mark bixby
hp csy r&d lab
march 2002

mpe/ix 7.5 sendmail support training

product overview

Major functionality:

- send SMTP e-mail
- receive SMTP e-mail to local mailboxes, programs, or files
- aliases can be created which map to local mailboxes, programs, files, or remote mailboxes
- powerfully flexible configuration language

Built from:

- Sendmail 8.12.1
- Sleepycat Berkeley DB library 3.3.11

patch strategy

New versions of Sendmail will use normal OS release and patch processes

system requirements

- will be released as a 7.0 patch
- will ship in 7.5 FOS
- requires TZ environment variable to be set correctly, preferably in the system logon UDC
- requires a local syslog daemon, either MPE FOS Syslog/iX or embedded spooling ISV syslog daemon
- the local e3000 must be configured correctly for DNS and must be listed correctly in the DNS database
- firewalls or other security devices must all the local e3000 to make DNS connections (port 53) and SMTP connections (port 25)

support

- Sendmail A.01.00 for MPE/iX will be fully supported by RC and WTEC
- Customers who call with questions regarding unsupported bixby.org freeware Sendmail 8.9.1 should be encouraged to upgrade to Sendmail A.01.00

file layout

- Uses the same scheme that was introduced with Apache A.02.00
- The MPE group A0100.SENDMAIL holds most of the files
- The symbolic link /SENDMAIL/CURRENT points to /SENDMAIL/A0100
- All customer modifiable files reside below /SENDMAIL/PUB, /SENDMAIL/CURRENT/cf, and /etc/mail

distribution highlights

- /SENDMAIL/PUB/JDAEMON – batch job for running the server daemon
- /SENDMAIL/PUB/SENDMAIL – symlink to /SENDMAIL/CURRENT/SENDMAIL
- /SENDMAIL/CURRENT/SENDMAIL – combined server daemon and local mail submission program
- /SENDMAIL/CURRENT/bin – dnscheck, hoststat, m4, mailq, newaliases, purgestat, vacation
- /SENDMAIL/CURRENT/sbin – editmap, mailstats, makemap, praliases, sendmail, smrsh

distribution highlights (cont.)

- /SENDMAIL/CURRENT/cf – directory tree for building *.cf config files; see the README file!
- /SENDMAIL/CURRENT/doc/op/op.ps – Sendmail Installation and Operation Guide – READ IT!
- /SENDMAIL/CURRENT/etc – contains the POSIX shell profile for Sendmail along with the sample config files installed to /etc/mail
- /SENDMAIL/CURRENT/man – man page documentation, I.e.:

```
export MANPATH=/SENDMAIL/CURRENT/man:$MANPATH

man sendmail
```

hpux compatibility symbolic links

/usr/bin/m4

/usr/bin/mailq

/usr/bin/mailstats

/usr/bin/newaliases

/usr/bin/praliases

/usr/bin/vacation

/usr/lib/sendmail

/usr/sbin/editmap

/usr/sbin/hoststat

/usr/sbin/mailstats

/usr/sbin/makemap

/usr/sbin/newaliases

/usr/sbin/purgestat

/usr/sbin/sendmail

/usr/sbin/smrsh

config files

- all config files live in /etc/mail which is populated from /SENDMAIL/CURRENT/etc/mail.sample at installation time if the /etc/mail files do not already exist
- all config files must be owned by the user SERVER.SENDMAIL and the POSIX group SENDMAIL
- the server daemon must be stopped and restarted for config file changes to take effect

config files (cont.)

- access – database map used to accept or reject incoming mail from selected domains
- aliases – database map for defining local recipient names in addition to the standard USER.ACCOUNT mailboxes
- domaintable – database map for rewriting domain names in mail headers
- genericstable – database map for rewriting the user and/or hostname portion of mail header addresses
- helpfile – the text returned by the SMTP protocol's HELP command
- local-host-names – ASCII file containing hostname aliases (if any) for the local machine

config files (cont.)

