Sendmail/Sendmail.cf/Пример Sendmail.cf — Urbanculture

обратно к статье «Sendmail»
Sendmail.cf

```
#
# Copyright (c) 1983, 1995 Eric P. Allman
# Copyright (c) 1988, 1993
#       The Regents of the University of California.  All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met:
# 1. Redistributions of source code must retain the above copyright
#    notice, this list of conditions and the following disclaimer.
# 2. Redistributions in binary form must reproduce the above copyright
#    notice, this list of conditions and the following disclaimer in the
#    documentation and/or other materials provided with the distribution.
# 3. All advertising materials mentioning features or use of this software
#    must display the following acknowledgement:
#       This product includes software developed by the University of
#       California, Berkeley and its contributors.
# 4. Neither the name of the University nor the names of its contributors
#    may be used to endorse or promote products derived from this software
#    without specific prior written permission.
#
# THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
# ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
# IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
# ARE DISCLAIMED.  IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
# FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
# DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
# OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
# LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
# OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
# SUCH DAMAGE.
#

##########################################################################
##########################################################################
#####
#####           SENDMAIL CONFIGURATION FILE
#####
##### built by czyborra@czyborra on Wed May 20 14:02:17 CEST 1998
##### in /etc
##### using /usr/share/sendmail/ as configuration include directory
#####
##########################################################################
##########################################################################

#####  @(#)cfhead.m4    8.9 (Berkeley) 1/18/97  #####
#####  @(#)cf.m4        8.24 (Berkeley) 8/16/95  #####
#####  $Id: sendmail.mc,v 1.1 1998/05/20 12:02:08 czyborra Exp $  #####

#####  @(#)linux.m4     8.2 (Berkeley) 8/21/93  #####


#####  @(#)local_procmail.m4    8.6 (Berkeley) 10/20/96  #####


#####  @(#)nouucp.m4    8.1 (Berkeley) 6/7/93  #####


#####  @(#)always_add_domain.m4 8.1 (Berkeley) 6/7/93  #####


#####  @(#)masquerade_envelope.m4       8.1 (Berkeley) 7/9/95  #####


#####  @(#)nocanonify.m4        8.1 (Berkeley) 6/7/93  #####


#####  @(#)nodns.m4     8.1 (Berkeley) 8/6/93  #####
```

```
#####  @(#)proto.m4     8.151 (Berkeley) 7/31/97  #####


# level 7 config file format
V7/Berkeley

##################
#   local info   #
##################

Cwlocalhost

# my official domain name
# ... define this only if sendmail cannot automatically determine your domain
#Dj$w.Foo.COM


CP.

# "Smart" relay host (may be null)
DSsmtp8:mail

# place to which unknown users should be forwarded
#Kuser user -m -a<>
#DLname_of_luser_relay

# operators that cannot be in local usernames (i.e., network indicators)
CO @ %

# a class with just dot (for identifying canonical names)
C..

# a class with just a left bracket (for identifying domain literals)
C[[

# Mailer table (overriding domains)
#Kmailertable dbm /etc/mailertable

# Domain table (adding domains)
#Kdomaintable dbm /etc/domaintable

# Generics table (mapping outgoing addresses)
#Kgenerics dbm /etc/genericstable

# Virtual user table (maps incoming users)
#Kvirtuser dbm /etc/virtusertable

# who I send unqualified names to (null means deliver locally)
DR

# who gets all local email traffic ($R has precedence for unqualified names)
DH

# dequoting map
Kdequote dequote

# class E: names that should be exposed as from this host, even if we masquerade
# class L: names that should be delivered locally, even if we have a relay
# class M: domains that should be converted to $M
#CL root
CE root

# who I masquerade as (null for no masquerading) (see also $=M)
DMcs.tu-berlin.de

# my name for error messages
DnMAILER-DAEMON


# Configuration version number
```

DZ8.8.8

```
###############
#   Options   #
###############

# strip message body to 7 bits on input?
O SevenBitInput=False

# 8-bit data handling
O EightBitMode=pass8


# wait for alias file rebuild (default units: minutes)
O AliasWait=10

# location of alias file
O AliasFile=/etc/aliases

# minimum number of free blocks on filesystem
O MinFreeBlocks=100

# maximum message size
#O MaxMessageSize=1000000

# substitution for space (blank) characters
O BlankSub=.

