Mail Client Neomutt

Christian Külker

2023-07-27

Contents

1 Introduction

`Neomutt` and `mutt` are renowned command-line clients, also referred to as **Mail User Agents (MUA)**. They allow users to send and receive emails directly from the terminal. While traditional local mail is efficiently managed by these tools, with some configuration, they can also handle IMAP (and POP3). In the past, I used tools like elm and pine, which had relatively simple user interfaces; `mutt` and `neomutt` continue in that tradition. My preference for `neomutt` over `mutt` is primarily due to its superior IMAP handling. That said, I haven't explored all the alternatives, so there might be better MUAs available. Feedback is welcome. This guide outlines the challenges faced when setting up basic mail functionalities in `neomutt` or `mutt` —functionalities that are typically available out-of-the-box in graphical interfaces.

1.1 IMAP Mail box naming

Surprisingly, renaming one's IMAP folder in the sidebar to something other than INBOX can be quite challenging. With older versions of `mutt`, it might even be impossible.

2 Dependencies

2.1 Autocrypt

`Autocrypt` requires `sqlite3` as dependency. While the configuration (`./configure`) works on Debian 9.13 Stretch and 10 Bullseye, the compilation with `--autocrypt` will stop for Debian 9.13 Stretch with an error.

```
autocrypt/db.c:272:9: warning: implicit declaration of function
    ''sqlite3_prepare_v3 [-Wimplicit-function-declaration]
     if (sqlite3_prepare_v3(AutocryptDB,
         ^~~~~~~~~~~~~~~~~~
autocrypt/db.c:281:32: error: ''SQLITE_PREPARE_PERSISTENT undeclared (first
    use
                            in this function)
                            -1, SQLITE_PREPARE_PERSISTENT, &AccountGetStmt,
                            NULL) != SQLITE_OK)
```

To successfully compile `neomutt` on Debian 9.13 Stretch, exclude the `--autocrypt` and `--sqlite` options from the `./configure` command.

2.2 gettext

To use `nls`, you need `gettext`. If it's not required, you can disable it with the `--disable-nls` flag.

```
aptitude install gettext
```

2.3 Ncurses

To utilize the `ncurses` user interface, the associated development library must be installed.

```
aptitude install libncurses-dev
```

2.4 libidn

The `stringprep.h` file requires the `libidn11-dev` package.

```
aptitude install libidn11-dev
```

2.5 gpgme

```
aptitude install libgpgme11-dev
```

2.6 SASL

```
aptitude install libsasl2-dev
```

2.7 Krb5

```
aptitude install libkrb5-dev
```

2.8 SQLite

`Autocrypt` requires `sqlite`

```
aptitude install libsqlite3-dev
```

2.9 GNU TLS

```
aptitude install gnutls-dev
```

2.10 Notmuch

```
aptitude install notmuch notmuch-mutt libnotmuch-dev
```

2.11 Hcache

While older version of `neomutt` (around 2016) had the configure option `--hcache` and this option actually shows up in `neomutt -v` it seems to be that the configure script do not want to know this option any more. However `hcache` will be used if a back-end is configured. Not all have been tested.

- gdbm
- bdb
- kyotocabinet
- lmdb (tested)
- qdbm
- rocksdb
- tdb
- tokyocabinet

Header compression: (not tested)

- lz4
- zlib
- zstd

For the Berkeley Database (current 5.3) probably do

```
aptitude install libdb-dev
```

However, it's not been tested. The tested backend was lmdb, which performed without issues.

```
aptitude install liblmdb-dev
```

3 Build

Building `neomutt` follows the conventional process for C programs: `configure`, `make`, and `make install`. Optional targets (goals) include `neomutt` and `test`. The compilation is both quick and reliable.

```
mkdir -p /srv/build
mkdir  -p /srv/src
cd  /srv/src
export V=20220429
export URL=https://github.com/neomutt/neomutt/archive/refs/tags/$V.tar.gz
wget -O neomutt-$V.tar.gz $URL
cd /srv/build
tar xvzf ../src/neomutt-$V.tar.gz
cd neomutt-$v
./configure --disable-doc --prefix=/srv/neomutt-20220429
```

