

## Fall 2005 Handout 6

### SMTP: Simple Mail Transfer Protocol

See pp. 62–64; RFCs 821, 1123, 1869, 1870, 1891, 1985, 2821. I am going to impersonate Robert De Niro.

```
1$ awk '$1 == "smtp"' /etc/services
smtp      25/tcp      mail

2$ netstat -a -f inet -P tcp | awk '2 <= NR && NR <= 4 || $1 ~ /\.smtp$/'
TCP: IPv4
  Local Address          Remote Address      Swind Send-Q Rwind Recv-Q  State
-----
  *.smtp                 *.*                 0      0 49152      0 LISTEN

3$ telnet i5.nyu.edu 25
Trying 128.122.253.152...
Connected to i5.nyu.edu.
Escape character is '^]'.
220 i5.nyu.edu ESMTP Sendmail 8.12.10+Sun/8.12.9; Wed, 30 Nov 2005 16:22:20 -0500 (EST)
MAIL FROM: <robert@DeNiro.com>
503 5.0.0 Polite people say HELO first
HELO hollywood.com                                vs. HELL; also try EHLO for ESMTP
250 i5.nyu.edu Hello i5.nyu.edu [128.122.253.152], pleased to meet you
MAIL FROM: <robert@DeNiro.com>
250 2.1.0 <robert@DeNiro.com>... Sender ok
RCPT TO: <mark.meretzky@nyu.edu>
250 2.1.5 <mark.meretzky@nyu.edu>... Recipient ok
DATA
354 Enter mail, end with "." on a line by itself
Subject: Bob says hi
I am Robert De Niro.
.
250 2.0.0 jAULMKTO018440 Message accepted for delivery
QUIT
221 2.0.0 i5.nyu.edu closing connection
Connection to i5.nyu.edu closed by foreign host.
4$
```

Lowercase **p** prints some of the headers; uppercase **P** prints all of the headers.

```

5$ mailx -N                               Don't show the table of contents.
? P8
Message 8:
From robert@DeNiro.com Wed Nov 30 16:22:27 2005
Received: from mx2.nyu.edu (MX2.NYU.EDU [128.122.109.22])
    by i5.nyu.edu (8.12.10+Sun/8.12.9) with ESMTP id jAULMQTO018443
    for <mm64@i5.nyu.edu>; Wed, 30 Nov 2005 16:22:26 -0500 (EST)
Received: from i5.nyu.edu (I5.NYU.EDU [128.122.253.152])
    by mx2.nyu.edu (8.12.10/8.12.10) with ESMTP id jAULMLS7017316
    for <mark.meretzky@nyu.edu>; Wed, 30 Nov 2005 16:22:21 -0500 (EST)
Received: from hollywood.com (i5.nyu.edu [128.122.253.152])
    by i5.nyu.edu (8.12.10+Sun/8.12.9) with SMTP id jAULMKTO018440
    for <mark.meretzky@nyu.edu>; Wed, 30 Nov 2005 16:22:20 -0500 (EST)
Date: Wed, 30 Nov 2005 16:22:20 -0500 (EST)
From: robert@DeNiro.com
Message-Id: <200511302122.jAULMKTO018440@i5.nyu.edu>
Subject: Bob says hi
Content-Length: 21
Status: R

```

I am Robert De Niro.

```

? d8
? q
Held 7 messages in /var/mail/mm64
6$

```

### Drive SMTP from an Expect script

```

1$ which expect
/opt/sfw/bin/expect

```

—On the Web at

<http://i5.nyu.edu/~mm64/x52.9547/src/sendmail/smtp.ex>

```

1 #!/opt/sfw/bin/expect --
2
3 set timeout 60 ;#default 10 seconds; -1 to wait forever; 0 no wait
4
5 spawn telnet i5.nyu.edu 25 ;#smtp port
6 #log_user 0
7 expect "ESMTP"
8
9 send "HELO hollywood.com\r"
10 expect "pleased to meet you"
11
12 send "MAIL FROM:<robert@DeNiro.com>\r"
13 expect "Sender ok"
14
15 send "RCPT TO:<mark.meretzky@nyu.edu>\r"
16 expect "Recipient ok"
17
18 send "DATA\r"
19 expect "Enter mail"
20

```

```

21 send "Subject: Bob says hi\r"
22 send "I am Robert De Niro.\r"
23 send ".\r"
24 expect "Message accepted for delivery"
25
26 #log_user 1
27 send "QUIT\r"
28 expect "closing connection"
29
30 exit 0

```

Who was logged in then?

```

1$ cd /var/adm
2$ ls -l [uw]tmpx
-rw-r--r--  1 root    bin          17856 Nov 30 16:08 utmpx
-rw-rw-r--  1 adm     adm           69192 Nov 30 16:08 wtmpx

```

—On the Web at

<http://i5.nyu.edu/~mm64/x52.9547/src/sendmail/utmpx.c>

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include <utmpx.h>
5
6 int main()
7 {
8     int status = EXIT_FAILURE; /* guilty until proven innocent */
9     const struct utmpx *p;
10
11     setutxent();
12
13     while ((p = getutxent()) != NULL) {
14         if (strcmp(p->ut_user, "mm64") == 0) {
15             char buffer[100];
16             cftime(buffer, NULL, &p->ut_tv.tv_sec);
17
18             printf("%d ", p->ut_type); /* what type of process */
19             printf("%s ", p->ut_host); /* where logged in from */
20             printf("%s ", p->ut_line); /* which terminal */
21             printf("%s\n", buffer); /* what time */
22
23             status = EXIT_SUCCESS;
24             break;
25         }
26     }
27
28     endutxent();
29     return status;
30 }

```

