aerc Terminal email client

START HERE

- **j** / **k** — Next / previous message
- **Ctrl** + **n** / **Ctrl** + **p** — Next / previous tab
- **Enter** — Open the message
- **?** — List keys for this view
- **C** — Compose a new message
- **r** › **r** — Reply to all
- **d** — Delete (asks first)
- **/** — Search the folder

Message list folders in the sidebar → (Enter) **Viewer** one tab per message → (r / f / C) **Composer** editor opens in a tab → (Esc, then y) **Review** send, attach, postpone

CONCEPTS

Everything is a command. Keys just run commands; press : to type one yourself, with Tab completion.

The composer runs $EDITOR in an embedded terminal tab, so editor keys go to the editor. Ctrl+x there opens the : prompt.

Binds are per view ([messages], [view], [compose]...) in binds.conf. The defaults need no file.

MOVING AROUND

j / **k**
Next / previous message

Ctrl + **d** / **Ctrl** + **u**
Half a page down / up

Ctrl + **f** / **Ctrl** + **b**
Full page down / up

g / **G** First / last message

J / **K** Next / previous folder

Ctrl + **n** / **Ctrl** + **p**
Next / previous tab

[› **t** / **]** › **t**
Previous / next tab (vim-style)

:cf *<folder>*
Jump to a folder by name

z › **z** Centre the selection

. Repeat the last command

READING

Enter
Open message in the viewer

q
Close viewer / quit from the list

In the viewer J / K step to the next / previous message. From the list, q asks first.

Ctrl + **Down** / **Ctrl** + **Up**
Next / previous message part

H Toggle all headers

o Open part with its handler

Ctrl + **l**
Open a link in the message

Ctrl + **y** Copy a link

:help keys
Open the key reference

WRITE AND REPLY

C / **m**
Compose a new message

r › **r** Reply to all

R › **r** Reply to sender only

r › **q**
Reply to all, quoting the message

R › **q**
Reply to sender, quoting the message

f Forward (in the viewer)

b Bounce to other addresses

WRITE AND REPLY (cont.)

Ctrl + **j** / **Ctrl** + **k**
Next / previous header field

Ctrl + **x**
Type a : command from the editor

:cc *<addr>*
Add a Cc recipient

Alt + **n** / **Alt** + **p**
Next / previous sending account

Ctrl + **o**
Complete an address

REVIEW AND SEND

y Send the message

Other keys on the review screen

Key	Action
n	Discard, no confirmation
p	Postpone to Drafts
e	Edit body and headers
a / d	Add / remove attachment
v	Preview
s / x	Toggle sign / encrypt

DELETE AND TAG

d Delete, asking first

D Delete immediately

a Archive the message

v
Mark message for a bulk action

DELETE AND TAG (cont.)

Space Mark and move on

V Visual mark mode

:read / :unread
Set read state

:flag
Flag a message

:move *<folder>*
Move to a folder

:copy *<folder>*
Copy to a folder

THREADS

T Toggle threaded view

z › **c** / **z** › **o**
Fold / unfold thread

z › **a**
Toggle fold of the thread

:sort -r date
Sort newest first

SEARCH AND FILTER

/ Search in the folder

n / **N** Next / previous result

**** Filter the list

Esc Clear search or filter

:filter -u
Show unread only

:filter -f *<from>*
Messages from a sender

SEARCH AND FILTER (cont.)

:filter -d *<range>*
By date, e.g. last week

:filter -b *<text>*
Search message bodies

:filter -x flagged
Flagged messages only

FOLDERS, SAVING AND PIPES

:check-mail
Fetch new mail now

S Save attachment to a path

|
Pipe message to a command

Ctrl + **t** Open a terminal tab

:attach *<path>*
Attach a file to the draft

COMMAND LINE

aerc
Start the client

aerc mailto:*<addr>*
Open a composer for an address

aerc -a *<account>*
Start on one account

:new-account
Run the account wizard

:quit
Leave aerc

CONFIG FILES

Files live in ~/.config/aerc: aerc.conf (look and behaviour), accounts.conf (credentials, keep private) and binds.conf (keys).

Use source-cred-cmd or outgoing-cred-cmd to read passwords from a password manager instead of writing them in accounts.conf.

RECIPES

Minimal IMAP account

```
# accounts.conf
[Work]
source   = imaps://me@example.com@imap.example.com
outgoing = smtps://me@example.com@smtp.example.com
default  = INBOX
from     = Me <me@example.com>
```

Add vim-style binds

```
# binds.conf
[messages]
gg = :select 0<Enter>
ga = :flag -a<Enter>
[view]
x = :close<Enter>
```

Apply a patch series

```
# mark the patches with v, then:
:pipe -m git am -3
# or: :patch apply <tag>
```

Unread from a person this week

```
:filter -u -f alice -d this week
```