- mailertable – database map to override mail routing for specified domains
- sendmail.cf – configures the mail daemon server
- sendmail.pid – the POSIX PID of the currently running server
- statistics – binary file used to collect delivery statistics
- submit.cf – configures the mail submission program
- virtusertable – database map for doing domain-specific aliasing and the hosting of multiple virtual domains on the same machine

configuring *.cf files

- submit.cf and sendmail.cf are created from macro files expanded by the m4 utility
- you can edit submit.cf and sendmail.cf directly to make MINOR parameter changes:

```
# "Smart" relay host
DSmy.relay.host.name
```

- major functionality changes REQUIRE you to edit the macro files and expand with m4
- so just play it safe and ALWAYS edit the macro files and expand with m4 for ALL changes:

```
define(`SMART_HOST',
`my.relay.host.name')
```

- see /SENDMAIL/CURRENT/cf/README for the list of major *.cf options

configuring *.cf files (submit.cf for the mail submission program)

To generate submit.cf:

1. **:HELLO SERVER.SENDMAIL**
2. **:XEQ SH.HPBIN.SYS -L**
3. **shell/iX> cd /SENDMAIL/CURRENT/cf/cf**
4. **shell/iX> cp submit-mpeix.mc.sample submit-mpeix.mc**
5. edit submit-mpeix.mc with the bytestream file editor (i.e. vi) of your choice to make your changes
6. **shell/iX> m4 ../m4/cf.m4 submit-mpeix.mc >submit-mpeix.cf**
7. **shell/iX> cp submit-mpeix.cf /etc/mail/submit.cf**

configuring *.cf files (submit-mpeix.mc.sample)

```
define(`confCF_VERSION',
   `Submit')dnl

define(`__OSTYPE__',`')dnl dirty
   hack to keep proto.m4 from
   complaining

define(`_USE_DECNET_SYNTAX_',
   `1')dnl support DECnet

define(`confRUN_AS_USER',
   `SERVER.SENDMAIL')dnl

define(`confTIME_ZONE',
   `USE_TZ')dnl

FEATURE(`msp')dnl
```

configuring *.cf files (sendmail.cf for the mail server program)

To generate sendmail.cf:

1. **:HELLO SERVER.SENDMAIL**
2. **:XEQ SH.HPBIN.SYS –L**
3. **shell/iX> cd /SENDMAIL/CURRENT/cf/cf**
4. **shell/iX> cp generic-mpeix.mc.sample generic-mpeix.mc**
5. edit generic-mpeix.mc with the bytestream file editor (i.e. vi) of your choice to make your changes
6. **shell/iX> m4 ../m4/cf.m4 generic-mpeix.mc >generic-mpeix.cf**
7. **shell/iX> cp generic-mpeix.cf /etc/mail/sendmail.cf**

configuring *.cf files (generic-mpeix.mc.sample)

```
OSTYPE(mpeix)dnl

DOMAIN(generic)dnl

define(`confFORWARD_PATH',
   `$z/.forward')dnl

FEATURE(masquerade_envelope)dnl

FEATURE(domaintable)dnl

FEATURE(mailertable)dnl

FEATURE(genericstable)dnl

FEATURE(virtusertable)dnl

FEATURE(always_add_domain)dnl

FEATURE(access_db)dnl

MAILER(local)dnl

MAILER(smtp)dnl
```

*.cf internals

- For ALL of the gory details, please see the /SENDMAIL/CURRENT/doc/op/op.ps Sendmail Installation and Operation Guide section 5 – “The Whole Scoop on the Configuration File”.
- This section is 45 pages long and cannot be fully covered by a few slides or speaker notes!
- Few customers are expected to delve into *.cf internals; those who do are likely to be smart enough to answer their own questions. ;-)
- This format was designed to be easy for software to parse, not for humans to read
- First character of a line defines its semantics

*.cf internals
(D – define macro)

- macros named with a single letter or a word in {braces}
- user defined macros should use uppercase names only
- macros are dereferenced by $name, where name includes the braces if present

```
# "Smart" relay host (may be
null)
DSsmart.relay.hostname
```

*.cf internals (C and F – define classes)

- a class can be thought of as a macro containing multiple values
- C defines with constants, F defines from files, pipes, or database maps:

```
Cwlocalhost
# file containing names of hosts
# for which we receive email
Fw/etc/mail/local-host-names
```

- classes perform matching in the left hand side of rewriting rules
  - $=class – match an entry in the class
  - $~class – match an entry not in the class

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

*.cf internals
(M – define mailer)