Minimal Debian 9.13 Stretch configuration

```
Summary of build options:

  Version:              20220429
  Host OS:              linux-gnu
  Install prefix:       /srv/neomutt-20220429
  Compiler:              cc
  CFlags:               -g -O2 -std=c99 -D_ALL_SOURCE=1 -D_GNU_SOURCE=1 \
                        -D__EXTENSIONS__ -DNCURSES_WIDECHAR \
                        -I/srv/neomutt-20220429/include
  LDFlags:              -L/srv/neomutt-20220429/lib
  Libs:                 -lidn -lncursesw -ltinfo -lanl

  GPGME:                no
  PGP:                  yes
  SMIME:                yes
  Notmuch:              no
  Header Cache(s):
  Header Compression(s):
  Lua:                  no
```

Minimal Debian 11 Bullseye configuration_

```
Summary of build options:

```

```
  Version:               20220429
  Host OS:               linux-gnu
  Install prefix:        /srv/neomutt-20220429
  Compiler:               cc
  CFlags:                -g -O2 -std=c99 -D_ALL_SOURCE=1 -D_GNU_SOURCE=1 \
                         -D__EXTENSIONS__ -DNCURSES_WIDECHAR \
                         -I/srv/neomutt-20220429/include
  LDFlags:               -L/srv/neomutt-20220429/lib
  Libs:                  -lidn -lncursesw -ltinfo -lanl

  GPGME:                 no
  PGP:                   yes
  SMIME:                 yes
  Notmuch:               no
  Header Cache(s):
  Header Compression(s):
  Lua:                   no
```

Certain features need to be specified via `./configure` to be able to use them later. In case you would like to create a `neomutt` configuration similar to an installed version, you can use `neomutt -v` to show the used `./configure` options. Some options like `sidebar` are already enabled as default options.

The following `./configure` option have been used for Debian.

```
             Stretch Buster Bullseye Remark
             9.13    10     11
--autocypt   no              yes      Enable AutoCrypt support (requires GPGME
    )
--gnutls     yes             yes      Enable TLS support using GnuTLS
--gpgme      yes             yes      Enable GPGME support
--gss        yes             yes      Use GSSAPI authentication for IMAP
--lmdb       yes             yes      Use LMDB for the header cache
--pgp        yes             yes      PGP support
--prefix     yes             yes      Install path
--sasl       yes             yes      Use the SASL network security library
--smime      yes             yes      SMIME support
--sqlite     no              yes      Enable SQLite support
--mixmaster  yes             yes      Enable Mixmaster support
--notmuch    yes             yes      Enable Notmuch support
```

```
./configure --disable-doc --prefix=/srv/neomutt-20220429 --autocrypt
↪  --gnutls\
 --gpgme --gss --pgp --sasl --smime --sqlite --lmdb --mixmaster --notmuch\
 --disable-maintainer-mode
```

When comparing the above configurations to the Debian 11 Bullseye `neomutt` `./configure` options, it's evident that selecting the right options for `neomutt` is not straightforward. Kudos to the Debian 11 Bullseye team for their configuration choices.

```
--build=x86_64-linux-gnu --prefix=/usr {--includedir=${prefix}/include}\
 {--mandir=${prefix}/share/man} {--infodir=${prefix}/share/info}\
 --sysconfdir=/etc --localstatedir=/var --disable-option-checking\
 --disable-silent-rules {--libdir=${prefix}/lib/x86_64-linux-gnu}\
 {--libexecdir=${prefix}/lib/x86_64-linux-gnu}\
 --disable-maintainer-mode --disable-dependency-tracking\
 --mandir=/usr/share/man --libexecdir=/usr/libexec
  ↪  --with-mailpath=/var/mail\
 --gpgme --lua --notmuch --with-ui --gnutls --gss --idn --mixmaster
  ↪  --sasl\
 --tokyocabinet --sqlite --autocrypt
```

This are the summaries for the less complex configuration for Debian 9.13 Stretch.

```
Summary of build options:

  Version:               20220429
  Host OS:               linux-gnu
  Install prefix:        /srv/neomutt-20220429
  Compiler:               cc
  CFlags:                -g -O2 -std=c99 -D_ALL_SOURCE=1 -D_GNU_SOURCE=1
                         -D__EXTENSIONS__ -I/srv/neomutt-20220429/include
                         -DNCURSES_WIDECHAR -isystem /usr/include/mit-krb5
  LDFlags:               -L/srv/neomutt-20220429/lib
                         -L/usr/lib/x86_64-linux-gnu/mit-krb5 -Wl,-z,relro
                         -lgssapi_krb5 -lkrb5 -lk5crypto -lcom_err
  Libs:                  -lidn -lgnutls -lncursesw -ltinfo -lsqlite3 -lsasl2
                         -lgpgme -lanl  -L/usr/lib/x86_64-linux-gnu -lgpgme
                         -lassuan -lgpg-error -L/usr/lib/x86_64-linux-gnu
                         -lgpg-error
  GPGME:                 yes
  PGP:                   yes
  SMIME:                 yes
  Notmuch:               no
  Header Cache(s):
  Header Compression(s):
  Lua:                   no
```

