

Mutt中文手册

译者: [Handsome Cheung](#) (E-mail:handsomecheung@gmail.com)

发布日期: 2009年12月24日(bate)

最后修改: 2009年12月24日

译者声明: 本手册以GNU的精神发布, 遵守GFDL许可证(GNU Free Documentation License), 详细内容请见[这里](#)。

译者前言: Mutt是一个功能强大的邮件客户端, 但要配置它并不简单, 英文手册又让很多人望而生畏。我深受自由软件精神的感动, 希望能为自由软件的传播作一点贡献, 所以就有了这本手册。由于水平有限, 我不能保证所有内容都准确无误, 拿不准的地方不敢贸然翻译, 留下原文。如果你发现文中的错误, 可以来信指出, 哪怕是错别字也可以。任何对于本手册质量有提高的建议我都会虚心接受。另外, 很多技术词汇无法找到让人满意的翻译, 所以如果你能够流畅阅读英文文档, 那么最好还是浏览[原手册](#)。这里是Mutt的官方网站: <http://www.mutt.org>。

Michael Elkins

<me@cs.hmc.edu>

版本: 1.5.19 (2009年1月5日)

摘要

“All mail clients suck. This one just sucks less.”-me, circa 1995年

目录

[1.概述](#)

- [1.Mutt的主页](#)
- [2.邮件列表](#)
- [3.从哪里得到Mutt](#)
- [4.Mutt在线资源](#)
- [5.捐助Mutt](#)
- [6.书写习惯](#)
- [7.版权声明](#)

[2.使用入门](#)

- [1.基本理念](#)
- [2.在菜单中移动](#)
- [3.使用编辑器](#)
 - [3.1 简介](#)
 - [3.2 输入记录](#)

[4.阅读邮件-Index和Pager](#)

[4.1 Index](#)

[4.2 Pager](#)

[4.3 Threaded模式](#)

[4.4 其它功能](#)

[5.发送邮件](#)

[5.1 简介](#)

[5.2 编辑邮件头](#)

[5.3 加密/签名邮件](#)

[5.4 通过mixmaster发送匿名邮件](#)

[5.5 发送format=flowed邮件](#)

[6.转发和bounce邮件](#)

[7.暂缓发送邮件](#)

[3.配置文件](#)

[1.配置文件的位置](#)

[2.配置文件的语法](#)

[3.地址组](#)

[4.定义/使用别名](#)

[5.更改默认键盘绑定](#)

[6.定义别名的字符集](#)

[7.根据信箱设置变量](#)

[8.键盘宏](#)

[9.使用颜色和mono属性](#)

[10.显示邮件头](#)

[11.后备地址](#)

[12.邮件列表](#)

[13.使用多个spool信箱](#)

[14.监视新邮件](#)

[15.自定义邮件头](#)

[16.指定默认保存邮箱](#)

[17.指定默认的Fcc信箱，编写邮件时](#)

[18.同时指定save-hook和fcc-hook的默认值](#)

[19.根据收件人改变设置](#)

[20.格式化邮件时更改设置](#)

[21.选择收件人的密钥](#)

[22.添加按键序列到键盘缓冲区](#)

[23.执行功能](#)

[24.邮件评分](#)

[25.垃圾邮件检测](#)

[26.设置和查询变量](#)

[26.1 命令](#)

[26.2 用户自定义变量](#)

[27.从另一个文件读入配置命令](#)

[28.移除hook](#)

[29.格式字符串](#)

[29.1 基本用法](#)

[4.高级用法](#)

[1.正则表达式](#)

[2.模式：搜索，提取和标记](#)

[2.1 模式修饰符](#)

[2.2 简单模式](#)

[2.3 组合模式](#)

[2.4 按日期搜索](#)

[3.使用标记](#)

[4.使用Hook](#)

[4.1 Hook中的邮件匹配](#)

[5.外部地址查询](#)

[6.邮箱格式](#)

[7.快捷符号](#)

[8.处理邮件列表](#)

[9.处理多个文件夹](#)

[10.编辑thread](#)

[10.1 连接thread](#)

[10.2 拆分thread](#)

[11.投递状态通知（DSN）支持](#)

[12.启用WWW浏览器](#)

[5.Mutt的MIME支持](#)

[1.在Mutt中使用MIME](#)

[1.1 在pager中查看MIME邮件](#)

[1.2 附件菜单](#)

[1.3 编辑菜单](#)

[2.配置文件mime.types](#)

[3.配置文件mailcap](#)

[3.1 mailcap文件基本内容](#)

[3.2 安全使用mailcap](#)

[3.3 mailcap的高级用法](#)

[3.4 mailcap文件示例](#)

[4.MIME自动显示](#)

[5.MIME Multipart/Alternative](#)

[6.附件搜索和计数](#)

[7.MIME Lookup](#)

[6.可选功能](#)

[1.General notes](#)

[1.1 启用/停用功能](#)

[1.2 URL语法](#)

[2.SSL/TLS支持](#)

[3.POP3支持](#)

[4.IMAP支持](#)

[4.1 浏览文件夹](#)

[4.2 认证](#)

[5.SMTP支持](#)

[6.管理多个帐户](#)

[7.本地缓存](#)

[7.1 邮件头缓存](#)

[7.2 邮件体缓存](#)

[7.3 维护](#)

[8.Exact address generation](#)

[7.性能优化](#)

[1.读写信箱](#)

[2.从远程文件夹阅读邮件](#)

[3.搜索和提取](#)

[8.参考](#)

[1.命令行参数](#)

[2.配置命令](#)

[3.配置变量](#)

[3.1 abort_nosubject](#)

[3.2 abort_unmodified](#)

[3.3 alias_file](#)

[3.4 alias_format](#)

[3.5 allow_8bit](#)

[3.6 allow_ansi](#)

[3.7 arrow_cursor](#)

[3.8 ascii_chars](#)

[3.9 askbcc](#)

[3.10 askcc](#)

[3.11 assumed_charset](#)

[3.12 attach_charset](#)

[3.13 attach_format](#)

[3.14 attach_sep](#)

[3.15 attach_split](#)

[3.16 attribution](#)

[3.17 autoedit](#)

[3.18 auto_tag](#)

[3.19 beep](#)

[3.20 beep_new](#)

[3.21 bounce](#)

[3.22 bounce_delivered](#)

[3.23 braille_friendly](#)

[3.24 check_mbox_size](#)

[3.25 charset](#)
[3.26 check_new](#)
[3.27 collapse_unread](#)
[3.28 uncollapse_jump](#)
[3.29 compose_format](#)
[3.30 config_charset](#)
[3.31 confirmappend](#)
[3.32 confirmcreate](#)
[3.33 connect_timeout](#)
[3.34 content_type](#)
[3.35 copy](#)
[3.36 crypt_use_gpgme](#)
[3.37 crypt_use_pka](#)
[3.38 crypt_autopgp](#)
[3.39 crypt_autosmime](#)
[3.40 date_format](#)
[3.41 default_hook](#)
[3.42 delete](#)
[3.43 delete_untag](#)
[3.44 digest_collapse](#)
[3.45 display_filter](#)
[3.46 dotlock_program](#)
[3.47 dsn_notify](#)
[3.48 dsn_return](#)
[3.49 duplicate_threads](#)
[3.50 edit_headers](#)
[3.51 editor](#)
[3.52 encode_from](#)
[3.53 envelope_from_address](#)
[3.54 escape](#)
[3.55 fast_reply](#)
[3.56 fcc_attach](#)
[3.57 fcc_clear](#)
[3.58 folder](#)
[3.59 folder_format](#)
[3.60 followup_to](#)
[3.61 force_name](#)
[3.62 forward_decode](#)
[3.63 forward_edit](#)
[3.64 forward_format](#)
[3.65 forward_quote](#)
[3.66 from](#)
[3.67 gecos_mask](#)
[3.68 hdrs](#)
[3.69 header](#)
[3.70 help](#)
[3.71 hidden_host](#)
[3.72 hide_limited](#)
[3.73 hide_missing](#)
[3.74 hide_thread_subject](#)
[3.75 hide_top_limited](#)
[3.76 hide_top_missing](#)
[3.77 history](#)
[3.78 history_file](#)

[3.79 honor_followup_to](#)
[3.80 hostname](#)
[3.81 ignore_linear_white_space](#)
[3.82 ignore_list_reply_to](#)
[3.83 imap_authenticators](#)
[3.84 imap_check_subscribed](#)
[3.85 imap_delim_chars](#)
[3.86 imap_headers](#)
[3.87 imap_idle](#)
[3.88 imap_keepalive](#)
[3.89 imap_list_subscribed](#)
[3.90 imap_login](#)
[3.91 imap_pass](#)
[3.92 imap_passive](#)
[3.93 imap_peek](#)
[3.94 imap_pipeline_depth](#)
[3.95 imap_servernoise](#)
[3.96 imap_user](#)
[3.97 implicit_autoview](#)
[3.98 include](#)
[3.99 include_onlyfirst](#)
[3.100 indent_string](#)
[3.101 index_format](#)
[3.102 ispell](#)
[3.103 keep_flagged](#)
[3.104 locale](#)
[3.105 mail_check](#)
[3.106 mailcap_path](#)
[3.107 mailcap_sanitize](#)
[3.108 header_cache](#)
[3.109 maildir_header_cache_verify](#)
[3.110 header_cache_pagesize](#)
[3.111 header_cache_compress](#)
[3.112 maildir_trash](#)
[3.113 mark_old](#)
[3.114 markers](#)
[3.115 mask](#)
[3.116 mbox](#)
[3.117 mbox_type](#)
[3.118 metoo](#)
[3.119 menu_context](#)
[3.120 menu_move_off](#)
[3.121 menu_scroll](#)
[3.122 meta_key](#)
[3.123 mh_purge](#)
[3.124 mh_seq_flagged](#)
[3.125 mh_seq_replied](#)
[3.126 mh_seq_unseen](#)
[3.127 mime_forward](#)
[3.128 mime_forward_decode](#)
[3.129 mime_forward_rest](#)
[3.130 mix_entry_format](#)
[3.131 mixmaster](#)
[3.132 move](#)

[3.133 message_cachedir](#)
[3.134 message_cache_clean](#)
[3.135 message_format](#)
[3.136 narrow_tree](#)
[3.137 net_inc](#)
[3.138 pager](#)
[3.139 pager_context](#)
[3.140 pager_format](#)
[3.141 pager_index_lines](#)
[3.142 pager_stop](#)
[3.143 crypt_autosign](#)
[3.144 crypt_autoencrypt](#)
[3.145 pgp_ignore_subkeys](#)
[3.146 crypt_replyencrypt](#)
[3.147 crypt_replysign](#)
[3.148 crypt_replysignencrypted](#)
[3.149 crypt_timestamp](#)
[3.150 pgp_use_gpg_agent](#)
[3.151 crypt_verify_sig](#)
[3.152 smime_is_default](#)
[3.153 smime_ask_cert_label](#)
[3.154 smime_decrypt_use_default_key](#)
[3.155 pgp_entry_format](#)
[3.156 pgp_good_sign](#)
[3.157 pgp_check_exit](#)
[3.158 pgp_long_ids](#)
[3.159 pgp_retainable_sigs](#)
[3.160 pgp_autoinline](#)
[3.161 pgp_replyinline](#)
[3.162 pgp_show_unusable](#)
[3.163 pgp_sign_as](#)
[3.164 pgp_strict_enc](#)
[3.165 pgp_timeout](#)
[3.166 pgp_sort_keys](#)
[3.167 pgp_mime_auto](#)
[3.168 pgp_auto_decode](#)
[3.169 pgp_decode_command](#)
[3.170 pgp_getkeys_command](#)
[3.171 pgp_verify_command](#)
[3.172 pgp_decrypt_command](#)
[3.173 pgp_clearsign_command](#)
[3.174 pgp_sign_command](#)
[3.175 pgp_encrypt_sign_command](#)
[3.176 pgp_encrypt_only_command](#)
[3.177 pgp_import_command](#)
[3.178 pgp_export_command](#)
[3.179 pgp_verify_key_command](#)
[3.180 pgp_list_secring_command](#)
[3.181 pgp_list_pubring_command](#)
[3.182 forward_decrypt](#)
[3.183 smime_timeout](#)
[3.184 smime_encrypt_with](#)
[3.185 smime_keys](#)
[3.186 smime_ca_location](#)

[3.187 smime_certificates](#)
[3.188 smime_decrypt_command](#)
[3.189 smime_verify_command](#)
[3.190 smime_verify_opaque_command](#)
[3.191 smime_sign_command](#)
[3.192 smime_sign_opaque_command](#)
[3.193 smime_encrypt_command](#)
[3.194 smime_pk7out_command](#)
[3.195 smime_get_cert_command](#)
[3.196 smime_get_signer_cert_command](#)
[3.197 smime_import_cert_command](#)
[3.198 smime_get_cert_email_command](#)
[3.199 smime_default_key](#)
[3.200 ssl_client_cert](#)
[3.201 ssl_force_tls](#)
[3.202 ssl_starttls](#)
[3.203 certificate_file](#)
[3.204 ssl_usesystemcerts](#)
[3.205 entropy_file](#)
[3.206 ssl_use_sslv2](#)
[3.207 ssl_use_sslv3](#)
[3.208 ssl_use_tlsv1](#)
[3.209 ssl_min_dh_prime_bits](#)
[3.210 ssl_ca_certificates_file](#)
[3.211 pipe_split](#)
[3.212 pipe_decode](#)
[3.213 pipe_sep](#)
[3.214 pop_authenticators](#)
[3.215 pop_auth_try_all](#)
[3.216 pop_checkinterval](#)
[3.217 pop_delete](#)
[3.218 pop_host](#)
[3.219 pop_last](#)
[3.220 pop_reconnect](#)
[3.221 pop_user](#)
[3.222 pop_pass](#)
[3.223 post_indent_string](#)
[3.224 postpone](#)
[3.225 postponed](#)
[3.226 preconnect](#)
[3.227 print](#)
[3.228 print_command](#)
[3.229 print_decode](#)
[3.230 print_split](#)
[3.231 prompt_after](#)
[3.232 query_command](#)
[3.233 query_format](#)
[3.234 quit](#)
[3.235 quote_regexp](#)
[3.236 read_inc](#)
[3.237 read_only](#)
[3.238 realname](#)
[3.239 recall](#)
[3.240 record](#)

[3.241 reply_regex](#)
[3.242 reply_self](#)
[3.243 reply_to](#)
[3.244 resolve](#)
[3.245 reverse_alias](#)
[3.246 reverse_name](#)
[3.247 reverse_realname](#)
[3.248 rfc2047_parameters](#)
[3.249 save_address](#)
[3.250 save_empty](#)
[3.251 save_history](#)
[3.252 save_name](#)
[3.253 score](#)
[3.254 score_threshold_delete](#)
[3.255 score_threshold_flag](#)
[3.256 score_threshold_read](#)
[3.257 send_charset](#)
[3.258 sendmail](#)
[3.259 sendmail_wait](#)
[3.260 shell](#)
[3.261 sig_dashes](#)
[3.262 sig_on_top](#)
[3.263 signature](#)
[3.264 simple_search](#)
[3.265 smart_wrap](#)
[3.266 smileys](#)
[3.267 sleep_time](#)
[3.268 smtp_authenticators](#)
[3.269 smtp_pass](#)
[3.270 smtp_url](#)
[3.271 sort](#)
[3.272 sort_alias](#)
[3.273 sort_aux](#)
[3.274 sort_browser](#)
[3.275 sort_re](#)
[3.276 spam_separator](#)
[3.277 spoolfile](#)
[3.278 status_chars](#)
[3.279 status_format](#)
[3.280 status_on_top](#)
[3.281 strict_threads](#)
[3.282 suspend](#)
[3.283 text_flow](#)
[3.284 thread_received](#)
[3.285 thorough_search](#)
[3.286 tilde](#)
[3.287 time_inc](#)
[3.288 timeout](#)
[3.289 tmpdir](#)
[3.290 to_chars](#)
[3.291 tunnel](#)
[3.292 use_8bitmime](#)
[3.293 use_domain](#)
[3.294 use_envelope_from](#)

[3.295 use_from](#)
[3.296 use_idn](#)
[3.297 use_ipv6](#)
[3.298 user_agent](#)
[3.299 visual](#)
[3.300 wait_key](#)
[3.301 weed](#)
[3.302 wrap](#)
[3.303 wrap_search](#)
[3.304 wrapmargin](#)
[3.305 write_inc](#)
[3.306 write_bcc](#)

[4.功能](#)

[4.1 通用菜单](#)
[4.2 index菜单](#)
[4.3 pager菜单](#)
[4.4 别名菜单](#)
[4.5 query菜单](#)
[4.6 附件菜单](#)
[4.7 编辑菜单](#)
[4.8 postpone菜单](#)
[4.9 浏览器菜单](#)
[4.10 PGP菜单](#)
[4.11 smime菜单](#)
[4.12 mix菜单](#)
[4.13 编辑器菜单](#)

[9.其它](#)

[1.致谢](#)
[2.关于本文档](#)

表格清单

[1.1 特殊条目书写习惯](#)
[2.1 通用导航键](#)
[2.2 常用行编辑器按键](#)
[2.3 index菜单常用按键](#)
[2.4 邮件状态标志](#)
[2.5 邮件收件人标志](#)
[2.6 pager菜单常用按键](#)
[2.7 ANSI转义序列](#)
[2.8 颜色序列](#)
[2.9 thread模式常用键](#)
[2.10 发送邮件常用按键](#)
[2.11 编辑菜单常用按键](#)
[2.12 PGP密钥菜单标志](#)
[3.1 键盘按键名称](#)
[4.1 POSIX正则表达式字符类](#)
[4.2 正则表达式重复操作符](#)
[4.3 GNU正则表达式扩展](#)
[4.4 模式修饰符](#)

- [4.5 简单搜索关键词](#)
- [4.6 日期单位](#)
- [8.1 命令行参数](#)
- [8.2 通用功能默认键绑定](#)
- [8.3 index菜单默认键绑定](#)
- [8.4 pager菜单默认键绑定](#)
- [8.5 别名菜单的默认键绑定](#)
- [8.6 query菜单默认键绑定](#)
- [8.7 附件菜单默认键绑定](#)
- [8.8 编辑菜单默认键绑定](#)
- [8.9 postpone菜单默认键绑定](#)
- [8.10 浏览器菜单默认键绑定](#)
- [8.11 PGP菜单默认键绑定](#)
- [8.12 smime菜单默认键绑定](#)
- [8.13 mix菜单默认键绑定](#)
- [8.14 编辑器菜单默认键绑定](#)

示例清单

- [3.1 一行多条配置命令](#)
- [3.2 注释配置文件](#)
- [3.3 在配置文件中转义引号](#)
- [3.4 在配置文件中使外部命令的输出](#)
- [3.5 在配置文件中使环境变量](#)
- [3.6 定义外部别名文件](#)
- [3.7 根据信箱设置排序方式](#)
- [3.8 显示哪些邮件头](#)
- [3.9 邮件头显示顺序](#)
- [3.10 自定义邮件头](#)
- [3.11 在save-hook中使用%-expandos](#)
- [3.12 嵌入push到folder-hook命令在](#)
- [3.13 配置垃圾邮件检测](#)
- [3.14 使用自定义变量使配置文件易读](#)
- [3.15 使用自定义变量来备份其他变量的值](#)
- [3.16 推迟自定义变量到运行时扩展](#)
- [3.17 在格式字符串中使用外部过滤器](#)
- [4.1 模式匹配中使用布尔操作符](#)
- [4.2 将my_hdr与send-hook结合](#)
- [5.1 附件计数](#)

第1章.概述

- [1.Mutt主页](#)
- [2.邮件列表](#)
- [3.从哪里得到Mutt](#)
- [4.Mutt在线资源](#)
- [5.捐助Mutt](#)
- [6.书写习惯](#)
- [7.版权声明](#)

Mutt是一个基于文本的小巧但强大的邮件客户端。Mutt具有很高的可配置性，而且其高级特性，如键绑定、键盘宏、thread模式、支持正则表达式的搜索和强大的模式匹配语言非常适合高级用户使用

1. Mutt的主页

Mutt的官方主页地址为<http://www.mutt.org/>。

2. 邮件列表

将正文为“subscribe”的邮件发送到以下邮件列表，即可订阅相应的邮件列表。

- <mutt-announce-request@mutt.org> — 一些关于mutt的信息通知
- <mutt-users-request@mutt.org> — 帮助，bug报告和功能建议
- <mutt-dev-request@mutt.org> — 开发小组的邮件列表

注意：

所有发往mutt-announce的邮件都将自动发往mutt-users，所以你不必两者都订阅。

3. 从哪里得到Mutt

Mutt 发行版可从 <ftp://ftp.mutt.org/mutt/> 下载。关于镜像站点列表，请查看<http://www.mutt.org/download.html>。

4. Mutt在线资源

Bug跟踪系统

Mutt的官方bug跟踪系统在<http://dev.mutt.org/>里面。

Wiki

Wiki地址（非官方）：<http://wiki.mutt.org/>。

IRC

IRC用户社区，irc.freenode.net，频道#mutt。

新闻讨论组

comp.mail.mutt。

5. 捐助Mutt

有很多方式可以为Mutt项目贡献一份力量。

对于那些新用户，如果老用户能与他们谈论Mutt，讨论问题，分享技巧，将使新用户受益匪

浅。

如果有人将Mutt翻译成其他语言，Mutt会非常感激。Mutt的开发者也一直在寻找熟练的翻译者以帮助提高和维护原来的翻译。

想为新功能提供代码以及为修复bug提供补丁，请查看开发者网页<http://dev.mutt.org/>了解更多详情。

6. 书写习惯

本节列出了本手册使用的书写习惯。见表1.1 “特殊条目书写习惯”。

表1.1.特殊条目书写习惯

条目	含义
printf(3)	Unix man page，运行man 3 printf
<PageUp>	键盘上已有定义的键
<create-alias>	Mutt中定义的函数
^G	Control + G组合键
\$mail_check	Mutt配置选项

运行以下的命令获得具体的说明：

```
mutt -h(译者注：原手册为"-v")
```

在命令用法说明中，（“{}”）则是指一组选项，它们中至少使用一个，方括号（“[]”）表示可选参数，其中有三个点的表示参数可以重复使用任意次。

7.版权声明

Mutt的版权1996-2009，为Michael R. Elkins<me@mutt.org>等所有。

此软件为自由软件，你可以在自由软件基金会发布的GNU通用公共许可证（可以是第2版或者之后的任意版本）下的重新分配以及修改它。

这项计划的发布是希望它能有所用途，但它没有任何担保，甚至连必要的商业或其他用途的担保都没有。更多细节请参阅GNU通用公共许可证。

你应该已经收到一份随软件一起发布的GNU通用公共许可证;如果还没有，写信给自由软件基金会,51 Franklin Street,Fifth Floor,Boston,MA 02110-1301,USA。

第2章.使用入门

目录

- [1. 基本理念](#)
- [2. 在菜单中移动](#)
- [3. 使用编辑器](#)

- [3.1 简介](#)
- [3.2 输入记录](#)

[4. 阅读邮件-Index和Pager](#)

- [4.1 Index](#)
- [4.2 Pager](#)
- [4.3 Threaded模式](#)
- [4.4 其它功能](#)

[5. 发送邮件](#)

- [5.1 简介](#)
- [5.2 编辑邮件头](#)
- [5.3 加密/签名邮件](#)
- [5.4 通过mixmaster发送匿名邮件](#)
- [5.5 发送format=flowed邮件](#)

[6. 转发和bounce邮件](#)

[7. 暂缓发送邮件](#)

本节将简要说明如何使用mutt。其它的许多功能将在这本手册其它地方描述，还有更多的信息可以在Mutt的FAQ以及其它网页中找到。见[Mutt的主页](#)可以了解更多信息。

本节列出的键绑定是发行包中默认的，你的系统管理员可能已经改变了默认配置，你可以在任何菜单中输入“?”来查看当前的键绑定。

首先，你需要在命令行键入mutt来启动mutt，有不同的命令行选项可用，在mutt的man page或者参考上查看。

1. 基本理念

Mutt是一个基于文本的应用程序，通过基于行 / 条目 / 页的菜单与用户交互。基于行的菜单就是所谓的“index”菜单（列出当前目录的所有邮件）或“alias”菜单（允许你选择收件人的列表）。基于页的菜单就是“pager”（一次显示一封邮件）或“帮助”菜单，“帮助”菜单列出了所有可用的键盘绑定。

用户界面包括一个位于顶部的帮助栏、以及菜单内容和命令行，菜单的下方附有一个状态栏。命令行用于显示提示和错误信息，也用于提示符和输入交互命令。

由于Mutt允许用户定制几乎所有的键盘绑定，所以“函数”可以手动（使用命令行）或使用宏来运行。宏允许用户将一系列的命令绑定到一个或几个按键上，从而避免一次又一次的重复动作。

许多命令（如储存或复制邮件到其它目录）可以适用于单个邮件，也可以用于一组（“被标记”）邮件。为了快速、高效地选择邮件，Mutt提供了一套丰富的邮件模式（如收件人，发件人，邮件体，发送/接收日期，等等），通过使用逻辑与、逻辑或以及逻辑非操作符，它们可以

组合成复杂的表达式。这些模式也可以用于（比如）搜索邮件或限制index只显示匹配的邮件。

Mutt支持一个称之为“hook”的概念，它允许用户在特定的条件下，比如进入一个目录、开始一个新邮件或者回复一封邮件，执行任意的命令和功能。这些hook可用于高度定制Mutt的行为，包括管理多个帐户，定制某一个文件夹的显示，或者是让每个文件夹自动打包，等等。

2.在菜单中移动

表2.1.“通用导航键”显示了在所有菜单中都可用的导航键。

表2.1.通用导航键

按键	函数	描述
j或<Down>	<next-entry>	移动到下一个条目
k或<Up>	<previous-entry>	移动到上一个条目
z或<PageDn>	<page-down>	转到下一页
Z或<PageUp>	<page-up>	转到上一页
=或<Home>	<first-entry>	跳转到第一个条目
*或<End>	<last-entry>	跳转到最后一个条目
q	<quit>	退出当前菜单
?	<help>	列出当前所有键绑定

3.使用编辑器

3.1.简介

Mutt中有一个内置的行编辑器可以输入文本，如电子邮件地址或文件名。其按键的操作方式非常类似Emacs。表2.2.“常用行编辑器按键”列出了其默认地键绑定。

表2.2.常用行编辑器按键

按键	函数	描述
^A 或 <Home>	<bol>	移动到行首
^B 或 <Left>	<backward-char>	向后移动一个字符
Esc B	<backward-word>	向前移动一个单词
^D 或 <Delete>	<delete-char>	删除光标所在的字符
^E 或 <End>	<eol>	移动到行末
^F 或 <Right>	<forward-char>	向前移动一个字符
ESC F	<forward-word>	向后移动一个单词
<Tab>	<complete>	补全文件名或别名
^T	<complete-query>	查询外部地址来补全地址

^K	<kill-eol>	删除至行末
Esc d	<kill-eow>	删除至单词末尾
^W	<kill-word>	删除从光标到单词开始
^U	<kill-line>	删除整行
^V	<quote-char>	引用随后输入的字符
<Up>	<history-up>	在输入记录中上翻一条记录
<Down>	<history-down>	在输入记录中下翻一条记录
<BackSpace>	<backspace>	删除光标前面的字符
Esc u	<upcase-word>	转换单词为大写
Esc l	<downcase-word>	转换单词为小写
Esc c	<capitalize-word>	大写单词首字母
^G	n/a	取消
<Return>	n/a	完成

你可以使用命令bind重新映射 *编辑器* 功能。例如，要使<Delete>键删除光标前面的字符，而不是光标所在的字符，你可以使用：

```
bind editor <delete> backspace
```

3.2.输入记录

Mutt保存了编辑器的输入记录，所有记录被保存在变量\$history中，也可以用\$history_file将记录持久地保存在一个外部文件中。你可以使用命令<history-up>和<history-down>在这些记录中来回选择。

Mutt保存了几类不同的记录列表，分别是：

- muttrc命令
- 地址和别名
- shell命令
- 文件名
- 模式
- 其它

Mutt会自动从记录中滤除重复的条目。与一些shell的行为相似，它还能忽略以一个空格开始的条目。The latter feature can be useful in macros to not clobber the history's valuable entries with unwanted entries。

4.阅读邮件-Index和Pager

和其他许多的邮件客户端类似，在Mutt中，可以在两种模式中浏览邮件。第一种是列出邮箱中邮件的索引，这就是“index”；第二种模式是显示邮件的内容。这是“pager”。

以下的几个小节描述了每种方式的一些特性。

4.1.Index

表2.3列出了在index菜单中常用的浏览和管理邮件的按键。通过\$index_format变量可以设置邮件在index菜单中的显示格式。

表2.3 index菜单常用按键

按键：	内容简介
c	转到其它信箱
Esc c	以只读模式转到其它信箱
C	复制当前邮件到其它信箱
Esc C	解码邮件并复制到文件夹
Esc s	解码邮件并移动到文件夹
D	删除模式匹配的邮件
d	删除当前邮件
F	标记为重要
l	显示模式匹配的邮件
N	标志新邮件
o	改变当前排序方法
O	反序改变当前排序方法
q	保存更改并退出
s	保存消息
T	标记模式匹配的邮件
t	标记邮件
Esc t	标记整个thread
U	取消删除模式匹配的邮件
u	取消删除邮件
v	查看附件
x	放弃更改并退出
<Return>	显示信息
<Tab>	跳转到下一条新的或未读邮件
@	显示作者的e-mail地址

\$	保存更改
/	搜索
ESC /	反向搜索
^L	清除并重画屏幕
^T	取消标记模式匹配的邮件

除了发件人和主题，在邮件标号旁还会列出邮件的简要信息。表2.4“邮件状态标志”中的0个或更多的“状态标志”可能会显示。其中一些标志可以使用<set-flag>和<clear-flag>（默认绑定到键“w“和”W“）开启或关闭，

此外，表2.5，“邮件收件人标志”中的标志反映的是邮件的收件人信息。这些可以通过\$to_chars变量配置。

表2.4.邮件状态标志

标志	描述
D	邮件被标记为删除
d	邮件附件标记为删除
K	包含的PGP公钥
N	新邮件
O	旧的邮件
P	邮件用过PGP方式加密
r	邮件已回复
S	邮件被签名，并且签名已验证
s	邮件被签名
!	邮件被标记为重要
*	邮件标记

表2.5 。邮件收件人标志

标志	描述
+	邮件是发给你唯一一个人的
T	邮件是发给你的，但是也发送或者cc给其它人
C	信息是cc给你的
F	信息是由你发出的
L	邮件是发送到一个已订阅的邮件列表地址

4.2 Pager

默认情况下，Mutt使用其内置的pager来显示邮件的内容。pager非常类似Unix下的*less*程序，虽

然它们功能不一样。

表2.6.pager菜单常用按键

按键	描述
<Return>	向下移动一行
<Space>	显示下一页（或下一封邮件，如果已到邮件末尾）
-	向上翻一页
n	显示下一个匹配项
S	跳过引用文本
T	切换是否显示引用文本
?	帮助菜单（查看键绑定）
/	使用正则表达式搜索
ESC /	使用正则表达式反向搜索
\	突出显示搜索的匹配项
^	跳转到邮件顶部

除了表2.6“pager菜单常用按键”中列出的键绑定外，许多index菜单中的键绑定在pager中也是可用的，如<delete-message>或<copy-message>（相比使用外部pager，这是一个优点）。

此外，内置pager还支持其他一些高级特性。其中之一,它接受“标准“nroff序列并能将其翻译成粗体或下划线。These sequences are a series of either the letter、backspace(^H)、the letter again for bold or the letter, backspace, “ _ ” for denoting underline。如果你的终端支持，Mutt将尝试分别以粗体和下划线来显示。如果不的话,你可以使用粗体、下划线、颜色对象来为它们指定颜色或mono属性。

此外，内置pager支持ANSI转义序列。mutt会将它们翻译成对应的颜色和字符属性。Mutt支持的序列有：

```
\e[Ps;Ps;..Ps;m
```

Ps可以是表2.7“ANSI转义序列”中所示的任一种转义码。

表2.7.ANSI转义序列

转义码	描述
0	关闭所有属性
1	粗体
4	下划线
5	闪烁
7	Reverse video on
3<color>	前景颜色是<color> （见表 2.8 “颜色序列”）

4<color>	背景颜色是<color> （见表 2.8 “颜色序列”）
----------	------------------------------

表2.8.颜色序列

颜色代码	颜色
0	黑色
1	红色
2	绿色
3	黄色
4	蓝色
5	红紫色
6	蓝绿色
7	白色

Mutt使用这些属性来处理text/enriched格式的邮件，通过外部的脚本他们也可以用来达到高亮显示的目的。

注意：

如果你改变了显示的颜色，比如改变了xterm的color2，那么此颜色将代替绿色显示。

注意：

请注意pager中搜索采用的是正则表达式，这和index中使用的更复杂的模式搜索不太一样。这是因为pager只需实现简单的文本搜索，而index提供了邮件几种属性的逻辑运算。

4.3 Threaded模式

当信箱以thread方式排序的时候，在index和pager模式中有一些额外的功能可用，如表2.9“thread模式常用键”所示。

表2.9.thread模式常用键

按键	功能	描述
^D	<delete-thread>	删除当前thread中的所有邮件
^U	<undelete-thread>	取消删除当前thread中的所有邮件
^N	<next-thread>	跳转到下一个thread的开始处
^P	<previous-thread>	跳转到上一个thread的开始处
^R	<read-thread>	将当前thread标记为已读
Esc d	<delete-subthread>	删除当前子thread中的所有邮件

Esc u	<undelete-subthread>	取消删除当前子thread中的所有邮件
Esc n	<next-subthread>	跳转到下一个子thread的开始处
Esc p	<previous-subthread>	跳转到上一个子thread的开始处
Esc r	<read-subthread>	将当前子thread标记为已读
Esc t	<tag-thread>	切换标记当前thread
Esc v	<collapse-thread>	切换折叠当前thread
Esc V	<collapse-all>	切换折叠所有thread
P	<parent-message>	跳转到父邮件

注意:

折叠一个thread会只显示这个thread的第一封邮件，隐藏其它邮件。thread包含了很多邮件时，你在屏幕上只会看到几个thread，此时这个功能就非常有用。见\$index_format中的序列%M。例如，你可以在\$index_format中使用“%?M?(#%03M)&(%4l)?”来显示隐藏邮件的数量，如果thread是折叠的。

参见：\$strict_threads

4.4.其它功能

*index*和*pager*菜单还有一些有趣的功能：

<create-alias> （默认： a）

根据目前邮件（或输入一个新的）创建一个新的别名。一旦完成编辑，一个别名将添加到\$alias_file变量指定的文件中，以便将来使用。

注意:

Mutt启动时不会读变量\$alias_file指定的文件，所以你必须用命令source来指定文件。

<check-traditional-pgp> （默认： Esc P）

此功能会在当前邮件中搜索以“传统”PGP方式（即没有适当的MIME标记）签名或加密的内容。从技术上来说，这个功能会暂时改变包含PGP数据的邮件体的MIME类型，这和功能[<edit-type>](#)的效果类似。

<edit> （默认： e）

这个命令（*index*和*pager*中可用）让你可以修改当前的原始邮件。在你完成编辑后，改变的邮件会附加到当前文件夹中，原始邮件将被标记为删除;如果邮件没有改变，那它就不会被替换。

`<edit-type>`（默认：在附件菜单、index和pager菜单中为^E;在编辑菜单中为^T）

这个命令可以临时修改附件类型来解决诸如错误字符集参数的问题。当从index或pager启用此功能时，你可以修改最顶层的附件的类型。在附件菜单中，你可以修改任何附件的类型。这个改变不是永久的，在切换文件夹后将丢失。