# avoid connecting to "expensive" mailers on initial submission?
O HoldExpensive=False

# checkpoint queue runs after every N successful deliveries
#O CheckpointInterval=10

# default delivery mode
O DeliveryMode=background

# automatically rebuild the alias database?
#O AutoRebuildAliases

# error message header/file
#O ErrorHeader=/etc/sendmail.oE

# error mode
#O ErrorMode=print

# save Unix-style "From_" lines at top of header?
#O SaveFromLine

# temporary file mode
O TempFileMode=0660

# match recipients against GECOS field?
#O MatchGECOS

# maximum hop count
#O MaxHopCount=17

# location of help file
O HelpFile=/usr/lib/sendmail.hf

# ignore dots as terminators in incoming messages?
#O IgnoreDots

# name resolver options
#O ResolverOptions=+AAONLY

# deliver MIME-encapsulated error messages?
O SendMimeErrors=True

# Forward file search path
O ForwardPath=$z/.forward.$w:$z/.forward

# open connection cache size
O ConnectionCacheSize=2

# open connection cache timeout
O ConnectionCacheTimeout=5m
```

```
# persistent host status directory
#O HostStatusDirectory=.hoststat

# single thread deliveries (requires HostStatusDirectory)?
#O SingleThreadDelivery

# use Errors-To: header?
O UseErrorsTo=False

# log level
O LogLevel=9

# send to me too, even in an alias expansion?
#O MeToo

# verify RHS in newaliases?
O CheckAliases=False

# default messages to old style headers if no special punctuation?
O OldStyleHeaders=True

# SMTP daemon options
#O DaemonPortOptions=Port=esmtp

# privacy flags
O PrivacyOptions=authwarnings

# who (if anyone) should get extra copies of error messages
#O PostMasterCopy=Postmaster

# slope of queue-only function
#O QueueFactor=600000

# queue directory
O QueueDirectory=/var/spool/mqueue

# timeouts (many of these)
#O Timeout.initial=5m
#O Timeout.connect=5m
#O Timeout.iconnect=5m
#O Timeout.helo=5m
#O Timeout.mail=10m
#O Timeout.rcpt=1h
#O Timeout.datainit=5m
#O Timeout.datablock=1h
#O Timeout.datafinal=1h
#O Timeout.rset=5m
#O Timeout.quit=2m
#O Timeout.misc=2m
#O Timeout.command=1h
#O Timeout.ident=30s
#O Timeout.fileopen=60s
O Timeout.queuereturn=5d
#O Timeout.queuereturn.normal=5d
#O Timeout.queuereturn.urgent=2d
#O Timeout.queuereturn.non-urgent=7d
O Timeout.queuewarn=4h
#O Timeout.queuewarn.normal=4h
#O Timeout.queuewarn.urgent=1h
#O Timeout.queuewarn.non-urgent=12h
#O Timeout.hoststatus=30m

# should we not prune routes in route-addr syntax addresses?
#O DontPruneRoutes

# queue up everything before forking?
O SuperSafe=True

# status file
O StatusFile=/etc/sendmail.st

# time zone handling:
#  if undefined, use system default
#  if defined but null, use TZ envariable passed in
#  if defined and non-null, use that info
#O TimeZoneSpec=

# default UID (can be username or userid:groupid)
O DefaultUser=1:1
```

```
# list of locations of user database file (null means no lookup)
#O UserDatabaseSpec=/etc/userdb

# fallback MX host
#O FallbackMXhost=fall.back.host.net

# if we are the best MX host for a site, try it directly instead of config err
#O TryNullMXList

# load average at which we just queue messages
#O QueueLA=8

# load average at which we refuse connections
#O RefuseLA=12

# maximum number of children we allow at one time
#O MaxDaemonChildren=12

# maximum number of new connections per second
#O ConnectionRateThrottle=3

# work recipient factor
#O RecipientFactor=30000

# deliver each queued job in a separate process?
#O ForkEachJob

# work class factor
#O ClassFactor=1800

# work time factor
#O RetryFactor=90000