```
3$ gcc -o ~/bin/utmpx utmpx.c
```

```
4$ ls -l ~/bin/utmpx
```

```

5$ utmpx
7 ftcg5faculty2.edlab.its.nyu.edu pts/22 Wed Nov 30 13:33:08 EST 2005

6$ echo $?                               ISee the exit status.
0                                         EXIT_SUCCESS

7$ grep '#define.*_PROCESS' /usr/include/utmp.h
#define INIT_PROCESS 5 /* Process spawned by "init" */
#define LOGIN_PROCESS 6 /* A "getty" process waiting for login */
#define USER_PROCESS 7 /* A user process */
#define DEAD_PROCESS 8

```

### ▼ Homework 6.1: impersonate a politician

Use SMTP to send me a letter impersonating the politician of your choice, living or dead, explaining why I should vote for you. You can type it directly into **telnet**. Or write an **expect** script that will mail the letter to yourself. When you're satisfied with it, change the script to mail the letter to me.

▲

### Drive SMTP from mail

**mail** passes its **-v** option to **sendmail**.

```

1$ cal | mail -s 'This month' -v mark.meretzky@nyu.edu
mark.meretzky@nyu.edu... Connecting to [127.0.0.1] via relay...
220 i5.nyu.edu ESMTP Sendmail 8.12.10+Sun/8.12.9; Wed, 30 Nov 2005 16:22:41 -0500 (EST)
>>> EHLO i5.nyu.edu
250-i5.nyu.edu Hello localhost [127.0.0.1], pleased to meet you
250-ENHANCEDSTATUSCODES
250-PIPELINING
250-8BITMIME
250-SIZE 20000000
250-DSN
250-ETRN
250-DELIVERBY
250 HELP
>>> MAIL From:<mm64@i5.nyu.edu> SIZE=202
250 2.1.0 <mm64@i5.nyu.edu>... Sender ok
>>> RCPT To:<mark.meretzky@nyu.edu>
>>> DATA
250 2.1.5 <mark.meretzky@nyu.edu>... Recipient ok
354 Enter mail, end with "." on a line by itself
>>> .
250 2.0.0 jAULMfTO018579 Message accepted for delivery
mark.meretzky@nyu.edu... Sent (jAULMfTO018579 Message accepted for delivery)
Closing connection to [127.0.0.1]
>>> QUIT
221 2.0.0 i5.nyu.edu closing connection

2$ ps -Af | awk 'NR == 1 || /\usr\/lib\/sendmail/' at same time as above
      UID    PID  PPID  C   STIME TTY      TIME CMD
      smmosp  238    1    0   May 15 ?        0:00 /usr/lib/sendmail -Ac -q15m
      root   1918    1    0   Jul 25 ?        6:00 /usr/lib/sendmail -bd -q15m

```

**Why was De Niro's letter sent from i5.nyu.edu to smtp.nyu.edu and back again?**

Handout 5, p. 16:

```
1$ nslookup
spawn nslookup
Default Server:  NYUNSB.NYU.EDU
Address:  128.122.253.37

> set type=mx
> i5.nyu.edu
Server:  NYUNSB.NYU.EDU
Address:  128.122.253.37

i5.nyu.edu  preference = 20, mail exchanger = SMTP.nyu.edu
nyu.edu nameserver = NS1.nyu.edu
nyu.edu nameserver = NS2.nyu.edu
nyu.edu nameserver = NYU-NS.BERKELEY.edu
nyu.edu nameserver = LAPIETRA.NYU.FLORENCE.IT
SMTP.nyu.edu internet address = 128.122.109.18
NS1.nyu.edu internet address = 128.122.253.83
NS2.nyu.edu internet address = 128.122.253.42
NYU-NS.BERKELEY.edu internet address = 128.32.222.90
> exit
```

**Where do the mail headers come from?**

RFC 822 defines the mail headers **Received:**, **Date:**, **From:**, **Message-Id:**.

The **H** command defines a header. See pp. 296, 304–305. Because of the flag **?D?** (p. 304), every mailer program with the flag **D** will add the following header to the mail it sends:

```
1$ grep '^H.*Date:' /etc/mail/sendmail.cf
H?D?Resent-Date: $a
H?D?Date: $a
```

The **\$a** is an internal (predefined) macro holding the date and time the mail was sent. See p. 299 for macros, pp. 638–640 for the internal ones.

How did the **H** line get into the 1715-line file `/etc/mail/sendmail.cf`? A human being writes the 28-line file `/usr/lib/mail/cf/i5.nyu.mc`. Then

- (1) Line 1 of the file `/usr/lib/mail/cf/i5.nyu.mc` includes the file `/usr/lib/mail/m4/cf.m4`.
- (2) Line 29 of the file `/usr/lib/mail/m4/cf.m4` includes the file `/usr/lib/mail/m4/cfhead.m4`.
- (3) Line 260 of the file `/usr/lib/mail/m4/cfhead.m4` includes the file `/usr/lib/mail/m4/proto.m4`, which contains lines that are ultimately copied into `/etc/mail/sendmail.cf`.