请注意，这个命令也可以在编辑菜单中使用，用来调整你要发送的附件的属性。

`<enter-command>`（默认：“:”）

此命令是用来执行配置文件muttrc中可用的任何命令。通常用于检查变量的设置, 或者临时与宏一起使用来改变设置。

`<extract-keys>`（默认：^K）

此命令从当前邮件或被标记的邮件中提取PGP公钥并将其添加到你的PGP公钥环中。

`<forget-passphrase>`（默认：^F）

此命令从存储中清除密码。当你拼错密码时，这是非常有用的。

`<list-reply>`（默认：L）

回复到通过命令lists或subscribe订阅的邮件列表，但如果变量\$`honor_followup_to`为yes的话，还会参考邮件头的Mail-Followup-To字段。使用此功能回复一个邮件列表可以避免向邮件的作者发出重复的副本。

`<pipe-message>`（默认：l）

将当前的或被标记的邮件pipe给外部的Unix命令。变量\$`pipe_decode`，\$`pipe_split`，\$`pipe_sep`和\$`wait_key`决定这一功能具体的行为。

`<resend-message>`（默认：Esc e）

Mutt使用当前邮件作为新邮件的模板。这个功能最好的描述是“recall from arbitrary folders”。它可以方便地转发MIME邮件，同时保持原始邮件的结构。请注意，一些邮件头字段由变量\$`weed`决定。

此功能在附件菜单也可用。你可以轻松地重新发送邮件，也包含以message/rfc822作为邮件体格式bounce邮件。

`<shell-escape>`（默认：!）

执行外部的Unix命令。变量\$`wait_key`可以用来控制是否让Mutt在命令返回时等待用户按键（为了让用户有时间阅读命令的输出结果），这取决与命令的返回状态。如果没有给出一个外部命令，将进入一个互交的shell。

`<toggle-quoted>`（默认：T）

pager在显示邮件体使用变量\$quote_regexp来检测被引用的文本。此功能是显示引用文本的开关。在对引文有兴趣，或有大量引文时，可以使用这个功能。

<skip-quoted>（默认：S）

在内置pager中，这个功能可以跳过引文，到下一处不是引文的地方。

5.发送邮件

5.1.简介

表2.10“发送邮件常用按键”列出的按键可在*index*和*pager*菜单中开始一封新邮件时使用。

表2.10.发送邮件常用按键

按键	功能	描述
m	<compose>	撰写新邮件
r	<reply>	回复发件人
g	<group-reply>	回复所有收件人
L	<list-reply>	回复到邮件列表
f	<forward>	转发邮件
b	<bounce>	bounce (remail)邮件
ESC k	<mail-key>	给某人发送一个PGP的公钥

bounce 邮件将原样发送邮件到你指定的收件人。转发邮件允许你添加注释或修改你要转发的邮件。这些功能将在下一章“转发和bounce邮件”作进一步的讨论。

之后Mutt会进入*编辑*菜单，并提示你输入“To:”字段。下一步，提示你键入“Subject:”字段，如果你是回复或转发邮件Mutt已经给你提供了一个预设的主题。\$askcc，\$askbcc，\$autoedit，\$bounce，\$fast_reply，和\$include这些变量将决定Mutt怎样处理这些问题。

Mutt会自动启动你在\$editor定义的编辑器来编辑邮件正文。如果设置了\$edit_headers变量，邮件头将出现在邮件正文的前面。任何你回复的邮件会和变量\$attribution，\$indent_string，\$post_indent_string一起按顺序加入邮件。转发邮件时，如果\$mime_forward为no，你转发邮件的副本将包含进去。如果你指定了\$signature，它将会被附加到邮件中。

当你编辑完成邮件正文，你将返回*编辑*菜单，编辑菜单提供了一些功能可用来修改、发送、推迟发送邮件，如表2.11“编辑菜单常用按键”所示。

表2.11.编辑菜单常用按键

按键	功能	描述
a	<attach-file>	添加文件作为附件
A	<attach-message>	附加邮件作为附件

ESC k	<attach-key>	添加一个PGP公钥
d	<edit-description>	编辑附件的描述
D	<detach-file>	删除文件
t	<edit-to>	编辑“To”字段
ESC f	<edit-from>	编辑“From”字段
r	<edit-reply-to>	编辑“Reply-To”字段
c	<edit-cc>	编辑“Cc”字段
b	<edit-bcc>	编辑“Bcc”字段
y	<send-message>	发送邮件
s	<edit-subject>	编辑主题
S	<smime-menu>	S/MIME选项
f	<edit-fcc>	指定“Fcc”信箱
p	<pgp-menu>	PGP选项
P	<postpone-message>	推迟发送邮件
q	<quit>	取消发送
w	<write-fcc>	保存邮件到某个文件夹
i	<ispell>	拼写检查（如果你的系统支持）
^F	<forget-passphrase>	从存储中清除密码

编辑菜单也被用来编辑附件，无论它是文件还是其它邮件。<attach-message>功能将会提示选择一个文件夹，然后你在该文件夹中标记你的要添加的邮件，它们将作为附件发送出去。

注意：

请注意在这个文件夹中一些操作像编辑新邮件，转发、回复邮件等是不被允许的。变量 \$status_format 中的状态指示器 %r 将会变为“A”，表示你处在 attach-message 模式。

5.2.编辑邮件头

在编辑邮件头时，有一些邮件头不会被包含在发送出去的邮件中，这称之为伪邮件头。

5.2.1.Fcc:伪邮件头

如果你指定”Fcc: 文件“作为一个邮件头，Mutt会保存邮件到文件，就像你在编辑菜单使用了<edit-fcc>功能一样。

5.2.2.Attach:伪邮件头

你也可以通过在邮件头中指定”Attach: 文件[说明]“的方式添加附件，其中”文件“是需要添加

的附件，”描述“是一个可选的字符串，用于说明所添加的文件。

5.2.3.Pgp:伪邮件头

如果你想使用PGP，可以指定

```
Pgp: [ E | S | S <id> ]
```

“E”表示加密，“S”表示签名，“S<id>”表示使用给定的密钥来签名，这个会长期地影响变量\$pgp_sign_as的值。

5.2.4.In-Reply-To:邮件头

回复邮件时，*In-Reply-To*:字段包含了你所回复邮件的Message-Id。如果删除这个字段，则不会生成*References*:字段，*References*:字段允许你创建一个新的thread邮件，例如，创建一封到邮件列表的新邮件，而不用输入列表的地址。

5.3.加密/签名邮件

如果你使用PGP或S/MIME来加密邮件，当邮件被发送前，mutt它会引导你选择一个密钥来处理邮件。如果有一个密钥的user ID和收件人的地址匹配，那么Mutt就不会有任何提示，直接使用这个密钥。不过，还会可能出现某些情况，有几个密钥；未验证的user ID；或者没有找到匹配的密钥。

在这些情况下，mutt会出现一个包含密钥清单的菜单，你可以从中选择一个。当退出这个菜单，或找不到任何匹配的密钥时，mutt会提示你输入user ID。你可以使用按键^G来取消。当你这样做了，mutt会返回到编辑菜单。

一旦你选择了密钥，发送时就会使用选择的公钥加密邮件。

密钥清单中的条目的大多数字段（也见\$pgp_entry_format）都是具有明显意义的。下面是关于其标志字段、功能字段和有效性字段的描述。

标志序列（%f）将被扩展为表2.12，“PGP菜单标志”中的一项。

表2.12.PGP菜单标志

标志	描述
R	密钥已经被回收，无法再使用。
X	密钥已经过期，无法使用。
d	你已经标记密钥为禁用。
c	There are unknown critical self-signature packets。

功能字段（%c）会被扩展成2个字符的序列，代表密钥的功能。第一个字符描述了密钥的加密能力：减号（“-”）表示密钥不能用于加密。符号点（“.”）表示它被标记为用于某个user ID的签名，不过也许能够用于加密。字母“e”表示这个密钥可以用于加密。

第二个字符描述了密钥的签名能力。同样的，减号“-”表示“不用于签名”，符号点“.”表示改密钥被标记为用于某个user ID的加密，“s”表示可以用于加密。

最后，有效性字段（%t）描述了user-id的有效性。问号（“?”）表示未定义有效性，减号（“-”）表示其被标记为一个不被信任的交往，空白符表示一个不完全信任的交往，加号（“+”）表示完全有效。

5.4 通过mixmaster发送匿名邮件

你可以编译mutt让它可以同Mixmaster（一个匿名转投系统）很好的协作。Mixmaster允许你通过一条转投系统链匿名发送邮件。Mixmaster支持mutt的版本是2.04（beta 45似乎是最新的）和2.03。早期的版本或后来所谓的3 betas（最新的大概是2.9b23）都不支持mutt。

要使用它，你须遵守一定的规则。最重要的是，你不能使用Cc和Bcc字段。告诉mutt要使用mixmaster，你必须先在编辑菜单使用mix功能选择一条转投系统链。

选择链的菜单分为两部分。在上半（大的）部分中，会列出一个你可以使用的转投系统的清单。在下半部分中，是当前选择的转投系统链。

你可以使用功能<chain-prev>和<chain-next>来浏览链，默认情况下，这两个功能被绑定在向左、向右箭头和h、l键上（思考一下vi的键盘绑定）。在当前位置插入一个转投系统，使用功能<insert>。将一个转投系统附加到当前链，使用<select-entry>或<append>。你也可以使用相应的功能从链中删除一个条目。最后，要放弃更改，直接退出菜单，否则使用功能<accept>（默认为按键Return）应用它们。

请注意，不同转投系统有不同的能力，菜单中的序列%c有所描述（见\$mix_entry_format）。最重要的就是“middleman”，表示为“M”：这意味着这个转投系统不能用于链的最后一个结点，只能转发邮件到其它的mixmaster转投系统。更详细地介绍，请见mixmaster的文档。

5.5. Sending format=flowed messages(译者注：次节暂未翻译)

5.5.1. Concept

format=flowed-style messages (or f=f for short) are text/plain messages that consist of paragraphs which a receiver's mail client may reformat to its own needs which mostly means to customize line lengths regardless of what the sender sent. Technically this is achieved by letting lines of a “flowable” paragraph end in spaces except for the last line.

While for text-mode clients like mutt it's the best way to assume only a standard 80x25 character cell terminal, it may be desired to let the receiver decide completely how to view a message.

5.5.2. Mutt support

Mutt only supports setting the required format=flowed MIME parameter on outgoing messages if the [\\$text_flow](#) variable is set, specifically it does not add the trailing spaces.

After editing the initial message text and before entering the compose menu, mutt properly space-stuffes the message. *Space-stuffing* is required by RfC3676 defining format=flowed and means to prepend a space to:

- all lines starting with a space
- lines starting with the word “From” followed by space
- all lines starting with “>” which is not intended to be a quote character

NOTE:

Mutt only supports space-stuffing for the first two types of lines but not for the third: It is impossible to safely detect whether a leading > character starts a quote or not. Furthermore, Mutt only applies space-stuffing *once* after the initial edit is finished.

All leading spaces are to be removed by receiving clients to restore the original message prior to further processing.

5.5.3. Editor considerations

As mutt provides no additional features to compose f=f messages, it's completely up to the user and his editor to produce proper messages. Please consider your editor's documentation if you intend to send f=f messages.

Please note that when editing messages from the compose menu several times before really sending a mail, it's up to the user to ensure that the message is properly space-stuffed.

For example, *vim* provides the `w` flag for its `formatoptions` setting to assist in creating f=f messages, see `:help fo-table` for details.

6.转发和bounce邮件

bounce和转发邮件可以让你发送现有的邮件到指定的收件人。bounce邮件会将邮件完完整整地发送到指定地点，就好像他们被包含在原始邮件的Bcc字段中一样。转发邮件允许你在发送前修改邮件（例如，添加些自己的看法）。bounce邮件是通过功能<bounce>完成的，而转发邮件是通过功能<forward>完成的，它们分别绑定到按键“b”和“f”。

转发将原始邮件的内容放到新邮件的邮件体中（开始和结尾处有说明）或原始邮件以MIME文件的形式包含，取决于变量`$mime_forward`的设定。附件的解码，如在pager中一样，可以通过变量`$forward_decode`和`$mime_forward_decode`分别控制。理想的转发格式是将其放入正文，因此，`$mime_forward`是`quadoption`类型，一般设为“ask-no”。

如果`$mime_forward`为no，变量`$weed`的设置会影响包含邮件头的情况。

编辑转发的邮件和发送，回复邮件一样。

7.暂缓发送邮件

经常想要暂缓发送一封你已经开始编辑的邮件。在编辑菜单中调用了<postpone-message>功能后，你的邮件的正文和附件都将存储在`$postponed`变量所指定的信箱中。你可以重新调出邮件，即使你已经退出，一段时间后再重新启动了Mutt。

一旦邮件被暂缓了，有以下几种方法可以继续它。在命令行中你可以使用选项“-P”，或者当你从 *index* 或 *pager* 编辑新邮件时，Mutt 将询问你是否发送被暂缓的邮件，如果有的话。如果有多封邮件被暂缓，会弹出一个 *postponed* 菜单，你可以选择要继续的邮件。

注意：

如果你暂缓了回复的邮件，邮件回复设置只有在发送后生效。同样地，你必须在与你回复邮件相同的文件夹，以便邮件的状态被更新。

参考 `$postpone`。

第3章.配置文件

目录

- [1.配置文件的位置](#)
- [2.配置文件的语法](#)
- [3.地址组](#)
- [4.定义/使用别名](#)
- [5.更改默认键盘绑定](#)
- [6.定义别名的字符集](#)
- [7.根据信箱设置变量](#)
- [8.键盘宏](#)
- [9.使用颜色和mono属性](#)
- [10.显示邮件头](#)
- [11.后备地址](#)
- [12.邮件列表](#)
- [13.使用多个spool信箱](#)
- [14.监视新邮件](#)
- [15.自定义邮件头](#)
- [16.指定默认保存邮箱](#)
- [17.指定默认的Fcc信箱，编写邮件时](#)
- [18.同时指定save-hook和fcc-hook的默认值](#)
- [19.根据收件人改变设置](#)
- [20.格式化邮件时更改设置](#)
- [21.选择收件人的密钥](#)
- [22.添加按键序列到键盘缓冲区](#)
- [23.执行功能](#)
- [24.邮件评分](#)
- [25.垃圾邮件检测](#)
- [26.设置和查询变量](#)
 - [26.1 命令](#)
 - [26.2 用户自定义变量](#)
- [27.从另一个文件读入配置命令](#)
- [28.移除hook](#)
- [29.格式字符串](#)
 - [29.1 基本用法](#)

1. 配置文件的位置

默认配置(或“首选项”)可以让mutt立即使用，但你通常还是需要对Mutt进行配置以适合你自己的习惯。当Mutt启动，它会先尝试读取“系统”的配置文件（被系统管理员所设置），除非启动Mutt时指定了选项“-n”。系统的配置文件通常是/usr/local/share/mutt/Mutttrc或者/etc/Mutttrc目录。然后Mutt将在你的home目录查找一个叫.mutttrc的文件。如果此文件不存在并且你的home目录下有一个叫.mutt的子目录，Mutt就会尝试加载文件.mutt/mutttrc。

.mutttrc是一个放置配置命令的文档。

此外，Mutt可以根据当前Mutt版本指定特定的配置文件而不使用上面提到的默认配置文件。举例来说，如果你的系统中有一个名为Mutttrc-0.88的文件在系统配置目录，并且你运行的是版本为0.88的Mutt，那么这个文件将代替Mutttrc文件被读取。对于用户配置文件也是一样的道理，如果home目录有一个文件.mutttrc-0.88.6，当你运行0.88.6的Mutt时，它将被Mutt读取，而不是默认的.mutttrc文件。可以使用命令行选项“-v”或在index菜单使用功能show-version（默认：V）来查看当前版本号。

2. 配置文件的语法

配置文件包含了一系列命令，文件的每一行可以包含一个或多个命令，一行放入多个命令时，必须用分号（;）隔开。

例3.1. 一行多条配置命令

```
set realname='Mutt user' ; ignore x-
```

（“#”）被用于“注释”的字符。你可以用它来注释你的配置文件。从“#”开始到行末尾的所有文本都会被忽略。例如：

例3.2. 注释配置文件

```
my_hdr X-Disclaimer: Why are you listening to me? # This is a comment
```

单引号（'）和双引号（"）用于引用包含空格或其他特殊字符的字符串。两者的区别类似许多常见的shell程序，即单引号屏蔽了所有特殊字符的作用，引用的基本的字符串（里面的变量以及用反斜杠转义的字符[见下段]将不被解释），而双引号引用的字符串都将被重新解释。例如，双引号内的符号backticks（`）将被解释，而单引号内的就不会。

反斜杠“\”会转义下一个字符，就如bash和zsh这样的shell一样。例如，如果想要把双引号“”放到字符串里，你可以使用反斜杠“\”使下一个字符被当成基本字符，而非需要被解释的字符。

例3.3. 在配置文件中转义引号

```
set realname="Michael \"MuttDude\" Elkins"
```

“\\”是指插入基本字符“\”。同C语言中的表示方法一样，“\n”和“\r”分别代表换行符和回车符。

行尾的“\”可以用来分割命令成多行，但断点不能出现在命令名称的中间。

在配置文件中还可以使用Unix命令的输出，方法是将命令放入backticks(`)内。例如：

例3.4. 在配置文件中使用外部命令的输出

```
my_hdr X-Operating-System: `uname -a`
```

在这一行被解析前，Unix命令“uname -a”的输出将取代“`uname -a`”。

注意：

配置文件是面向行的，因此Unix命令的输出只有第一行是有效的。

环境变量和Mutt变量都可以通过“\$”加变量名来访问。例如：

例3.5. 在配置文件中使用环境变量

```
set record=+sent_on_${HOSTNAME}
```

会导致 Mutt 将已发送的邮件保存到名为“sent_on_kremvax”的文件夹中，如果环境变量 \$HOSTNAME 设置为“kremvax”（细节见 \$record）。

Mutt 扩展变量是在它被定义的时候，而不是它被使用时。变量被赋值后如果等号右边的值发生改变，该变量不会受到影响。

下一段描述了 Mutt 的命令。完整的清单，参考命令索引。

所有的配置文件都被默认为是以当前 locale 指定的字符集来编码的，由变量 \$charset 来记录，这个变量没有默认值，它的值是在 mutt 启动的时候确定的。如果一个配置文件使用与 locale 相同的字符集来编码，那么就应该使用变量 \$config_charset：在之后的所有行都将用 \$charset 代替 \$config_charset 重新编码。

如果可能的话，应该避免使用这种机制。因为这会带来以下的一些问题：

- These variables should be set early in a configuration file with \$charset preceding \$config_charset so Mutt know what character set to convert to.
- 如果设置了 \$config_charset，那么每一个配置文件都应该使用这个字符集来编码，因为这它是个全局变量而不是只对一个配置文件有效的。
- Because Mutt first recodes a line before it attempts to parse it, a conversion introducing question marks or other characters as part of errors (unconvertable characters, transliteration) may introduce syntax errors or silently change the meaning of certain tokens (e.g. inserting question marks into regular expressions).

3. 地址组

用法：

```
group [ -group name ... ] { -rx expr ... | -addr expr ... }
```

```
ungroup [ -group name ... ] { * | -rx expr ... | -addr expr ... }
```

group用于将地址或正则表达式添加到指定组。group命令中的不同类型的参数可以按任意顺序使用。选项-rx和-addr决定其后的字符串（不能以连字符开头）被解释为什么：一个正则表达式还是一个电子邮件地址。

通过使用选项-group，这些地址组也可以使用命令alias、lists、subscribe、alternates隐式的创建。

一旦定义之后，这些地址组可以在模式搜索中使用（查找匹配一个组的邮件）。

ungroup是用来移除一个组中的地址或正则表达式。其语法和命令group类似，特殊字符*可以用来清空一个组内的所有内容。

4.定义/使用别名

用法：

```
alias [ -group name ... ] key address [ address ... ]
```

在你给某人发邮件时，要记住或者输入对方的地址是非常麻烦的。Mutt允许您创建“别名”，将完整地址与一个很短很好记住的字符串关联。

注意：

如果你将一个别名与多个地址关联，你必须用逗号（“,”）分开每个地址。

alias的可选参数“-group”用于将地址添加到指定的group。

要删除别名（“*”指所有的别名）：

```
unalias [ -group name ... ] { * | key ... }
```

```
alias muttdude me@cs.hmc.edu (Michael Elkins)
alias theguys manny, moe, jack
```

与其他邮件客户端不同，Mutt不需要将别名定义到一个特殊的文件。alias命令可以出现在任何一个文件中，只要此文件被source。因此，你可以有多个别名文件，或者你也可以把你所有的别名定义到muttrc中。

另一方面，<create-alias>功能只对一个文件有效，即变量\$alias_file所指向的文件（默认为~/.muttrc）。这个文件没有任何特殊的地方，从这个意义上讲，Mutt可以将别名添加到任何文件中，同时为了使新添加的别名立即生效你需要显示地source这个文件。

例如：

例3.6. 定义外部别名文件

```
source /usr/local/share/Mutt.aliases
source ~/.mail_aliases
set alias_file=~/.mail_aliases
```

你可以在任何 Mutt 提示输入地址的地方使用别名，如 *To:* 或 *Cc:* 提示符。如果你设置了 `$edit_headers` 变量，你也可以在相应的邮件头中使用别名。

此外，在使用别名的时候，您可使用 `tab` 键补全为写完的别名。如果有多个匹配别名，Mutt 会显示一个所有匹配的菜单。如果要显示所有别名的列表，你就不能在已经输入了半截别名的情况下按下 `tab` 键，你应该在提示符的开始处或一个逗号之后敲击 `tab` 键。

在别名菜单中，你可以使用 `select-entry` 键（默认：<回车>）选择尽可能多的别名，并使用 *退出* 键（默认：q）返回地址提示栏。

5.更改默认键盘绑定

用法：

`bind map key function`

此命令允许你更改默认的键盘绑定方式（使用一个按键执行操作）。

map 指定键绑定在哪个菜单上有效。可以指定多个 `map`，用逗号（不允许有空格）隔开。目前可以定义的 `map` 有：

`generic`

这不是一个真实的菜单，它算是其他菜单（除了 `pager` 和编辑器模式以外）的一个后备菜单。如果一个按键在菜单中没有找到相应的绑定，`mutt` 就会在这个菜单中来查找。这允许你将一个按键绑定到多个菜单，而不用在每一个菜单中都绑定一次。

`alias`

别名菜单是你在 `muttrc` 中定义的别名的列表。它将完整的收件人地址与一个很短的名字相关联。

`attach`

附件菜单列出收到的邮件中的附件。

`browser`

浏览菜单用于浏览本地的文件，以及列出你所有的收件箱。

`editor`

编辑器是基于行的用户输入文本数据的编辑器。

index

index菜单是邮箱中邮件的列表。

compose

编辑菜单是发送新邮件时所使用的画面。

pager

pager菜单是用来显示邮件/附件的内容以及帮助列表。

pgp

pgp菜单是选择用于加密邮件的OpenPGP密钥的菜单。

smime

smime菜单是选择用于加密邮件的OpenSSL证书的菜单。

postpone

postpone菜单和index菜单相似，用来重新调出已经开始编写并保存了的邮件。

query

查询菜单用于浏览\$query_command返回的结果。

mix

用于选择mixmaster转投系统的菜单(如果mutt编译进了Mixmaster支持)。

key是代表你想要绑定的按键（或按键序列）。要指定控制字符，使用序列\Cx，其中x是该控制字符的字母（例如，要指定control-A，使用“\Ca”）。请注意，x以及\C的大小写会被忽略，所以，\CA，\Ca，\cA和\ca都是相等的。另一种形式是以“\”开头后跟三个八进制的数字（例如\177相当于“\c?”）。此外，key也可以是一个符号名称，如表3.1“符号名称”所示。

表3.1. 符号名称

符号名称	含义
\t	制表符（tab）
<tab>	制表符（tab）
<backtab>	backtab / shift-tab
\r	回车
\n	换行符
\e	escape
<esc>	escape
<up>	上箭头
<down>	下箭头

<left>	左箭头
<right>	右箭头
<pageup>	上一页
<pagedown>	下一页
<backspace>	退格
<delete>	删除
<insert>	插入
<enter>	Enter
<return>	Return
<home>	Home
<end>	End
<space>	空格
<f1>	F1
<f10>	F10

*key*不需要放在引号中，除非它包含一个空格（“ “）或分号（”；“）。

*function*指定*key*被按下后执行何种动作。关于*function*的完整列表，见参考。特殊功能<noop>没有绑定到任何按键上。

6.定义别名的字符集

用法：

`charset-hook` *alias charset*

`iconv-hook` *charset local-charset*

命令`charset-hook`定义一个字符集的别名。如果一封邮件指定的是一个mutt不能识别的编码集，这个功能就可以让mutt正确地显示它。

命令`iconv-hook`为一个字符集定义一个系统特定名称。当系统的字符转换库对某个字符集使用了一个mutt不能识别的名称时，这个命令就非常有用了。

7.根据信箱设置变量

用法：

`folder-hook` *[!]regexp command*

通常想要根据不同的信箱来设置不同的变量。`folder-hook`命令所提供了让你可以执行任何配置命令的方法。*regexp*是一个正则表达式，所有与这个正则表达式匹配的邮箱被读取前，*command*会被执行。如果一个信箱匹配多个`folder-hook`，它们将按在muttrc中定义的顺序来依次执行。

注意:

如果你在模式之前使用了指向\$spoolfile的快捷符号“!”，你必须把它放到双引号或单引号内，以区别于表达式的逻辑非运算符。

注意:

设置在你离开信箱后不会被保存。例如，下面这条语句是在信箱被读取前执行改变排序方法的命令：

```
folder-hook mutt set sort=threads
```

然而，当你进入不同的信箱后，排序方法不会覆盖原先的值。要定义一个预设的命令，可以使用的符号“.”来匹配所有信箱，将该命令放在其它folder-hook命令之前，因为命令的执行顺序是其在配置文件中的顺序所决定的。下面的例子将所有信箱以date-sent方式排序，但名称包含“mutt”的信箱都以threads方式排序。

例3.7.根据信箱设置排序方式

```
folder-hook . set sort=date-sent
folder-hook mutt set sort=threads
```

8. 键盘宏

用法:

```
macro menu key sequence [ description ]
```

当你想用一個按鍵來執行了一系列的操作時，宏是非常有用的。當你在menu中按下key時，Mutt的表現會如你輸入了sequence一樣。所以，如果經常一次輸入一連串的命令來執行，那麼你可以創建一個宏用一個按鍵和很少的按鍵來執行這些命令。

menu定義宏在哪些[菜單](#)中有效，可以定義多個菜單，以逗號隔開，中間不能有空格。

key和sequence與鍵盤綁定中的用法類似，另外有幾點要指出的。首先是sequence中的控制字符也可以指定為^x。如果要使用符號（“^”），你需要這樣寫，“^^”。其次是要指定某個確定的按鍵如up或直接調用某個確定的功能，您應該使用格式<按鍵名稱>和<功能名稱>。按鍵名稱參見小節“更改默認鍵盤綁定”，功能名稱見索引。

在編寫宏時直接使用功能名稱的好處是可以忽略當前的鍵綁定，所以這樣可以不依賴用戶的鍵綁定設置。這大大增加了宏的健壯性和可移植性，而且這樣的宏可以很容易的被多個用戶所使用（比如系統的Muttrc文件）。

作為可選地設置，在sequence之後，你可以輸入一些描述性的話，這會顯示幫助菜單中。

注意:

帮助菜单的宏列表中，如果有超出窗口宽度的项会被截断，而不会折行。

9. Using color and mono video attributes(译者注： 此节暂未翻译)

Usage:

`color object foreground background`

`color { header | body } foreground background regexp`

`color index foreground background pattern`

`uncolor index { * | pattern ... }`

If your terminal supports color, you can spice up Mutt by creating your own color scheme. To define the color of an object (type of information), you must specify both a foreground color **and** a background color (it is not possible to only specify one or the other).

header and *body* match *regexp* in the header/body of a message, *index* matches *pattern* (see [Section 2, “Patterns: Searching, Limiting and Tagging”](#)) in the message index.

object can be one of:

- attachment
- bold (hiliting bold patterns in the body of messages)
- error (error messages printed by Mutt)
- hdrdefault (default color of the message header in the pager)
- indicator (arrow or bar used to indicate the current item in a menu)
- markers (the “+” markers at the beginning of wrapped lines in the pager)
- message (informational messages)
- normal
- quoted (text matching [\\$quote_regexp](#) in the body of a message)
- quoted1, quoted2, ..., quotedN (higher levels of quoting)
- search (hiliting of words in the pager)
- signature
- status (mode lines used to display info about the mailbox or message)
- tilde (the “~” used to pad blank lines in the pager)
- tree (thread tree drawn in the message index and attachment menu)

- underline (highlighting underlined patterns in the body of messages)

foreground and *background* can be one of the following:

- white
- black
- green
- magenta
- blue
- cyan
- yellow
- red
- default
- colorx

foreground can optionally be prefixed with the keyword `bright` to make the foreground color boldfaced (e.g., `brightred`).

If your terminal supports it, the special keyword *default* can be used as a transparent color. The value *brightdefault* is also valid. If Mutt is linked against the *S-Lang* library, you also need to set the *COLORFGBG* environment variable to the default colors of your terminal for this to work; for example (for Bourne-like shells):

```
set COLORFGBG="green;black"
export COLORFGBG
```

NOTE:

The *S-Lang* library requires you to use the *lightgray* and *brown* keywords instead of *white* and *yellow* when setting this variable.

NOTE:

The `uncolor` command can be applied to the index object only. It removes entries from the list. You **must** specify the same pattern specified in the `color` command for it to be removed. The pattern “*” is a special token which means to clear the color index list of all entries.

Mutt also recognizes the keywords *color0*, *color1*, ..., *colorN-1* (N being the number of colors supported by your terminal). This is useful when you remap the colors for your display (for example by changing the color associated with *color2* for your xterm), since color names may then lose their normal meaning.

If your terminal does not support color, it is still possible change the video attributes through the use of

the “mono” command:

Usage:

mono *object attribute*

mono { header | body } *attribute regexp*

mono index *attribute pattern*

unmono index { * | *pattern* ... }

For *object*, see the color command. *attribute* can be one of the following:

- none
- bold
- underline
- reverse
- standout

10.显示邮件头

用法:

ignore *pattern* [*pattern* ...]

unignore { * | *pattern* ... }

邮件经常会被处理程序自动加上很多邮件头字段，其中的很多字段显示在屏幕上没有什么用处。此命令用来指定一般情况下你不希望在pager菜单中看到的字段。

你不需要指定字段完整的名字。例如，“ignore content-”会忽略所有以“content-”开头的字段”。“ignore *”会忽略所有的字段。

要从已经添加的项中删除某些项，使用“unignore”命令，“unignore”命令会使Mutt显示与pattern匹配的邮件头字段。例如，如果你使用了“ignore x-”，同时也可以使用“unignore x-mailer”命令来显示x-mailer字段。

“unignore *”将删除所有已被忽略的项。

例如:

例3.8.显示哪些邮件头

```
# Sven's draconian header weeding
ignore *
unignore from date subject to cc
unignore organization organisation x-mailer: x-newsreader: x-mailing-list:
unignore posted-to:
```

用法：

```
hdr_order header [ header ... ]
```

```
unhdr_order { *| header ... }
```

使用命令`hdr_order`你可以指定一个顺序，在你阅读邮件时`mutt`将按照这个顺序显示邮件头字段。

“`unhdr_order *`”will clear all previous headers from the order list, thus removing the header order effects set by the system-wide startup file.