Debian 11 Bullseye

```
Summary of build options:

  Version:               20220429
  Host OS:               linux-gnu
  Install prefix:        /srv/neomutt-20220429
  Compiler:               cc
  CFlags:                -g -O2 -std=c99 -D_ALL_SOURCE=1 -D_GNU_SOURCE=1
                         -D__EXTENSIONS__ -I/srv/neomutt-20220429/include
                         -DNCURSES_WIDECHAR -isystem /usr/include/mit-krb5
  LDFlags:               -L/srv/neomutt-20220429/lib
                         -L/usr/lib/x86_64-linux-gnu/mit-krb5 -Wl,-z,relro
                         -lgssapi_krb5 -lkrb5 -lk5crypto -lcom_err
  Libs:                  -llmdb -lidn -lgnutls -lncursesw -ltinfo -lsqlite3
                         -lsasl2 -lgpgme -lanl  -L/usr/lib/x86_64-linux-gnu
                         -lgpgme -lassuan -lgpg-error
                         -L/usr/lib/x86_64-linux-gnu -lgpg-error
  GPGME:                 yes
  PGP:                   yes
  SMIME:                 yes
  Notmuch:               no
  Header Cache(s):       lmdb
  Header Compression(s):
  Lua:                   no
```

4 Installation

As is standard, the `make install` command transfers files to the specified `--prefix` location. For **Debian 9.13 Stretch and 11 Bullseye**, a minimal configuration will copy the following files. Note: internationalization files are excluded from this list.

```
├──
 bin│
   └── neomutt├──
 etc│
   └── neomuttrc├──
 libexec│
   └── neomutt│
       ├── pgpewrap│
       └── smime_keys└──
 share
    ├── doc
    │   └── neomutt
    │       ├── colorschemes
```

```
14      │       │   ├── neonwolf-256.neomuttrc
15      │       │   ├── solarized-dark-256.neomuttrc
16      │       │   ├── vombatidae.neomuttrc
17      │       │   └── zenburn.neomuttrc
18      │       ├── hcache-bench
19      │       │   ├── neomutt-hcache-bench.sh
20      │       │   ├── neomuttrc
21      │       │   └── README.md
22      │       ├── keybase
23      │       │   ├── attachmentView.png
24      │       │   ├── decrypt.sh
25      │       │   ├── install.sh
26      │       │   ├── keybase.py
27      │       │   ├── LICENSE
28      │       │   ├── neomuttrc
29      │       │   ├── pagerMode.png
30      │       │   ├── pgpdecrypt.sh
31      │       │   ├── pgpverify.sh
32      │       │   ├── README.md
33      │       │   └── verify.sh
34      │       ├── logo
35      │       │   ├── neomutt-128.png
36      │       │   ├── neomutt-256.png
37      │       │   ├── neomutt-32.png
38      │       │   ├── neomutt-64.png
39      │       │   └── neomutt.svg
40      │       ├── lua
41      │       │   ├── test_lua-api_runner.neomuttrc
42      │       │   └── test_lua-api_spec.lua
43      │       ├── oauth2
44      │       │   ├── mutt_oauth2.py
45      │       │   └── mutt_oauth2.py.README
46      │       ├── samples
47      │       │   ├── colors.default
48      │       │   ├── colors.linux
49      │       │   ├── gpg.rc
50      │       │   ├── mairix_filter.pl
51      │       │   ├── markdown2html.py
52      │       │   ├── Mush.rc
53      │       │   ├── Pine.rc
54      │       │   ├── sample.mailcap
55      │       │   ├── sample.neomuttrc
56      │       │   ├── sample.neomuttrc-starter
57      │       │   ├── sample.neomuttrc-tlr
58      │       │   ├── smime_keys_test.pl
59      │       │   ├── smime.rc
60      │       │   └── Tin.rc
61      │       └── vim-keys
62      │           ├── README.md
```

```
63      │           └── vim-keys.rc
64      ...
```

5 History

Version	Date	Notes
0.1.1	2023-07-27	Improve style, grammar, release to quick guide
0.1.0	2022-07-18	Initial release

6 Disclaimer of Warranty

7 Limitation of Liability