- defines mailer programs and their interfaces
- highly unlikely to be modified by customers!

```
Mlocal,
  P=/bin/tsmail,
  F=lsDFMAw5:/|@qmu9,
  S=EnvFromL/HdrFromL,
  R=EnvToL/HdrToL,
  T=DNS/RFC822/X-Unix,
  A=tsmail $u
```

- see speaker notes for parameter details

*.cf internals
(H – define header)

- defines the format of header lines inserted into the message
- macro references in the header template will be expanded
- rulesets can be associated with headers to perform validation
- **H*hname*: *htemplate*** – unconditional
- **H?*mflags*?*hname*: *htemplate*** – conditional upon mailer flags
- **H?${*macro*}?*hname*: *htemplate*** – conditional upon macro existence

```
H?P?Return-Path: <$g>
```

- see speaker notes for details

*.cf internals (O – set option)

- zillions of sendmail options can be specified, and some of these can be overridden by /SENDMAIL/CURRENT/SENDMAIL command line parameters
- **-O *option=value***

O AliasFile=/etc/mail/aliases

- see speaker notes for details

*.cf internals (S and R – rewriting rules)

- the complicated, nasty, but powerful heart of sendmail
- **S*n*** – defines the current ruleset
- **R*lhs rhs comments*** – adds a rule
- one or more tabs separate lhs, rhs, and comment
- if the lhs pattern matches the address, the matching portion is replaced by the rhs string

*.cf internals (ruleset lhs metacharacters)

- **$*** Match zero or more tokens
- **$+** Match one or more tokens
- **$-** Match exactly one token
- **$=x** Match any phrase in class x
- **$~x** Match any word not in class x

*.cf internals (ruleset rhs metacharacters)

- **$*n*** Substitute indefinite token n from LHS
- **$[*name*$]** Canonicalize name
- **$(*map key* $@*arguments* $:*default* $)** - Generalized keyed mapping function
- **$>*n*** "Call" ruleset n
- **$#*mailer*** Resolve to mailer
- **$@*host*** Specify host
- **$:*user*** Specify user

*.cf internals (a simple example from ruleset 4)

- a snippet from ruleset 4:

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

- use sendmail –bt to test rulesets:

```
echo "4 foo%mpetest@mpetest" |
sendmail -bt
ADDRESS TEST MODE (ruleset 3 NOT
automatically invoked)
Enter <ruleset> <address>
> final input: foo % mpetest @
mpetest
final   returns: foo @ mpetest
```

- MUCH has been omitted from this discussion; please see op.ps for the gory details!

configuring database map files

- typically used by optional sendmail features like access_db
- Berkeley DB database hash or btree files containing extra configuration data in key/value pairs
- maintained with the makemap and editmap utilities
- by convention, a map file named “foo” contains the ASCII input data, whereas “foo.db” contains the compiled binary database structures
- in commands and *.cf files, a map file reference of “foo” actually refers to “foo.db”!

configuring database map files (cont.)

```
MPETEST:/BIXBY/PUB> cat - >foo
key1 value1
two abcdef
abra cadabra
:eod

MPETEST:/BIXBY/PUB> makemap hash
foo <foo

MPETEST:/BIXBY/PUB> ls -l foo*

-rw-r--r--   1 MGR.BIXBY
BIXBY         36 Feb 25 13:52
foo

-rw-r--r--   1 MGR.BIXBY
BIXBY      49152 Feb 25 13:52
foo.db

MPETEST:/BIXBY/PUB> makemap -u
hash foo

key1    value1
two     abcdef
abra    cadabra
```

common non-default config changes

- defining a smart relay host in sendmail.cf to route all outbound e-mail via a single mail gateway
- define alternate routing for certain outbound mail domains via the mailertable feature
- creating aliases for inbound e-mail instead of using the basic USER.ACCT@host.name mailboxes

aliases database map

- a special type of map file containing one or more comma-delimited values per key
- /etc/mail/aliases and aliases.db
- maintained by SERVER.SENDMAIL with the newaliases and praliases commands
- defines username aliases for mail being delivered to the local machine, I.e. postmaster@local.host.name
- installation default entries:

```
postmaster: SERVER.SENDMAIL
MAILER-DAEMON: postmaster
```

aliases database map (cont.)

