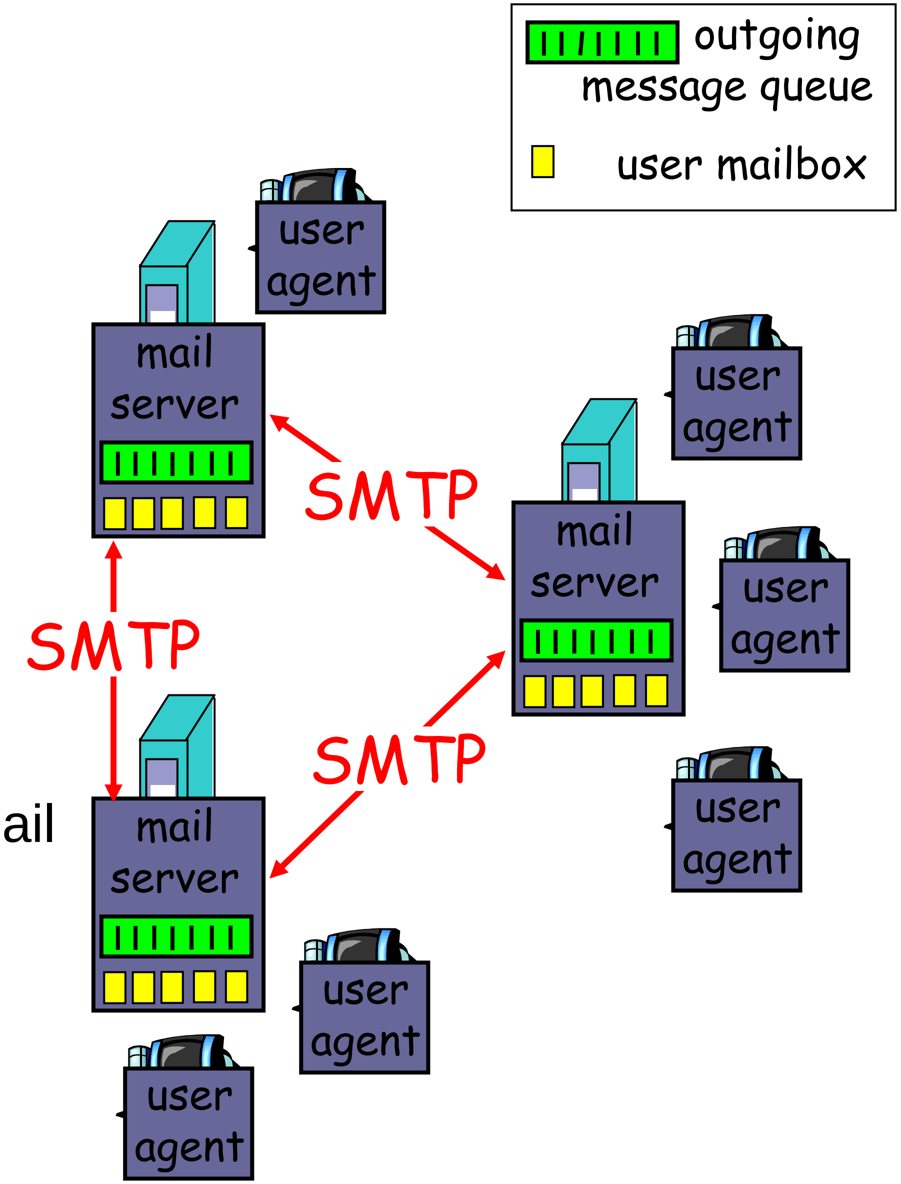
Electronic Mail

Electronic Mail

Three major components:

- user agents
- mail servers
- simple mail transfer protocol: SMTP

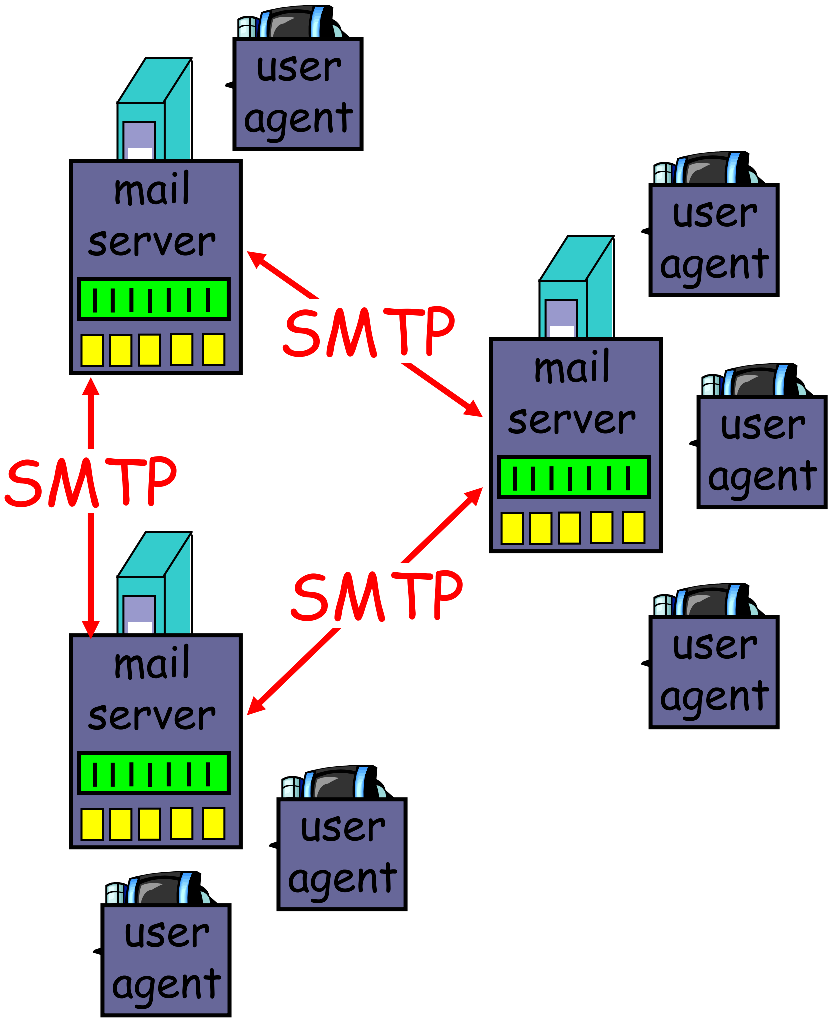
- User Agent
- a.k.a. “mail reader”
- composing, editing, reading mail messages
- e.g., Eudora, Outlook, elm, Netscape Messenger
- outgoing, incoming messages stored on server



Electronic Mail: mail servers

Mail Servers

- mailbox contains incoming messages (yet to be read) for user
- message queue of outgoing (to be sent) mail messages
- SMTP protocol between mail servers to send email messages
  - client: sending mail server
  - "server": receiving mail server



Electronic Mail: SMTP [RFC 821]

- uses TCP to reliably transfer email msg from client to server, port 25
- direct transfer: sending server to receiving server
- three phases of transfer
  - handshaking (greeting)
  - transfer of messages
  - closure
- command/response interaction
  - commands: ASCII text
  - response: status code and phrase
- messages must be in 7-bit ASCII

Sample SMTP interaction

```
S: 220 hamburger.edu
C: HELO crepes.fr
S: 250  Hello crepes.fr, pleased to meet you
C: MAIL FROM: <alice@crepes.fr>
S: 250 alice@crepes.fr... Sender ok
C: RCPT TO: <bob@hamburger.edu>
S: 250 bob@hamburger.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Do you like ketchup?
C:   How about pickles?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection
```

Try SMTP interaction for yourself:

- **`telnet servername 25`**
- see 220 reply from server
- enter HELO, MAIL FROM, RCPT TO, DATA, QUIT commands
- above lets you send email without using email client (reader)

SMTP Example:

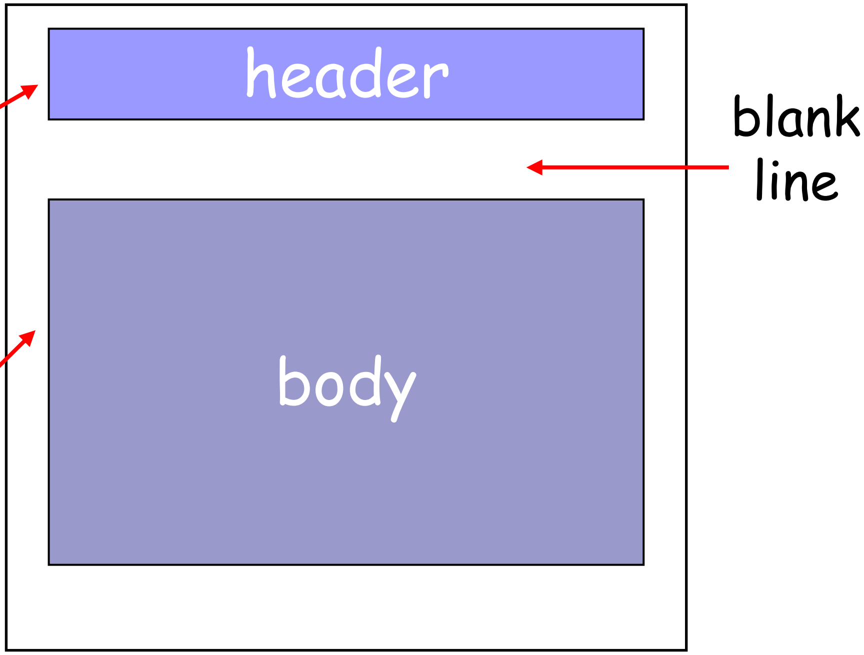
```
[pasward@sandrock pasward]$ telnet www.ynpcsc.gov.cn 25
Trying 202.98.190.193...
Connected to www.ynpcsc.gov.cn.
Escape character is '^]'.
220 www.ynpcsc.gov.cn ESMTP Server (Microsoft Exchange
Internet Mail Service 5.5.2650.21) ready
MAIL FROM: <jqRandom@hotmail.com>
250 OK - mail from <jqRandom@hotmail.com>
RCPT TO: <pasward@big.uwaterloo.ca>
250 OK - Recipient <pasward@big.uwaterloo.ca>
DATA
354 Send data.  End with CRLF.CRLF
yet another open relay
.
250 OK
Quit
221 closing connection
Connection closed by foreign host
```

And this was e-mailed to me:

```
pasward@big pasward]$ cat /var/spool/mail/pasward
From jqRandom@hotmail.com  Wed Mar  6 12:57:00 2002
Return-Path: <jqRandom@hotmail.com>
Received: from www.ynpcsc.gov.cn ([202.98.190.193])
  by big.uwaterloo.ca (8.11.0/8.11.0) with ESMTP id g26HuvY23119
  for <pasward@big.uwaterloo.ca>;
Wed, 6 Mar 2002 12:56:58 -0500
Date: Wed, 6 Mar 2002 12:56:58 -0500
From: jqRandom@hotmail.com
Message-Id: <200203061756.g26HuvY23119@big.uwaterloo.ca>
Received: from sandrock.uwaterloo.ca ([129.97.105.32])
  by www.ynpcsc.gov.cn with SMTP (Microsoft Exchange Internet
    Mail Service Version 5.5.2650.21) id GH8G9YG8;
  Thu, 7 Mar 2002 01:58:55 +0800
yet another open relay
```

