Electronic Mail (Email)

Kameswari Chebrolu

Background

- One of Internet's oldest (1971) and very widely used application
- Provides an asynchronous mode of communication
  - Fast, easy, inexpensive
- Evolved over the years to support many features
  - Attachments, confirmation, group-emails, spam-filters etc

Other Email Services

- CC, BCC, high-priority, alternative recipients
- Mailing lists
- Mailboxes to store and manage emails
- Email forwarding (e.g. when people move away); automated replies

- Filters: Detect spam, Message from friends go to mailbox1 etc

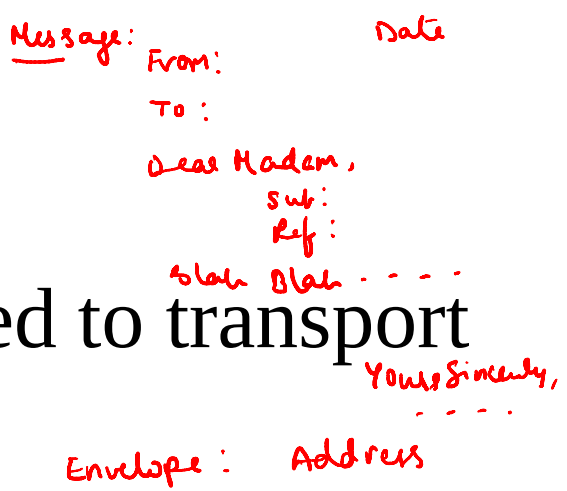
Two Basic Components

- User Agents (Mail Readers)
  - Local programs that help people read, send and manage emails
  - Can be text based or GUI based
  - E.g. pine, elm, Microsoft Outlook, Mozilla Thunderbird

Two Basic Components

- Message Transfer Agents
  - Daemons (processes) that run in background to move messages from source to destination host
  - Agents implement required protocols

Message Format



- Envelope encapsulates the message
- Envelope contains information needed to transport message
  - Destination address, priority, security level etc
  - Message transfer agents use it to route message to correct destination
- Envelope derived from the header fields in the message

RFC 5322 (old RFC 822)

- Message within envelope contains header and body
  - Both are represented in ASCII text
  - Header contains control information for user agents
  - Body is for human consumption
  - Header separated from message body by blank line
- Header is a series of <CRLF> terminated lines
  - Contains type and value separated by a colon
    - From: , To:, Date: , Message-ID, , CC: , BCC: , Sender: , Subject:, Received: , Return-Path:, Reply-To: etc

Message Body and MIME

- Early days, email was made of English text messages, expressed in ASCII
  - No support for foreign languages
  - No support for non-text attachments (pdf, doc, jpg, audio files etc)
- Solution: Multipurpose Internet Mail Extensions (MIME)
  - Additional headers
  - Define content types and subtypes
  - Add structure to message body
  - Encoding rules for non ASCII messages (convert them to ASCII)

Headers added by MIME

Header	Meaning
MIME-Version	Identifies the MIME Version
Content-Description	ASCII string that tells what is in the message
Content-ID	Unique identifier
Content-Type	Type of data contained in the message
Content-Transfer-Encoding	How the data in message is encoded (e.g. 7bit, Base64)

Content-Type:

Text/Plain – Unformatted text
Text/Enriched – Simple formatted text
Image/Jpeg – Picture in jpeg format
Video/Mpeg – Video in mpeg format
Application/Msword – word document
Message/Mixed – Message made of independent pieces (each piece has own header line)

