Mail Message Format

Vermont Technical College
Peter C. Chapin

Mail Protocol Stack

Describes a structure for the body of complex messages.

MIME

Mail Message

Describes the format of mail messages themselves.

These slides

Protocol for sending mail messages over the network.

SMTP

Transport Layer Protocol (TCP)

(reliable, stream oriented connection)

RFC-5322

- Internet Message Format

  - Latest version... obsoletes RFC-2822, which obsoletes RFC-822.

  - Companion to RFC-5321 (SMTP, which obsoletes RFC-2821 which obsoletes RFC-821).

- Read it!

SMTP is Low Level

- ## SMTP messages.

  - ### When SMTP sends a message, it does not care about the message structure.

    - Just sends a collection of US-ASCII text lines.
    - Does not consider the message content or form.

- ## Mail messages.

  - ### RFC-5322 does not care about how the messages are sent.

    - Just describes the message headers, etc.
   - Any form of message transport is fine.

Overall Message Structure

- "Plain" text
  - *Only* US-ASCII characters are allowed!
    - No non-ASCII character supported.
    - No binary data supported.
    - What??
      - MIME provides a way of encoding arbitrary characters and binary data.

- Line lengths (including CR/LF)
  - Lines MUST be limited to 1000 characters.
    - SMTP limitation.
  - Lines SHOULD be limited to 80 characters.

Overall Message Structure (cont)

One line for each SMTP server. **Layer Violation!**

One completely blank line separates headers from body. This line cannot even contain any spaces.

SMTP Envelope
RFC-5322 Headers
Message Body (structured by MIME)
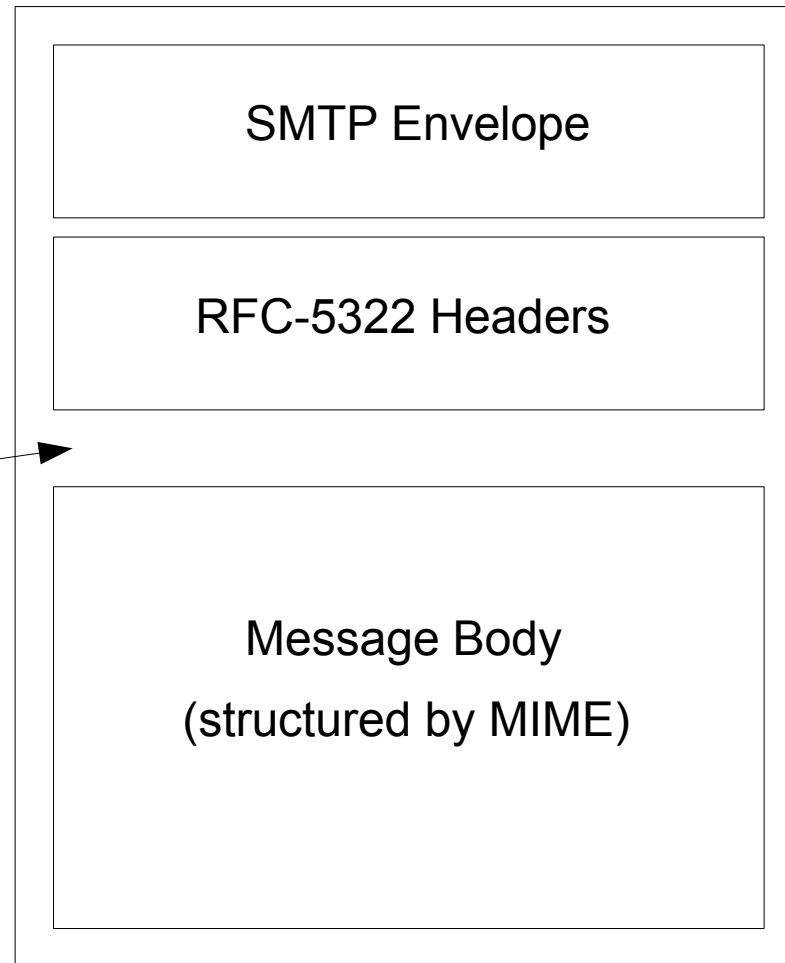
From, To, Subject, etc.