例3.9.邮件头显示顺序

```
hdr_order From Date: From: To: Cc: Subject:
```

11. 后备地址

用法：

```
alternates [ -group name ... ] regexp [ regexp ... ]
```

```
unalternates [ -group name ... ] { *| regexp ... }
```

根据不同的对象，`mutt`会以不同的方式对待邮件，这取决于你是发件人还是收件人等等。例如，当你回复一封由你发出的邮件，`mutt`会自动发给这封邮件的原始收件人——一般来讲，你向你自己发一封信是没有意义的（参见`$reply_to`）。

许多用户使用不同的地址接受电子邮件。要充分利用`mutt`的这个特性，就必须让`mutt`你的电子邮件地址。这就是`alternates`命令的目的：它可以使用多个正则表达式来指定你的邮件地址。

因为地址使用正则表达式来匹配，而不是精确地严格地比较，所以你应该尽可能精确的指定你的地址，以免避免错误匹配。比如，你指定的是：

```
alternates user@example
```

`mutt`将会把“`some-user@example`”作为你的地址，这将造成很多不便。作为一种解决办法，在这种情况下地址应该被指定为：

```
alternates '^user@example$'
```

若使用`-group`选项把随后的所有正则表达式添加到the named group。

同样的，`unalternates`命令作为`alternates`命令的列外情况。如果一个地址匹配`alternates`命令，但这并不是你的地址，你可以使用`unalternates`命令指定更确切一些。

要从`alternates`的指定中删除一个正则表达式，利用`unalternates`命令来指定这个`regexp`。同样的，如果`alternates`的`regexp`匹配`unalternates`上的条目，`unalternates`的条目就会被移除。如果`unalternates`的`regexp`指定为“`*`”，那么`alternates`的所有条目都会被移除。

12.邮件列表

用法：

```
lists [ -group name ... ] regex [ regex ... ]
```

```
unlists [ -group name ... ] { *| regex ... }
```

```
subscribe [ -group name ... ] regex [ regex ... ]
```

```
unsubscribe [ -group name ... ] { *| regex ... }
```

在处理邮件列表方面Mutt有一些不错的特性。为了充分利用它们，你必须指定哪些地址属于邮件列表，以及你订阅了哪些邮件列表。一旦你做了这些，[<list-reply>](#)功能就能很好的工作。此外，当你向已订阅的列表发送邮件时，mutt会在邮件头中加入一个“Mail-Followup-To”字段，以告诉其他用户的邮件客户端不要回复到你的邮件地址。

注意：

“Mail-Followup-To”字段是非标准扩展，并不被所有的邮件客户端支持。添加它并不能完全防止收到个人直接CC来的邮件。也要注意Mail-Followup-To字段的生成是由变量\$followup_to控制的。

更确切地说，Mutt保持了一个清单，其中记录了已知的和已订阅的邮件列表地址。每一个订阅的邮件列表是众所周知的。标记一个地址为邮件列表，使用“list”的命令。标记一个地址为你已经订阅的邮件列表，使用“subscribe”命令。

这两个命令你都可使用正则表达式。To mark all messages sent to a specific bug report's address on mutt's bug tracking system as list mail，例如，你可以这样写“subscribe [0-9] * @ bugs.guug.de ”。通常，给出邮件地址的一部分就足够了。

要更确切地指定地址，你就不能定义的太含糊。例如，你订阅了Mutt的邮件列表，你将收到来自mutt-users@mutt.org的电子邮件。所以，告诉Mutt这个地址是一个邮件列表，你可能会将“lists mutt-users@”加入你的配置文件。还要告诉Mutt你已经订阅了它，添加“subscribe mutt-users”到你的配置文件代替。如果你会收到邮件地址为mutt-users@example.com的人的来信，你就需要使用“lists ^mutt-users@mutt\\.org\$”或“subscribe ^mutt-users@mutt\\.org\$”来匹配实际的地址。

使用-group选项会把随后的所有正则表达式添加到the named group。

“unlists”命令从已知的和已订阅的邮件列表清单中删除一个条目。使用“unlists *”删除所有的条目。

要从已订阅的邮件列表清单中删除一个条目，但在已知的邮件列表清单中保留它，使用“unsubscribe”命令。

13.使用多个spool信箱

用法：

```
mbox-hook [!]pattern mailbox
```

此命令是用来将指定邮箱内的已读邮件自动移动到另一个邮箱里面，当你退出或切换文件夹时。*pattern*是一个正则表达式，与其匹配的信箱被当作“spool”信箱，*mailbox*指定已读邮件被保存到哪一个信箱。

与其他一些*hook*命令不同，只有第一个匹配模式是有效的（因此不可能将已读邮件保存到多个信箱）。

14. 监视新邮件

用法：

```
mailboxes mailbox [mailbox ...]
```

```
unmailboxes { *| mailbox ... }
```

此命令指定信箱，并定时检查其是否收到新邮件。

信箱可以是本地目录，也可以文件（取决于Maildir/Mh还是Mbox/Mmdf格式）。如果Mutt被编译进了POP和/或IMAP支持，信箱也可以是的POP/IMAP信箱的URL地址。URL语法在1.2节“URL语法”中有说明，POP和IMAP分别在第3节“POP3支持”和第4节“IMAP支持”中说明。

Mutt提供了许多高级的特性来处理（多个）信箱和其中的新邮件，请参阅第9节“处理多个信箱”获得详细信息（包括what situations以及Mutt检查新邮件的频率）。

“unmailboxes:”命令用来从mailboxes定义的文件夹中删除一个信箱。使用“unmailboxes *”删除所有的信箱。

注意：

mailboxes命令中的信箱在命令执行时被确定，因此，如果定义这些信箱时用的名称中包含快捷字符（如“=”和“！”），任何对这些快捷字符由影响的变量（如\$folder和\$spoolfile）应在mailboxes命令之前定义。如果没有这些快捷字符，定义信箱时应使用绝对路劲。否则mutt会试图在mutt启动的地方以相对路径的方式寻找信箱，这样会造成很多麻烦。

Mbox和Mmdf格式的信箱，新邮件的检测是通过比较文件的访问和/或修改时间：如果文件在被修改之后没有被访问的话，Mutt就认为文件中又新邮件。一些工具像biff或frm或任何其他会访问信箱的程序可能会导致Mutt发现不了的新邮件，如果它们不恰当地重置访问时间的话。其他可能导致Mutt发现不了新邮件的原因是一些备份工具（更新了访问时间）或文件系统被挂载时没有使用更新访问时间支持。

如果Mbox或Mmdf格式的信箱在检测新邮件时有一些不可靠，可以使用\$check_mbox_size选项使Mutt以跟踪检测文件大小的方式来检查新邮件。

15.自定义邮件头

用法：

```
my_hdr string
```

```
unmy_hdr { *| field ... }
```

`my_hdr`命令允许你创建自己的邮件头字段，并会添加到你发送的每一封邮件中。

例如，如果你想添加一个“Organization:”字段到你发送的所有邮件中，你可以将命令

例3.10.自定义邮件头

```
my_hdr Organization: A Really Big Company, Anytown, USA
```

放在您的`.muttrc`文件中。

注意：

关键字和冒号（“:”）之间不允许有空格出现。电子邮件标注（RFC2822）规定空格在这里是非法的，Mutt严格遵守标准。

如果你只想在一封邮件中添加一个字段，您可以设置`$edit_headers`变量，或者使用`{`功能（默认是：“E”）在compose菜单上，这样你就可以连同正文一起编辑邮件头字段。

要删除自定义的邮件头字段，使用`unmy_hdr`命令。您可以使用星号（“*”），删除所有字段，或只删除指定的字段。例如，要删除所有“To”和“Cc”字段，你可以使用：

```
asterisk
```

16.指定默认保存邮箱

用法：

```
save-hook [!]pattern mailbox
```

此命令用于覆盖保存邮件时使用的默认邮箱。如果邮件匹配`pattern`，`mailbox`将被用作默认信箱来保存邮件，详见[Hook中的邮件匹配](#)。

为了提供更大的灵活性，Mutt允许在指定`mailbox`时，使用[\\$index_format](#)下的扩展变量。

示例：

例3.11.在save-hook中使用%-expandos

```
# default: save all to ~/Mail/<author name>
```

```
save-hook . ~/Mail/%F
```

```
# save from me@turing.cs.hmc.edu and me@cs.hmc.edu to $folder/elkins
save-hook me@(turing\\.)*cs\\.hmc\\.edu$ +elkins

# save from aol.com to $folder/spam
save-hook aol\\.com$ +spam
```

也可参见fcc-save-hook命令。

17.指定默认的Fcc: mailbox，编写邮件时

用法：

```
fcc-hook [!]pattern mailbox
```

此命令在变量\$record之外指定一个信箱保存已发送的邮件。Mutt搜索到收件人名单中有匹配`regex`的条目时，就使用`mailbox`作为Fcc: mailbox的默认值。如果没有找到匹配项，信息将被保存到\$record指定的邮箱中。

为了提供更大的灵活性，Mutt允许在指定`mailbox`时，使用\$index_format下的扩展变量。

关于`pattern`的具体格式请查看[Message Matching in Hooks](#)。

例如： fcc-hook [@.]aol\\.com\$ +spammers

上例将所有发给域名为aol.com的邮件保存到信箱“spammers”中。可参见fcc-save-hook命令。

18.同时指定save-hook和fcc-hook的默认值

用法：

```
fcc-save-hook [!]pattern mailbox
```

此命令是一个快捷命令，相当于同时使用了fcc-hook和save-hook命令，以及它们的参数，还包括`mailbox`下的\$index_format的%-expansion扩展。

19.根据收件人改变设置

用法：

```
reply-hook [!]pattern command
```

```
send-hook [!]pattern command
```

```
send2-hook [!]pattern command
```

这些命令可根据收件人来执行任意的配置命令。`pattern`用来匹配邮件，细节见“[Hook中的邮件匹配](#)”。当邮件与`pattern`匹配时，`command`被执行时。

`reply-hook`只在你回复邮件的时候执行，而不是发送邮件的时候。`send-hook`在写新邮件和回复邮件是都会执行。

注意：

`reply-hook`会先于`send-hook`执行，与在配置文件中指定的先后顺序无关。

`send2-hook`的执行是在邮件发生改变的时候，无论是由于编辑，还是使用compose菜单更改收件人或主题，`send2-hook`在`send-hook`之后执行。它可以根据发件人的地址来设置如`$sendmail`之类的参数。

`send-hook`和`reply-hook`有时会发生相同匹配的情况，命令根据`muttrc`中指定的顺序执行。

例如：`send-hook mutt "set mime_forward signature=''"`

此命令另一种典型用法是根据收件人更改`$attribution`，`$signature`和`$locale`变量的值，从而改变`attributions`和签名的语言。

注意：

`send-hook`只在`mutt`第一次得到收件人名单后执行一次。之后再添加收件人不会使`send-hook`执行。同时还要注意，当从`send-hook`执行时，修改收件人或主题字段的命令`my_hdr`，不会对当前邮件有任何影响。

20. 格式化邮件时更改设置

用法：

`message-hook` *[!]pattern command*

这个命令可以在查看或格式化邮件时执行任意的配置命令。显示的时候，邮件匹配*pattern*，那么*command*就会执行。当出现多个匹配时，`mutt`会按照命令在`muttrc`中的定义顺序依次执行。

查看“[Hook中的邮件匹配](#)”获得*pattern*的确切格式和有关信息。

示例：

```
message-hook ~A 'set pager=builtin'
message-hook '~f freshmeat-news' 'set pager="less \"+/^  subject: .*\"'
```

21. 选择收件人的密钥

用法：

`crypt-hook` *pattern keyid*

当使用PGP/GnuPG或OpenSSL来加密邮件，你可能希望将一个密钥与某一个地址自动地联系起来，因为不能从改地址推测出所用的公共密钥，也可能因为某中原因你需要改变mutt正常情况下使用的密钥。命令`crypt-hook`提供了一个方法，通过这个方法你可以指定一个公钥，在给特定的收件人发信时自动使用它。

在这里`keyid`的含义广泛的：它可以是key ID的数值、邮件地址、甚至名字。

22.添加按键序列到键盘缓冲区

用法：

`push string`

此命令用来将字符串添加到键盘缓冲区。该字符串可以包含控制字符，按键名称和功能名称，如宏里面的`sequence`一样。你可以使用它在启动时，或进入某些文件夹时自动的执行一系列命令。例如，下面的命令可以再进入文件夹时自动地折叠全部thread：

例3.12.嵌入`push`到`folder-hook`命令

```
folder-hook . 'push <collapse-all>'
```

23 执行功能

用法：

`exec function [function ...]`

此命令用来执行任意功能。“功能”一节列出了所有的功能。“`exec function`”，等同于“`push <function>`”。

24.邮件评分

用法：

`score pattern value`

`unscore { *| pattern ... }`

`score`命令的作用是当邮件与`pattern`匹配时，给邮件加上值为`value`的分数。`pattern`是一个字符串，其格式在“模式”一节有所描述（注：为了提高效率，需要扫描整个邮件的模式在index菜单不可用，如`~b`、`~B`或`~h`）。`value`是一个正数或负数。一条邮件最后得的总分等于所有与其匹配的`score`条目的总和。不过，你可以在`value`之前使用前缀等号（=），使邮件在匹配`pattern`时得分等于`value`。总分为负的记为0。

`unscore`命令已定义的`score`条目中移除一条。您必须指定与定义`score`时相同的`pattern`来移除它。`pattern`“*”意味者移除所有`score`定义的条目。

25.垃圾邮件检测

用法：

```
spam pattern format
```

```
nospam { *| pattern }
```

Mutt支持外部的垃圾邮件过滤器。通过使用命令spam和nospam定义spam模式，你可以根据其垃圾邮件特征来提取，搜索，排序邮件。你还可以在变量\$index_format中使用序列%H，让index菜单中显示邮件的垃圾邮件特征。（Tip: try %?H?[%H] ? to display spam tags only when they are defined for a given message）

第一个步骤需要使用命令spam，定义你的外部过滤器的垃圾邮件模式。*pattern*应该是一个正则表达式，匹配邮件头字段。如果信箱中的任何邮件匹配这个表达式，那么它会收到一个“spam tag”或“spam attribute”（除非它也匹配nospam中的pattern，见下文）。这完全是由你来决定的，用参数*format*来控制。*format*可以是任何静态文本，但它也可以是包括向后引用的*pattern*表达式。（“向后引用”是指包含在括号中的子表达式）%1会被替换成正则表达式中一个向后引用的表达式，%2则被换成第二，等等。

如果你使用多个垃圾邮件过滤器，那么一封邮件可能会有一个以上的与垃圾邮件相关的邮件头。你可以为每一个过滤器定义spam模式。如果邮件匹配两个或更多的pattern，而且变量\$spam_separator被设置为一个字符串，那么邮件的spam tag将由所有的*format*字符串连接起来，中间用变量\$spam_separator隔开。

举例来说，假设我用DCC，SpamAssassin和PureMessage。我可能会这样设置：

示例3.13 配置垃圾邮件检测

```
spam "X-DCC-.*-Metrics:.*(....)=many"          "90+/DCC-%1"
spam "X-Spam-Status: Yes"                        "90+/SA"
spam "X-PerlMX-Spam: .*Probability=([0-9]+)%"   "%1/PM"
set spam_separator=", "
```

如果我收到一封被DCC用“many”标记了关键字“Fuz2”的邮件，并且PureMessage标记该邮件97%的概率为垃圾邮件，那么这封邮件的spam tag就应该是90+/DCC-Fuz2, 97/PM（DCC处理结果中“=many”之前的四个字符是被检查的关键字-在此例中，是“Fuz2”）。

如果变量\$spam_separator没有设置，那么每个spam pattern匹配的字符串会被下一个匹配项覆盖，而不是将所有*format*字符串连接起来，最后你只会得到最后一个匹配项。

spam tag是用来在index菜单中显示的，如果你在变量\$index_format中使用了%H的话。这也是字符串的~H模式匹配表达对比赛的<search>和<limit>职能。它是分类的垃圾邮件的属性将使用作为一种关键。

这是一个相当复杂的例子，实际上大多数人只会使用一个垃圾邮件过滤器。简单的越配置，mutt就越高效，特别是在排序上。

一般而言，当你按spam tag排序时，mutt会在字符上进行排序 -即将字母和数字一起排序。然

而，如果spam tag以一个数字开头，mutt将首先按算术值排序，只有在两项的数字相等时才以字符排序（这有点像UNIX的命令`sort -n`）。完全没有垃圾邮件特征的邮件——这就是，不匹配任何一个spam pattern的邮件——在排序时优先级最低。数字从0开始向上排序。最后，对非数值的字符串进行排序，“a”的优先级低于“z”。显然，一般情况下，当你可以从过滤器取得原始数据时，按spam tag排序是非常有用的。但如果你不能，mutt仍然可以做一些有用的事情。

命令nospam用于spam pattern的例外情况。如果一个邮件的某个字段匹配一个spam命令，但可能不希望给这封邮件一个spam tag，你可以使用命令nospam写一个pattern列表。

如果nospam的pattern和某个spam的pattern完全相同，那将会从spam的列表中将该条目移除，而不是增加一个例外条件。同样的，如果spam的pattern于某个nospam的完全相同，那这个nospam对应的条目将被移除。如果nospam的pattern是“*”，那么spam和nospam中的所有条目都将被移除。如果你结合folder-hook来使用spam和nospam，这可能是默认的行为。

你可以定义任意个spam或nospam命令。你甚至可以让mutt进行原始的垃圾邮件检测——例如，如果你认为所有来自MAILER-DAEMON的邮件都是垃圾邮件，你可以这样使用命令spam：

```
spam "^From: .*MAILER-DAEMON"          "999"
```

26 设置和查询变量

26.1 命令

下面的命令可以控制和查询变量：

用法：

```
set { [ no | inv ] variable | variable=value } [...]
```

```
toggle variable [ variable ... ]
```

```
unset variable [ variable ... ]
```

```
reset variable [ variable ... ]
```

这些命令用来设置（或取消）变量。变量有四种基本的类型：布尔，数值，字符串和quadoption。布尔变量可以被设置（true）或取消（false）。数值变量可以被赋值为正整数。字符串变量包括一些可打印字符，如果包含空格或制表符则必须放在引号中，你也可以分别使用转义序列“\n”和“\t”来表示换行和退格。quadoption变量是用来控制对于某些行为是否给出提示，或指定一个默认的行为。yes会使行为自动被执行就好像你对于mutt的问题回答了yes。同样，no将使行为不被执行就好像你回答了“no”。ask-yes会触发一个提示，它的默认回答是“yes”，ask-no将触发一个默认回答为“no”的提示。

变量前缀“no”将取消该变量。例如：`set noaskbcc`。

对于布尔变量，你可以使用变量前缀inv来切换变量的开关（true或false）。当编写宏的时候，这是非常有用的。例如：`set invsmart_wrap`。

命令toggle会自动给指定的变量加上前缀inv。

命令unset会自动给指定的变量加上前缀no。

在index菜单中使用功能<enter-command>，在变量名称前加上一个问号，可以查询变量的值：

```
set ?allow_8bit
```

实际上只是布尔和quadoption变量才需要问号。

命令reset将指定变量的值重置到编译时的默认值（在本手册中有说明）。如果你使用命令set以及使用变量前缀“&”，这和命令reset具有同样的效果。

对于命令reset，有一个特殊变量“all”，使用这个特殊变量你可以重置所有变量到系统默认值。

26.2.User-defined variables

26.2.1 简介

除了在配置变量一节所列出的变量外，mutt还支持以my_开头的用户定义变量，例如，my_cfgdir。

命令set会创建一个自定义my_变量，如果它已经存在，就改变它的值。命令unset和reset会删除变量。

由于用户定义的变量被扩展的方式和环境变量一样（除[shell-escape](#)命令和backtick符号的扩展），这个特性可以使配置文件更易读。

26.2.2 示例

下面的例子定义以及使用了变量my_cfgdir来简化source命令：

例3.14.使用自定义变量使配置文件易读

```
set my_cfgdir = $HOME/mutt/config

source $my_cfgdir/hooks
source $my_cfgdir/macros
# more source commands...
```

在宏里面自定义变量也可以用来备份变量的值。在下面的示例中，\$delete的值临时改变了，它的初始值被保存在变量my_delete中。当宏执行完时，\$delete又恢复了初始值。

例3.15.使用自定义变量来备份其他变量的值

```
macro pager ,x '\
<enter-command>set my_delete=$delete<enter>\
<enter-command>set delete=yes<enter>\
...\
<enter-command>set delete=$my_delete<enter>'
```

mutt是在解析配置文件时扩展变量的，上个例子中`$my_delete`的值是在配置文件被读取时变量`$delete`的值。如果之后在同一个文件或其他配置文件中改变了`$delete`的值，这将不会影响`$my_delete`的值。然而，扩展也可以被推迟到运行时，如下一个例子所示，通过转义符号`$`。

例3.16.推迟自定义变量到运行时扩展

```
macro pager <PageDown> "\
<enter-command> set my_old_pager_stop=\$pager_stop pager_stop<Enter>\
<next-page>\
<enter-command> set pager_stop=\$my_old_pager_stop<Enter>\
<enter-command> unset my_old_pager_stop<Enter>"
```

注意在`<enter-command>`和配置命令`set`之间有一个空格，防止mutt将宏命令保存到记录中。

27.从另一个文件读入配置命令

用法：

`source filename`

此命令允许使用另外的包含配置命令的文件。例如，我把我定义的所有别名放在`~/.mail_aliases`里，然后可以配置`~/.muttrc`是mutt可以读取我的别名文件，这样我就能集中管理别名。

如果该文件以波形符（“`~`”）开头，这代表它在你home目录之下。

如果文件名以竖线（`|`）结尾，那么`filename`就表示可执行的程序，并且mutt会读取它的输出（例如：`source ~/bin/myscript|`）。

28.移除hooks

用法：

`unhook { *| hook-type }`

此命令允许你移除已经定义的hooks。你可以使用字符`"*`作为参数来移除所有的hooks，或者你也可以只移除一种类型的hook，就像这样，`unhook send-hook`。

29.格式字符串

29.1.基本用法

格式字符串是一个常见的概念，你会发现在配置mutt时发现它，特别是在变量`$index_format`、`$pager_format`、`$status_format`和其他“`*_format`”变量。它也是很简单的，很可能你已经知道如何使用它们了。

最基本的格式字符串的元素是一个字符跟一个百分比符号。例如，变量`$index_format`中的`%s`代表邮件的主题字段变数。The“expandos”available are documented with each format variable, but there

are general modifiers available with all formatting expandos, too。Those are our concern here。

一些修饰语是从C语言借来的（尽管你可能在Perl，Python，shell，或另一种语言中见过它）。修饰符还有[-]m.n，如%-12.12s。如这些编程语言一样，这些修饰语允许你指定结果字符串的最小和最大单位，以及它的对齐方式。如果符号“-”跟在百分号后，字符串将会左对齐，而不是右对齐。如果它之后紧跟了一个数字，这个数字表示它的最小长度——如果字符串的实际长度小于这个值，将用空间填充。如果有一个小数点，后面又跟上另一个数字，这是允许的最大长度——字符串不得超过此宽度，无论它实际有多长。这三个元素都是可选的，因此，这些字符串的格式都是合法的：%-12s %4c %.15F %-12.15L。

Mutt为格式字符串增加了一些修饰符。如果你使用等号（=）作为数字的前缀（如上面的减号），这将强迫该字符串置于最小长度的中央。例如，%=14y将%y的扩展保留14个字符——在\$index_format中，这是X-Label:字段。如果扩展后的字符串小于14个字符，它将置于14字符长的空间的中央。如果一封邮件的X-Label字段是“test”，则最后的结果看起来像这样，“ test”。

还有两个鲜为人知的修饰符号，它们将影响一个expando扩展的结果。如果任一个格式修饰符和expando字母之间出现下划线（“_”），它将结果都变为小写。如果你使用了冒号（“:”），它会将所有小数点替换成下划线。

29.2.过滤器

任何以（“|”）结束的格式字符串将在扩展后pipe到字符串中的第一个词，以空格隔开。mutt会显示返回的字符串。如果返回的字符串以%结束，那么它将第二次被格式化。这使得过滤器可以生成一个包括% expandos的字符串。

格式字符串中所有的% expandos将在脚本被调用之前被扩展：

例3.17.在格式字符串中使用外部过滤器

```
set status_format="script.sh '%r %f (%L)' |"
```

这使mutt在调用脚本之前扩展%r，%f和%L。这个例子还表明参数可以被引用：单引号里的字符串在扩展后将作为唯一的参数传递给脚本。

一个实际例子是安装在mutt文档里的samples子目录中的脚本mutt_xtitle：它可以用作\$status_format的过滤器，用来设置当前终端的标题，如果支持的话。

第4章.高级用法

目录

[1.正则表达式](#)

[2.模式：搜索，提取和标记](#)

[2.1 模式修饰符](#)

[2.2 简单模式](#)

[2.3 组合模式](#)

[2.4 按日期搜索](#)

[3.使用标记](#)

[4.使用Hook](#)

[4.1 Hook中的邮件匹配](#)

[5.外部地址查询](#)

[6.邮箱格式](#)

[7.快捷符号](#)

[8.处理邮件列表](#)

[9.处理多个文件夹](#)

[10.编辑thread](#)

[10.1 连接thread](#)

[10.2 拆分thread](#)

[11.投递状态通知（DSN）支持](#)

[12.启用WWW浏览器](#)

1.正则表达式

Mutt中所有的模式字符串，包括组合模式，必须使用正则表达式（regex）来指定，正则表达式使用“POSIX extended”语法（与egrep和GNU awk使用的正则表达式语法大致相同）。为方便起见，我们下面简要介绍一下此语法。

如果模式中至少包含一个大写字母，那么搜索是区分大小写的，否则大小写不敏感。

注意：

请注意，在配置命令中使用正则表达式时，符号“\”必须被转义：“\\”。

正则表达式用来描述一组字符串。它通过各种运算符连接一些小的表达式来组成。

注意：

正则表达式可以放在"或'里面，当正则表达式里面包含空白时，这是非常有用的。关于“和'的更多信息，参考[配置文件的语法](#)。匹配字符“或'，你必须在它之前使用反斜杠“\”。

正则表达式的基本功能是匹配单个字符。大多数字符包括所有的字母和数字可以自我匹配。有特殊含义的元字符可能需要用一个反斜杠转义。

“.”匹配任何单个字符。“^”和“\$”是元字符，分别匹配一行的开始和结束。

“[”和“]”之间包含一组字符，它匹配这组字符中的任何一个字符，如果其中的第一个字符是“^”，那么匹配不是这些字符的任意一个字符。例如，正则表达式[0123456789]匹配任何一个数字。一段连续的ASCII字符可以指定第一个和最后一个，中间用连字符“-”隔开。在方括号中大多数的元字符失去了其本来包含的特殊意义。要包括字符“]”，把它放在最前面。同样地，包括字符“^”，把它放到除第一个以外的任意位置。要包括连字符“-”，把它放在最后。

系统中有一些预设的字符类。字符类由“[:”，关键字，和“:]”组成。表4.1，“POSIX正则表达式字符类”列出了POSIX标准的字符类。

表4.1.POSIX正则表达式字符类

字符集	描述
[:alnum:]	字母与数字
[:alpha:]	字母
[:blank:]	空格或制表符
[:cntrl:]	控制字符
[:digit:]	数字
[:graph:]	可打印及可见的字符（空白符是打印的，但不是可见的，“a”就是可打印及可见的）
[:lower:]	小写字母
[:lower:]	可打印的字符（非控制字符）
[:punct:]	标点符号（不是字母、数字、控制字符或空白符的字符）
[:space:]	空白符（如空格，制表符和换页符等等）
[:upper:]	大写字母
[:xdigit:]	十六进制数字

字符类只有在方括号内才有效。

注意：

这些字符类中的方括号是字符类名称的一部份，它必须被放在一个额外的方括号里。例如，`[:digit:]`等价于`[0-9]`。

有两种特殊的序列可用。它们适用于非ASCII码字符集，它可以包含单个的符号（也叫collating element，由一个以上的字符构成），即几个顺序完全相同的字符：

Collating Symbols

Collating Symbol是一个包含在“[.”和“.]”中的多字符的collating element。例如，如果“ch”是一个collating element，那么`[.ch.]`就一个匹配这个collating element的表达式，而`[ch]`是一个匹配“c”或“c”的表达式。

Equivalence Class

一个Equivalence Class是一组等价的字符的本地名称。这个名字被包含在“[=”和“=]”之间。例如，名称为“e”可能会用来表示“è”、“é”和“e”。在此情况下，`[=e=]`可以匹配“è”、“é”和“e”中的任何一个。

正则表达式匹配单个字符时可在匹配字符后面跟上表4.2 “正则表达式的重复操作符”所列操作

符。

表4.2.正则表达式的重复操作符

操作符	描述
?	该操作符是可选的，匹配零次或一次
*	匹配零次或多次
+	匹配一次或多次
{n}	匹配 n 次
{n,}	至少匹配 n 次
{,m}	至多匹配 m 次
{n,m}	至少匹配 n 次，但不超过 m 次

两个正则表达式可以用逻辑与连接；之后表达式匹配每一个子表达式。

两个正则表达式可以用管道符“|”连接;所产生的效果是匹配两个子表达式中的任意一个。

重复优先于相与，相与又高于逻辑或。整个子表达式可包含在圆括号内来忽略这些优先规则。

注意:

如果你编译Mutt时加入了GNU *rx*，下面表4.3 “GNU正则表达式扩展”所列的操作符也可以使用。

表4.3.GNU正则表达式扩展

操作符	描述
\\y	匹配空单词的开始或结尾
\\B	在单词中匹配一个空字符串
\\<	匹配单词的开始
\\>	匹配单词的结尾
\\w	匹配所有单词字符（字母、数字或下划线）
\\W	匹配所有非单词字符
\\`	匹配缓冲区 (字符串)的开始处
\\'	匹配缓冲区结尾

但是请注意，这些操作符不是POSIX所定义的，所以并不能保证它们在各种系统上都能使用。

2. 模式： 搜索， 提取和标记

Mutt中的许多命令允许你使用模式来匹配邮件（`limit`，`tag-pattern`，`delete-pattern`，等

等）。表4.4 “模式修饰符”给出了几种选择邮件的方式。

表4.4.模式修饰符

模式修饰符	描述
~A	所有邮件
~b <i>EXPR</i>	正文中匹配 <i>EXPR</i> 的邮件
=b <i>STRING</i>	正文包含 <i>STRING</i> 的邮件。如果启用了IMAP，将在服务器上执行搜寻 <i>STRING</i> 的工作，而不是下载每封邮件并在本地搜索。
~B <i>EXPR</i>	在整个邮件中匹配 <i>EXPR</i> 的邮件
~c <i>EXPR</i>	“Cc:”字段匹配 <i>EXPR</i> 的邮件
%c <i>GROUP</i>	Cc给 <i>GROUP</i> 成员的邮件
~C <i>EXPR</i>	“To:”或“Cc:”字段匹配 <i>EXPR</i> 的邮件
%C <i>GROUP</i>	发送或Cc给 <i>GROUP</i> 成员的邮件
~d [<i>MIN</i>]- [<i>MAX</i>]	“date-sent”匹配日期范围的邮件
~D	已删除的邮件
~e <i>EXPR</i>	“Sender”字段匹配 <i>EXPR</i> 的邮件
%e <i>GROUP</i>	“Sender”字段包含 <i>Group</i> 成员的邮件
~E	过期的邮件
~F	标记为重要的邮件
~f <i>EXPR</i>	来自 <i>EXPR</i> 的邮件
%F <i>GROUP</i>	来自 <i>GROUP</i> 成员的邮件
~g	使用密钥签名的邮件
~G	使用密钥加密的邮件
~h <i>EXPR</i>	邮件头匹配 <i>EXPR</i> 的邮件
~H <i>EXPR</i>	spam attribute匹配 <i>EXPR</i> 的邮件
~i <i>EXPR</i>	“Message-ID”字段匹配 <i>EXPR</i> 的邮件
~k	包含PGP密钥的邮件
~L <i>EXPR</i>	由 <i>EXPR</i> 发出或来自 <i>EXPR</i> 的邮件
%L <i>GROUP</i>	由 <i>GROUP</i> 成员发出或来自他们的邮件
~l	寄给已知邮件列表的邮件
~m [<i>MIN</i>]- [<i>MAX</i>]	从 <i>MIN</i> 到 <i>MAX</i> 的邮件*)

~n [<i>MIN</i>]- [<i>MAX</i>]	评分在 <i>MIN</i> 到 <i>MAX</i> 范围内的邮件*)
~N	新邮件
~O	旧邮件
~p	发给你（参考后备地址）的邮件
~P	由你（参考后备地址）发出的邮件
~Q	已回复的邮件
~r [<i>MIN</i>]- [<i>MAX</i>]	“date-received”匹配日期范围的邮件
~R	已读的邮件
~s <i>EXPR</i>	Subject”字段匹配 <i>EXPR</i> “的邮件。
~S	superseded messages
~t <i>EXPR</i>	发给 <i>EXPR</i> 的邮件
~T	被标记的邮件
~u	寄给已订阅的邮件列表的邮件
~U	未读邮件
~v	折叠thread之中的邮件
~V	已验证的邮件
~x <i>EXPR</i>	“References”字段匹配 <i>EXPR</i> 的邮件
~X [<i>MIN</i>]- [<i>MAX</i>]	包含 <i>MIN</i> 到 <i>MAX</i> 个附件的邮件*)
~y <i>EXPR</i>	“X-Label”字段匹配 <i>EXPR</i> 的邮件
~z [<i>MIN</i>]- [<i>MAX</i>]	大小介于 <i>MIN</i> 到 <i>MAX</i> 之间的邮件*)
~=	重复的邮件（见\$duplicate_threads）
~\$	未索引的邮件 (需要以thread方式排列)
~ (<i>PATTERN</i>)	thread中所包含匹配 <i>PATTERN</i> 的邮件，例如，所有thread包含的来自你的邮件：~ (~P)

上表中的*EXPR*是指正则表达式。在模式中使用正则表达式需要特别的注意。具体来说，Mutt对这些模式的解析将剥除一层反斜杠（“\”），一般它用于引用。如果你打算在正则表达式中使用反斜杠，你就需要使用两个反斜杠（“\\”）。你可以强制mutt将模式中的*EXPR*看作一个简单的字符串而非正则表达式，使用“=”代替“~”。例如，“=b *.*”表示所有包含基本字符串“*.*”的邮件。简单字符串的匹配不如正则表达式强大，但它的效率更高。尤其是对IMAP文件夹，因为搜寻匹配字符串的工作可以在服务器上执行，而不需要获取每封邮件。特别地，IMAP对待“=h”：它的格式必须是“header: substring”，不会部份地匹配邮件头的字段名。如果你只想要简单地找出包含特定邮件头的邮件，而不关心其具体的值，那么可以省略“substring”。

*) 也可以使用其它的一些形式，如“<[*MAX*]“，”>[*MIN*]”， “[*MIN*]”和“-[*MAX*]”。

2.1.模式修饰符

注意：

匹配地址列表的模式（尤其是c，C，p，P和t），如果整个地址列表中有一个地址匹配，那么这个模式就会匹配这封邮件。如果你想保证地址列表中的每一项都匹配的话，你需要在给模式加上前缀“^”。下面这个例子匹配所有只包含来自德国的收件人的邮件。

```
^~C \.de$
```

2.2.简单模式

Mutt中有两种所谓的“简单搜索”，当搜索或提取邮件时，如果给出的查询语句没有包含一个可用的操作符（即它不包含这些符号：“~”，“=“或”%“），查询也将会进行。如果要查询一个特殊字符，必须在特殊字符之前使用反斜杠（“\”）转义。

第一种是通过检查查询语句是否属于表4.5 “简单搜索关键词”所列出的关键字（大小写不敏感）：如果是的话，Mutt使用关键字所对应的模式修饰符代替关键字。如果你要查询的内容和某个关键字有冲突，你需要把它变成一个正则表达式，以避免和关键字一致。例如，如果你想查询所有匹配“flag”的邮件（使用\$simple_search），而不是要查询被标记为重要的邮件，那么就要写成“[f]lag”。

表4.5.简单搜索关键词

关键字	模式修饰符
all	~A
.	~A
^	~A
del	~D
flag	~F
new	~N
old	~O
repl	~Q
read	~R
tag	~T
unread	~U

第二种简单搜索，是使用组合的模式搜索，利用\$simple_search为模板。Mutt会你的查询字符串插入到模板中来执行搜索。

2.3.组合模式

指定一个以上的查询规则会执行逻辑与操作。例如：

```
~t mutt ~f elkins
```

将选择收件人列表包含“mutt”的，并且在“From:”字段包含“elkins”的邮件。

Mutt同样可以通过以下操作符达到更多的模式组合搜索：

- ! -- 逻辑非操作符
- | -- 逻辑或操作符
- () -- 分组操作符

下面的例子可以说明组合模式搜索。这个例子将选择的所有“To:”或“Cc:”字段中不包含“mutt”，但来自“elkins”的邮件。

例4.1.模式匹配中使用布尔操作符

```
!(~t mutt|~c mutt) ~f elkins
```

下面这个例子在正则表达式中使用空白符（注意'和"）。例子中，该邮件的主题必须匹配“^Junk +From +Me\$”，它必须来自“Jim +Somebody”或“Ed +SomeoneElse”：

```
'~s "^Junk +From +Me$" ~f ("Jim +Somebody"|"Ed +SomeoneElse")'
```

注意：

如果正则表达式中包含括号，或管道符("|"),你必须将表达式放到双引号或单引号内，因为这些字符也被用来分隔不同的模式。例如：`~f "me@(mutt\.org|cs\.hmc\.edu)"`。

没有引号，括号不会结束。它的作用是分离两个逻辑与的模式：`~f me@(mutt\.org` 和 `cs\.hmc\.edu)`。这样总会产生错误的结果。

2.4.按日期搜索

Mutt支持两种类型的日期，*绝对的*和*相对的*。

绝对的：日期的格式必须是日/月/年（月份和年份是可选的，默认为当前的月份和年份）。例子，一个有效的日期范围是：

```
Limit to messages matching: ~d 20/1/95-31/10
```

如果你省略了开始日期，仅仅指定“-DD/MM/YY”，那么在指定日期之前的邮件将被选中。如果您省略了结束日期，指定的是“DD/MM/YY-”，那么所有在指定日期之后的邮件将被选中。如果你只指定了一个日期，而没有破折号（“-”），那么只有在指定日期发送的邮件将被选中。

偏差：您可以给绝对日期加上一个偏差值。偏差是由符号（+或-），后跟数字，最后加上表4.6所列出的时间单位构成的。你可以用“*”代替“+”或者“-”，这相当于同时使用了“+”或者“-”。

表4.6.日期单位

单位	描述
y	年
m	月
w	周
d	天

例如： 要选择2001年1月15号前后两周的邮件，你要使用以下方式：

```
Limit to messages matching: ~d 15/1/2001*2w
```

相对的： 这种类型相对于当前的日期，可以被指定为：

- `>offset` (“offset单位”时间以前的邮件)
- `<offset` (“offset单位”时间之内的邮件)
- `=offset` (“offset单位”时间当天的邮件)

`offset`为一个正数与表4.6 “时间单位”中所列的单位。

例如： 选择1个月之内的邮件，你可以使用

```
Limit to messages matching: ~d <1m
```

注意：

使用相对搜索时都是相对去本地时区，并非index上显示的日期，除非你将格式%[...]加入了\$index_format变量。

3. 使用标记

有时需要一次对很多邮件进行操作，而不是一次一个。比如将一个邮件列表的邮件保存一个单独的文件夹，或删除包含某个主题的所有邮件。要标记所有匹配模式的邮件，用功能<tag-pattern>，默认绑定在“shift-T”上。你也可以使用功能<tag-message>来单独标记每封邮件，默认绑定为“t”。关于Mutt中模式的语法见“模式”。

一旦你标记了需要的邮件，可以使用“tag-prefix”操作符，默认为分号“;”。“tag-prefix”操作符使用后，下一个操作就将应用于所有已标记的邮件，如果此操作可以运用与这种方式的话。如果设置了\$auto_tag变量，下一个操作将自动应用于已标记的邮件，而无需在使用“tag-prefix”。

