



CEU

6 Capa de Aplicación

6.2 Correo electrónico

RdE 2014-2015

6 Guión del Tema 6

- 6. Capa de transporte:
 - 6.1 DNS.
 - **6.2 Correo electrónico.**
 - 6.3 WWW.
 - 6.4 Multimedia.

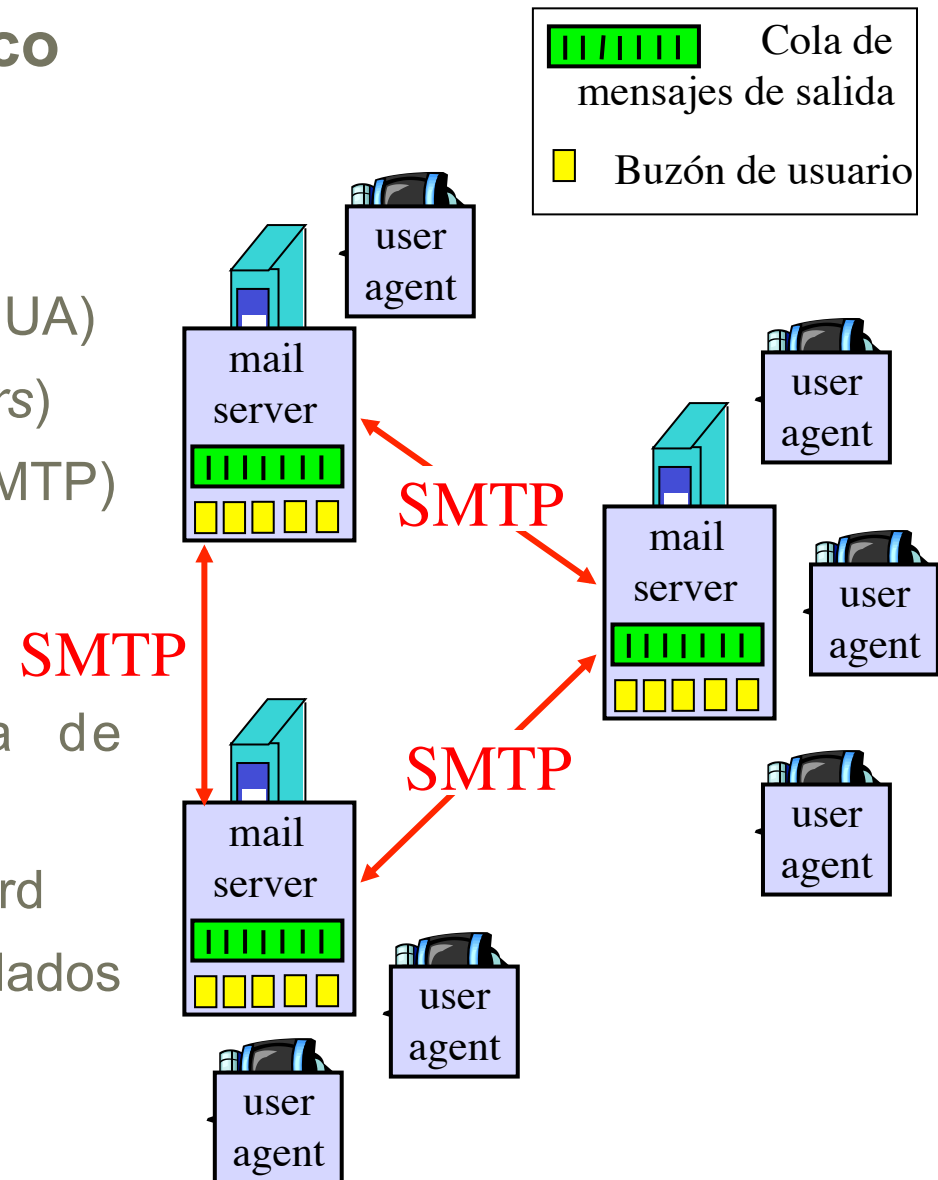
6.2 Correo Electrónico

3 Componentes principales

- Agentes de usuario (*user agents* UA)
- Servidores de correo (*mail servers*)
- *Simple Mail Transfer Protocol* (SMTP)