Mail message format

- SMTP: protocol for exchanging email msgs
- RFC 822: standard for text message format:
  - header lines, e.g.,
- To:
- From:
- Subject:
- *<span>different</span> from SMTP commands*!
- body
- the “message”, ASCII characters only



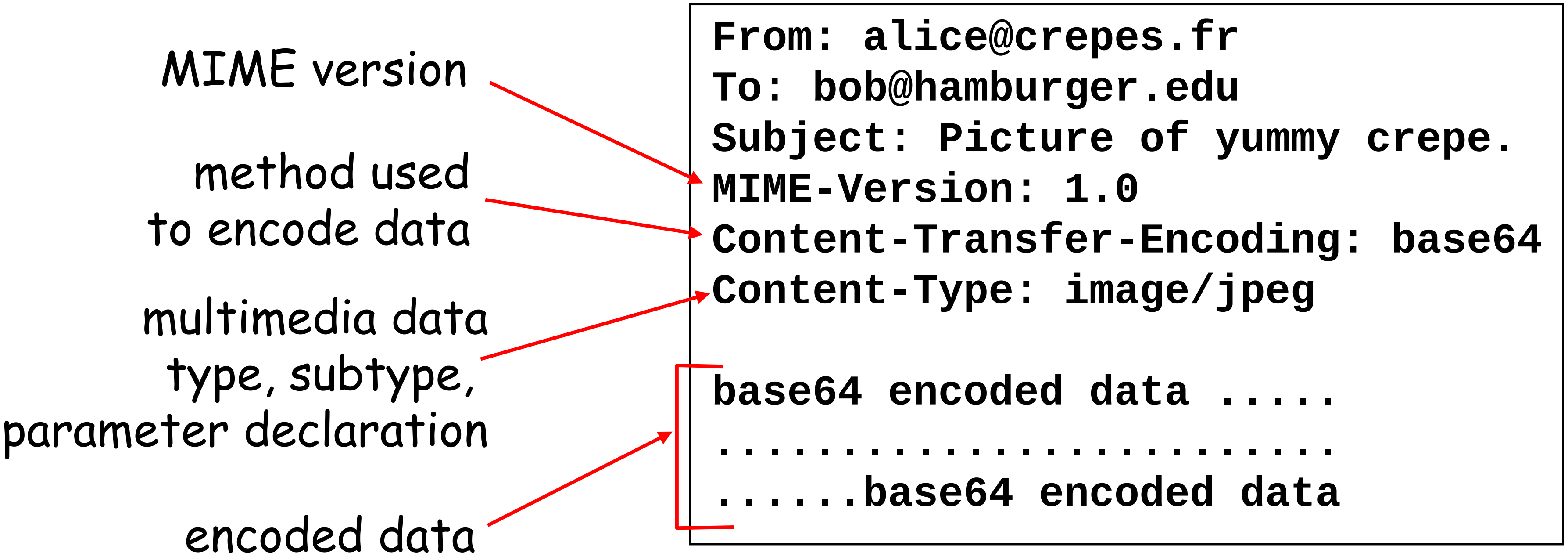
MIME – Multipurpose Internet Mail Extensions

Problems with international languages:

- Languages with accents (French, German).
- Languages in non-Latin alphabets (Hebrew, Russian).
- Languages without alphabets (Chinese, Japanese).
- Messages not containing text at all (audio or images).

