Procmail

David Frey

Was ist `procmail`?

`procmail` [1] ist ein vielfältiger *Mailpräprozessor*. Die typischen Anwendungsbeispiele sind:

① Mailinglist-Sortierer,

② Spam-Filter,

③ Mail-Scoring [2],

④ generelles Mail-Filter von *stdin* nach *stdout*.

Ich gehe auf das Mail-Scoring nicht weiter ein.

Installation

`procmail` wird

① entweder aus dem `.forward`-File,

② oder direkt vom MTA aus aufgerufen (falls dieser dies unterstützt).

Wird `procmail` aus dem `.forward`-File heraus aufgerufen, sieht dieses in der Regel folgendermassen aus (vom MTA abhängig):

```
|/usr/bin/procmail -tYf-
```

Recipees

Ein Regelsatz wird “Recipee” genannt. Es gibt 2 Arten von *Recipees*:

① *delivering*: Durchführung wird gestoppt, falls die *Aktion* erfolgreich durchgeführt wurde.[a]

② *non-delivering*: Durchführung geht weiter. Non-delivering recipees entstehen durch Output eines Programs resp. Filters, das in `procmail` zurückgepipt wird oder durch *nesting blocks*.

Mit dem `c`-Flag (“carbon copy”) lassen sich “delivering recipees” in “non-delivering” recipees umwandeln.

[a] in ein File geschrieben, in ein Programm gepipt oder geforwardet

Syntax

Aus `man procmailrc` [3]:

```
:0 [flags] [:[locallockfile]]
zero or more conditions
exactly one action line
```

① erste Zeile:

- Was soll angeschaut werden?
  (Body/Header, Gross/Kleinschreibung)
- optionales Locking
  optionaler Locking-Filename

② zweite Zeile und nachfolgende:
Filterbedingungen

Locking wird dann eingeschaltet, falls mehrere Procmail-Prozesse auf die gleiche Mailbox zugreifen könnten.

Flags

Die wichtigsten Flags (defaults fett):

H Header `egrep`pen.

B Body `egrep`pen.

D case-sensitive suchen.

A/a *And*. Vorherige Bedingung(en)/Rule waren erfolgreich/wurden erfolgreich durchlaufen.

E/e *Else*. Vorherige Bedingung(en)/Rule waren/wurde nicht erfolgreich durchlaufen.

h Header der Pipe/dem File oder Mail destination feeden.

b Body der Pipe/File oder mail destination feeden.

f Pipe ist ein Filter.

c Carbon-Copy (für delivering recipees)

Bei 'E' werden alle nachfolgenden 'E' Regeln übersprungen.

Conditions

Conditions können aus folgendem bestehen:

`*`	Regular-Expression
`!`	Invertierte Bedingung
`$`	Bedingung anhand der `sh`-Double-Quote-Expansion parsen und neu parsen
`?`	Exitcode des nachfolgenden Programms verwenden
`<`	Mail grösser als die angegebene Limite.
`>`	Mail kleiner als die angegebene Limite.
`var` `??`	Vergleich von *var* mit dem nachfolgenden Inhalt.
`\`	Quoting-Charakter.

Regular Expressions

`procmail` hat dieselben *Regular Expressions* wie `egrep` [4, 5] plus zusätzliche `procmail`-spezifische:

- `^`, `$` zusätzlich Match einer Newline.
- `^^` Anker für ganz am Anfang resp. ganz am Ende des Suchbereiches.
- `\<`[a] Beginn eines Wortes.
- `\>`[a] Ende eines Wortes.
- `\/` Von da an alles der `procmail`-Variable *MATCH* zuweisen.

Action-Lines

! An die angegebene Mailadresse forwarden,

| Das angegebene Programm starten (falls nötig wird eine Shell gestartet),

{ Start eines *nesting blocks* bis zum nächsten '}'.

Mailboxformate: Falls ein '.' angehängt wird, ist es ein *mh*-Folder, ansonsten ein File im traditionellen BSD-mbox-Format.

SHELLMETAS entscheiden über das "falls nötig".

Header-Munging mit `formail`

`formail` [7] kann

- Header addieren, löschen, und verändern,
- *Mail-Digests* explodieren,
- doppelte Mails anhand der Message-ID erkennen,
- Mail ins BSD-mbox-Format zwingen.