```
2$ grep '^H.*Date:' /usr/lib/mail/m4/proto.m4
H?D?Resent-Date: $a
H?D?Date: $a
```

If you don't know the names or the directories of the files to search, search them all:

```
3$ cd /usr/lib/mail
4$ grep 'Date:' `find . -type f 2> /dev/null` | more
```

### ▼ Homework 6.2: read the H command for the Received: header

Find the **H** command for the **Received:** header in the file `/etc/mail/sendmail.cf`. (It occupies six consecutive lines in that file.) You can even trace its origins to the **H** command in `/usr/lib/mail/m4/cf.m4` and the `CONFRECEIVED_HEADER` (p. 615) parameter in `/usr/lib/mail/m4/cfhead.m4`.

To decipher the **Received:** header, you'll need the internal macros on pp. 638–639, starting with the macro `$$s`, for “hostname of sender’s machine”. See p. 300 for

```

$?a ... $.           if macro $a is defined, then ....
$?a ... $| ... $.   if macro $a is defined, then ... else ....

```

which can be nested.

▲

### sendmail

**sendmail** can receive a letter from several sources, including

- (1) from the same host via the programs `/bin/mail`, `/bin/mailx`, or `/usr/local/bin/pine`;
- (2) from another host via SMTP: Simple Mail Transfer Protocol, RFC 2821, TCP port 25.

**sendmail** can send a letter to several destinations, including

- (1) to the same host via the program `/usr/lib/mail.local`;
- (2) to another host via SMTP.

**sendmail -bd** is a MTA (Mail Transfer Agent); **sendmail -Ac** is a MSA (Mail Submission Agent).

```

1$ ps -Af | awk 'NR == 1 || $2 <= 1 || /\usr\lib\sendmail/'
      UID  PID  PPID  C   STIME TTY      TIME CMD
      root    0    0  0   May 15 ?        0:04 sched
      root    1    0  0   May 15 ?        58:07 /etc/init -v
      smmsp  238    1  0   May 15 ?        0:00 /usr/lib/sendmail -Ac -q15m
      root  1918    1  0   Jul 25 ?        6:00 /usr/lib/sendmail -bd -q15m

```

```

2$ awk -F: '$1 == "smmsp"' /etc/passwd
smmsp:x:25:25:SendMail Message Submission Program:/:

```

```

3$ awk -F: '$3 == 25' /etc/group
smmsp::25:smmsp

```

**init** launches **S88sendmail** at runlevel 2 with the argument **start**. See Handout 4, p. 2, line 348.

```

4$ cd /etc
5$ ls -li init.d/sendmail
      1274 -rwxr--r--  5 root      sys           3202 Apr  6  2002 init.d/sendmail

```

```

6$ ls -li `find . -inum 1274 -print`
1274 -rwxr--r--  5 root      sys           3202 Apr  6  2002 ./init.d/sendmail
1274 -rwxr--r--  5 root      sys           3202 Apr  6  2002 ./rc0.d/K36sendmail
1274 -rwxr--r--  5 root      sys           3202 Apr  6  2002 ./rc1.d/K36sendmail
1274 -rwxr--r--  5 root      sys           3202 Apr  6  2002 ./rc2.d/S88sendmail
1274 -rwxr--r--  5 root      sys           3202 Apr  6  2002 ./rcS.d/K36sendmail

```

The **sendmail** options **-bd** (“be a daemon”), **q15m** (“queue interval 15 minutes”), and **-Ac** come from `/etc/init/rc2.d/S88sendmail`.

—The file `/etc/init.d/sendmail` on `i5.nyu.edu`

```

1 #!/sbin/sh
2 #
3 # Copyright (c) 1992 - 2001 by Sun Microsystems, Inc.
4 # All rights reserved.
5 #
6 #ident    "@(#)sendmail    1.1901/12/05 SMI"
7
8 ERRMSG1='WARNING: /var/mail is NFS-mounted without setting actimeo=0,'
9 ERRMSG2='this can cause mailbox locking and access problems.'
10 SERVER_PID_FILE="/var/run/sendmail.pid"
11 CLIENT_PID_FILE="/var/spool/clientmqueue/sm-client.pid"
12 DEFAULT_FILE="/etc/default/sendmail"
13 ALIASES_FILE="/etc/mail/aliases"
14
15 check_queue_interval_syntax()
16 {
17     default="15m"
18     if [ $# -lt 1 ]; then
19         answer=$default
20         return
21     fi
22     if echo $1 | egrep '^([0-9]*[1-9][0-9]*[smhdw])+$' >/dev/null 2>&1; then
23         answer=$1
24     else
25         answer=$default
26     fi
27 }
28
29 case "$1" in
30 'restart')
31     [ -f $SERVER_PID_FILE ] && kill -1 `head -1 $SERVER_PID_FILE`
32     [ -f $CLIENT_PID_FILE ] && kill -1 `head -1 $CLIENT_PID_FILE`
33     ;;
34
35 'start')
36     if [ -f /usr/lib/sendmail -a -f /etc/mail/sendmail.cf ]; then
37         if [ ! -d /var/spool/mqueue ]; then
38             /usr/bin/mkdir -m 0750 /var/spool/mqueue
39             /usr/bin/chown root:bin /var/spool/mqueue
40         fi
41         if [ ! -f $ALIASES_FILE.db ] && [ ! -f $ALIASES_FILE.dir ] \
42             && [ ! -f $ALIASES_FILE.pag ]; then
43             /usr/sbin/newaliases
44         fi
45         MODE="-bd"
46         [ -f $DEFAULT_FILE ] && . $DEFAULT_FILE
47         #
48         # * MODE should be "-bd" or null (MODE= or MODE="") or
49         #   left alone. Anything else and you're on your own.
50         # * QUEUEOPTION should be "p" or null (as above).
51         # * [CLIENT]QUEUEINTERVAL should be set to some legal value;
52         #   sanity checks are done below.
53         # * [CLIENT]OPTIONS are catch-alls; set with care.

```