Message format: multimedia extensions

- MIME: multimedia mail extension, RFC 2045, 2056
- Additional lines in msg header declare MIME content type



MIME version

method used to encode data

multimedia data type, subtype, parameter declaration

encoded data

```
From: alice@crepes.fr
To: bob@hamburger.edu
Subject: Picture of yummy crepe.
MIME-Version: 1.0
Content-Transfer-Encoding: base64
Content-Type: image/jpeg

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

MIME types

`Content-Type: type/subtype; parameters`

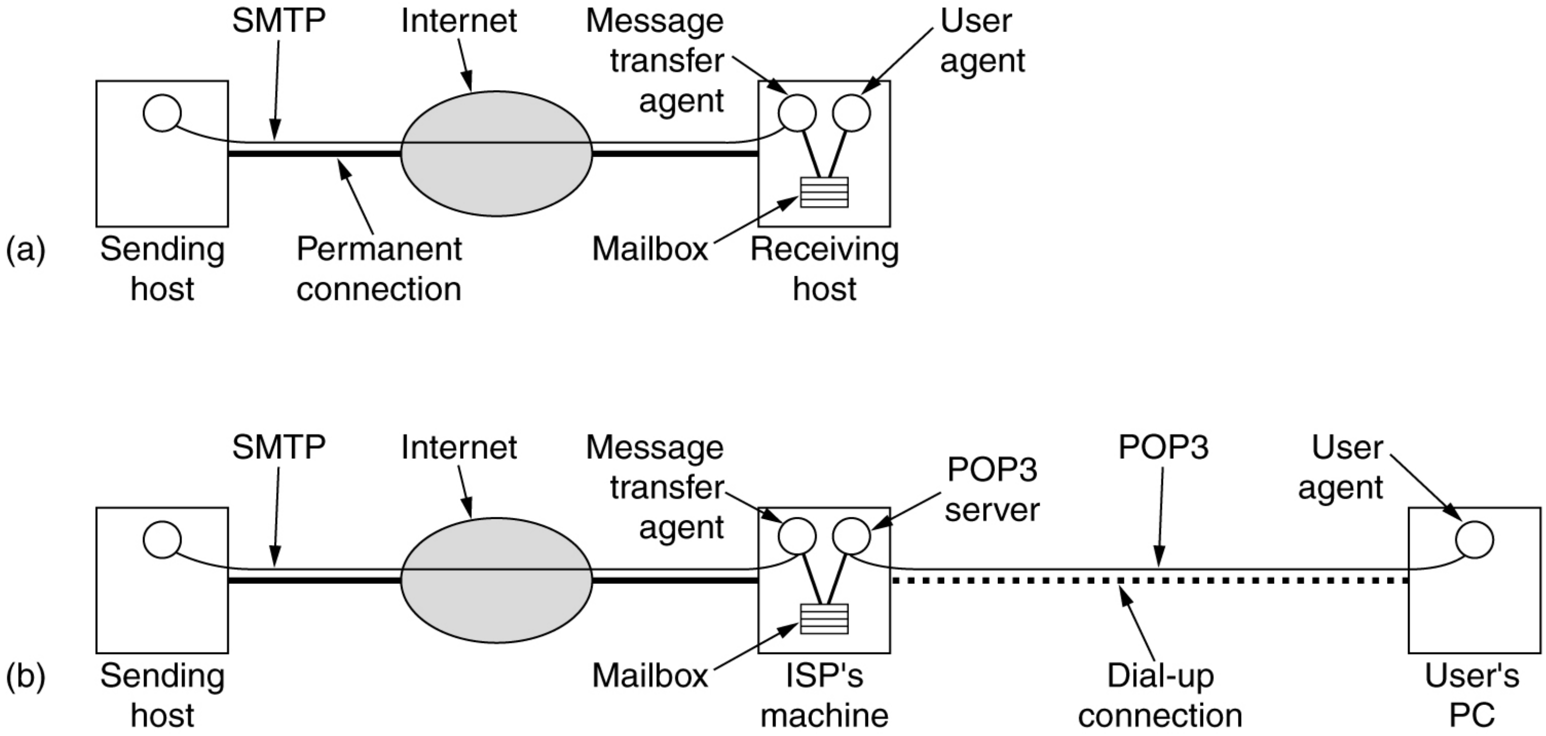
- Text
