Mutt 사용 가이드

Michael Elkins

me (at) cs.hmc.edu

임은재

eunjea (at) kldp.org

v 1.3.25, 2002/01/15

이 문서는 이메일 클라이언트(MUA)인 Mutt의 사용, 설정 가이드입니다. 이 문서는 mutt 패키지에 들어 있는 linuxdoc 형태의 문서를 번역, 편집하여 docbook 형식으로 변환한 것입니다. 이 문서의 마지막 버전은 http://kldp.org/~eunjea/에 있습니다.

고친 과정

고침 1.3.252002-01-15고친이 eunjea
v1.3.25

고친 과정

고침 1.3.232001-12-06고친이 eunjea
v1.3.23

1. 소개

Mutt 는 작지만 매우 강력한 텍스트 기반의 MIME 메일 클라이언트이다. mutt은 키 바인딩, 키보드 매크로등의 유연하고 자유로운 설정이 가능하며, 메일 쓰레딩, 정규 표현식 검색등을 지원하는등 파워 유저에게 알맞는 메일 클라이언트이다.

1.1. Mutt 홈 페이지

http://www.mutt.org/

mutt 한글 사용자들을 위해 http://mutt.kldp.org (http://mutt.kldp.org/)가 마련되어 있다.

1.2. 메일링 리스트

메일링 리스트에 가입하고자 한다면, 메일 본문에 **subscribe** 라고 써서 메일링-리스트 이름**-request**`@mutt.org` 로 보내면 된다.

mutt 한글 메일링 리스트: http://list.kldp.org/mailman/listinfo/mutt

- mutt-announce@mutt.org (mailto:mutt-announce-request@mutt.org) -- 공지 사항
- mutt-users@mutt.org (mailto:mutt-users-request@mutt.org) -- 도움, 버그 리포팅과 기능 신청
- mutt-dev@mutt.org (mailto:mutt-dev-request@mutt.org) -- 개발 메일링 리스트

주의**: mutt-announce** 에 포스팅되는 글은 **mutt-users** 에도 자동적으로 올라오므로 양쪽에 가입할 필요가 없다.

1.3. 배포 사이트

- ftp://ftp.mutt.org/pub/mutt/
- ftp://ftp.guug.de/pub/mutt/

미러 사이트의 목록 : http://www.mutt.org/download.html.

1.4. IRC

Mutt에 관심있는 사람들과 채팅을 하고자 한다면 OpenProjects.Net (www.openprojects.net) (http://www.openprojects.net)의 **#mutt** 채널을 이용해라.

1.5. 뉴스 그룹 (유즈넷)

comp.mail.mutt (news:comp.mail.mutt).

1.6. 저작권

Mutt is Copyright (C) 1996-2000 Michael R. Elkins <me@cs.hmc.edu> and others

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111, USA.

2. 시작 하기

이번장은 Mutt의 사용법에 대해 간단하게 알아본다. 더 많은 기능에 대한 설명이 이 문서에 존재하며, 그외 많은 정보를 Mutt FAQ 와 다양한 웹 페이지에서 찾을수 있다. Mutt 홈페이지 (http://www.mutt.org/)를 참고해라.

여기서 설명하는 모든 키설정은 기본값이다. 이 값은 배포본에 의해 만들어진 패키지이거나 시스템 관리자가 시스템 설정파일을 변경하였을수 있으므로 아무 메뉴에서나 "?"를 사용하면 현재 키 바인딩 상태를 알수 있다.

첫번째 할것은 간단히 명령줄에서 mutt를 쳐서 mutt를 실행시키는 것이다. 다양한 명령줄(command-line) 옵션이 있으며, man 페이지 또는 실행 옵션를 참고하라.

2.1. 메뉴에서 이동

다음은 메뉴에서의 일반적인 키이다.

j 또는 Down	next-entry	다음 항목으로 이동
k 또는 Up	previous-entry	이전 항목으로 이동
z 또는 PageDn	page-down	다음 페이지
Z 또는 PageUp	page-up	이전 페이지
= 또는 Home	first-entry	처음 항목으로 이동
* 또는 End	last-entry	마지막 항목으로 이동
q	quit	현재 메뉴에서 나감
?	help	현재 메뉴의 모든 키바인딩 보여주기

2.2. 입력란에서의 편집

Mutt은 내장 라인 에디터를 가지고 있으며, 이것은 이메일 주소, 제목, 파일 이름등을 입력할 때 사용된다. 이때 사용되는 키들은 Emacs의 그것과 매우 비슷하다.

^A 또는 <Home>	bol	처음으로 이동
^B 또는 <Left>	backward-char	왼쪽으로 이동
Esc B	backward-word	단어의 처음으로 이동

```
^D 또는 <Delete> delete-char     커서의 글자 삭제
^E 또는 <End>    eol             줄의 끝으로 이동
^F 또는 <Right>  forward-char    오른쪽으로 이동
Esc F forward-word 단어의 끝으로 이동
<Tab>           complete         파일 이름, 알리아스 (이메일 주소) 완성 하기
^T              complete-query   쿼리로 주소 완성하기
^K              kill-eol         커서 오른쪽 라인 삭제
ESC d kill-eow 커서 오른쪽 단어 삭제
^W              kill-word        커서 왼쪽 단어 삭제
^U              kill-line        한줄 삭제
<Up>            history-up       기억된 것에서 이전 문자
<Down>          history-down     기억된 것에서 다음 문자
<BackSpace>     backspace        커서 앞쪽의 글자 지우기
Esc u upcase-word 대문자로 변환
Esc l downcase-word 소문자로 변환
Esc c capitalize-word 단어를 대문자로 시작
^G              n/a              취소
<Return>        n/a              편집 끝냄
```

이 키들은 bind 명령을 사용해서 재 정의 할수 있다. 예를 들어 **Delete** 키를 backspace 키와 같은 역할을 하도록 한다면,

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

2.3. 메일 읽기 - 인덱스(Index)와 페이저(Pager)

다른 메일 클라이언트처럼 Mutt 도 두가지 모드를 가지고 있다, 현재 메일함이 포함하고 있는 메일의 목록을 보여주는 인덱스(index)와 메일의 내용을 보는 페이저(pager)이다.

다음은 각 모드의 기능들과 키이다.

2.3.1. 메일 인덱스

```
c               다른 메일함 열기
ESC c           메일함을 읽기 전용으로 열기
C               현재 메일을 다른 메일함으로 복사
ESC C           메일를 디코딩하고 폴더로 복사
ESC s           메일를 디코딩하고 폴더에 저장
D               패턴(pattern)에 일치하는 메일 삭제
d               현재 메일 삭제
F               중요 표시하기
l               패턴(pattern)에 일치하는 메일 표시
N               새 메일로 표기
o               정렬 방법 바꾸기
O               역으로 정렬하기
q               편지함에 변경 사항 저장 후 끝내기
```

```
s               메일/첨부물 파일로 저장
T               패턴과 일치하는 메일에 태그 붙임
t               태그 붙이기
ESC t           글타래 전체에 태그 붙임
U               패턴과 일치하는 메일 복구
u               현재 메일 복구
v               첨부물 보여주기
x               바뀐 사항을 저장하지 않고 메뉴 나감
<Return>        메일 읽기
<Tab>           다음 읽지 않은 메일로 가기
@               글쓴이 이메일 주소 보기
$               변경 사항 저장
/               찾기
ESC /           반대 방향으로 찾기
^L              화면 정리하고 다시 그리기
^T              패턴과 일치하는 메일 태그 지움
```

2.3.1.1. 상태 표시 플래그 (flag)

다음은 각 메일의 상태를 표시해주는 플래그의 의미이다:

D

삭제됨 (삭제를 위해 표시됨)

d

첨부물을 포함하고 있는 메일로 삭제 표시됨

K

PGP 공개키를 포함

N

새 메일

O

오래된 메일

P

PGP 암호화된 메일

r

답장을 보낸 메일

S

PGP 사인된 메일로, 성공적으로 서명 확인이 됨

s

PGP 사인된 메일

!

중요 메일

*

태그

몇가지 상태 플래그는 끄거나 켤수 있다.

- **set-flag** (기본: w)
- **clear-flag** (기본: W)

또한, 다음 플래그는 각 메일의 주소에 따라 반영되며, to-chars 변수를 사용하여 바꿀수 있다.

```
+       당신에게만 보내진 메일
T       여러명에게 보내지거나 cc 에 다른 메일주소가 있는 경우
C       cc 로 보내진 메일
F       당신이 보낸 메일
L       가입된 메일링 리스트에서 보내진 메일
```

2.3.2. 페이저 (Pager)

기본적으로 mutt은 메일 본문을 표시하는데 내장된 페이저를 사용하게 된다. 이 내장 페이저는 **less**와 흡사하다.

```
<Return>        한줄 아래로
<Space>         다음 페이지 (메일의 끝부분일 경우 다음 메일로 이동)
-               이전 메일로
n               다음 찾기
S               인용문 다음으로 넘어가기
T               인용문의 화면표시 상태 바꾸기
?               키정의 보여주기
/               정규 표현식을 이용해 검색
ESC /           정규 표현식을 이용해 역순 검색
\               검색 패턴 컬러링 선택
^               메일의 처음으로 이동
$               메일의 끝으로 이동
```

인덱스(index)에서의 기능들이 역시 페이저에서도 작동하는데, 예를 들어 **delete-message**(메일 삭제) 나 **copy-message**(메일 복사)등이다. (이것은 내장 페이저를 사용할 때의 이점이다.)

또한 내장된 페이저는 몇가지 부가적인 기능을 지원하다. 첫번째는 "표준" nroff sequence가 메일 본문에 사용될 경우 이것을 번역해서 볼드, 밑줄 글자들을 그려낸다. 이것은 터미날의 지원을 필요로 하며, 만약 지원하지 않는다면 이것들을 위해서 색(color)을 지정해 줄수 있다.

또한 내장 페이저는 ANSI escape sequence를 지원하여 글자의 속성을 바꾸는 것이 가능하다. Mutt은 이것을 해당 색과 글자의 속성으로 번역한다. mutt이 지원하는 ANSI sequence:

```
ESC [ Ps;Ps;Ps;...;Ps m
Ps 는
0    모든 속성 끔
1    볼드
4    밑줄
5    깜박임
7    Reverse video 켬
3x   전경 색 x
4x   배경 색 x

색
0    검정
1    빨강
2    초록
3    노랑
4    청색
5    진홍
6    청록
7    백색
```

Mutt에서 이 속성들은 text/enriched 메세지를 처리하는데 사용하며, 이것은 또한 외부 MIME 자동보기 스크립트를 위한 하이라이팅(highlighting) 목적으로 사용한다.

2.3.3. 글타래 모드

메일이 **thread**(글타래)모드로 정렬(sort)되었을 경우, 인덱스와 페이저 모드에서 몇가지 추가적인 기능이 있다.

```
^D      delete-thread           현재 글타래의 모든 메일 삭제
^U      undelete-thread         현재 글타래의 모든 메일 복구
^N      next-thread             다음 글타래로 이동
^P      previous-thread         이전 글타래로 이동
^R      read-thread             현재 글타래를 읽은 것으로 표시
ESC d   delete-subthread        부속 글타래의 모든 메세지 지움
ESC u   undelete-subthread      부속 글타래의 모든 메세지 복구하기
ESC n   next-subthread          다음 부속 글타래로 이동
ESC p   previous-subthread      이전 부속 글타래로 이동
```

```
ESC r   read-subthread          현재 부속 글타래에 읽은 표시하기
ESC t   tag-thread              현재 글타래에 태그 붙임
ESC v collapse-thread 현재 글타래 펴기/접기
ESC V collapse-all 모든 글타래들 펴기/접기
P       parent-message          글타래의 부모 메세지로 이동
```

글타래를 접거나 펼수 있는데 접었을 때에는 그 글타래의 첫번째 메일만 표시되고 나머지는 메일들은 숨겨지게 된다. 이것은 글타래에 매우 많은 메일이 포함되어 있을때 유용하게 사용될수 있다.

Tip: %M 을 index-format 변수에서 참고해라. 예를 들어 index-format를 "%?M?(#%03M)&(%4l)?" 와 같이 설정하면 글타래에 몇개의 메일이 포함되어 있는지 표시해 준다.

참고: strict-threads.

2.3.4. 그밖의 기능

create-alias (기본: a)

새로운 알리아스를 만드는 것은 현재 메일의 주소를 추가하거나 새로운 메일 주소를 추가할 때 사용된다. 편집이 끝나면 alias-file변수에 정의된 파일에 alias가 추가된다.

주의**:** alias-file 변수를 사용할때에는 source변수와 함께 사용 되어야 한다.

check-traditional-pgp (기본: ESC P)

이 기능은 전통적인(MIME 이 아닌) PGP 사인/암호화된 메일을 찾아준다. 기술적으로, 이 기능은 PGP 데이타를 가지고 있는 메일의 MIME 내용의 타입을 일시적으로 바꾼다. edit-type 또한 비슷한 역할을 한다.

display-toggle-weed (기본: h)

메일 헤더를 보여주거나 숨긴다. ignore 참고.

edit (기본: e)

(인덱스와 페이저 모드에서 가능) 현재 메일을 메일함에 저장된 상태 그대로 수정할수 있게 해준다. 수정이 끝나면 현재 메일함에 저장되며 원본 메일은 삭제 표시된다.

edit-type (기본: ^E 첨부물 메뉴, 페이저, 인덱스, ^T 작성 메뉴)

첨부물의 유형을 수정할때 사용한다.

enter-command (기본: ":")

설정파일에서 사용할수 있는 명령들을 직접 입력할때 사용된다. 일반적으로 변수의 설정을 확인하거나 macro를 mutt 실행중에 바꿀때 사용할수 있다.

extract-keys (기본: ^K)

현재 또는 태그한 메일에서 PGP 공개키를 추출하여 공개 키 ring에 추가 한다.

forget-passphrase (기본: ^F)

PGP 암호구문(passphrase)을 메모리에서 삭제한다. 암호를 잘못 입력했을때 유용하다.

list-reply (기본: L)

지정된 메일링 리스트로 답장하기, 현재 메일에 포함된 이메일 주소가 lists에 지정된 메일링 리스트 주소와 일치하면 메일링 리스트로 답장을 보낸다.

pipe-message (기본: |)

현재 메일이나 태그된 메일을 외부 명령어로 연결(pipe)할때 쓴다. 관련 변수인 pipe-decode, pipe-split, pipe-sep, wait-key 를 참조할것.

resend-message (기본: ESC e)

메일 다시 보내기, 현재 메일을 템플레이트로 새로운 메일을 보낸다. 이 기능은 MIME 메일을 포워딩하면서 원하는 데로 수정하여 보낼때 편리하게 사용될수 있다. 이때 포함되는 메일의 헤더는 weed에 의해 결정된다. 이 기능은 첨부물 메뉴에서도 가능하다.

shell-escape (기본: !)

하부 쉘 실행하기. wait-key변수로 명령 실행후 상태를 조절할수 있다.

toggle-quoted (기본: T)

인용문 보기/숨기기. 이때 인용문의 인식은 quote-regexp 변수에 의해 결정할수 있다.

skip-quoted (기본: S)

페이저에서 인용문을 무시하고 메일 본문의 내용을 첫번째 줄에 표시한다.

2.4. 메일 보내기

다음의 키들이 메일을 보내기 위해 사용된다.

```
m       compose         메일 작성하기
r       reply           답장 작성
g       group-reply     모든 수신자에게 답장
L       list-reply      메일링 리스트로 답장
f       forward         메일 포워딩
b       bounce          메일 전달 (바운스,bounce,remail)
ESC k   mail-key        PGP 공개키 보내기
```

메일 전달과 포워딩에 관한 자세한것은 forwarding-mail을 참조하라.

기본적으로 mutt는 메일 작성와 답장, 포워딩시에 수신자(To:)와 제목(Subject:)를 먼저 물어보는데, 이것은 askcc, askbcc, autoedit, fast-reply 변수들로 조정이 가능하다.

다음, 메일 본문의 편집은 editor 에 지정된 편집기가 사용된다(기본은 vi).

edit-headers가 설정된 경우에는 메일의 헤더를 편집할수 있으며, 관련 변수로는 attribution, indent-string, post-indent-string를 사용할수 있다.

메일 포워딩시에 mime-forward가 설정되어 있을 경우 원본은 첨부물로 보내지게 된다. signature 가 설정되어 있는 경우 메일에 추가된다.

메일 본문 작성이 끝나면 메일 작성 메뉴(compose)로 돌아오게 되며 다음과 같은 명령을 사용할수 있다:

```
a       attach-file             첨부물 추가
A attach-message 현재 메일에 메일 추가
ESC k   attach-key              PGP 공개키 추가
d       edit-description        첨부물 설명 수정
D       detach-file             첨부물 삭제
t       edit-to                 To 수정
ESC f   edit-from               From 수정
r       edit-reply-to           Reply-To 수정
c       edit-cc                 Cc 수정
b       edit-bcc                Bcc 수정
y       send-message            메일 보내기
s       edit-subject            제목 수정
f       edit-fcc                복사본을 저장할 메일함 지정
p       pgp-menu                PGP 옵션 고르기 ("i" 버전만 해당)
P       postpone-message        현재 메일를 저장 후 나중에 보냄
q       quit                    메일 보내기 취소
w write-fcc 메일 저장
i       ispell                  ispell로 스펠링 검사 (ispell이 있는 경우)
^F      forget-passphrase       메모리에서 PGP 암호 문구 지움
```

참고**:** 현재 메일에 다른 메일을 첨부할때는 특정 메일함을 선택한후 첨부할 메일에 태그를 하면 된다.

2.4.1. 메일 헤더 수정하기

보내는 메일의 헤더를 수정하여 다음 두가지의 특별한 목적으로 사용할수 있다.

`Fcc:` 파일 이름 이것은 작성 메뉴에서 edit-fcc 와 같은 기능(복사본을 저장할 파일)을 한다.

`Attach:` 파일 이름 [설명] 첨부할 파일의 이름과 설명을 직접 지정해 줄수도 있다.

답장의 헤더에서 In-Reply-To: 헤더를 삭제할 경우 mutt는 References: 헤더를 자동으로 추가하지 않게되므로, 이것은 새로운 글타래를 만들게 된다.

edit-headers 참고.

2.4.2. PGP와 Mutt

PGP의 사용을 원한다면, 다음과 같이 지정이 가능하다.

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

“E” 암호화, “S” 사인됨, ‘S<id>” id의 pgp 키(이것은 pgp-sign-as변수로 설정이 가능하다.)

mutt 상에서 메일을 PGP 암호화 하여 보낼때 어떤 pgp 키를 사용할 것인지 물어보는 과정은 보통 수신자의 이메일 주소를 인증된 ID 에서 찾지 못하는 경우이다.

또한, 상황에 따라 다른 pgp 키를 사용해야 할 경우에는 메일 보내기 메뉴에서 (기본: p 와 a)로 pgp 키를 고를수 있다. 취소는 `^G` 로 한다.

원하는 pgp 키를 고르면 메일은 선택된 공개키를 사용하여 암호화되어 보내진다.

pgp 키 고르기 메뉴 (pgp-entry-format변수를 참고)에서 보여주는 pgp 키 목록은 각 pgp 키들의 상태를 보여주는데 다음과 같은 플래그를 가질수 있으며, 의미는 다음과 같다.

%f (플래그 필드):

```
R               취소된(revoked)된 pgp 키로 사용할수 없음
X               유효 기간이 지난 pgp 키로 사용할수 없음
d               사용할수 없도록 표시됨
c               인식하지 못하는 self-signature 패킷이 있음
```

%c (능력 필드)는 두개의 문자로 표시된다:

(**-** 마이너스 기호) 암호화에 사용될수 없는 pgp 키

(**.** 마침표) 유저 ID중 서명키로 표시되었지만, 암호화에 사용할수 있는 pgp 키.

(**e**) pgp 키가 암호화에 사용될수 있음.

두번째 문자는 pgp 키의 사인 가능성을 보여준다:

(**-** 마이너스 기호) 사인 할수 없음

(**.** 마침표) 유저 ID중에서 서명키로 표시됨.

(**s**) 사인 가능한 pgp 키.

마지막으로 %t 필드는 인증 여부를 보여준다.

(**?**) 인증되지 않음, (**-**) 는 신뢰할수 없는 조합, 공백은 부분적으로 신뢰되는 조합으로 표시됨, (**+**) 는 완전히 인증된 것을 뜻한다.

2.4.3. mixmaster를 통한 익명 메일 보내기

mutt은 Mixmaster와 같이 사용하도록 설정될수 있다. Mismaster는 remailer 체인을 사용하여 익명 메일을 보내는데 사용된다. mutt이 지원하는 mixmaster 버전은 2.04 와 2.03이다.

이것을 사용하기 위해서는 몇가지 제한점이 있는데, 가장 중요한 것은 `Cc` 와 `Bcc` 헤더를 사용할수 없다는 것이다. mutt이 mixmaster를 사용하도록 하기 위해서 compose 메뉴상의 mix 기능을 사용해서 remailer 체인을 선택해야 한다.

체인 선택 화면은 둘로 나눠지며, 위쪽에는 사용할수 있는 remailer의 목록이 표시되고, 아래쪽에는 현재 선택된 체인이 표시된다.

