc/rc.d directory, 213

 /usr/local/nmh/lib/slocal program, 64

 /usr/sbin/logrotate file, 56

 /usr/share/dict/words file

 loading, 288–289

 usage, 288–293

V

Valid HTML keywords, 19

value argument, 178

Values and zero-length string, 250

 /var/log/maillog file, 55

 /var/log/milter.log file, 56

 /var/log/syslog file, 54

 /var/run/ourmilter.sock socket, 51

 /var/run/yourmilter directory, 208

Version item, 100

Vikings, 10

Virtual Conspiracy website, 37

Viruses, 8–9

VERFY command, 77–78

W

Web addresses, masking, 78–80

Web interface

email readers, 233

whitelisting, 241

Web references, 21

character-entity encoding, 22

<a command, 27

disguising, 26

HTML commands, 27–28

Web servers, running, 65–67

Websites, xv

while loop, 217

White space characters, 259–260

Whitelisting, 232, 241–242

Wildcard DNS records, 34–35

Words, breaking up with HTML comments,
18–19

Words global variable, 289

Worms, 8–9

write program, 81

www host name, 52

X

xfi_eom() handler function, 132

X-milter: header, 131

XMTPL, VRFY command, 77–78

xxfi_ handler functions, 226

common characteristics, 153

ctx argument, 154

declaring, 153

types, 154–155

xxfi_abort() handler function, 117, 152, 155–
156, 169, 184–185, 191, 192, 194, 197

calling common subroutine to deallocate
envelope data, 200

ctx private-context pointer, 198

deallocation routines called from, 227

example, 199

recording that Milter aborted, 200

usage, 197–199

xxfi_body() handler function, 143, 152, 154,
156, 255

allowing selected local recipients to receive
messages, 190

archiving copy of outbound email, 190

arguments, 186–187

calling, 186

ctx private-context pointer, 186

example, 189–190

len argument, 187

return values, 188

saving (buffering) body to file or in
memory, 190

saving (buffering) body without
attachments, 190

screening body for viruses, 190

storing attachments in database, 190

usage, 186–189

xxfi_close() function handler, 193

xxfi_close() handler function, 118, 152, 155,
169, 184, 185, 199, 228

calling, 200–201

ctx private-context pointer, 201

ensuring allocated envelope data
deallocated, 203

example, 202–203

summarizing actions taken by connecting
site, 203

usage, 200–202

xxfi_connect() handler function, 114, 118–119,
152, 155–157

detecting if connection is on loopback
interface, 161

example, 159–160

haddr argument, 159

keeping track of connections, 161

looking up host name and IP number,
161

return values, 158–159

sendmail macros, 160–161

INDEX

- xxfi_connect() handler function *continued*
 - sfstatat type, 158
 - usage, 158–159
- xxfi_envfrom() handler function, 114, 117,
 - 152, 154, 156, 165–166, 170, 185, 191,
 - 197, 225–226
 - comparing IP address to list of rejected addresses, 171
 - envelope sender allowed to send mail in domain, 171
 - example of, 169–170
 - rejecting connections with small encryption key length, 171
 - return values, 168–169
 - sendmail macros, 170
 - usage, 166–169
- xxfi_envrcpt() handler function, 114, 117, 142,
 - 152, 154, 156, 191
 - addressees of message inside, 239
 - arguments, 172
 - argv argument, 172
 - calling, 171
 - counting number of good recipients, 176
 - ctx private-context pointer, 172
 - example, 175–176
 - list of honey-pot (bait) recipients, 176
 - missing recipients can be found, 176
 - return values, 174
 - sendmail macros, 176
 - usage, 171–174
 - validating whitelisting pairs, 176
- xxfi_eoh() handler function, 152, 154, 156
 - calling, 183
 - comparing number of envelope recipients to number of header recipients, 186
 - ctx private-context pointer, 183
 - example, 185
 - flagging missing headers, 186
 - logging statistical review of headers, 186
 - return values, 184
 - usage, 183–184
- xxfi_eom() handler function, 117, 127, 129,
 - 132–133, 135–136, 138–139, 141, 143–
 - 145, 147, 152, 154, 156, 169, 185–186,
 - 190–191, 240
 - adding envelope recipient, 197
 - adding headers found to be missing, 196
 - argument, 194
 - calling, 227
 - changing value of headers, 197
 - constrained by time limits, 191–192
 - ctx private-context pointer, 194
 - deallocation routines called from, 227
 - decoding body, 197
 - example, 196
 - logging summary of everything Milter did with envelope, 196
 - removing envelope recipient, 197
 - removing junk headers, 197
 - return values, 195
 - screening body to detect spam, viruses, and unwanted attachments, 197
 - sendmail macros, 193
 - usage, 194–195
- xxfi_header() handler function, 152, 154, 156,
 - 176–177, 185, 191
 - arguments, 178
 - checking header values for adherence to standards, 182
 - ctx private-context pointer, 178
 - detecting bogus Received: headers, 182
 - example, 180–182
 - name argument, 178
 - recording presence of header, 182
 - return values, 180
 - unwanted headers, 182
 - usage, 178–180
 - value argument, 178
- xxfi_helo() handler function, 114, 152, 155, 161
 - arguments, 162
 - calling, 162
 - ctx private-context pointer, 162

detecting spamming software patterns in
 HELO/EHLO string, 165
example, 164
host argument, 162
return values, 163–164
sendmail macros, 165
sfsistat type, 162
usage, 162–164
verifying correct cipher suite, 165

xxfi_prefix, 99
xxfi_rcpt() handler function, 97

Z

Zero-length file for logging messages, 55
Zero-length string, 250
Zombie mail machine, 9

