

Blog (/blog/)

Posts (/posts/)

Tech (/tech/)

Kontakt (/kontakt/)

Letting mu4e reply from a catchall address

If you ever watched me register on a website, you might have noticed that I always pick the sites top level domain as the local part of my email. So, on `example.com`, I would register as `example.com@bascht.com` - that way I can make sure I have proper filters in place and in case there is spam delivered to any of those addresses, I have reasonable suspicion who dropped the ball. (Also `gopass` can defer the user name if I just abide by my own conventions).

To keep things tidy, I also want to also **reply** from `example.com@bascht.com` and this is where it gets interesting:

Sadly not all email clients (and by that, I mean almost none) are able to **send** emails from a varying address. I remember Thunderbird let me edit the `From` field in the olden days, but I don't know if it is still able to.

Since I use `mu4e` (<https://djcbssoftware.nl/code/mu/mu4e.html>) for all my email needs – with a smidgen of Emacs Lisp - it can be taught to pick the correct sender when replying.

First, let's define a function that will process the `mu4e` message (`msg`):

```
(defun bascht/mu4e-change-from-to-catchall (msg)
  "Set the From address based on the To address of the original message if I reply."
  (setq user-mail-address
    (if (and msg (mu4e-message-contact-field-matches msg :to "bascht.com"))
        (plist-get (car-safe (mu4e-message-field msg :to)) :email)
        (cdr-safe (assoc 'user-mail-address (mu4e-context-vars (mu4e-context-current)))))))
```

To set the `From` address, I found no other way than to directly modify `user-mail-address`, so in case the domain of `:to` matches `bascht.com`, I will just use it as my sender address. In any other case, I want `mu4e` just to fall back to the default address of the current context.

⚠ That last bit is important. Otherwise `mu4e` will stick to the `user-mail-address` and you will compose new emails with the wrong sender.

Let's hook into `mu4e-compose-pre-hook`:

```
(add-hook! 'mu4e-compose-pre-hook
  (bascht/mu4e-change-from-to-catchall mu4e-compose-parent-message) ;; ← our function is ca
  (spell-fu-mode)                                                    ;; parent message is th
  (bascht/switch-spellcheck "de_DE")                                 ;; message we reply to
  (evil-insert-state))
```

I left the rest of the hook in place in case you want to copy or adapt it, but it's not strictly needed.

The rest of the body will:

- Switch on `spell-fu-mode`. I don't have a global spell mode enabled since it tends to get annoying and slows down most editors. I rather opt-in.
- I switch to `de_DE` per default, as most of my emails are composed in German.
- I directly switch to `(evil-insert-state)` since I want to start typing. 🤪

And that's it. Upon composing a reply to a message, Emacs will now pick the correct address for From.

— *(Many thanks to everyone who replied to my [Question on Mastodon \(/posts/2022/11/09/anyone-with-emacs-lisp-knowhow-how-do/!\)](#))*

[Jetzt kommentieren \(https://quill.p3k.io/?dontask=1&me=https%3A%2F%2Fcommentpara.de%2F&reply=https%3A%2F%2Fbascht.com%2Ftech%2F2022%2F11%2F09%2Fletting-mu4e-reply-from-a-catchall-address%2F\)](https://quill.p3k.io/?dontask=1&me=https%3A%2F%2Fcommentpara.de%2F&reply=https%3A%2F%2Fbascht.com%2Ftech%2F2022%2F11%2F09%2Fletting-mu4e-reply-from-a-catchall-address%2F)

[mu4e-reply-from-a-catchall-address%2f\)](https://quill.p3k.io/?dontask=1&me=https%3A%2F%2Fcommentpara.de%2F&reply=https%3A%2F%2Fbascht.com%2Ftech%2F2022%2F11%2F09%2Fletting-mu4e-reply-from-a-catchall-address%2F)

Ähnliche Beiträge

- ["waylonwalker.com \(/posts/2022/01/28/waylonwalker.com/\)](#)
- ["OpenSSL 3.0 Officially Released - Phoronix \(/posts/2021/09/07/openssl-3.0-officially-released-phoronix/\)](#)
- ["What's Accenture? \(/posts/2021/04/28/whats-accenture/\)](#)
- ["Josef "Jeff" Sipek » Email vs. Tool du Jour \(/posts/2021/04/15/josef-jeff-sipek-email-vs.-tool-du-jour/\)](#)
- ["How America could fall like Rome | Boing Boing \(/posts/2021/03/28/how-america-could-fall-like-rome-boing-boing/\)](#)

Posted

November, 09th 2022

Tags

[#tech](https://bascht.com/tags/tech/) (<https://bascht.com/tags/tech/>), [#mu4e](https://bascht.com/tags/mu4e/) (<https://bascht.com/tags/mu4e/>), [#emacs](https://bascht.com/tags/emacs/) (<https://bascht.com/tags/emacs/>), [#email](https://bascht.com/tags/email/) (<https://bascht.com/tags/email/>), [#catchall](https://bascht.com/tags/catchall/) (<https://bascht.com/tags/catchall/>)

 [Tech-Archive \(/tech/\)](https://bascht.com/tech/)

Twitter: [@bascht](https://twitter.com/bascht) (<https://twitter.com/bascht>) GitHub: [@bascht](https://github.com/bascht) (<https://github.com/bascht>) Xing: [Sebastian Schulze](https://www.xing.com/profile/Sebastian_Schulze10/N52.057a44) (https://www.xing.com/profile/Sebastian_Schulze10/N52.057a44).

[Source-Code](https://github.com/bascht/bascht.com/tree/bascht.com/content/tech/2022-11-09-mu4e-reply-with-catchall-address.org) (<https://github.com/bascht/bascht.com/tree/bascht.com/content/tech/2022-11-09-mu4e-reply-with-catchall-address.org>) und [History](https://github.com/bascht/bascht.com/commits/bascht.com/content/tech/2022-11-09-mu4e-reply-with-catchall-address.org) (<https://github.com/bascht/bascht.com/commits/bascht.com/content/tech/2022-11-09-mu4e-reply-with-catchall-address.org>) dieser Seite