체인의 사용은 `chain-prev` 과 `chain-next` 기능으로 조정할수 있으며, 각각 왼쪽/오른쪽 화살표 키와 `h` 와 `l`키가 기본값이다. remailer을 현재 체인에 넣기 위해 서는 `insert`을 사용한다. remailer의 추가는 `select-entry` 와 `append`가 사용된다.

remailer마다 서로 다른 기능을 가지고 있는데 이것은 remailer 메뉴의 %c 값으로 표시된다 (mix-entry-format를 참고). 자세한 기능들은 mixmaster 문서를 참고하라.

2.5. 메일 포워딩과 바운스

바운싱과 포워딩은 받은 메일을 다른 수신인에게 보내는 것이다.

메일을 바운스 할때는 sendmail명령을 사용해서 원본의 수신자가 새로 지정한 수신인인 것처럼 보내지며, 포워딩은 원본을 수정하거나 자신의 글을 추가할때 사용한다.

다음과 같은 키가 기본 설정이다:

```
f       forward         포워드
b       bounce          전달 (bounce,remail)
```

포워딩은 원본 메세지를 새 메일의 본문으로 사용하거나 MIME 첨부물로 보낼수 있다. (이것은 mime-forward 변수에 의해 결정된다.)

첨부물의 디코딩은 forward-decode 와 mime-forward-decode 변수로 조절할수 있다.

포워딩 메일의 형식은 내용에 따라 달라질수 있으며, mime-forward 변수의 값은 quadoption(yes,no,ask-yes/no)이 될수 있다.

또한 포함되는 원본 메일의 헤더는 weed 변수가 적용된다 (mime-forward로 보내지지 않을때만 해당).

포워딩하는 메일은 일반적인 메일 작성이나 답장 보내기와 같은 절차를 거친다.

2.6. 발송 연기

이미 작성한 메일의 발송을 미룰때 사용되는 기능이다. 발송이 연기된 메일과 첨부물은 postponed 변수에 설정한 메일함에 저장되며, 필요할때 다시 불러 쓸수 있다.

발송 연기된 메일을 다시 불러오는 방법은 mutt를 "-p" 옵션을 주고 실행하거나 메일 작성 메뉴에서는 mutt이 발송 연기된 메일을 불러올것인지 자동적으로 물어볼 것이다. 발송 연기된 메일이 하나 이상일때는 어떤 메일을 불러올것인지 선택할수 있다.

주의**:** 발송 연기된 메일이 답장일 경우, 원본 메일의 상태는 답장이 보내지기 전까지 변하지 않으며, 원본이 위치한 메일함에서 보내져야만 상태가 업데이트 된다.

postpone 변수 참고.

3. 설정

이미 소개했듯이 mutt은 매우 유연한 설정을 가지고 있으며, 이를 이용해 자신만의 mutt를 만들수 있다.

mutt이 실행될때 "-n" 실행 옵션을 사용하지 않는 이상 기본적으로 "시스템" 설정 파일을 먼저 읽게 되며, 이 시스템 설정 파일은 일반적으로 `/usr/local/share/mutt/Muttrc` 나 `/etc/Muttrc`에 위치한다. 다음 mutt는 홈 디렉토리의 `.muttrc`을 찾고, 이것이 없다면 홈 디렉토리의 하위 디렉토리(`~/.mutt/muttrc`)에서 muttrc 파일을 찾게 된다.

일반적으로 홈 디렉토리의 `.muttrc`파일을 명령어 설정 파일로 사용하게 될것이다.

추가적으로 mutt는 버전을 명시한 파일이름을 지원하는데 예를 들어 mutt의 버전이 0.88이면 `.muttrc-0.88` 를 설정파일로 사용하게 된다. 즉, 홈 디렉토리에 `.muttrc`와 `.muttrc-0.88`이 같이 존재 한다면 `.muttrc-0.88`가 사용된다.

mutt 의 버전은 "-v" 실행 옵션으로 확인하거나 mutt 실행중에는 `show-version` 키 (기본: V)를 누르면 알수 있다.

3.1. 문법

설정 파일의 내용은 명령어와 설정 변수들로 이루어진다. 한줄에는 하나 이상의 명령어가 올수 있으며, 이때에는 명령어들을 세미콜론(;)으로 구분한다.

```
set realname='임 은재' ; ignore x-
```

"#"는 주석을 달때 사용되며 # 이후의 문자는 당연히 무시된다.

```
my_hdr X-Disclaimer: 뭘봐요? # 이것은 주석 :-)
```

인용 부호 (',")는 공백이나 특별 문자가 포함될때 사용할수 있다. 두가지 인용부호의 차이점은 쉘 프로그램에서의 경우와 비슷하게 적용되는데 다음 예제를 보면 쉽게 이해할수 있을것이다.

만약 다음과 같이 문장사이에 " 를 넣으려면 \ 를 붙여준다.

```
set realname="Im \"Sexy\" Eunjea"
```

마찬가지로 "\\" 는 "\"가 들어갈 것이며, "\n" 와 "\r"는 C 에서와 마찬가지로 각각 라인피드(linefeed)와 캐리지 리턴(carriage-return)을 의미한다.

또한 \ 가 줄의 끝에 사용되면 다음줄은 하나의 명령으로 인식할 것이다. 이것은 매우 긴 명령을 보기 쉽도록 여러줄로 나누어 표기할때 사용될수 있다.

또한 (")를 사용해서 외부 명령어를 설정 파일에 사용할수 있다. 다음 예를 보라.

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

역시 환경변수의 값을 사용할수도 있다.

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

명령어 참고.

3.2. 알리아스 정의/사용

사용법: `alias` 키 주소 [, 주소, ...]

보통 자신과 이메일을 주고 받는 사람들의 이메일 주소를 기억하다는 것은 매우 귀찮은 일이므로, mutt은 이메일 주소록을 만들수 있도록 “aliases” 기능을 제공한다.

주의**:** 만약 일정 그룹(하나 이상의 주소)의 알리아스를 만들려 한다면, 각 주소는 (“,”)로 분리되어야 한다.

다음은 모든(*) 알리아스 정의를 삭제한다: `unalias` [* | **key ...**]

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

다른 메일 클라이언트들과 다르게 mutt은 알리아스 파일을 위해 다른 파일을 필요로 하지 않으며, `alias` 명령은 설정 파일의 어느곳에나 위치할수 있다. 하지만, source를 사용해서 여러개의 알리아스 파일을 지정해 사용할수도 있다.

예:

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

알리아스는 mutt이 주소를 물어보는 곳(**To:** 나 **Cc:** 등)에서 사용될수 있으며, 이때 TAB 키를 누르면 알리아스의 목록을 보여주고 원하는 주소를 선책 할수 있다. 여러개의 메일주소를 지정할 경우 각 메일주소의 구분은 (,) 로 해준다.

3.3. 키 설정 바꾸기

사용법: `bind` **map key function**

이 명령은 기본 키 설정을 원하는 키로 바꿀때 사용한다.

map (키가 사용될 메뉴)에 올수 있는 값은 다음과 같다:

- generic
- alias
- attach
- browser
- editor

- index
- compose
- pager
- pgp
- postpone

key 는 사용할 키이다. 콘트롤 문자를 사용할때에는 **\Cx** 의 형식으로 **x** 에 ctrl 과 같이 사용될 문자를 지정해준다 (예를 들어 control-A 는 "\Ca"). 대소문자는 무시되므로, **\CA, \Ca, \cA**, **\ca** 는 모두 같은 의미가 된다. 선택적으로 세자리 수의 키값을 "\" 와 함께 사용할수 있다 (예: **\177** 와 **\c?** 는 같은 의미이다).

추가적으로 **key** 값에 올수 있는 것들:

```
\t              tab
<tab>           tab
\r              carriage return
\n              newline
\e              escape
<esc>           escape
<up>            up arrow
<down>          down arrow
<left>          left arrow
<right>         right arrow
<pageup>        Page Up
<pagedown>      Page Down
<backspace>     Backspace
<delete>        Delete
<insert>        Insert
<enter>         Enter
<return>        Return
<home>          Home
<end>           End
<space>         Space bar
<f1>            function key 1
<f10>           function key 10
```

key 값은 공백을 포함하지 않는 이상 인용부호를 사용할 필요가 없다.

function 에는 **key** 가 눌렸을때 어떤 일을 할것인지 지정해준다. 기능들의 목록은 기능(함수) 편을 참고하라. `noop` 명령은 해당키의 기능을 끌때 사용된다.

3.4. 알리아스의 **character set** 설정

사용법: `charset-hook` **alias charset** 사용법: `iconv-hook` **charset local-charset**

`charset-hook` 명령은 문자셋의 알리아스를 지정할때 쓰인다. 이것은 메일의 문자셋을 mutt이 자동으로 인식하지 못할때 사용할수 있다.

`iconv-hook` 명령은 해당 문자셋의 이름이 시스템의 것과 다를때 지정해 줄수 있다.

3.5. 메일함 관련 설정

사용법: `folder-hook` [!]정규 표현식 명령어

각 메일함마다 다른 설정을 사용할때 쓰인다.

패턴은 정규 표현식으로 일치하는 메일함이 있는 경우 명령어가 실행된 후에 메일함을 읽는다.

주의**:** 만약 "!" 를 사용할 경우 정규 표현식은 인용 부호로 감싸줘야 한다.

다음 예는 모든 메일함을 보낸 시간으로 정렬한다:

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

다음 예는 mutt 메일함을 쓰레드 형식으로 정렬한다:

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

3.6. 키보드 매크로

사용법: `macro` **menu** 키 **sequence** [설명]

매크로는 일련의 키의 연속(sequence)을 하나의 키로 정의하는 것이다.

즉, **menu** 에서 키를 누르면 **sequence**를 입력한 것과 같은 역할을 하게 되는 것으로, 몇가지 명령을 하나의 키로 단축할수 있다. **key** 와 **sequence** 는 bind와 같은 요령으로 설정한다.

추가적으로 **sequence**에는 **^x**를 사용해서 콘트롤 문자를 사용하거나, 키의 이름(예:<UP>)이나 명령어들을 사용할수도 있다.

bind, 기능(함수) 참고.

sequence 대신에 명령어를 직접 정의하면 해당 명령어의 키 설정이 바뀌어도 그대로 사용할 수 있다는 장점이 있다. 다음 예를 보라.

```
folder-hook . 'macro index d "<save-message>=trash\n\n<sync-mailbox>" "휴지통으로"'
```

매크로 정의시 설명을 추가하면 도움말 화면에서 이것을 볼수 있다.

주의**:** 도움말 화면에서 매크로 설명은 화면 폭에 의해서 짤릴수 있다.

3.7. 색 사용하기

사용법: `color` 대상 전경색 배경색 [정규 표현식]

사용법: `color` index 전경색 배경색 패턴

사용법: `uncolor` index 패턴 [패턴 ...]

터미날이 색을 지원한다면 Mutt 상의 각 아이템에 색을 지정해 사용할수 있다. 색의 지정은 전경색과 배경색을 같이 지정해 주어야 한다.

다음과 같은 대상 들이 사용될수 있다:

- attachment (첨부물)
- body (메일 본문)
- bold (메일 본문에서 볼드 문자 하이라이팅)
- error (Mutt의 오류 메세지)
- header (메일 헤더)
- hdrdefault (페이저에서의 기본 헤더 색)
- index (메일 인덱스)
- indicator (메뉴에서 사용되는 화살표 또는 반전 막대)
- markers (페이저에서 “+” 로 시작되는 줄)
- message (메일의 정보)
- normal
- quoted (인용문, quote-regexp변수로 인용문을 결정할수 있다.)
- quoted1, quoted2, ..., quoted**N** (인용문의 단계)
- search (검색 결과의 하이라이팅)
- signature (서명)
- status (메일함과 메일의 정보가 표시되는 상태창)
- tilde (페이저에서 “<732>” 문자로 채워진 빈줄)
- tree (쓰레드 정렬시)
- underline (메일 본문의 밑줄)

전경색 과 배경색 에는 다음 값이 올수 있다:

- white
- black
- green

- magenta
- blue
- cyan
- yellow
- red
- default
- color**x**

전경색 은 선택적으로 `bright` 문자를 덧붙이면 밝거나 굵은 문자를 표시하게 할수 있다 (예, `brightred`).

터미날이 지원한다면, 특별한 키워드인 **default** 를 사용해서 투명하게 설정할수도 있다. 역시 **brightdefault** 도 가능하다. 만약 Mutt 컴파일시 **S-Lang** 라이브러리와 링크되었다면, **COLORFGBG** 환경변수를 사용해서 터미날의 기본색을 지정해 주어야 한다. 다음 예를 보라.

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

주의**:** **S-Lang** 라이브러리의 경우 **white** 와 **yellow** 대신에 **lightgray** 와 **brown** 값을 사용해야 한다.

주의**:** uncolor 명령은 오직 인덱스 대상에만 적용된다.

또한 mutt는 **color0**, **color1**, … 형식의 키워드를 인식한다. **colorN-1** (**N**은 터미날이 지원하는 색을 표현하는 숫자).

만약 흑백 터미날을 사용한다면 “mono” 키워드를 사용해서 새로운 속성의 지정 가능하다:

사용법: `mono` **<**대상**> <**속성**>** [정규 표현식]

사용법: `mono` index 속성 패턴

사용법: `unmono` index 패턴 [패턴 ...]

속성의 값은 다음과 같다:

- none
- bold
- underline
- reverse
- standout

3.8. 원하지 않는 헤더 숨기기

사용법: `[un]ignore` **pattern** [**pattern** ...]

이메일에는 보통 네트워크상에서 거쳐가는 시스템들에 의해 여러가지 헤더들이 추가되는데, 이들 대부분이 표시되지 않아도 되는 것들이다. 이 명령은 어떤 헤더를 보여 줄것인지 또는 감출것인지 정의할때 사용한다.

헤더 필드 이름은 완벽하게 적어주지 않아도 된다. 예를 들어 “ignore content-” 는 “content-”로 시작하는 모든 헤더를 표시하지 않을 것이다.

다음 예와 같이 “unignore” 명령을 사용해서 일단 모두 표시하지 않는 것으로 설정한후에 원하는 것들만 다시 지정해주는 방법도 있다.

예:

```
ignore *
unignore from date subject to cc
unignore organization organisation x-mailer: x-newsreader: x-mailing-list:
unignore posted-to:
```

3.9. 메일링 리스트

사용법: `[un]lists` 주소 [주소 ...]

사용법: `[un]subscribe` 주소 [주소 ...]

mutt은 몇가지 메일링 리스트와 관련된 기능을 가지고 있으며, 이것을 사용하기 위해서는 메일링 리스트의 주소와 가입 여부를 명시해 주어야 한다.

이것이 끝나면 list-reply (메일링 리스트로 답장) 기능을 사용할수 있다. 추가적으로 가입된 메일링 리스트로 메일을 보낼때에는 mutt이 Followup-To 헤더를 자동으로 추가할 것이다 (followup-to 참고)

“lists”를 사용해서 "알고 있는" 메일링 리스트의 주소를 “subscribe”에는 "가입된" 메일링 리스트의 주소를 적어준다.

“unlists” 명령을 사용하면 "알고 있는/가입된" 메일링 리스트의 설정을 해제할수 있다.

가입하지 않았지만 "알고 있는" 메일링 리스트라면 “unsubscribe” 명령을 사용할수 있다.

3.10. 다중 스풀 (spool) 메일함 사용

사용법: `mbox-hook` [!]패턴 메일함

여러개의 스풀 메일함을 사용할 경우 읽은 메일을 해당 메일함으로 옮길때 사용한다. 패턴은 정규 표현식으로 스풀 메일함을 지정하고 메일함 은 읽은 메일이 저장될 곳을 지정해 주면 된다.

다른 **hook** 명령들과는 다르게, 첫번째 패턴만이 사용된다. (즉, 여러개의 메일함으로 동시에 저장되는것은 불가능하다.)

3.11. 수신 메일함 설정

사용법: `mailboxes` [!]파일명 [파일명 ...]

어떤 메일함에 새로운 메일이 도착하며 그것을 감시할 것인가를 설정한다. 기본적으로 주 메뉴의 상태창에 이 메일함(들)에 새 메일이 도착했음을 표시해준다.

메일함을 바꿀때 **space**키를 누르면 새 메일이 도착한 메일함으로 이동한다.

디렉토리 브라우져에서 TAB키를 누르면 이 메일함들을 보여주며, 새 메일이 있는지 확인할 수 있다. 또한, mutt이 실행될때 `-y` 옵션을 붙이면 자동적으로 이 모드로 시작된다.

주의**:** 새 메일의 인식은 마지막 수정 시간과 마지막 접근 시간을 비교함으로 이루어 진다. `biff` 나 `frm` 같이 일정 간격으로 메일함에 접근 하는 프로그램들이 접근 시간을 제대로 reset 해주지 못하면 mutt은 새 메일을 인식하지 못할것이다. 또한 백업 도구들도 공통적으로 파일 접근 시간을 바꾸어 놓는다.

주의**:** "=" 나 "!" 같은 shortcuts이 사용되는 메일함을 설정할 때에는 관련된 변수(folder, spoolfile)가 먼저 선언되어야 한다.

3.12. 메일 헤더 정의

사용법: `my_hdr` **string**

사용법: `unmy_hdr` **field** [**field** ...]

"my_hdr" 명령은 자신만의 메일 헤더를 정의 할수 있게 해준다.

예를 들어, "Organization:" 헤더를 보내는 메일에 넣으려면 다음을 `.muttrc` 파일에 넣어준다.

```
my_hdr Organization: Eunjea\'s Secret Penguin Labs
```

주의**:** 키워드와 콜론 (":") 사이에는 공백 문자가 올수 없다 (이것은 RFC822에 준하며, mutt는 이것을 지킨다).

3.13. 메일 헤더 정렬

사용법: `hdr_order` **header1 header2 header3**

메일을 읽을때 보여지는 메일 헤더의 내용을 원하는 데로 정렬할수 있다.

"unhdr_order *" 는 이미 정의되어 있는 헤더 정렬 명령을 해제할때 사용한다.

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

3.14. 저장시 기본 파일 이름

사용법: `save-hook` [!]패턴 파일명

메일을 저장 할때 사용되는 기본 파일 이름을 지정한다.

파일명은 **From:** 또는 **to:**의 주소가 패턴(정규 표현식)과 일치할때 사용된다.

패턴의 정확한 사용법은 pattern-hook을 참고하라.

예:

```
save-hook me@(turing\\.)?cs\\.hmc\\.edu$ +elkins
save-hook aol\\.com$ +spam
```

fcc-save-hook 참고.

3.15. 기본 Fcc: 메일함 지정

사용법: `fcc-hook` [!]패턴**(regexp)** 메일함

이 명령은 보내는 메일을 record 변수에 지정된 메일함외에 다른 곳에 저장할때 사용한다.

mutt은 보내는 메일의 수신자중 패턴 에 일치하는 것이 있으면 여기에 지정된 메일함으로 저장하고 일치하지 않는 경우 record에 지정된 메일함에 저장한다.

pattern-hook에서 자세한 패턴 사용법을 참고하라.

예: `fcc-hook kldp.org$ +kldp`

위의 예는 메일의 수신자가 kldp.org 도메인을 가지고 있는 경우 '+kldp' 메일함에 저장되도록 한것이다.

fcc-save-hook 참조.

3.16. 기본 파일 이름과 Fcc: 메일함 지정

사용법: `fcc-save-hook` [!]패턴 메일함

이 명령 하나로 fcc-hook 와 save-hook을 대신할수 있다.

3.17. 수신자에 따라 다른 설정 적용

사용법: `send-hook` [!]패턴 명령어

이 명령은 메일의 수신자에 따라 특정 명령을 적용하도록 해준다. 패턴은 정규 표현식으로 수신자의 주소가 이것에 일치하면 **command** 를 실행하게 된다. 만약 여러번 일치하게 되면 설정파일에 적혀 있는 순서대로 명령을 차례대로 실행한다.

pattern-hook에서 자세한 패턴 사용법을 참고하라.

예: `send-hook mutt "set mime_forward signature=""`

전형적인 사용 예로는 수신자에 따라 attribution, signature. locale등을 다르게 지정해 줄수 있다.

주의**:** send-hook 명령은 오직 한번만 실행된다. 즉, 메일 작성후에 수신자를 변경하거나, 메일을 수정한 후에는 적용되지 않는다.

3.18. 메일의 형식에 따라 설정 바꾸기

사용법: `message-hook` [!]패턴 명령어

메일을 읽거나 포맷팅후에 해당 메일의 정보에 따라 명령을 줄수 있다. 메일의 내용이 패턴에 일치하면 명령어 를 실행 시킨다.

pattern-hook에서 자세한 패턴 사용법을 참고하라.

예:

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

3.19. 수신자의 PGP 키

사용법: `pgp-hook` 패턴 **keyid**

pgp-hook 명령은 메일을 PGP 암호화 하여 보낼때 패턴에 따라 특정키가 사용되도록 설정할 수 있다.

3.20. 키보드 버퍼에 일련의 키 입력 기억 시키기

사용법: `push` 문자

특정 문자열을 키보드 버퍼에 기억시킬때 사용한다. 문자는 제어문자와 키 이름, 기능의 이름등이 사용될수 있다. 이것은 mutt이 실행될때나 특정 메일함에 들어갈때 사용될수 있을것이다.

3.21. 명령 실행

사용법: `exec` 기능 [기능 ...]