  - example subtypes: **`plain`**, **`html`**
- Image
  - example subtypes: **`jpeg`**, **`gif`**
- Audio
  - exampe subtypes: **`basic`** (8-bit mu-law encoded), **`32kadpcm`** **(32 kbps coding)**

- Video
  - example subtypes: **`mpeg`**, **`quicktime`**
- Application
  - other data that must be processed by reader before "viewable"
  - example subtypes: **`msword`**, **`octet-stream`**

Multipart Type

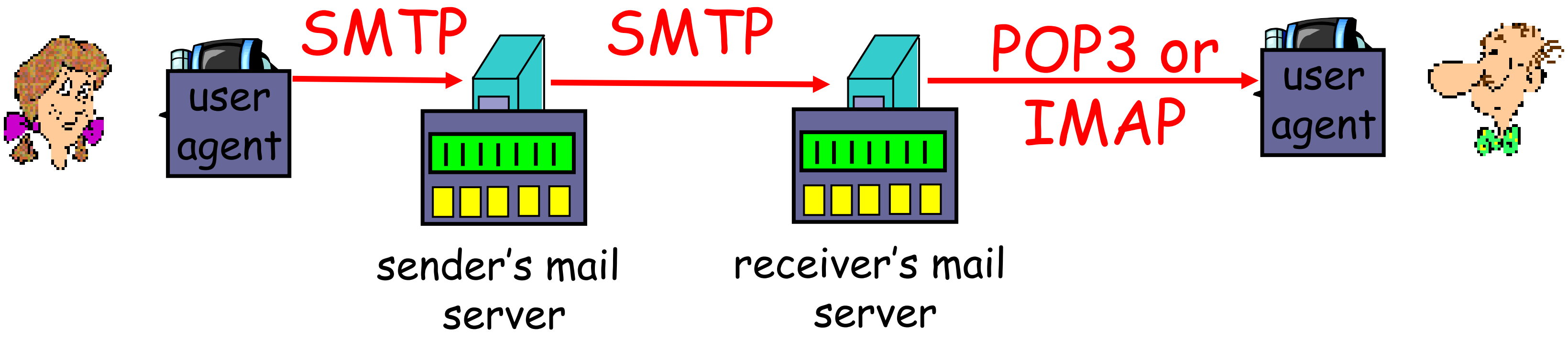
```
From: alice@crepes.fr
To: bob@hamburger.edu
Subject: Picture of yummy crepe.
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary=98766789