Body consists of lines of ASCII text.

Raw vs Cooked Messages

- RFC-5322 defines a raw format.

  - Most mail programs display messages in a digested ("cooked") form.

  - Normal users don't want to see all the raw details.

    - Provides information they don't want to know

    - Confusing

    - Is often encoded in complex ways (MIME)

      - HTML
      - Binary attachments are encoded
      - Non-ASCII character sets are encoded.

  - Most mail programs do provide an optional raw view.

Long Header Lines

- Lines SHOULD be limited to 80 characters.
  - Long header lines can be wrapped (or "folded") by indenting them.
    - From: Alice <alice@someplace.com>,
      Bob <bob@someotherplace.com>,
      Carol <carol@noplace.org>
    - The amount of indentation is not important.
      - Even a single space is enough.
    - Some mail programs try to indent "nicely" to promote human readability.
      - Even though humans rarely read mail messages raw these days.

Headers

- Each header (or "field") has a name and a body.
  - `Field-Name: BODY`
    - Name and body are separated by a colon.
  - Fields can be "structured" or "unstructured."
    - Structured fields have their own syntax.
      - `Date: Mon, 16 Mar 2009 17:30:00 -0400`
    - Unstructured fields are free text.
      - `Subject: This is pretty much anything.`
  - Structured fields can usually contain comments.
    - `From: Peter <pchapin@vtc.edu> (Ha!)`

Originator Fields

- Fields that describe who sent the message.
  - `From:`
    - Could be a list of addresses
  - `Sender:`
    - Used when sender is different from author (for example a secretary) or when From contains a list.
  - `Reply-To:`
    - Could be a list of addresses
    - Used when replies should go to a different place than From (for example, when replies should go to a mailing list).

Destination Fields

- ## Fields used to describe who gets the message.

  - ### `To:`

    - The addresses to which the mail is sent.

      - Could be different than what SMTP uses in `RCPT TO` command. Thus you can get email apparently not addressed to you.

  - ### `Cc:`

    - Carbon copy recipients.

  - ### `Bcc:`

    - Blind carbon copy recipients.

      - Sending email systems remove Bcc recipient list from messages being delivered to normal recipients.

Identification Fields

- Fields that identify the message itself.
  - `Message-ID:`
    - All messages SHOULD have a globally unique ID.
      - `<stuff@morestuff>`
      - `stuff` might be time stamp + process ID of sending process.
      - `morestuff` might be the sender's domain.
      - Suggested algorithm in the RFC.
  - `In-Reply-To:`
    - Message ID of original message.
      - Used by mail programs to "thread" message displays.
  - `References:`
    - List of all related messages.

Information Fields

- Various fields that contain information about the message.
    - `Subject:`
        - User's summary of the message content.
    - `Comments:`
        - From RFC-5322: "Additional comments on the text of the body of the message."
    - `Keywords:`
        - A comma separated list of keywords related to the message content.
            - Mail programs could use this information for searching purposes.
            - As far as I know, none do this.

Extension Fields

- User defined fields.
  - Any field starting with `X-` is an extension.
    - Example: `X-Priority: 7`
  - Mail programs SHOULD ignore extensions they do not understand.
  - Allows application specific features to be added later.

Example

- RFC-5322 Message...

  - ```
    From: God <god@heaven.org>
    To: Satan <satan@hell.com>
    Subject: Lunch?
    Date: Mon, 19 Mar 2009 10:30:00 -0400
    Reply-To: BigG <iknowbest@gmail.com>

    Hi! Feel like doing lunch today, or are you
    too busy?
    ```

  - Notice...

    - Blank line separating headers from body.

    - Field order is not significant.

    - `Date` field is required.

Demonstration

Show a few real email messages