기능을 실행할때 쓰인다. 기능들의 목록은 기능(함수)을 참고하라. “exec 기능” 과 “push <기능>” 는 같은 역할을 한다.

3.22. 메일 스코어링

사용법: `score` 패턴 점수

사용법: `unscore` 패턴 [패턴 ...]

메일 스코어링 명령은 패턴 (패턴) 과 일치하는 메일에 점수를 준다 (주의: 다음 패턴들은 사용될수 없다. `<732>b`, `<732>B`, `<732>h`).

점수 는 음/양수가 될수 있으며, 메일의 총 점수는 각 `score` 명령의 패턴에 일치하는 만큼 더해진다. 하지만, 점수에 = 부호를 사용하여 반복되어 점수가 더해지는 것을 방지 할수도 있다.

`unscore` 명령은 설정된 `score`을 해제할때 사용한다.

3.23. 변수값의 설정

사용법: `set` [no|inv]**variable**[=**value**] [**variable** ...]

사용법: `toggle` **variable** [**variable** ...]

사용법: `unset` **variable** [**variable** ...]

사용법: `reset` **variable** [**variable** ...]

이 명령은 설정 변수의 값을 설정한다. 기본적으로 다음 네가지 형태의 값이 사용될수 있다:

1. **boolean** : **set** (yes) 또는 **unset** (no).

2. 숫자

3. 문자 : 공백이나 TAB을 포함하는 경우 인용 부호로 감싸주어야 하며, **\n** 는 새줄, **\t**는 TAB을 표현할때 사용한다.

4. **quadoption** : 은 네가지 값이 올수 있음을 뜻하며 **yes** 나 **no** 일 경우 해당 기능을 물어보지 않고 실행/취소 하며, **ask-yes** 와 **ask-no** 는 mutt이 해당 기능을 실행하기 전에 물어보는 프롬프트를 보여주며 각각 yes와 no의 기본값을 설정한다.

각 변수에 “no” 를 붙여서 unset 할수도 있다. 예: `set noaskbcc`.

boolean 값에는 추가적으로 `inv`를 변수 이름앞에 붙여 사용할수 있는데, 이것은 이전 설정값의 반대값이 된다는 의미이다. 이것은 macro를 설정할때 유용하게 쓰일수 있다 (예: `set invsmart_wrap`).

`toggle` 명령을 사용하는 것과 변수 이름앞에 `inv` 를 붙이는 것은 같은 효과를 준다.

역시, `unset` 명령과 변수 이름앞에 `no` 를 붙이는 것도 같다.

index 메뉴에서는 명령어 실행 기능을 사용해서 변수앞에 ? 를 붙이면 현재 값을 볼수 있다:

```
set ?allow_8bit
```

물음표는 boolean 와 quadoption 값을 볼때만 필요하다.

`reset` 명령은 해당 변수값을 컴파일시의 기본값으로 돌려 놓는다 (기본값은 이 문서에 언급되어 있다). 만약 `set` 명령과 변수앞에 "&"를 덧붙이면 이것은 `reset` 과 같은 효과를 준다. 또한 `reset` 명령에 "all" 을 지정하면 모든 설정값이 초기화 된다.

3.24. 초기 명령어를 다른 파일에서 읽어오기

사용법: `source` 파일 이름

이 명령을 사용해서 다른 파일의 내용을 설정 파일로 불러 오는 것이 가능하다. 예를 들어 알리아스를 `<732>/.mail_aliases` 파일에 저장되도록 할수 있다.

파일 이름이 "<732>"로 시작하면 그것은 홈 디렉토리에 파일이 위치하고 있는 것을 뜻한다.

Tip: 만약 파일 이름이 | 로 끝난다면, 해당 파일을 실행하고 그 결과를 사용하게 된다 (예: tt/source /bin/myscript|/).

3.25. hook 지우기

사용법: `unhook` [* | **hook** 종류]

이 명령으로 이전에 설정된 hook 을 해제할수 있다. "*" 문자를 사용해서 모든 hook 설정을 해저하거나, 다음과 같이 특정 hook을 지정할수 있다. `unhook send-hook`.

4. 고급 사용법

4.1. 정규 표현식

패턴은 "POSIX 확장" 정규 표현식 (regexp)을 사용해야 한다. 여러분의 편리를 위해 정규 표현식에 대해 간단하게 소개한다.

패턴에 대문자가 포함되어 있지 않다면 대소문자를 가리지 않고 검색한다. 즉, 대문자가 하나라도 포함되어 있다면 case sensitive 검색을 할것이다. 검색 패턴에 "\"를 사용할때는 "\\"와 같이 사용해야 한다.

정규 표현식은 문자의 패턴을 기술한 것이며, 다양한 표현식을 조합한 수학 공식과도 비슷하게 만들어 진다.

검색 패턴에 공백을 포함하고 있는 경우나 구분을 위해 인용 부호 (",')가 사용된다 (muttrc-syntax를 참고). " 나 ' 를 검색하는 경우에는 백 슬레쉬(\)를 붙여준다.

점(".")은 newline을 제외한 모든 문자를 나타낸다. "^"은 줄의 시작을 나타내며, "$"는 반대로 줄의 끝을 나타낸다.

'[' 와 ']'는 집합의 시작과 끝을 나타낸다. '-'는 범위를 나타내는 문자이다. 이러한 문자들을 검색하려면 이미 말했듯이 문자 앞에 '\'를 붙여주면 된다 (예: [a-zA-Z0-9\-]).

집합이 "[^"로 시작되면 역집합을 표현한다. 즉, "[0-9]"는 0과 9사이의 숫자를 나타낸다면, "[^0-9]"는 0과 9사이의 숫자가 아닌 경우를 표현한다. 마찬 가지로 "[^abc]"는 a,b,c 가 아닌 모든 문자를 나타낸다.

미리 정의된 집합을 사용할수 있는데, 이 경우 "[:" 와 ":]"로 감싸주면 된다. 다음은 POSIX 표준에 의해 정의된 것이다.

[:alnum:]

알파벳과 숫자(alphanumeric)

[:alpha:]

알파벳 문자.

[:blank:]

공백이나 TAB.

[:cntrl:]

제어(control) 문자.

[:digit:]

숫자.

[:graph:]

프린트와 보기(visible)가 가능한 문자 (예를 들어 공백은 프린트가 가능하지만, visible한 문자는 아니며, "a"는 둘다 가능하다).

[:lower:]

알파벳 소문자.

[:print:]

프린트가 가능한 문자 (제어 문자를 제외한 문자)

[:punct:]

구두점 문자 (알파벳, 숫자, 제어, 공백문자가 아닌것).

[:space:]

공백 문자 (공백, 탭, 폼피드등)

[:upper:]

알파벳 대문자.

[:xdigit:]

16진수.

이 문자 정의를 사용할때에는 다음과 같이 브라켓으로 감싸줘야 한다 (예: **[[:digit:]]** 와 **[0-9]**는 같은 의미이다).

다음은 반복되는 단어에 사용되는 표현들이다:

?

바로 앞의 문자가 없거나 하나 있는 것을 표현함.

*

0개 이상의 문자.

+

바로 앞의 문자가 한개 이상 임을 나타냄. *와 비슷하게 동작하나 하나 이상을 표현함.

{n}

n번 반복되는 것.

{n,}

n번 이상 반복.

{,m}

최대 **m**번 반복.

{n,m}

n번 이상, **m** 번 이하 반복되는 것을 표현.

두개의 정규 표현식은 연결될수 있으며, "|"로 분리하면 어느쪽이나 일치하는 것을 의미한다.

참고**:** 만약 mutt를 GNU **rx** 패키지와 컴파일 했다면 다음 표현들을 정규 표현식에 사용할수 있다:

\\y

문자의 시작이나 끝에 공백문자가 있는 경우.

\\B

문자열 중간에 공백이 있는 경우.

\\<

문자의 앞에 공백이 있는 경우.

\\>

문자의 뒤에 공백이 있는 경우.

\\w

일반적인 문자 (알파벳,숫자 또는 밑줄).

\\W

일반적인 문자가 아닌것을 표현.

\\‘

버퍼(문자)의 앞에 공백이 있는 경우.

\\’

버퍼(문자)의 뒤에 공백이 있는 경우.

이 표현들은 POSIX에 의한 정의가 아니기 때문에 일반적인 시스템의 라이브러리에서는 지원하지 않을수도 있다.

4.2. 패턴

mutt에서 사용되는 많은 명령어에 다음과 같은 패턴을 사용하여 메세지를 선택할수 있다:

```
~A                  모든 메일
~b EXPR             EXPR 이 본문에 포함한 메일
~B EXPR             EXPR 이 포함된 메일
~c USER             USER 에게 cc(carbon-copy)된 메일
~C EXPR             to: 또는 cc: 이 EXPR 에 일치하는 메일
~D                  삭제된(삭제하기 위해 표시된) 메일
~d [MIN]-[MAX]      보낸 시간("date-sent")의 범위에 속하는 메일
~E                  만기된 메일
~e EXPR             "Sender" 헤더에 EXPR이 일치하는 메일
~F                  플래그가 붙은 메일
~f USER             USER 로 부터 온 메일
~g                  PGP 사인된 메일
~G                  PGP 암호화된 메일
~h EXPR             헤더에 EXPR 이 일치되는 메일
~k                  PGP 키를 포함하고 있는 메일
~i ID               "Message-ID" 가 ID와 일치하는 메일
~L EXPR             EXPR로 부터 오거나 EXPR로 송신된 메일
~l                  알고 있는 메일링 리스트의 메일
~m [MIN]-[MAX]      메일의 범위가 MIN to MAX 인 메일 *)
```

```
~n [MIN]-[MAX]   메일의 점수가 MIN 에서 MAX 인 메일 *)
~N               새 메일
~O               오래된 메일
~p               수신자가 자신인 메일 ($alternates 변수 참고)
~P               당신이 보낸 메일 ($alternates 변수 참고)
~Q               답장을 보낸 메일
~R               읽은 메일
~r [MIN]-[MAX]   받은 시간 ("date-received") 의 범위에 속하는 메일
~S               superseded messages
~s SUBJECT       제목 ("Subject")에 SUBJECT를 포함하고 있는 메일
~T               태그된 메일
~t USER          USER에게 발송된 메일
~U               읽지 않은 메일
~v               접혀있는 글타래의 메일
~x EXPR          'References' 헤더에 EXPR이 일치하는 메일
~y EXPR          'X-Label' 헤더에 EXPR이 일차하는 메일
~z [MIN]-[MAX]   메일의 크기가 이 범위에 속하는 메일 *)
```

EXPR, USER, ID, SUBJECT 에는 정규 표현식가 사용된다.

*) `<[MAX]`, `>[MIN]`, `[MIN]-`, `-[MAX]`와 같은 표현도 가능하다.

4.2.1. 패턴 수식어

주소의 패턴 (c,C,p,P,t)과 함께 정규 표현식을 사용하여 제한적인 의미를 만들수 있다. 예를 들어 다음의 예는 수신자의 이메일 주소가 ".de"로 끝나는 모든 이메일을 의미한다.

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

4.2.2. 패턴의 연결 (복합 검색)

하나 이상의 조건 검색은 다음과 같이 한다:

```
~t kldp ~f eunjea
```

위의 예는 수신자 주소중 "kldp"를 포함하고 있는 메일중에 "eunjea"가 "From"헤더와 일치하는 메일을 찾는다.

mutt는 또한 다음과 같은 연산자를 사용해서 복합 검색을 할수 있다:

- ! -- NOT 연산자
- | -- OR 연산자
- () -- 그룹 연산자

다음 예는 복합 검색의 예로, “eunjea”로 부터 온 메일중에 “To”와 “Cc” 헤더에 ‘mutt” 문자가 포함되지 않은 메일을 찾는다.

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

다음 예는 정규 표현식과 공백을 포함한 예로, “Jim +Somebody” 또는 “Ed +SomeoneElse”로 부터 온 메일중에 제목이 “^Junk +From +Me$”와 일치하는 메일을 찾는다:

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

4.2.3. 날짜 검색

기간 검색. 날짜는 반듯이 DD/MM/YY 형태여야 한다. 다음은 올바른 날짜 범위 검색의 예이다:

```
메일 검색: ~d 20/1/95-31/10
```

만약 시작 날짜(언제 부터)를 생략하고 “-DD/MM/YY” 형식으로 검색하면 그 날짜 이전의 모든 날짜가 선택된다. 또한 두번째 날짜(언제 까지)를 생략하고 “DD/MM/YY-” 형식으로 검색하면 그 날짜 이후를 검색한다. 해당 날짜의 메일만을 검색할때는 검색하고자 하는 날짜만 적어주면 된다.

전후로 찾기. (+ 나 -)와 숫자, 그리고 다음 문자를 사용할수 있다:

```
y 년도
m 월
w 주
d 날짜
```

특별한 경우에 “*”를 사용할수도 있는데, 이것은 +,- 를 동시에 준것과 같다.

예: 2001년 1월 15일을 기준으로 2주 전후의 메일들을 찾는다:

```
메일 검색: ~d 15/1/2001*2w
```

상대 조건 검색. 현재 날짜를 기준으로 검색하며, 다음과 같이 사용한다:

- **>offset** (**offset** 이전의 메일)
- **<offset** (**offset** 이후의 메일)
- **=offset** (**offset** 과 정확히 일치하는 메일)

offset 에는 숫자와 다음 문자를 사용한다:

```
y       years
```

```
m       months
w       weeks
d       days
```

예: 1개월 이내의 메일 찾기

```
메일 검색: ~d <1m
```

주의**:** 날짜를 사용한 검색은 **local** 시간대를 사용하며, index-format를 사용해서 포맷을 바꾸지 않는 이상 인덱스에 나타나는 값과는 차이가 있다.

4.3. 태그 사용

가끔 여러개의 메일을 동시에 저장,삭제 하거나 특정 제목을 가진 메일을 동시에 처리해야 하는등의 경우가 있다. 이때에는 검색 패턴과 일치하는 메일들에 태그를 사용하면 되는데, 이때에는 tag-pattern 기능 (기본: shift-T)를 사용한다 (검색 패턴 참고). 각각의 메일에 태그하려면 “t”를 사용하면 된다.

일단 태그된 메일에 동시에 어떠한 작업을 하려면 “tag-prefix” (기본: “;” (세미콜론))을 누른 후에 하면 된다. 만약 auto-tag 변수가 설정되어 있는 경우에는 “tag-prefix”없이 태그된 메일이 있는 경우 자동으로 적용된다.

4.4. Hook 사용

hook은 EMACS 편집기에서 차용한 아이디어로 일정한 명령들을 실행하고 난후에 작업을 시작하는 것이다. 예를 들어 메일을 읽을때 또는 보낼때마다 서로 다른 명령/설정을 사용할수 있게 해준다. mutt의 세계에서 **hook**은 정규 표현식 또는 패턴과 설정 옵션/명령들로 이루어진다. 자세한 것은 다음을 보라:

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

주의**:** hook으로 인해 한번 변경된 설정은 mutt의 세션이 종료되기 전까지 남아 있는다. 그러므로, 일반적으로 사용되는 설정값의 hook을 같이 설정하거나 이전 값을 해제시키는 방법을 사용한다. 다음은 send-hook 과 my_hdr 의 사용예이다.

```
send-hook . 'unmy_hdr From:'
send-hook ~Cb@b.b my_hdr from: c@c.c
```

4.4.1. hook 명령에서 패턴 사용

`send-hook`, `save-hook`, `fcc-hook`, `message-hook` 등의 메일과 관련된 hook은 다른것들과 약간 다르게 작동한다. 다른 hook 에는 정규 표현식만으로 충분하지만, 메일과 관련된 이 hook들은 좀더 세밀한 조절을 필요로 하기 때문에 패턴이 사용된다.

예를 들어, 수신자의 주소에 따라 From: 헤더를 바꾸려면 다음과 같이 할수 있다:

```
send-hook '~t ^eunjea@kldp\.org$' 'my_hdr From: Mutt User <user@host>'
```

즉, 이 명령은 **eunjea@kldp.org**에 메일을 보낼때 실행될 것이다.

마찬가지로 정규 표현식을 사용한 좀더 복잡한 조건을 만들수도 있다. default-hook을 참고하라.

4.5. 외부 프로그램을 사용한 메일 주소 쿼리

mutt는 외부 스크립트(wrapper)를 사용해서 LDAP, ph/qi, bbdb, NIS 같은 데이터 베이스와의 연결이 가능하다. query-command 변수를 사용해서 외부 스크립트를 정의한다:

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

외부 스크립트는 질의를 커맨드 라인에서 받아들여야 하며, 한줄이상의 결과값을 되돌려 주도록 한다. 각 라인은 주소와 이름 그리고 추가적인 정보를 tab으로 구분하여 출력되도록 한다. 일치하는 것이 없다면 0이 아닌 종료 코드와 한줄의 오류 메세지를 출력하게 한다.

다음과 같은 여러줄의 결과값을 사용할수 있다:

```
Searching database ... 20 entries ... 3 matching:
me@cs.hmc.edu           Michael Elkins  mutt dude
blong@fiction.net       Brandon Long    mutt and more
roessler@guug.de        Thomas Roessler mutt pgp
```

mutt에서는 두가지 방식으로 쿼리 기능에 접근할수 있다. 첫번째는 인덱스 메뉴에서의 쿼리 기능 (기본: Q)으로 이때는 먼저 질의를 직접 입력하고 그 결과가 있으면 쿼리 메뉴가 표시된다. 이 메뉴에서는 결과로부터 알리아스를 만들거나 메일을 보낼수 있다. 또는 여러개의 주소를 태그해서 한번에 메일 보내기나 새로운 질의 하기 또는 현재 결과에 새로운 결과값을 추가하는 것등을 할수 있다.

또 다른 방법으로는 주소를 완성하기 위해 사용될수 있다. 이것은 알리아스의 자동 채우기 기능과 비슷한데, 주소를 입력하는 모든 프롬프트에서 complete-query 기능 (기본: ^T)의 사용

이 가능하다. 이것은 현재 당신이 입력 해놓은 것을 바탕으로 질의를 한후에 그 결과로 주소를 완성하게 해주는 것이다. 만약 결과가 여러개라면 쿼리 메뉴를 보여줄 것이다.

4.6. 메일함의 형식

mutt는 다음 네가지의 서로 다른 메일함 형식을 지원한다: mbox, MMDF, MH, Maildir.

메일함 형식은 자동으로 인식하므로 서로 다른 형태의 메일함을 동시에 사용하는 것도 가능하다. mutt에서 메일함을 새로 생성할때에는 mbox-type 변수의 값이 사용된다.

mbox. 이것은 UNIX에서 가장 폭넓게 사용되는 형식이다. 모든 메일이 하나의 파일에 저장된다. 각 메일은 다음과 줄을 가지고 있으며:

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

이것은 메일의 시작을 나타낸다.

MMDF. 이것은 **mbox**의 변형된 포맷이다. 각 메세지는 "^A^A^A^A" (control-A x 4)로 구분된다.

MH. 이 메일함 포맷은 디렉토리와 메일마다 서로 다른 파일을 가진다. 주의**:** mutt은 `.mh_sequences` 또는 `.xmhcache` 를 사용해서 메일함의 형식을 인식한다.

Maildir. Qmail MTA 에 의해 사용되는 새로운 메일함 형식이다. **MH**와 비슷하지만, 세개의 하부 디렉토리를 가지고 있다: **tmp**, **new**, **cur**. 메일들의 파일 이름은 독특하게 고안되어 NFS 상에 위치한 메일함을 두개의 프로그램이 동시에 접근할수 있다. 즉, 파일 잠금 (file locking)이 필요 없다.

4.7. 메일함 단축키

메일함을 표현하는 단축키가 정의되어 있으며, 이것들은 파일이나 메일함을 묻는 프롬프트 등 어디에서나 사용할수 있다.

- ! -- spoolfile (메일이 들어오는) 메일함
- > -- mbox 파일
- < -- record (읽은 파일이 저장되는) 파일
- - 또는 !! -- 마지막으로 접근한 파일
- <732> -- 홈 디렉토리
- = 또는 + -- folder 디렉토리
- @**alias** -- 알리아스의 주소에 의한 save-hook

4.8. 메일링 리스트 다루기

mutt은 많은 양의 메일을 쉽게 다루기 위한 옵션을 가지고 있다. 그 첫번째는 mutt으로 하여금 어떤 주소가 메일링 리스트의 것인지 알게 하는 것이다 (기술적으로 이것은 반듯시 메일링 리스트여야 하는 것은 아니지만, 가장 보편적으로 사용된다). 이것은 muttrc 파일에 lists 를 사용하면 된다.

이제 mutt는 **index** 모드에서 메일링 리스트로부터 온 메일에는 일반 메일과 구분이 되도록 "L"을 메일의 상태 플래그에 표시할 것이다. 두번째는 메일링 리스트의 메일에 답장을 보낼때 사용할수 있는 "list-reply"기능 (기본 키:L)이다. 이것은 답장을 메일의 송신자가 아닌 가입된 메일링 리스트로 보낸다.

또한, mutt는 `Mail-Followup-To` 헤더를 지원한다. 이 헤더는 수신자가 가입된 메일링 리스트이며, followup-to 옵션이 켜져 있을때 첨가되며, group-replie나 list-replie(followups)시에 원래 수신자에게만 메일이 보내지도록 한다. 즉, 이것은 당신에게 메일이 보내지는 것을 방지하는데, 왜냐하면 어차피 메일링 리스트로부터 메일이 올것이기 때문이다.