```

54     #
55     if [ -n "$QUEUEOPTION" -a "$QUEUEOPTION" != "p" ]; then
56         QUEUEOPTION=""
57     fi
58     if [ -z "$QUEUEOPTION" -o -n "$QUEUEINTERVAL" ]; then
59         check_queue_interval_syntax $QUEUEINTERVAL
60         QUEUEINTERVAL=$answer
61     fi
62     check_queue_interval_syntax $CLIENTQUEUEINTERVAL
63     CLIENTQUEUEINTERVAL=$answer
64     /usr/lib/sendmail $MODE -q$QUEUEOPTION$QUEUEINTERVAL $OPTIONS &
65     /usr/lib/sendmail -Ac -q$CLIENTQUEUEINTERVAL $CLIENTOPTIONS &
66     #
67     # ETRN_HOSTS should be of the form
68     # "s1:c1.1,c1.2      s2:c2.1 s3:c3.1,c3.2,c3.3"
69     # i.e., white-space separated groups of server:client where
70     # client can be one or more comma-separated names; N.B. that
71     # the :client part is optional; see etrn(1M) for details.
72     # server is the name of the server to prod; a mail queue run
73     # is requested for each client name. This is comparable to
74     # running "/usr/lib/sendmail -qRclient" on the host server.
75     #
76     # See RFC 1985 for more information.
77     #
78     for i in $ETRN_HOSTS; do
79         SERVER=`echo $i | /usr/bin/sed -e 's/:.*$//'`
80         CLIENTS=`echo $i | /usr/bin/sed -n -e 's/,/ /g' \
81             -e ':/s/^.*/p'`
82         /usr/sbin/etrn $SERVER $CLIENTS >/dev/null 2>&1 &
83     done
84 fi
85
86 if /usr/bin/nawk 'BEGIN{s = 1}
87     $2 == "/var/mail" && $3 == "nfs" && $4 !~ /actimeo=0/ &&
88     $4 !~ /noac/{s = 0} END{exit s}' /etc/mnttab; then
89
90     /usr/bin/logger -p mail.crit "$ERRMSG1"
91     /usr/bin/logger -p mail.crit "$ERRMSG2"
92 fi
93 ;;
94
95 'stop')
96     [ -f $SERVER_PID_FILE ] && kill `head -1 $SERVER_PID_FILE`
97     if [ -f $CLIENT_PID_FILE ]; then
98         kill `head -1 $CLIENT_PID_FILE`
99         rm -f $CLIENT_PID_FILE
100 fi
101 /usr/bin/pkill -x -u 0 sendmail
102 ;;
103
104 *)
105     echo "Usage: $0 { start | stop | restart }"
106     exit 1
107 ;;

```



```
108 esac
109 exit 0
```

The regular expression in the above line 22 makes sure that the queue interval has at least one non-zero digit. If the mail queue directory `/var/spool/mqueue` does not exist, lines 37–40 will create it.

```
7$ cd /var/spool
8$ ls -ld mqueue
drwxr-x---  2 root  bin          19456 Nov 30 16:22 mqueue
```

There are two possible binary databases of mail aliases in `/etc/mail`, `aliases.db` (from <http://www.sleepycat.com/>) and the pair of files `aliases.dir` and `aliases.pag` (see `ndbm(3C)`). If neither database exists, the above lines 41–44 will create them.

```
9$ cd /etc/mail
10$ ls -l aliases.@(db|dir|pag)
-rw-r-----  1 root  smmsp      40960 Jul 26 15:33 aliases.db
```

The file `/etc/default/sendmail` in line 46 is run with a dot so it could create environment variables (Handout 4, p. 2), but it does not exist.

The `etrn` program in the above lines 66–83 sends the SMTP (Simple Mail Transfer Protocol) command `ETRN` (“extended turn”) to the servers listed in `$ETRN_HOSTS`, prodding them for on-demand delivery of mail.

The above lines 86–92 do nothing, since the directory `/var/mail` is not mounted as a Network File System directory in the file `/etc/mnttab`. `actimeo=0` turns off NFS caching; see p. 246 or `mount_nfs(1M)`.

```
11$ awk '$3 == "ufs" && ++n <= 2 || $3 == "nfs"' /etc/mnttab
/dev/md/dsk/d0 / ufs rw,intr,largefiles,logging,xattr,onerror=panic,suid,dev=1540000
/dev/md/dsk/d6 /usr ufs rw,intr,largefiles,logging,xattr,onerror=panic,suid,dev=1540
```

```
12$ awk '$1 == "sendmailvars:"' /etc/nsswitch.conf
sendmailvars:  files
```

—The file `/etc/mail/aliases` on `i5.nyu.edu`

```
1 # Copyright (c) 1983-2001 by Sun Microsystems, Inc.
2 # All rights reserved.
3 #
4 #ident  "@(#)aliases 1.1401/08/27 SMI"
5
6 ##
7 # Aliases can have any mix of upper and lower case on the left-hand side,
8 # but the right-hand side should be proper case (usually lower)
9 #
10 # >>>>>>>>> The program "newaliases" must be run after
11 # >> NOTE >> this file is updated for any changes to
12 # >>>>>>>>> show through to sendmail.
13 ##
14
15 # The following alias is required by the mail protocol, RFC 2821
16 # Set it to the address of a HUMAN who deals with this system's mail problems.
17 postmaster: root
18
19 # Alias for mailer daemon; returned messages from our MAILER-DAEMON
20 # should be routed to our local Postmaster.
21 MAILER-DAEMON: postmaster
```

```

22
23 # General redirections for pseudo accounts.
24 bin:      root
25 daemon:   root
26 system:   root
27 toor:     root
28 uucp:     root
29
30 # Well-known aliases.
31 manager:  root
32 dumper:   root
33 operator: root
34
35 # trap decode to catch security attacks
36 decode:   root
37
38 # Aliases to handle mail to programs or files, eg news or vacation
39 nobody: /dev/null
40
41 # Sample aliases:
42
43 # Alias for distribution list, members specified here:
44 #staff:wnj,mosher,sam,ecc,mckusick,sklower,olson,rwh@ernie
45
46 # Alias for distribution list, members specified elsewhere:
47 #keyboards: :include:/usr/jfarrell/keyboards.list
48
49 # Alias for a person, so they can receive mail by several names:
50 #epa:eric
51
52 #####
53 # Local aliases below #
54 #####
55 root:      esys@nyu.edu
56 SysAdmin:  root
57 sys:       root
58 adm:       root
59 #####
60 #
61 konvit:    gk31@nyu.edu
62 kraskey:   alison@nyu.edu
63 comment:   its.es.sysmgrs@nyu.edu,puskar
64 hc30:      nobody
65 dsg221:    nobody

```

### The /etc/mail/sendmail.cf file

Each command in the **sendmail** begins with one uppercase letter (pp. 298–299).

To define a user-defined **sendmail** macro, see pp. 299–300. The predefined **sendmail** macros are in pp. 638–639.