Cliente de correo

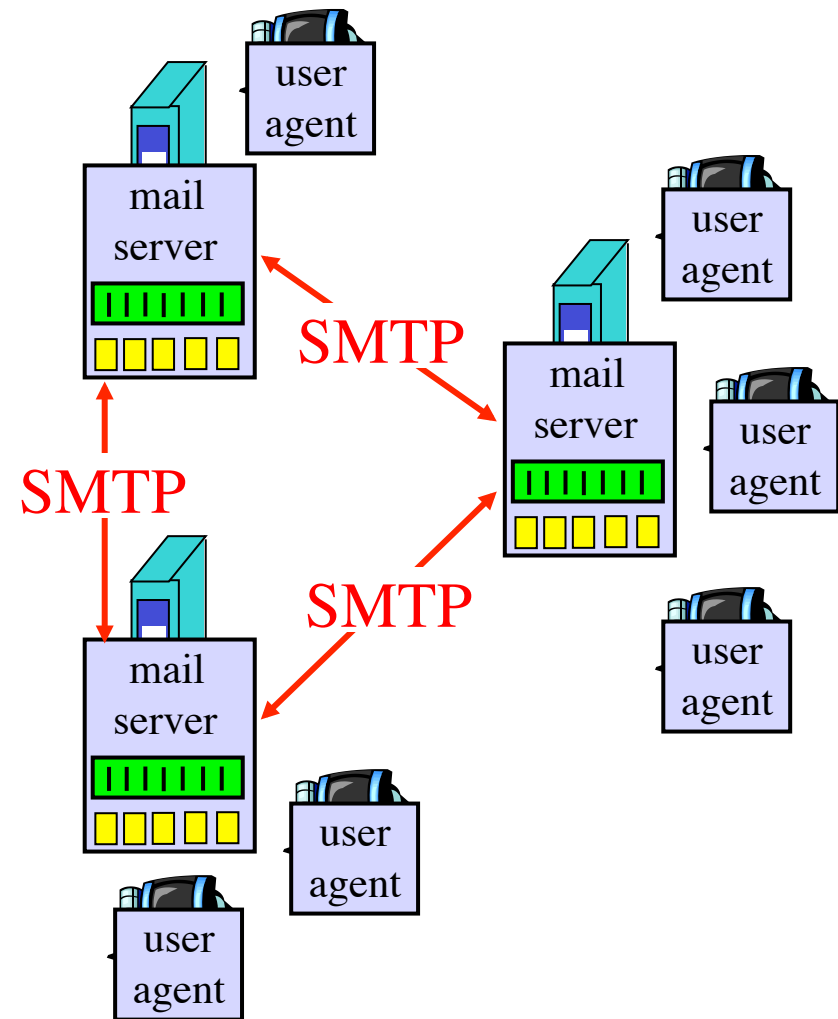
- Redacción, edición, lectura de mensajes de correo
- P.e., Eudora, Outlook, Thunderbird
- Lectura de los mensajes guardados en el servidor



6.2 Servidores de correo

Servidores de Correo

- Buzón conteniendo los mensajes entrantes para el usuario
- Cola de mensajes de correo de salida (para ser enviados)
- Protocolo SMTP entre los servidores y clientes de correo:
 - ☐ Cliente: envía al servidor de correo
 - ☐ Servidor: recibe correo

