Email – Electronic Mail

Electronic mail paradigm

Most heavily used application on any network

Electronic version of paper-based office memo

Quick, low-overhead written communication

Dates back to time-sharing systems, in 1960s

Because e-mail is encoded in an electronic medium, new forms of interaction are possible

Automatic processing - sorting, reply

Can carry other content: if basic Simple Mail Transfer Protocol (SMTP), based on TCP/IP, was delivering only simple text messages, by use of Multi-purpose Internet Mail Extension (MIME) now have delivery of other types of data (voice, images, video clips, …)

History: first standard: CCITT X.400 - too complex; base for OSI’s MOTIS application; replaced by TCP/IP based standards RFC 821 (transmission protocol), and RFC 822 (message format)
Email Architecture

Components:

– user agents: “mail readers”
  - composing, editing, reading the mail messages (Pine, Netscape Messenger, Outlook)
  - outgoing & incoming messages are stored on server

– message transfer agents

Mail Transfer Agent, used to move mail (ex. Sendmail)

Mail Delivery Agent, used to deliver mail

  Sendmail, POP (Post Office Protocol [RFC 1939]), IMAP (Internet Mail Access Protocol [RFC 1730])

– mail servers:
  • mailbox contains incoming messages (yet to be read) for user
  • message queue of outgoing (to be sent) mail messages

– SMTP (Simple Mail Transfer Protocol): protocol between mail servers, for delivery to receiver’s server
  • client: the sending mail server, server: receiving mail server
Mailbox: Electronic mailbox

E-mail users have an electronic mailbox into which incoming mail is deposited; identified by an e-mail address.

User then accesses mail with a mail reader program.

Usually associated with computer account (user's account ID); one user may have different electronic mailboxes.

On networked multi-user computer, need to identify computer as well as mailbox:
e-mail address is composed of computer name and mailbox name.

Mail transfer

• E-mail communication is really a two-part process:
  • User composes mail with an e-mail interface program
  • Mail transfer program delivers mail to destination
    • Waits for mail to be placed in outgoing message queues
    • Picks up message and determines recipient(s)
    • Becomes client and contacts server on recipient's computer
    • Passes message to server for delivery
Mail Transfer Illustration
Mail access protocols

• TCP/IP protocol suite includes *Post Office Protocol* (POP) for remote mailbox access; useful for dialup connection
• Can download all mail at once and read off-line
• Can compose mail off-line and mail in one connection
  • Computer with mailboxes runs POP server
  • User runs POP client on local computer
  • POP client can access and retrieve messages from mailbox
• Requires authentication (password)
• Local computer uses SMTP for outgoing mail
Mail access protocols
**Simple Mail Transfer Protocol** (SMTP) is the standard application protocol for delivery of mail from source to destination

- Provides reliable delivery of messages
- Uses TCP well-known port 25 for message exchange between client and server
- Command/Response interaction:
  - commands: ASCII text
  - response: status code and phrase

Other functions:

- E-mail address lookup & address verification

**General characteristics**

Three phases of transfer: handshaking, mail transfer, closure

Attempts to provide reliable service

No guarantee to recover lost messages

No end to end acknowledgement to originator

Error indication delivery not guaranteed

Generally considered reliable!
SMTP Mail Flow

(a) Outgoing Mail

User Agent

Header

Message body

Header

Header

SMTP Sender

TCP to port 25 on foreign SMTP receiver

User Mailboxes

SMTP Receiver

TCP from foreign SMTP sender to local port 25

(b) Incoming Mail
Mail Message Contents

Each queued message has in composition:

Message text

   RFC 822 header with: message envelope and list of recipients

   Message body, composed by user

A list of mail destinations

   Derived by user agent from header

   May be listed in header

   May require expansion of mailing lists

   May need replacement of mnemonic names with mailbox names

If Blind Carbon Copies (BCC) indicated, user agent needs to prepare correct message format
SMTP Sender
Takes message from queue
Transmits to proper destination host
  Via SMTP transaction
  Over one or more TCP connections to port 25
Host may have multiple senders active
Host should be able to create receivers on demand
When delivery complete, sender deletes destination from list for that message
When all destinations processed, message is deleted
  Optimization
If message destined for multiple users on a given host, it is sent only once
  Delivery to users handled at destination host
If multiple messages ready for given host, a single TCP connection can be used
  Saves overhead of setting up and dropping connection
Possible Errors:

- Host unreachable
- Host out of operation
- TCP connection fail during transfer
- Sender can re-queue mail
  - Give up after a period
- Faulty destination address
  - User error
  - Target user changed address
  - Redirect if possible
  - Inform user if not delivered

SMTP Protocol - Reliability

- Used to transfer messages from sender to receiver over TCP connection
- Attempts to provide reliable service
- No guarantee to recover lost messages
- No end to end acknowledgement to originator
- Error indication delivery not guaranteed

**Generally considered reliable!**
SMTP Receiver
Accepts arriving message
Places in user mailbox or copies to outgoing queue for forwarding
Receiver must:
  - Verify local mail destinations
  - Deal with errors
    - Transmission
    - Lack of disk space
Sender responsible for message until receiver confirm complete transfer
  - Indicates mail has arrived at host, not user

SMTP Forwarding
Mostly direct transfer from sender host to receiver host
May go through intermediate machine via forwarding capability
  - Sender can specify route
  - Target user may have moved
Format for Text Messages
RFC 882

Message viewed as having envelope and contents

Envelope contains information required to transmit and deliver message

Message is sequence of lines of text

Uses general memo framework

Header usually keyword followed by colon followed by arguments

<table>
<thead>
<tr>
<th>Header</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>To:</td>
<td>Email address(es) of primary recipient(s)</td>
</tr>
<tr>
<td>Cc:</td>
<td>Email address(es) of secondary recipient(s)</td>
</tr>
<tr>
<td>Bcc:</td>
<td>Email address(es) for blind carbon copies</td>
</tr>
<tr>
<td>From:</td>
<td>Person or people who created the message</td>
</tr>
<tr>
<td>Sender:</td>
<td>Email address of the actual sender</td>
</tr>
<tr>
<td>Received:</td>
<td>Line added by each transfer agent along the route</td>
</tr>
<tr>
<td>Return-Path:</td>
<td>Can be used to identify a path back to the sender</td>
</tr>
</tbody>
</table>

RFC 822 header fields related to message transport.

<table>
<thead>
<tr>
<th>Header</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date:</td>
<td>The date and time the message was sent</td>
</tr>
<tr>
<td>Reply-To:</td>
<td>Email address to which replies should be sent</td>
</tr>
<tr>
<td>Message-Id:</td>
<td>Unique number for referencing this message later</td>
</tr>
<tr>
<td>In-Reply-To:</td>
<td>Message-Id of the message to which this is a reply</td>
</tr>
<tr>
<td>References:</td>
<td>Other relevant Message-Ids</td>
</tr>
<tr>
<td>Keywords:</td>
<td>User chosen keywords</td>
</tr>
<tr>
<td>Subject:</td>
<td>Short summary of the message for the one-line display</td>
</tr>
</tbody>
</table>

Some fields used in the RFC 822 message header.
Multipurpose Internet Mail Extension (MIME)

Extension to RFC 822 for message format; given in RFC 2045, 2056
Additional lines in message header declare MIME content type

Five new message header fields

**MIME version**

**Content type**

**Content transfer encoding**

**Content Description**

**Content Id**

<table>
<thead>
<tr>
<th>Header</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIME-Version:</td>
<td>Identifies the MIME version</td>
</tr>
<tr>
<td>Content-Description:</td>
<td>Human-readable string telling what is in the message</td>
</tr>
<tr>
<td>Content-Id:</td>
<td>Unique identifier</td>
</tr>
<tr>
<td>Content-Transfer-Encoding:</td>
<td>How the body is wrapped for transmission</td>
</tr>
<tr>
<td>Content-Type:</td>
<td>Nature of the message</td>
</tr>
</tbody>
</table>

* RFC 822 headers added by MIME.
Message Format: Multimedia Extensions

From: xyz@myServer.edu
To: abc@mailServer.iitb.edu
Subject: Picture.
MIME-Version: 1.0
Content-Transfer-Encoding: base64
Content-Type: image/jpeg

base64 encoded data ..... 
............................ 
...... base64 encoded data .