```
1$ grep -n '^D' /etc/mail/sendmail.cf
91:DSSmtp.nyu.edu      server for all outgoing mail; see SMART_HOST macro in pp. 632, 638
138:DnMAILER-DAEMON   name in which error messages are sent, p. 638
150:DZ8.12.9          configuration version number, p. 639
```

A `sendmail` class is a list of words (pp. 300–302). The ones with predefined meanings are in pp. 640–641.

```
2$ grep -n '^[CF]' /etc/mail/sendmail.cf
80:Cwlocalhost
82:Fw/etc/mail/local-host-names
88:CP.                pseudo-domains: pp. 301–302, 320, 641
95:CO @ %             operators that cannot be part of username: pp. 301–302, 641
98:C..
101:C[[
104:C{Accept}OK RELAY
109:C{ResOk}OKR
113:FR-o /etc/mail/relay-domains
120:C{tls}VERIFY ENCR Transport Layer Security connection types (RFC 2246)
141:CPREDIRECT
1346:C{src}E F D U    names of rulesets
```

A declaration for a *mailer* starts with **M** and continues onto the lines indented with tabs:

```
3$ awk '
{
    if ($0 ~ /^M(local|prog|smtp),/) {
        m = 1
    } else if ($0 !~ /^\t/) {
        m = 0
    }
}

m != 0 {print NR ":" $0}
' /etc/mail/sendmail.cf | more
1628:Mlocal, P=/usr/lib/mail.local, F=lsDFMAw5:/|@qfSmn9, S=EnvFromL/HdrFromL, R=EnvToL/HdrToL,
1629:    T=DNS/RFC822/X-Unix,
1630:    A=mail.local -d $u
1631:Mprog, P=/bin/sh, F=lsDFMoqeu9, S=EnvFromL/HdrFromL, R=EnvToL/HdrToL, D=$z:/,
1632:    T=X-Unix/X-Unix/X-Unix,
1633:    A=sh -c $u
1699:Msmtpt, P=[IPC], F=mDFMuX, S=EnvFromSMTP/HdrFromSMTP, R=EnvToSMTP, E=\r\n, L=990,
1700:    T=DNS/RFC822/SMTP,
1701:    A=TCP $h
```

### The m4 macro preprocessor

We have two versions of `m4`:

```
1$ find / -type f -name m4 2> /dev/null
/usr/ccs/bin/m4
/usr/xpg4/bin/m4
```

The one used by `/usr/lib/mail/cf/Makefile` is `/usr/ccs/bin/m4` (“C Compilation System”). Put a back quote and a single quote around each quoted word.

```

2$ cd ~mm64/public_html/x52.9547/src/sendmail
3$ cat -n smoky.m4
   1  define(`Old_Smokey', `spaghetti')
   2  define(`snow', `cheese')
   3  define(`true_lover', `meatball')
   4  define(`For_courting_too_slow', `When somebody sneezed')
   5  On top of Old_Smokey
   6  All covered with snow,
   7  I lost my true_lover
   8  For_courting_too_slow.

4$ /usr/ccs/bin/m4 smoky.m4 | cat -n
   1
   2
   3
   4
   5  On top of spaghetti
   6  All covered with cheese,
   7  I lost my meatball
   8  When somebody sneezed.

```

#### The `dnl` macro in `m4`

The `dnl` macro (lowercase DNL) deletes all characters up to the next newline:

```

1$ cd ~mm64/public_html/x52.9547/src/sendmail
2$ cat -n dnl.m4
   1  define(`Old_Smokey', `spaghetti')dnl
   2  define(`snow', `cheese')dnl
   3  define(`true_lover', `meatball')dnl
   4  define(`For_courting_too_slow', `When somebody sneezed')dnl
   5  On top of Old_Smokey
   6  All covered with snow,
   7  I lost my true_lover
   8  For_courting_too_slow.

3$ /usr/ccs/bin/m4 dnl.m4 | cat -n
   1  On top of spaghetti
   2  All covered with cheese,
   3  I lost my meatball
   4  When somebody sneezed.

```

#### Quotes in `m4`

Quotes prevent macro expansion:

```

1$ cd ~mm64/public_html/x52.9547/src/sendmail
2$ cat -n quotes.m4
   1  define(N, 10)dnl
   2  The value of `N' is N.

3$ /usr/ccs/bin/m4 quotes.m4
   1  The value of N is 10.

```

Quote the first argument of the `define` macro to allow redefinition:

```

4$ cat -n redefin1.m4
   1  define(N, 10)dnl
   2  define(N, 20)dnl
   3  The value of 'N' is N.

5$ /usr/ccs/bin/m4 redefin1.m4
/usr/ccs/bin/m4:reddefin1.m4:2 bad macro name
define(10,20)

6$ cat -n redefin2.m4
   1  define(N, 10)dnl
   2  define('N', 20)dnl
   3  The value of 'N' is N.