반대로, mutt에서 `Mail-Followup-To` 헤더를 가진 메일에 group-reply나 list-reply를 할때는 honor-followup-to옵션에 의존하게 된다. 이 경우 list-reply는 답장은 무조건 메일링 리스트로 보내지게 된다.

참고: 헤더를 수정할수 있도록 설정되었을 경우 `Mail-Followup-To`헤더를 직접 추가할수도 있다. 그외의 경우에는 mutt이 자동으로 메일이 보내질때 추가하게 된다.

어떤 메일링 리스트 메일은 답장이 글쓴이가 아니라 메일링 리스트로 보내지도록 "Reply-To" 헤더가 사용되는 경우가 있다. 이것은 때로는 직접 글쓴이에게 개인적으로 답장을 보내려 할때 문제가 될수 있는데, 대부분의 메일 클라이언트가 자동으로 "Reply-To" 헤더를 사용하기 때문이다. 이 경우 mutt에서는 reply-to 옵션을 사용하여 이 헤더를 사용할 것인지를 결정할수 있게 해준다.

"X-Label:" 헤더를 사용해서 메일링 리스트나 제목등을 인식하도록 하는것도 가능하다. 이때에는 index-format 변수의 "%y"와 "%Y"를 사용한다. "X-Label:" 헤더는 표준 메일 헤더가 아니지만, procmail 같은 메일 필터링 프로그램을 사용하여 쉽게 추가할수 있다.

마지막으로 mutt는 메일들의 threads sort가 가능하다. 글타래는 같은 제목을 가진 메일들의 그룹으로 일반적으로 트리 구조로 표현된다. 당신이 유즈넷 클라이언트를 사용해 보았다면 쉽게 이해할 것이다. 이 기능은 매우 많은 메일이 오가는 메일링 리스트의 관리를 쉽게 해준다.

4.9. 배달 상태 알림 (Delivery Status Notification (DSN)) 지원

이 기능은 RFC1894에 정의된것으로, mutt에서는 이 기능을 사용하기 위해 두가지 변수를 지원한다. dsn-notify는 몇가지 결과 (실패한 메일, 전송된 메일등)를 수신자에게 요청하는데 사용되며, dsn-return 요청은 얼마만큼의 메세지(헤더 또는 메일 전체)가 되돌려 지는지를 결정한다.

4.10. POP3 지원 (선택)

만약 Mutt 가 POP3 지원되도록 컴파일 (**configure** 스크립트를 **--enable-pop** 옵션을 주고 컴파일 한 경우) 했을때에는 POP3 서버를 통해 원격으로 메일을 다룰수 있다.

원격 POP3 메일함은 다음과 같이 표현된다: `pop://popserver/`.

다음과 같이 특정 포트를 지정할수도 있다: `pop://popserver:port/`.

각 폴더마다 서로 다른 유저명을 사용할수도 있다: `pop://username@popserver[:port]/`.

원격 메일함의 새 메일을 확인하는 간격은 pop-checkinterval로 조정할수 있으며 기본값은 60초이다.

만약 SSL 을 지원하도록 (**configure --with-ssl**) 컴파일 되었다면 POP3 서버와의 연결을 암호화 할수 있다. 이것은 당연히 서버에서도 SSL 암호화 연결을 지원해야 한다. POP3/SSL 연결을 위해서는 다음과 같은 형식을 사용한다: `pops://[username@]popserver[:port]/`.

POP3 메일의 또 따른 접근은 **fetch-mail** 기능이다. 이것은 pop-host로 접속하여 메일을 spoolfile 메일함으로 가져온다.

주의**:** 만약 메일을 로컬 메일함으로 가져오는 기능만을 사용할 것이라면 이것만을 위해 특별히 설계된 fetchmail (http://www.ccil.org/~esr/fetchmail)같은 프로그램의 사용을 고려해라.

4.11. IMAP 지원 (선택)

IMAP 지원 (**configure --enable-imap**)이 되도록 컴파일된 경우 원격 IMAP 서버의 메일함 사용이 가능하다.

원격 imap서버의 폴더는 `imap://imapserver/INBOX`와 같은 형식을 사용함으로 접근 가능하다. `INBOX`은 특별한 이름으로 스풀 메일함을 의미한다. 만약 다른 메일 폴더에 접근하려면 `imap://imapserver/path/to/folder`와 같은 형식을 사용하면 된다.

다음과 같이 사용할 포트를 설정할수도 있다: `imap://imapserver:port/INBOX`.

또한 각 imap 서버에 따라 다른 유저명을 사용하는 것도 가능하다:
`imap://username@imapserver[:port]/INBOX`.

mutt 컴파일시 SSL 지원되도록 (**configure --with-ssl**) 컴파일 되었다면 IMAP 서버와의 연결을 암호화 할수 있다. 이것은 당연히 서버에서도 SSL을 지원해 주어야 하며, 다음과 같은 방법으로 IMAP/SSL 폴더를 지정한다:
`imaps://[username@]imapserver[:port]/path/to/folder`

다음과 같이 Pine 호환 설정도 가능하다:
`{[username@]imapserver[:port][/ssl]}path/to/folder`

IMAP 서버의 폴더를 브라우징 할때 **toggle-subscribed** 명령을 사용할수 있다 (imap-list-subscribed변수 참조).

IMAP 서버를 통한 새 메일 확인은 지연현상이 많이 일어날수 있으므로, mail-check timeout 변수들을 신중히 설정해야 할 것이다. 다음은 내가 사용하고 있는 값들이다.

```
set mail_check=90
```

```
set timeout=15
```

이 값은 내가 사용하는 느린 모뎀라인에서 좋은 결과를 보인다.

4.11.1. 폴더 브라우저

mutt는 로컬 파일 브라우저 처럼 IMAP 서버상의 메일함에 접근하는 것을 지원하며, 다음과 같은 차이점이 있다:

- 파일 퍼미션 대신에 "IMAP"이 표시되며, "+" 는 해당 항목이 메일과 하부 디렉토리를 포함하고 있다는 것을 알려준다. 보통 cyrus 같은 서버들의 폴더가 메일과 하부 디렉토리를 가지고 있다.
- 메일과 하부 디렉토리를 포함하고 있는 경우 선택 키 (기본: `enter`)는 하부 디렉토리로 이동하는데 쓰이며, 해당 폴더에서 메일을 읽으려 한다면 `view-file` (기본: `space`)를 사용해야 한다.
- 메일함의 삭제는 `delete-mailbox` (기본: `d`)를 사용할수 있으며, 메일함의 `subscribe`와 `unsubscribe`는 각각 `s` 와 `u`키를 사용한다.

4.11.2. 인증

mutt는 네가지의 IMAP 서버 인증 방법을 지원한다: SASL, GSSAPI, CRAM-MD5, LOGIN (Grant Edwards가 exchange 유저들을 위한 NTLM 인증 지원을 위한 패치를 만들었는데, 아직 주 소스에는 포함되지 않았다). 또한, 가상-프로토콜인 ANONYMOUS 를 지원한다. 이것은 공개 IMAP 서버에 계정없이 로그인할때 사용된다. ANONYMOUS를 사용하기 위해서는 유저 이름을 공백이나 "anonymous"로 설정한다.

SASL은 다른 어떠한 인증 방식보다도 안전한 프로토콜이며, 모든 세션이 네크워크 상에서 암호화 된다 (DIGEST-MD5, GSSAPI 도 가능). 그러므로, 이것은 서버가 지원한다면 가장 권장되는 인증 방법이다. SASL의 사용을 위해서는 Cyrus SASL 라이브러리가 설치되어 있어야 하며, mutt 컴파일시 **--with-sasl**옵션을 사용해야 한다.

mutt은 모든 인증 방식을 다음과 같은 순서로 시도할 것이다: SASL, ANONYMOUS, GSSAPI, CRAM-MD5, LOGIN.

인증을 제어하기 위한 변수들:

- imap-user - 접속하려는 IMAP 서버의 계정 사용자 이름. 이 값은 메일함의 패스(`{user@host}`)에 의해 무시된다.
- imap-pass - 모든 인증방법에 사용될 암호.
- imap-authenticators - IMAP 인증 방법을 콜론으로 구분하여 나열한다.

4.12. 다중 IMAP/POP 지원 (선택)

만약 여러개의 IMAP 과(또는) POP 서버를 동시에 사용하고 있다면 account-hook 명령을 사용할수 있다. 이 명령은 folder-hook 과 비슷하게 작동하지만, 원격 메일함에 접근할때 사용된다.

사용예:

```
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"'
```

4.13. URL을 웹 브라우져로 연결 하기 (외부)

메일의 내용에 URL이 포함되어 있는 경우에 urlview 라는 외부 프로그램 (ftp://ftp.guug.de/pub/mutt/contrib/에서 다운로드 가능)을 사용하면, 이 URL들이 메뉴로 표시되어 웹 브라우져로 편하게 열어 볼수 있다.

mutt에서의 설정 예:

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

ctrl-b 를 누르면 해당 메일에 포함된 URL의 리스트가 표시된다.

5. Mutt의 MIME 지원

Mutt에서 MIME 사용을 위해서는 두가지 설정 파일이 필요한데, 각각 IANA MIME 형식을 매핑해 놓은 `mime.types` 와 각 MIME 형식에 대응하는 명령들을 지정하는 `mailcap` 이다.

5.1. Mutt에서 MIME 사용

Mutt에서는 pager, compose, 첨부물 메뉴에서 MIME을 처리 할수 있다.

5.1.1. MIME 메일 보기

메일 인덱스와 읽기(pager)에서는 메일을 일반 텍스트로 디코딩 한다. mutt이 기본적으로 지원하는 MIME 형식은 `text/plain, text/enriched, message/rfc822, message/news`이다. 추가적으로 여러 형태의 PGP/MIME 과 application/pgp 같은 여러 형태의 PGP MIME을 지원한다.

mutt은 첨부물의 형식을 다음과 같은 형식으로 표시한다:

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

Description 에는 첨부물에 대한 설명 또는 파일 이름이, Encoding 에는 7bit/8bit/quoted-printable/base64/binary중 하나가 표시된다.

만약 mutt이 MIME 형식을 처리하지 못하면 다음과 같이 표시한다:

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

5.1.2. 첨부물 메뉴

첨부물을 보기 위한 view-attachments 명령의 기본 키 설정은 'v' 이다. 첨부물 메뉴에서는 메일에 첨부된 첨부물의 목록을 보여주며, 저장, 프린트, pipe, 삭제, 보기등이 가능하다. 만약 여러개의 첨부물이 있다면 태그를 사용하여 모든 표시된 첨부물에 원하는 명령을 한번에 적용할수도 있다.

당신은 이 메뉴에서 답장을 보내는 것도 가능하며, 이때 현재(또는 태그한) 첨부물을 본문에 인용한다.

각 첨부물은 텍스트 형태로 보거나, mailcap 에 설정한 외부 프로그램을 사용하여 볼수 있다.

또한 일반적으로 메일과 관련된 기능들(예를 들어 resend-message, 답장, 포워딩)에 message/rfc822 형식을 적용할수 있다.

첨부물 메뉴에서 더 자세한 설명을 찾을수 있다.

5.1.3. 메일 작성 메뉴

compose 메뉴는 메일을 보내기 전에 볼수 있는 화면이다. 여기서는 수신자 목록, 제목, 본문 등을 수정할수 있으며, 보내려는 메일의 첨부물의 목록을 보여준다. 이 메뉴에서 각 첨부물에 대한 프린트, 복사, 필터, pipe, 수정, 작성, 다시보기, 이름 바꾸기등이 가능하다. 또한 첨부물의 정보나 인코딩, 설명의 수정도 가능하다.

첨부물은 다음과 같이 표시된다: - 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>

'-' (마이너스) 기호는 mutt이 메일을 보낸후 (또는 발송 연기, 취소)에 삭제함을 의미한다. 이것은 toggle-unlink (기본: u)로 바꿀수 있다. 다음 필드는 MIME 형식이며, 이것은 edit-type (기본: ^T)로 수정할수 있다. 다음은 첨부물의 인코딩 방식으로 edit-encoding (기본: ^E)로 수정 가능하다. 다음 필드는 첨부물의 크기로 kilobyte나 megabyte로 표시된다. 그 다음은 파일 이름으로 rename-file (기본: R)으로 수정 가능하다. 마지막 필드는 첨부물의 설명이며 수정은 edit-description (기본: d)로 한다.

5.2. `mime.types` 파일에 MIME 형식 설정

메일에 첨부물을 추가할때 mutt은 순서대로 `${HOME}/.mime.types`, `/usr/local/share/mutt/mime.types` 또느 `/etc/mime.types`를 찾게 된다.

mime.types 파일은 MIME 형식과 확장자로 기술한다:

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

예제 `mime.types` 파일은 Mutt 배포본에 포함되어 있으며 여기에는 여러분이 사용하는 대부분의 MIME 형식이 들어있다.

만약 첨부되는 파일의 형식을 인식하지 못하면 mutt는 파일을 직접 살펴보고 바이너리 정보가 없다면 일반 텍스트로 인식하고 `text/plain`로 표시할 것이다. 만약 첨부물이 바이너리 정보를 포함한다면, `application/octet-stream`로 표시한다. mutt이 결정한 MIME 형식은 `edit-type` (기본: ^T)를 사용해 변경할수 있다.

5.3. `mailcap` 파일에 외부 프로그램 설정

mutt은 RFC 1524에 의한 일반적인 mailcap 파일 형식을 지원한다. 이 mailcap 파일은 Mutt이외에도 MIME을 지원하는 다른 프로그램들에 의해 사용된다 (예: Netscape, XMosaic, lynx, metamail).

mutt은 이 파일을 사용해서 내부적으로 처리하지 못하는 MIME 형식의 파일을 처리 하게되며, 다음과 같은 순서로 이 파일을 찾게된다.

```
${HOME}/.mailcap:/usr/local/share/mutt/mailcap:/etc/mailcap:/etc/mailcap:/usr/etc/mailcap:/u
```

`$HOME` 은 당신의 홈 디렉토리이다.

metamail 패키지에는 보통 mailcap 파일을 포함하고 있으며 `/usr/local/etc/mailcap`에 설치된다.

5.3.1. `mailcap` 파일의 기본

기본적은 형식으로 text/plain 메세지를 more 로 연결하려면:

```
text/plain; more
```

메세지를 파일로 보내기:

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

text/html 형식의 메세지를 lynx 로 보기:

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

이 경우, lynx는 stdin 으로 부터 직접 보기는 불가능하므로, 반듯이 %s 를 사용해야 한다.

주의**:** 몇몇 오래된 버전의 **lynx**는 **text/html** 형식의 파일을 볼때 **mailcal** 파일을 참고하는 버그가 있다**.** 결과적으로 **lynx**가 다시 **lynx**를 무한적으로 부르게 된다**.**

다음과 같이 lynx를 사용해서 text/html 형식을 text/plain으로 변환하는 방법도 있다:

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

text/html 파일은 lynx로 보고, 나머지 모든 text 형식은 more 를 사용한다면:

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

이것은 가장 간단한 형태의 mailcap 파일이다.

5.3.2. mailcap 의 안전한 사용

MIME 변수에 포함된 쉘 문자들로 인해 보안상 문제를 일으킬수도 있다.

mutt은 이 위험에서 벗어나기 위해 위험한 문자들을 인용문자로 감싸는 방법을 사용한다. mailcap-sanitize를 참조.

Mutt와 mailcap에 의해 실행되는 프로그램은 안전할수 있으나, mailcap을 사용하는 다른 프로그램들을 위해서 다음 예를 참고하라:

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

5.3.3. 향상된 mailcap 사용법

5.3.3.1. 옵션 필드

content-type 과 view 명령 필드에는 추가적인 옵션을 사용 하는데, 각 옵션들은 ';'로 분리해서 나열할수 있다. mutt는 다음과 같은 옵션들을 인식한다:

copiousoutput

이 플래그는 결과를 내장 페이저(또는 외부)에서 볼때 사용한다. 다음 예를 참고하라:

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

이것은 lynx로 하여금 text/html를 text/plain로 변환하여 출력한뒤 mutt의 기본 내장 페이저에서 표시하도록 할것이다.

needsterminal

첨부물의 MIME 자동보기시에 사용되는 옵션으로, 외부 프로그램의 실행이 종료된후 wait-key변수를 적용할 것인지를 결정하게 된다.

compose=<command>

지정된 MIMT 형식의 첨부물을 만들때 사용되는 명령어를 지정해 준다. 메일 작성(compose) 메뉴에서 사용된다.

composetyped=<command>

compose 옵션과 같은 용도로 사용되나, mutt은 표준 MIME 헤더를 기대할 것이다. 이것은 지정된 변수, 파일명, 설명들이 될수 있다. 메일 작성(compose) 메뉴에서 사용된다.

print=<command>

지정된 MIME 형식의 프린트시 사용되는 명령어. 첨부물(attachment), 메일 작성(compose) 메뉴에서 사용된다.

edit=<command>

지정된 MIME 형식을 편집할때 사용되는 명령어.

nametemplate=<template>

이 필드는 %s에 의해 나타내는 파일의 형식을 지정한다. 이것은 어떤 프로그램들은 파일의 확장자로 파일의 형식을 인식하기 때문에 사용된다. 예를 들어, lynx는 `.html`로 끝나는 파일명을 가진 파일만을 `text/html`로 표시한다. 따라서, mailcap 파일의 내용은 다음과 같다:

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

test=<command>

이 필드는 해당 명령어를 조건에 따라 실행 할것인지 먼저 테스트하는데 사용된다. 만약 결과값이 0이 아니라면 명령을 실행하지 않고 다음 적합한 목록을 찾게된다. 다음 예를 보라:

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

이 예제는 mutt이 먼저 RunningX를 실행하여 그 결과가 0 이면 X가 실행 중이므로, netscape을 실행하고 0이 아니라면 lynx를 사용할 것이다.

5.3.3.2. 검색 순서

mailcap 내용을 검색할때 mutt는 그 목적에 가장 적합한 것을 사용하게 된다. 예를 들어 `image/gif`를 프린트 하려 한다면, print 명령을 가진 mailcap 내용을 사용하게 된다:

```
image/*;        xv %s
```

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

Mutt은 `image/*`는 무시하고 `image/gif`를 print 명령으로 사용한다.

다음 예와 같이 MIME 자동보기와 첨부물 보기등을 위해 한가지 MIME 형식에 따라 몇가지 명령을 설정해 줄수 있다.

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

MIME 자동보기를 위해 mutt은 세번째 명령을 사용할 것이다. 이유는 copiousoutput 태그가 있기 때문이다. 대화형 보기시에는 첫번째와 두번째 설정을 시도할 것이다.

5.3.3.3. 명령 확장(Command Expansion)

mailcap 파일안의 명령어에 mutt는 다음과 같은 변수를 전달해 줄수 있다:

%s

해당 프로그램에 파일 이름을 전달할때 사용한다. 이 파일의 내용은 메일의 본문이다.

%t

%t는 메일 내용의 형식을 mailcap 정의의 첫번째 변수와 같이 표현한다 (예: `text/html`, `image/gif`).

%{<parameter>}

지정된 Content-type:의 변수를 나타낸다. 예를 들어 메일의 헤더가 다음과 같다면:

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

%{charset}는 iso-8859-1가 될것이다.

\%

이것은 %로 바뀔것이다.

mutt은 현재 RFC 1524에 명시되어 있는 %F 와 %n 키워드를 지원하지 않는데, 이것은 이 변수들이 이미 Mutt에서 내부적으로 지원하고 있는 multipart 메세지를 위한 것이기 때문이다.

5.3.4. 예제 **mailcap** 파일

기본적인 예:

```
# 나는 X를 항상 실행하고 있어요!
video/*;        xanim %s > /dev/null
image/*;        xv %s > /dev/null
```

```
# netscape가 항상 실행되고 있음
text/html;      netscape -remote 'openURL(%s)'
```

다음은 내가 쓰는 mailcap 파일로 몇가지 사용 예를 보여준다:

```
# 비디오 관련 파일을 xanim을 사용하여 본다. xanim가 실행될때의
# 메세지는 /dev/null 로 보낸다.
video/*;        xanim %s > /dev/null

# html은 실행중인 netscape으로 보냄.
text/html;      netscape -remote 'openURL(%s)'; test=RunningNetscape

# netscape 가 실행되고 있지 않다면 실행한다.
text/html;      netscape %s; test=RunningX

# X 가 실행되고 있지 않은 경우 lynx 를 사용
text/html;      lynx %s

# text/html 를 text/plain로 변환해서 보기
text/html;      lynx -dump %s; copiousoutput

# enriched.sh 스크립트로 text/enriched를 text/html로 변환
text/enriched;  enriched.sh ; copiousoutput

# 프린트시 enscript를 사용해서 컬럼을 나누기
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

# X가 실행중인 경우 xv로 이미지 파일 보기
image/*;xv %s; test=RunningX; \
        edit=xpaint %s

# netpbm 툴을 사용해서 이미지를 텍스트로 변환하기
image/*;  (anytopnm %s | pnmscale -xysize 80 46 | ppmtopgm | pgmtopbm |
pbmtoascii -1x2 ) 2>&1 ; copiousoutput

# 엑셀 파일을 NT 박스로 보내기
application/ms-excel; open.pl %s
```

5.4. MIME 자동 보기

이 기능을 사용하면 페이저에서 MIME 첨부물을 자동으로 볼수 있다.