MIME version
method used to encode data
encoded data
• Extends and automates encoding mechanisms - *Multipart Internet Mail Extensions*

• Allows inclusion of separate components - programs, pictures, audio clips - in a single mail message (see next table)

• Sending program identifies the components, so receiving program can automatically extract and inform mail recipient

• Separator line gives information about specific encoding

• MIME is extensible - sender and receiver agree on encoding scheme

• MIME is compatible with existing mail systems

• Everything encoded as ASCII

• Headers and separators ignored by non-MIME mail systems

• MIME *encapsulates* binary data in ASCII mail envelope
<table>
<thead>
<tr>
<th>Type</th>
<th>Subtype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text</td>
<td>Plain</td>
<td>Unformatted text</td>
</tr>
<tr>
<td></td>
<td>RichText</td>
<td>Text including simple formatting commands</td>
</tr>
<tr>
<td>Image</td>
<td>Gif</td>
<td>Still picture in GIF format</td>
</tr>
<tr>
<td></td>
<td>Jpeg</td>
<td>Still picture in JPEG format</td>
</tr>
<tr>
<td>Audio</td>
<td>Basic</td>
<td>Audible sound</td>
</tr>
<tr>
<td>Video</td>
<td>Mpeg</td>
<td>Movie in MPEG format</td>
</tr>
<tr>
<td>Application</td>
<td>Octet-stream</td>
<td>An uninterpreted byte sequence</td>
</tr>
<tr>
<td></td>
<td>Postscript</td>
<td>A printable document in PostScript</td>
</tr>
<tr>
<td>Message</td>
<td>Rfc822</td>
<td>A MIME RFC 822 message</td>
</tr>
<tr>
<td></td>
<td>Partial</td>
<td>Message has been split for transmission</td>
</tr>
<tr>
<td></td>
<td>External-body</td>
<td>Message itself must be fetched over the net</td>
</tr>
<tr>
<td>Multipart</td>
<td>Mixed</td>
<td>Independent parts in the specified order</td>
</tr>
<tr>
<td></td>
<td>Alternative</td>
<td>Same message in different formats</td>
</tr>
<tr>
<td></td>
<td>Parallel</td>
<td>Parts must be viewed simultaneously</td>
</tr>
<tr>
<td></td>
<td>Digest</td>
<td>Each part is a complete RFC 822 message</td>
</tr>
</tbody>
</table>

The MIME types and subtypes defined in RFC 1521.
MIME Transfer Encodings

Reliable delivery across wide largest range of environments

*Content transfer encoding* field takes on six values (see next table):

- Three of them (7bit, 8bit, binary): no encoding done
  - Provide some info about nature of data
  - SMTP transfer uses 7bit form

**Quoted-printable**

- Data contains largely printable ASCII characters
  - Non-printing characters represented by hex code

**Base64 (Radix-64 Encoding)**

- Maps arbitrary binary input (6 bits) onto printable output 8 bit characters

**X-token**

- Named nonstandard encoding
<table>
<thead>
<tr>
<th>Encoding</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7bit</td>
<td>The data are all represented by short lines of ASCII characters.</td>
</tr>
<tr>
<td>8bit</td>
<td>The lines are short, but there may be non-ASCII characters (octets with the high-order bit set).</td>
</tr>
<tr>
<td>binary</td>
<td>Not only may non-ASCII characters be present but the lines are not necessarily short enough for SMTP transport.</td>
</tr>
<tr>
<td>quoted-printable</td>
<td>Encodes the data in such a way that if the data being encoded are mostly ASCII text, the encoded form of the data remains largely recognizable by humans.</td>
</tr>
<tr>
<td>base64</td>
<td>Encodes data by mapping 6-bit blocks of input to 8-bit blocks of output, all of which are printable ASCII characters.</td>
</tr>
<tr>
<td>x-token</td>
<td>A named nonstandard encoding.</td>
</tr>
</tbody>
</table>

**MIME Transfer Encodings**

**Base64 encoding scheme**

04/11/2008