7$ /usr/ccs/bin/m4 redefin2.m4
The value of N is 20.

```

Quote the second argument of the **define** macro to prevent expansion:

```

8$ cat -n chain1.m4
   1  define(N, 10)dnl
   2  define(M, N)dnl
   3  define('N', 20)dnl
   4  The value of 'M' is M.

9$ /usr/ccs/bin/m4 chain1.m4
The value of M is 10.

10$ cat -n chain2.m4
   1  define(N, 10)dnl
   2  define(M, 'N')dnl
   3  define('N', 20)dnl
   4  The value of 'M' is M.

11$ /usr/ccs/bin/m4 chain2.m4
The value of M is 20.

```

We will always quote both arguments of the **define** macro.

### The .mc file for our sendmail MTA (Message Transfer Agent)

When run as an MTA, **sendmail** reads the configuration file `/etc/mail/sendmail.cf` created from `/usr/lib/mail/cf/i5.nyu.mc`.

—The file `/usr/lib/mail/cf/i5.nyu.mc` on `i5.nyu.edu`

```

1 include('../m4/cf.m4')
2 VERSIONID('i5.nyu.edu')
3 OSTYPE(solaris2)
4 define('ALIAS_FILE', '/etc/mail/aliases')dnl
5 define('STATUS_FILE', '/etc/mail/statistics')dnl
6 define('confDEF_USER_ID', 600:600)dnl
7 define('confQUEUE_LA', 3)dnl
8 define('confREFUSE_LA', 3)dnl
9 define('confTO_QUEUEWARN', 8h)dnl
10 define('confMAX_DAEMON_CHILDREN', 64)dnl
11 define('confCONNECTION_RATE_THROTTLE', 16)dnl

```

```

12 define(`confPRIVACY_FLAGS',`goaway')dnl
13 define(`confDONT_PROBE_INTERFACES',`TRUE')dnl
14 define(`confMAX_RCPTS_PER_MESSAGE',`100')dnl
15 define(`confMAX_MESSAGE_SIZE',`2000000')dnl
16 define(`SMART_HOST',`smtp.nyu.edu')dnl
17 FEATURE(always_add_domain)dnl
18 FEATURE(nouucp,`reject')dnl
19 FEATURE(redirect)dnl
20 FEATURE(use_cw_file)dnl
21 FEATURE(mailertable,`dbm -o /etc/mail/mailertable')dnl
22 FEATURE(`access_db',`dbm -T<TMPF> -o /etc/mail/access')dnl
23 FEATURE(`no_default_msa')dnl
24 MAILER(local)dnl
25 MAILER(smtp)dnl
26 MAILER_DEFINITIONS
27 LOCAL_NET_CONFIG
28 R$* < @ $* .nyu.edu. > $*      $#esmtplib $@ $2.nyu.edu $: $1 < @ $2.nyu.edu > $3

```

The macro `OSTYPE` in the above line 3 includes one of the files in the directory

```

1$ cd /usr/lib/mail/ostype
2$ ls -l | more
total 8
-r--r--r--  1 root    mail      999 Apr  6  2002 solaris2.m4
-r--r--r--  1 root    mail     1008 Apr  6  2002 solaris2.ml.m4
-r--r--r--  1 root    mail     937 Apr  6  2002 solaris2.pre5.m4
-r--r--r--  1 root    mail     836 Apr  6  2002 solaris8.m4

```

Similarly, the macro `FEATURE` in the above lines 18–23 includes one of the files in the directory

```

3$ cd /usr/lib/mail/feature
4$ ls -l always_add_domain.m4
-r--r--r--  1 root    mail      743 Apr  6  2002 always_add_domain.m4

```

And the `MAILER` macro in the above lines 24–25 includes one of the files in the directory

```

5$ cd /usr/lib/mail/mailler
6$ ls -l | more
total 32
-r--r--r--  1 root    mail     3646 Apr  6  2002 local.m4
-r--r--r--  1 root    mail     5728 Apr  6  2002 smtp.m4
-r--r--r--  1 root    mail     5578 Apr  6  2002 uucp.m4

```

Each `define` in the file `/usr/lib/mail/cf/i5.nyu.mc` becomes a statement in the file `/etc/mail/sendmail.cf`. For example, the `ALIAS_FILE` macro in the above line 4 becomes the `O` (“option”) statement

```

7$ cat -n /etc/mail/sendmail.cf | sed -n 166,167p
166  # location of alias file
167  O AliasFile=/etc/mail/aliases

```

The default user id (p. 619) in the above line 6 is

```

8$ awk -F: '$3 == 600' /etc/passwd
mailnull:x:600:600:mail account:/home/mailnull:/bin/false

```

The whitespace before the `$#esmtplib` in line 28 is a tab.

**Sendmail features in /usr/lib/mail/cf/i5.nyu.mc**

Line 17: **always\_add\_domain** (pp. 626, 635): add **@i5.nyu.edu** to mail addressed to a local loginname. If I send a letter to a local person and a remote person, each will receive a copy with the names of both recipients. In both copies, the local name will have a **@i5.nyu.edu**, so the remote person will know the full email address of the local person.