이것이 작동하려면 mailcap 파일에서 사용되는 외부 프로그램에 `copiousoutput` 옵션을 사용해야 한다. 일반적으로 페이저에서 볼수 있는 것은 텍스트 형태로 변환된 것이다.

muttrc에 `auto_view` 명령을 사용하면 원하는 형식의 파일을 자동으로 볼수 있다.

예:

```
auto_view text/html text/enriched application/x-
gunzip application/postscript image/gif application/x-tar-gz
```

mailcap 파일에 각 MIME 형식에 따라 사용할 외부 프로그램을 설정해준다.

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

5.5. MIME Multipart/Alternative

mutt은 multipart/alternative 형태의 MIME을 표시를 위해 먼저 alternative_order 목록을 확인하게 된다. 다음은 alternative_order 목록의 예이다:

```
alternative_order text/enriched text/plain text application/postscript im-
age/*
```

다음, mutt는 MIME 자동보기의 정의된 형태를 확인하고, 실패하면 텍스트 형태를, 그리고 마지막으로 어떻게 다루어져야 하는지 알고 있는 형태를 시도하게 된다.

6. 참조

6.1. 실행 옵션

아무런 옵션을 주지 않고 `mutt`를 실행시키면 mutt는 스풀 메일함을 읽는다. 다음의 실행 옵션들로 다른 메일함을 읽도록 하거나 명령줄에서 새 메일을 보내는 것등이 가능하다.

```
-a      메일에 파일 첨부하기
-c      carbon-copy (Cc) 주소
-e      설정 파일을 읽은후 명령어 실행
-F      다른 설정 파일 읽기
```

```
-f      메일함 읽기
-h      도움말 보기
-H      헤더와 메일 본문을 포함 파일 읽기
-i      메일 작성시 포함될 파일
-n      시스템 Muttrc 파일 읽지 않기
-m      기본 메일함 유형
-p      발송 연기된 메일 부르기
-R      메일함을 읽기 전용으로 열기
-s      메일의 제목 (인용 부호로 감싸줄것)
-v      mutt의 버전과 컴파일시의 정의 보기
-x      mailx(1) 작성 모드 에뮬레이트 하기
-y      mailboxes 에 정의된 메일함들을 표시하기
-z      메일함에 새 메일이 없으면 종료
-Z      첫번째 메일함의 새 메일을 열거나, 없을 경우 종료
```

메일함의 메일 읽기

`mutt` [-nz] [-F **muttrc**] [-m 메일함의 종류] [-f 메일함]

새 메일 작성하기

`mutt` [-n] [-F **muttrc**] [-a 파일] [-c 주소] [-i 파일명] [-s 제목] 주소 [주소 ...]

또한, Mutt 는 준비된 메일을 보내는 “batch” 모드를 지원한다. 다음과 같이 간단하다. 예제를 보라,

```
mutt -s "숙제 #2" professor@bigschool.edu
< <732>/run2.dat
```

이 명령은 “professor@bigschool.edu”에게 "숙제 #2"라는 제목으로 메일의 내용은 <732>/run2.dat 파일이 사용되어 보내진다.

6.2. 명령어

다음의 명령어들이 사용된다.

- account-hook 패턴 명령어
- alias 키 주소 [, 주소, ...]
- alias [* | 키 ...]
- alternative-order **mimetype** [**mimetype** ...]
- MIME 자동보기 **mimetype** [**mimetype** ...]
- bind **map** 키 **function**
- charset-hook **alias charset**

- color **object foreground background** [**regexp**]
- color **index** 패턴 [패턴 ...]
- exec **function** [**function** ...]
- fcc-hook 패턴 **mailbox**
- fcc-save-hook 패턴 **mailbox**
- folder-hook 패턴 명령어
- hdr-order **header** [**header** ...]
- hdr-order **header** [**header** ...]
- charset-hook **charset local-charset**
- ignore 패턴 [패턴 ...]
- ignore 패턴 [패턴 ...]
- lists 주소 [주소 ...]
- lists 주소 [주소 ...]
- macro **menu** 키 **sequence** [**description**]
- mailboxes **filename** [**filename** ...]
- mbox-hook 패턴 **mailbox**
- message-hook 패턴 명령어
- color **object attribute** [**regexp**]
- color **index** 패턴 [패턴 ...]
- my-hdr **string**
- my-hdr **field** [**field** ...]
- pgp-hook 패턴 키 **id**
- push **string**
- set **variable** [**variable** ...]
- save-hook **regexp filename**
- score-command 패턴 **value**
- score-command 패턴 [패턴 ...]
- send-hook **regexp** 명령어
- set [no|inv]**variable**[**=value**] [**variable** ...]
- set **variable** [**variable** ...]
- source **filename**
- lists 주소 [주소 ...]

- lists 주소 [주소 ...]
- set **variable** [**variable** ...]
- unhook **hook-type**

6.3. 설정 변수

변수값의 유형중에 quadoption은 ask-yes, ask-no, yes, no 의 값을 가질수 있으며, ask-yes/no 는 한번 확인하는 과정을 거치며 각각 yes와 no 가 기본이 된다는 의미이다. yes, no로 설정할 경우 확인과정을 거치지 않는다. 또한 boolean 은 보통 yes 또는 no 의 값을 사용할수 있다.

6.3.1. abort_nosubject

유형: quadoption

기본: ask-yes

yes : 메일 작성시 제목을 입력하지 않으면 메일 작성 취소. **no** : 제목을 입력하지 않아도 메일 작성 가능.

6.3.2. abort_unmodified

유형: quadoption

기본: yes

yes : 메일 본문의 내용에 변화가 없으면 메일 전송 취소.

6.3.3. alias_file

유형: path

기본: "<732>/.muttrc"

“create-alias” 명령에 의해 알리아스가 저장될 파일 지정

주의**:** Mutt 는 이 파일을 “source” 명령을 사용해서 지정해 주기 전까지는 인식하지 않는다.

6.3.4. alias_format

유형: 문자열

기본: "%4n %2f %t %-10a %r"

알리아스 메뉴에서 각 알리아스를 표시할때 사용되는 포맷을 결정 한다. 다음과 같이 printf(3)-스타일을 사용할수 있다:

%a

알리아스 이름

%f

플래그 - 기본적으로 "d" 는 삭제를 위해 표시된다.

%n

번호

%r

메일 주소

%t

포함을 위한 태그가 되어 있는 알리아스의 표시

6.3.5. allow_8bit

유형: boolean

기본: yes

no : 메일에 8 bit data가 포함되어 있는 QP (Quoted Printable)또는 Base64 인코딩 하여 보낸다.

6.3.6. allow_ansi

Type: boolean Default: no

ANSI 코드를 포함한 메일을 표시할때 색을 사용할것 인지 결정. 주의: ANSI 코드를 포함한 메일은 매우 드물고, 보안상의 문제등이 발생할수 있으므로, 사용에 주의해야 한다.

6.3.7. alternates

유형: 정규 표현식

기본: ""

당신이 메일을 수신하는 주소를 정규 표현식으로 지정한다.

6.3.8. arrow_cursor

유형: boolean

기본: no

yes로 설정되었을 경우 메일, 메일함등을 고르는 반전 막대가 사용되지 않고 (“->”) 가 사용된다. 느린 접속을 통해서 원격으로 mutt를 사용하는 사람들에게 유용한 설정이다.

6.3.9. ascii_chars

유형: boolean

기본: no

yes로 설정할 경우 글타래 표시때 **ACS** 문자 대신에 ASCII 문자를 사용

6.3.10. askbcc

유형: boolean

기본: no

메일 작성때 Bcc 수신자를 물어볼 것인가.

6.3.11. askcc

유형: boolean

기본: no

메일 작성때 cc 수신자를 물어본 것인가.

6.3.12. attach_format

유형: 문자열

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

첨부물 메뉴의 포맷을 결정한다. 다음과 같은 printf 스타일이 사용될수 있다:

%D

삭제 플래그

%d

설명

%e

MIME 인코딩 형식

%f

파일 이름

%I

설명 (I=인라인, A=첨부물)

%m

주 MIME 형식

%M

부속 MIME 형식

%n

첨부물 번호

%s

크기

%t

태그 플래그

%u

unlink (삭제) 플래그

%>X

오른쪽 공백을 채울 문자열 ("X")

%|X

줄의 끝까지를 채울 문자열 ("X")

6.3.13. attach_sep

유형: 문자열

기본: "\n"

저장, 프린트, 외부 프로그램으로 연결등이 사용될때 태그 표시된 첨부물 사이의 식별 문자

6.3.14. attach_split

유형: boolean

기본: yes

unset(no)로 설정될 경우, 표시된 첨부물들을 저장, 프린트, 외부 프로그램으로 연결등이 실행될때 mutt은 이것을 하나의 첨부물 로 처리한다. 반대로 “attach-sep” 이 첨부물 사이에 있다면 각각의 첨부물을 따로 처리하게 된다.

6.3.15. attribution

유형: 문자열

기본: "On %d, %n wrote:"

답장 작성시 원본 인용문 상단에 표시될 문자를 결정한다. 여기서 사용될수 있는 값은 “index-format”를 참조.

6.3.16. autoedit

유형: boolean

기본: no

“edit-headers”와 함께 사용될 경우 수신자, 제목을 묻지 않고 곧바로 메일 본문 편집으로 간다. 수신자, 제목등의 수정할수 있는 send 메뉴는 본문의 편집후에도 가능하다.

“fast-reply” 참고.

6.3.17. auto_tag

유형: boolean

기본: no

yes로 설정될 경우 **index** 메뉴에서 실행되는 명령은 자동적으로 태그 표시된 메일에 적용된다. unset(no)일 경우 태그된 메일에 어떠한 명령을 적용하려면 tag-prefix (기본: ";")를 사용한다.

6.3.18. beep

유형: boolean

기본: yes

오류 발생시 beep음을 낼것인가.

6.3.19. beep_new

유형: boolean

기본: no

새 메일이 도착했을때 beep음을 낼것인지 결정한다. 이 옵션은 “beep” 변수와는 독립적이다.

6.3.20. bounce_delivered

유형: boolean

기본: yes

바운스하는 메일에 Delivered-To 헤더를 포함할 것인지. Postfix 사용자는 no 로 설정해야 할 것이다. (왜?)

6.3.21. charset

유형: 문자열

기본: ""

사용하는 언어 설정, 예를 들어 한국어 환경 이라면 euc-kr 로 설정한다.

6.3.22. check_new

유형: boolean

기본: yes

주의**: maildir**과 **MH** 유형의 메일함에만 적용됨 mailboxes.

maildir 과 MH 스타일의 메일함을 사용하는 유저에게만 해당되며, 새 메일이 왔는지 확인하는 옵션

6.3.23. collapse_unread

유형: boolean

기본: yes

no 로 설정할 경우 메일 제목의 쓰레드 표시때 읽지 않은 메일이 있다면 글타래를 펼치지 않는다.

6.3.24. uncollapse_jump

유형: boolean

기본: no

yes : 현재 글타래가 펼쳐지지 않았을 경우 다음 읽지 않은 메일로 이동.

6.3.25. compose_format

유형: 문자열

기본: "-- Mutt: Compose [Approx. msg size: %l Atts: %a]%>-"

Compose (메일 작성) 메뉴의 상태 표시줄의 포맷을 지정한다. 여기에 사용되는 문자는 “status-format”와 유사하나 고유의 값을 가지고 있다:

%a

첨부물의 갯수

%h

로컬 호스트 이름

%l

현재 메일의 크기 (byte 근사값)

%v

Mutt의 버전

“status-format” 참조.

6.3.26. confirmappend

유형: boolean

기본: yes

메일함에 메일을 추가할때 확인하는 과정을 거칠것인지 결정.

6.3.27. confirmcreate

유형: boolean

기본: yes

새 메일함을 만들때 확인하는 과정을 거칠것인지 결정.

6.3.28. connect_timeout

유형: 숫자

기본: 30

IMAP 또는 POP 서버 접속시 이 시간내에 연결이 이루어 지지 않으면 연결을 포기한다. 마이너스 값을 지정하면 연결될때 까지 기다린다.

6.3.29. copy

유형: quadoption

기본: yes

보내지는 메일의 복사본을 저장 할것인지 결정.

“record”, “save-name”, “force-name”, “fcc-hook” 를 참조 할것.

6.3.30. date_format

유형: 문자열

기본: "!%a, %b %d, %Y at %I:%M:%S%p %Z"

이 변수는 “index-format”의 “%d”에 의해 표시되는 날짜의 포맷을 조정할수 있으며, 이 값은 **strftime**에 전달된다 (올바른 설정을 위해 man **strftime(3)**를 참조하라).

“!” 로 시작하지 않는 이상, 요일, 월의 표시는 “locale” 설정에 따라 확장된다. “!” 로 시작되면 **C** 로케일 설정을 사용하게 된다 (영문).

6.3.31. default_hook

유형: 문자열

기본: "<732>f %s !<732>P | (<732>P <732>C %s)"

이 변수는 send-hooks, message-hooks, save-hooks, fcc-hooks 에서 패턴을 생략하고 정규 표현식만을 사용할수 있도록 해준다.

6.3.32. delete

유형: quadoption

기본: ask-yes

삭제 표시된 메일이 삭제될때 확인 절차를 거칠것인지 결정한다.

6.3.33. delete_untag

유형: boolean

기본: yes

메일에 삭제 표시가 되면 태그를 해제한다.

6.3.34. digest_collapse

유형: boolean

기본: yes

revattach 메뉴에서 각 메일의 서브파트를 보여주지 않으며, 이것을 보기 위해서는 'v' 키를 사용한다.

6.3.35. display_filter

유형: path

기본: ""

메일의 표시때 외부 필터링 프로그램을 사용할때 지정해준다.

6.3.36. dotlock_program

유형: path

기본: "/usr/local/bin/mutt_dotlock"

mutt_dotlock (8) 실행 파일의 path

6.3.37. dsn_notify

유형: 문자열

기본: ""

주의**:** Sendmail 8.8.x 이상에서만 사용할 것.

이 변수는 알림 메세지가 리턴 되었을때 사용된다. 변수의 값은 하나 이상을 사용할수 있으며 . 로 구분해준다: **never**, 알림을 요청하지 않음. **failure**, 전송 실패 알림 요청. **delay**, 지연, 성공, 전송 성공시 알림.

예: set dsn_notify="failure,delay"

6.3.38. dsn_return

유형: 문자열

기본: ""

주의**:** Sendmail 8.8.x 이상에서만 사용할것.

되돌아 오는 DSN 메세지의 크기를 조절한다. 변수의 값은 **hdrs** (헤더만 돌아옴) 또는 **full**이 될수 있다.

예: set dsn_return=hdrs

6.3.39. duplicate_threads

Type: boolean Default: yes

메일의 쓰레드 정렬시에 적용되는 변수. 설정될 경우 message-id 가 같은 메일은 쓰레드 표시 때 이퀄 기호를 표시한다.

6.3.40. edit_headers

유형: boolean

기본: no

보내는 메일의 헤더를 본문과 함께 편집할수 있도록 한다.

6.3.41. editor

유형: path

기본: ""

mutt에서 사용될 편집기를 설정한다. 기본으로는 EDITOR 또는 VISUAL 환경 변수나 "vi"가 사용된다.

6.3.42. encode_from

유형: boolean

기본: no

설정 될경우 메일 본문이 "From "으로 시작할 경우 메일을 QP (quoted-printable) 인코딩 하여 보낸다.

6.3.43. envelope_from

유형: boolean

기본: no

메일을 보낼때 송신자를 sendmail의 "-f" 옵션을 사용하여 강제로 지정한다. 만약 sendmail에 이미 이 옵션을 사용했거나 지원하지 않을 경우에는 설정하지 않는다.

6.3.44. escape

유형: 문자열

기본: "<732>"

내장 에디터에서 사용될 escape 문자.

6.3.45. fast_reply

유형: boolean

기본: no

설정될 경우 답장/포워딩시 수신자, 제목을 묻지 않는다.

주의**:** 이 변수는 “autoedit”가 설정되어 있을 경우에는 효과가 없다.

6.3.46. fcc_attach

유형: boolean

기본: yes

보내지는 메일과 첨부물을 함께 저장할 것인지 결정한다.

6.3.47. fcc_clear

유형: boolean

기본: no

yes로 설정될 경우 보내지는 메일이 인코딩/사인 되었다 하더라도 FCC로 저장되는 메일은 인코딩/사인하지 않고 저장된다.

6.3.48. folder

유형: path

기본: "<732>/Mail"

메일함들이 위치하는 기본 위치를 지정한다. ‘+’ 와 ‘=’는 이 값에 의해 결정되므로 이 값을 변경하고자 한다면 ‘+’ 와 ‘=’를 사용한 다른 변수들을 확인한 후에 변경해야 한다.

6.3.49. folder_format

유형: 문자열

기본: "%2C %t %N %F %2l %-8.8u %-8.8g %8s %d %f"

파일 브라우져의 포맷을 지정한다. 사용되는 문자는 “index-format”와 유사하나 고유의 값을 가지고 있다:

%C

파일의 번호

%d

폴더가 마지막으로 수정된 날짜/시간

%f

파일 이름

%F

파일 허가권 (permission)

%g

그룹 이름 (이름이 없는 경우 gid 숫자)

%l

하드 링크 (hard link) 번호

%N

N 새 메일 표시

%s

크기 (byte)

%t

* 태그

%u

소유자 이름 (이름이 없는 경우 uid 숫자)

%>X

오른쪽 공백을 채울 문자열 ("X")

%|X

줄의 끝까지를 채울 문자열 ("X")

6.3.50. followup_to

유형: boolean

기본: yes

보내는 메일에 **Mail-Followup-To** 헤더를 추가 할것인지 결정한다. yes로 설정되면 당신이 가입한 (“subscribe” 나 “lists”에 정의된) 메일링 리스트에 답장을 보낼때 이 헤더를 추가하게 된다.

이 헤더가 가지는 목적은 두가지로 첫번째는 메일링 리스트에 보낸 메일의 답장이 이중으로 오는 것을 방지한다. 두번째로는 가입하지 않은 메일링 리스트에 보낸 메일의 답장이 당신에게로 오도록 함이다. **Mail-Followup-To** 헤더는 가입한 메일링 리스트의 경우 메일링 리스트의 주소만을 가입하지 않은 메일링 리스트에 보내는 경우에는 당신의 메일 주소만을 포함하게 된다.

6.3.51. force_name

유형: boolean

기본: no

이 변수는 “save-name”과 비슷하나, 보내는 메일을 송신자 이름(이메일 주소 ID)의 파일로 저장한다. (파일이 없으면 생성된다.)

“record” 참조.

6.3.52. forward_decode

유형: boolean

기본: yes

포워딩시에 MIME 인코딩된 원본 메일을 일반 텍스트로 디코딩해서 보낼것인지 결정한다. 원본 메일의 헤더 또한 RFC2047에 따라 디코딩한다. 이 변수는 “mime-forward”가 설정도지 않은 경우에만 해당하며, 다른 방법으로는 “mime-forward-decode”를 사용할수 있다.

6.3.53. forward_format

유형: 문자열

기본: "[%a: %s]"

메일의 포워딩시 제목의 형태를 결정한다. 여기에 사용될수 있는 값은 “index-format”와 같다.

6.3.54. forward_quote

유형: boolean

기본: no

yes로 설정될 경우 포워딩시 원본 메일을 본문에 인용문으로 포함시킨다 (“mime-forward” 가 **unset(no)**)일 경우 “indent-string”를 사용해서 인용한다).

6.3.55. from

유형: e-mail 주소

기본: ""

송신자의 메일 주소를 결정한다. 이 변수는 my_hdr (send-hooks 포함) 그리고 “reverse-name”에 의해 무시된다.

기본값은 EMAIL 환경변수를 사용한다.

6.3.56. gecos_mask

유형: 정규 표현식

기본: "^[^,]*"

A regular expression used by mutt to parse the GECOS field of a password entry when expanding the alias. By default the regular expression is set to "^[^,]*" which will return the string up to the first "," encountered. If the GECOS field contains a string like "lastname, firstname" then you should set the gecos_mask=".*".

This can be useful if you see the following behavior: you address a e-mail to user ID stevef whose full name is Steve Franklin. If mutt expands stevef to "Franklin" stevef@foo.bar then you should set the gecos_mask to a regular expression that will match the whole name so mutt will expand "Franklin" to "Franklin, Steve".

6.3.57. hdrs

유형: boolean

기본: yes

no로 설정될 경우 “my-hdr”로 설정된 헤더를 사용하지 않는다.

6.3.58. header

유형: boolean

기본: no

답장에 원본의 헤더까지 포함한다. “weed” 설정이 적용된다.

6.3.59. help

유형: boolean

기본: yes

화면의 첫번째 줄에 주요 기능과 바인딩된 키가 표시된다.

Note: 만약 해당 기능이 여러 문자에 바인딩 되어 있다면, 제대로 표시되지 않을 것이며, Mutt 가 실행되고 있는 도중에 키 설정을 바꾸는 경우에도 업데이트 되지 않을 것이다.