# shall we sort the queue by hostname first?
#O QueueSortOrder=priority

# minimum time in queue before retry
#O MinQueueAge=30m

# default character set
#O DefaultCharSet=iso-8859-1

# service switch file (ignored on Solaris, Ultrix, OSF/1, others)
#O ServiceSwitchFile=/etc/service.switch

# hosts file (normally /etc/hosts)
#O HostsFile=/etc/hosts

# dialup line delay on connection failure
#O DialDelay=10s

# action to take if there are no recipients in the message
#O NoRecipientAction=add-to-undisclosed

# chrooted environment for writing to files
#O SafeFileEnvironment=/arch

# are colons OK in addresses?
#O ColonOkInAddr

# how many jobs can you process in the queue?
#O MaxQueueRunSize=10000

# shall I avoid expanding CNAMEs (violates protocols)?
#O DontExpandCnames

# SMTP initial login message (old $e macro)
O SmtpGreetingMessage=$j Sendmail $v/$Z; $b

# UNIX initial From header format (old $l macro)
O UnixFromLine=From $g  $d

# delimiter (operator) characters (old $o macro)
O OperatorChars=.:%@!^/[]+

# shall I avoid calling initgroups(3) because of high NIS costs?
#O DontInitGroups

# are group-writable :include: and .forward files (un)trustworthy?
#O UnsafeGroupWrites
```

```
# where do errors that occur when sending errors get sent?
#O DoubleBounceAddress

# what user id do we assume for the majority of the processing?
#O RunAsUser=sendmail

############################
#   Message precedences   #
############################

Pfirst-class=0
Pspecial-delivery=100
Plist=-30
Pbulk=-60
Pjunk=-100

####################
#   Trusted users   #
####################

# this is equivalent to setting class "t"
#Ft/etc/sendmail.ct
Troot
Tdaemon

########################
#   Format of headers   #
########################

H?P?Return-Path: <$g>
HReceived: $?sfrom $s $.$?_($?s$|from $.$_)
        $.by $j ($v/$Z)$?r with $r$. id $i$?u
        for $u; $|;
        $.$b
H?D?Resent-Date: $a
H?D?Date: $a
H?F?Resent-From: $?x$x <$g>$|$g$.
H?F?From: $?x$x <$g>$|$g$.
H?x?Full-Name: $x
# HPosted-Date: $a
# H?l?Received-Date: $b
H?M?Resent-Message-Id: <$t.$i@$j>
H?M?Message-Id: <$t.$i@$j>
#
##########################################################################
##########################################################################
#####
#####                   REWRITING RULES
#####
##########################################################################
##########################################################################

############################################
###  Ruleset 3 -- Name Canonicalization  ###
############################################
S3

# handle null input (translate to <@> special case)
R$@                     $@ <@>

# strip group: syntax (not inside angle brackets!) and trailing semicolon
R$*                     $: $1 <@>                       mark addresses
R$* < $* > $* <@>       $: $1 < $2 > $3                 unmark <addr>
R@ $* <@>               $: @ $1                         unmark @host:...
R$* :: $* <@>           $: $1 :: $2                     unmark node::addr
R:include: $* <@>       $: :include: $1                 unmark :include:...
R$* [ $* : $* ] <@>     $: $1 [ $2 : $3 ]               unmark IPv6 addrs
R$* : $* [ $* ]         $: $1 : $2 [ $3 ] <@>           remark if leading colon
R$* : $* <@>            $: $2                           strip colon if marked
R$* <@>                 $: $1                           unmark
R$* ;                      $1                           strip trailing semi
R$* < $* ; >               $1 < $2 >                    bogus bracketed semi

# null input now results from list:; syntax
R$@                     $@ :; <@>

# strip angle brackets -- note RFC733 heuristic to get innermost item
R$*                     $: < $1 >                       housekeeping <>
R$+ < $* >                 < $2 >                       strip excess on left
```

```
R< $* > $+                  < $1 >                      strip excess on right
R<>                         $@ < @ >                    MAIL FROM:<> case
R< $+ >                     $: $1                       remove housekeeping <>

# make sure <@a,@b,@c:user@d> syntax is easy to parse -- undone later
R@ $+ , $+                  @ $1 : $2                   change all "," to ":"

# localize and dispose of route-based addresses
R@ $+ : $+                  $@ $>96 < @$1 > : $2        handle <route-addr>

# find focus for list syntax
R $+ : $* ; @ $+            $@ $>96 $1 : $2 ; < @ $3 >  list syntax
R $+ : $* ;                 $@ $1 : $2;                 list syntax

# find focus for @ syntax addresses
R$+ @ $+                    $: $1 < @ $2 >              focus on domain
R$+ < $+ @ $+ >             $1 $2 < @ $3 >              move gaze right
R$+ < @ $+ >                $@ $>96 $1 < @ $2 >         already canonical

# do some sanity checking
R$* < @ $* : $* > $*        $1 < @ $2 $3 > $4           nix colons in addrs

# if we have % signs, take the rightmost one
R$* % $*                    $1 @ $2                     First make them all @s.
