

Overview of Email Systems

Fank A. Kuse
29/07/09

Overview of Email Systems

The Internet email system is based on a number of local components, each of which has a specific role. Each computer communicates with the others through the use of open protocols. A sample message is described below with the help of a diagram below.

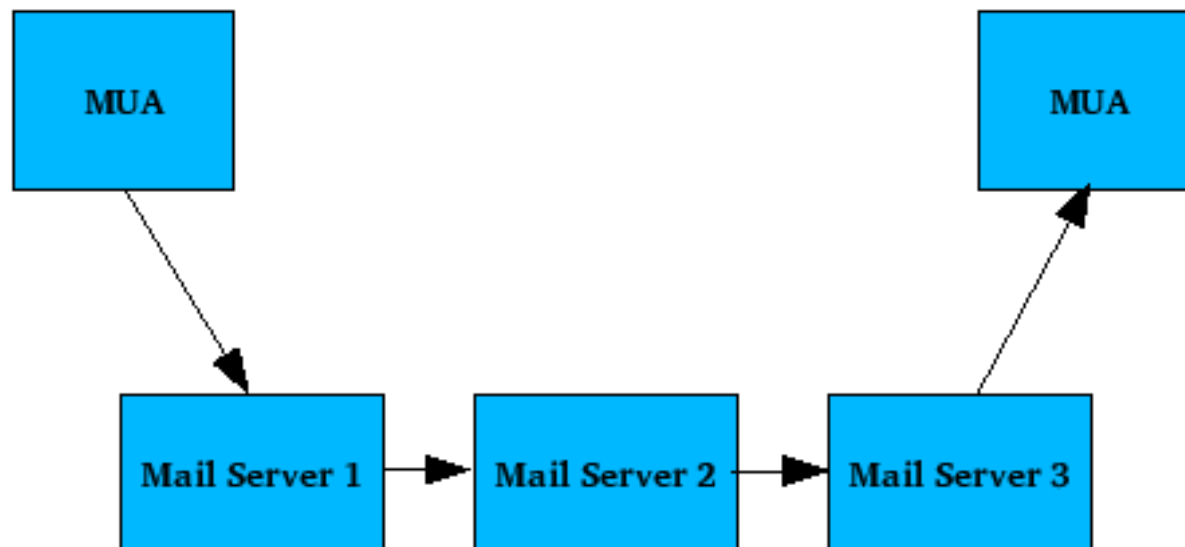


Figure 1 - Internet Mail Model

The diagram shows the path of the delivery of a single email. The mail is generated by a mail user agent (MUA). It is then passed to a mail server which has to decide whether it can deliver the mail locally or not. If not, then this delivery is deferred. The mail is then available to a mail user agent (MUA) to read it. The final MUA has the responsibility of retrieving the mail as well as passing it to a mail user interface (MUI) to display it to the user. Now each mail server decides whether to deliver locally or not depends on other configuration parameters implemented by the MTA in operation.

Examples of MUAs are 'mutt', 'Eudora', 'outlook' and 'Thunderbird'. Examples of MUIs are 'netscape' and 'outlook'.