6.3.60. hidden_host

유형: boolean

기본: no

메일 주소에 “hostname”의 호스트 이름이 추가될때 무시되는 문자열. 이 변수는 Message-ID 과 주 도메인 이름에는 영향을 주지 않는다.

6.3.61. hide_missing

Type: boolean Default: yes

분실되거나 없는 메일을 글타래에서 인식하지 않는다.

6.3.62. history

유형: 숫자

기본: 10

히스토리 버퍼에 기억될 문자열의 갯수.

6.3.63. honor_followup_to

유형: quadoption

기본: yes

그룹-답장시 Mail-Followup-To 헤더에 의해 수신자가 정해질 것인지.

6.3.64. hostname

유형: 문자열

기본: ""

메일 주소의 “@” 다음에 사용될 호스트 이름. 이 설정은 컴파일시에 /etc/resolv.conf 로 부터 정의된 호스트 이름 보다 우선된다.

6.3.65. ignore_list_reply_to

유형: boolean

기본: no

메일링 리스트에서 온 메일에 답장을 할때 해당되는 설정으로, 만약 “Reply-To:”와 “To:” 헤더가 동일할 경우 “Reply-To:”는 무시한다. 이때 답장은 **list-reply**를 사용한다. **group-reply**는 송신자와 메일링 리스트에 동시에 보내게 된다.

6.3.66. imap_authenticators

유형: 문자열

기본: ""

IMAP 서버에 로그인 할때 사용되는 방법(method)을 정의한다. 한번에 여러가지를 정의 할수 있으면 이때에는 콜론으로 분리한다. mutt는 IMAP 서버에 접속할때 가장 안전한 방법부터 차례대로 시도할 것이다.

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

6.3.67. imap_delim_chars

유형: 문자열

기본: "/."

IMAP 패스를 표시하는데 사용될 각 폴더의 구문 문자.

6.3.68. imap_force_ssl

유형: boolean

기본: no

IMAP 서버에 접속할때 SSL 을 사용한다.

6.3.69. imap_home_namespace

유형: 문자열

기본: ""

IMAP 폴더 표시때 나타내는 path.

6.3.70. imap_keepalive

Type: number Default: 900

IMAP 접속시 사용되는 keepalive 시간을 조절한다.

6.3.71. imap_list_subscribed

유형: boolean

기본: no

IMAP 폴더 표시때 가입(subscribed)된 폴더만 보여줄 것인지 결정한다. 이것은 **toggle-subscribed** 명령과 함께 사용할수 있다.

6.3.72. imap_pass

유형: 문자열

기본: ""

IMAP 접속을 위한 암호를 지정. 지정 하지 않을 경우 fetch-mail 이 실행될때 입력할수 있다. 주의: 보안상 안전할 때에만 사용한다. 이 암호는 암호화 되지 않은 일반 텍스트로 저장 되어 있으므로 누구나 읽을수 있다는 위험이 있다.

6.3.73. imap_passive

유형: boolean

기본: yes

IMAP을 통한 새 메일 확인시 새로운 접속을 만들지 않고, 기본의 접속을 사용한다. 이것은 매번 사용자/암호를 입력하지 않아도 되며, 느린 접속일때 유용하다.

6.3.74. imap_peek

유형: boolean

기본: yes

설정될 경우 메일이 읽은것으로 표시되었다 하더라도 서버에서 모두 가져온다.

6.3.75. imap_servernoise

유형: boolean

기본: yes

IMAP 서버로 부터의 오류 메세지를 표시한다.

6.3.76. imap_user

유형: 문자열

기본: ""

IMAP 서버의 사용자 이름.
지정 하지 않을 경우 로컬 박스의 사용자 이름을 사용한다.

6.3.77. implicit_autoview

유형: boolean
기본: no
“yes”로 설정될 경우 모든 MIME 첨부물을 mailcap 항목에서 찾아본다.

6.3.78. include

유형: quadoption
기본: ask-yes
답장시 원본 인용할 것인지 결정한다.

6.3.79. indent_string

유형: 문자열
기본: "> "
인용문의 각 줄 앞에 들어갈 문자를 지정. 되도록이면 바꾸지 않는 것을 권장한다.

6.3.80. index_format

유형: 문자열
기본: "%4C %Z %{%b %d} %-15.15L (%4l) %s"
메일 인덱스 페이지의 포맷을 지정할수 있다.
“포맷 문자” 은 “C”에서 printf 함수에 사용되는 그것과 비슷하다:

%a
: 송신자의 주소

%b
: 메일함의 파일 이름

%B
: 리스트 또는 메일함 (%b).

%c

메일의 크기 (byte)

%C

현재 메일의 번호

%d

송신자의 시간대를 “date_format”로 변환한 날짜/시간

%D

로컬 시간대를 기준으로 메일이 보내진 시간을 “date_format”로 변환한 날짜/시간.

%e

현재 글타래의 메일 번호

%E

현재 글타래의 총 메일 갯수

%f

From: 헤더의 내용

%F

작성자 이름 또는 수신자 이름 (메일이 당신에게서 보내진 것일때)

%i

현재 메일의 message-id

%l

메일 내용의 행수

%L

메일링 리스트에서 온 메일

%m

현재 메일함의 총 메일 갯수

%M

글타래가 접혔을때 보이지 않는 메일의 갯수

%N

메일 점수 (스코어)

%n

글쓴이의 이름 (없으면 이메일 주소)

%O

원 저장 폴더 (_O_riginal save folder).

%s

메일의 제목

%S

메일의 상태 (N/D/d/!/r/*)

%t

수신자 (to:)

%T

to-chars 에 의한 문자.

%u

메일 작성자의 로그인 명

%v

메일 작성자 또는 수신자(당신이 보낸 메일인 경우)의 이름

%y

'x-label:' 헤더.

%Y

'x-label' 헤더. (1) 쓰레드의 부분이 아닐때, (2) 쓰레드의 가장 위일 경우, (3) 'x-label' 과 이전 메일의 'x-label'이 다른 경우.

%Z

메일 상태 플래그

%{fmt}

메일의 시간을 발송자의 시간대를 기준으로 변환한 것. "fmt"는 "strftime" 함수의 포맷이 사용된다.

%[fmt]

메일의 시간을 로컬 시간대를 기준으로 변환한 것. "fmt"는 "strftime" 함수의 포맷이 사용된다.

%(fmt)

메일을 받은 로컬 시간. “fmt”는 “strftime” 함수의 포맷이 사용된다.

%<fmt>

현재 시각. “fmt”는 “strftime” 함수의 포맷이 사용된다.

%>X

오른쪽 공백을 채울 문자열 ("X")

%|X

줄의 끝까지를 채울 문자열 ("X")

“to-chars” 를 참조.

6.3.81. ispell

유형: path

기본: "/usr/bin/ispell"

ispell (스펠링 첵크 프로그램)의 경로 지정.

6.3.82. keep_flagged

유형: boolean

기본: no

스풀 메일함에서 읽은 메일에 표시할 경우 “mbox” 메일함으로 메일을 옮기지 않는다. “mbox-hook” 에도 적용된다.

6.3.83. locale

유형: 문자열

기본: "C"

strftime(3)에 의해 사용될 로케일 설정. 허용되는 값은 **LC_TIME** 환경 변수에 사용될수 있는 것과 같다.

6.3.84. mail_check

유형: 숫자

기본: 5

얼마나 자주 새 메일을 체크할 것인지. (초단위)

6.3.85. mailcap_path

유형: 문자열

기본: ""

mutt이 지원하지 않는 MIME 메일을 표시할때 사용되는 mailcap 파일의 위치를 지정한다.

6.3.86. mailcap_sanitize

유형: boolean

기본: yes

mailcap % 의 문자를 잘 알려진 안전한 문자로 제한한다.

당신이 무엇을 하고 있는지 알기 전까지는 바꾸지 말것**!**

6.3.87. maildir_trash

유형: boolean

기본: no

설정될 경우 삭제를 위해 표시된 메일은 삭제되지 않고 (T)rash 표시된 메일함에 저장된다.
참고**:** 이것은 maildir 형태의 메일함에서만 적용된다.

6.3.88. mark_old

유형: boolean

기본: yes

현재 메일함의 새로온 메일을 읽지 않았을 경우 old 표시를 할 것인지 결정한다.

6.3.89. markers

유형: boolean

기본: yes

내장 페이저에 의해 wrap된 라인을 표시할때 “+” 문자를 표시할것인지 결정한다.
“smart-wrap” 참조.

6.3.90. mask

유형: 정규 표현식

기본: "!^\.[^.]"

파일 브라우져에 사용되는 정규 표현식으로 “!”를 사용해서 매치되는 파일을 제외한 것만 표시하도록 할수 있다.

6.3.91. mbox

유형: path

기본: "<732>/mbox"

스풀 메일함(spoolfile)에서 읽은 메일이 저장될 메일함.

6.3.92. mbox_type

유형: folder magic

기본: mbox

새로운 폴더를 만들때 사용되는 메일함의 유형 (mbox, MMDF, MH, Maildir)

6.3.93. metoo

유형: boolean

기본: no

unset(no)일 경우 답장 작성시 수신자의 주소중에서 자신의 주소를 삭제 한다.

6.3.94. menu_scroll

유형: boolean

기본: no

설정될 경우, 메뉴가 한줄 단위로 위아래로 스크롤된다. **unset**될 경우 전체 화면을 지우고 다시 그리게 되므로, 느린 접속 사용자들에게 유용한 옵션이다.

6.3.95. meta_key

유형: boolean

기본: 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 0xf4, then this is treated as if the user had pressed ESC then “x”. This is because the result of removing the high bit from “0xf4” is “0x74”, which is the ASCII character “x”.

6.3.96. mh_purge

유형: boolean

기본: no

unset될 경우 삭제 표시된 메일은 mh 스타일에 따라 **,<old file name>**형식으로 저장된다. set될 경우에는 메일은 삭제된다.

6.3.97. mh_seq_flagged

유형: 문자열

기본: "flagged"

플래그 표시된 메일에 사용되는 MH 시퀀스.

6.3.98. mh_seq_replied

유형: 문자열

기본: "replied"

답장을 보낸 메일을 표시하는 MH 시퀀스.

6.3.99. mh_seq_unseen

유형: 문자열

기본: "unseen"

읽지 않은 메일을 표시하는 MH sequence의 이름.

6.3.100. mime_forward

유형: quadoption

기본: no

설정될 경우 포워딩되는 메일은 MIME 첨부물로 보내진다.

Also see “forward-decode” and “mime-forward-decode”.

6.3.101. mime_forward_decode

유형: boolean

기본: no

"mime-forward" 가 **set**되어 있을때 포워딩하는 메일을 text/plain로 디코딩할지 결정한다. 이외에는 "forward-decode"가 사용된다.

6.3.102. mime_forward_rest

유형: quadoption

기본: yes

recvattach 메뉴에서 MIME 메세지의 다중 첨부물을 포워딩할때, 어떠한 이유로든 첨부물의 디코딩이 불가능한 경우 새로운 메일에 첨부되도록 한다. (?)

6.3.103. mix_entry_format

유형: 문자열

기본: "%4n %c %-16s %a"

mixmaster 체인 고르기 메뉴에서 사용될 리메일러의 표시 형태를 결정한다. 다음의 printf 형식의 시퀀스를 지원한다:

%n

메뉴상의 숫자.

%c

리메일러의 능력.

%s

리메일러의 짧은 이름.

%a

리메일러의 이메일 주소.

6.3.104. mixmaster

유형: path

기본: "mixmaster"

Mixmaster 실행 파일의 패스를 지정한다.

6.3.105. move

유형: quadoption

기본: ask-no

스풀 메일함에서 읽은 메일을 “mbox”로 옮겨오는 것을 제어하거나, “mbox-hook” 명령의 결과를 제어한다.

6.3.106. message_format

유형: 문자열

기본: "%s"

“attachment” (첨부물) 메뉴에서 첨부물의 형식을 표시하는 포맷을 정의한다. 사용할수 있는 시퀀스 문자들은 “index-format”를 참고한다.

6.3.107. pager

유형: path

기본: "builtin"

외부 페이저를 사용할 경우 지정해준다. 기본적으로 내장 페이저를 사용한다.

6.3.108. pager_context

유형: 숫자

기본: 0

내장 페이저에서 한 페이지로 인식할 행수를 지정한다. 기본적으로 현재 표시되지 않은 줄을 시작으로 표시된다.

6.3.109. pager_format

유형: 문자열

기본: "-%Z- %C/%m: %-20.20n %s"

내장 또는 외부 페이저로 메일을 읽기 전에 표시되는 메일의 “상태”를 줄의 형식을 결정한다. 사용되는 시퀀스 문자는 “index-format”를 참조한다.

6.3.110. pager_index_lines

유형: 숫자

기본: 0

0 이상으로 설정될 경우 페이저에서 메일의 내용과 메일의 목록(인덱스)을 함께 표시하며, 표시되는 목록의 행수를 결정한다.

6.3.111. pager_stop

유형: boolean

기본: no

설정될 경우 내장 페이저에서 **next-page** 기능이 실행될때 다음 메일로 이동하지 않는다.

6.3.112. pgp_autosign

유형: boolean

기본: no

발송되는 모든 메일에 자동으로 PGP/MIME 사인을 한다. 이 값은 **pgp-menu**에서 사인할것인지 아닌지를 결정함으로서 무시될수 있다.

6.3.113. pgp_autoencrypt

유형: boolean

기본: no

설정될 경우 발송되는 메일을 자동으로 PGP/MIME 암호화한다.

6.3.114. pgp_ignore_subkeys

유형: boolean

기본: yes

OpenPGP subkey를 무시한다.

6.3.115. pgp_entry_format

유형: 문자열

기본: "%4n %t%f %4l/0x%k %-4a %2c %u"

PGP 키 선택 메뉴의 형태를 결정할수 있다. 사용되는 문자는 “index-format”와 비슷하나 고유의 시퀀스를 가지고 있다:

%n

숫자

%k

키 id

%u

유저 id

%a

알고리즘

%l

키의 길이

%f

플래그

%c

능력

%t

key-uid 와 관련된 신뢰/인증

%[<s>]

키의 날짜 (<s>는 strftime(3)에 의해 사용됨)

6.3.116. pgp_good_sign

유형: 정규 표현식

기본: ""

pgp-verify-command의 결과값에 여기에 설정한 패턴과 일치하면 확인된 서명으로 인식한다.

6.3.117. pgp_long_ids

유형: boolean

기본: no

설정될 경우 64 bit PGP key ID를 사용한다. 이외에는 32 bit Key ID가 사용된다.

6.3.118. pgp_replyencrypt

유형: boolean

기본: no

설정될 경우 PGP 암호화된 메일의 답장은 자동으로 PGP 암호화 한다.

6.3.119. pgp_replysign

유형: boolean

기본: no

설정될 경우 PGP 사인된 메일의 답장은 자동으로 PGP 사인한다.

주의**:** 이것은 서명/암호화된 메일일 경우에는 작동하지 않는다.

6.3.120. pgp_replysignencrypted

유형: boolean

기본: no

설정될 경우 PGP 암호화된 메일의 답장은 자동으로 PGP 사인한다.

6.3.121. pgp_retainable_sigs

유형: boolean

기본: no

설정될 경우 사인과 암호화된 메일은 multipart/signed 와 multipart/encrypted로 분리된다.

6.3.122. pgp_show_unusable

유형: boolean

기본: yes

설정될 경우 mutt는 사용할수 없는 키를 PGP 키 선택 메뉴에서 표시한다. 여기서 사용할수 없는 키란 취소, 만기되거나 사용자에 의해 “사용불가” 표시된 키를 의미한다.

6.3.123. pgp_sign_as

유형: 문자열

기본: ""

만약 당신의 한개 이상의 키쌍을 가지고 있다면, 어떤 키를 사용할 것인지 설정한다. 사용할 키의 keyid (예를 들어 “0x00112233”)의 사용을 권장한다.

6.3.124. pgp_strict_enc

유형: boolean

기본: yes

설정될 경우 mutt는 자동으로 PGP/MIME 사인된 메일을 **quoted-printable**로 인코딩한다.

6.3.125. pgp_timeout

유형: 숫자

기본: 300

캐쉬된 암호구문(passphrase)을 몇초후에 무효화 할것인지 결정한다.

6.3.126. pgp_verify_sig

유형: quadoption

기본: yes

“yes”로 설정할 경우 PGP/MIME 서명을 자동으로 확인한다. “ask-yes” 또는 “ask-no”로 설정할 경우에는 확인할 것인지 물어보게 되며, “no”일 경우에는 서명을 확인하지 않는다.

6.3.127. pgp_sort_keys

유형: sort order

기본: address

‘pgp 키’ 메뉴에서 키의 목록을 표시할때 정렬 옵션을 정의한다. 다음 값을 사용할수 있다:

address

유저 id의 알파벳 순

keyid

key id의 알파벳 순

date

키의 날짜순으로

trust

키의 신뢰도 순으로

역순으로 정렬하고자 한다면 ‘reverse-’를 붙여준다.

6.3.128. pgp_create_traditional

유형: quadoption

기본: no

이 옵션은 mutt이 오래된 형식(old-style)의 PGP 암호화 또는 서명 메일을 만들도록 한다.

PGP/MIME은 메일의 문자셋이 us-ascii이 아닐경우 자동으로 사용되거나, 하나 이상의 MIME 파트를 포함할 경우 사용된다.

이 형태의 PGP 사용은 사용하지 말것을 권장한다.

6.3.129. pgp_decode_command

유형: 문자열

기본: ""

application/pgp 형식의 첨부물을 디코딩하는 명령어.

PGP 명령어 포맷은 고유의 printf 시퀀스 문자를 가지고 있다:

%p

암호 구문(passphrase)가 필요할 경우 PGPPASSFD=0로 확장됨. 참고: 이것은 %?와 함께 사용될수 있다.

%f

메일이 포함하고 있는 파일의 이름.

%s

multipart/signed 첨부물 서명 부분의 파일 이름.

%a

pgp-sign-as의 값.

%r

하나 또는 그 이상의 key ID.

이 설정들은 사용되는 PGP 버전에 의해 유동적일수 있다. 시스템에 설치되어 있는 pgp*.rc와 gpg.rc 들과 문서를 참고하라.

6.3.130. pgp_getkeys_command

유형: 문자열

기본: ""

공개키 정보가 필요할때 사용되는 명령어. %r 은 printf 타입의 시퀀스가 사용된다.

6.3.131. pgp_verify_command

유형: 문자열

기본: ""

PGP/MIME 서명을 확인하는 명령어.

6.3.132. pgp_decrypt_command

유형: 문자열

기본: ""

PGP/MIME 암호화된 메일을 해독하는 명령어.

6.3.133. pgp_clearsign_command

유형: 문자열

기본: ""

This format is used to create a "clearsigned" old-style PGP attachment. 오래된 PGP "clearsigned" 방식의 첨부물을 추가하는 명령어. 이 형태의 사용은 강력히 반대 한다.

6.3.134. pgp_sign_command

유형: 문자열

기본: ""

메일에 분리된 multipart/signed PGP/MIME 서명 하는 명령어.

6.3.135. pgp_encrypt_sign_command

유형: 문자열

기본: ""

메일의 본문을 서명/암호화 하는 명령어.

6.3.136. pgp_encrypt_only_command

유형: 문자열

기본: ""

메일의 본문을 서명없이 암호할때 쓰이는 명령어.

6.3.137. pgp_import_command

유형: 문자열

기본: ""

메세지에 포함된 키를 유저의 공개 키링으로 추가하는 명령어.

6.3.138. pgp_export_command

유형: 문자열

기본: ""

공개 pgp 키를 유저의 키링(key ring)에서 추출할때 사용할때 사용하는 명령어.

6.3.139. pgp_verify_key_command

유형: 문자열

기본: ""

pgp 키를 확인할때 사용되는 명령어.

6.3.140. pgp_list_secring_command

유형: 문자열

기본: ""

gpg --list-keys --with-colons. php 키링에서 비공개 키의 목록을 보는 명령어. 출력되는 형식은 “gpg --list-keys --with-colons”의 결과와 유사해야 한다.

이 형식은 mutt에 따라오는 pgpring 유틸리티를 사용할수도 있다.

6.3.141. pgp_list_pubring_command

유형: 문자열

기본: ""

php 키링에서 공개 키의 목록을 보는 명령어. 출력되는 형식은 “gpg --list-keys --with-colons”의 결과와 유사해야 한다.

이 형식은 mutt에 따라오는 pgpring 유틸리티를 사용할수도 있다.

6.3.142. forward_decrypt

유형: boolean

기본: yes

암호화된 메세지의 포워딩시에 그대로 보낼것인지 디코딩해서 보낼것인지 결정한다. 이 변수는 “mime-forward”가 **set**되거나 “mime-forward-decode” 가 **unset**되었을때 만 사용된다.

6.3.143. ssl_starttls

유형: quadoption

기본: yes

설정될 경우 (기본), mutt는 서버의 STARTTLS 기능을 사용하게 된다.

6.3.144. certificate_file

유형: path

기본: ""

인증서가 저장되어 있는 파일을 지정한다. 알려지지 않은 인증서를 받아들일 경우 이 파일에 저장되어 다음 접속에 자동적으로 사용된다.

또한, 당신은 CA 인증서를 수동으로 이 파일에 추가할수 있다.

예: set certificate_file=<732>/.mutt/certificates

6.3.145. ssl_usesystemcerts

유형: boolean

기본: yes

yes로 설정될 경우 mutt 는 시스템의 CA 인증서를 사용한다.