在宏或push命令中，你可使用“tag-prefix-cond”操作符。如果没有被标记的邮件，mutt会“忽略”宏的其余部分并中止宏的执行。如果遇到“end-cond”操作，Mutt中将取消“忽略”宏的操作；在“end-cond”操作符作用过后，宏剩下的部分将正常的执行。

4.使用Hook

*Hook*这个概念可以在许多其它的程序中看到，它允许你在某个操作之前执行任意的命令。例如，你可能希望根据你当前所在的信箱或者发送邮件的收件人来调整你的设置。在Mutt中，*Hook*包括一个正则表达式或模式，以及一个配置选项/命令。参见

- folder-hook
- send-hook
- message-hook
- save-hook
- mbox-hook
- mbox-hook
- fcc-save-hook

获得每种可用类型的*hook*的具体细节。

注意:

如果一个hook改变了设置，这些变化仍然有效，直到当前会话的结束。这是情况通常是不希望发生的，所以在其它hook之前应该添加一个默认的hook以保证恢复默认的设置。下面的例子关于send-hook和my_hdr:

例4.2.将my_hdr与send-hook结合

```
send-hook . 'unmy_hdr From:'  
send-hook ~C'^b@b\.b$' my_hdr from: c@c.c
```

4.1.Hook中的邮件匹配

和邮件有关的hook（message-hook，reply-hook，send-hook，send2-hook，save-hook，fcc-hook）的执行方式会稍微不同。其他类型的hook，一个正则表达式就足够了。但是，在处理邮件时需要更高级的语句来控制匹配，因为会出于不同的目的你往往那个会有不同的匹配标准。

在hook命令中Mutt允许你使用的模式来匹配邮件。它的工作方式与*限制*、*搜索*邮件是完全一样的，但只限于当你匹配那些mutt从邮件头提取出的字段（即from，to，cc，date，subject，等）的操作符时。

例如，如果你想当你向某个指定的地址发送邮件时，设置你的回复地址，你可以这样做：

```
send-hook '~t ^me@cs\.hmc\.edu$' 'my_hdr From: Mutt User <user@host>'
```

这将执行指定的命令，当向*me@cs.hmc.edu*发送电子邮件时。

然而，也许你并不想写一个完整的pattern语句。你任然可以使用简单的正则表达式，像其他类型的hook一样，在这种情况下Mutt会把它转换成完整的pattern语句，使用变量\$default_hook所定义的内容来转换。mutt是在hook定义的时候进行转换的，因此，\$default_hook的值会在这个时候被使用。

5.外部地址查询

Mutt提供了简单的接口来利用脚本连接到外部数据库如LDAP，ph/qi，bbdb，或NIS。利用变量\$query_command指定链接外部数据库的命令。例如：

```
set query_command = "mutt_ldap_query.pl '%s'"
```

这个脚本要能够接受查询的命令，还应该能返回一段信息，每个匹配项一行，每行以制表符分隔地址，名称和其它一些可选信息。对于错误，例如如果没有匹配结果，返回一个非零的结束代码和一断错误信息。

多个匹配的例子：

```
Searching database ... 20 entries ... 3 matching:
me@cs.hmc.edu           Michael Elkins   mutt dude
blong@fiction.net       Brandon Long    mutt and more
roessler@does-not-exist.org Thomas Roessler mutt pgp
```

mutt的查询功能有两种实现机制。一种是在index菜单使用<query>功能（默认键：Q）查询。它将会提示你输入一个查询语句，然后显示一个包含匹配项的菜单。从查询菜单中你可以选择地址来创建别名或发送邮件。你也可以标记多个地址来发送邮件；或者开始一个新的查询，可以将新的查询结果附加到当前匹配项后。

另外一种查询机制是地址补全，类似别名的自动补全。在任何提示输入地址的地方，你可以使用<complete-query>功能（默认键：^T）根据当前你已输入的地址来执行查询。和别名一样，mutt会去查找你输入的内容（从上一个空格或逗号开始）。如果只有一个匹配结构，mutt会自动补全你输入的地址。如果有多个匹配项，mutt会显示一个菜单。在此菜单上，你可以选择一个或多个地址加入到输入栏。

6. 邮箱格式

Mutt可以读写四种不同的邮箱格式：mbox，MMDF，MH and Maildir。mutt可以自动检测邮箱的类型，因此没有注明每一个邮箱的类型。当创建新的邮箱是，Mutt会按照变量\$mbox_type指定的类型来创建邮箱。

mbox：这是UNIX中使用最广泛的邮箱格式。所有的邮件被储存在一个单一的文件中。每封邮件以类似下面形式的一行开始：

```
From me@cs.hmc.edu Fri, 11 Apr 1997 11:44:56 PST
```

（这一行通常被称为“From_”）。

MMDf：这是`mbox`格式的变体。每封邮件被包含“`^A^A^A^A`”（四个Ctrl + A）的一行分隔。

MH：和`mbox`以及`MMDf`完全不同，邮箱由一个目录组成，每封邮件都存储在单独的一个文件中。文件名表示邮件的号码（不过，这可能与Mutt中显示的邮件号码不同）。被删除的邮件会重新命名，在文件名前添加一个逗号（,）。Mutt通过检测寻找`.mh_sequences`或者`.xmhcache`这两个目录来检测这种类型的邮箱（需要区分正常目录和MH信箱）。

Maildir：最新的邮箱格式，被Qmail（可取代的sendmail的MTA）使用。和MH类似，只是它在邮箱中加入了3个子目录：`tmp`，`new`和`cur`。邮件的文件名以一种独特的方式被选择，即使在这两个程序通过NFS同时写邮箱的时候，这意味着这种格式并不需要文件锁定。

7.快捷符号

Mutt内置了一些快捷符号来表示特定的邮箱。当你需要输入一个文件或信箱路径时，你就可以使用这些快捷符号。

- `!` -- `$spoolfile`指定的信箱（即收件箱）
- `>` -- `$mbox`指定的邮箱
- `<` -- `$record`指定的信箱
- `^` -- 当前信箱
- `-` 或 `!!` -- 上次访问的文件
- `~` -- home目录
- `=` 或 `+` -- `$folder`指定的目录
- `@alias` -- 该别名所对应的地址

8.处理邮件列表

Mutt的一些配置选项使得处理邮件列表更加容易。首先，你必须让Mutt知道哪些地址是邮件列表（从技术上讲，这个地址可以不是一个邮件列表，但它的主要用处是针对邮件列表的），以及你订阅了哪些。这些通过配置文件里的`lists`和`subscribe`命令来完成。

Mutt知道邮件列表后，它就可以实现几个功能，第一是能够在`index`菜单显示你收到的那些邮件列表（例如，一个已订阅的列表）的名称。这将非常有助于区分同一个信箱中个人邮件和邮件列表。在变量`$index_format`中，当“list”出现在“To”字段中时，参数“`%L`”将返回字符串“To<list>”；当它出现在“Cc”字段中时，将返回“Cc<list>”（否则返回作者的姓名）。

来自邮件列表的邮件的“To”和“Cc”字段经常会包含很多地址。大多数人回复的时候并不删除邮件的作者，这就导致作者会收到致两个或两个以上的邮件副本。<list-reply>功能（在`index`和`pager`菜单中默认为“L”），有助于减少这种麻烦，它保证只回复到邮件列表的地址而不是所有收件人（除了指定的Mail-Followup-To，见下文）。

Mutt支持Mail-Followup-To字段。当你给你很多地址（其中包含一个或多个邮件列表地址）发送一封邮件时，如果设置了变量\$followup_to，mutt就会产生一个Mail-Followup-To字段，其中包含要发送邮件的所有收件人地址，但没有你自己的地址。这表明使用group-replies或者list-replies（也称为“followups”）时，邮件只会送到原始的收件人，而不会在发送给你——你将从你订阅的邮件列表收到邮件的副本。

反过来，当你在包含Mail-Followup-To字段的邮件里使用group-replying或list-replying时，mutt会首先分析这个字段，如果设置了变量\$honor_followup_to的话。这种情况下使用list-reply将确保邮件回复到邮件列表，即使在Mail-Followup-To字段中的收件人列表中没有指定它的地址。

注意：

邮件头可以编辑的话，你可以手动创建一个Mail-Followup-To字段。只有在你发送邮件，Mail-Followup-To字段不存在时，Mutt中才会自动生成此字段。

一些邮件列表管理员使用的是另一种方法是生成一个“Reply-To”字段，指向邮件列表的地址，而不是指向邮件的作者。但如果希望回复到邮件的作者是，这就会产生问题，因为大多数邮件客户端会自动回复到“Reply-To”字段中的地址。Mutt中的变量\$reply_to可以帮助你决定使用什么地址。如果设置为ask-yes或ask-no，系统会提示你是回复到“Reply-To”字段中的地址，还是直接回复到"From"字段中的地址。当设置为yes时，将使用“Reply-To”字段。

“X-Label:”字段用于进一步验证邮件列表或者主题（或者仅用于对邮件的注释）。变量\$index_format的参数“%y”和"%Y"可以用来在index菜单显示“X-Label:”字段，在模式中使用修饰符“~y”可以用正则表达式来匹配“X-Label:”字段。“X-Label:”不是一个标准的邮件头字段，但procmail和其他的邮件过滤程序可以很容易的将它地插入邮件头。

最后，Mutt可以让信箱中的邮件按照thread模式排序。thread是一组涉及同一主题的邮件。这是将邮件以及它的所有回复按照树状结构来呈现。如果你曾经使用过thread模式的消息客户端，就会发现它们是相同的概念。它使得处理大量的邮件列表非常容易，因为你可以快速地删除不感兴趣的thread，并迅速找到有价值的主题。

9.处理多个文件夹

Mutt支持多个文件夹，可以在所有文件夹中监视新邮件（见14节“监视新邮件”获得细节）。

在index菜单中，并处于空闲状态（见\$timeout），Mutt会周期性地检查指定信箱中是否有新邮件，在配置文件中通过命令mailboxes来指定信箱。其时间间隔取决于信箱类型：local/IMAP信箱取决于\$mail_check，POP信箱取决于\$pop_checkinterval的设定。

另外目录菜单也支持检查新邮件，如果设置了变量<check-new>的话。按下TAB键将出现一个菜单，显示了命令mailboxes所指定的信箱，并标记了其中是否包含新邮件。如果在命令行启动mutt时使用了选项-y，Mutt将自动进入这个模式。

在pager，index和目录菜单中，包含了功能<buffy-list>（默认绑定为“.”），此功能将在屏幕底部显示包含新邮件的文件夹。

Index菜单中，Mutt默认会在状态栏显示信箱中新邮件的数量，参阅变量\$index_format获得很多细节。

当切换信箱时，Mutt会自动在输入栏填上第一个包含新邮件的信箱（如果有的话），按空格键可以切换至下一个包含新邮件的信箱。

10. 编辑threads

在Mutt中可以动态地调整一些由于软件错误或用户错误的操作习惯而导致混乱的thread。你可以很好地修复这些混乱，消除不必要的烦恼，然后更顺畅的通信。

10.1. 连接thread

在回复邮件时一些邮件客户端常常“忘记”正确地设置“In-Reply-To:”和“References:”字段。这就会破坏threads的结构，因为Mutt不能依靠足够的信息来正确地设置thread。你可以标记回复邮件来修复这样的错误。移动到父邮件上，使用功能<link-threads>（默认绑定为“&”）。回复邮件就会正确地链接到这个“父”邮件上。

你还可以一次连接多个子邮件，标记它们然后使用命令tag-prefix(';')或设置变量auto_tag。

10.2. 拆分thread

在邮件列表上，一些习惯不好的用户在开始一项新的讨论时，喜欢任意选一封邮件使用“reply”命令，然后将主题改变成另一个完全无关的。你可以使用功能<break-thread>（默认绑定为“#”）来修复这个问题，这会将子thread变成另外一个不同的thread。

11. 投递状态通知（DSN）支持

对于电子邮件的投递状态，RFC1894定义了一套MIME content types。可以把它们看作是“回执单”。

关于DSN支持，有两个变量。\$dsn_notify是用来请求不同投递结果（如投递失败，投递成功等等）。\$dsn_return设置回执投递信息时附上邮件的哪些部分（邮件头或整个邮件）。

如果使用\$sendmail来投递邮件，你需要使用Berkeley sendmail 8.8.x（或更高版本），一个支持DSN并兼容Sendmail的命令行选项-N及-R的MTA可被邮件客户端用于发出回执单的请求。请查阅你的MTA的文档确认是否支持DSN。

对于使用\$smtp_url的SMTP投递，mutt是否尝试DSN请求取决于服务器的能力。

12. 启用WWW浏览器

如果邮件中包含URL地址，那么你一定要获得这些URL地址的列表然后使用它们其中之一来启动一个浏览器。一个外部的程序urlview提供了这个功能。urlview可以在<ftp://ftp.mutt.org/mutt/contrib/>获得，配置命令可以这样写：

```
macro index \cb |urlview\n
macro pager \cb |urlview\n
```

第5章.Mutt的MIME支持

目录

[1.在Mutt中使用MIME](#)

- [1.1 在pager中查看MIME邮件](#)
- [1.2 附件菜单](#)
- [1.3 编辑菜单](#)

[2.配置文件mime.types](#)

[3.配置文件mailcap](#)

- [3.1 mailcap文件基本内容](#)
- [3.2 安全使用mailcap](#)
- [3.3 mailcap的高级用法](#)
- [3.4 mailcap文件示例](#)

[4.的MIME Autoview](#)

[5.MIME Multipart/Alternative](#)

[6.附件搜索和计数](#)

[7.MIME Lookup](#)

通过很多人的努力，Mutt成为了一个重要的文本模式的MIME MUA。我们努力提供满足用户需求的MIME功能，并尽可能地符合标准。关于MIME支持，Mutt用到了两个额外的配置文件。一个是mime.types文件，包含了文件扩展名到IANA标准的MIME类型的映射。另一个是mailcap文件，指定了用于处理特定MIME类型的外部程序。

1.在Mutt中使用MIME

在Mutt里有三个地方/菜单可以处理MIME，分别是pager菜单（查看邮件时）、附件菜单和编辑菜单。

1.1.在pager中查看MIME邮件

当你在index选择一封邮件然后在pager中查看，Mutt会解析邮件并将其呈现到一个文本模式的窗口中。Mutt内部支持多种MIME类型，其中包括text/plain，text/enriched，message/rfc822，message/news。In addition，the export controlled version of Mutt recognizes a variety of PGP MIME types, including PGP/MIME and application/pgp.

Mutt用两行信息标注附件。这两行的形式如下：

```
[-- Attachment #1: Description --]
[-- Type: text/plain, Encoding: 7bit, Size: 10000 --]
```

上面的Description是附件的描述或文件名，Encoding这些编码方式中的一种：7bit/8bit/quoted-printable/base64/binary。

如果是Mutt无法处理的MIME类型，它会显示一行信息，如：

```
[-- image/gif is unsupported (use 'v' to view this part) --]
```

1.2. 附件菜单

`view-attachments`（默认绑定：“v”）能够显示邮件的附件菜单。附件菜单显示了邮件中所含附件的清单。在附件菜单上，你可以保存，打印，删除，查看附件，以及将附件通过管道传递给外部程序。你可以先标记一组附件，然后用“tag-prefix”操作符对这些附件采取统一地操作。你也在附件菜单中回复邮件，这时当前选中的（或被标记的）附件会包含在你的回复中。你可以以文本模式查看附件，或使用mailcap中定义的程序来查看它们。

最后，你可以对类型为message/rfc822的附件使用与邮件有关的大多数功能（如[<resend-message>](#)，[<reply>](#)和[<forward>](#)）。

可在附件菜单中查看帮助菜单获得更多信息。

1.3. 编辑菜单

编辑菜单是你发送邮件之前所看到的那个菜单。你可以在上面修改邮件的收件人、主题、以及其他方面的信息。它还列出了邮件的附件清单，包括邮件体。在编辑菜单中，你可以打印，复制，过滤，查看，编辑邮件，重命名一个或一组标记的附件，也可以将有件通过管道传递给外部程序。你还可以修改附件信息，特别是附件类型、编码和描述。

附件以如下形式列出：

```
- 1 [text/plain, 7bit, 1K] /tmp/mutt-euler-8082-0 <no description>
  2 [applica/x-gunzip, base64, 422K] ~/src/mutt-0.85.tar.gz <no description>
```

'-'表示在发送（或是延迟、取消）邮件之后Mutt会删除这个文件。这可以使用功能[toggle-unlink](#)（默认u）来关闭或打开。下一个字段是MIME的content-type，你可以使用功能[edit-type](#)（默认：^T）来更改它。下一个字段是附件的编码方式，它允许一个二进制的邮件以7bit方式编码，功能[edit-encoding](#)（默认：^E）可以更改它。下一个字段是附件大小，四舍五入到千字节或兆字节。下一个字段是文件名，功能[rename-file](#)（默认：R）可以更改文件名。最后一个字段是附件的描述，功能[edit-description](#)（默认：d）可以更改它。

2. 配置文件mime.types

当你想邮件添加一个附件时，Mutt首先会在\${HOME}/.mime.types查找用户的mime.types文件，然后在/usr/local/share/mutt/mime.types或者/etc/mime.types查找系统的mime.types文件。

mime.types文件中每行都包含一个MIME类型和其对应的扩展名，以空白分开。例如：

```
application/postscript      ps eps
application/pgp              pgp
audio/x-aiff                 aif aifc aiff
```

mutt的发行包中包含了mime.types的样本，其中包含大多数你可能会用到的MIME类型。

如果Mutt不能通过你添加的文件的扩展名确定MIME类型，那它就会检查这个文件。如果该文件中不包含二进制信息，Mutt就认为这是纯文本文件，并标记为text/plain。如果包含二进制信息，Mutt就将其标记为application/octet-stream。你可以在编辑菜单中使用命令edit-type（默认：^T）更改mutt为附件指定的MIME类型。MIME类型包含一个主类型和一个子类型，之间以'/'区分。6个主类型：application, text, image, video, audio，经过一系列互联网讨论已经被批准的model类型。只要在mime.types文件中搜索到匹配的条目，mutt就能识别附件。Mutt还能识别其它的主类型，如chemical类型，that is widely used in the molecular modeling community to pass molecular data in various forms to various molecular viewers. 最好仅当收件人希望收到这个文件时，才使用不能识别的MIME类型文件。

3.配置文件mailcap

Mutt支持RFC1524定义的MIME类型，特别地，Unix独有的格式是RFC1524附录A中定义的。这个文件格式就是通常说的mailcap格式。许多MIME兼容程序利用mailcap格式，允许你在一个地方对所有程序指定所有MIME类型的处理方式。使用这种格式的知名程序包括Netscape，XMosaic，lynx和metamail。

为了处理Mutt内部无法处理的各种MIME类型，Mutt可通过一些外部的配置文件以找到额外的处理方式。这些文件的默认搜索路径字符串是一个用冒号分隔下列文件的清单：

1. \$HOME/.mailcap
2. \$PKGDATA/DIR/mailcap
3. \$SYSCONFDIR/mailcap
4. /etc/mailcap
5. /usr/etc/mailcap
6. /usr/local/etc/mailcap

其中\$HOME是你的home目录。\$PKGDATA/DIR和\$SYSCONFDIR目录取决于Mutt安装在哪儿：前者默认包含的是共享数据，后者是系统配置文件。

可通过运行以下命令获得默认的路径：

```
mutt -nF /dev/null -Q mailcap_path
```

特别地，metamail发行包中含有一个mailcap文件，通常在/usr/local/etc/mailcap，其中是一些基础条目。

3.1.mailcap文件基本内容

mailcap文件由许多行组成，分别是有注释行，空白行，定义行。

注释行由字符“#”开头，后面可以是你想要的任何东西。

空白行就是空白。

定义行包含content type、处理命令、和若干个可选字段。每个字段用分号“;”隔开。

content type使用MIME的主类型/子类型方法指定。例如， `text/plain`, `text/html`, `image/gif`, 等等，另外，mailcap支持两种格式的通配符，一种是使用“*”指定子类型，另一种是隐式地指定所有子类型。例如， `image/*`，或`video`，会分别匹配所有image类型和video类型。

处理命令是一个查看指定类型的Unix程序。这里支持两种不同类型的程序。预设的是将MIME邮件体以标准输入传递到程序。你可以使用参数“%s”来改变这种行为。这将导致Mutt将MIME邮件体保存为一个临时文件，然后用这个临时文件替换掉参数“%s”来调用处理命令。在两种情况下，Mutt都会在程序结束的时候调用终端来显示结果，此时Mutt也会删除临时文件，如果有的话。

最简单的形式是，你可以把text/plain类型的邮件通过标准输入传递给外部pager “more”：

```
text/plain; more
```

或者，你也可以以文件的形式传递给邮件：

```
text/plain; more %s
```

可能你还想用lynx以交互方式查看text/html类型的邮件：

```
text/html; lynx %s
```

在这种情况下，lynx不支持从标准输入浏览一个文件，所以你必须使用参数“%s”。

注意：

一些旧版本的lynx存在一个bug，它会查看mailcap文件中`text/html`类型的处理命令。它将找到调用了lynx的那一行，然后运行它。这将造成lynx将自身作为一个需要显示的对象来处理。

另一方面，也许你并不想交互的调用lynx，你只想让lynx把text/html转换成text/plain，那么你可以使用：

```
text/html; lynx -dump %s | more
```

也许你想使用lynx来查看text/html文件，而使用一个pager来查看所有其他的text格式，那么你可以使用：

```
text/html; lynx %s  
text/*; more
```

这是最简单的形式mailcap文件。

3.2安全使用mailcap（译者注：此节暂未翻译）

The interpretation of shell meta-characters embedded in MIME parameters can lead to security problems in general. Mutt tries to quote parameters in expansion of %s syntaxes properly, and avoids risky characters by substituting them, see the \$mailcap_sanitize variable.

Although mutt's procedures to invoke programs with mailcap seem to be safe, there are other applications parsing mailcap, maybe taking less care of it. Therefore you should pay attention to the following rules:

Keep the %-expandos away from shell quoting. Don't quote them with single or double quotes. Mutt does this for you, the right way, as should any other program which interprets mailcap. Don't put them into backtick expansions. Be highly careful with eval statements, and avoid them if possible at all. Trying to fix broken behavior with quotes introduces new leaks - there is no alternative to correct quoting in the first place.

If you have to use the %-expandos' values in context where you need quoting or backtick expansions, put that value into a shell variable and reference the shell variable where necessary, as in the following example (using \$charset inside the backtick expansion is safe, since it is not itself subject to any further expansion):

```
text/test-mailcap-bug; cat %s; copiousoutput; test=charset=%{charset} \  
  && test "`echo $charset | tr '[A-Z]' '[a-z]'"` != iso-8859-1
```

3.3.mailcap的高级用法

3.3.1.可选字段

除了必要的content-type和处理命令，你可以添加其它选项，使用分号';'分隔每个字段。Mutt中又以下几种可选字段：

copiousoutput

此标志告诉Mutt，处理命令可能通过标准输出产生大量的文本信息。这会使Mutt调用一个pager（内置的pager或使用pager变量设置的外部pager）来显示处理命令的输出结果。如果没有这个选项，Mutt会假定处理命令是交互形式的。你可以使用它来取代在基本用法的例子中将lynx -dump的输出pipe给more做法：

```
text/html; lynx -dump %s ; copiousoutput
```

这将使lynx将text/html格式的输出版式化成text/plain格式，然后mutt使用标准pager来显示结果。

needterminal

当使用auto_view来查看附件时，这个选项可以决定是否honor变量\$wait_key。使用交互方式来查看附件时，并且mailcap文件中相应的条目含有选项needterminal，Mutt将根据\$wait_key和程序的退出状态来决定在外部程序退出后你是否需要按下一个键。所有其他情况下，你不需要按键。

compose=<command>

这个选项指定一个命令用来创建指定MIME类型的附件。在编辑菜单中可用。

`composetyped = <command>`

这个选项指定一个命令用来创建指定MIME类型的附件。与compose命令的不同之处在于mutt将expect标准的MIME头。它可以为新附件指定参数，文件名，描述等等。在编辑菜单可用。

`print=<command>`

此选项指定一个命令来打印特定的MIME类型。在附件菜单和编辑菜单中可用。

`edit=<command>`

此选项指定一个命令来编辑特定的MIME类型。在编辑菜单中可用，也可以用它来创建新附件。对于文本附件Mutt将使用默认的编辑器。

`nametemplate = <template>`

这个选项给处理命令字段的参数“%s”指定一个格式。某些程序要正确的解释文件，需要特定的扩展名。例如，当文件扩展名为.html时，lynx才能正确解释text/html类型的文件。所以，你要让lynx能够浏览text/html类型的文件，需要在mailcap文件加入：

```
text/html; lynx %s; nametemplate=%s.html
```

`test=<command>`

此选项指定一个命令来测试mailcap中的条目是否应该使用。命令根据后面的小节“命令的解析”中所定义的规则来解析。如果该程序返回0，表示测试通过，Mutt将使用此条目。如果返回非零值，表示测试失败，Mutt会继续查找合适的条目。注意，content-type必须匹配,Mutt才会执行测试。例如：

```
text/html; netscape -remote 'openURL(%s)' ; test=RunningX
text/html; lynx %s
```

这个例子中，Mutt会运行程序RunningX测试，如果X窗口管理器正在运行，RunningX返回0，如果没有，返回非0值。如果RunningX返回0，Mutt就会调用netscape来显示text/html的文件。如果RunningX返回非0值，Mutt就跳到下一个条目，使用Lynx来显示text/html文件。

3.3.2. 查找顺序

查找mailcap文件条目时，Mutt将选择最有用的条目。例如，如果你尝试打印一个image/gif，在mailcap文件有以下条目，Mutt将从中选择一条：

```
image/*;          xv %s
image/gif;        ; print= anytopnm %s | pnmtops | lpr; \
                  nametemplate=%s.gif
```

Mutt将跳过image/*，使用image/gif这个条目来实现打印图片。

此外，你还可以使用auto_view指定两种查看附件的方式，一种是自动地，另一种是用从附件菜单互交地查看附件。你还可以利用测试功能，根据你的环境来决定是否使用互交的方式浏览附件。

```
text/html;      netscape -remote 'openURL(%s)' ; test=RunningX
text/html;      lynx %s; nametemplate=%s.html
text/html;      lynx -dump %s; nametemplate=%s.html; copiousoutput
```

对于auto_view，Mutt会选择第三条，因为有copiousoutput标记。对于互交方式，Mutt会运行程序RunningX以确定它是否应该使用第一个条目。如果RunningX返回非0值，Mutt就使用第二个条目。

3.3.3.命令的解析

mailcap文件的各种命令通过函数system()传递到shell /bin/sh。但在传递给/bin/sh -c之前，其中的一些参数首先会被解析成特定的扩展。Mutt扩展的关键字有：

%s

如“mailcap文件基本内容”小节，这个关键字被扩展成文件名。此文件包含了邮件体，可以进行查看/打印/编辑，调用编辑器对邮件的编辑结果也就是放到这个文件中。另外，使用这个关键字会使Mutt不会将邮件体通过标准输入传递给查看/打印/编辑程序。

%t

Mutt会把%t扩展成邮件的content type（如同mailcap中定义行的第一个参数，即text/html或image/gif）的文本形式。

%{<parameter>}

Mutt中，这会被扩展成邮件Content-Type:字段中parameter所对应的值。例如，如果邮件中包含以下一行：

```
Content-Type: text/plain; charset=iso-8859-1
```

那么Mutt会将%{charset}扩展成iso-8859-1。默认的mailcap文件使用这个特性让终端可以使用字符集来显示邮件。

\%

这将被“%”替换。

Mutt目前不支持RFC 1524中指定的%F和%n。这些参数主要是为multipart messages服务的，Mutt内部已经可以处理这些事了。

3.4.mailcap文件示例

这个mailcap文件比较简单、标准：

```
# I'm always running X :)
video/*;          xanim %s > /dev/null
image/*;          xv %s > /dev/null

# I'm always running netscape (if my computer had more memory, maybe)
text/html;        netscape -remote 'openURL(%s)'
```

这个mailcap文件展示了很多例子：

```
# Use xanim to view all videos  Xanim produces a header on startup,
# send that to /dev/null so I don't see it
video/*;          xanim %s > /dev/null

# Send html to a running netscape by remote
text/html;        netscape -remote 'openURL(%s)'; test=RunningNetscape

# If I'm not running netscape but I am running X, start netscape on the
# object
text/html;        netscape %s; test=RunningX

# Else use lynx to view it as text
text/html;        lynx %s

# This version would convert the text/html to text/plain
text/html;        lynx -dump %s; copiousoutput

# I use enscrip to print text in two columns to a page
text/*;           more %s; print=enscript -2Gr %s

# Netscape adds a flag to tell itself to view jpegs internally
image/jpeg;xv %s; x-mozilla-flags=internal

# Use xv to view images if I'm running X
# In addition, this uses the \ to extend the line and set my editor# for images
image/*;xv %s; test=RunningX; \
    edit=xpaint %s

# Convert images to text using the netpbm tools
image/*; (anytopnm %s | pnmscale -ysize 80 46 | ppmtopgm | pgmtopbm |
pbmtoascii -1x2 ) 2>&1 ; copiousoutput

# Send excel spreadsheets to my NT box
application/ms-excel; open.pl %s
```

4.MIME自动显示

除了在mailcap文件中显使地告诉Mutt使用定义的程序来查看MIME类型的附件外，Mutt还支持在pager中自动显示MIME附件。

要使它工作，你必须在mailcap文件定义一个打开附件的程序，并加上copiousoutput选项。通常，你还可以使用一个条目来将附件转换成可以在pager中显示的文本形式。

然后，在muttrc中使用命令auto_view列出你希望自动显示的content-types。

例如，如果你设置auto_view为：

```
auto_view text/html application/x-gunzip \
```

Mutt可以使用下列mailcap条目自动显示这些类型的附件。

```
text/html;          lynx -dump %s; copiousoutput; nametemplate=%s.html
image/*;            anytopnm %s | pnmscale -xsize 80 -ysize 50 | ppmtopgm | \
                    pgmtopbm | pbmtoascii ; copiousoutput
application/x-gzip;  gzcat; copiousoutput
application/x-tar-gz; gunzip -c %s | tar -tf - ; copiousoutput
application/postscript; ps2ascii %s; copiousoutput
```

“unauto_view”用于清除已经添加到auto_view的名单。message-hook可以利用这个命令实现根据大小等等信息决定是否自动显示邮件。“unauto_view *”将所有之前定义的auto_view名单删除。

5.MIME Multipart/Alternative

Mutt 在显示 Multipart/Alternative 类型的邮件上有一些技巧。首先，Mutt 会检查清单 alternative_order，以确定是否有一个可用的首选类型。清单 alternative_order 包含若干 mimetypes，其中支持隐式和显式的通配符，例如：

```
alternative_order text/enriched text/plain text application/postscript image/*
```

其次，Mutt将检查这些类型是否定义了auto_view，并使用它。如果没有的话，Mutt将查找一个文本类型。作为最后一次尝试，Mutt将查找它知道如何处理的任何一种类型。

要删除MIME类型从alternative_order清单，请使用unalternative_order命令。

6. 附件搜索和计数

如果你曾经丢失过邮箱中的附件，那么Mutt附件计数和搜索可能对你有用。你可以让邮件索引显示每封邮件的附件数量，或者通过附件数量来搜索邮件。使用命令“attachments”和“unattachments”你还可以指定哪种类型的附件参与计数、哪种类型的附件不参与计数。

要进行统计，Mutt首先要完整地解析所有的MIME邮件。这会降低操作的效率，特别是对远程邮件信箱，比如IMAP，因为先下载所有的邮件，而不管用户是否真的想查看它们。

语法如下：

```
attachments    {+|-}disposition mime-type
unattachments  {+|-}disposition mime-type
attachments    ?
```

disposition是附件的Content-disposition字段的类型——inline或attachment。你可以用I或A来表示。

disposition跟在前缀“+”或“-”后面。如果前缀是“+”，表示允许这个disposition和这种MIME类型参与计数。如果是“-”，表示这个disposition和这种MIME类型前面“+”规则的例外。下面有例子可以

说明这是非常有用的。

很明显，mime-type是指你希望生效的附件类型。mime类型的格式总是主类型/子类型，其中主类型描述了你要查找的附件的粗略类型，子类型描述了在这个粗略类型中的具体。主类型必须是基本字符（或特殊字符“*”），但子类型可以是一个正则表达式（因此，“*/.*”匹配所有MIME类型）。

当使用命令attachments时，所指定的类型被添加到一个清单中。当你使用命令unattachments时，该类型会从清单中移除。此时，该类型并不扩展，也不会匹配指定的MIME类型——他们只是要添加到清单中的文本。只有在它们实际的计算一封邮件时，它们才会去匹配。

用一些例子来说明也许会好一点。这个例子中没有被注释掉的部份定义了默认的配置。

例5.1.附件计数

```
## Removing a pattern from a list removes that pattern literally. It
## does not remove any type matching the pattern.
##
## attachments      +A */.*
## attachments      +A image/jpeg
## unattachments    +A */.*
##
## This leaves "attached" image/jpeg files on the allowed attachments
## list. It does not remove all items, as you might expect, because the
## second */.* is not a matching expression at this time.
##
## Remember: "unattachments" only undoes what "attachments" has done!
## It does not trigger any matching on actual messages.

## Qualify any MIME part with an "attachment" disposition, EXCEPT for
## text/x-vcard and application/pgp parts. (PGP parts are already known
## to mutt, and can be searched for with ~g, ~G, and ~k.)

##
## I've added x-pkcs7 to this, since it functions (for S/MIME)
## analogously to PGP signature attachments. S/MIME isn't supported
## in a stock mutt build, but we can still treat it specially here.
##
attachments      +A */.*
attachments      -A text/x-vcard application/pgp.*
附件—application/x-pkcs7- 。 *
## Discount all MIME parts with an "inline" disposition, unless they're
## text/plain. (Why inline a text/plain part unless it's external to the
## message flow?)
##
attachments      +I text/plain
## These two lines make Mutt qualify MIME containers. (So, for example,
## a message/rfc822 forward will count as an attachment.) The first
## line is unnecessary if you already have "attach-allow */.*", of
## course. These are off by default! The MIME elements contained
## within a message/* or multipart/* are still examined, even if the
## containers themselves don't qualify.
##
#attachments      +A message/* multipart/*
#attachments      +I message/* multipart/*

## You probably don't really care to know about deleted attachments.
attachments      -A message/external-body
attachments      -I message/external-body
```

输入命令“`attachments ?`”,将列出你当前Muttrc中的设定，以便将它们粘贴到其他地方。

7.MIME Lookup

命令`mime_lookup`指定一个不按照mailcap条目进行处理的mime-type的清单。此设置的目的是处理如`application/octet-stream`这样的二进制类型。当附件的MIME类型被命令`mime_lookup`列出，那么会将附件的扩展名与文件`mime.types`中的扩展名比较。最后按照与其匹配的扩展名的对应mime-type来处理这个附件，处理方式由mailcap中的相应条目的规则或muttrc配置选项（如`auto_view`）来决定。一般用法是：

```
mime_lookup application/octet-stream application/X-Lotus-Manuscript
```

命令`unmime_lookup`用于禁用指定MIME类型（如果已经设置了的话）的`mime_lookup`功能。例如，在系统muttrc定义的那些类型。

第6章.可选功能

目录

[1.General notes](#)

[1.1 启用/停用功能](#)

[1.2 URL语法](#)

[2.SSL/TLS的支持](#)

[3.POP3支持](#)

[4.IMAP支持](#)

[4.1 浏览文件夹](#)

[4.2 认证](#)

[5.SMTP支持](#)

[6.管理多个帐户](#)

[7.本地缓存](#)

[7.1 邮件头缓存](#)

[7.2 邮件体缓存](#)

[7.3 维护](#)

[8.Exact address generation](#)