- ow Email , orks

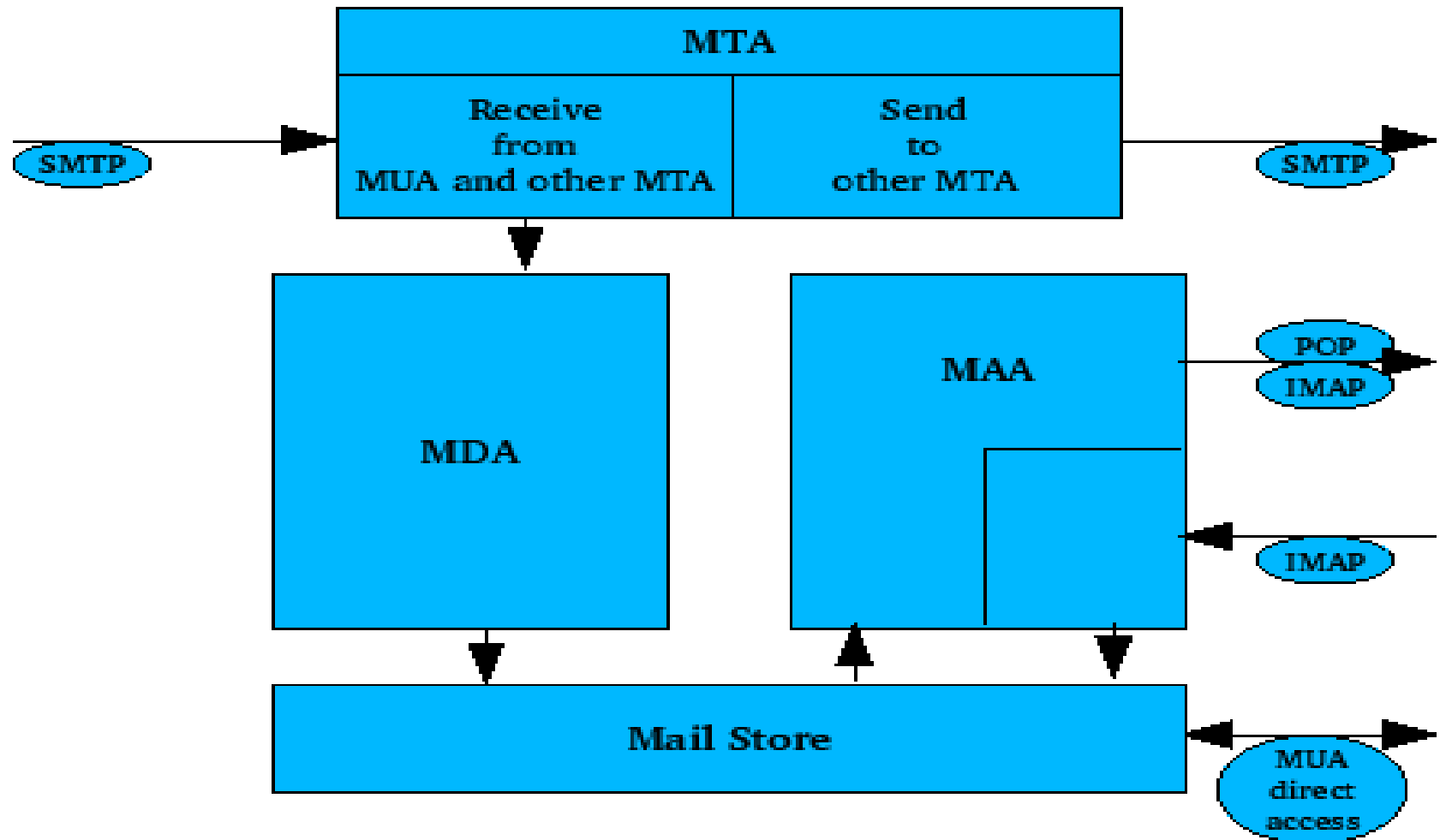


Figure 2 – Mail Server Components

Briefly, each server consults a local configuration file or files together with information from DNS servers regarding the remote hosts. This is all then used to determine what is the local. For non-local mail the server then uses this information to determine the address of the next mail server to send the mail to. Each mail server has the structure shown in Fig 2.

The Mail Transport Agent (MTA) receives mail from other mail servers and as well as mail from the user. If the mail is not for local delivery it is then sent by the MTA to another server.

If the mail is for local 'delivery' it is passed to a mail delivery agent * ! ! A+. The ! ! A is responsible for storing the mail in the user's mailbox. The mailbox is simply a way of storing 'data' for instance 'a file' a series of separate files or even an S78 'data'. The 'remote' storage structure is 'defined' by whatever the ! ! A supports.

When a user wants to view his mail he uses a ! ! A which either retrieves the mail 'directly' or connects a server side component which retrieves the mail from the mailbox and 'passes' it to the ! ! A. Such server side components 'do not' fit into the traditional ! ! A/ ! ! A/ ! ! A model and we shall call them ! ! A-less Agents * ! ! A+.

The !) A %ommuni%ates with a ! AA usin\$ an o&en &roto%ol whi%h is usually either the .ost Offi%e . roto%ol * . O . + or the nternet ! ail A%%ess . roto%ol * ! A . +. The . O . &roto%ol normally "eletes mail from the mailstore when it is &asse " to the %lient an " ! A . normally leaves it there. The ! A . &roto%ol also allows the !) A to alter the mailstore' for e/am&le #y "eletin\$ mail or movin\$ it from one "ire%tory to another.

The !) A may store the mail locally on the machine it is running on. This normally happens if the .O. is used. This local storage then allows future access to the content of the server which is particularly useful for machines that are not permanently connected to the network.

! A . ' on the other hand normally operates without local copies but it can also operate in what is called "connected mode" which maintains a local copy thereby allowing mail to be manipulated without a network connection.

In this mode the local and server mailstores are synchronized when a network connection is made. 2.0 2ot all !) A supports this feature.

Sometimes a program other than a mail client retrieves the mail and stores it locally for a mail client to access without having to connect to the server itself. Such programs pull mail onto their machine in contrast to a standard mail transfer agent (MTA) which has mail pushed to it by other MTAs. This can be useful if users do not want to allow connections to their machines from the Internet or are operating behind a firewall. An example of such a program is fetchmail.

As mail is passed from the originating mail client through the various servers to the final mail client, a series of headers is added which records details of the journey and also controls the processing of the mail by both the intervening servers and the final mail client.