6.3.146. entropy_file

유형: path

기본: ""

SSL 라이브러리 함수의 초기화시 사용될 무작위 데이터를 포함하고 있는 파일,

6.3.147. ssl_use_sslv2

유형: boolean

기본: yes

SSL 인증시 SSLv2의 사용 여부.

6.3.148. ssl_use_sslv3

유형: boolean

기본: yes

SSL 인증시 SSLv3의 사용 여부.

6.3.149. ssl_use_tlsv1

유형: boolean

기본: yes

SSL 인증시 TLSv1의 사용 여부.

6.3.150. pipe_split

유형: boolean

기본: no

pipe-message 명령과 관련된 변수로 unset 되었을 경우에는 태그된 메일들을 한번에 외부 명령으로 내보낸다. yes로 설정되었을 경우 메일을 현재 정렬 방법에 따라 차례대로 외부 명령으로 내보내며 “pipe-sep” 구분 문자를 붙인다.

6.3.151. pipe_decode

유형: boolean

기본: no

pipe-message 명령과 관련된 변수. 설정될 경우 외부 명령으로 연결하기 전에 헤더를 숨기고, 메일을 PGP/MIME 디코딩한후 보낸다.

6.3.152. pipe_sep

유형: 문자열

기본: "\n"

태그한 메일을 외부 명령으로 연결할때 추가되는 문자.

6.3.153. pop_authenticators

유형: 문자열

기본: ""

POP 서버에 접속할때 사용되는 인증 방법을 콜론으로 구분하여 나열한다. 인증 방법은 'user', 'apop', SASL, 'digest-md5', 'gssapi', 'cram-md5'등이 사용될수 있다. 따로 설정되지 않았을 경우 지원하는 모든 인증 방법을 가장 안전한것부터 시도해 본다.

예: set pop_authenticators="digest-md5:apop:user"

6.3.154. pop_auth_try_all

유형: boolean

기본: yes

지원하는 모든 인증 방법을 사용해서 POP 서버에 접속한다. 만약 POP 서버가 어떤 인증방법을 지원하지만 로그인에 실패하면 mutt는 더 이상 POP 서버에 접속하지 않는다.

6.3.155. pop_checkinterval

유형: 숫자

기본: 60

얼마나 자주 POP 서버의 새 메일을 확인할 것인지 설정 (초 단위).

6.3.156. pop_delete

유형: quadoption

기본: ask-no

fetch-mail 기능에 의해 POP 서버로 부터 메일이 성공적으로 다운로드 되었을때 서버에서 메일을 삭제할것인지 결정한다.

6.3.157. pop_host

유형: 문자열

기본: ""

fetch-mail 기능에 의해 사용될 POP 서버. 다음과 같이 포트, 사용자, 암호를 같이 설정할수도 있다:

[pop[s]://][username[:password]@]popserver[:port]

6.3.158. pop_last

유형: boolean

기본: no

설정될 경우 mutt는 fetch-mail 기능이 실행될때 "마지막" POP 명령을 사용해 새 메일을 POP 서버로 부터 가져온다.

6.3.159. pop_reconnect

유형: quadoption

기본: ask-yes

POP 서버와의 접속이 끊어졌을때 재접속을 시도할것인지 결정한다.

6.3.160. pop_user

유형: 문자열

기본: ""

POP 서버의 계정.

기본값으로 로컬 사용자 이름을 사용한다.

6.3.161. pop_pass

유형: 문자열

기본: ""

POP 계정의 암호를 지정한다. 설정되지 않았을 경우에는 POP 메일함을 열때 물어볼 것이다. 경고: 이것은 안전할 경우에만 설정한다. 예를 들어 superuser는 당신의 muttrc 파일을 읽어볼 수 있다.

6.3.162. post_indent_string

유형: 문자열

기본: ""

“attribution” 변수와 비슷하지만, 메일을 인용한 후에 추가되는 문자열을 지정한다.

6.3.163. postpone

유형: quadoption

기본: ask-yes

메일 발송을 취소할때 발송 연기 메일함(“postponed”)으로 저장할지를 물어보는 프로프트를 보여줄 것인지 결정한다.

6.3.164. postponed

유형: path

기본: "<732>/postponed"

발송 연기되는 메일이 저장되는 메일함을 지정한다. “postpone”를 참고. variable.

6.3.165. preconnect

유형: 문자열

기본: ""

이 옵션은 ssh를 이용해서 암호화된 POP 접속을 만드는데 사용될수 있다. 다음 예를 보라:

preconnect="ssh -f -q -L 1234:mailhost.net:143 mailhost.net sleep 20 < /dev/null > /dev/null"

이제 ’{localhost:1234}foo’ 로 메일함 ’foo’에 접근할수 있다.

주의: 이것이 제대로 작동하기 위해서는 암호 입력 과정없이 ssh 로그인이 가능하도록 미리 설정되어야 한다.

6.3.166. print

유형: quadoption

기본: ask-no

메일의 프린트시 프롬프트를 보여줄것 인지 결정한다. 이것은 실수로 “p”를 자주 누르는 사람(나를 포함)들에게 유용하다.

6.3.167. print_command

유형: path

기본: "lpr"

프린트에 사용될 프로그램을 지정한다.

6.3.168. print_decode

유형: boolean

기본: yes

print-message 명령과 관련된 변수이다. 설정될 경우 메일이 디코딩된후 print-command 변수에 설정된 프린트 명령으로 보내진다.

6.3.169. print_split

유형: boolean

기본: no

print-message 명령과 관련된 변수이다. 설정될 경우 print-command 명령이 하나의 메일에 한번씩 실행된다. unset일 경우에는 모든 메일이 한번에 print-command 로 보내지며, 각 메일은 폼 피드로 연결된다.

enscript(1) 과 같은 프린팅 유틸리티를 사용하는 사람들은 이 옵션을 사용하는것이 좋을것이다.

6.3.170. prompt_after

유형: boolean

기본: yes

외부 “pager”를 사용할 경우, 외부 명령의 종료후 프롬프트를 보여줄 것인지 결정한다. unset될 경우 종료후 곧바로 인덱스 메뉴로 돌아온다.

6.3.171. query_command

유형: path

기본: ""

메일 주소 쿼리를 위한 외부 프로그램을 지정한다. 프로그램의 패스와 함께 반듯이 %s 를 사용해줘야 한다. 자세한 것은 “query” 를 참고하라.

6.3.172. quit

유형: quadoption

기본: yes

mutt를 종료할때 프롬프트를 보여줄 것인지 결정한다.

6.3.173. quote_regexp

유형: 정규 표현식

기본: "^([\t]*[|>:}#])+"

내장 페이저가 인용문이라고 판단할 기호를 정규 표현식으로 설정.

주의**: quotedx** 패턴을 내장 페이저에서 사용하기 위해서는 정확히 인용 부호와 일치되는 정규 표현식을 사용해야 한다.

6.3.174. read_inc

유형: 숫자

기본: 10

메일함을 읽어 들일때 상태창에 표시되는 진행 상황에 사용되는 값이다. 이 값은 얼마만큼의 간격을 두고 메일의 수를 표시할 것인가를 정한다.

“write-inc” 참고.

6.3.175. read_only

유형: boolean

기본: no

모든 메일함을 읽기 전용으로 연다.

6.3.176. realname

유형: 문자열

기본: ""

메일을 보낼때 사용되는 "본명"을 설정한다.

기본은 /etc/passwd 파일의 GCOS 필드를 참고하며, from 변수가 설정되어 있을 경우 이 값은 사용되지 않는다.

6.3.177. recall

유형: quadoption

기본: ask-yes

새로운 메일을 작성할때 발송 연기된 메일이 있다면 프롬프트를 보여줄 것인지 결정한다 (“postponed” 참고).

6.3.178. record

유형: path

기본: ""

보내지는 메일이 저장될 곳을 지정한다 (“my-hdr”로 **Bcc:** 헤더를 지정하는 방법도 있다).

record 값은 “force-name”, “save-name”, “fcc-hook”에 의해 무시될수 있다.

6.3.179. reply_regexp

유형: 정규 표현식

기본: "^(re([\[0-9\]+])*|aw):[\t]*"

메일이 답장이라고 인식하는데 사용되는 정규 표현식. 기본적으로 영문에서는 "Re:" 와 독일어에서는 "Aw:"가 사용된다.

6.3.180. reply_self

유형: boolean

기본: no

unset 되어 있을때는 당신이 보낸 답장에 답장을 쓸때 수신자 주소를 mutt는 이것이 당신에게 보내지는 것이 아니라고 인식하고 수신자 주소를 바꾼다.

6.3.181. reply_to

유형: quadoption

기본: ask-yes

답장 작성시 Reply-to: 헤더를 참고할 것인지 결정한다. no 로 설정되면 From: 헤더의 주소를 사용하게 된다.

6.3.182. resolve

유형: boolean

기본: yes

현재 메일에 어떠한 명령을 실행한 후에 자동으로 다음 메일로 커서를 이동시킨다.

6.3.183. reverse_alias

유형: boolean

기본: no

인덱스 메뉴에서 송신자의 본명을 표시할것인지 결정한다. yes로 설정될 경우 송신자의 메일 주소와 일치하는 알리아스를 참고해서 주소 대신에 이름을 표시한다. 예를 들어 다음과 같은 알리아스가 설정되어 있고:

```
alias juser abd30425@somewhere.net (Joe User)
```

수신된 메일의 From: 헤더가 다음과 같다면:

```
From: abd30425@somewhere.net
```

인덱스 메뉴에서 “abd30425@somewhere.net” 대신에 “Joe User”를 표시하게 된다. 이것은 메일주소만으로 누구인지 알아보기 힘든 경우 유용할 것이다.

6.3.184. reverse_name

유형: boolean

기본: no

답장 작성시 메일을 수신한 곳을 기준으로 **From:**을 사용한다. 즉, 메일의 수신 주소가 다른 경우에 사용될수 있다.

6.3.185. reverse_realname

유형: boolean

기본: yes

reverse-name 변수의 추가적으로 본명 또한 수신 헤더를 참고하여 답장시 From: 의 내용을 만든다. unset 되어 있을 경우에는 realname를 사용한다.

6.3.186. rfc2047_parameters

유형: boolean

기본: no

RFC-2047-encoded MIME 옵션을 사용한다. 첨부물의 파일 이름은 다음과 같은 형태로 인코딩 된다: =?iso-8859-1?Q?file=5F=E4=5F991116=2Ezip?=

6.3.187. save_address

유형: boolean

기본: no

메일의 저장시 송신자의 주소가 사용된다. 만약 “save-name” 또는 “force-name”가 같이 설정되었을 경우에는 fcc 메일함 또한 변경된다.

6.3.188. save_empty

유형: boolean

기본: yes

unset일 경우 메일을 포함하고 있지 않은 메일함을 삭제된다 (“spoolfile”은 제외).

주의**:** 이것은 mbox 나 MMDF 메일함의 경우에만 해당하며, MH 나 Maildir 형식의 메일함은 삭제하지 않는다.

6.3.189. save_name

유형: boolean

기본: no

보내는 메일을 어느곳에 저장할 것인지 결정한다. 설정될 경우 “folder” 디렉토리안에 수시자의 메일 주소에서 유저 이름과 같은 파일이 있다면 그곳에 저장되고 없는 경우에는 “record”변수에 지정된 메일함에 저장한다.

Also see the “force-name” variable.

6.3.190. score

유형: boolean

기본: yes

unset될 경우 메일의 스코어링을 사용하지 않는다. 이것은 특정 메일함에서 스코어링을 사용하지 않거나 “score-threshold-delete” 변수가 사용될때 유용하다.

6.3.191. score_threshold_delete

유형: 숫자

기본: -1

메일의 점수가 이 값보다 작거나 같을 경우 자동으로 삭제 표시한다. 기본적으로 메일의 점수는 언제가 0보다 크므로, 기본 설정값으로는 언제나 삭제 표시를 하지 않을 것이다.

6.3.192. score_threshold_flag

유형: 숫자

기본: 9999

메일의 점수가 이 값보다 크거나 같을 경우 자동으로 중요 표시한다.

6.3.193. score_threshold_read

유형: 숫자

기본: -1

메일의 점수가 이 값보다 작거나 같을 경우 자동으로 읽은 메일 표시한다.

6.3.194. send_charset

유형: 문자열

기본: "us-ascii:iso-8859-1:utf-8"

보내지는 메일의 문자셋의 목록. 만약 사용하는 “charset”이 iso-8859-1이 아니거나, 수신자가 UTF-8를 사용할수 없는 경우 일반적으로 사용되는 문자셋을 사용하는 것이 좋다 (예: set send_charset="us-ascii:iso-8859-1:euc-kr").

주의**:** mutt 1.2.x 버전에서는 set send_charset=euc-kr 로 설정 해준다.

6.3.195. sendmail

유형: path

기본: "/usr/sbin/sendmail -oem -oi"

메일 발송에 사용되는 프로그램과 옵션을 설정한다.

6.3.196. sendmail_wait

유형: 숫자

기본: 0

“sendmail”의 메일 발송 프로세스를 얼마 동안 기다릴 것인지 초단위로 결정한다.

0 보다 클 경우

“sendmail”이 메일을 보낼때 까지 입력한 만큼 기다리다가 계속 한다.

0

sendmail이 처리를 끝낼때 까지 계속 기다린다.

<0

sendmail을 기다리지 않고 백 그라운드로 처리한다.

0이 아닐 경우 하부 프로세스는 임시파일을 사용하며, 오류가 발생 했을 경우 어디서 이 파일을 찾을수 있는지 보여준다.

6.3.197. shell

유형: path

기본: ""

mutt이 사용할 하부 쉘. 기본적으로 사용자의 로그인 쉘이 사용된다.

6.3.198. sig_dashes

유형: boolean

기본: yes

“signature”의 이전줄에 “-- ”를 자동으로 추가한다. 이것은 매우 권장되는 것으로 많은 프로그램들이 “-- \n”을 기준으로 서명을 인식하기 때문이다. 예를 들어 mutt에서도 이것을 기준으로 페이저에서 서명을 표시할때 다른 색을 사용할수 있다.

6.3.199. sig_on_top

유형: boolean

기본: no

인용문이나 포워딩되는 메일의 윗부분에 서명이 포함된다. 이것은 일반적으로 사용되지 않는 것이 네티켓을 지키는 것이다.

6.3.200. signature

유형: path

기본: "<732>/.signature"

보내는 메일에 포함되는 서명 파일을 지정한다. 파일의 이름에 “|”를 붙여주면 해당 파일을 실행한 후에 그 결과를 사용하게 된다.

6.3.201. simple_search

유형: 문자열

기본: "<732>f %s | <732>s %s"

<732> 를 포함하지 않은 검색 문자열을 어떻게 확장할 것인지 결정한다. “패턴” 에 자세한 검색 패턴에 대한 설명이 있다.

예를 들어 일반적은 메일 검색시 joe 만을 입력하였다면, mutt는 자동적으로 다음 패턴을 사용해서 검색한다:

<732>f joe | <732>s joe

6.3.202. smart_wrap

유형: boolean

기본: yes

내장 페이저에서 메일 내용의 한줄의 길이가 터미날의 폭보다 길때 사용되는 wrap 기능을 사용할것인지 결정한다. “markers” 변수 참고.

6.3.203. smileys

유형: 정규 표현식

기본: "(>From)|(:[-^]?[][)(><}{|/DP])"

내장 페이저가 “quote-regexp”를 사용해서 인용문을 인식할때 만약 줄의 시작에 스마일리(smileys)가 사용될 경우 제대로 인식하지 못하는 것을 방지하기 위해 사용된다. 즉, 스마일리로 인식할 문자를 정규 표현식으로 표현한다.

6.3.204. sleep_time

유형: 숫자

기본: 1

메일의 이동하거나 삭제한후 알림 메세지가 보여지면서 지체되는 시간을 지정해 준다. 기본값은 1초이다.

6.3.205. sort

유형: sort order

기본: date

index 메뉴에서 메일들의 정렬 방법

```
date 또는 date-sent
date-received
from
mailbox-order (정렬하지 않음)
```

```
score
size
subject
threads
to
```

"reverse-"를 붙이면 역으로 정렬한다 (예: set sort=reverse-date-sent).

6.3.206. sort_alias

유형: sort order

기본: alias

알리아스 메뉴에서 알리아스들의 정렬 방법:

```
address (이메일 주소)
alias (알리아스 이름)
unsorted (정렬하지 않음)
```

6.3.207. sort_aux

유형: sort order

기본: date

쓰레드 정렬을 사용할때 해당 글타래안의 메일들을 어떻게 정렬할 것인지 결정한다. 여기에 쓰일수 있는 값은 "sort"의 그것과 동일하다 (thread 제외). 또한, last- , reverse- 같은 접두사를 사용할수 있으며, last- 는 reverse- 뒤에 와야 한다.

6.3.208. sort_browser

유형: sort order

기본: subject

파일 브라우져에서 파일의 정렬 방법을 설정한다. 기본값은 파일의 이름으로 정렬한다. 사용될수 있는 값은:

```
alpha (알파벳으로)
date
size
unsorted
```

"reverse-" 를 붙이면 역으로 정렬한다 (예: set sort_browser=reverse-date).

6.3.209. sort_re

유형: boolean

기본: yes

이 변수는 “strict-threads”이 unset 되고, 쓰레드 정렬을 할때만 사용할수 있다. 설정될 경우 mutt는 오직 제목이 “reply-regexp”로 시작되는 메일만을 쓰레드에 포함시킨다.

6.3.210. spoolfile

유형: path

기본: ""

스풀 메일함이 일반적이지 않은 위치에 있는 경우 설정 해준다. mutt는 이 값을 사용해서 $MAIL 환경변수를 정의한다.

6.3.211. status_chars

유형: 문자열

기본: "-*%A"

“status-format”에서 사용되는 "%r" 지시자의 문자를 지정한다. 첫번째 문자는 메일함이 바뀌지 않았을때, 두번째는 메일함이 수정되었을때, 세번째는 메일함이 읽기 전용모드일때 (이것은 toggle-write (기본: %) 명령으로 바꿀수 있다.), 네번째 문자는 현재 폴더가 메일 첨부물 모드로 열린 경우를 나타낸다.

6.3.212. status_format

유형: 문자열

기본: "-%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)---"

인덱스 메뉴에서 상태창의 포맷을 결정한다. 사용되는 문자들은 “index-format”와 비슷하지만, 고유의 값을 가지고 있다:

%b

새 메일을 포함하고 있는 메일함의 갯수 *

%d

삭제한 메일의 갯수 *

%h

로컬 호스트 이름

%f

현재 메일함의 완전한 패스

%F

표시된 메일의 갯수 *

%l

현재 메일함의 크기 (byte) *

%L

현재 메일의 크기 (byte) *

%m

메일함의 메일 갯수 *

%M

현재 보여지는 메일의 갯수 *

%n

메일함의 새 메일 갯수

%o

읽지 않은 오래된 메일의 갯수

%p

발송 연기된 메일의 갯수 *

%P

인덱스에서 현재 위치를 표현하는 퍼센트

%r

status-chars에 따른 수정됨/읽기 전용/쓸수 없음/첨부된 메일의 지시자.

%s

현재 정렬 방식 (sort)

%S

현재 보조 정렬 방식 (sort-aux)

%t

태그 표시된 메일의 갯수 *

%u

읽지 않은 메일의 갯수 *

%v

Mutt 의 버전

%V

현재 사용중인 제한 패턴 *

%>X

오른쪽 공백을 채울 문자열 ("X")

%|X

줄의 끝까지를 채울 문자열 ("X")

* = 0이 아닌 경우 표시될수 있다.

위 값들은 선택적으로 0이 아닌 경우에만 표시되도록 할수 있다. 다음 구조를 보라:

%?<sequence_char>?<optional_string>?

sequence_char 에는 위의 문자들이 올수 있으며, **optional_string** 에는 그 값이 0이 아닐때만 표시하려는 것을 사용한다.

다음 예는 메일함의 새 메일의 숫자를 표시하는가를 보여준다:

```
%?n?%n new messages.?
```

추가적으로 두개의 문자를 바꿀수도 있는데, 만약 첫번째것의 값이 0이고, 두번째것은 0이 아니라면 다음과 같은 형식을 사용할수 있다:

```
%?<sequence_char>?<if_string>&<else_string>?
```

결과값을 소문자로 표시하려면 시퀀스 문자에 "_" 를 붙여준다. 다음 예는 로컬 호스트 이름을 소문자로 표시한다:

```
%_h
```

6.3.213. status_on_top

유형: boolean

기본: no

"상태 창"을 화면의 첫줄에 나오게 한다.

6.3.214. strict_threads

유형: boolean

기본: no

설정될 경우 쓰레드 "sort"시 "In-Reply-To"와 "References"만을 참고한다. 기본적으로, 같은 제목을 가진 메일을 그룹으로 인식한다.

6.3.215. suspend

유형: boolean

기본: yes

unset일 경우 **susp**키 (보통 "control-Z")의 사용이 불가능 해진다. 이것은 mutt를 xterm -e mutt 과 같이 xterm안에서 실행할때 유용하다.

6.3.216. text_flowed

유형: boolean

기본: no

mutt로 하여금 text/plain; format=flowed 형식의 첨부물을 만들도록 한다. 이 포맷을 사용하기 위해서는 지원되는 편집기가 필요하다.

