

Email Architecture

with sendmail and postfix

Karst Koymans / Jeroen van der Ham

Informatics Institute
University of Amsterdam

Tuesday, October 4, 2011

- 1 Structure
- 2 Email flows
- 3 Sendmail
- 4 Postfix
- 5 Other MTA's

Outline

- 1 Structure
- 2 Email flows
- 3 Sendmail
- 4 Postfix
- 5 Other MTA's

Organisation

- Say **example** is an example organisation
 - **cs** is an autonomous suborganisation
 - **bio** is a managed suborganisation
 - **inf** is an autonomous part of bio
 - **po** is a managed part of bio

Structure of the organisation

- example
 - cs
 - bio
 - inf
 - po

DNS mirrors the structure

- Where are the cuts?
- example.edu.
 - cs.example.edu.
 - bio.example.edu.
 - inf.bio.example.edu.
 - po.bio.example.edu.

Email mirrors the structure

mail.* are mail relays and servers

- mail.example.edu.
 - mail.cs.example.edu.
 - mail.bio.example.edu.
 - mail.inf.bio.example.edu.
 - mail.po.bio.example.edu.

MX records (1)

example.edu.	MX	0	mail.example.edu.
cs.example.edu.	MX	0	mail.cs.example.edu.
		10	mail.example.edu.
bio.example.edu.	MX	0	mail.bio.example.edu.
		10	mail.example.edu.

MX records (2)

inf.bio.example.edu.	MX	0	mail.inf.bio.example.edu.
		5	mail.bio.example.edu.
		10	mail.example.edu.
po.bio.example.edu.	MX	0	mail.po.bio.example.edu.
		5	mail.bio.example.edu.
		10	mail.example.edu.

Email addresses

- Employee “The Boss” working in department “inf”
 - boss@inf.bio.example.edu
 - The.Boss@bio.example.edu
 - emp0@example.edu

Email forwarding

- emp0@example.edu is forwarded to
 - The.Boss@bio.example.edu, which is in turn forwarded to
 - boss@inf.bio.example.edu
- Forwarding can be
 - user based (.forward)
 - system based (alias file or database)

Outline

- 1 Structure
- 2 Email flows**
- 3 Sendmail
- 4 Postfix
- 5 Other MTA's

SMTP flow (inbound) (1)

- Directly to mailhost in MX record
 - emp0@example.edu enters at top
 - boss@inf.bio.example.edu enters at leaf

SMTP flow (inbound) (2)

- Always to mail.example.edu.
 - Requires “split” DNS
 - Different outside MX record for inf.bio.example.edu., pointing to mail.example.edu.
 - Alternatively block port 25 from the outside
 - What is your opinion about this solution?

SMTP flow (outbound)

- Directly to outside world
 - No “corporate” policy
 - Needs smart hosts decentrally
- Flowing up the tree step by step
 - Uses the “smart host” option
- Directly to the top of the tree
 - Uses the “smart host” option

Mail access

- Only leaf mail servers supply mail access
- Intermediate servers are relay only
- In case you want to deliver higher in the tree
 - Create an extra child for mail delivery
 - Separate SMTP relay from local delivery and IMAP access
 - Possibly separate outgoing and incoming email,
much like separating authoritative and caching name servers

Outline

- 1 Structure
- 2 Email flows
- 3 Sendmail**
- 4 Postfix
- 5 Other MTA's

sendmail configuration (for Ubuntu 12.04 LTS)

- Debian specific (based on sendmail 8.14.4)
- Has an extensive init script to control sendmail execution
- Uses a separate sendmail.conf file to source inside init script
- Uses a helper program (sendmailconfig) to generate the main configuration file sendmail.mc

sendmail configuration directory

- /etc/mail as configuration directory
 - sendmail.mc, which is used to generate
 - sendmail.cf
 - using the m4 macro processor
 - local-host-names
 - aliases
 - access
 - ...

m4 macros

- Inside `/usr/share/sendmail/cf`
 - m4 source files `m4/*`
 - `cf.m4`, `cfhead.m4`, `proto.m4`
 - `debian/*`, `domain/*`, `feature/*`, ...
 - `hack/*`, `mailer/*`, `ostype/*`, ...

sendmail.mc

- OSTYPE(debian)
- DOMAIN(debian-mta)
- DAEMON_OPTIONS(...)
- FEATURE(...)
 - no_default_msa
 - access_db
 - ...
- MAILER
 - local
 - smtp

debian.m4

- `define(conf. . .)`
- Lots of configuration parameters, to name a few
 - `confSMTP_LOGIN_MSG`
 - `confCW_FILE`
 - `confDEF_USER_ID`

debian-mta.m4

- Many more conf... options
 - confMAX_HOP
 - confDONT_BLAME_SENDMAIL
 - All kinds of TimeOut(TO)-timers
 - confTO_MAIL
 - confTO_QUIT
 - ...

sendmail.cf macros

- Macro
 - C<class>
 - referenced as \$=<class>
 - F<class_in_file>
 - for example Fw/etc/mail/local-host-names
 - also referenced as \$=<class_in_file>
 - D<name>
 - referenced as \$<name>
 - Classes and names are often single letter identifiers

sendmail.cf hostnames

- `sendmail -bt -d0.4`
 - for debugging local hostname(s)
 - `Dj$w.$m` (set internally; automatically)
 - hence `$j=$w.$m`
- What is inside `$=w` class?
 - Many “hostnames”, also numeric

sendmail.cf map lookup

- K<mapname> <type> <detail>
 - mailertable hash -o /etc/mail/mailertable.db
 - generics hash -o /etc/mail/genericstable.db
 - virtuser hash -o /etc/mail/virtusertable.db

sendmail.cf options

- AliasFile
- ForwardPath
- DaemonPortOptions (UseMSP)
- Timeout
- *LA (Queue, Refuse, Delay)
- SmtptGreetingMessage
- ...