--98766789
Content-Transfer-Encoding: quoted-printable
Content-Type: text/plain

Dear Bob,
Please find a picture of a crepe.
--98766789
Content-Transfer-Encoding: base64
Content-Type: image/jpeg

base64 encoded data .....
.........................
......base64 encoded data
--98766789--
```

Final Delivery



(a) Sending and reading mail when the receiver has a permanent Internet connection and the user agent runs on the same machine as the message transfer agent. (b) Reading e-mail when the receiver has a dial-up connection to an ISP.

Mail access protocols



- SMTP: delivery/storage to receiver's server
- Mail access protocol: retrieval from server
  - POP: Post Office Protocol [RFC 1939]
    - authorization (agent <-->server) and download
  - IMAP: Internet Mail Access Protocol [RFC 1730]
    - more features (more complex)
    - manipulation of stored msgs on server
  - HTTP: Hotmail , Yahoo! Mail, etc.

POP3 protocol

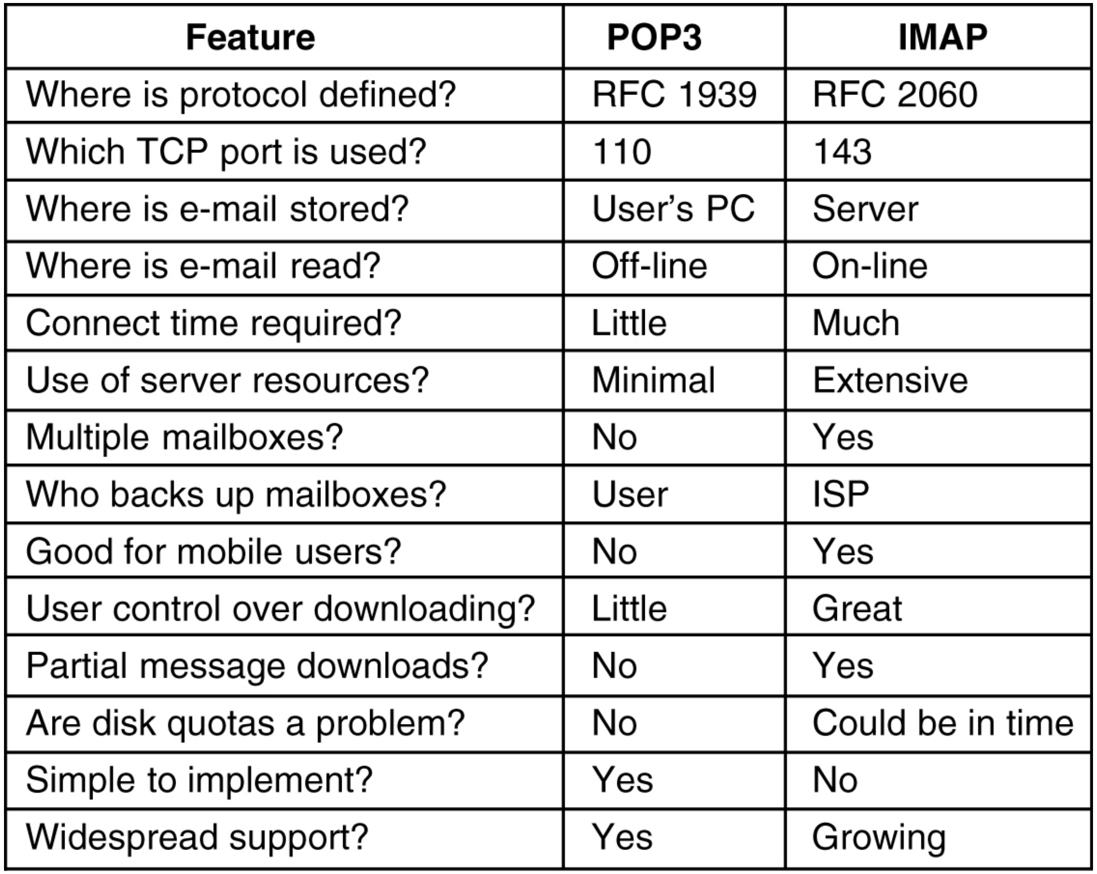
authorization phase

- client commands:
  - **user:** declare username
  - **pass:** password
- server responses
  - **+OK**
  - **-ERR**
- transaction phase, client:
- **list:** list message numbers
- **retr:** retrieve message by number
- **dele:** delete
- **quit**

```
S: +OK POP3 server ready
C: user alice
S: +OK
C: pass hungry
S: +OK user successfully logged on
```

```
C: list
S: 1 498
S: 2 912
S: .
C: retr 1
S: <message 1 contents>
S: .
C: dele 1
C: retr 2
S: <message 1 contents>
S: .
C: dele 2
C: quit
S: +OK POP3 server signing off
```

A comparison of POP3 and IMAP.



Feature	POP3	IMAP
Where is protocol defined?	RFC 1939	RFC 2060
Which TCP port is used?	110	143
Where is e-mail stored?	User’s PC	Server
Where is e-mail read?	Off-line	On-line
Connect time required?	Little	Much
Use of server resources?	Minimal	Extensive
Multiple mailboxes?	No	Yes
Who backs up mailboxes?	User	ISP
Good for mobile users?	No	Yes
User control over downloading?	Little	Great
Partial message downloads?	No	Yes
Are disk quotas a problem?	No	Could be in time
Simple to implement?	Yes	No
Widespread support?	Yes	Growing