이 옵션이 설정될 경우 indent-string는 무시된다.

6.3.217. thread_received

Type: boolean Default: no

설정될 경우 쓰레드 메세지가 보내진 시간대신에 받은 시간을 사용한다.

6.3.218. thorough_search

유형: boolean

기본: no

"패턴"에서 설명된 **<732>b**와 **<732>h** 검색에 영향을 주는 변수로 설정될 경우 헤더와 첨부물을 디코딩한후 검색한다. unset일 경우에는 메일함에 있는 상태 그대로 검색한다.

6.3.219. tilde

유형: boolean

기본: no

yes로 설정될 경우 내장 페이저에서 빈줄의 시작에 <732>를 표시한다.

6.3.220. timeout

유형: 숫자

기본: 600

얼마 동안 키 입력이 없을 경우 새 메일을 확인할 것인지 정해준다. 값은 초 단위이다.

6.3.221. tmpdir

유형: path

기본: ""

메일을 표시/작성할때 사용되는 임시 파일들이 사용할 path.

6.3.222. to_chars

유형: 문자열

기본: " +TCFL"

인덱스 메뉴들에서 메일의 수신자에 따라 표현되는 문자를 정의한다.

첫번째 문자는 메일의 수신자가 본인이 아닌경우 (기본: 공백). 두번째는 메일이 당신에게로만 보내진 경우 (기본: +). 세번째 문자는 To: 헤더의 주소가 당신이나, 다른 사람에게도 보내진 메일 (기본: T). 네번째는 CC 헤더의 주소들 중에 당신의 주소가 있는 경우. 다섯번째는 당신이 보낸 메일. 여섯번째 문자는 가입된 메일링 리스트의 메일 (기본: L)을 각각 나타낸다.

6.3.223. tunnel

유형: 문자열

기본: ""

IMAP/POP3 서버에 접속할때 ssh 를 사용해 미리 터널을 만든다 (http://kltp.kldp.org/eunjea/ssh/ 참고).

예:

tunnel="ssh -q mailhost.net /usr/local/libexec/imapd"

주의: 이것이 제대로 작동하기 위해서는 ssh 접속시 암호를 묻지 않도록 설정되어야 한다 (인증키 사용).

6.3.224. use_8bitmime

유형: boolean

기본: no

주의**:** -B8BITMIME 옵션을 지원하는 sendmail (8.8.x 이상)에서만 사용 가능하다.

설정될 경우 mutt는 8bit 메일을 ESMTP 옵션으로 보낼때 “sendmail” 의 -B8BITMIME 플래그를 사용한다.

6.3.225. use_domain

유형: boolean

기본: yes

yes일 경우 호스명이 없는 주소가 입력되면 로컬 주소(“hostname”)를 자동으로 붙여준다.

6.3.226. use_from

유형: boolean

기본: yes

Mutt이 보내지는 메일에 ‘From:’ 헤더를 추가할 것인지 결정한다. **unset(no)**일 경우 ‘From:’ 헤더가 추가되지 않으며 이 경우 “my-hdr” 를 사용해서 직접 지정해 준다.

6.3.227. user_agent

유형: boolean

기본: yes

메일 작성시 "User-Agent" 헤더를 포함할지 결정한다.

6.3.228. visual

유형: path

기본: ""

내장 편집기에서 **<732>v**명령에 의해 사용되는 visual 편집기를 지정.

6.3.229. wait_key

유형: boolean

기본: yes

yes일 경우 mutt는 **shell-escape**, **pipe-message**, **pipe-entry**, **print-message**, **print-entry**등을 실행하고 아무 키나 입력될때 까지 기다린다.

이것은 또한 "MIME 자동보기"로 첨부물을 볼때에도 적용되며, unset 되었을 경우에는 외부 명령이 0이 아닌 값을 돌려주었을 경우에만 키 입력을 기다린다.

6.3.230. weed

유형: boolean

기본: yes

메일의 표시, 포워딩, 프린트, 답장시 헤더를 숨긴다.

6.3.231. wrap_search

유형: boolean

기본: yes

검색시 메일함의 끝에 도달하면 멈출것인지 아니면 다시 위에서부터 다시 검색할 것인지 결정한다.

6.3.232. wrapmargin

Type: number Default: 0

내장 페이저가 smart wrapping 할때 사용되는 터미날의 폭을 지정.

6.3.233. write_inc

유형: 숫자

기본: 10

메일함을 쓸때 상태창에 표시되는 진행 상황에 사용되는 값이다. 이 값은 얼마만큼의 간격을 두고 메일의 수를 표시할 것인가를 정한다.

"read-inc" 참고.

6.3.234. write_bcc

유형: boolean

기본: yes

메일을 보낼때 Bcc 헤더를 포함할 것인지 결정한다. Exim 사용자는 사용하기를 원할 것이다.

6.4. 기능 (함수)

이것은 기본 키 정의와 각 기능들의 목록이다. 각각의 키는 bind 명령으로 바꿀수 있다.

6.4.1. generic

generic 메뉴는 실제 메뉴가 아니라 **pager** 와 **editor** 에서 공통적으로 적용되는 것들이다. 이 메뉴의 키 정의를 바꾸면 모든 메뉴에 적용된다 (예외는 따로 표시).

bottom-page	L	페이지 마지막으로
current-bottom	정의 안됨	현재 항목을 페이지의 마지막으로
current-middle	정의 안됨	현재 항목을 페이지의 중간으로
current-top	정의 안됨	현재 항목을 페이지의 처음으로
enter-command	:	muttrc 명령어 입력
exit	q	현재 메뉴 나감
first-entry	=	처음 항목으로 이동
half-down	]	1/2 페이지 내리기
half-up	[	1/2 페이지 올리기
help	?	도움말 보기
jump	번호	인덱스 번호로 가기
last-entry	*	마지막 항목으로 이동
middle-page	M	페이지 중간으로 이동
next-entry	j	다음 항목으로 이동
next-line	>	한줄 내리기
next-page	z	다음 페이지로 이동
previous-entry	k	이전 항목으로 이동
previous-line	<	한줄 올리기
previous-page	Z	이전 페이지로 이동
refresh	^L	화면 정리하고 다시 그리기
search	/	정규 표현식으로 검색
search-next	n	다음 찾기
search-opposite	정의 안됨	현재 반대 방향으로 검색
search-reverse	ESC /	정규 표현식을 이용해 역순 검색
select-entry	RET	현재 항목 선택
shell-escape	!	하부 쉘 실행
tag-entry	t	현재 항목에 태그 붙이기
tag-prefix	;	태그가 붙은 항목에 명령어 적용
top-page	H	첫 페이지로 이동

6.4.2. index

bounce-message	b	메일을 다른 사람에게 보냄
change-folder	c	다른 메일함 열기
change-folder-readonly	ESC c	다른 메일함을 읽기 전용으로 열기
clear-flag	W	메일의 상태 플래그 해제
copy-message	C	메일을 파일/메일함으로 복사
create-alias	a	현재 메일의 발신인으로 부터 알리아스 만들기

decode-copy	ESC C	메일을 디코딩하여 파일/메일함으로 복사
decode-save	ESC s	메일을 디코딩하여 파일/메일함으로 저장
delete-message	d	현재 항목 지움
delete-pattern	D	패턴에 일치하는 메일 삭제
delete-subthread	ESC d	부속 글타래의 모든 메일 지움
delete-thread	^D	글타래의 모든 메일 지움
display-address	@	송신인의 주소 보기
display-toggle-weed	h	헤더 출력 토글
display-message	RET	메일 읽기
edit	e	현재 메일 편집
edit-type	^E	현재 메일의 형식 편집
exit	x	변경 사항을 저장하지 않고 나감
extract-keys	^K	PGP 공개 열쇠 추출
fetch-mail	G	POP 서버로 부터 메일 받아오기
flag-message	F	메일에 '중요' 표시
forget-passphrase	^F	메모리에서 PGP 암호 문구 지움
forward-message	f	메일 포워딩
group-reply	g	모든 수신자에게 답장
limit	l	패턴에 일치하는 메일만 표시
list-reply	L	지정된 메일링 리스트로 답장
mail	m	새 메일 작성
mail-key	ESC k	PGP 공개 열쇠 발송
next-new	TAB	다음 새 메일로 이동
next-subthread	ESC n	다음 부속 글타래로 이동
next-thread	^N	다음 글타래로 이동
next-undeleted	j	삭제되지 않은 다음 메일로 이동
next-unread	정의 안됨	다음 읽지 않은 메일로 이동
parent-message	P	글타래의 부모 메일로 이동
pipe-message	|	메세지/첨부물을 쉘 명령으로 연결
previous-new	ESC TAB	이전 새 메시지로 이동
previous-page	Z	이전 페이지로 이동
previous-subthread	ESC p	이전 부속 글타래로 이동
previous-thread	^P	이전 글타래로 이동
previous-undeleted	k	삭제되지 않은 마지막 메일로 이동
previous-unread	정의 안됨	읽지 않은 이전 메일로 이동
print-message	p	현재 항목 프린트
query	Q	주소를 얻기 위해 외부 프로그램 실행
quit	q	변경 사항 저장후 나감
read-subthread	ESC r	현재 부속 글타래에 읽은 표시하기
read-thread	^R	현재 글타래에 읽은 표시하기
recall-message	R	발송 연기한 메일 부르기
reply	r	답장 쓰기
resend-message	ESC e	현재 메세지를 MIME 형식을 유지하여 새 메세지의 템플레이트로 사용
save-message	s	메일/첨부물 파일로 저장
set-flag	w	메일의 상태 플래그 설정하기
show-version	V	Mutt의 버전과 제작일 보기
show-limit	ESC l	현재 사용 중인 범위 패턴을 보여줌
sort-mailbox	o	메일 정렬

sort-reverse	O	메일을 역순으로 정렬
sync-mailbox	$	메일함의 변경사항 저장
tag-pattern	T	패턴에 일치하는 메일에 태그 붙임
tag-thread	ESC t	현재 글타래의 모든 메일에 태그 붙임
toggle-new	N	메일의 '새글' 플래그 상태 바꿈
toggle-write	%	편지함을 다시 쓸 것인지 선택
undelete-message	u	현재 항목 복구
undelete-pattern	U	패턴에 일치하는 메일을 복구
undelete-subthread	ESC u	부속 글타래의 모든 메일 복구
undelete-thread	^U	글타래의 모든 메일 복구
untag-pattern	^T	패턴에 일차하는 메일의 태그를 해제
view-attachments	v	MIME 첨부물 보기

6.4.3. pager

bottom	정의 안됨	메일의 끝으로 이동
bounce-message	b	메일을 다른 사람에게 보냄
change-folder	c	다른 메일함 열기
change-folder-readonly	ESC c	다른 메일함을 읽기 전용으로 열기
copy-message	C	메일을 파일/메일함으로 복사
create-alias	a	현재 메일의 발신인으로 부터 알리아스 만들기
decode-copy	ESC C	메일을 디코딩하여 파일/메일함으로 복사
decode-save	ESC s	메일을 디코딩하여 파일/메일함으로 저장
delete-message	d	현재 항목 지움
delete-subthread	ESC d	부속 글타래의 모든 메일 지움
delete-thread	^D	글타래의 모든 메일 지움
display-address	@	송신인의 주소 보기
display-toggle-weed	h	헤더 출력 토글
edit	e	현재 메일 편집
edit-type	^E	현재 메일의 형식 편집
enter-command	:	muttrc 명령어 입력
exit	i	주 메뉴로 돌아감
extract-keys	^K	PGP 공개 열쇠 추출
flag-message	F	메일에 '중요' 표시
forget-passphrase	^F	메모리에서 PGP 암호 문구 지우기
forward-message	f	메일 포워딩
group-reply	g	모든 수신자에게 답장
half-up	정의 안됨	1/2 페이지 올리기
half-down	정의 안됨	1/2 페이지 내리기
help	?	도움말 보기
list-reply	L	지정된 메일링 리스트로 답장
mail	m	새 메일 작성
mail-key	ESC k	PGP 공개 열쇠 발송
mark-as-new	N	메일의 '새글' 플래그 상태 바꿈
next-line	RET	한줄 내리기
next-entry	J	다음 항목으로 이동
next-new	TAB	다음 새 메일로 이동

next-page		다음 페이지로 이동
next-subthread	ESC n	다음 부속 글타래로 이동
next-thread	^N	다음 글타래로 이동
next-undeleted	j	삭제되지 않은 다음 메일로 이동
next-unread	정의 안됨	다음 읽지 않은 메일로 이동
parent-message	P	글타래의 부모 메일로 이동
pipe-message	|	메세지/첨부물을 쉘 명령으로 연결
previous-line	BackSpace	한줄 올리기
previous-entry	K	이전 항목으로 이동
previous-new	정의 안됨	이전 새 메시지로 이동
previous-page	-	이전 페이지로 이동
previous-subthread	ESC p	이전 부속 글타래로 이동
previous-thread	^P	이전 글타래로 이동
previous-undeleted	k	삭제되지 않은 마지막 메일로 이동
previous-unread	정의 안됨	읽지 않은 이전 메일로 이동
print-message	p	현재 항목 프린트
quit	Q	변경 사항 저장후 나감
read-subthread	ESC r	현재 부속 글타래에 읽은 표시하기
read-thread	^R	현재 글타래에 읽은 표시하기
recall-message	R	발송 연기한 메일 부르기
redraw-screen	^L	화면 정리하고 다시 그리기
reply	r	답장 쓰기
save-message	s	메일/첨부물 파일로 저장
search	/	정규 표현식으로 검색
search-next	n	다음 찾기
search-opposite	정의 안됨	현재 반대 방향으로 검색
search-reverse	ESC /	정규 표현식을 이용해 역순 검색
search-toggle	\	검색 결과의 색 토글
shell-escape	!	하부 쉘 실행
show-version	V	Mutt의 버전과 제작일 보기
skip-quoted	S	인용문 무시하기
sync-mailbox	$	변경 사항 저장
tag-message	t	메일에 태그 붙이기
toggle-quoted	T	인용문 보기/숨기기
top	^	메일의 처음으로 이동
undelete-message	u	현재 항목 복구
undelete-subthread	ESC u	부속 글타래의 모든 메일 복구
undelete-thread	^U	글타래의 모든 메일 복구
view-attachments	v	MIME 첨부물 보기

6.4.4. alias

search	/	정규 표현식으로 검색
search-next	n	다음 찾기
search-reverse	ESC /	정규 표현식을 이용해 역순 검색

6.4.5. query

create-alias	a	발신인의 알리아스 만들기
mail	m	새 메일 작성
query	Q	주소를 얻기위해 외부 프로그램 사용
query-append	A	현재 결과에 새 결과를 덮붙임
search	/	정규 표현식으로 검색
search-next	n	다음 찾기
search-opposite	정의 안됨	현재 반대 방향으로 검색
search-reverse	ESC /	정규 표현식을 이용해 역순 검색

6.4.6. attach

bounce-message	b	메일을 다른 사람에게 보냄
decode-copy	ESC C	메일을 디코딩하여 파일/메일함으로 복사
decode-save	ESC s	메일을 디코딩하여 파일/메일함으로 저장
delete-entry	d	현재 항목 지움
display-toggle-weed	h	헤더 출력 토글
edit-type	^E	첨부물 내용 유형 편집
extract-keys	^K	PGP 공개 열쇠 추출
forward-message	f	메일 포워딩
group-reply	g	모든 수신자에게 답장
list-reply	L	지정된 메일링 리스트로 답장
pipe-entry	|	메세지/첨부물을 쉘 명령으로 연결
print-entry	p	현재 항목 프린트
reply	r	답장 하기
resend-message	ESC e	현재 메세지를 MIME 형식을 유지하여 새 메세지의 템플레이트로 사용
save-entry	s	메세지/첨부물 파일로 저장
undelete-entry	u	현재 항목 복구
view-attach	RET	필요하다면 mailcap 을 사용해 첨부물을 보여줌
view-mailcap	m	강제로 mailcap 을 사용하기
view-text	T	첨부물을 텍스트 형식으로 보기

6.4.7. compose

attach-file	a	첨부 파일 추가
attach-message	A	첨부 메일 추가
attach-key	ESC k	PGP 공개키 추가
copy-file	C	메일/첨부물을 파일로 저장
detach-file	D	현재 항목 삭제
display-toggle-weed	h	헤더 숨기기/보여주기
edit-bcc	b	BCC 편집
edit-cc	c	CC 편집
edit-description	d	첨부물 설명 편집
edit-encoding	^E	첨부물의 전송 인코딩 방식 수정

```
edit-fcc                    f   보내지는 메일이 저장될 파일
edit-from                ESC f   from: 편집
edit-file                 ^X e   파일을 고친 후 첨부하기
edit-headers                E   헤더를 포함한 메일 편집
edit                        e   메일 편집
edit-mime                   m   mailcap 항목을 사용하여 첨부물 편집
edit-reply-to               r   Reply-To 편집
edit-subject                s   메일 제목 편집
edit-to                     t   TO 목록 편집
edit-type                  ^T   첨부물 내용 유형 편집
filter-entry                F   쉘 명령어를 통해 첨부물 걸르기
forget-passphrase          ^F   메모리에서 PGP 암호 문구 지우기
ispell                      i   ispell로 메시지 검사
new-mime                    n   mailcap 항목을 사용해 새로운 첨부물 작성
pgp-menu                    p   PGP 옵션 보여줌
pipe-entry                  |   메세지/첨부물을 쉘 명령으로 연결하기
postpone-message            P   현재 메세지를 저장 후 나중에 보냄
print-entry                 l   현재 항목 프린트
rename-file                 R   첨부 파일 새이름/이동
send-message                y   메일 보냄
toggle-unlink               u   발신 후 파일을 지울지 선택
view-attach               RET   필요하다면 mailcap 항목을 사용해 첨부물을 보여줌
write-fcc    w   메시지를 메일함에 저장
```

6.4.8. postpone

```
delete-entry                d   메일 삭제
undelete-entry              u   삭제 취소
```

6.4.9. browser

```
change-dir                  c   디렉토리 바꾸기
check-new                 TAB   메일함에서 새 메일 확인
enter-mask                  m   파일 매스크 입력
search                      /   정규 표현식으로 검색
search-next                 n   다음 찾기
search-reverse          ESC /   정규 표현식을 이용해 역순 검색
select-new                  N   현재 디렉토리에서 새 파일 선택
sort                        o   메일 정렬
sort-reverse                O   메일 역순 정렬
toggle-mailboxes          TAB   메일함를 볼지 모든 파일을 볼지 선택
view-file               SPACE   파일 보기
subscribe                   s   현재 메일함에 subscribe 하기 (IMAP 만)
unsubscribe                 u   현재 메일함에 unsubscribe (IMAP 만)
toggle-subscribed           T   subscribe한 메일함/모든 메일함 선택 (IMAP 만)
```

6.4.10. pgp

view-name	%	키의 사용자 ID 보기
verify-key	c	PGP 공개키 확인하기

6.4.11. editor

backspace	BackSpace	커서 앞쪽의 글자 지우기
backward-char	^B	왼쪽으로 이동
backward-word	ESC b	단어의 처음으로 이동
bol	^A	처음으로 이동
buffy-cycle	Space	메일함 고르기
capitalize-word	ESC c	단어를 대문자로 시작
complete	TAB	파일 이름, 알리아스 (이메일 주소) 완성 하기
complete-query	^T	쿼리로 주소 완성하기
delete-char	^D	커서의 글자 삭제
downcase-word	ESC l	소문자로 변환
eol	^E	줄의 끝으로 이동
forward-char	^F	오른쪽으로 이동
forward-word	ESC f	단어의 끝으로 이동
history-down	정의 안됨	기억된 것에서 다음 문자
history-up	정의 안됨	기억된 것에서 이전 문자
kill-eol	^K	커서 오른쪽 라인 삭제
kill-eow	ESC d	커서 오른쪽 단어 삭제
kill-line	^U	한줄 삭제
kill-word	^W	커서 왼쪽 단어 삭제
quote-char	^V	다음에 입력되는 문자를 quote 하기
transpose-chars	정의 안됨	앞뒤 문자 바꾸기
upcase-word	ESC u	대문자로 변환

7. 기타

7.1. 감사드리는 분들

Kari Hurtta <kari.hurtta@fmi.fi>는 MIME parsing code 개발을 함께 하였다.

Mutt의 개발에 도움을 준 사람들:

```
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 <edmundo@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>,
Bj?rn 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 M?kel? <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>,
TAKIZAWA Takashi <taki@luna.email.ne.jp>,
Allain Thivillon <Allain.Thivillon@alma.fr>,
Gero Treuner <gero@faveve.uni-stuttgart.de>,
Vsevolod Volkov <vvv@lucky.net>,
Ken Weinert <kenw@ihs.com>

7.2. 이 문서에 관해서

이 문서는 SGML로 씌여졌으며, 변환에는 sgml-tools (http://www.sgmltools.org/) 패키지가 사용되었다.

이 문서의 영문 원본은 linuxdoc 형식으로 작성되었으며, docbook으로 변환/번역 하였습니다. 마지막 번역본의 마지막 버전은 : http://kldp.org/~eunjea/mutt/에 있으며, 오류, 번역되지 않은 부분은 mutt.kldp.org (http://mutt.kldp.org/)의 메일링 리스트로 보내주십시요.