sendmail.cf headers

- HReceived:
 - `$?sfrom $s $. $?_ ($?s$|from $. $ _)$`.
 - `$? {auth_type} (authenticated$? {auth_ssf} bits=${auth_ssf}$.)$`.
 - `by $j ($v/$Z)$?r with r. id $i`
 - `$? {tls_version} (version=${tls_version} cipher=${cipher} bits=${cipher_bits} verify=${verify})$`.
 - `$?u for $u; $|; $. $b`

sendmail.cf mailers

- M<mailer> <attributes>
 - local (maybe procmail as MDA)
 - prog, *file*, *include* (builtin)
 - smtp, esmtp, smtp8, relay, bsmtp, fido
 - procmail (as mail filter, called with “-m”)

sendmail.cf rulesets and rules

- **S**<name>=<number> indicates a ruleset
 - canonify=3 (always first)
 - parse=0 (resolves <mailer,host,user>)
 - check_relay (to disable open relaying)
 - check_mail (checks MAIL FROM:)
 - check_rcpt (checks RCPT TO:)
- **R**<LHS>  <RHS>  <comment> indicates a rule

sendmail.cf rules

- LHS (Left Hand Side)
 - \$*, \$+, \$- (token matching)
 - \$@ (matching zero tokens, used for empty input)
 - \$=, \$~ (class matching, positive or negative)
- RHS (Right Hand Side)
 - \$1, \$2, ... (substitution of matched parts)
 - \$:, \$@ (control flow)
 - \$>, \$?\$|\$\$. (recursion; conditional)
 - \$[...\$], \$(...\$) (IP lookup; map lookup)

Sendmail ruleset testing

Command	Use for
<code>=S<ruleset>, =M</code>	showing rulesets and mailers
<code>\$<m>, \$=<c></code>	evaluating macros
<code>/parse <address></code>	address parsing
<code>/try <mailer> <address></code>	address given to mailer
<code>/map <map> <lookup></code>	map lookup

Table: Some sendmail -bt commands

Outline

- 1 Structure
- 2 Email flows
- 3 Sendmail
- 4 Postfix**
- 5 Other MTA's

Postfix

- (Mostly) compatible with sendmail
 - supplies `/usr/{lib,sbin}/sendmail` emulation
- Good performance
- Safe and secure
- Modular and flexible

General postfix features

- Support for multiple transports
- Easy virtual domain configuration
- Extensive UCE/SPAM control
- Rewriting through table lookups

Postfix modular setup

- One resident master process
 - compare to inetd super server
- Some semi-resident daemons
 - started via master.cf file
 - something like inetd.conf

Postfix queues

- maildrop (local incoming)
- incoming (after cleanup)
- active (being worked on)
 - deferred (temporary failure)
 - hold (needs human intervention)
 - corrupt (needs human inspection)

Postfix security

- Runs not setuid root
- Uses chroot environment
- Is modular and not monolithic
- Filtering of outside information

Postfix daemons

- pickup (mail from maildrop via postdrop (“sendmail”))
- smtpd (remote mail from the Internet)
- cleanup (repairs incoming mail)
- qmgr (processes mail queues)
- local (local delivery)
- smtp (remote delivery)

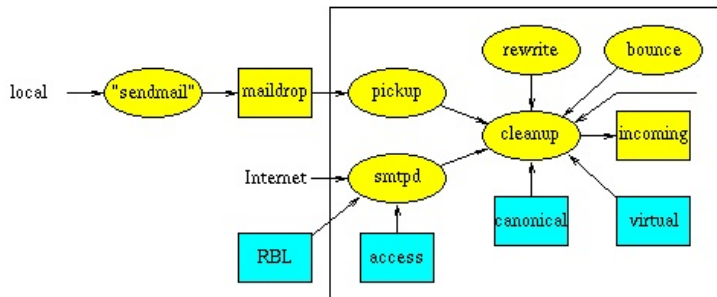
Postfix assistants

- (trivial-)rewrite
 - canonicalisation (compare “ruleset 3”)
 - resolving (compare “ruleset 0”)
- bounce
 - error mailer
 - defer messages

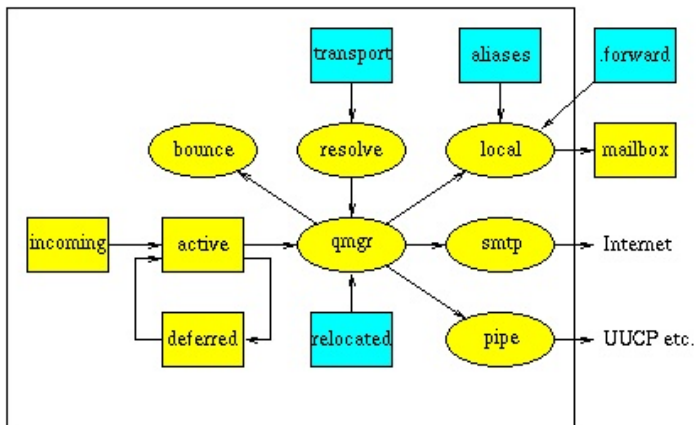
Postfix/Sendmail tables

Postfix	Sendmail
virtual	virtusertable
canonical	genericstable
transport	mailertable
access	access
relocated	- (aliases)

Postfix architecture inbound



Postfix architecture outbound



Outline

- 1 Structure
- 2 Email flows
- 3 Sendmail
- 4 Postfix
- 5 Other MTA's**

qmail

- Who looked at qmail and wants to explain?

Exim

- Who looked at Exim and wants to explain?