Example

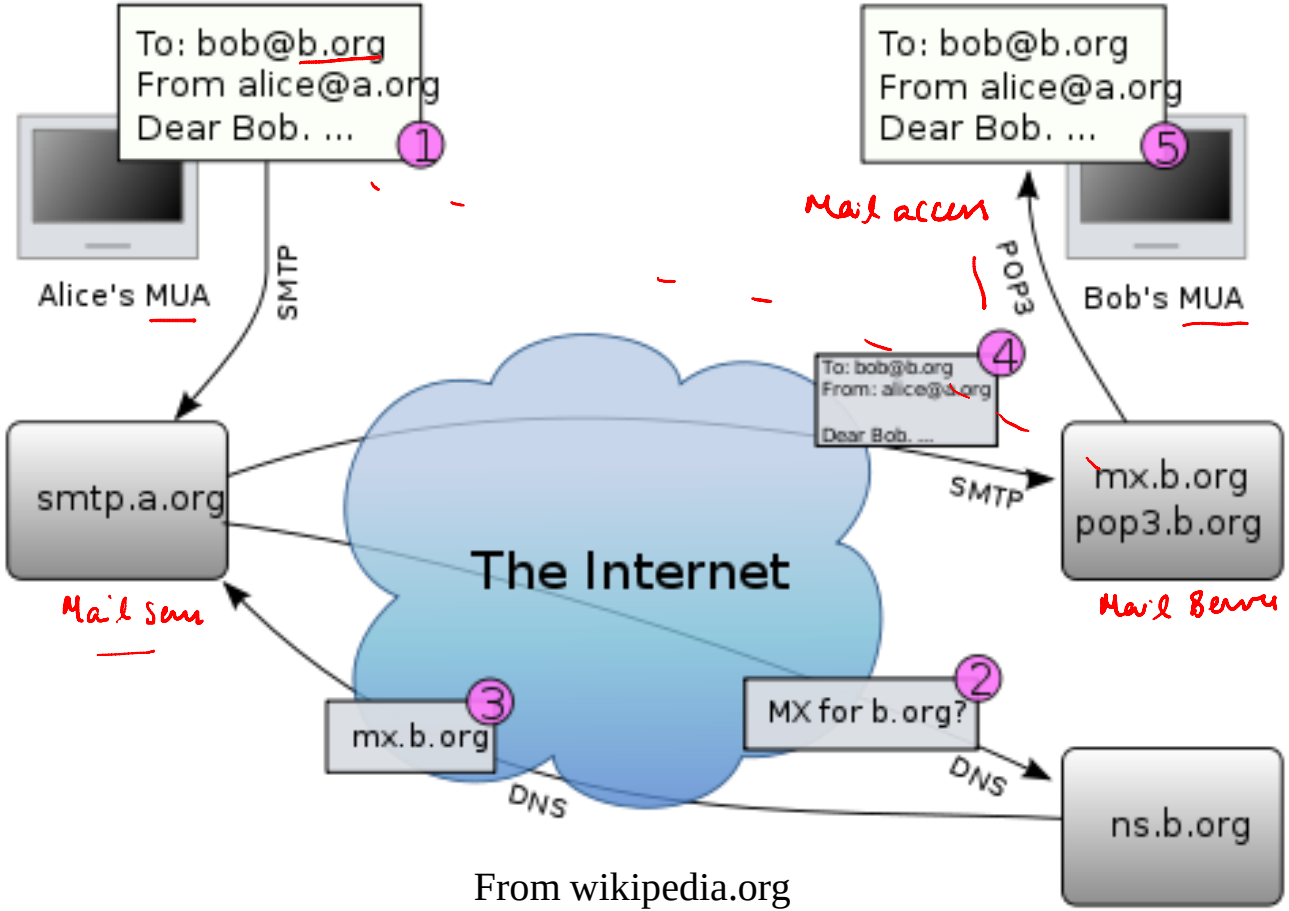
- Look at the uploaded message format

Implementation Choices

- First email system was based on file transfer protocol
  - No internal structure; Many features missing
- How about direct connection between sending and receiving host?
  - Receiver machine may not always be on
  - User may want to retrieve mail from multiple machines
- Need always-on hosts → Shared Mail Servers

Architecture

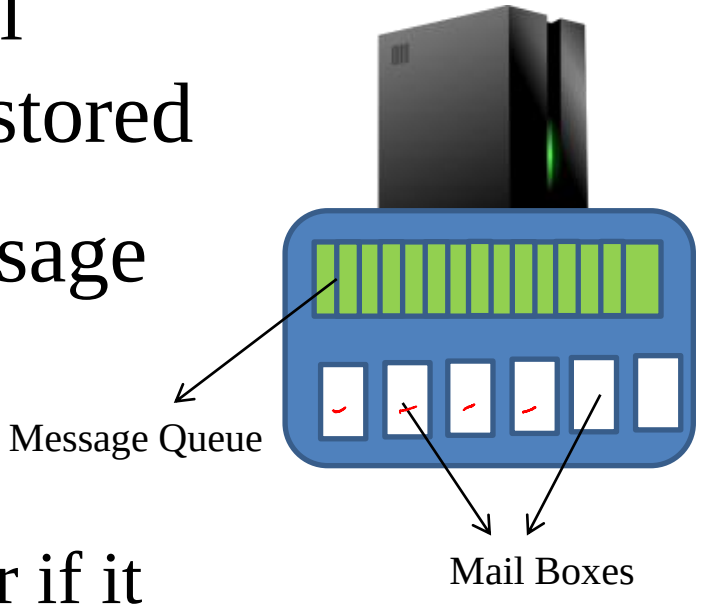
- Sending mails is a PUSH operation
  - SMTP protocol
- Receiving mails is a PULL operation
  - POP3, IMAP, HTTP



From wikipedia.org

Mail Servers

- Each user has a mailbox in a mail server, where user messages are stored
- Mail server also maintains a message queue of outgoing messages
  - In case of failure, attempts retransmissions and informs sender if it drops the message
- Both client and server side of SMTP run on a mail server



Summary

- Email is one of the oldest and a very popular application
- Enabled by User agents and Message transfer agents
- Email Messages are ASCII based and made up of header fields and body
- Architecture based on mail servers who move messages between hosts
  - Mail transfer protocol: SMTP
  - Mail access protocol: POP3/IMAP/HTTP