1.General notes

1.1.启用/禁用功能

Mutt支持多种可选功能，这些功能可以在编译的时候通过`configure`脚本的某些参数来启用或禁用。这些功能被列在`configure --help`的输出中的“Optional features”一节中。

在命令行使用`mutt -v`可以查看启用了哪些功能禁用了哪些。如果`compile option`中的内容以“+”开头，表示已启用，如果是“-”，表示已禁用。例如，如果`mutt`在编译时使用GnuTLS代替了OpenSSL，那么`mutt -v`的输出将包括：

```
- USE_SSL_OPENSSL + USE_SSL_GNUTLS
```

1.2.URL语法

Mutt可以支持IMAP，POP3和SMTP协议，这些需要使用URL来连接服务器。Mutt中指定URL的标准语法是（[]中的内容表示它是可选的）：

```
proto[s]://[username[:password]@]server[:port]/[path]
```

`proto`是通信协议：对于IMAP为`imap`，POP3为`pop`，SMTP为`smtp`。“s”的代表“安全连接”，如果使用了“s”，`mutt`将试图使用SSL或TLS建立一个加密连接。如果没有指明端口（`port`），`mutt`会使用系统上该协议默认的端口。

Mutt中所有的协议都支持身份认证，在URL上可以直接加入用户名，而不使用变量`pop_user`或`imap_user`。许多邮件系统把符号“@”也作为用户名的一部份。密码也可以在URL中给出，但如果URL是保存在配置文件中的，那么不建议你这么做。

可选路径的概念只涉及到IMAP。

举个例子，对于IMAP，你可以使用一个替代端口：`imap://imapserver:port/INBOX`。你也可以为每个文件夹指定不同的用户名：`imap://username@imapserver[:port]/INBOX`，或者`imap://username2@imapserver[:port]/path/to/folder`。使用`imaps://`代替`imap://`将使`mutt`利用SSL或TLS在不同的端口上建立一个加密连接。

2.SSL/TLS支持

如果编译`mutt`时加入了IMAP，POP3和/或SMTP支持，那就可以编译进SSL或TLS支持（分别使用OpenSSL或GnuTLS，加入选项`--enable-ssl=...`或`--enable-gnutls=...`来运行`configure`脚本）。如果协议中有代表“安全连接”的“s”。那么Mutt就以加密方式与远程服务器通信。

3.POP3支持

如果在编译时加入了POP3支持（运行`configure`脚本时加入选项`--enable-pop`），`mutt`就又能力在远程POP3服务器上抓取邮件到本地信箱。

使用URL可以连接到远程POP3服务器，`pop`协议为不加密方式，`pops`为加密方式。见1.2节，“URL语法”获得细节。

在POP3服务器上查询新邮件很费时。由于这个原因，Mutt在服务器上检查新邮件的频率由变量`$pop_checkinterval`控制，默认为每60秒查询一次。

另一种收取POP3邮件的方式是使用功能`<fetch-mail>`（默认：G）。它会连接到`$pop_host`收取

所有的新邮件，并将其存放在本地的\$spoolfile信箱。

注意：

如果你只是想要抓取所有的邮件到本地邮箱，你可以考虑使用一个专业的程序，如fetchmail，getmail或其它类似的程序。

4.IMAP支持

如果Mutt编译进了IMAP支持（以选项`--enable-imap`运行`configure`脚本），它就能够处理远程IMAP服务器上的文件夹。

通过文件夹的URL，使用imap或imaps协议你可以处理远程信箱（见1.2节，“URL语法”获得详细信息）。另外，pine-compatible notation也被支持，比如可以这样，{[username@]imapserver[:port][ssl]}path/to/folder。

注意，并非所有的服务器都使用“/”来分隔文件层次。Mutt会检查服务器使用哪一种分隔符，然后转换成这种分隔符。

浏览IMAP服务器上的文件夹时，你可以使用命令`toggle-subscribed`在你指定的问价夹和所有文件夹之间切换。参见变量\$imap_list_subscribed。

查询IMAP服务器上的新邮件可能会有明显的延迟。所以，你需要好好设置变量\$mail_check和变量\$timeout。我个人是这样设置的：

```
set mail_check=90
set timeout=15
```

对与我比较低的网络带宽，这个设置会有比较好的效果。

注意：

如果你使用v12.250之前的UW服务器以mbox格式储存邮件的话，当另一个客户端选择了同一个文件夹，该服务器会报告与客户端的连接断开。

4.1.浏览文件夹

到1.2版，mutt支持了浏览IMAP服务器上的信箱。这基本和浏览本地文件大致一样，除了以下区别：

- 在文件权限的栏位，mutt将显示字符串“IMAP”，在它之前如果出现符号“+”，则表示该条目包含了邮件和子文件夹。在类Cyrus的服务器上文件夹中通常会包含邮件和子文件夹。
- 如果一个条目中包含邮件和子文件夹的话，按下选择键（默认为enter键）会列出所有子

文件夹的视图。如果你要查看该文件夹中的邮件，必须使用`view-file`（默认为`space`键）。

- 你可以使用命令`create-mailbox`、`delete-mailbox`和`rename-mailbox`（默认绑定：`c`、`d`和`r`）来新建，删除以及重命名信箱。你也可以`subscribe`和`unsubscribe`（通常它们分别绑定为`s`和`u`）信箱。

4.2.认证

Mutt支持四种IMAP服务器的认证方法：`SASL`，`GSSAPI`，`CRAM-MD5`和`LOGIN`（Grant Edwards有一个补丁可以加入`NTLM`认证方式，但它尚未被纳入主要方式）。此外，还支持`pseudo-protocol ANONYMOUS`，它可以让你登录到一个公共的IMAP服务器，而无需拥有帐户。如果使用匿名登录，把你的用户名留空或写成“anonymous”。

`SASL`是一个超级认证方式，它是几个协议（包括`GSSAPI`，`CRAM-MD5`，`ANONYMOUS`和`DIGEST-MD5`）中最安全的方式。使用这种方式（包括`DIGEST-MD5`，或者`GSSAPI`），你的整个会话都会被加密，避过那些网络嗅探器。如果可能的话最好使用它们。要使用它，你必须安装Cyrus SASL库，以及编译mutt时加入`--with-sasl`。

Mutt会按照后面的顺序依次尝试所有编译了的和服务器上可用的方式：`SASL`，`ANONYMOUS`，`GSSAPI`，`CRAM-MD5`，`LOGIN`。

几个控制认证的变量：

- `$imap_user` - 你请求IMAP服务器认证的用户名，针对所有认证方式。它显式地覆盖了使用信箱路径的用户名（即使用邮箱名称的形式：`{user@host}`）。
- `$imap_pass` - 你可以填写一个密码供需要密码的认证方式使用。
- `$imap_authenticators` - 一个用冒号分隔的IMAP认证方式列表，包括你希望尝试的方式。如果指定了的话，它会覆盖mutt的默认行为（以上面的顺序尝试所有方式）。

5.SMTP支持

除了支持传统的邮件投递方式——通过`sendmail`兼容的程序，如果使用`--enable-smtp`编译了SMTP支持，mutt也可以使用SMTP方式投递邮件。

如果设置了变量`$smtp_url`，mutt会连接指定的SMTP服务器来发送邮件；如果没有设置，mutt将使用`$sendmail`所指定的程序来投递。

又关URL语法，请参阅1.2小节，“URL语法”。

内置的SMTP支持加密（`smtps`协议使用SSL或TLS方式进行加密），以及使用SASL进行认证。SASL的认证机制由`$smtp_authenticators`指定，其中默认是一个空列表，mutt会从安全等级由高到低的顺序尝试每一个可用的方法。

6.管理多个帐户

如果你有多个IMAP,POP 和/或SMTP账户，你可能会发现管理所有的身份认证很不方便而且容易出错。那么命令account-hook可能对你有所帮助。这个hook工作起来和folder-hook类似，不同的是可以在你访问远程信箱的任何时候被调用（包括在文件浏览器里）。而不仅仅是在你打开信箱时（比如查询新邮件，储存Fcc邮件，保存邮件到文件夹）。因此，account-hook只应该用在和连接相关的设置上，如密码或tunnel命令，而不能用于如发件人地址或名称上（because in general it should be considered unpredictable which account-hook was last used）。

一些例子：

```
account-hook . 'unset imap_user; unset imap_pass; unset tunnel'
account-hook imap://host1/ 'set imap_user=me1 imap_pass=foo'
account-hook imap://host2/ 'set tunnel="ssh host2 /usr/libexec/imapd"'
account-hook smtp://user@host3/ 'set tunnel="ssh host3 /usr/libexec/smtpd"'
```

7.本地缓存

Mutt包含两种类型的本地缓存：（1）“邮件头缓存”和（2）“邮件体缓存”，两种都将在本节介绍。

邮件头缓存是可选的，因为它依赖于外部的库。如果mutt编译了POP和/或IMAP支持，那么邮件体缓存则总是启用的。因为邮件体缓存会被它们使用（邮件体缓存无需外部的库）。

7.1. 邮件头缓存

作为可选支持，Mutt为一些类型的文件夹（IMAP，POP，Maildir和MH）提供了邮件头缓存。邮件头缓存大大地提高了速度，因为对于远程文件夹，邮件头只需要下载一次。对于Maildir和MH，从单个文件读取邮件头远远比在一个文件中的若干封邮件中查找邮件头要快。（Maildir和MH类型中每封邮件分别储存在单个文件中）

配置configure脚本时使用了选项-*enable-hcache*才可以获得邮件头缓存支持。在默认情况下未启用，因为它需要外部的库：tokyocabinet，qdbm，gdbm或bdb中的一个。

如果启用的话，\$header_cache可以指向一个文件或目录。如果设置为指向一个文件，对所有的文件夹都使用一个数据库文件（这可能导致性能很低），如果指向目录，那么每个文件夹使用不同的文件。

对于一个文件夹一个文件的情况，远程文件夹的数据库文件根据URL来命名，而本地文件夹的数据库文件将根据其路径的MD5校验值来命名。这些数据库文件可以被安全的删除，如果你的磁盘空间紧张。通过下面的命令你可以算出本地文件夹对应的数据库文件名称：

```
$ printf '%s' '/path/to/folder' | md5sum
```

命令md5sum也可能叫md5，根据你的操作系统而定。

7.2邮件体缓存

这两种缓存方法可以结合起来，使用同一个目录来存储（对于IMAP/POP，使用有意义的文件

名)，这样可以简化手动维护不便。

除了缓存邮件头外，mutt也可以缓存整个邮件体。这样就能够更快地显示POP和IMAP文件夹中的邮件，因为邮件只会被下载一次。

配置邮件体缓存时变量`$message_cachedir`必须指向一个目录。mutt会在目录中创建一个子目录层，命名方式为：`proto:user@hostname`，这里`proto`为`pop`或`imap`”。在每个目录中，mutt保存邮件到单独的文件中（像Maildir一样），所以通过创建符号链接，这些缓存目录可以作为只读的Maildir目录被mutt查看。

如果磁盘空间紧张，可以移除全部缓存，mutt会悄悄地再次抓取缺失的项目。

7.3.维护

Mutt（至今）没有维护邮件头的缓存数据库的特性，以至于文件变得非常大时只有将它们删除。当移除邮件时，其缓存文件是否会被重新使用依赖与邮件头缓存所依赖的数据库的库文件。

对于邮件体缓存，mutt可以保持本地缓存与远程信箱同步，如果设置了变量`$message_cache_clean`的话。`clean`意味着从缓存中删除已经不在信箱中的邮件，这种情况只会在其他邮件客户端或mutt的一个使用了不同的邮件体缓存路径的实例删除了邮件时发生（Mutt从缓存中删除一封邮件时，会同步到信箱）。`clean`会占用大量的时间，一般情况不应被设置。

8.Exact address generation

在阅读和编辑邮件时，Mutt支持“Name <user@host>”的地址语法。老的语法“user@host (Name)”则只在阅读邮件的时候支持。选项`--enable-exact-address`可以告诉configure在编译的时候开启后一种语法在编辑邮件时的支持。开启了后在`mutt -v`的输出中会包含`EXACT_ADDRESS`。

第7章.性能优化

目录

[1.读写邮箱](#)

[2.从远程文件夹阅读邮件](#)

[3.搜索和提取](#)

1.读写邮箱

mutt读取信箱的性能可以通过两种方式提高：

1. 对于远程文件夹（IMAP和POP），以及以“一封邮件一个文件”方式储存的文件夹（Maildir和MH），使用邮件头缓存可以使mutt的性能提升很大。每个文件夹使用一个数据库可能会进一步地提高性能。
2. Mutt提供了变量`$read_inc`和`$write_inc`来指定更新进度的频率。如果这些值太低，mutt就会花很多时间来更新进度，比它实际读/写文件夹花更多的时间。

例如，当打开一个包含几千封邮件的maildir文件夹，\$read_inc默认的值可能太低了。它可以使用folder-hook根据不同的文件夹来调整大小：

```
# use very high $read_inc to speed up reading hcache'd maildirs
folder-hook . 'set read_inc=1000'
# use lower value for reading slower remote IMAP folders
folder-hook ^imap 'set read_inc=100'
# use even lower value for reading even slower remote POP folders
folder-hook ^pop 'set read_inc=1'
```

这些设置都作用在每封邮件上。然而，由于不同的邮件在大小或者某个操作上会有很大的不同，这种基于文件夹的增大变量值的设置可能也不能达到预期的要求，因为会出现太低或太高的更新频率。因此，Mutt还可以限制每秒的进度更新频率，使用变量\$time_inc。

2.从远程文件夹阅读邮件

从远程文件夹，比如IMAP和POP阅读邮件会比较慢，尤其是对较大的信箱，因为mutt对每个会话只缓存很少的邮件（通常是10）。

要提高效率，并永久地缓存所有邮件，请参阅mutt的邮件体缓存获得详情。

3.搜索和提取

通过行为search或limit来查找邮件时，mutt对于某一些模式会区分正则表达式和普通字符串。以前缀“~”开头的为正则表达式，前缀“=”开头的为普通字符串。

虽然正则表达式搜索的效率很高，但它会比纯字符串搜索慢，在大文件夹中最明显。因此，如果你对于搜索的信息知道的足够多的话，最后使用普通字符串代替正则表达式。

例如，在一个大文件夹中提取所有来自或发给某某的邮件，用=User@代替~User@会快很多。尤其在搜索邮件体时，因为这需要在大量的内容中搜寻。

请注意，使用字符串搜索时会区分大小写，而使用正则表达式搜索，在全为小写字母时是不区分大小写的。

第8章.参考

目录

[1.命令行选项](#)

[2.配置命令](#)

[3.配置变量](#)

[3.1 abort_nosubject](#)

[3.2 abort_unmodified](#)

[3.3 alias_file](#)

[3.4 alias_format](#)

[3.5 allow_8bit](#)

[3.6 allow_ansi](#)

[3.7 arrow_cursor](#)
[3.8 ascii_chars](#)
[3.9 askbcc](#)
[3.10 askcc](#)
[3.11 assumed_charset](#)
[3.12 attach_charset](#)
[3.13 attach_format](#)
[3.14 attach_sep](#)
[3.15 attach_split](#)
[3.16 attribution](#)
[3.17 autoedit](#)
[3.18 auto_tag](#)
[3.19 beep](#)
[3.20 beep_new](#)
[3.21 bounce](#)
[3.22 bounce_delivered](#)
[3.23 braille_friendly](#)
[3.24 check_mbox_size](#)
[3.25 charset](#)
[3.26 check_new](#)
[3.27 collapse_unread](#)
[3.28 uncollapse_jump](#)
[3.29 compose_format](#)
[3.30 config_charset](#)
[3.31 confirmappend](#)
[3.32 confirmcreate](#)
[3.33 connect_timeout](#)
[3.34 content_type](#)
[3.35 copy](#)
[3.36 crypt_use_gpgme](#)
[3.37 crypt_use_pka](#)
[3.38 crypt_autopgp](#)
[3.39 crypt_autosmime](#)
[3.40 date_format](#)
[3.41 default_hook](#)
[3.42 delete](#)
[3.43 delete_untag](#)
[3.44 digest_collapse](#)
[3.45 display_filter](#)
[3.46 dotlock_program](#)
[3.47 dsn_notify](#)
[3.48 dsn_return](#)
[3.49 duplicate_threads](#)
[3.50 edit_headers](#)
[3.51 editor](#)
[3.52 encode_from](#)
[3.53 envelope_from_address](#)
[3.54 escape](#)
[3.55 fast_reply](#)
[3.56 fcc_attach](#)
[3.57 fcc_clear](#)
[3.58 folder](#)
[3.59 folder_format](#)
[3.60 followup_to](#)

[3.61 force_name](#)
[3.62 forward_decode](#)
[3.63 forward_edit](#)
[3.64 forward_format](#)
[3.65 forward_quote](#)
[3.66 from](#)
[3.67 geccos_mask](#)
[3.68 hdrs](#)
[3.69 header](#)
[3.70 help](#)
[3.71 hidden_host](#)
[3.72 hide_limited](#)
[3.73 hide_missing](#)
[3.74 hide_thread_subject](#)
[3.75 hide_top_limited](#)
[3.76 hide_top_missing](#)
[3.77 history](#)
[3.78 history_file](#)
[3.79 honor_followup_to](#)
[3.80 hostname](#)
[3.81 ignore_linear_white_space](#)
[3.82 ignore_list_reply_to](#)
[3.83 imap_authenticators](#)
[3.84 imap_check_subscribed](#)
[3.85 imap_delim_chars](#)
[3.86 imap_headers](#)
[3.87 imap_idle](#)
[3.88 imap_keepalive](#)
[3.89 imap_list_subscribed](#)
[3.90 imap_login](#)
[3.91 imap_pass](#)
[3.92 imap_passive](#)
[3.93 imap_peek](#)
[3.94 imap_pipeline_depth](#)
[3.95 imap_servernoise](#)
[3.96 imap_user](#)
[3.97 implicit_autoview](#)
[3.98 include](#)
[3.99 include_onlyfirst](#)
[3.100 indent_string](#)
[3.101 index_format](#)
[3.102 ispell](#)
[3.103 keep_flagged](#)
[3.104 locale](#)
[3.105 mail_check](#)
[3.106 mailcap_path](#)
[3.107 mailcap_sanitize](#)
[3.108 header_cache](#)
[3.109 maildir_header_cache_verify](#)
[3.110 header_cache_pagesize](#)
[3.111 header_cache_compress](#)
[3.112 maildir_trash](#)
[3.113 mark_old](#)
[3.114 markers](#)

[3.115 mask](#)
[3.116 mbox](#)
[3.117 mbox_type](#)
[3.118 metoo](#)
[3.119 menu_context](#)
[3.120 menu_move_off](#)
[3.121 menu_scroll](#)
[3.122 meta_key](#)
[3.123 mh_purge](#)
[3.124 mh_seq_flagged](#)
[3.125 mh_seq_replied](#)
[3.126 mh_seq_unseen](#)
[3.127 mime_forward](#)
[3.128 mime_forward_decode](#)
[3.129 mime_forward_rest](#)
[3.130 mix_entry_format](#)
[3.131 mixmaster](#)
[3.132 move](#)
[3.133 message_cachedir](#)
[3.134 message_cache_clean](#)
[3.135 message_format](#)
[3.136 narrow_tree](#)
[3.137 net_inc](#)
[3.138 pager](#)
[3.139 pager_context](#)
[3.140 pager_format](#)
[3.141 pager_index_lines](#)
[3.142 pager_stop](#)
[3.143 crypt_autosign](#)
[3.144 crypt_autoencrypt](#)
[3.145 pgp_ignore_subkeys](#)
[3.146 crypt_replyencrypt](#)
[3.147 crypt_replysign](#)
[3.148 crypt_replysignencrypted](#)
[3.149 crypt_timestamp](#)
[3.150 pgp_use_gpg_agent](#)
[3.151 crypt_verify_sig](#)
[3.152 smime_is_default](#)
[3.153 smime_ask_cert_label](#)
[3.154 smime_decrypt_use_default_key](#)
[3.155 pgp_entry_format](#)
[3.156 pgp_good_sign](#)
[3.157 pgp_check_exit](#)
[3.158 pgp_long_ids](#)
[3.159 pgp_retainable_sigs](#)
[3.160 pgp_autoinline](#)
[3.161 pgp_replyinline](#)
[3.162 pgp_show_unusable](#)
[3.163 pgp_sign_as](#)
[3.164 pgp_strict_enc](#)
[3.165 pgp_timeout](#)
[3.166 pgp_sort_keys](#)
[3.167 pgp_mime_auto](#)
[3.168 pgp_auto_decode](#)

[3.169 pgp_decode_command](#)
[3.170 pgp_getkeys_command](#)
[3.171 pgp_verify_command](#)
[3.172 pgp_decrypt_command](#)
[3.173 pgp_clearsign_command](#)
[3.174 pgp_sign_command](#)
[3.175 pgp_encrypt_sign_command](#)
[3.176 pgp_encrypt_only_command](#)
[3.177 pgp_import_command](#)
[3.178 pgp_export_command](#)
[3.179 pgp_verify_key_command](#)
[3.180 pgp_list_secring_command](#)
[3.181 pgp_list_pubring_command](#)
[3.182 forward_decrypt](#)
[3.183 smime_timeout](#)
[3.184 smime_encrypt_with](#)
[3.185 smime_keys](#)
[3.186 smime_ca_location](#)
[3.187 smime_certificates](#)
[3.188 smime_decrypt_command](#)
[3.189 smime_verify_command](#)
[3.190 smime_verify_opaque_command](#)
[3.191 smime_sign_command](#)
[3.192 smime_sign_opaque_command](#)
[3.193 smime_encrypt_command](#)
[3.194 smime_pk7out_command](#)
[3.195 smime_get_cert_command](#)
[3.196 smime_get_signer_cert_command](#)
[3.197 smime_import_cert_command](#)
[3.198 smime_get_cert_email_command](#)
[3.199 smime_default_key](#)
[3.200 ssl_client_cert](#)
[3.201 ssl_force_tls](#)
[3.202 ssl_starttls](#)
[3.203 certificate_file](#)
[3.204 ssl_usesystemcerts](#)
[3.205 entropy_file](#)
[3.206 ssl_use_sslv2](#)
[3.207 ssl_use_sslv3](#)
[3.208 ssl_use_tlsv1](#)
[3.209 ssl_min_dh_prime_bits](#)
[3.210 ssl_ca_certificates_file](#)
[3.211 pipe_split](#)
[3.212 pipe_decode](#)
[3.213 pipe_sep](#)
[3.214 pop_authenticators](#)
[3.215 pop_auth_try_all](#)
[3.216 pop_checkinterval](#)
[3.217 pop_delete](#)
[3.218 pop_host](#)
[3.219 pop_last](#)
[3.220 pop_reconnect](#)
[3.221 pop_user](#)
[3.222 pop_pass](#)

[3.223 post_indent_string](#)
[3.224 postpone](#)
[3.225 postponed](#)
[3.226 preconnect](#)
[3.227 print](#)
[3.228 print_command](#)
[3.229 print_decode](#)
[3.230 print_split](#)
[3.231 prompt_after](#)
[3.232 query_command](#)
[3.233 query_format](#)
[3.234 quit](#)
[3.235 quote_regexp](#)
[3.236 read_inc](#)
[3.237 read_only](#)
[3.238 realname](#)
[3.239 recall](#)
[3.240 record](#)
[3.241 reply_regexp](#)
[3.242 reply_self](#)
[3.243 reply_to](#)
[3.244 resolve](#)
[3.245 reverse_alias](#)
[3.246 reverse_name](#)
[3.247 reverse_realname](#)
[3.248 rfc2047_parameters](#)
[3.249 save_address](#)
[3.250 save_empty](#)
[3.251 save_history](#)
[3.252 save_name](#)
[3.253 score](#)
[3.254 score_threshold_delete](#)
[3.255 score_threshold_flag](#)
[3.256 score_threshold_read](#)
[3.257 send_charset](#)
[3.258 sendmail](#)
[3.259 sendmail_wait](#)
[3.260 shell](#)
[3.261 sig_dashes](#)
[3.262 sig_on_top](#)
[3.263 signature](#)
[3.264 simple_search](#)
[3.265 smart_wrap](#)
[3.266 smileys](#)
[3.267 sleep_time](#)
[3.268 smtp_authenticators](#)
[3.269 smtp_pass](#)
[3.270 smtp_url](#)
[3.271 sort](#)
[3.272 sort_alias](#)
[3.273 sort_aux](#)
[3.274 sort_browser](#)
[3.275 sort_re](#)
[3.276 spam_separator](#)

[3.277 spoolfile](#)
[3.278 status_chars](#)
[3.279 status_format](#)
[3.280 status_on_top](#)
[3.281 strict_threads](#)
[3.282 suspend](#)
[3.283 text_flow](#)
[3.284 thread_received](#)
[3.285 thorough_search](#)
[3.286 tilde](#)
[3.287 time_inc](#)
[3.288 timeout](#)
[3.289 tmpdir](#)
[3.290 to_chars](#)
[3.291 tunnel](#)
[3.292 use_8bitmime](#)
[3.293 use_domain](#)
[3.294 use_envelope_from](#)
[3.295 use_from](#)
[3.296 use_idn](#)
[3.297 use_ipv6](#)
[3.298 user_agent](#)
[3.299 visual](#)
[3.300 wait_key](#)
[3.301 weed](#)
[3.302 wrap](#)
[3.303 wrap_search](#)
[3.304 wrapmargin](#)
[3.305 write_inc](#)
[3.306 write_bcc](#)

[4.功能](#)

[4.1 通用菜单](#)
[4.2 index菜单](#)
[4.3 pager菜单](#)
[4.4 别名菜单](#)
[4.5 query菜单](#)
[4.6 附件菜单](#)
[4.7 编辑菜单](#)
[4.8 postpone菜单](#)
[4.9 浏览器菜单](#)
[4.10 PGP的菜单](#)
[4.11 smime菜单](#)
[4.12 mix菜单](#)
[4.13 编辑器菜单](#)

1.命令行参数

运行不带参数的`mutt`，Mutt会尝试读取你的收件箱。但同样可以使用命令行让Mutt读取其他信箱或者发送一封邮件。

表8.1.命令行参数

参数	描述
-A	扩展一个别名
-a	附加文件到邮件
-b	指定密送（BCC）地址
-c	指定抄送（Cc）地址
-D	打印mutt所有变量的值到标准输出
-e	在配置文件被读取后，执行指定的命令
-f	加载指定的信箱
-F	指定一个muttrc文件
-h	打印命令行参数的帮助
-H	指定一个草稿文件，Mutt会从中读取邮件头和邮件体
-i	指定一个将被包含在邮件体中的文件
-m	指定默认的信箱类型
-n	不读取系统muttrc文件
-p	调出一个被暂缓的邮件
-Q	查询配置变量的值
-R	以只读模式打开信箱
-s	指定邮件的主题（如果包含空格，使用引号）
-v	查看版本号和编译选项
-x	simulate the mailx(1) compose mode
-y	显示（mailboxes所指定）信箱列表
-z	如果信箱中没有邮件则立即退出
-Z	打开第一个包含新邮件的文件夹，如果没有则立即退出

阅读信箱中的邮件

```
mutt [ -nz ] [ -F muttrc ] [ -m type ] [ -f mailbox ]
```

compose新邮件

```
mutt [ -n ] [ -F muttrc ] [ -a file ] [ -c address ] [ -i filename ] [ -s subject ] [ [ file ... ] - ]
address [ address ... ]
```

Mutt还支持以“batch”模式发送准备好的邮件。只是将你想要发送的文件重定向到标准输入。例如：

```
mutt -s "data set for run #2" professor@bigschool.edu < ~/run2.dat
```

这条命令会给“professor@bigschool.edu”发送一封主题为“ data set for run #2”的邮件。邮件的正文是文件“~/run2.dat”的内容。

`-a file`所带的所有文件将以MIME类型作为邮件的附件。要添加若干文件，使用“`--`”分隔文件和收件人地址：`mutt -a *.png -- some@one.org`

2.配置命令

以下是mutt中可用的命令。

- [account-hook](#) *pattern command*
- [alias](#) [`-group name ...`] *key address* [*address ...*]
- [unalias](#) [`-group name ...`] { **| key ...* }
- [alternates](#) [`-group name ...`] *regexp* [*regexp ...*]
- [unalternates](#) [`-group name ...`] { **| regexp ...* }
- [alternative-order](#) *mimetype* [*mimetype ...*]
- [unalternative-order](#) { **| mimetype ...* }
- [auto-view](#) *mimetype* [*mimetype ...*]
- [unauto-view](#) { **| mimetype ...* }
- [bind](#) *map key function*
- [charset-hook](#) *alias charset*
- [iconv-hook](#) *charset local-charset*
- [color](#) *object foreground background*

[color](#) { *header | body* } *foreground background regexp*

[color](#) *index foreground background pattern*
- [uncolor](#) *index pattern ...*
- [exec](#) *function* [*function ...*]
- [fcc-hook](#) *[!/]pattern mailbox*
- [fcc-save-hook](#) *[!/]pattern mailbox*
- [folder-hook](#) *[!/]regexp command*
- [group](#) [`-group name ...`] { `-rx expr ...` | `-addr expr ...` }
- [ungroup](#) [`-group name ...`] { **| -rx expr ...* | `-addr expr ...` }
- [hdr_order](#) *header* [*header ...*]
- [unhdr_order](#) { **| header ...* }

- [ignore](#) *pattern* [*pattern* ...]
- [unignore](#) { *| *pattern* ... }
- [lists](#) [-group *name*] *regexp* [*regexp* ...]
- [unlists](#) [-group *name* ...] { *| *regexp* ... }
- [macro](#) *menu key sequence* [*description*]
- [mailboxes](#) *mailbox* [*mailbox* ...]
- [unmailboxes](#) { *| *mailbox* ... }
- [mbox-hook](#) [!]*pattern mailbox*
- [message-hook](#) [!]*pattern command*
- [mime-lookup](#) *mimetype* [*mimetype* ...]
- [unmime-lookup](#) { *| *mimetype* ... }
- [mono](#) *object attribute*
- [mono](#) { header | body } *attribute regexp*
- [mono](#) index *attribute pattern*
- [unmono](#) index { *| *pattern* ... }
- [my_hdr](#) *string*
- [unmy_hdr](#) { *| *field* ... }
- [crypt-hook](#) *pattern keyid*
- [push](#) *string*
- [reset](#) *variable* [*variable* ...]
- [save-hook](#) [!]*pattern mailbox*
- [score](#) *pattern value*
- [unscore](#) { *| *pattern* ... }
- [reply-hook](#) [!]*pattern command*
- [send-hook](#) [!]*pattern command*
- [send2-hook](#) [!]*pattern command*
- [set](#) { [no | inv] *variable* | *variable=value* } [...]
- [unset](#) *variable* [*variable* ...]
- [source](#) *filename*

- [spam](#) *pattern format*
- [nosпам](#) { *| *pattern* }
- [subscribe](#) [-group *name* ...] *regexp* [*regexp* ...]
- [unsubscribe](#) [-group *name* ...] { *| *regexp* ... }
- [toggle](#) *variable* [*variable* ...]
- [unhook](#) { *| *hook-type* }

3.配置变量

3.1. abort_nosubject

类型：quadoption

默认值：ask-yes

如果设置为`yes`，编辑邮件时提示输入主题，不输入的话，编辑将被中止。如果设置为`no`，将不会中止邮件的撰写。

3.2. abort_unmodified

类型：quadoption

默认：yes

如果设置为`yes`，如果没有任何修改，编辑将自动中止（只有在文件第一次编辑后才进行此项检查）。当设置为`no`，编辑将不会中止。

3.3. alias_file

类型：路径

默认值：“~/muttrc”

默认的文件中保存着别名信息，被功能[<create-alias>](#)所创建。添加到该文件中的条目采用`$config_charset`指定的字符集编码，如果设置了的话。否则使用当前的字符集。

注意：Mutt不会自动source此文件。如果变量指向了一个专用的别名文件，你必须显试地使用命令“source”使其生效。

默认值为当前使用的muttrc文件，或“~/muttrc”（如果没有找到用户的muttrc文件）。

3.4. alias_format

类型：字符串

默认值：“%4n %2f %t %-10a %r”

指定别名菜单中数据显示的格式。有以下类似printf(3)的序列：

%a

别名

%f

标志——“d”表示一个标记为删除的别名

%n

序号

%r

名别包含的邮件地址

%t

表明别名是否被标记的符号

3.5. allow_8bit

类型：布尔

默认：yes

发送邮件时控制是否使用Quoted- Printable或Base64编码方式将8-bit数据转换为7-bit。

3.6. allow_ansi

类型：布尔

默认值：no

控制是否解析邮件中的ANSI颜色码（以及富文本格式里的颜色标签）。包含这些代码的邮件非常罕见，但如果设为yes的话，其文本将被设置为相应的颜色。请注意，这可能会覆盖你本身的颜色设置，而且会带来一些安全问题，因为邮件可能包含下面这一行

```
[-- PGP output follows ...
```

并并使用和你附件相同的颜色（见\$crypt_timestamp）。

3.7. arrow_cursor

类型：布尔

默认值：no

如果设置为yes，会使用箭头（“->”）来指示当前选择的条目，而不是高亮整个条目。在低速的网络中这个功能可以提高响应速度，因为当移动到下一个或上一个条目时这样可以更少地重画屏幕。

3.8. `ascii_chars`

类型：布尔

默认值：no

如果设置为yes，Mutt会用纯ASCII字符取代默认的ACS字符显示thread和附件树型结构。

3.9. `askbcc`

类型：布尔

默认值：no

如果设置为yes，在发送邮件时Mutt中会提示你输入密送地址（Bcc）。

3.10. `askcc`

类型：布尔

默认值：no

如果设置为yes，在发送邮件时Mutt中会提示你输入抄送地址（Cc）。

3.11. `assumed_charset`

类型：字符串

默认值：（空）

这个变量是一个以冒号分隔的字符编码方案清单，作用于那些没有指定字符编码的邮件。没有指明字符编码的邮件头字段和邮件体将按照清单中的一种编码方案来处理。默认情况下，没有指明字符编码的邮件头字段和邮件体都按“us-ascii”处理。

例如，日本用户可以这样设置：

```
set assumed_charset="iso-2022-jp:euc-jp:shift_jis:utf-8"
```

不过，对于邮件体，只有清单的第一值有效。

3.12. `attach_charset`

类型：字符串

默认值：（空）

这个变量是一个以冒号分隔的字符编码方案清单，作用于文本文件的附件。如果不设置，会用变量\$charset的值代替。例如,下面的配置适用于处理日文文本：

```
set attach_charset="iso-2022-jp:euc-jp:shift_jis:utf-8"
```

注：对于日本用户，如果包含“iso-2022-*”则必须放到最前面。

3.13. attach_format

类型： 字符串

默认：“%u%D%I %t%4n %T%.40d%> [%.7m/%.10M, %.6e%?C?, %C?, %s]“

这个变量描述了“附件”菜单的格式。有以下类似printf(3)的序列可用：

%C

字符集

%c

需要字符集转换（“n”或“c”）

%D

删除标记

%d

描述

%e

MIME内容传输编码

%f

文件名

%I

disposition（“I”表示inline，“A”表示attachment）

%m

MIME主类型

%M

MIME子类型

%n

附件数量

%Q

如果该MIME类型参与了附件计数，则显示“Q”。

%s

大小

%t

标记位

%T

图形树字符

%u

unlink (=to delete) flag

%X

number of qualifying MIME parts in this part and its children (关于速度的影响，请参见“附件搜索和计数”部分)

%>X

右对齐余下的字符串，使用字符“X”填充

%lX

使用字符“X”填充至行末

/*X

使用字符“X”软填充

关于“软填充”的解释，见\$index_format部分。

3.14. attach_sep

类型：字符串

默认值：“\n”