R$* @ $* @ $*               $1 % $2 @ $3                Undo all but the last.
R$* @ $*                    $@ $>96 $1 < @ $2 >         Insert < > and finish

# else we must be a local name
R$*                         $@ $>96 $1


################################################
###  Ruleset 96 -- bottom half of ruleset 3  ###
################################################

S96

# handle special cases for local names
R$* < @ localhost > $*          $: $1 < @ $j . > $2         no domain at all
R$* < @ localhost . $m > $*     $: $1 < @ $j . > $2         local domain
R$* < @ [ $+ ] > $*             $: $1 < @@ [ $2 ] > $3      mark [a.b.c.d]
R$* < @@ $=w > $*               $: $1 < @ $j . > $3         self-literal
R$* < @@ $+ > $*                $@ $1 < @ $2 > $3           canon IP addr

# look up domains in the domain table
#R$* < @ $+ > $*                $: $1 < @ $(domaintable $2 $) > $3


# pass to name server to make hostname canonical
#R$* < @ $* $~P > $*            $: $1 < @ $[ $2 $3 $] > $4

# local host aliases and pseudo-domains are always canonical
R$* < @ $=w > $*                $: $1 < @ $2 . > $3
R$* < @ $j > $*                 $: $1 < @ $j . > $2
R$* < @ $=M > $*                $: $1 < @ $2 . > $3
R$* < @ $* $=P > $*             $: $1 < @ $2 $3 . > $4
R$* < @ $* . . > $*             $1 < @ $2 . > $3


################################################
###  Ruleset 4 -- Final Output Post-rewriting  ###
################################################
S4

R$* <@>                 $@                          handle <> and list:;

# strip trailing dot off possibly canonical name
R$* < @ $+ . > $*       $1 < @ $2 > $3

# eliminate internal code -- should never get this far!
R$* < @ *LOCAL* > $*    $1 < @ $j > $2

# externalize local domain info
R$* < $+ > $*           $1 $2 $3                    defocus
R@ $+ : @ $+ : $+       @ $1 , @ $2 : $3            <route-addr> canonical
R@ $*                   $@ @ $1                     ... and exit


# delete duplicate local names
R$+ % $=w @ $=w         $1 @ $2                     u%host@host => u@host
```

```
##############################################################
###   Ruleset 97 -- recanonicalize and call ruleset zero   ###
###                (used for recursive calls)              ###
##############################################################

S97
R$*                     $: $>3 $1
R$*                     $@ $>0 $1


######################################
###   Ruleset 0 -- Parse Address   ###
######################################

S0

R$*                     $: $>Parse0 $1          initial parsing
R$*                     $: $>98 $1              handle local hacks
R$*                     $: $>Parse1 $1          final parsing

SParse0
R<@>                    $#local $: <@>          special case error msgs
R$* : $* ; <@>          $#error $@ 5.1.3 $: "list:; syntax illegal for recipient addresses"
R<@ $+>                 $#error $@ 5.1.1 $: "user address required"
R$*                     $: <> $1
R<> $* < @ [ $+ ] > $*  $1 < @ [ $2 ] > $3
R<> $* <$* : $* > $*    $#error $@ 5.1.1 $: "colon illegal in host name part"
R<> $*                  $1
R$* < @ . $* > $*       $#error $@ 5.1.2 $: "invalid host name"
R$* < @ $* .. $* > $*   $#error $@ 5.1.2 $: "invalid host name"

# handle numeric address spec
R$* < @ [ $+ ] > $*     $: $>98 $1 < @ [ $2 ] > $3      numeric internet spec
R$* < @ [ $+ ] > $*     $#smtp8 $@ [$2] $: $1 < @ [$2] > $3     still numeric: send

# now delete the local info -- note $=O to find characters that cause forwarding
R$* < @ > $*            $@ $>Parse0 $>3 $1              user@ => user
R< @ $=w . > : $*       $@ $>Parse0 $>3 $2              @here:... -> ...