- left-hand side is the user alias
- separated by a colon
- right-hand side is one or more delivery destinations:
  - USER.ACCOUNT or user@host.name or another alias
  - /path/to/local/file for appending
  - :include: /file/of/aliases (plaintext ASCII file)
  - "| /program/file parm1 parm2 parm3 …"
- local usernames may be escaped with a backslash (I.e. \USER.ACCOUNT) to prevent recursive alias lookups

.forward files

- an optional ASCII file named .forward residing in the local user's home group which tells sendmail where to forward the user's mail
- format is the same as the right-hand side of an aliases entry, I.e. one or more comma-separated destinations
- can be used to invoke the vacation autoresponder:

```
\USER.ACCOUNT,
"|/SENDMAIL/CURRENT/bin/vacation
USER.ACCOUNT"
```

access_db feature – accept or reject incoming e-mail based on envelope address or relaying mail server name

1. `:HELLO SERVER.SENDMAIL`
2. `:XEQ SH.HPBIN.SYS –L`
3. `/bin/cat - >/etc/mail/access`
   `imaspammer.com  REJECT`
   `:EOD`
4. `makemap hash /etc/mail/access`
   `</etc/mail/access`

domaintable feature – rewrite domain names in e-mail headers

1. `:HELLO SERVER.SENDMAIL`
2. `:XEQ SH.HPBIN.SYS –L`
3. `/bin/cat - >/etc/mail/domaintable`
   `oldcompany.com newcompany.com`
   `:EOD`
4. `makemap hash /etc/mail/domaintable`
   `</etc/mail/domaintable`

genericstable feature – rewrite user and/or domain addresses in outgoing e-mail headers

1. `:HELLO SERVER.SENDMAIL`
2. `:XEQ SH.HPBIN.SYS –L`
3.
```
/bin/cat -
>/etc/mail/genericstable
USER.ACCOUNT@my.local.host
customer_servce@company.com
:EOD
```
4.
```
makemap hash
/etc/mail/genericstable
</etc/mail/genericstable
```

Note that domains being modified by genericstable must be added to /etc/mail/sendmail.cf class {G}.

mailertable feature – override default mail routing in sendmail.cf

```
1. :HELLO SERVER.SENDMAIL
2. :XEQ SH.HPBIN.SYS –L
3. /bin/cat -
   >/etc/mail/mailertable
   .bitnet   smtp:relay.bit.net
   :EOD
4. makemap hash
   /etc/mail/mailertable
   </etc/mail/mailertable
```

virtusertable feature – remap incoming user and hostnames to local users

1. `:HELLO SERVER.SENDMAIL`
2. `:XEQ SH.HPBIN.SYS –L`
3.
```
/bin/cat -
>/etc/mail/virtusertable
info@bar.com   INFO.BAR
info@foo.com   INFO.FOO
:EOD
```
4.
```
makemap hash
/etc/mail/virtusertable
</etc/mail/virtusertable
```

Note that virtual hostnames must be listed in /etc/mail/local-host-names.

starting the mail daemon

- Make sure a syslog daemon is running before you start the mail daemon!
- To start the MPE FOS syslog daemon, **`:STREAM JSYSLOGD.PUB.SYSLOG`**
- To start the mail daemon, **`:STREAM JDAEMON.PUB.SENDMAIL`**

stopping the mail daemon

- Use the POSIX kill signal from SERVER.SENDMAIL or any user with SM capability:

```
kill $(head -n 1
/etc/mail/sendmail.pid)
```

- Only use **:ABORTJOB** as a last resort!

sending e-mail with mailx

- interactively:

```
mailx someuser@some.host
Subject: hello world
Hi,

How are you doing?
:EOD

EOT
```

- from a pipe:

```
echo "How are you doing?" |
mailx -s "hello world"
someuser@some.host
```

- from a disk file:

```
mailx -s "hello world"
someuser@some.host
</diskfile/containing/msg/body
```

- no attachments!
- limited control of mail headers!

sending e-mail with SENDMAIL

1. **/bin/cat - >message.txt**
   To: someuser@some.host
   Cc: otheruser@other.host
   Bcc: secretuser@another.host
   Subject: hello world

   Hi there!
   :EOD
2. **/SENDMAIL/CURRENT/SENDMAIL -t <message.txt**

- the **-t** option reads the destination addresses from the message headers
- destination addresses can alternatively be specified on the SENDMAIL command line
- if you want attachments you must generate the MIME headers yourself

sending e-mail with forged headers

1. **/bin/cat - >message.txt**
   From: forger@foobar.com
   To: someuser@some.host
   Cc: otheruser@other.host
   Bcc: secretuser@another.host
   Subject: hello world