操作（保存，打印，pipe等）一组附件时，附件之间的分隔符。

3.15. attach_split

类型：布尔

默认：yes

如果设置为`no`，当操作（保存，打印，pipe等）一组附件时，Mutt会将附件连接起来并把他们视为一个单独的附件来处理。`$attach_sep`定义的分隔符会添加到每个附件之后。设置为`yes`，Mutt会逐一处理每个附件。

3.16. attribution

类型：字符串

默认值：“On %d, %n wrote:”

回复邮件时这个字符串会放在最前面。要获得完整的类似`printf(3)`序列的清单，参见小节 `$index_format`。

3.17. autoedit

类型：布尔

默认值：no

当连同`$edit_headers`一起设置为`yes`时，Mutt会跳过发送菜单（提示你输入主题和收件人的地方），使你可以立即开始编辑邮件。当你完成编辑邮件时发送菜单可能还会跳出来一次。

可参见`$fast_reply`。

3.18. auto_tag

类型：布尔

默认值：no

当设置为`yes`时，`index`菜单中所有作用于单个邮件的功能将自动应用到全部被标记的邮件上（如果有的话）。设置为`no`时，你必须使用功能`<tag-prefix>`（默认绑定到“;”）使下一个功能作用于所有被标记的邮件。

3.19. beep

类型：布尔

默认：yes

当设置为`yes`，mutt会在发生错误时发出报警音。

3.20. beep_new

类型：布尔

默认值：no

当这个变量设置为`yes`时，mutt will beep whenever it prints a message notifying you of new mail。这个变量是独立于变量`$beep`的设置。

3.21. bounce

类型：quadoption

默认值：ask-yes

控制是否让你确认bounce邮件。如果设置为`yes`，会直接bounce邮件，不会让你确认。设置此变量为`no`一般没有用处，因而不推荐这样设置，因为你将无法bounce邮件。

3.22. bounce_delivered

类型：布尔

默认：yes

设置此变量为`yes`，在bounce邮件时mutt会自动添加Delivered-To字段。Postfix的用户可能希望将这个变量设置为`no`。

3.23. braille_friendly

类型：布尔

默认值：no

设置为`yes`时，mutt将光标放置在菜单当前行的开始初，即使变量`$arrow_cursor`设置为`no`，这样可以方便盲人用Braille显示器跟随这些菜单。默认情况下这个变量设置为`no`，because many visual terminals don't permit making the cursor invisible。

3.24. check_mbox_size

类型：布尔

默认值：no

当这个变量设置为`yes`时，mutt会利用文件大小来检查mbox和mmdf文件夹中的新邮件，而不利用访问时间检查。

此变量默认为`no`，并且只应在这些文件夹的类型的邮件检测不可靠或不能工作时才启用此功能。

请注意，对于mbox和mmdf类型的文件夹，应该在使用命令“mailboxes”之前启用此变量。因为mutt需要快速扫描这些信箱以确定它们的初始状态。此后新邮件的状态将利用文件大小的变化进行跟踪。

3.25. charset

类型：字符串

默认值：（空）

你的终端显示和输入文本数据时使用的字符集。这也是`$send_charset`的备选值。

Mutt在启动时会尝试从环境变量获得这个值，如`$LC_CTYPE`或`$LANG`。

注意：只应在Mutt不能确定用什么字符集来正常显示时才设置这个变量。

3.26. check_new

类型：布尔

默认：yes

注意：此选项只对`maildir`和`MH`有效。

当设置为`yes`时，Mutt会在信箱打开时进行新邮件检查。特别对MH信箱，此操作可能会花相当长的时间，因为它需要扫描目录以及检查每个文件，已确定它们是否已经被查看过。如果这个变量设置为`no`，当信箱打开时不会进行新邮件检查。

3.27. collapse_unread

类型：布尔

默认：yes

当设置为`no`时，Mutt不会折叠包含未读邮件的thread。

3.28. uncollapse_jump

类型：布尔

默认值：no

当设置为`yes`，如果当前的thread没有折叠，Mutt会跳转到下一封未读邮件，如果有的话。

3.29. compose_format

类型：字符串

默认值：“-- Mutt: Compose [Approx. msg size: %l Atts: %a]%>-“

控制“编辑”菜单中状态栏显示的格式。这个字符串类似`$status_format`，但也有它自己类`printf(3)`序列：

`%a`

附件总数

`%h`

本地主机名

`%l`

当前邮件的近似大小（以字节为单位）

`%v`

Mutt版本

见`$status_format`的描述获得有关如何设置`$compose_format`的更多信息。

3.30. config_charset

类型：字符串

默认值：（空）

设定后，Mutt会将rc文件从这个字符集重新编码为当前字符集（变量\$charset设置），并使用这个字符集将别名写入\$alias_file指定的别名文件。

请注意，如果设置这个变量，那么之前一定要先设置变量\$charset。

应当避免重新编码，as it may render unconvertable characters as question marks which can lead to undesired side effects (for example in regular expressions)。

3.31. confirmappend

类型：布尔

默认：yes

设置为yes，将邮件附加到现有信箱时Mutt会提示确认。

3.32. confirmcreate

类型：布尔

默认：yes

设置为yes，当保存邮件到一个不存在的信箱中时，Mutt会在创建信箱之前提示确认。

3.33. connect_timeout

型号：数值

默认值：30

Mutt在经过这么多秒后还不能建立（到IMAP，POP或SMTP的）连接就判断为超时。若为负值，mutt将无限等待直到连接成功建立。

3.34. content_type

类型：字符串

默认值：“text/plain”

设置编写新的邮件时邮件体默认的Content-Type。

3.35. copy

类型：quadoption

默认：yes

这个变量控制是否备份你发出的邮件。也参见\$record，\$save_name，\$force_name和“fcc-hook”。

3.36. crypt_use_gpgme

类型：布尔

默认值：no

此变量控制使用GPGME-enabled加密后端。如果设为yes，而且Mutt编译进了gpgme支持，那么对于S/MIME和PGP加密，会优先使用gpgme方式。请注意，你需要在.muttrc里设置这个选项;在交互模式下设置它不会有任何效果。

3.37. crypt_use_pka

类型：布尔

默认值：no

控制在签名验证时是否使用PKA（见<http://www.g10code.de/docs/pka-intro.de.pdf>）（只被GPGME后端支持）。

3.38. crypt_autopgp

类型：布尔

默认：yes

这个变量控制Mutt是否自动启用PGP加密/签名邮件。另见\$crypt_autoencrypt，\$crypt_replyencrypt，\$crypt_autosign，\$crypt_replysign和\$smime_is_default。

3.39. crypt_autosmime

类型：布尔

默认：yes

这个变量控制Mutt是否自动启用S/MIME加密/签名邮件。另见\$crypt_autoencrypt，\$crypt_replyencrypt，\$crypt_autosign，\$crypt_replysign和\$smime_is_default。

3.40. date_format

类型：字符串

默认值：“!%a, %b %d, %Y at %I:%M:%S%p %Z”

此变量控制变量\$index_format“的序列%d”所打印的日期的格式。它传递给函数strftime(3)来处理日期，正确的语法见手册页。

月份和星期名称根据变量\$locale指定的值来扩展，除非字符串的第一个字符串是感叹号（“!”）。如果字符串的第一个字符感叹号，这个感叹号将被丢弃，并字符串中的月份和星期名称，将按照C locale（即美国英语）来扩展。

3.41. default_hook

类型：字符串

默认值：“~f %s !~P | (~P ~C %s)”

此变量控制“message-hook”，“[reply-hook](#)”，“send-hook”，“[send2-hook](#)”，“save-hook”和“fcc-hook”将怎样被解释，如果它们用正则表达式代替了模式匹配。hook在声明的时候被扩展，所以hook在声明的时候就根据这个变量的值被解释了。

默认值会匹配来自正则表达式匹配的发件人或发送/抄送的收件人的邮件。

3.42. delete

类型：quadoption

默认值：ask-yse

控制在退出或同步信箱时是否真的删除邮件。如果设置为yes，标记为删除的邮件会被mutt自动清除，没有任何提示。如果设置为no，标记为删除的邮件将被保存在信箱中。

3.43. delete_untag

类型：布尔

默认：yes

如果这个选项设为yes，mutt不会标记被删除的邮件。它应用于你显式地删除一封邮件，或移动一封邮件到别的信箱。

3.44. digest_collapse

类型：布尔

默认：no

如果这个选项被设为yes，mutt不会在附件菜单上显示multipart/digest的子部份。要查看它们的子部份，在菜单上按“v”。

3.45. display_filter

类型：路径

默认值：（空）

设置指定一个命令来过滤邮件。当邮件查看时，邮件先通过标准输入到\$display_filter，然后再从其标准输出被读取。

3.46. dotlock_program

类型：路径

默认：“/usr/local/bin/mutt_dotlock”

包含mutt_dotlock(8)程序的路径。

3.47. dsn_notify

类型：字符串

默认值：（空）

此变量设置投递状态的请求。该字符串为一个以逗号分隔（没有空格）的列表，包含了以下一些值：*never*，永远不请求回执。*failure*，传输失败时请求回执。*delay*，邮件延迟是被告知。*success*，成功传输时被告知。

示例：

```
set dsn_notify="failure,delay"
```

注：使用\$sendmail来投递邮件，你需要使用sendmail 8.8.x，或更高版本，一个支持DSN并兼容sendmail(1)的命令行选项-N的MTA。对于SMTP投递，会自动检测DSN支持，所以它取决于服务器上是否使用了DSN。

3.48. dsn_return

类型：字符串

默认值：（空）

此变量控制DSN回执单返回邮件的哪一部分。它可以被设定为*hdrs*：仅返回邮件头，或*full*：返回整封邮件。

示例：

```
set dsn_return=hdrs
```

注：使用\$sendmail来投递邮件，你需要使用sendmail 8.8.x，或更高版本，一个支持DSN并兼容sendmail(1)的命令行选项-R的MTA。对于SMTP投递，会自动检测DSN支持，所以它取决于服务器上是否使用了DSN。

3.49. duplicate_threads

类型：布尔

默认：yes

这个变量控制在\$sort设置为*threads*时，mutt是否将具有相同Message-ID放在同一个会话内。If it is *set*, it will indicate that it thinks they are duplicates of each other with an equals sign in the thread tree.

3.50. edit_headers

类型：布尔

默认值：no

此变量允许你连同邮件体一起编辑邮件头。

注意，因为通用性的原因，对“References:”和“Date:”字段的更改将被忽略。

3.51. editor

类型： 路径

默认： （空）

此变量指定mutt所使用的编辑器。它默认的值是环境变量中的\$VISUAL或\$EDITOR，如果这两个环境变量都没有设置的话，就是字符串“vi”。

3.52. encode_from

类型： 布尔

默认值： no

When *set*, mutt will quoted-printable encode messages when they contain the string “From ” (note the trailing space) in the beginning of a line. This is useful to avoid the tampering certain mail delivery and transport agents tend to do with messages (in order to prevent tools from misinterpreting the line as a mbox message separator).

3.53. envelope_from_address

类别： e-mail地址

默认值： （空）

手动设置发送邮件的*envelope*发件人。如果\$use_envelope_from未被设置，此变量将被忽略。

3.54. escape

类型： 字符串

默认值： “~”

在内建编辑器中使用function的转义字符。

3.55. fast_reply

类型： 布尔

默认值： no

当设置为*yes*时，当回复邮件时，mutt不会提示你输入收件人和主题，转发邮件时mutt不会提示你输入主题。

注：当是设置了变量\$autoedit，此变量不受影响。

3.56. fcc_attach

类型： 布尔

默认： yes

这个变量控制是否将已发邮件中的附件和邮件体一起保存。

3.57. fcc_clear

类型：布尔

默认值：no

当这个变量设为yes，FCCs将不会以加密/签名的方式来储存邮件，即使实际上邮件是被加密/签名的。（只对PGP）

3.58. folder

类型：路径

默认值：“~/Mail”

指定默认的信箱路径。在mutt中的路径以“+”或“=”开头时，会被扩展成此变量的值。注意，如果你更改此变量的值，确保更改发生使用“+”或“=”之前。快捷符号的扩展发生在处理命令“mailboxes”时。

3.59. folder_format

类型：字符串

默认值：“%2C %t %N %F %2l %-8.8u %-8.8g %8s %d %f

这个变量可以让你根据自己的习惯自定义的文件浏览器的显示风格。这个字符串类似变量\$index_format，但有它自己的一套类printf(3)序列：

%C

当前文件号

%d

最后修改的日期/时间

%f

文件名

%F

文件权限

%g

所属组（如果缺失的话显示GID）

%l

硬链接的数量

%N

如果文件夹中有新邮件，显示"N"，否则留空

%s

文件大小，以字节为单位

%t

如果文件被标记，显示“*”，否则留空

%u

所有者（如果哦缺失，显示UID）

%>X

右对齐余下的字符串，使用字符“X”填充

%lX

使用字符“X”填充至行末

/*X

使用字符“X”软填充

关于“软填充”，见\$index_format。

3.60 followup_to

类型：布尔

默认：yes

控制发送邮件时是否生成“Mail-Followup-To:”字段。设置成yes的话，当你回复到一个命令“[subscribe](#)”或“lists”指定的邮件列表时，mutt会生成此字段。

这个字段有两个目的。首先，防止你收到你发给邮件列表的邮件的副本；第二，ensuring that you do get a reply separately for any messages sent to known lists to which you are not subscribed。

对于订阅的邮件列表，该字段只包含列表的地址。对于没有订阅的邮件列表，会包含列表的地址和你的邮件地址。没有这个字段的话，对那些发送到一个已订阅列表的邮件进行group reply时，将同时发送到列表和对方的地址。造成对方收到两份一样的邮件。

3.61 force_name

类型：布尔

默认值：no

此变量类似\$save_name，不同的是Mutt会以收件人地址的用户名为名称保存一份已发邮件的副本，即使信箱并不存在。

参见变量\$record。

3.62 forward_decode

类型：布尔

默认：yes

控制当转发一封MIME邮件，将其解码为text/plain格式。The message header is also RFC2047 decoded。这个变量只有在变量\$mime_forward为no的时候使用，否则将为使用变量\$mime_forward_decode。

3.63 forward_edit

类型：quadoption

默认：yes

这个变量控制在用户转发邮件时，是否自动调出编辑器编辑。对于那些总是不加任何修改地转发邮件的人，应该设定为“no”。

3.64 forward_format

类型：字符串

默认：“[%a: %s]”

此变量控制转发邮件时的默认主题。它使用和变量\$index_format相同的序列。

3.65 forward_quote

类型：布尔

默认值：no

当设为yes，在以正文方式转发邮件时（当变量\$mime_forward为no），会使用变量\$indent_string设置的字符引用被转发的邮件。

3.66 from

类别：e-mail地址

默认值：（空）

当设置后，此变量包含一个默认的地址。使用“my_hdr”（including from a “send-hook”）和\$reverse_name可以将它覆盖。如果变量\$use_from未被设置，此变量会被忽略。

变量默认值为环境变量\$EMAIL的值。

3.67 gecos_mask

类型：正则表达式

默认值：“`^[^,]*`”

括展别名的时候，`mutt`用来解析GECOS字段的正则表达式。默认值会返回到第一个“,”时的字符串。如果GECOS field包含一个字符串，形式为“`lastname, firstname`”，那么你应该将其设置为“`.*`”。

这是有用的，如果你遇到以下情况：你给用户“`stevef`”写信，其全称是“`Steve Franklin`”。如果`mutt`把“`stevef`”扩展成“`Franklin`”，那么你应该设置`$gecos_mask`为一个能匹配整个名字的正则表达式，以便`mutt`能将“`Franklin`”扩展成“`Franklin, Steve`”。

3.68 hdrs

类型：布尔

默认：yes

当设置为`no`，命令“`my_hdr`”添加的邮件头字段便不会被创建。此变量必须在撰写新邮件或回复邮件之前被设置为“`no`”，以便立即生效。如果设为`yes`，用户定义的邮件头字段会添加到每一封新邮件。

3.69 header

类型：布尔

默认值：no

当设为`yes`，会使`mutt`将你回复的邮件的邮件头包含到编辑缓冲区中。The `$weed` setting applies。

3.70 help

类型：布尔

默认：yes

当设置为`yes`时，屏幕上第一行会显示帮助栏。

注：将功能绑定到一个按键序列而非一个按键上的绑定将不会显示在帮助栏上。此外，在`Mutt`正在运行时改变绑定，帮助栏可能不会更新。因为这个变量主要是针对新用户的，所以没有它会麻烦。

3.71 hidden_host

类型：布尔

默认值：no

When *set*, `mutt` will skip the host name part of `$hostname` variable when adding the domain part to addresses. This variable does not affect the generation of Message-IDs, and it will not lead to the cut-off of first-level domains.

3.72 hide_limited

类型：布尔
默认值：no

当设置为yes，在一个thread中，mutt将不会显示被“提取”功能隐藏的邮件。

3.73 hide_missing

类型：布尔
默认：yes

设置为yes，mutt will not show the presence of missing messages in the thread tree。

3.74 hide_thread_subject

类型：布尔
默认：yes

设置为yes时，mutt不会显示thread树中与父邮件或上一封邮件主题相同的邮件的主题。

3.75 hide_top_limited

类型：布尔
默认值：no

设置为yes，mutt不会显示被“提取”功能所隐藏的、位于一个thread顶部的邮件。请注意，当变量\$hide_limited为yes时，此选项无效。

3.76 hide_top_missing

类型：布尔
默认：是

当设置，狗就不会显示存在遗失的邮件上方的线程在线程树。请注意，当[\\$hide_missing](#) 设置，此选项将无效。

3.77 history

型号：数值
默认值：10

此变量控制每一类字符串历史记录缓存的大小（保存字符串的数量）。每次设置此变量时，缓存会清除。

3.78 history_file

类型：路径
默认值：“~/mutthistory

mutt将历史记录保存到此文件中。

3.79 honor_followup_to

类型：quadoption

默认：yes

这个变量控制在group-replying邮件时，是否honor Mail-Followup-To字段。

3.80 hostname

类型：字符串

默认值：（空）

指定mutt所在主机的完整的主机名，包含主机名和其所属的DNS域名。它用于本地邮件的域名部分（即“@”之后的部分），也用于Message-Id字段。

其值在启动的时候确定：如果uname(3)函数所返回的节点名称中包含主机名和域名，these are used to construct \$hostname。如果返回域名部分，Mutt将在“/etc/resolv.conf”中查找“domain”或“search”来确定域名。或者，也可以在编译时使用一个固定的域名，以免Mutt查找到的域名是不可用的。

另见\$use_domain和\$hidden_host。

3.81 ignore_linear_white_space

类型：布尔

默认值：no

This option replaces linear-white-space between encoded-word and text to a single space to prevent the display of MIME-encoded “Subject:” field from being divided into multiple lines.

3.82 ignore_list_reply_to

类型：布尔

默认值：no

在回复一个邮件列表（由命令“[subscribe](#)”或“lists”定义）时，此变量影响function <reply>的行为。当设置为yes，如果“Reply-To:”字段的值和“To:”字段一样，Mutt假设“Reply-To:”字段是由邮件列表设定的（用于自动回复到邮件列表），并忽略这个字段。当这个变量设为yes时要直接回复到邮件列表，使用功能<[list-reply](#)>; 功能<group-reply>会同时回复到发件人和邮件列表。

3.83 imap_authenticators

类型：字符串

默认值：（空）

这是一个用冒号分隔的列表，包含用于登录IMAP服务器的验证方法，mutt会按顺序尝试它们。

验证方法是“login”或IMAP “AUTH=xxx”字符串的右边部份，如“digest-md5”，“gssapi”或“cram-md5”。此变量不区分大小写。如果不设置（默认）mutt将由从高到低的安全等级尝试所有可用的方法。

示例

```
set imap_authenticators="gssapi:cram-md5:login"
```

注：只有在上一个方法不可用时Mutt才会尝试下一个验证方法。如果一个方法可用但验证失败，mutt将无法连接到IMAP服务器。

3.84 imap_check_subscribed

类型：布尔

默认值：no

当设置为yes，mutt会从服务器抓取你订阅的文件夹，并查询其中的新邮件，就像你使用命令“mailboxes”一样。

3.85 imap_delim_chars

类型：字符串

默认：“/.”

这个选项包含一个字符列表，这些字符在显示IMAP路径时作为文件夹分隔符。特别在使用快捷符号“=”指定变量*folder*时它特别有用。

3.86 imap_headers

类型：字符串

默认值：（空）

除了默认的字段（“Date:”，“From:”，“Subject:”，“To:”，“Cc:”，“Message-Id:”，“References:”，“Content-Type:”，“Content-Description:”，“In-Reply-To:”，“Reply-To:”，“Lines:”，“List-Post:”，“X-Label:”），mutt在显示index菜单前需要从IMAP服务器上抓取的邮件头字段。你可能想加入更多字段供垃圾邮件检测。

注：这是一个空格分隔的列表，项目应大写且不含冒号，如“X-BOGOSITY X-SPAM-STATUS”表示“X-Bogosity:”和“X-Spam-Status:”字段。

3.87 imap_idle

类型：布尔

默认值：no

当设为yes时，mutt将尝试使用IMAP IDLE扩展在当前信箱中检查新邮件。有些服务器（dovecot was the inspiration for this option）不能很好的响应mutt请求。如果你经常遇到连接周期性的僵

死，试试设置此选项为no。

3.88 imap_keepalive

类型：数值

默认值：900

This variable specifies the maximum amount of time in seconds that mutt will wait before polling open IMAP connections, to prevent the server from closing them before mutt has finished with them. The default is well within the RFC-specified minimum amount of time (30 minutes) before a server is allowed to do this, but in practice the RFC does get violated every now and then. Reduce this number if you find yourself getting disconnected from your IMAP server due to inactivity.

3.89 imap_list_subscribed

类型：布尔

默认值：no

此变量配置只显示订阅的文件夹或显示所有文件夹。在IMAP文件浏览器里可以使用功能<toggle-subscribed>来切换这个选项。

3.90 imap_login

类型：字符串

默认值：（空白）

你用来IMAP服务器的用户名。

这个变量的默认值为\$imap_user的值。

3.91 imap_pass

类型：字符串

默认值：（空）

指定IMAP帐户的密码。如果没有设置，当你调用功能[<fetch-mail>](#)或者打开一个IMAP文件夹时，Mutt中会提示你输入密码。

警告：你只应该在一台比较安全的机器上使用此选项，因为超级用户可以读取你的muttrc，即使你是唯一一个具有权限读取该文件的人。

3.92 imap_passive

类型：布尔

默认：yes

当设为yes，mutt不会打开一个新的IMAP连接来检查新邮件。Mutt将在以建立的连接上进行新邮件检查。如果你不想再次输入用户名/密码，或者建立链接非常缓慢时，这个选项是很有用的。

3.93 imap_peek

类型：布尔

默认：yes

当设为yes，当你从服务器抓取邮件时，mutt不会隐式地标记邮件为已读。一般这是很有用的，但它会使关闭IMAP文件夹有点缓慢。This option exists to appease speed freaks。

3.94 imap_pipeline_depth

型号：数量

默认值：15

控制IMAP命令在发送到服务器前可以参与排队的数量。比较大的排队数量能够降低mutt等待服务器响应的时间，而且可以使得IMAP服务器反应更快。但并非所有的服务器都能够正确处理排队命令，因此，如果你遇到问题，你可能需要尝试将此选项设置为0。

注：此变量的更改不会影响到已打开的连接。

3.95 imap_servernoise

类型：布尔

默认：yes

为yes时，mutt会将IMAP服务器的警告信息作为错误信息显示。这些信息往往是无害的，而且一般是由超出用户能力范围的服务器配置问题引所起的，所以你随时可以禁用它。

3.96 imap_user

类型：字符串

默认值：（空）

你需要访问IMAP服务器的用户名。

此变量默认为你本地计算机上的用户名。

3.97 implicit_autoview

类型：布尔

默认值：no

如果设置为“yes”，对每一个没有定义内部显示的MIME附件，mutt会查找mailcap中对应的包含“copiousoutput”的条目。如果找到这样的条目，mutt会使用这个条目中定义的浏览器将文件转换成文本形式。

3.98 include

类型： quadoption

默认值： ask-yes

控制是否在回复时包含邮件原文。

3.99 include_onlyfirst

类型： 布尔

默认值： no

控制是否在回复时包含邮件的第一个附件。

3.100 indent_string

类型： 字符串

默认值： “> “

指定邮件引文每一行之前的字符串。强烈建议你不要修改这个值， as it tends to agitate the more fanatical netizens。

此选项是一个格式字符串， 请参阅变量\$index_format的描述。

对于format=lowed样式， 引用机制被完整的定义了。所以如果变量\$text_flow为yes的话， 此选项将被忽略。

3.101 index_format

类型： 字符串

默认值： “%4C %Z %{%b %d} %-15.15L (%?l?%4l&%4c?) %s“

这个变量可以让你根据自己的喜好来自定index菜单的显示。

“格式字符串”类似C语言的printf(3)函数格式化输出（见man page获得细节）时所使用的字符串。Mutt中定义的序列有：

%a

作者地址

%A

reply-to地址（如果有的话， 否则显示作者的地址）

%b

邮件文件夹（信箱）的文件名

%B

the list to which the letter was sent, or else the folder name (%b)。

%c

邮件的字节数

%C

邮件号码

%d

邮件的日期和时间信息，格式由变量\$date_format指定，会转为发件人的时区

%D

邮件的日期和时间信息，格式由变量\$date_format指定，转换为本地时区

%e

邮件在thread中的号码

%E

当前thread的邮件数量

%f

发件人（地址+姓名），From:或者Return-Path:的值

%F

作者姓名，或收件人姓名（如果邮件由你发出）

%H

垃圾邮件属性

%i

当前邮件的message-id

%l

邮件的行数（在maildir，mh，IMAP文件夹中有可能工作不正常）

%L

如果“To:或Cc:”字段中的地址与命令“[subscribe](#)”指定的地址一致，则显示“To <list-name>”，否则和%F一样。

%m

信箱中邮件的总数

%M

隐藏邮件的数量，如果thread折叠起来。

%N

邮件评分

%n

作者的姓名（如果缺失，显示地址）

%O

original save folder where mutt would formerly have stashed the message: list name or recipient name if not sent to a list

%P

内置pager的进度指示器（文件显示了多少）

%s

邮件主题

%S

邮件状态（“N”/“D”/“d”/“!”/“r”/*）

%t

“To:”字段（收件人）

%T

the appropriate character from the \$to_chars string

%u

作者的用户名

%v

作者或者收件人（如果邮件是由你发出的）的名字

%X

附件的数量（请参见“附件”一节了解关于速度的影响）

%y

“X-Label:”字段，如果有的话

%Y

“X-Label:”字段，if present，and （1）not at part of a thread tree，（2）at the top of a

thread, or (3) “X-Label:”is different from preceding message's “X-Label:”

%Z

邮件状态标志

%{fmt}

邮件的日期和时间转换为发件人的时区，“fmt”被库函数`strftime(3)`扩展;感叹号开头禁用locale

%[fmt]

邮件的日期和时间转换为本地时区，“fmt”被库函数`strftime(3)`扩展;感叹号开头禁用locale

%(fmt)

收到邮件的本地日期和时间。“fmt”被库函数`strftime(3)`扩展;感叹号开头禁用locale

%<fmt>

当前本地时间。“fmt”被库函数`strftime(3)`扩展;感叹号开头禁用locale

%>X

右对齐余下的字符串，使用字符“X”填充

%lX

使用字符“X”填充至行末

%*X

使用字符“X”软填充

“软填充”需要解释一下：正常情况下使用“%>”的右对齐会照常显示左边的内容，仅仅在有空间时才显示填充的内容。相比之下，软填充优先显示右边的内容，保证了显示右边内容所需的空
间，同样的，仅仅在有空间时才显示填充的内容。如果必要，软填充会吃掉左边的内容，以保证有足够的空间来显示右边的内容。

注意，这些序列在命令“save-hook”、“fcc-hook”、“fcc-save-hook”中也可用。

3.102 ispell

类型：路径

默认值：“ispell”

如何调用ispell（GNU的拼写检查软件）。

3.103 keep_flagged

类型：布尔

默认值：no

如果设为yes，当被标记为重要的邮件被阅读后，邮件不会从你的spool信箱移动到\$mbox信箱，命令mbox-hook也不对这些邮件生效。

3.104 locale

类型：字符串

默认值：“c”

strftime(3)格式化日期所用的locale。合法的值是你的系统接受的用于locale环境变量\$LC_TIME的字符串。

3.105 mail_check

型号：数值

默认值：5

此变量配置mutt检查新邮件的频率（以秒为单位）。也见变量\$timeout。

3.106 mailcap_path

类型：字符串

默认值：（空）

此变量指定尝试显示不直接被mutt支持的MIME文件时所需要的文件。

3.107 mailcap_sanitize

类型：布尔

默认：yes

设置为yes/0}，mutt will restrict possible characters in mailcap % expandos to a well-defined set of safe characters。This is the safe setting, but we are not sure it doesn't break some more advanced MIME stuff。

不要更改此设置，除非你真的知道自己在做什么！

3.108 header_cache

类型：路径

默认值：（空）

这个变量指向邮件头缓存的数据库。如果指向一个目录，Mutt会让每个信箱使用一个缓存数据库，如果指向一个文件则所有的信箱使用一个缓存数据库。默认情况下没有被设置，所以不使用邮件头缓存。

标题缓存可以大大提高打开POP，IMAP,MH或Maildir文件夹的速度，见“本地缓存”获得细节。

3.109. maildir_header_cache_verify

类型：布尔

默认：yes

对于Maildir，当邮件头缓存正在被使用时，检查是否有mutt以外的程序在修改maildir文件。文件夹被打开时，会为每一封邮件调用一个`stat(2)`。（对于NFS文件夹可能非常缓慢）。

3.110. header_cache_pagesize

类型：字符串

默认值：“16384”

当mutt编译进了gdbm或bdb4作为邮件头缓存后端时，这个选项可以设置数据库页面大小。太大或太小的值会浪费空间，内存或CPU。通常情况下默认值就合适了。

3.111. header_cache_compress

类型：布尔

默认：yes

当mutt编译了qdbm或tokyocabinet作为邮件头缓存后端，此选项用来设置该数据库是否被压缩。压缩可以使数据库文件只占用约1/5的磁盘空间，但解压缩可能会导致打开缓存文件夹非常缓慢（其速度也远远快过不使用邮件头缓存）。

3.112 maildir_trash

类型：布尔

默认值：no

如果设为yes，标记为被删除的邮件将被保存下来而不是被删除。注：这仅适用于maildir风格的邮箱。此设置将不会影响其他类型的信箱。

3.113 mark_old

类型：布尔

默认：yes

控制当你没有阅读新邮件而退出一个信箱时，mutt是否将这些未读邮件标记为*旧的*。此选项设为yes，下次启动mutt，这些邮件在indexindex菜单中将被标记为“O”，表示它们是旧的。

3.114 markers

类型：布尔

默认：yes

控制内部pager中折行的显示。设置为yes，所有折行的开始处显示一个“+”。

可见变量\$smart_wrap。

3.115 mask

类型：正则表达式

默认值：“![^]\.[[^].]“

用于文件浏览器的正则表达式，可以使用*逻辑非*操作符“!”开头。只有文件名与表达式匹配的文件才会被显示。匹配始终是区分大小写的。

3.116 mbox

类型：路径

默认值：“~/mbox”

This specifies the folder into which read mail in your \$spoolfile folder will be appended。

也见变量\$move。

3.117 mbox_type

类型：folder magic

默认值：mbox

默认信箱类型用于创建新文件夹时使用。可以是“mbox”，“MMDF”，“MH”以及“Maildir”中的任何一个。

3.118 metoo

类型：布尔

默认值：no

设置为*no*，当回复邮件时Mutt会从收件人列表中删除你的地址（请参阅命令“alternates”）。

3.119 menu_context

型号：数值

默认值：0

此变量控制滚动菜单时显示上下文的行数。（类似\$pager_context）

3.120 menu_move_off

类型：布尔

默认：yes

当设置为`no`，最后一个条目永远不会离开菜单底部，除非条目少于行数。当设为`yes`，最后一个条目可能离开底部。

3.121 menu_scroll

类型：布尔

默认值：yes

当设置为`yes`，当你试图将光标移动到屏幕边界以外，菜单会一行一行地上下移动。如果设为`no`，菜单会清屏并显示下一页或上一页（对慢速的连接很有用，避免了许多重画屏幕）。

3.122 meta_key

类型：布尔

默认值：no

If *set*, forces Mutt to interpret keystrokes with the high bit (bit 8) set as if the user had pressed the Esc key and whatever key remains after having the high bit removed. For example, if the key pressed has an ASCII value of 0xf8, then this is treated as if the user had pressed Esc then “x”. This is because the result of removing the high bit from 0xf8 is 0x78, which is the ASCII character “x”.