```
1$ cat -n /etc/mail/sendmail.cf | sed -n 1620,1626p
1620 #
1621 # Common code to add local domain name (only if always-add-domain)
1622 #
1623 SAddDomain
1624 R$* < @ $* > $*      $@ $1 < @ $2 > $3      already fully qualified
1625
1626 R$+                  $@ $1 < @ *LOCAL* >      add local qualification
```

Line 18: **nouucp** (pp. 291–292, 625). No one uses the Unix-to-Unix Copy Protocol any more. See the typo in line 18?

Line 19: **redirect** (pp. 625–627). If the **/etc/mail/aliases** file contains

```
senior: alumnus@bigbiz.com.REDIRECT
```

then mail sent to **senior@i5.nyu.edu** will be returned to the sender with a message telling them to send it to **alumnus@bigbiz.com**.

Line 20: **use\_cw\_file** (p. 625). A **sendmail class** is a list of names (p. 300). For example, the internal (predefined) class **w** is all the names of this host, starting with **localhost** (pp. 301, 640). Our **sendmail** will accept mail addressed to any host on this list. Without the **use\_cw\_feature**, however, this list would be only the single word **localhost**. In fact, on **i5.nyu.edu** it *is* only the single word **localhost**.

```
2$ grep '^[CF]w' /etc/mail/sendmail.cf
Cwlocalhost
Fw/etc/mail/local-host-names

3$ cd /etc/mail
4$ ls -l local-host-names
-rw-r--r--  1 root    bin          0 Apr  6  2002 local-host-names
```

The origin of the **C** and **F** commands in **/etc/mail/sendmail.cf** can be traced back to

```
5$ cat -n /usr/lib/mail/m4/proto.m4 | sed -n 109,113p
109 Cwlocalhost
110 ifdef('USE_CW_FILE',
111 `# file containing names of hosts for which we receive email
112 Fw`'confCW_FILE',
113     `dnl`)
```

Even though the macro **USE\_CW\_FILE** is empty, it still counts as being defined:

```
6$ cat -n /usr/lib/mail/feature/use_cw_file.m4 | sed -n 20,24p
20 # if defined, the sendmail.cf will read the /etc/mail/local-host-names file
21 # to find alternate names for this host. Typically only used when several
22 # hosts have been squashed into one another at high speed.
23
24 define('USE_CW_FILE', `')
```

See p. 615 for the **sendmail** macro **confCW\_FILE**.

```
7$ cat -n /usr/lib/mail/m4/cfhead.m4 | sed -n 304p
304 define('confCW_FILE', 'MAIL_SETTINGS_DIR''local-host-names')
```

Line 21: `mailertable` (pp. 626–627). We don't have the data base manager `dbm` and we don't have `/etc/mail/mailertable`. But it would have intercepted mail for one host and sent it to another.

Line 22: `access_db` (pp. 626, 634). We don't have the data base manager `dbm` and we don't have `/etc/mail/access`. It would have prevented us from receiving mail from known spammers, or relaying mail from spammers to other hosts.

```
128.122.253.191 OKAY
128.122.253.192 REJECT          generic error message
128.122.253.193 DISCARD       no error message
128.122.253.194 550 no spam accepted RFC 821 code number
hotsex@          550 no spam accepted RFC 821 code number
```

Line 23: `no_default_msa` (pp. 625). Let the `DAEMON_OPTIONS` macro (p. 612) override default MSA specs.

### The .mc file for our sendmail MSA (Message Submission Agent)

The command line option `-Ac` tells `sendmail` to read the configuration file `/etc/mail/submit.cf` created from `/usr/lib/mail/cf/submit.mc`,

—The file `/usr/lib/mail/cf/submit.mc` on `i5.nyu.edu`

```
1 divert(-1)
2 #
3 # Copyright (c) 2001, 2002 Sendmail, Inc. and its suppliers.
4 #   All rights reserved.
5 #
6 # By using this file, you agree to the terms and conditions set
7 # forth in the LICENSE file which can be found at the top level of
8 # the sendmail distribution.
9 #
10 #ident    "@(#)submit.mc    1.3 02/12/29 SMI"
11 #
12
13 #
14 # This is the prototype file for a set-group-ID sm-msp sendmail that
15 # acts as a initial mail submission program.
16 #
17
18 divert(0)dnl
19 VERSIONID(`$Id: submit.mc,v 8.6.2.4 2002/12/29 03:54:34 ca Exp $')
20 define(`confCF_VERSION', `Submit')dnl
21 define(`__OSTYPE__',``)dnl dirty hack to keep proto.m4 from complaining
22 define(`_USE_DECNET_SYNTAX_', `1')dnl support DECnet
23 define(`confTIME_ZONE', `USE_TZ')dnl
24 define(`confDONT_INIT_GROUPS', `True')dnl
25 dnl
26 dnl If you use IPv6 only, change [127.0.0.1] to [IPv6:::1]
27 FEATURE(`msp', `[127.0.0.1'])dnl
```

Let's see what the above line 27 does ("Message Submission Program"):



```
1$ cd /usr/lib/mail/feature
2$ ls -l msp.m4
-r--r--r--  1 root      mail           2910 Sep 23  2003 msp.m4
```

The internal macro `$h` (p. 638) is the “recipient host”. The macro `RELAY_MAILER_ARGS` (p. 630)

```
3$ grep 587 msp.m4
ifndef(_ARG2_, `MSA`, `define(`RELAY_MAILER_ARGS', `TCP $h 587`)'`)
```

587 is the port number:

```
4$ awk '$2 ~ /^587\\/' /etc/services
submission  587/tcp          # Mail Message Submission
submission  587/udp          #      see RFC 2476
```

Create the `sendmail.cf` file (p. 293)

```
1$ cd /usr/lib/mail/cf
2$ /usr/ccs/bin/m4 i5.nyu.mc > ~/sendmail.cf
```

Did we create a `sendmail.cf` file that is the same as `/etc/mail/sendmail.cf`?

```
3$ ls -l /usr/lib/mail/cf/i5.nyu.mc sendmail.cf /etc/mail/sendmail.cf
-r--r--r--  1 root      bin           53889 Oct  7  2003 /etc/mail/sendmail.cf
-rw-r--r--  1 konvit   staff         973 Oct  7  2003 /usr/lib/mail/cf/i5.nyu.mc
-rw-----  1 mm64     users        53922 Nov 30 16:23 sendmail.cf
```

There are six differences. Three of them are comments. A fourth is the different version numbers in the `sendmail` macro `Z` (pp. 321, 639).