   Hi there!
   :EOD
2. **/SENDMAIL/CURRENT/SENDMAIL -t –f forger@foobar.com <message.txt**

- the **–f** option sets the message envelope address, but a warning header is included:

  X-Authentication-Warning: local.e3000.host: USER.ACCT set sender to forger@foobar.com using –f

- genericstable and other options can forge without traces

sending e-mail – how it works (client)

1. mailx creates a fully-formatted message and passes it to SENDMAIL as configured in /etc/mailx.rc
2. SENDMAIL reads a fully formatted message from stdin and queues the message in /var/spool/clientmqueue
3. SENDMAIL attempts to contact the mail daemon on localhost port 25.
4. If the mail daemon answers, the message is transferred using SMTP and deleted from clientmqueue, else the message is left in clientmqueue

sending e-mail – how it works (server)

1. JDAEMON processes /var/spool/clientmqueue once at startup to handle any messages submitted while JDAEMON wasn't running
2. new messages are read from port 25 using SMTP and queued in /var/spool/mqueue
3. the remote mail server is resolved via DNS, and a connection is tried to port 25
4. if the message is delivered successfully, it is removed from mqueue, else it remains until mqueue is processed again by the local mail daemon
5. local messages are delivered by /bin/tsmail to /usr/mail/USER.ACCOUNT

receiving e-mail with mailx

- invoke /bin/mailx with no parameters, and it will read e-mail from /usr/mail/USER.ACCOUNT
- a numbered headers summary is printed; refer to these numbers in mailx commands
- **`type nnnn`** – display a message
- **`delete nnnn`** – delete a message
- **`help`** – for further details
- **`quit`** – exits after updating the mailbox
- see “man mailx” or the Shell & Utilities manual for further details

receiving e-mail programmatically

- incoming e-mail will be delivered to programs specified in the aliases database or .forward files
- the e-mail will be delivered to the program via a POSIX pipe connected to the program's stdin
- the POSIX newline character \n (ASCII LF) is used as a record separator
- if the program terminates with a non-zero POSIX exit status, any info written to stderr will be returned in a bounce message

receiving e-mail – how it works

1. JDAEMON listens for incoming SMTP connections on port 25
2. Messages are read using SMTP protocol and queued to /var/spool/mqueue
3. If the destination address appears to be local, the aliases database and .forward file (if any) is used to resolve the final delivery address
4. sendmail.cf determines the mailer to be used based on the final delivery address
5. Local messages get delivered by /bin/tsmail to /usr/mail/USER.ACCOUNT
6. Messages for remote destinations are delivered via SMTP

migrating from freeware 8.9.1

- must create new JDAEMON from /SENDMAIL/CURRENT/JDAEMON.sample
- all config files reside in /etc/mail instead of /SENDMAIL/PUB/etc
- 8.9.1 sendmail.cf is NOT compatible with 8.12.1
- copy all 8.9.1 ASCII map files to /etc/mail and rebuild with makemap and newaliases
- 8.9.1 queued messages won't be seen by 8.12.1
- 8.12.1 uses two queues (/var/spool/clientmqueue and mqueue) instead of 8.9.1's /SENDMAIL/PUB/mqueue

migrating from freeware 8.9.1 (cont.)

- 8.9.1 would submit new messages directly to the queue disk files, but 8.12.1 speaks SMTP to localhost port 25
- 8.12.1 uses two main config files, /etc/mail/submit.cf for submitting new messages, and sendmail.cf for general mail routing
- 8.12.1 does not include the Majordomo mailing list software that was bundled with 8.9.1. HP does not support Majordomo!

mpe/ix implementation issues (features not implemented)

- LDAP directory lookups
- TLS/SSL encrypted e-mail transport
- SASL secure authentication
- mail filtering
- optional chroot()-based security features
- optional nice()-based dispatching priority adjustments

mpe/ix implementation issues (things that work differently)