3.123 mh_purge

类型：布尔

默认值：no

当设置为`no`，mutt会模仿mh的行为，在mh文件夹中重新命名被标记删除的邮件为“,<*old file name*>”，而不是真正地删除它们。邮件保留在磁盘上，但会使读文件夹的程序忽略它。如果设为`yes`，邮件将会被简单地删除。

这个变量就像\$maildir_trash之于Maildir文件夹。

3.124 mh_seq_flagged

类型：字符串

默认值：“flagged”

MH序列中已标记邮件的名字

3.125 mh_seq_replied

类型：字符串

默认值：“replied”

MH序列中用于标记已回复的邮件的名字。

3.126 mh_seq_unseen

类型：字符串
默认值：“unseen”

MH序列中用于标记unseen邮件的名字。

3.127 mime_forward

类型：quadoption
默认值：no

当设置为yes，你转发的邮件将会被当作message/rfc822格式的附件发出，而不包含在邮件的正文中。这可以使收件人如同你收到邮件时那样完整地查看MIME邮件。如果你需要在MIME和非MIME之间选择，设置此变量为“ask-no”或“ask-yes”。

参见\$forward_decode和\$mime_forward_decode。

3.128 mime_forward_decode

类型：布尔
默认值：no

当变量\$mime_forward设为yes时，此选项控制当转发MIME邮件时，将邮件解码为text/plain格式。如果\$mime_forward为no，那么将使用变量\$forward_decode。

3.129 mime_forward_rest

类型：quadoption
默认：yes

若设置为yes，当转发一封MIME邮件中的多个附件时，一些不能被解码的附件将被添加到新的邮件中。

3.130 mix_entry_format

类型：字符串
默认值：“%4n %c %-16s %a”

这个变量描述了mixmaster链选择菜单的格式。下面是其支持的类printf(3)样式序列：

%n

序号

%c

转投系统的能力

%s

转投系统的缩略名

%a

转投系统的e-mail地址

3.131 mixmaster

类型： 路径

默认值：“mixmaster”

此变量设置系统中Mixmaster程序所在的路径。你需要利用一些参数来使用已知的转投系统，最终通过mixmaster链发送邮件。

3.132 move

类型： quadoption

默认值： no

控制Mutt是否将已读读邮件从spool信箱移动到\$mbox信箱， or as a result of a “mbox-hook” command。

3.133 message_cachedir

类型： 路径

默认： （空）

设置此变量指向目录，mutt会将你的IMAP和POP服务器上的邮件副本缓存到这里。你可以在任何时候删除里面的内容。

当此变量指向目录，远程服务器上每一封邮件mutt只会抓取一次，并可以执行正则表达式来快速搜索本地文件夹。

可以参见变量\$message_cache_clean。

3.134 message_cache_clean

类型： 布尔

默认值： no

如果为yes，当缓存同步时，mutt会清理掉过时的条目。You probably only want to set it every once in a while, since it can be a little slow (especially for large folders)。

3.135 message_format

类型： 字符串

默认值：“%s”

这是“附件”菜单中对message/rfc822类型的附件所显示的字符串。要获得完整类printf(3)序列定义，见小节\$index_format。

3.136 narrow_tree

类型：布尔

默认值：no

当设为yes，会使thread树变窄，以便适合更深的thread在屏幕上显示。

3.137 net_inc

型号：数值

默认值：10

需要传输大量网络数据的操作会在每\$net_inc千字节时更新它们的进展。如果设为0，则不会显示任何进展信息。

另见\$read_inc、\$write_inc和\$net_inc。

3.138 pager

类型：路径

默认值：“builtin”

此变量指定查看邮件时所使用的pager。值“builtin”是指使用mutt内置的pager，否则此变量应被指向你希望使用的外部pager的路径。

使用外部pager可能会有一些不便：你需要额外的按键以完成操作，因为你不能直接从这个pager中调用mutt的功能，and screen resizes cause lines longer than the screen width to be badly formatted in the help menu。

3.139 pager_context

型号：数值

默认值：0

此变量在内置pager中查看下一页或上一页时显示上下文的行数。默认情况下，Mutt会将屏幕的最后一行的下一行放在顶部来显示下一页（0行上下文）。

3.140 pager_format

类型：字符串

默认：“-%Z- %C/%m: % -20.20n %s%* -- (%P)”

此变量控制内置和外部pager中邮件上面的一栏“状态”信息。有效的序列在小节\$index_format中列出。

3.141 pager_index_lines

型号：数值

预设值： 0

确定使用pager查看邮件时mini-index的行数。当前的邮件会处在mini-index的中间，除非是靠近文件夹顶部或底部的邮件，使用户可以看到此邮件前后的其它邮件。这是很有用的，例如，确定当前thread还有多少封邮件需要查看。有一行是保留的，用作index的状态栏，因此，设置为6，实际只会显示5行mini-index。值为0的话将没有index显示。如果当前文件夹的邮件数量少于\$pager_index_lines的值，那么mini-index将只使用它需要的行数。

3.142 pager_stop

类型：布尔

默认值： no

当设为yes时，当一封邮件结束时并调用了功能<next-page>时，内置pager不会移动到下一封邮件。

3.143. crypt_autosign

类型：布尔

默认值： no

此变量使Mutt自动对发出的邮件进行签名。当需要加密而不需要进行签名时，可以在pgp菜单中更改。如果\$smime_is_default为yes，那么会使用OpenSSL来创建S/MIME邮件，并且使用S/MIME菜单里更改设置。（Crypto only）

3.144. crypt_autoencrypt

类型：布尔

默认： no

这个变量会使Mutt使用PGP加密发出的邮件。This is probably only useful in connection to the “send-hook” command。当需要签名而无须加密时，可以在pgp菜单中更改。如果\$smime_is_default为yes，那么会使用OpenSSL来创建S/MIME邮件，并且在S/MIME菜单里更改设置。（Crypto only）

3.145. pgp_ignore_subkeys

类型：布尔

默认： yes

Setting this variable will cause Mutt to ignore OpenPGP subkeys. Instead, the principal key will inherit the subkeys' capabilities. *Unset* this if you want to play interesting key selection games. （PGP only）

3.146. crypt_replyencrypt

类型：布尔

默认：yes

为yes时，当回复一封加密邮件时，自动使用PGP或OpenSSL加密回复邮件。（Crypto only）

3.147. crypt_replysign

类型：布尔

默认值：yes

为yes时，当回复一封被签名的邮件时，自动使用PGP或OpenSSL签名以回复的邮件。

注意：当邮件被同时加密和签名时，这个选项不会工作！（Crypto only）

3.148. crypt_replysignencrypted

类型：布尔

默认：no

如果为yes，回复一封被加密的邮件，自动使用PGP或OpenSSL签名回复邮件。此变量与\$crypt_replyencrypt结合起来使用，对所有自动加密的邮件进行签名。它解决了变量\$crypt_replysign的问题，mutt无法确定一封被加密的邮件是否也是被签名了。（Crypto only）

3.149. crypt_timestamp

类型：布尔

默认：yes

If *set*, mutt will include a time stamp in the lines surrounding PGP or S/MIME output, so spoofing such lines is more difficult. If you are using colors to mark these lines, and rely on these, you may *unset* this setting. （Crypto only）

3.150. pgp_use_gpg_agent

类型：布尔

默认值：no

如果设为yes，mutt将使用gpg-agent(1)进程。（PGP only）

3.151. crypt_verify_sig

类型：quadoption

默认：yes

为“yes”，则总是尝试验证PGP或S/MIME签名。如果为“ask-*”，询问是否验证签名。如果为“no”，则从不验证签名。（Crypto only）

3.152. smime_is_default

类型：布尔

默认值：no

mutt默认使用PGP进行签名/加密操作。要更改默认设置，使用OpenSSL代替PGP，那么就必须设置此变量为yes。但是，这个变量并不影响答复邮件，因为mutt会自动选择与原始邮件相同的方式来签名/加密邮件。（注意，此变量为被覆盖，如果变量\$crypt_autosmime设为no的话）（S/MIME only）

3.153. smime_ask_cert_label

类型：布尔

默认：yes

此标志控制mutt在添加证书到数据库时，是否要求你输入一个标签。此变量默认为yes。（S/MIME only）

3.154. smime_decrypt_use_default_key

类型：布尔

默认：yes

如果为yes（默认），mutt会使用默认的密钥解密。否则，如果管理多个证书—密钥对，mutt会尝试利用信箱地址来决定使用哪一个密钥。如果不能找到，mutt会要求你选择一个。（S/MIME only）

3.155. pgp_entry_format

类型：字符串

默认值：“%4n %t%f %4l/0x%k %-4a %2c %u”

利用这个变量你可以根据个人的习惯来定制PGP密钥选择菜单。这个字符串和\$index_format类似，但它有自己的类printf(3)序列：

%n

序号

%k

key id

%u

用户id

%a

算法

%l

密钥长度

%f

标志位

%c

密钥功能

%t

有效性

%[<s>]

date of the key where <s> is an `strftime(3)` expression

(PGP only)

3.156. `pgp_good_sign`

类型：正则表达式

默认值：（空）

If you assign a text to this variable, then a PGP signature is only considered verified if the output from `$pgp_verify_command` contains the text. Use this variable if the exit code from the command is 0 even for bad signatures. (PGP only)

3.157. `pgp_check_exit`

类型：布尔

默认：yes

为yes时，在签名或加密时mutt会检查PGP子进程的返回值。非0的返回值意味着子进程执行失败。(PGP only)

3.158. `pgp_long_ids`

类型：布尔

默认值：no

如果设为yes，使用64位的PGP密钥ID，如果为no，使用正常的32位密钥ID。(PGP only)

3.159. `pgp_retainable_sigs`

类型：布尔

默认值：no

如果设为`yes`，签名和加密的邮件将包括内嵌的`multipart/signed`和`multipart/encrypted`邮件体部分。

对于加密和签名的邮件列表，外层的（`multipart/encrypted`）可以很容易地移除，内嵌的`multipart/signed`部分被保留。（PGP only）

3.160. `pgp_autoinline`

类型：布尔

默认值：no

此选项控制mutt使用内联方式（传统的）的PGP来加密或签名邮件。当不想使用内联方式时，可以在pgp菜单里更改。

注意，对于包含一个以上MIME部分的邮件，Mutt会自动使用PGP/MIME。Mutt可以被配置成当内联方式（传统的）无效时，询问用户是否发送PGP/MIME邮件。

还可参考变量`$pgp_mime_auto`。

还请注意，我们非常不赞成你使用传统方式的PGP邮件。（PGP only）

3.161. `pgp_replyinline`

类型：布尔

默认值：yes

设置这个变量会使Mutt回复那些内联PGP encrypted/signed的邮件时，自动使用内联（传统）方式处理邮件。当不希望使用内联方式时，可以使用pgp菜单更改它。此选项不自动检测，如果（回答的）的信息是内嵌，而是依靠内部以前Mutt中检查/标记的邮件。

注意，对于包含一个以上MIME部分的邮件，Mutt会自动使用PGP/MIME。Mutt可以被配置成当内联方式（传统的）无效时，询问用户是否发送PGP/MIME邮件。

另见变量`$pgp_mime_auto`。

还请注意，我们强烈反对你使用传统方式的PGP邮件。（PGP only）

3.162. `pgp_show_unusable`

类型：布尔

默认：yes

设为`yes`时，mutt会在PGP密钥菜单上显示不可用的密钥。包括那些被撤销、已过期、或者被用户标记为记为“不可用”的密钥。（PGP only）

3.163. `pgp_sign_as`

类型：字符串

默认值：（空）

如果你有一个以上的密钥对，此选项允许你指定使用哪一个私钥。建议你使用keyid的形式来指定你的密钥（如0x00112233）。（PGP only）

3.164. pgp_strict_enc

类型：布尔

默认：yes

为yes时，Mutt自动将使用PGP/MIME签名的邮件编码为quoted-printable。请注意，设置此变量为no可能会导致为验证签名的问题。只有当你知道自己在做什么时才更改它。（PGP only）

3.165. pgp_timeout

型号：数值

默认值：300

缓存密码的过期时间（秒）（PGP only）

3.166. pgp_sort_keys

类型：排序

默认：address

指定PGP菜单条目的排序方式。有以下方式：

address

按user id字母排序

keyid

按key id字母排序

date

按密钥创建时期排序

trust

按密钥的信任程度排序

如果你按以上值反向排序，使用前缀与“reverse-”。（PGP only）

3.167. pgp_mime_auto

类型：quadoption

默认：ask-yes

此选项控制在使用inline（传统）模式发送（签名/加密的）邮件失败时是否提醒你改用PGP/MIME发送。

还请注意，我们强烈反对你使用传统方式的PGP邮件。（PGP only）

3.168. pgp_auto_decode

类型：布尔

默认：no

If *set*, mutt will automatically attempt to decrypt traditional PGP messages whenever the user performs an operation which ordinarily would result in the contents of the message being operated on. For example, if the user displays a pgp-traditional message which has not been manually checked with the [<check-traditional-pgp>](#) function, mutt will automatically check the message for traditional pgp.

3.169. pgp_decode_command

类型：字符串

默认：（空）

这是一个格式字符串，它指定了用于解码application/pgp附件的命令。

这些命令有它自己的类printf(3)序列：

%p

当需要密码时，扩展成PGPPASSFD=0，否则为一个空字符串。注：This may be used with a %? construct。

%f

展开成包含邮件的文件的文件名。

%s

展开成包含multipart/signed附件的签名部分的文件的文件名，当验证它的时候。

%a

变量\$pgp_sign_as的值。

%r

一个或多个key ID。

For examples on how to configure these formats for the various versions of PGP which are floating around，查看位于系统文档目录的samples/子文件夹里的pgp和pgp配置文件示例。（PGP only）

3.170. pgp_getkeys_command

类型：字符串

默认值：（空）

当 mutt 需要公钥信息的时候会调用这个命令。Of the sequences supported by \$pgp_decode_command, %r is the only printf(3)-like sequence used with this format.（PGP的唯一）

3.171. pgp_verify_command

类型：字符串

默认值：（空）

此命用来验证PGP签名。

这是一个格式字符串，可用的类printf(3)序列，见命令\$pgp_decode_command。（PGP的唯一）

3.172. pgp_decrypt_command

类型：字符串

默认：（空）

此命令用来解密一封PGP加密邮件。

这是一个格式字符串，可用的类printf(3)序列，见命令\$pgp_decode_command。（PGP only）

3.173. pgp_clearsign_command

类型：字符串

默认值：（空）

此命理个用来创建旧的“clearsigned”PGP邮件。请注意，我们强烈反对你使用这种格式。

这是一个格式字符串，可用的类printf(3)序列，见命令\$pgp_decode_command。（PGP only）

3.174. pgp_sign_command

类型：字符串

默认值：（空）

此命令为multipart/signed的PGP/MIME邮件体部分创建一个独立的PGP签名。

这是一个格式字符串，可用的类printf(3)序列，见命令\$pgp_decode_command。（PGP only）

3.175. pgp_encrypt_sign_command

类型：字符串

默认值：（空）

此命令用来同时签名和加密邮件体部分。

这是一个格式字符串，可用的类`printf(3)`序列，见命令`$pgp_decode_command`。（PGP only）

3.176. `pgp_encrypt_only_command`

类型：字符串

默认值：（空白）

此命令用来加密邮件体部分，不签名。

这是一个格式字符串，可用的类`printf(3)`序列，见命令`$pgp_decode_command`。（PGP only）

3.177. `pgp_import_command`

类型：字符串

默认值：（空）

此命令用来从邮件导入密钥到用户的公钥还。

这是一个格式字符串，可用的类`printf(3)`序列，见命令`$pgp_decode_command`。（PGP only）

3.178. `pgp_export_command`

类型：字符串

默认值：（空白）

此命令用来从用户的公钥环中导出一个公钥。

这是一个格式字符串，可用的类`printf(3)`序列，见命令`$pgp_decode_command`。（PGP only）

3.179. `pgp_verify_key_command`

类型：字符串

默认值：（空）

此命令在密钥选择菜单中验证密钥信息。

这是一个格式字符串，可用的类`printf(3)`序列，见命令`$pgp_decode_command`。（PGP only）

3.180. `pgp_list_secring_command`

类型：字符串

默认值：（空）

此命令用来列出私钥环的内容。输出格式必须和以下命令的输出类似：

```
gpg --list-keys --with-colons.
```

这种格式也可以由mutt附带的工具pgpring生成。

这是一个格式字符串，可用的类printf(3)序列，见命令\$pgp_decode_command。（PGP only）

3.181. pgp_list_pubring_command

类型：字符串

默认值：（空）

此命令用来列出公钥环的内容。输出格式必须和以下命令的输出类似：

```
gpg --list-keys --with-colons.
```

这种格式也可以由mutt附带的工具pgpring生成。

这是一个格式字符串，可用的类printf(3)序列，见命令\$pgp_decode_command。（PGP only）

3.182. forward_decrypt

类型：布尔

默认：yes

控制转发邮件时加密邮件的处理方式。为yes时，会脱掉外层。此变量仅在变量\$mime_forward为yes，而且变量\$mime_forward_decode为no时有效。（PGP only）

3.183. smime_timeout

类型：数值

默认值：300

缓存密码的过期时间（秒）（S/MIME only）

3.184. smime_encrypt_with

类型：字符串

默认值：（空）

设置加密的算法。可用的算法有“des”，“des3”，“rc2-40”，“rc2-64”，“rc2-128”。如果不“设置，那么使用3des”（TripleDES）。（S/MIME only）

3.185. smime_keys

类型：路径

默认值：（空）

S/MIME没有PGP中的公钥与私钥，mutt需要自己来处理密钥/证书的储存和检索。这是最基本的操作，mutt将密钥和证书储存在两个不同的目录，分别以其哈希值命名。还有一个索引文件，其

中包含信箱地址的keyid对，你可以手动修改它。这个变量指向密钥的位置。（S/MIME only）

3.186. smime_ca_location

类型：路径

默认值：（空）

这个变量包含被信任的证书的目录或文件的名称。（S/MIME only）

3.187. smime_certificates

类型：路径

默认值：（空）

S/MIME没有PGP中的公钥与私钥，mutt需要自己来处理密钥/证书的储存和检索。这是最基本的操作，mutt将密钥和证书储存在两个不同的目录，分别以其哈希值命名。还有一个索引文件，其中包含信箱地址的keyid对，你可以手动修改它。这个变量指向密钥的位置。（S/MIME only）

3.188. smime_decrypt_command

类型：字符串

默认：（空白）

这个格式字符串指定了用于解密application/x-pkcs7-mime附件的命令。

openssl命令它自己的一整套printf(3)序列的格式，与PGP的类似：

\$f
展开成包含该邮件的文件的文件名

%s
展开成包含multipart/signed附件签名部分的文件的文件名，当验证它的时候

%k
变量\$smime_default_key指定的密钥对

%c
一个或多个证书ID

%a
解密算法

%C
CA的位置：取决与变量\$smime_ca_location指向一个目录还是文件，目录扩展成“CApath

`$smime_ca_location`”，文件则扩展成“- CAfile `$ smime_ca_location`”。

关于如果使用这些命令，参见系统文档目录的`samples/`子目录里的`smime.rc`。（S/MIME only）

3.189. `smime_verify_command`

类型：字符串

默认值：（空）

此命令用来验证`multipart/signed`类型的S/MIME签名。

这是一个格式字符串，可用的类`printf(3)`序列，见命令`$smime_decrypt_command`。（S/MIME only）

3.190. `smime_verify_opaque_command`

类型：字符串

默认值：（空）

此命令用来验证`application/x-pkcs7-mime`类型的S/MIME签名。

这是一个格式字符串，可用的类`printf(3)`序列，见命令`$smime_decrypt_command`。（S/MIME only）

3.191. `smime_sign_command`

类型：字符串

默认值：（空）

此命令用来创建`multipart/signed`类型的S/MIME签名，其能够被所有的邮件客户端处理。

这是一个格式字符串，可用的类`printf(3)`序列，见命令`$smime_decrypt_command`。（S/MIME only）

3.192. `smime_sign_opaque_command`

类型：字符串

默认值：（空）

此命令用来创建`application/x-pkcs7-signature`类型的S/MIME签名，其只能被支持S/MIME扩展的邮件客户端处理。

这是一个格式字符串，可用的类`printf(3)`序列，见命令`$smime_decrypt_command`。（S/MIME only）

3.193. `smime_encrypt_command`

类型：字符串

默认值：（空）

此命令用来创建加密的S/MIME邮件。

这是一个格式字符串，可用的类printf(3)序列，见命令\$smime_decrypt_command。（S/MIME only）

3.194. smime_pk7out_command

类型：字符串

默认值：（空）

此命令用来提取S/MIME签名的PKCS7部分，以便能够得到X509证书。

这是一个格式字符串，可用的类printf(3)序列，见命令\$smime_decrypt_command。（S/MIME only）

3.195. smime_get_cert_command

类型：字符串

默认值：（空）

此命令用来从PKCS7部分中提取出X509证书。

这是一个格式字符串，可用的类printf(3)序列，见命令\$smime_decrypt_command。（S/MIME only）

3.196. smime_get_signer_cert_command

类型：字符串

默认值：（空）

此命令从S/MIME签名中只提取X509证书，因此可以比较该证书的拥有者和邮件的“From:”字段。

这是一个格式字符串，可用的类printf(3)序列，见命令\$smime_decrypt_command。（S/MIME only）

3.197. smime_import_cert_command

类型：字符串

默认值：（空）

此命令通过smime_keys导入证书。

这是一个格式字符串，可用的类printf(3)序列，见命令\$smime_decrypt_command。（的S / MIME只）

3.198. `smime_get_cert_email_command`

类型：字符串

默认值：（空）

此命令提取存储X509证书以及验证（o check whether the certificate was issued for the sender's mailbox）所需要的邮件地址。

这是一个格式字符串，可用的类`printf(3)`序列，见命令`$smime_decrypt_command`。（S/MIME only）

3.199. `smime_default_key`

类型：字符串

默认值：（空）

用来签名的默认密钥对。必须被设置为keyid（OpenSSL生成的哈希值），否则不能正常工作。（S/MIME only）

3.200. `ssl_client_cert`

类型：路径

默认值：（空）

The file containing a client certificate and its associated private key。

3.201. `ssl_force_tls`

类型：布尔

默认值：no

如果这个变量为yes，那么Mutt会加密所有到远程服务器的连接。此外，它会尝试使用TLS，即使服务器不支持。此选项会覆盖`$ssl_starttls`。

3.202. `ssl_starttls`

类型：quadoption

默认：yes

如果为yes（默认），mutt会使用STARTTLS，对于具备该特性的服务器。如果为no，mutt无论如何也不会使用STARTTLS。

3.203. `certificate_file`

类型：路径

默认值：“~/`.mutt_certificates`”

此变量指定保存你所信任证书的文件。当遇到一个未知的证书时，mutt会询问你是否接受它。如果你接受它，证书会被保存在此文件中，并且以后自动接受该证书。

你也可以手动添加CA证书到该文件中。服务器上被这些CA证书签署过的证书也会被自动接受。

示例

```
set certificate_file=~/.mutt/certificates
```

3.204. ssl_usesystemcerts

类型：布尔

默认：yes

为yes时，当mutt检查一个证书是否被你所信任的CA证书签署过时，mutt会使用系统的CA证书。

3.205. entropy_file

类型：路径

默认值：（空）

包括随机数据的文件，用来初始化的SSL库函数。

3.206. ssl_use_sslv2

类型：布尔

默认：yes

这个变量指定是否在SSL验证时尝试使用SSLv2。

3.207. ssl_use_sslv3

类型：布尔

默认：yes

这个变量指定是否在SSL验证时尝试使用SSLv3。

3.208. ssl_use_tlsv1

类型：布尔

默认：yes

这个变量指定是否在SSL验证时尝试使用TLSv1。

3.209. ssl_min_dh_prime_bits

类型：数值

默认值： 0

此变量指定在进行Diffie-Hellman密钥交换时可接受大小的最小值。设为0将使用GNUTLS库的默认值。

3.210. ssl_ca_certificates_file

类型： 路径

默认： （空）

此变量指定一个包含信任CA证书的文件。任何被这些CA证书所签署的服务器证书都会被自动接受。

示例：

```
set ssl_ca_certificates_file=/etc/ssl/certs/ca-certificates.crt
```

3.211 pipe_split

类型： 布尔

默认值： no

用于跟随在<tag-prefix>之后的功能<pipe-message>。如果这个变量为*no*，当pipe所有被标记的邮件时，mutt会连接这些邮件并将它们全部pipe。当设为*yes*时，Mutt将逐一pipe这些邮件。在两种情况下，邮件都会按当前的顺序被pipe，并且将变量\$pipe_sep定义的分隔符后添加到每封邮件后。

3.212 pipe_decode

类型： 布尔

默认值： no

用于命令<pipe-message>。为*no*的话，邮件被pipe前不会进行任何预处理。当设为*yes*时，Mutt会先整理邮件头、尝试编码邮件。

3.213 pipe_sep

类型： 字符串

默认： “/n”

当pipe一些邮件到外部的Unix命令，在这些邮件中插入的分隔符。

3.214 pop_authenticators

类型： 字符串

默认值： （空）

这是一个用冒号分隔的列表，包含mutt用于登录POP服务器的验证方法，mutt会按顺序尝试这些方法来登录。验证方法可以是“user”、“apop”或任何的SASL机制，如“digest-md5”、“gssapi”或“cram-md5”。此选项不区分大小写。如果不设置此选项（默认），mutt将尝试所有可用的方法，以安全性从高到低的顺序。

示例

```
set pop_authenticators="digest-md5:apop:user"
```

3.215 pop_auth_try_all

类型：布尔

默认：yes

为yes的话，Mutt会尝试所有可用的验证方法。当设为no，Mutt只又在前一个方法不可用时才尝试下一个验证方法。如果一个方法是可用的，但验证失败了，Mutt就连接不上POP服务器。

3.216 pop_checkinterval

型号：数值

预设值：60

此变量配置mutt检查新邮件的频率（以秒计），如果当前选中的信箱是POP信箱。

3.217 pop_delete

类型：quadoption

默认值：ask-no

如果设为yes，当使用功能[<fetch-mail>](#)时，mutt会删除从POP服务器上成功下载的邮件。当设为no，Mutt下载邮件后，仍然将它们留在POP服务器上。

3.218 pop_host

类型：字符串

默认值：（空）

功能[<fetch-mail>](#)中的POP服务器名称。你也可以指定其端口，用户名和密码，如：

```
[pop[s]://][username[:password]@]popserver[:port]
```

方括号“[...]”表示可选部分。

3.219 pop_last

类型：布尔

默认值：no

如果设为yes，调用功能[<fetch-mail>](#)时，mutt会尝试使用POP命令“LAST”以从POP服务器上只抓取未读邮件。

3.220 pop_reconnect

类型：quadoption

默认值：ask-yes

控制mutt是否尝试重新连接到服务器，如果连接丢失。

3.221 pop_user

类型：字符串

默认值：（空）

你登录POP服务器的用户名。

此变量默认只为你本地计算机上的用户名

3.222 pop_pass

类型：字符串

默认值：（空）

指定POP帐户的密码。如果不设置，当你打开一个POP信箱时Mutt会提示你输入密码。

警告：你应该只在安全的机器上使用此选项，因为超级用户可以查看你的muttrc文件，即使你是唯一一个有权查看该文件的人。

3.223 post_indent_string

类型：字符串

默认值：（空）

类似变量\$attribution，Mutt会将这个字符串添加到正在回复到邮件后。

3.224 postpone

类型：quadoption

默认值：ask-yes

控制是否将邮件保存到\$postponed信箱中，当不立即发送邮件时。

参见变量\$recall。

3.225 postponed

类型：路径

默认值：“~/postponed”

Mutt允许你无限期地“推迟发送一封邮件”。当你选择推迟一封邮件，Mutt将它保存在这个变量指定的信箱中。

参见变量\$postpone。

3.226 preconnect

类型：字符串

默认值：（空）

设置之后，如果mutt没有连接到服务器，一个shell命令会被执行。这对于建立安全连接非常有用，比如使用ssh(1)。如果命令返回非零值，mutt放弃打开服务器。例如：

```
set preconnect="ssh -f -q -L 1234:mailhost.net:143 mailhost.net \  
sleep 20 < /dev/null > /dev/null"
```

“mailhost.net”上的信箱“foo”现在可以访问，“{localhost:1234}foo”。

注意：要使这个示例工作，你必须能够不输入密码登录到远程计算机。

3.227 print

类型：quadoption

默认值：ask-no

控制是否真的打印邮件。默认情况下，设置为“ask-no”，因为有些人常常不小心按下“p”键。

3.228 print_command

类型：路径

默认值：“lpr”

它指定用于打印邮件的pipe命令。

3.229 print_decode

类型：布尔

默认：yes

用于命令<print-message>。如果这个变量设为yes，邮件在传递到\$print_command所指定的外部命令前会先被解码。如果设为no，打印邮件时不会对邮件执行任何操作。如果你使用的是一些能够正确格式华电子邮件以进行打印的高级的打印过滤器，那么后一个设置就适合你。

3.230 print_split

类型：布尔

默认值: no

用于命令<print-message>。如果设为yes，\$print_command指定的命令会对每一封要打印的邮件各执行一次。如果为no，该命令只会执行一次，所有的邮件以邮件分隔符连接起来。

那些使用程序enscript(1)的邮件打印模式的人很可能想要将这个变量设为yes。

3.231 prompt_after

类型: 布尔

默认: yes

如果你使用外部的\$pager，当pager退出时，这个变量会导致Mutt提示你键入一个命令，而不是直接返回index菜单。如果为no，Mutt将返回index菜单，当外部pager退出时。

3.232 query_command

类型: 路径

默认值: (空)

指定一个mutt用于查询外部地址的命令。字符串中应包含“%s”，它将被用户输入的查询关键字取代。见“query”获取更多信息。

3.233 query_format

类型: 字符串

默认值: “%4c %t %-25.25a %-25.25n %?e?(%e)?“

此变量指定了“查询”菜单的格式。可用的类printf(3)序列如下:

%a

地址

%c

当前条目号

%e

额外信息*

%n

名字

%t

“*”，如果当前条目被标记，否则为空白

%>X

右对齐余下的字符串，使用字符“X”填充

%lX

使用字符“X”填充至行末

/*X

使用字符“X”软填充

关于“软填充”，见\$index_format。

* = can be optionally printed if nonzero, see the \$status_format documentation。

3.234 quit

类型：quadoption

默认：yes

这个变量控制是否真的“退出”mutt。为yes的话，当你要求退出时，它会退出mutt。如果为no，会毫无反应。如果设置为ask-yes或ask-no，系统会提示你确认操作，当你尝试退出时。

3.235 quote_regexp

类型：正则表达式

默认值：“^([\ \t]*[|>:}#])+”

在内部pager中用于检测邮件体中引文字段的正则表达式。可以使用命令<toggle-quoted>切换是否显示引文，或使用“color quoted”来标记颜色。

不同深度的引文可以用不同的颜色标记（“color quoted1”，“color quoted2”，等等）。它查测引文深度的过程为，从匹配文本中移除最后一个字符，然后再递归执行这个过程直到不能匹配任何文本。

匹配检测可能会被变量\$smileys所指定的正则表达式覆盖。

3.236 read_inc

型号：数值

默认值：10

如果设置大于0的值，Mutt会显示进度，当Mutt读取信箱或执行搜索动作时。当此变量设置的数值的邮件被读取或搜索时，mutt会更新一次进度。例如，如果设为25，当mutt会在第25封时更新进度，然后在第50封时再更新一次。这个变量是为了在操作那些费时的大信箱时显示进度的。当设置为0，那么过程中mutt不会更新进度。

可参见变量\$write_inc，\$net_inc和\$time_inc，以及“性能优化”小节。

3.237 read_only

类型：布尔

默认值：no

如果设为yes，所有文件夹都以只读模式打开。

3.238 realname

类型：字符串

默认值：（空）

此变量指定“真实的”或“个人”名字，发送邮件时使用。

默认情况下，这是从/etc/passwd中提取的GECOS field。请注意，如果用户设定了变量\$from，这个变量将不会被使用。

3.239 recall

类型：quadoption

默认值：ask-yes

控制Mutt是否在撰写新邮件时调出被推迟的邮件。

设置此变量一般不实用，因此不推荐这么做。

还见变量\$postponed。

3.240 recall

类型：路径

默认值：“~/sent”

指定保存已发邮件的文件位置。（这是保存你已发邮件的主要方法，另一种方法是使用命令“my_hdr”建立一个“Bcc:”字段，其中包含你的电子邮件地址）

\$record的值会被变量\$force_name和\$save_name，以及命令“fcc-hook”覆盖。

3.241 reply_regexp

类型：正则表达式

默认值：“^(re([\\0-9\\+])*|aw):[\\t]*”

A regular expression used to recognize reply messages when threading and replying。默认值对应英文的“Re:”与德文的“Aw:”。

3.242 reply_self

类型：布尔
默认值：no

为`no`的话，当你回复一封由你发出的邮件，Mutt会认为你要回复到邮件的收件人，而不是到你
自己。

也见命令“`alternates`”。

3.243 `reply_to`

类型：quadoption
默认值：ask-yes

如果为`yes`，回复邮件时，Mutt会使用“`Reply-to:`”字段中的地址作为要回复到的地址。如果
为`no`，会使用“`From:`”字段中的地址。当你阅读一封设置了“`Reply-To:`”字段的邮件列表，你想私
下回复给邮件作者，这个变量就特别有用。

3.244 `resolve`

类型：布尔
默认：yes

为`yes`时，当你执行了一个修改当前邮件的命令后，光标会自动跳到下一封（可能是没有被删除
的）邮件上。

3.245 `reverse_alias`

类型：布尔
默认值：no

如果发现有别名与发件人地址匹配，mutt是否在index菜单中会显示别名的“个人”名称。例如，
如果有以下别名：

```
alias juser abd30425@somewhere.net (Joe User)
```

然后你收到包含如下字段的邮件：

```
From: abd30425@somewhere.net
```

这将在index菜单上显示“Joe User”，而不是“abd30425@somewhere.net”。对于不太友好的e-mail
地址，这非常又用。

3.246 `reverse_name`

类型：布尔
默认值：no

It may sometimes arrive that you receive mail to a certain machine, move the messages to another machine, and reply to some the messages from there. If this variable is *set*, the default *From:* line of the reply messages is built using the address where you received the messages you are replying to **if** that address matches your “alternates”. If the variable is *unset*, or the address that would be used doesn't match your “alternates”, the *From:* line will use your address on the current machine.

也请参见“后备地址”部分。

3.247 reverse_realname

类型：布尔

默认：yes

这个变量用于对变量\$reverse_name的行为的微调。当它被设为yes，mutt将使用原始邮件的地址，其中可能包含真实姓名。当它被设为no，mutt会使用变量\$realname覆盖任何真实姓名。

3.248 rfc2047_parameters

类型：布尔

默认值：no

当此变量设为no，Mutt will decode RFC2047-encoded MIME parameters。You want to set this variable when mutt suggests you to save attachments to files named like

```
=?iso-8859-1?Q?file=5F=E4=5F991116=2Ezip?=
```

设置此变量不会立即生效，直到你更改文件夹。

注意，RFC2047的编码方式不是标准的，尽管会在一些地方遇到。

也请注意，设定这个参数不会影响mutt *generates*这种编码。相反，mutt会无条件地使用RFC2231指定的编码方式。

3.249 save_address

类型：布尔

默认值：no

如果设为yes，mutt will take the sender's full address when choosing a default folder for saving a mail。如果\$save_name或\$force_name也设为yes，FCC文件夹的选择也会被改变。

3.250 save_empty

类型：布尔

默认：yes

当设为no，退出时没有保存邮件的信箱将被移除（有一个例外是\$spoolfile不会被移除）。为yes的话，信箱不会被移除。

注意：这仅适用于mbox和MMDF文件夹，Mutt不会移除MH和Maildir目录。

3.251 save_history

类型：数值

默认值：0

此变量控制保存（每个类别）操作记录的\$history_file文件的大小。

3.252 save_name

类型：布尔

默认值：no

此变量控制已发邮件副本的保存方式。当设为yes，mutt会检查收件人地址所对应的信箱是否存在（这是通过在\$folder目录搜索收件人地址的*用户名*部分实现的）。如果信箱存在，已发邮件将被保存到改信箱，否则保存到\$record信箱。

还可以参见变量\$force_name。

3.253 score

类型：布尔

默认：yes

当这个变量设置为no，评分功能将被禁用。使用\$score_threshold_delete以及相关的变量，可以选择性的禁用某些文件夹的评分功能。

3.254 score_threshold_delete

类型：数值

默认：-1

当邮件的评分等于或低于此变量的值，mutt会自动将邮件标记为删除。由于mutt的评分总是大于或等于零，所以默认的值将永远不会将邮件标记为删除。

3.255 score_threshold_flag

类型：数值

默认值：9999

当邮件的评分等于或大于此变量的值，mutt会自动将“标记”邮件。

3.256 score_threshold_read

类型：数值

默认：-1

当邮件的评分等于或小于此变量的值，mutt会自动将邮件标记为已读。由于mutt的评分总是大于或等于零，所以默认的值将永远不会将邮件标记为已读。

3.257 send_charset

类型：字符串

默认值：“us-ascii:iso-8859-1:utf-8”

这是一个用冒号分隔字符集列表，用于发送邮件。Mutt中使用的第一个字符集将其文字可以转换准确。如果你的\$charset不是“iso-8859-1”，并且收件人可能识别“UTF-8”，it is advisable to include in the list an appropriate widely used standard character set（如“iso-8859-2”，“koi8-r”或“iso-2022-jp”）either instead of or after“iso-8859-1”。

如果不能使用这些字符集的其中之一将文字准确地转换，mutt将使用\$charset作为后备。

3.258 sendmail

类型：路径

默认：“/usr/sbin/sendmail -oem -oi”

指定用于投递邮件的程序和参数。Mutt expects that the specified program interprets additional arguments as recipient addresses。

3.259 sendmail_wait

类型：数值

默认值：0

指定在将\$sendmail放入后台之前等待的秒数。

Mutt按以下几种方式解析变量：

>0	继续前等待\$sendmail完成的秒数
0	永远等待，直到\$sendmail完成
<0	总是把\$sendmail放入后台，无需等待

请注意，如果你了一个非0的值，子进程的输出将会放到一个临时文件里。如果有错误产生，你将被告知在何处找到这些输出。

3.260. shell

类型：路径

默认值：（空白）

生成一个子shell的命令。默认情况下，使用用户的登录shell（`/etc/passwd`中的值）。

3.261. sig_dashes

类型：布尔

默认：yes

为yes时，包含“-- ”（注意末尾的空格）的行将被插入到你的\$signature之前。我们强烈建议你不要设置为no，除非你的签名仅包含你的姓名。原因是许多程序会使用“-- \n”来检测你的签名。例如，Mutt可以在内置pager中用不同的颜色高亮显示签名。

3.262. sig_on_top

类型：布尔

默认值：no

如果设为yes，签名会被包括到任何引文或转发内容中。我们强烈建议你不要设置此变量，除非你真的知道自己在做什么，并且做好被别人责备的心理准备。

3.263 signature

类型：路径

默认值：“~/.signature”

指定存放你签名的文件，签名会附加到所有发出的邮件上。如果文件名以一个管道符（“|”）结束，就认为文件是一个shell命令，从此命令的标准输出来读取签名。

3.264 simple_search

类型：字符串

默认值：“~f %s | ~s %s”

指定Mutt如何将一个简单搜索扩展成pattern搜索。简单搜索不包含任何pattern操作符“~”。见“patterns”获取更多pattern搜索的信息。

例如，你只需在搜索或提取提示符中键入“joe”，Mutt会自动扩展成变量所指定的值——将“%s”替换成joe。默认值下，“joe”会被扩展成：“~f joe | ~s joe”。

3.265 smart_wrap

类型：布尔

默认：yes

控制内部pager中超过屏幕长度的行的显示。为yes时，长的行会在单词边界处被折行。如果

为`no`，行只是简单的在屏幕边界出折行。也见变量`$markers`。

3.266 smileys

类型：正则表达式

默认：“(>From)|(:[-^]?[!()(><}{|/DP]){/0}“

The *pager* uses this variable to catch some common false positives of `$quote_regexp`, most notably smileys and not consider a line quoted text if it also matches `$smileys`。 This mostly happens at the beginning of a line.