Some of these are ! ulti4&ur&ose nternet ! ail
E/tension * ! ! E+ hea" ers whi%h are use" for a
ran\$e of %ontrol &ur&oses' in%lu" in\$ su&&ort for
non4AS : %hara%ter sets' su&&ort for em#e " " e "
%ontent su%h as ima\$es' an" su&&ort for
atta%hments. , hen a !) A atta%hes a file it
re%or"s its ty&es as a ! ! E hea" er an" it is then
the res&onsi#ility of the !) A to #e a#le to " e%o" e
it.

. i%torial e/am&le of some of this mo"el are
"is%usse" in more "etail #elow.

Example of an Email - headers

; .+ - ea"er a" "e" #y ! TA

```
Ae%eive "< from taurus.%us.%am.a%.uk
*B; 92.; CD.EF.GFH i" ent l e/im+
#y mauve.%si.%am.a%.uk with esmt&
*E/im F.00+ i" ;0; J/34000;; 34006
Fri' ;0 ! ay 2002 ;;<G0<E9 K0; 00
```

```
Ae%eive "< from &h; 0 *helo l lo%alhost+
#y taurus.%us.%am.a%.uk with local-smt&
*E/im F.; 0+ i" ;0; Jin4000G . 04006
Fri' ;0 ! ay 2002 ;;<G0<2G K0; 00
```

From< .hili& - a9el =&h; 0 > %us.%am.a%.uk?

To< @ulius : aesar =(ulius > an%ient4rome.net?

%%< ! ark Anthony = ! arkA > %leo.%o.uk?

! essa\$e Format

- **Envelope**

- Aoutin\$ information for the L&ostmanL

- **Message Header**

- Sen " er
- Ae%i&ients *sim&le' lists' %o&ies' #lin" %o&ies+
- Other fiel"s of %ontrol * "ate' su#(e%t+

- **Message Body**

- Free te/t
- Stru%ture" " o%ument *i.e.< ! ! E+

Example of Internet Mail Format

From: hili@us.ibm.com

To: ulius@aesar.rome.net

Subject: ark Anthony

Subject: ark Anthony

@ulius'

I'm going to be running a course on ...

- Format was originally "rfc822" by AF: D22 in 1982
- Now uses "rfc2822" by AF: 2822
- Internet Mail consists of
 - Header lines
 - Blank line
 - Body lines

A messa\$e in transit

- A messa\$e is transmitt" with an *envelope*<

MAIL FROM:<ph10@cus.cam.ac.uk>

RCPT TO:<julius@ancient-rome.net>

- The envelo&e is se&arate from the AF : 2D22 messa\$e

- Envelo&e *AF : 2D2 ;+ fiel"s nee" not #e the same as the hea"er *AF : 2D22+ fiel"s

- ! TAs are *mainly+ %on%erne" with envelo&es

Just like the Post Office...

- Error *M#oun%eN+ messa\$es have null sen"ers

MAIL FROM:<>

E / am&le of S ! T . Session

telnet relay.ancient-rome.net 25

220 relay.an%ient4rome.net ES ! T . E/im ...

EHLO taurus.cus.cam.ac.uk

2G04relay.an%ient4rome.net ...

2G04S OE ;0FDG7C0

2G04. . E8 2 2P

2G0 - E8 .

MAIL FROM:<ph10@cus.cam.ac.uk>

2G0 OK

RCPT TO:<julius@ancient-rome.net>

2G0 A%%e&te "

DATA

EGF Enter messa\$e' en "in\$ with M.N

Received: from ...

(continued >>>>)

From: ...

To: ...

etc...

2G0 OK i" l ;0s . " r4000EF - 400

quit

22 ; relay.an%ient4rome.net %losin\$
%onn...

S ! T . return %o " es

2xx OK

Exx sen " more " ata

Fxx tem&orary failure

Gxx &ermanent failure

! essa\$e Filterin\$ for S&am an " Qirus

An email en%ounterin\$ a firewall may #e teste " #y s&am an " virus filters #efore it is allowe " to &ass insi" e the firewall.

This filters usually test to see if the messa\$e Jualifies as s&am or malware.

f this messa\$e %ontains malware or virus it's Juarantine " an " the sen " er notifie " .

A messa\$e that makes it throu\$h the filters \$ets " elivere " via the ! 1A to the %orre%t mail#o/.

This %an then #e then #e retrieve " #y the !) A

2.0< Some messa\$es i" entifie " as s&am %an #e " elete " without sen " er notifi%ation.

AF : s

- Documents that define email standards are called "Request For Comments" (RFCs). Examples include RFC 822 (Text-based Message Format) and RFC 2822 (Internet Message Format).

Troubleshooting Email Issues

- **transient failures**

If a transient error occurs, the ! TA will hang onto the message & eventually retrying the delivery until it either succeeds or fails or until the ! TA decides that the transient issue is really a permanent condition.

- **permanent failures**

If the ! TA cannot deliver the message *it has received a fatal error message or failure to complete the transfer after repeated attempts it queues the message back to the sender. If the sender is a mailing list the queue may be handled by automate queue handling software.

Any 7 questionsR