Variablen setzen

```
PATH=/usr/bin:/bin:/usr/local/bin
MAILDIR=$HOME/Mail
PROCMAIL=$HOME/.procmail
INCDIR=$PROCMAIL
LOGFILE=$PROCMAIL/maillog
VERBOSE=no
LOGABSTRACT=no
DEBUG=no
```

doppelte Mails löschen

```
# Delete duplicated messages (from procmailrc(5)):
:0 Wh: $PROCMAIL/msgid.lock
* DEBUG ?? no
|formail -D 8192 $PROCMAIL/msgid.cache
```

Vacation

```
:0
* ?test -r $HOME/.vacation.db
|vacation $LOGNAME
```

Das Directory in `MAILDIR` soll vorhanden sein.

`LOGFILE` kann mit `mailstat` statistisch ausgwertet werden.

Subject-Beautification

```
# Fix Outfuck's stupid subject lines.
:0 fhw
* Subject: *AW:
|sed -e '/Subject:/s/AW: */Re: /g'

:0 fhw
* Subject: *(WG|FW): *\/
|sed -e '/Subject:/{ s/\(WG\|FW\): *//;s/$/ (fwd)/g; }'

# Fold multiple Re: Re: lines into one
:0 fhw
* Subject: *[Rr][eE][^:]*: *([Rr][eE][^:]*: *)+ *\/
|sed -e '/Subject:/s/[Rr][eE][^:]*: \([Rr][eE][^:]*: *\)\+ /Re: /g'
```

PGP-contents-Types hinzufügen

```
# Add missing PGP-Content-Types (iX 2/1999)
:0 fBW
* "------BEGIN PGP MESSAGE------
|formail -a "Content-Type: application/pgp; format=text; x-action=encryptsign"

:0 fBW
* "------BEGIN PGP SIGNED MESSAGE------
|formail -a "Content-Type: application/pgp; format=text; x-action=sign"
```

QP/Base64 Mails nach 8bit wandeln [8]

```
# Convert from QP/Base 64 to 8bit (from procmailex(5))
:0
* ^Content-Type: *text/plain
{
  :0 fbw
  * ^Content-Transfer-Encoding: *quoted-printable
  |mimencode -u -q

    :0 Afhw
    |formail -I "Content-Transfer-Encoding: 8bit"

  :0 fbw
  * ^Content-Transfer-Encoding: *base64
  |mimencode -u -b -p

    :0 Afhw
    |formail -I "Content-Transfer-Encoding: 8bit"
}
```

Mailinglisten aufteilen

```
# Sort mailing lists
:0 :
* ^TO(owner-)?lugs@lugs\.ch
lugs

:0 :
* ^TO(owner-)?noseequel@.*
nosql

:0 :
* ^TOdict-announce@dict.org
dictd

# From Anthony Towns <aj@humbug.org.au>:
:0 H:
* ^X-Mailing-List: .*[<] *\/[^ ][^@]*
|formail -ds >> `echo $MATCH|sed -e s,/,_,g`

:0 :
* (^TO|^X-Mailing-List:.*)debian-events-de
|formail -ds >> debian-events-de
```

Anti-Spam

```
:0
* From: \/.*
* !?echo "$MATCH"|egrep -q -f $INCDIR/whitelist
{
  # Anti-Spam
  INCLUDERC=$INCDIR/tag-agis
  INCLUDERC=$INCDIR/tag
}
```

und

```
:0 :
* !^TO<eigene Adresse>
* !^TO<Aliase>
|formail -A "X-AntiSpam: Not addressed to me" >> spam
```

Leute killfilen

```
:0 :
* From: \/.*
* ?echo "$MATCH"|egrep -q -f $INCDIR/killfile
lowlife
```

Literatur

[1] Stephen R. van den Berg. *procmail(1)*.

[2] Stephen R. van den Berg. *procmailsc(1)*.

[3] Stephen R. van den Berg. *procmailrc(1)*.

[4] Free Software Foundation. *egrep(1)*.

[5] Jeffrey E. F. Friedl. *Mastering Regular Expressions*. O'Reilly & Associates, Inc., 103 Morris Street, Suite A Sebastopol, CA 95472, USA, first edition, January 1997. ISBN 1-56592-257-3.

[6] IEEE, editor. *Portable Operating System Interface (POSIX) — Part 2: Shell and Utilities (Volume 1)*. Information technology—Portable Operating System Interface (POSIX). IEEE Computer Society, 345 E. 47th St, New York, NY 10017, USA, December 1993. ISBN 1-55937-406-3.

[7] Stephen R. van den Berg. *formail(1)*.

[8] Stephen R. van den Berg. *procmailex(1)*.