R$- < @ $=w . >         $: $(dequote $1 $) < @ $2 . >   dequote "foo"@here
R< @ $+ >               $#error $@ 5.1.1 $: "user address required"
R$* $=O $* < @ $=w . >  $@ $>Parse0 $>3 $1 $2 $3        ...@here -> ...

SParse1
# handle virtual users
#R$+ < @ $=w . >        $: < $(virtuser $1 @ $2 $@ $1 $: @ $) > $1 < @ $2 . >
#R<@> $+ + $* < @ $* . >
                        $: < $(virtuser $1 + * @ $3 $@ $1 $: @ $) > $1 + $2 < @ $3 . >
#R<@> $+ + $* < @ $* . >
                        $: < $(virtuser $1 @ $3 $@ $1 $: @ $) > $1 + $2 < @ $3 . >
#R<@> $+ < @ $+ . >     $: < $(virtuser @ $2 $@ $1 $: @ $) > $1 < @ $2 . >
#R<@> $+                        $: $1
#R< error : $- $+ > $*  $#error $@ $( dequote $1 $) $: $2
#R< $+ > $+ < @ $+ >    $: $>97 $1

# short circuit local delivery so forwarded email works
#R$+ . USENET < @ $=w . >       $#usenet $: $1          handle usenet specially
R$=L < @ $=w . >        $#local $: @ $1         special local names
R$+ < @ $=w . >         $#local $: $1                   regular local name

# not local -- try mailer table lookup
#R$* <@ $+ > $*         $: < $2 > $1 < @ $2 > $3        extract host name
#R< $+ . > $*           $: < $1 > $2                    strip trailing dot
#R< $+ > $*             $: < $(mailertable $1 $) > $2   lookup
#R< $~[ : $+ > $*       $>95 < $1 : $2 > $3             check -- resolved?
#R< $+ > $*             $: $>90 <$1> $2                 try domain


# resolve fake top level domains by forwarding to other hosts



# pass names that still have a host to a smarthost (if defined)
R$* < @ $* > $*         $: $>95 < $S > $1 < @ $2 > $3   glue on smarthost name

# deal with other remote names
R$* < @$* > $*          $#smtp8 $@ $2 $: $1 < @ $2 > $3         user@host.domain
```

```
# if this is quoted, strip the quotes and try again
R$+                     $: $(dequote $1 $)              strip quotes
R$+ $=O $+              $@ $>97 $1 $2 $3                        try again

# handle locally delivered names
R$=L                    $#local $: @ $1                 special local names
R$+                     $#local $: $1                   regular local names

##########################################################################
###   Ruleset 5 -- special rewriting after aliases have been expanded   ###
##########################################################################

S5

# deal with plussed users so aliases work nicely
R$+ + *                 $#local $@ $&h $: $1
R$+ + $*                $#local $@ + $2 $: $1 + *

# prepend an empty "forward host" on the front
R$+                     $: <> $1

# send unrecognized local users to a relay host
#R< > $+                $: < $L . > $( user $1 $)       look up user
#R< $* > $+ <> $*       $: < > $2 $3                    found; strip $L
#R< $* . > $+           $: < $1 > $2                    strip extra dot

# see if we have a relay or a hub
R< > $+                 $: < $H > $1                    try hub
R< > $+                 $: < $R > $1                    try relay
R< > $+                 $: < > < $1 $(dequote "" $&h $) >       nope, restore +detail
R< > < $+ + $* > $*        < > < $1 > + $2 $3           find the user part
R< > < $+ > + $*        $#local $@ $2 $: @ $1           strip the extra +
R< > < $+ >             $@ $1                           no +detail
R$+                     $: $1 $(dequote "" $&h $)       add +detail back in
R< local : $* > $*      $: $>95 < local : $1 > $2       no host extension
R< error : $* > $*      $: $>95 < error : $1 > $2       no host extension
R< $- : $+ > $+         $: $>95 < $1 : $2 > $3 < @ $2 >
R< $+ > $+              $@ $>95 < $1 > $2 < @ $1 >

######################################################################
###  Ruleset 90 -- try domain part of mailertable entry         ###
######################################################################

S90
#R$* <$- . $+ > $*      $: $1$2 < $(mailertable .$3 $@ $1$2 $@ $2 $) > $4
#R$* <$~[ : $+ > $*             $>95 < $2 : $3 > $4     check -- resolved?