3.267 sleep_time

类型：数值

默认值：1

指定当切换到其它文件夹和从当前文件夹删除邮件时，某些提示信息显示的停顿时间，以秒计。默认是暂停一秒，为0则取消停顿。

3.268 smtp_authenticators

类型：字符串

默认值：（空白）

这是一个用冒号分隔的验证方法列表，`mutt`使用这些方法来登录SMTP服务器。验证方法可以任何一种SASL机制，如“`digest-md5{/0`”，“`{0}gssapi`”或“`cram-md5`”。此选项不区分大小写。如果“未设置”（默认），`mutt`将尝试所有可用的方法，以安全性从高到低的顺序。

示例

```
set smtp_authenticators="digest-md5:cram-md5"
```

3.269 smtp_pass

类型：字符串

默认值：（空）

指定你的SMTP帐户的密码。如果未设置，当你第一次通过SMTP发送邮件，`Mutt`中会提示你输入密码。关于通过SMTP发送邮件的配置，参见`$smtp_url`。

警告：你只应该在一个安全的机器上使用这个选项，因为超级用户可以参看你的`muttrc`，即使你是唯一可以读取该文件的用户。

3.270 smtp_url

类型：字符串

默认值：（空）

定义投递邮件时应该交付的SMTP地址。应该使用SMTP URL的形式，例如：

```
smtp[s]://[user[:pass]@]host[:port]/
```

...其中方括号“[...]”表示可选部份。设置这个变量会覆盖变量\$sendmail的值。

3.271 sort

类型：排序

默认：date

指定在“index”菜单中如何对邮件排序。有效值如下：

- date或date-sent
- date-received
- from
- mailbox-order (未排序)
- score
- size
- spam
- subject
- threads
- to

你也可以使用前缀“reverse-”来指定排序方式（例如：“set sort=reverse-date-sent”）。

3.272 sort_alias

类型：排序

默认值：alias

指定“别名”菜单中的条目的排序方式。以下有效的值：

- address（按邮件地址的字母顺序排列）
- alias（按别名的字母顺序排列）
- unsorted（保持在中muttrc定义的顺序）

3.273 sort_aux

类型： 排序

默认： date

当以threads方式排序时，这个变量控制不同的thread之间、thread内的分支如何排序。可以设定为任何\$sort的值，除了“threads”（在这种情况下，mutt会使用“date-sent”）。除了前缀“reverse-”，你还可以使用前缀“last-”，“last-”必须跟在“reverse-”之后。The “last-” prefix causes messages to be sorted against its siblings by which has the last descendant, using the rest of \$sort_aux as an ordering。例如，

```
set sort_aux=last-date-received
```

这意味着，如果一个thread中有一封新邮件，这个thread将是最后一个显示（也可能是第一个，如果设置了“set sort=reverse-threads”）。

注意：For reversed \$sort order \$sort_aux is reversed again (which is not the right thing to do, but kept to not break any existing configuration setting)。

3.274 sort_browser

类型： 排序

默认： alpha

指定如何排序文件浏览器中的条目。默认情况下，项目是按字母顺序排列。有效值：

- alpha（按字母顺序排列）
- date
- size
- unsorted

你可以使用前缀“reverse-”来指定排列方式（例如：“设置set sort_browser=reverse-date”）。

3.275 sort_re

类型： 布尔

默认： yes

当以threads方式排序，而且\$strict_threads设为*no*时，这个变量才有效。In that case, it changes the heuristic mutt uses to thread messages by subject. With \$sort_re *set*, mutt will only attach a message as the child of another message by subject if the subject of the child message starts with a substring matching the setting of \$reply_regexp. With \$sort_re *unset*, mutt will attach the message whether or not this is the case, as long as the non-\$reply_regexp parts of both messages are identical.

3.276 spam_separator

类型： 字符串

默认值： “,”

此变量控制当匹配多个spam字段时的动作：如果没有设置，那么每个字段将覆盖上一个匹配的值。如果设置了此变量，那么每个字段会依次附加到上个匹配项后面，使用此变量作为分隔符。

3.277 spoolfile

类型：路径
默认值：（空）

如果你的spool信箱不在默认的位置，mutt不能找到，你可以使用这个变量来指定它的位置。如果环境变量\$MAIL未被设置，Mutt会自动将\$MAIL设为这个变量的值。

3.278 status_chars

类型：字符串
默认值：“-*%A”

控制\$status_format中指示器“%r”所使用的字符。第一个字符在信箱改变时使用。第二个在信箱改变，并需要同步时使用。第三个在信箱是只读模式时，或者在退出时信箱不能被写入（你可以使用功能<toggle-write>，默认为“%”，来切换是否将改变写入信箱）。第四个表明当前文件夹以attach-message模式（比如编辑新邮件，回复、转发邮件等等操作）打开。

3.279 status_format

类型：字符串
默认值：“-%r-Mutt: %f [Msgs:%?M?%M/?%m%n? New:%n?%?o? Old:%o?%?d? Del:%d?%?F? Flag:%F?%?t? Tag:%t?%?p? Post:%p?%?b? Inc:%b?%?l? %l?]---(%s/%S)-%>-(%P)---“

控件“index”菜单上的状态栏的显示格式。这个字符串类似\$index_format，但它有一组自己的类printf(3)序列：

- %b
包含新邮件的信箱数量*
- %d
已删除的邮件数量*
- %f
当前信箱的完整路劲
- %F
已标记邮件的数量*
- %h

	本地主机名
%l	
	当前信箱的大小（以字节为单位）*
%L	
	显示邮件（即，当前提取的邮件）的大小（以字节为单位）*
%m	
	信箱中邮件的数量*
%M	
	显示邮件（即，当前提取的邮件）的数量*
%n	
	信箱中新邮件的数量*
%o	
	旧的未读邮件的数量*
%p	
	推迟发送的邮件数量*
%P	
	当前显示在整个index中的百分比
%r	
	已修改/只读/不可写/attach-message指示符，参考变量\$status_chars
%s	
	当前排序方式（\$sort）
%S	
	当前aux排序方式（\$sort_aux）
%t	
	被标记的邮件数量*
%u	
	未读邮件的数量*

%V

Mutt版本号

%V

currently active limit pattern，如果有的话*

%>X

右对齐余下的字符串，使用字符“X”填充

%lX

使用字符“X”填充至行末

/*X

使用字符“X”软填充

关于“软填充”，见\$index_format部份。

* = 非0情况下可选显示的内容

在取值非0的情况下，上述序列中的一些可以选择显示。例如，你可能希望查看被标记邮件的数量，但如果数量为0的话，那它显示的意义就不大。可以使用下面的结构，选择性的显示信息：

%%?<sequence_char>?<optional_string>?

*sequence_char*是上述序列之一，*optional_string*是在*sequence_char*非0情况下你希望显示的字符串。*optional_string*中可以包含其他的序列，以及正常文本，一般来说你不希望只包含可选字符串。

下面是一个例子，说明如何选择性的显示信箱中的新邮件：

%%?n?%n new messages.?

你也可以使用下面的结构在两个字符串之间切换：

%%?<sequence_char>?<if_string>&<else_string>?

如果*sequence_char*的值为非0，*if_string*就被扩展，否则*else_string*被扩展。

使用前缀（“_”），你可以将任何类printf(3)序列的值转换为小写。例如，如果你要小写显示本地主机名，你可以使用：“%_h”。

如果你使用前缀（“:”），mutt使用下划线代替扩展中的点。这在IMAP文件夹中可能非常有用。

3.280 status_on_top

类型：布尔

默认值：no

设置这个变量会时“状态栏”显示在第一行，而不是在屏幕的底部。如果\$help为yes，则会将它放到底部。

3.281 strict_threads

类型：布尔

默认值：no

如果设为yes，mutt只会利用字段“In-Reply-To”和“References:”来进行thread排序。默认情况下，邮件按照相同的主题被放在一个“伪thread”中。也许这并不是总是可取的，如在个人邮箱中，有一些不相关的邮件，他们的主题都是“hi”，这些邮件会被放在一起。参见变量\$sort_re——以不太极端的方式控制这种行为。

3.282 suspend

类型：布尔

默认：yes

当设为no，当哟用户按下终端的susp键，通常为“^Z”，mutt不会停止。如果你使用像命令“xterm -e mutt”一样在终端内部执行mutt，这是非常有用的。

3.283 text_flow

类型：布尔

默认值：no

当设为yes时，mutt会生成“format=flowed”，content type为“text/plain; format=flowed”的邮件体。一些邮件客户端可以很好处理这种格式。要想使用这种格式的特性，你还需要你的编辑器支持。

请注意，此选项设为yes时，变量\$indent_string会被忽略。

3.284 thread_received

类型：布尔

默认值：no

当设为yes，mutt使用接受日期来按thread排列邮件，而不是发送日期。

3.285 thorough_search

类型：布尔

默认值：no

影响~b和~h的搜索行为（“模式”小节所述）。如果设为yes，邮件头以及邮件体/附件会在搜索前被解码。如果设为no，messages are searched as they appear in the folder。

用户搜索附件或非ASCII字符时，应设此变量为`yes`，因为解码包括了MIME的解析/解码，可能还有字符集转换。否则mutt会尝试匹配收到的原始邮件（例如使用Quoted-Printable编码的邮件或编码邮件头的邮件），这可能会产生不正确的搜索结果。

3.286 tilde

类型：布尔

默认值：no

设为`yes`，内部pager会将屏幕底部的空白行以波形符（“~”）。

3.287 time_inc

类型：数值

默认值：0

连同变量`$read_inc`，`$write_inc`，和`$net_inc`一起控制进度更新的频率。此变量设置了最大的更新时间，单位为毫秒。在缓慢的机器上，或者通过远程系统运行mutt时，它可以提高系统的吞吐量。

另外请参见“性能优化”部分。

3.288 timeout

类型：数值

默认值：600

当mutt等待用户在菜单或交互提示符中输入时，mutt会在闲置一段时间后挂起，直到用户继续输入。根据不同的情况，mutt会阻止一些操作，比如检查新邮件或保持IMAP连接。

此变量mutt等待超时的秒数。

小于或等于0的值会让mutt永不超时。

3.289. tmpdir

类型：路径

默认值：（空）

这个变量指定mutt显示和编辑邮件时存放临时文件的地方。如果没有设置，那么使用环境变量`$TMPDIR`。如果`$TMPDIR`没有设置，就使用目录“/tmp”。

3.290. to_chars

类型：字符串

默认值：“+TCFL”

控制标示收件人的字符。第一个字符表示该邮件不是给你的。第二个表示你是该邮件唯一的收

件人。第三个表示你的地址出现在“To:”字段中，但你不是邮件唯一的收件人。第四个表示你的地址出现在“Cc:”字段中，但你不是唯一的收件人。第五个表示邮件是由你发出的。第六个字符表示邮件来自你订阅的邮件列表。

3.291. tunnel

类型：字符串

默认：（空白）

设置这个变量会使mutt用管道代替原始套接字。你可以使用这个变量来设置IMAP/POP3/SMTP服务器的身份验证。示例：

```
set tunnel="ssh -q mailhost.net /usr/local/libexec/imapd"
```

注意：要使此示例的设置生效，你必须能够在无须输入密码的情况下登录远程机器。

3.292. use_8bitmime

类型：布尔

默认值：no

警告：不要设置此变量，除非你使用的是支持选项-B8BITMIME的sendmail版本（如sendmail8.8.x），否则你可能无法发送邮件。

设为yes后，当使用ESMTP协议发送8-bit邮件时，Mutt将使用选项-B8BITMIME来调用\$sendmail。

3.293 use_domain

类型：布尔

默认：yes

当设置为yes，mutt会让所有本地地址（那些没有“@host”部分的地址）使用\$hostname定义的域名。如果为no，那么mutt将不做处理。

3.294. use_envelope_from

类型：布尔

默认值：yes

When *set*, mutt will set the *envelope* sender of the message. If \$envelope_from_address is *set*, it will be used as the sender address. If *unset*, mutt will attempt to derive the sender from the From: header.

Note that this information is passed to sendmail command using the -f command line switch. Therefore setting this option is not useful if the \$sendmail variable already contains -f or if the executable pointed to by \$sendmail doesn't support the -f switch.

3.295. use_from

类型：布尔

默认：yes

当设为yes时，Mutt在发送邮件时会生成“From:”字段。如果设为no，就不会生成“From:”字段，除非用户使用命名“my_hdr”指定一个。

3.296. use_idn

类型：布尔

默认：yes

当设为yes，Mutt中会显示解码后的国际域名。注意：你可以使用国际域名作为地址，即使是该变量设为no。此变量只影响解码。

3.297. use_ipv6

类型：布尔

默认：yes

当为yes，mutt会查找它所连接主机的IPv6地址。如果这个选项为no，Mutt将仅局限于IPv4地址。通常情况下，默认设置能很好地工作。

3.298 user_agent

类型：布尔

默认：yes

当设为yes时，mutt会在发送邮件时增加一个“User-Agent:”字段，标明编辑此邮件的mutt版本。

3.299. visual

类型：路径

默认值：（空）

指定可视化编辑器，其被内置编辑器的命令“~v”使用。

3.300. wait_key

类型：布尔

默认：yes

此变量控制一个外部程序被函数<shell-escape>，<pipe-message>，<pipe-entry>，<print-message>，和<print-entry>调用后，mutt是否要求你按下一个按键。

它还可用于，自动查看一个附件类型时，mailcap中此类型所对应的条目若含有参数needsterminal时，一个互交的外部程序会被调用。

当设为yes，Mutt总是要求用户按键。如果设为no，那么只有在外程序返回非0值时mutt才会等

待用户按键。

3.301. weep

类型：布尔

默认：yes

当设置为yes时，mutt在显示、转发、打印或者回复邮件时过滤邮件头。

3.302. wrap

型号：数值

默认值：0

当设置为一个正值，mutt将在\$wrap个字符处折行。当设置为负值，mutt will wrap text so that there are \$wrap characters of empty space on the right side of the terminal。

3.303. wrap_search

类型：布尔

默认：yes

Controls whether searches wrap around the end.

When *set*, searches will wrap around the first (or last) item. When *unset*, incremental searches will not wrap.

3.304. wrapmargin

类型：数值

默认：0

（不推荐）相当于设置\$wrap为一个负值。

3.305. write_inc

类型：数量

默认值：10

对一个信箱进行写操作时，mutt将在每\$write_inc封邮件时更新进度。如果设置为0，那么mutt将不会更新进度。

参见变量\$read_inc，\$net_inc和\$time_inc，以及“性能优化”部分。

3.306. write_bcc

类型：布尔

默认：yes

控制mutt在发送邮件时是否写出“Bcc:”字段。Exim用户也许希望将此选项设为no。如果mutt通过SMTP（见\$smtp_url）发送邮件，那么这个选项将无效：在这种情况下mutt将不会写出“Bcc:”字段。

4. 功能

以下列出了可用功能与相应的键绑定。给出的是默认的键绑定与功能的描述。要更改键绑定使用命令bind。

4.1. 通用菜单

通用菜单不是一个真正的菜单，它指定了在除*pager*和*编辑器*外的所有菜单中通用的功能。改变这个菜单的设置会影响所有菜单的键绑定（除了上文所指出的菜单）。

表8.2. 通用功能默认键绑定

功能	默认键绑定	描述
<top-page>	H	移动到该页顶部
<next-entry>	j	移动到下一个条目
<previous-entry>	k	移动到上一个条目
<bottom-page>	L	移动到该页底部
<refresh>	^L	清除并重绘屏幕
<middle-page>	M	移动到该页的中间
<search-next>	n	查找下一个匹配项
<exit>	q	退出此菜单
<tag-entry>	t	标记当前条目
<next-page>	z	移动到下一页
<previous-page>	Z	移动到上一页
<last-entry>	*	移动到最后一个条目
<first-entry>	=	移动到第一个条目
<enter-command>	:	输入一个muttrc配置命令
<next-line>	>	向下滚动一行
<previous-line>	<	向上滚动一行
<half-up>	[向上滚动1/2页
<half-down>]	向下滚动1/2页
<help>	?	显示当前菜单的帮助
<tag-prefix>	;	将到下一个功能应用到所有已标记的邮件
<tag-prefix-cond>		将下一个功能只应用到已标记的邮件
<end-cond>		end of conditional execution (noop)

<shell-escape>	!	在子shell执行一个程序
<select-entry>	<Return>	选择当前条目
<search>	/	使用正则表达式搜索
<search-reverse>	ESC /	使用正则表达式向后搜索
<search-opposite>		以相反的方向寻找下一个匹配项
<jump>		跳转到指定序号的条目上
<current-top>		移动条目到屏幕顶部
<current-middle>		移动条目到屏幕中间
<current-bottom>		移动条目到屏幕底部
<what-key>		显示按键的keycode

4.2. index菜单

表8.3. index菜单默认键绑定

功能	默认键绑定	描述
<create-alias>	a	为发件人创建一个别名
<bounce-message>	b	remail给其他人
<break-thread>	#	拆分thread
<change-folder>	c	转到其它信箱
<change-folder-readonly>	Esc c	以只读模式转到其它信箱
<next-unread-mailbox>		打开下一个包含新邮件的信箱
<collapse-thread>	ESC v	折叠/展开当前thread
<collapse-all>	Esc V	折叠/展开所有thread
<copy-message>	C	复制邮件到文件/信箱
<decode-copy>	Esc C	解码邮件(text/plain)并复制邮件
<decode-save>	Esc s	解码邮件(text/plain)并移动邮件
<delete-message>	d	删除当前条目
<delete-pattern>	D	删除模式匹配的邮件
<delete-thread>	^D	删除thread中所有邮件
<delete-subthread>	Esc d	删除子thread中所有邮件
<edit>	e	编辑原始邮件
<edit-type>	^E	修改附件的content type
<forward-message>	f	转发邮件
<flag-message>	F	标记为重要
<group-reply>	g	回复所有收件人

<fetch-mail>	G	从POP服务器抓取邮件
<imap-fetch-mail>		从IMAP服务器强行抓取邮件
<display-toggle-weed>	h	显示邮件并开关邮件头的显示
<next-undeleted>	j	移动到下一封未删除的邮件
<previous-undeleted>	k	移动到上一封未删除的邮件
<limit>	l	显示模式匹配的邮件
<link-threads>	&	连接标记的邮件到当前邮件
<list-reply>	L	回复指定的邮件列表
<mail>	m	撰写新邮件
<toggle-new>	N	标志为新邮件
<toggle-write>	%	切换信箱是否可写
<next-thread>	^N	跳转到下一个thread
<next-subthread>	Esc n	跳转到下一个子thread
<query>	Q	调用外部程序查询地址
<quit>	q	保存更改并退出
<reply>	r	回复邮件
<show-limit>	Esc l	显示当前搜索的模式
<sort-mailbox>	o	改变当前排序方法
<sort-reverse>	O	反序改变当前排序方法
<print-message>	p	打印当前条目
<previous-thread>	P	跳转到上一个thread
<previous-subthread>	Esc p	跳转到上一个子thread
<recall-message>	R	打开发送暂缓的邮件
<read-thread>	^R	标记当前thread为已读
<read-subthread>	ESC r	标记当前子thread为已读
<resend-message>	Esc e	以当前邮件为模板发送新的邮件
<save-message>	s	保存邮件/附件
<tag-pattern>	T	标记模式匹配的邮件
<tag-subthread>		标记当前子thread
<tag-thread>	Esc t	标记当前线程
<untag-pattern>	^T	取消标记模式匹配的邮件
<undelete-message>	u	取消删除当前条目
<undelete-pattern>	U	取消删除模式匹配的邮件
<undelete-subthread>	Esc u	取消删除子thread中的所有邮件
<undelete-thread>	^U	取消删除thread中的所有邮件

<view-attachments>	v	查看MIME附件
<show-version>	V	显示Mutt的版本号和发布日期
<set-flag>	w	设置邮件的状态
<clear-flag>	W	清除邮件的状态
<display-message>	<Return>	显示邮件
<buffy-list>	.	列出包含新邮件的信箱
<sync-mailbox>	\$	保存信息的更改
<display-address>	@	显示发件人完整的地址
<pipe-message>		pipe邮件/附件到一个shell命令
<next-new>		跳转到下一封新邮件
<next-new-then-unread>	<Tab>	跳转到下一封新邮件或未读邮件
<previous-new>		跳转到上一封新邮件
<previous-new-then-unread>	Esc <Tab>	跳转到上一封新邮件或未读邮件
<next-unread>		跳转到下一封未读邮件
<previous-unread>		跳转到上一封未读邮件
<parent-message>	P	跳转thread的父邮件
<extract-keys>	^K	提取公钥
<forget-passphrase>	^F	清除内存中缓存的密码
<check-traditional-pgp>	Esc P	检查传统的PGP方式
<mail-key>	Esc k	发送一个PGP公钥
<decrypt-copy>		解密后复制
<decrypt-save>		解密后移动

4.3. pager菜单

表8.4. pager菜单默认键绑定

函数	默认的关键	内容简介
<break-thread>	#	拆分thread
<create-alias>	a	为发件人创建一个别名
<bounce-message>	b	remail给其他人
<change-folder>	c	转到其它信箱
<change-folder-readonly>	Esc c	以只读模式转到其它信箱
<next-unread-mailbox>		打开下一个包含新邮件的信箱
<copy-message>	C	复制邮件到文件/信箱
<decode-copy>	Esc C	解码邮件(text/plain)并复制邮件

<delete-message>	d	删除当前条目
<delete-thread>	^D	删除thread中所有邮件
<delete-subthread>	Esc d	删除子thread中所有邮件
<edit>	e	编辑原始邮件
<edit-type>	^E	修改附件的content type
<forward-message>	f	转发邮件
<flag-message>	F	标记为重要
<group-reply>	g	回复所有收件人
<imap-fetch-mail>		从IMAP服务器强行抓取邮件
<display-toggle-weed>	h	显示邮件并开关邮件头的显示
<next-undeleted>	j	移动到下一封未删除的邮件
<next-entry>	J	移动到下一个条目
<previous-undeleted>	k	移动到上一封未删除的邮件
<previous-entry>	K	移动到上一个条目
<link-threads>	&	连接标记的邮件到当前邮件
<list-reply>	L	回复指定的邮件列表
<redraw-screen>	^L	清除并重绘屏幕
<mail>	m	撰写新邮件
<mark-as-new>	N	标志为新邮件
<search-next>	n	查找下一个匹配项
<next-thread>	^N	跳转到下一个thread
<next-subthread>	Esc n	跳转到下一个子thread
<print-message>	p	打印当前条目
<previous-thread>	^P	跳转到上一个thread
<previous-subthread>	ESC p	跳转到上一个子thread
<quit>	Q	保存更改并退出
<exit>	q	退出此菜单
<reply>	r	回复邮件
<recall-message>	R	打开发送暂缓的邮件
<read-thread>	^R	标记当前thread为已读
<read-subthread>	ESC r	标记当前子thread为已读
<resend-message>	Esc e	以当前邮件为模板发送新的邮件
<save-message>	s	保存邮件/附件
<skip-quoted>	S	跳过引文
<decode-save>	Esc s	解码邮件(text/plain)并移动邮件

<tag-message>	t	标记当前条目
<toggle-quoted>	T	切换是否显示引文
<undelete-message>	u	取消删除当前条目
<undelete-subthread>	Esc u	取消删除子thread中的所有邮件
<undelete-thread>	^U	取消删除thread中的所有邮件
<view-attachments>	v	查看MIME附件
<show-version>	V	显示Mutt的版本号和发布日期
<search-toggle>	\	切换是否高亮显示匹配项
<display-address>	@	显示发件人完整的地址
<next-new>		跳转到下一封新邮件
<pipe-message>		pipe邮件/附件到一个shell命令
<help>	?	显示当前菜单的帮助
<next-page>	<Space>	移动到下一页
<previous-page>	-	移动至上一页
<top>	^	跳转到邮件顶部
<sync-mailbox>	\$	保存信箱的更改
<shell-escape>	!	在子shell调用程序
<enter-command>	:	输入muttrc配置命令
<buffy-list>	.	列出包含新邮件的信箱
<search>	/	使用正则表达式搜索
<search-reverse>	ESC /	使用正则表达式向后搜索
<search-opposite>		以相反的方向查找下一个匹配项
<next-line>	<Return>	向下滚动一行
<jump>		跳转到指定序号的条目
<next-unread>		跳转到下一封未读邮件
<previous-new>		跳转到上一封新邮件
<previous-unread>		跳转到上一封未读邮件
<half-up>		向上滚动1/2页
<half-down>		向下滚动1/2页
<previous-line>		向上滚动一行
<bottom>		跳转到邮件的底部
<parent-message>	P	跳转到thread的父邮件
<check-traditional-pgp>	Esc P	检查传统的PGP方式
<mail-key>	ESC k	发送一个PGP公钥
<extract-keys>	^K	提取公钥

<forget-passphrase>	^F	清除内存中缓存的密码
<decrypt-copy>		解密后复制
<decrypt-save>		解密后移动
<what-key>		显示按键的keycode

4.4. 别名菜单

表8.5. 别名菜单的默认键绑定

功能	默认键绑定	描述
<delete-entry>	d	删除当前条目
<undelete-entry>	u	取消删除当前条目

4.5. query菜单

表8.6. query菜单默认键绑定

功能	默认键绑定	描述
<create-alias>	a	为发件人创建一个别名
<mail>	m	撰写新邮件
<query>	Q	从外部程序查询地址
<query-append>	A	append new query results to current results

4.6. 附件菜单

表8.7. 附件菜单默认键绑定

功能	默认键绑定	描述
<bounce-message>	b	remail给其他人
<display-toggle-weed>	h	显示邮件并开关邮件头的显示
<edit-type>	^E	修改附件的content type
<print-entry>	p	打印当前条目
<save-entry>	s	保存邮件/附件
<pipe-entry>	l	pipe邮件/附件到一个shell命令
<view-mailcap>	m	强制使用mailcap查看附件
<reply>	r	回复邮件
<resend-message>	ESC e	以当前邮件为模板发送新的邮件
<group-reply>	g	回复所有收件人
<list-reply>	L	回复到指定的邮件列表

<forward-message>	f	转发邮件
<view-text>	T	以文本方式查看附件
<view-attach>	<Return>	使用mailcap查看附件，如果需要的话。
<delete-entry>	d	删除当前条目
<undelete-entry>	u	取消删除当前条目
<collapse-parts>	v	切换是否显示子部分
<check-traditional-pgp>	Esc P	检查传统的PGP方式
<extract-keys>	^K	提取公钥
<forget-passphrase>	^F	清除内存中缓存的密码

4.7. 编辑菜单

表8.8. 编辑菜单默认键绑定

功能	默认键绑定	描述
<attach-file>	a	附加文件到邮件
<attach-message>	A	将邮件当作附件发送
<edit-bcc>	b	编辑BCC字段
<edit-cc>	c	编辑CC字段
<copy-file>	C	保存邮件/附件
<detach-file>	D	删除当前条目
<toggle-disposition>	^D	切换附件的disposition，inline与attachment之间
<edit-description>	d	编辑附件描述
<edit-message>	e	编辑邮件
<edit-headers>	E	编辑邮件，包括邮件头
<edit-file>	^X e	编辑附加的文件
<edit-encoding>	^E	编辑附件的传输编码
<edit-from>	ESC f	编辑from字段
<edit-fcc>	f	编辑fcc字段
<filter-entry>	F	filter attachment through a shell command
<get-attachment>	G	获得附件的副本
<display-toggle-weed>	h	显示邮件并开关邮件头的显示
<ispell>	i	运行拼写检查程序ispell
<print-entry>	l	打印当前条目
<edit-mime>	m	使用mailcap编辑附件
<new-mime>	n	compose new attachment using mailcap entry
<postpone-message>	P	暂缓发送

<edit-reply-to>	r	编辑Reply-To字段
<rename-file>	R	重命名/移动附件
<edit-subject>	s	编辑subject字段
<edit-to>	t	编辑To字段
<edit-type>	^T	修改附件的content type
<write-fcc>	w	保存邮件到文件夹
<toggle-unlink>	u	切换是否在发送后删除文件
<toggle-recode>		是否重新编码附件
<update-encoding>	U	更新附件的编码信息
<view-attach>	<Return>	使用mailcap查看附件，如果需要的话
<send-message>	y	发送邮件
<pipe-entry>	l	pipe邮件/附件到一个shell命令
<attach-key>	ESC k	附加一个PGP公钥
<pgp-menu>	p	显示PGP选项
<forget-passphrase>	^F	清除内存中缓存的密码
<smime-menu>	S	显示S/MIME选项
<mix>	M	通过mixmaster转投链发送邮件

4.8. postpone菜单

表8.9. postpone菜单默认键绑定

功能	默认键绑定	描述
<delete-entry>	d	删除当前条目
<undelete-entry>	u	取消删除当前条目

4.9. 浏览器菜单

表8.10. 浏览器菜单默认键绑定

功能	默认键绑定	描述
<change-dir>	c	更改目录
<display-filename>	@	显示当前选定文件的名称
<enter-mask>	m	enter a file mask
<sort>	o	排序
<sort-reverse>	O	反向排序
<select-new>	N	在此目录选择一个新的文件
<check-new>		检查新邮件

<toggle-mailboxes>	<Tab>	切换浏览信箱或所有文件
<view-file>	<Space>	查看文件
<buffy-list>	.	列出包含新邮件的信箱
<create-mailbox>	C	创建新信箱（IMAP only）
<delete-mailbox>	d	删除当前信箱（IMAP only）
<rename-mailbox>	r	重命名当前信箱（IMAP only）
<subscribe>	s	订阅当前信箱（IMAP only）
<unsubscribe>	u	退订当前信箱（IMAP only）
<toggle-subscribed>	T	切换查看已订阅或所有信箱（IMAP only）

4.10. PGP菜单

表8.11. PGP菜单默认键绑定

功能	默认键绑定	描述
<verify-key>	c	验证PGP公钥
<view-name>	%	查看密钥的user id

4.11. smime菜单

表8.12. smime菜单默认键绑定

功能	默认键绑定	描述
<verify-key>	c	验证PGP公钥
<view-name>	%	查看密钥的user id

4.12. mix菜单

表8.13. mix菜单默认键绑定

函数	默认键绑定	描述
<accept>	<Return>	确认当前链
<append>	a	附加一个转投服务器到链中
<insert>	i	插入一个转投服务器到链中
<delete>	d	从链中删除一个转投服务器
<chain-prev>	<Left>	选择当前链中的上一个元素
<chain-next>	<Right>	选择当前链中的下一个元素

4.13. 编辑器菜单

表8.14. 编辑器菜单默认键绑定

功能	默认键绑定	描述
<bol>	^A	跳转到行的开头
<backward-char>	^B	向左移动一个字符
<backward-word>	Esc b	移动光标到单词的开头
<capitalize-word>	Esc c	大写词首
<downcase-word>	Esc l	转换单词为小写
<upcase-word>	Esc u	转换单词为大写
<delete-char>	^D	删除光标下的字符
<eol>	^E	跳转到行末
<forward-char>	^F	向右移动一个字符
<forward-word>	ESC f	移动光标到词尾
<backspace>	<Backspace>	删除光标前面的字符
<kill-eol>	^K	删除从光标处到行末的字符
<kill-eow>	Esc d	删除从光标出到单词末尾的字符
<kill-line>	^U	删除所有字符
<quote-char>	^V	quote the next typed key
<kill-word>	^W	删除从光标到词首的字符
<complete>	<Tab>	补全文件名或别名
<complete-query>	^T	通过query补全地址
<buffy-cycle>	<Space>	cycle among incoming mailboxes
<history-up>		上一条历史记录
<history-down>		下一条历史记录
<transpose-chars>		transpose character under cursor with previous

第9章. 杂项

目录

- [1. 致谢](#)
- [2. 关于本文档](#)

1. 致谢

Kari Hurtta <kari.hurtta@fmi.fi> co-developed the original MIME parsing code back in the ELM-ME days.

下面这些人对Mutt的开发提供了很多帮助：

- Vikas Agnihotri <vikasa@writeme.com>

- Francois Berjon <Francois.Berjon@aar.alcatel-alsthom.fr>
- Aric Blumer <aric@fore.com>
- John Capo <jc@irbs.com>
- David Champion <dgc@uchicago.edu>
- Brendan Cully <brendan@kublai.com>
- Liviu Daia <daia@stoilow.imar.ro>
- Thomas E. Dickey <dickey@herndon4.his.com>
- David DeSimone <fox@convex.hp.com>
- Nickolay N. Dudorov <nnd@wint.itfs.nsk.su>
- Ruslan Ermilov <ru@freebsd.org>
- Edmund Grimley Evans <edmund@rano.org>
- Michael Finken <finken@conware.de>
- Sven Guckes <guckes@math.fu-berlin.de>
- Lars Hecking <lhecking@nmrc.ie>
- Mark Holloman <holloman@nando.net>
- Andreas Holzmann <holzmann@fmi.uni-passau.de>
- Marco d'Itri <md@linux.it>
- Björn Jacke <bjacke@suse.com>
- Byrial Jensen <byrial@image.dk>
- David Jeske <jeske@igcom.net>
- Christophe Kalt <kalt@hugo.int-evry.fr>
- Tommi Komulainen <Tommi.Komulainen@iki.fi>
- Felix von Leitner (a.k.a ¿Fefe¿) <leitner@math.fu-berlin.de>
- Brandon Long <blong@fiction.net>
- Jimmy Mäkelä <jmy@flashback.net>
- Lars Marowsky-Bree <lmb@pointer.in-minden.de>
- Thomas ¿Mike¿ Michlmayr <mike@cosy.sbg.ac.at>
- Andrew W. Nosenko <awn@bcs.zp.ua>
- David O'Brien <obrien@Nuxi.cs.ucdavis.edu>

- Clint Olsen <olsenc@ichips.intel.com>
- Park Myeong Seok <pms@romance.kaist.ac.kr>
- Thomas Parmelan <tom@ankh.fr.eu.org>
- Ollivier Robert <roberto@keltia.freenix.fr>
- Thomas Roessler <roessler@does-not-exist.org>
- Roland Rosenfeld <roland@spinnaker.de>
- Rocco Rutte <pdmef@gmx.net>
- TAKIZAWA Takashi <taki@luna.email.ne.jp>
- Allain Thivillon <Allain.Thivillon@alma.fr>
- Gero Treuner <gero@70t.de>
- Vsevolod Volkov <vvv@lucky.net>
- Ken Weinert <kenw@ihs.com>

2. About this document

这份文档使用[DocBook](#)写成， 要翻译它的话可以使用[Gnome XSLT toolkit](#)。

[Close](#)

Page Saved

- [Open Pocket](#)
-
- [Remove Page](#)
-
-
-
-
-

[Save](#)