```
4$ diff sendmail.cf /etc/mail/sendmail.cf
22c22
< ##### built by mm64@i5.nyu.edu on Wed Nov 30 16:23:42 EST 2005
---
> ##### built by kroot@i6.home.nyu.edu on Tue Oct 7 12:48:48 EDT 2003
58c58
< ##### $Id: proto.m4,v 8.649.2.24 2003/08/04 21:14:26 ca Exp $ #####
---
> ##### $Id: proto.m4,v 8.649.2.17 2003/03/28 17:20:53 ca Exp $ #####
150c150
< DZ8.12.10
---
> DZ8.12.9
478c478
< #O MaxMimeHeaderLength=2048/1024
---
> #O MaxMimeHeaderLength=1024/256
873,874c873
< R< error : $- : $+ > $*          $#error $@ $(dequote $1 $) $: $2
< R< error : $+ > $*          $#error $: $1
---
> R< error : $- $+ > $*          $#error $@ $(dequote $1 $) $: $2
1371c1370
< R$* $| $*          $: $1 $| $>"Local_trust_auth" $2
---
> R$* $| $*          $: $1 $| $>"Local_trust_auth" $1
```

`/etc/mail/sendmail.cf` is four seconds newer than `/usr/lib/mail/cf/i5.nyu.mc`. The time of last modification is the ninth field of the structure in `stat(2)`.

—On the Web at

<http://i5.nyu.edu/~mm64/x52.9547/src/sendmail/stat.pl>

```

1 #!/bin/perl
2
3 if (@ARGV != 1) {
4     die "$0: requires one command line argument";
5     exit 1;
6 }
7
8 @F = stat($ARGV[0]) or die "$0: $!";
9 print scalar(localtime($F[9])), "\n";
10 exit 0;

```

```

5$ cd ~mm64/public_html/x52.9547/src/sendmail
6$ stat.pl /usr/lib/mail/cf/i5.nyu.mc
Tue Oct  7 12:48:44 2003

```

```

7$ cd ~mm64/public_html/x52.9547/src/sendmail
8$ stat.pl /etc/mail/sendmail.cf
Tue Oct  7 12:48:48 2003

```

The superuser says

```

Date: Sat, 23 Oct 2004 14:58:05 -0400
From: Gary Konvit <gary.konvit@nyu.edu>
the cf file was probably created with sendmail version installed at
the time of creation. Subsequent patches installed newer versions of
the build structure, but did not update the cf. The Mime.. parameter
was changed in the newer config - the 4 seconds I can't really say.

```

**Test the rewrite rules**

```

2$ /usr/lib/sendmail -bt
ADDRESS TEST MODE (ruleset 3 NOT automatically invoked)
Enter <ruleset> <address>
> /tryflags HS
> /try esmtp mm64
Trying header sender address mm64 for mailer esmtp
canonify          input: mm64
Canonify2         input: mm64
Canonify2         returns: mm64
canonify          returns: mm64
1                 input: mm64
1                 returns: mm64
HdrFromSMTP       input: mm64
PseudoToReal      input: mm64
PseudoToReal      returns: mm64
MasqSMTP          input: mm64
MasqSMTP          returns: mm64 < @ *LOCAL* >
MasqHdr           input: mm64 < @ *LOCAL* >
MasqHdr           returns: mm64 < @ i5 . nyu . edu . >
HdrFromSMTP       returns: mm64 < @ i5 . nyu . edu . >
final             input: mm64 < @ i5 . nyu . edu . >
final             returns: mm64 @ i5 . nyu . edu
Rcode = 0, addr = mm64@i5.nyu.edu
> /quit

```

### Configure our own sendmail

—On the Web at

<http://i5.nyu.edu/~mm64/x52.9547/src/sendmail/minimal.mc>

```

1 include(`../m4/cf.m4')
2 VERSIONID(`i5.nyu.edu')
3 OSTYPE(solaris8)
4
5 #pp. 617, 644
6 DAEMON_OPTIONS(`Port=10566, Name=MTA')dnl
7
8 MAILER(local)dnl
9 MAILER(smtp)dnl
10 MAILER_DEFINITIONS
11 R$* < @ $* .nyu.edu. > $*      $#esmtp $@ $2.nyu.edu $: $1 < @ $2.nyu.edu > $3

1$ cd /usr/lib/mail/cf                Must be in this dir because of the .. in line 1.
2$ /usr/ccs/bin/m4 \
   ~mm64/public_html/x52.9547/src/sendmail/minimal.mc > \
   ~mm64/public_html/x52.9547/src/sendmail/sendmail.cf

3$ cd ~mm64/public_html/x52.9547/src/sendmail
4$ ls -l sendmail.cf
-rw-----  1 mm64      users      38775 Nov 30 16:23 sendmail.cf

5$ /usr/lib/sendmail -bd -C ./sendmail.cf
550 Permission denied (real uid not trusted)

```

To read SMTP from the standard input instead of from the SMTP port 25,

```
6$ /usr/lib/sendmail -bs -C ./sendmail.cf
451 4.0.0 can not chdir(/var/spool/mqueue/): Permission denied
Program mode requires special privileges, e.g., root or TrustedUser.
451 4.0.0 can not chdir(/var/spool/mqueue/): Permission denied
Program mode requires special privileges, e.g., root or TrustedUser.
```

```
7$ /usr/lib/sendmail -bs           Read SMTP from standard input.
8$ /usr/lib/sendmail abc1234       Read a letter from standard input.
```

□