- Sendmail programs don't read stdin terminal keyboard input correctly. Workarounds:
  - **`/bin/cat - | makemap hash mymap`**
  - **`makemap hash mymap <diskfile`**
- DeliveryMode=background on MPE is a hybrid between "background" and "interactive"
- Symlinks invoke different personalities of SENDMAIL (I.e. mailq, newaliases, etc), and these only work properly from the POSIX shell because the CI doesn't initialize ARGV[0]
- newaliases gives a "cannot change ownership" warning which can be ignored

mpe/ix implementation issues (setuid/setgid program file bits)

- Sendmail expects a full implementation of setuid(), setgid(), and a superuser uid of 0
- The main sendmail porting challenge was to provide such a uid/gid emulation layer
- This emulation is enabled via the /SENDMAIL/CURRENT/SENDMAIL program file setuid and setgid bits which are not currently used by MPE, I.e.:

```
chmod u+s,g+s
/SENDMAIL/CURRENT/SENDMAIL

-rwsr-sr-x   1 MGR.SENDMAIL
SENDMAIL 2424320 Feb  5 16:41
SENDMAIL
```

dns issues

- the #1 sendmail problem!
- before using sendmail, run the dnscheck script:

1. **`:HELLO SERVER.SENDMAIL`**
2. **`:XEQ SH.HPBIN.SYS -L`**
3. **`/SENDMAIL/CURRENT/bin/dnscheck`**

- make any recommended config changes and then rerun the script until success is reported
- see speaker notes for sample dnscheck output

dns issues (cont.)

- single-token hostname? I.e. uname –n returns “jazz”?
- domain name in /SYS/NET/RESLVCNF? I.e. “domain external.hp.com”?
- one or more nameserver entries in RESLVCNF?
- fully-qualified domain name has a DNS “A” record specifying the IP address of the local machine?
- IP address has a DNS “PTR” record specifying the fully-qualified domain name?
- if no to any of the above, sendmail may hang, refuse to start, be unable to recognize the local host, and generally fill up syslog with interesting msgs

firewall issues

- the #2 sendmail problem!
- your 3000 needs to query port 53 on DNS servers to resolve the destination mail server hostname
- your 3000 needs to receive DNS query answers
- your 3000 listens on its port 25 for incoming e-mail
- your 3000 needs to connect to port 25 on destination mail servers
- does your firewall allow your 3000 to talk to the Internet?
- does your firewall allow the Internet to talk to your 3000?
- firewall blocking results in connection refused, timeouts, or just no activity!

troubleshooting

- check syslog first!
- if nothing in syslog:
  - if your third-party spooling package has an embedded syslog daemon, you will probably need to use that one instead of Syslog/iX
  - is the syslog daemon running?
  - does the syslog daemon have read access to the config file and write access to the log file?
  - is syslog configured to log mail events?
- if syslog or e-mail message headers show strange timestamps, verify TZ is set properly, preferably in your system logon UDC

troubleshooting (cont.)

- if syslog shows DNS lookup failures:
  - run the dnscheck script to verify DNS is configured properly
  - verify that your firewall allows your 3000 to talk to DNS servers on port 53
- if syslog shows connection failures to remote mail servers, verify that your firewall allows your 3000 to connect to port 25; if it does not, you may need to configure sendmail.cf to use a smart host mail relay
- long delays submitting new messages are indicative of DNS problems; check syslog and run the dnscheck script

troubleshooting (cont.)

- if local users are submitting messages that aren't being delivered:
  - verify that JDAEMON is running; if it is not, messages will be queued in /var/spool/clientmqueue
  - transient network problems may cause a backlog in the mail daemon queue /var/spool/mqueue; run /SENDMAIL/CURRENT/bin/mailq as SERVER.SENDMAIL to get a formatted queue listing

troubleshooting (cont.)

- if remote users are sending messages that aren't being delivered to the local 3000:
  - check syslog for remote connection attempts; if there are none, does your firewall allow connections to port 25 on your 3000, and are your 3000's DNS entries visible to the remote users?
  - verify that the remote users are using valid e-mail addresses for your 3000, I.e. USER.ACCOUNT or an alias, plus the correct hostname

troubleshooting (cont.)