#R$* < . $+ > $*                $@ $>90 $1 . <$2> $3    no -- strip & try again
#R$* < $* > $*          $: < $(mailertable . $@ $1$2 $) > $3    try "."
#R< $~[ : $+ > $*               $>95 < $1 : $2 > $3     "." found?
#R< $* > $*                     $@ $2                   no mailertable match

######################################################################
###  Ruleset 95 -- canonify mailer:[user@]host syntax to triple ###
######################################################################

S95
R< > $*                         $@ $1                   strip off null relay
R< error : $- $+ > $*           $#error $@ $( dequote $1 $) $: $2
R< local : $* > $*              $>CanonLocal < $1 > $2
R< $- : $+ @ $+ > $*<$*>$*      $# $1 $@ $3 $: $2<@$3>  use literal user
R< $- : $+ > $*                 $# $1 $@ $2 $: $3       try qualified mailer
R< $=w > $*                     $@ $2                   delete local host
R< $+ > $*                      $#relay $@ $1 $: $2     use unqualified mailer

######################################################################
###  Ruleset CanonLocal -- canonify local: syntax               ###
######################################################################

SCanonLocal
# strip trailing dot from any host name that may appear
R< $* > $* < @ $* . >           $: < $1 > $2 < @ $3 >

# handle local: syntax -- use old user, either with or without host
R< > $* < @ $* > $*             $#local $@ $1@$2 $: $1
R< > $+                         $#local $@ $1    $: $1

# handle local:user@host syntax -- ignore host part
R< $+ @ $+ > $* < @ $* >        $: < $1 > $3 < @ $4 >

# handle local:user syntax
```

```
R< $+ > $* <@ $* > $*           $#local $@ $2@$3 $: $1
R< $+ > $*                      $#local $@ $2    $: $1

###################################################################
###  Ruleset 93 -- convert header names to masqueraded form     ###
###################################################################

S93

# handle generics database
#R$+ < @ $=G . >        $: < $1@$2 > $1 < @ $2 . > @    mark
#R$+ < @ *LOCAL* >      $: < $1@$j > $1 < @ *LOCAL* > @ mark
#R< $+ > $+ < $* > @    $: < $(generics $1 $: $) > $2 < $3 >
#R< > $+ < @ $+ >       $: < $(generics $1 $: $) > $1 < @ $2 >
#R< $* @ $* > $* < $* > $@ $>3 $1 @ $2                  found qualified
#R< $+ > $* < $* >      $: $>3 $1 @ *LOCAL*             found unqualified
#R< > $*                        $: $1                           not found

# special case the users that should be exposed
R$=E < @ *LOCAL* >      $@ $1 < @ $j . >                leave exposed
R$=E < @ $=M . >        $@ $1 < @ $2 . >
R$=E < @ $=w . >        $@ $1 < @ $2 . >

# handle domain-specific masquerading
R$* < @ $=M . > $*      $: $1 < @ $2 . @ $M > $3        convert masqueraded doms
R$* < @ $=w . > $*      $: $1 < @ $2 . @ $M > $3
R$* < @ *LOCAL* > $*    $: $1 < @ $j . @ $M > $2
R$* < @ $+ @ > $*       $: $1 < @ $2 > $3               $M is null
R$* < @ $+ @ $+ > $*    $: $1 < @ $3 . > $4             $M is not null

###################################################################
###  Ruleset 94 -- convert envelope names to masqueraded form   ###
###################################################################

S94
R$+                     $@ $>93 $1
#R$* < @ *LOCAL* > $*   $: $1 < @ $j . > $2

###################################################################
###  Ruleset 98 -- local part of ruleset zero (can be null)     ###
###################################################################

S98
#
##########################################################################
##########################################################################
#####
#####                     MAILER DEFINITIONS
#####
##########################################################################
##########################################################################


################################################
###   Local and Program Mailer specification   ###
################################################