- if a sendmail config change doesn't appear to take effect:
  - stop and restart the mail daemon when making *.cf changes
  - if you changed an ASCII database map file, don't forget to run makemap or editmap to compile the binary *.db file
  - if you changed ASCII /etc/mail/aliases, don't forget to run newaliases to compile the binary /etc/mail/aliases.db file

debug flags

- mind-bending amounts of low-level debugging output can be generated with the /SENDMAIL/CURRENT/SENDMAIL debug flags parameter
- **`-dcategory.level`**
- **`-d12`** Set category 12 to level 1
- **`-d12.3`** Set category 12 to level 3
- **`-d3-17`** Set categories 3 through 17 to level 1
- **`-d3-17.4`** Set categories 3 through 17 to level 4
- probably only useful when looking at the source code to understand what's being debugged
- see speaker notes for list of categories

syslog log levels

- syslog verbosity can be increased
- the default log level of 9 covers most failures and mundane successes
- override via sendmail.cf:

```
# log level
O LogLevel=15
```

- override via SENDMAIL command line:

```
-O LogLevel=15
```

- See speaker notes for complete list of log levels

syslog message formats

The MPE user USER.ACCT on the local e3000 with a hostname of myhost.mydomain.com has just submitted a new message with 1 recipient consisting of a message body size of 5 bytes:

```
Feb  6 12:14:42 localhost sendmail[65622]: g16HEgik065622:
from=USER.ACCT, size=5, class=0, nrcpts=1,
msgid=<200202061714.g16HEgik065622@myhost.mydomain.com>,
relay=USER.ACCT@localhost
```

The new message is being relayed via the local host, i.e. Sendmail is connecting to TCP port 25 (SMTP) on the local host in order to queue the message:

```
Feb  6 12:14:43 localhost sendmail[65623]: g16HEgs9065623:
from=<USER.ACCT@myhost.mydomain.com>, size=5, class=0, nrcpts=1,
msgid=<200202061714.g16HEgik065622@myhost.mydomain.com>,
proto=ESMTP, daemon=MTA, relay=localhost [127.0.0.1]
```

syslog message formats (cont.)

The new message has been successfully queued on the local host and will eventually be delivered to destuser@remhost.mydomain.com:

```
Feb  6 12:14:43 localhost sendmail[65622]: g16HEgik065622:
to=destuser@remhost.mydomain.com, ctladdr=USER.ACCT (153/126),
delay=00:00:01, xdelay=00:00:01, mailer=relay, pri=30091,
relay=localhost [127.0.0.1], dsn=2.0.0, stat=Sent (g16HEgs9065623
Message accepted for delivery)
```

The Sendmail daemon on the local host is now processing the queue for the message being sent from USER.ACCT@myhost.mydomain.com to destuser@remhost.mydomain.com. The remote mail server's IP address is 192.168.0.1, and the message was successfully sent:

```
Feb  6 12:14:46 localhost sendmail[65625]: g16HEgs9065623:
to=<destuser@remhost.mydomain.com>,
ctladdr=<USER.ACCT@myhost.mydomain.com> (153/126), delay=00:00:03,
xdelay=00:00:03, mailer=esmtp, pri=120377,
relay=remhost.mydomain.com. [192.168.0.1], dsn=2.0.0, stat=Sent
(g16HNwC810485863 Message accepted for delivery)
```

syslog message formats (cont.)

In this next example, the remote user destuser@remhost.mydomain.com is sending an incoming message to some user on the local e3000. The remote mail server that has connected to your local e3000 is remhost.mydomain.com, and its IP address is 192.168.0.1:

```
Feb  6 12:15:13 localhost sendmail[131160]: g16HFDs9131160:
from=<destuser@remhost.mydomain.com>, size=31, class=0, nrcpts=1,
msgid=<200202061724.g16HOMLs065645@remhost.mydomain.com>,
proto=ESMTP, daemon=MTA, relay=remhost.mydomain.com [192.168.0.1]
```

The local Sendmail daemon has successfully delivered the incoming message to the local user USER.ACCT:

```
Feb  6 12:15:14 localhost sendmail[131161]: g16HFDs9131160:
to=<USER.ACCT@myhost.mydomain.com>, delay=00:00:01,
xdelay=00:00:01, mailer=local, pri=30042, dsn=2.0.0, stat=Sent
```

for further information

- 7.5 Communicator
- 7.5 Configuring & Managing MPE/iX Internet Services
- http://jazz.external.hp.com/src/sendmail/
- http://www.sendmail.org/
- http://www.sleepycat.com/ for the Berkeley DB database used for map files

any questions?