#####  @(#)local.m4     8.23 (Berkeley) 5/31/96  #####

Mlocal,         P=/usr/bin/procmail, F=lsDFMAw5:/|@qSPfhn9, S=10/30, R=20/40,
                T=DNS/RFC822/X-Unix,
                A=procmail -Y -a $h -d $u
Mprog,          P=/bin/sh, F=lsDFMoqeu9, S=10/30, R=20/40, D=$z:/,
                T=X-Unix,
                A=sh -c $u

#
#  Envelope sender rewriting
#
S10
R<@>                    $n                      errors to mailer-daemon
R$+                     $: $>50 $1              add local domain if needed
R$*                     $: $>94 $1              do masquerading

#
#  Envelope recipient rewriting
#
S20
R$+ < @ $* >            $: $1                   strip host part
```

```
#
#  Header sender rewriting
#
S30
R<@>                    $n                      errors to mailer-daemon
R$+                     $: $>50 $1              add local domain if needed
R$*                     $: $>93 $1              do masquerading

#
#  Header recipient rewriting
#
S40
R$+                     $: $>50 $1              add local domain if needed
#R$*                    $: $>93 $1              do all-masquerading

#
#  Common code to add local domain name (only if always-add-domain)
#
S50
R$* < @ $* > $*         $@ $1 < @ $2 > $3               already fully qualified
R$+                     $@ $1 < @ *LOCAL* >             add local qualification


######################*****##############
###   PROCMAIL Mailer specification   ###
##################*****##################

#####  @(#)procmail.m4  8.6 (Berkeley) 4/30/97  #####

Mprocmail,      P=/usr/bin/procmail, F=DFMSPhnu9, S=11/31, R=21/31, T=DNS/RFC822/X-Unix,
                A=procmail -Y -m $h $f $u

####################################
###   SMTP Mailer specification   ###
####################################

#####  @(#)smtp.m4      8.33 (Berkeley) 7/9/96  #####

Msmtp,          P=[IPC], F=mDFMuX, S=11/31, R=21, E=\r\n, L=990,
                T=DNS/RFC822/SMTP,
                A=IPC $h
Mesmtp,         P=[IPC], F=mDFMuXa, S=11/31, R=21, E=\r\n, L=990,
                T=DNS/RFC822/SMTP,
                A=IPC $h
Msmtp8,         P=[IPC], F=mDFMuX8, S=11/31, R=21, E=\r\n, L=990,
                T=DNS/RFC822/SMTP,
                A=IPC $h
Mrelay,         P=[IPC], F=mDFMuXa8, S=11/31, R=61, E=\r\n, L=2040,
                T=DNS/RFC822/SMTP,
                A=IPC $h

#
#  envelope sender rewriting
#
S11
R$+                     $: $>51 $1                      sender/recipient common
R$* :; <@>              $@                              list:; special case
R$*                     $: $>61 $1                      qualify unqual'ed names
R$+                     $: $>94 $1                      do masquerading


#
#  envelope recipient rewriting --
#  also header recipient if not masquerading recipients
#
S21
R$+                     $: $>51 $1                      sender/recipient common
R$+                     $: $>61 $1                      qualify unqual'ed names


#
#  header sender and masquerading header recipient rewriting
#
S31
R$+                     $: $>51 $1                      sender/recipient common
R:; <@>                 $@                              list:; special case

# do special header rewriting
R$* <@> $*              $@ $1 <@> $2                    pass null host through
R< @ $* > $*            $@ < @ $1 > $2                  pass route-addr through
```

```
R$*                     $: $>61 $1                      qualify unqual'ed names
R$+                     $: $>93 $1                      do masquerading


#
#  convert pseudo-domain addresses to real domain addresses
#
S51

# pass <route-addr>s through
R< @ $+ > $*            $@ < @ $1 > $2                  resolve <route-addr>

# output fake domains as user%fake@relay


#
#  common sender and masquerading recipient rewriting
#
S61

R$* < @ $* > $*         $@ $1 < @ $2 > $3               already fully qualified
R$+                     $@ $1 < @ *LOCAL* >             add local qualification


#
#  relay mailer header masquerading recipient rewriting
#
S71

R$+                     $: $>61 $1
R$+                     $: $>93 $1
```