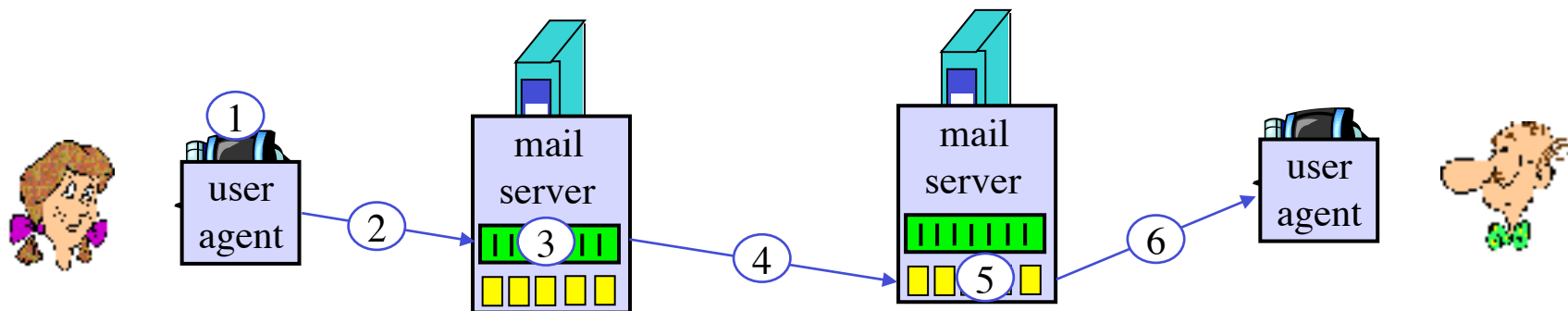


6.2 SMTP [RFC 2821]

- Usa TCP para la transferencia confiable de mensajes de correo desde el cliente al servidor, puerto 25
- Transferencia directa: desde el cliente al servidor
- Tres fases de transferencia (protocolo orientado a conexión):
 - ☐ *handshaking* (conexión)
 - ☐ Transferencia de mensajes
 - ☐ Cierre
- Interacción en forma de comandos y respuestas:
 - ☐ Comandos: texto ASCII
 - ☐ Respuesta: código de estado y frase
- Los mensajes deben estar en ASCII de 7 bit

6.2 Ejemplo: Alice envía un mensaje a Bob

- 1) Alice usa un UA para componer un mensaje “para” bob@ceu.es
- 2) El UA de Alice envía el mensaje a su servidor de correo; el mensaje se coloca en la cola de mensajes
- 3) El lado cliente de SMTP abre una conexión TCP con el servidor de correo de Bob
- 4) El cliente SMTP envía el mensaje de Alice sobre la conexión TCP
- 5) El servidor de correo de Bob coloca el mensaje en el buzón de Bob
- 6) El UA de Bob recibe el mensaje



6.3 Ejemplo de interacción SMTP

```
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
```

6.3 Ejemplo de interacción SMTP

- `telnet servername 25`
- Véase la respuesta 220 reply del servidor
- Introducir los comandos HELO, MAIL FROM, RCPT TO, DATA, QUIT

6.2 Características y comparación con HTTP

Características de SMTP:

- SMTP usa conexiones persistentes
- SMTP requiere que el mensaje esté en ASCII de 7-bit
- El servidor SMTP usa CRLF.CRLF para determinar el final del mensaje

Comparación con HTTP:

- HTTP: *pull*, el cliente pide
- SMTP: *push*, el cliente pone
- Ambos tiene comandos/respuestas ASCII con códigos de estado
- HTTP: cada objeto encapsulado tiene su propio mensaje de respuesta
- SMTP: múltiples objetos enviados en mensajes multiparte

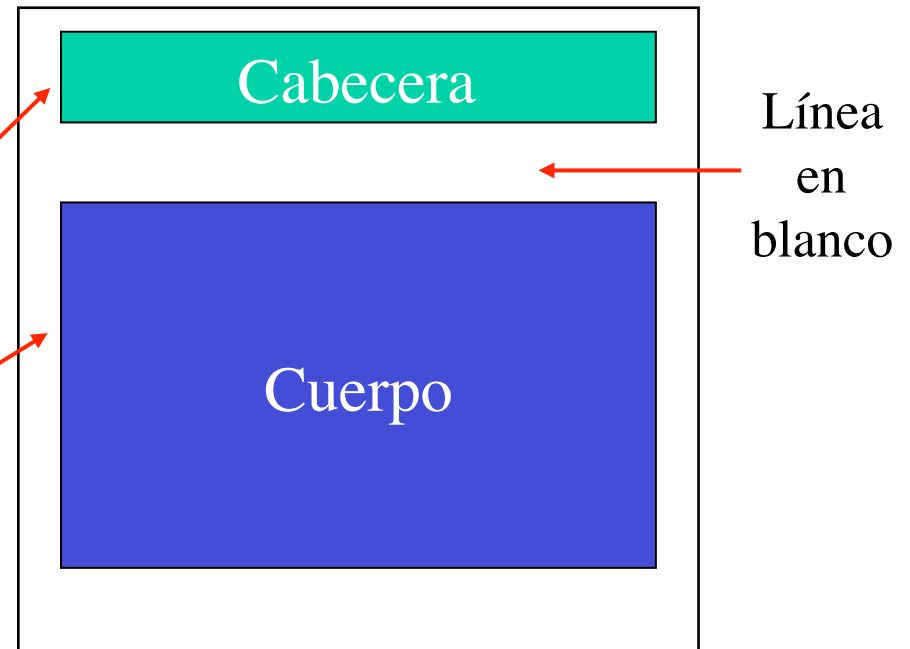
6.2 Formato del mensaje de correo

SMTP: Protocolo para el intercambio de mensajes

RFC 822: estándar de formato para mensajes de texto:

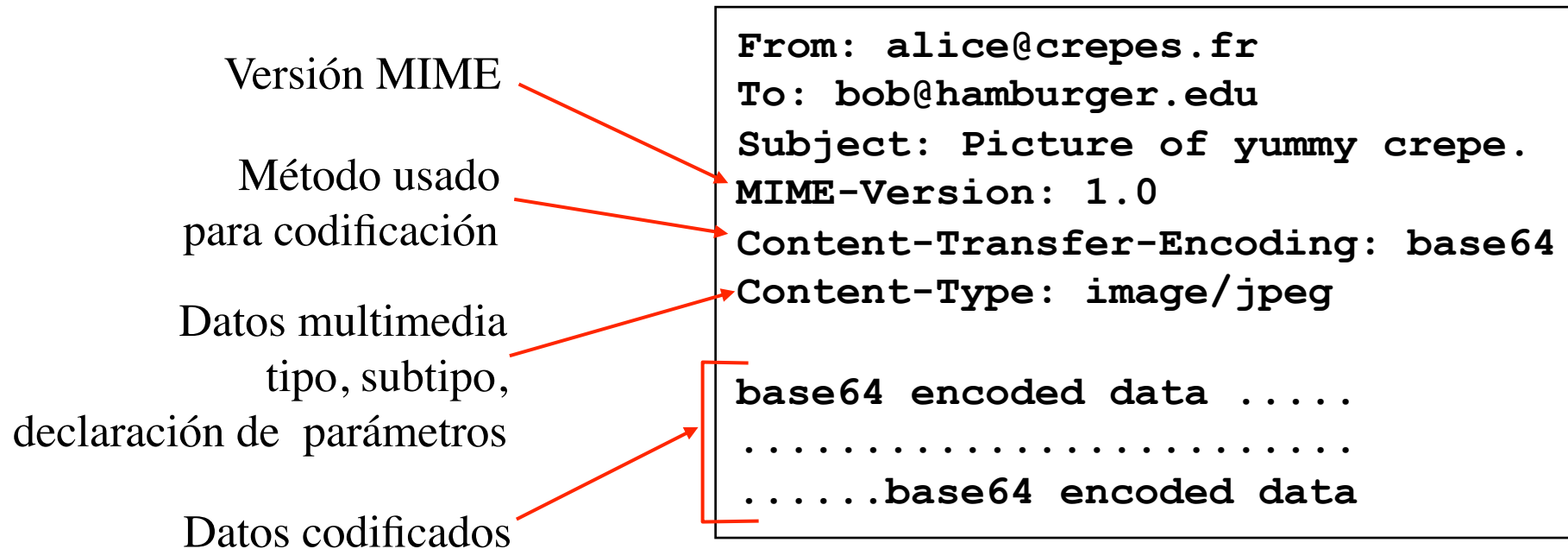
- Líneas de cabecera, p.e.,
 - ☐ Para:
 - ☐ Desde:
 - ☐ Objeto:

Difiere de los comandos SMTP!
- Cuerpo
 - ☐ El mensaje
 - ☐ Caracteres ASCII sólo



6.2 Formato del mensaje: extensiones multimedia

- MIME: multimedia mail extension, RFC 2045, 2056
- Líneas adicionales en la cabecera del mensaje declara un tipo de contenido MIME



6.2 Tipos MIME

Content-Type: `type/subtype; parameters`

Text

- Ejemplos de subtipos:
`plain, html`

Image

- Ejemplos de subtipos:
`jpeg, gif`

Audio

- Ejemplos de subtipos:
`basic` (8-bit mu-law encoded), `32kadpcm` (32 kbps coding)

Video

- Ejemplos de subtipos:
`mpeg, quicktime`

Application

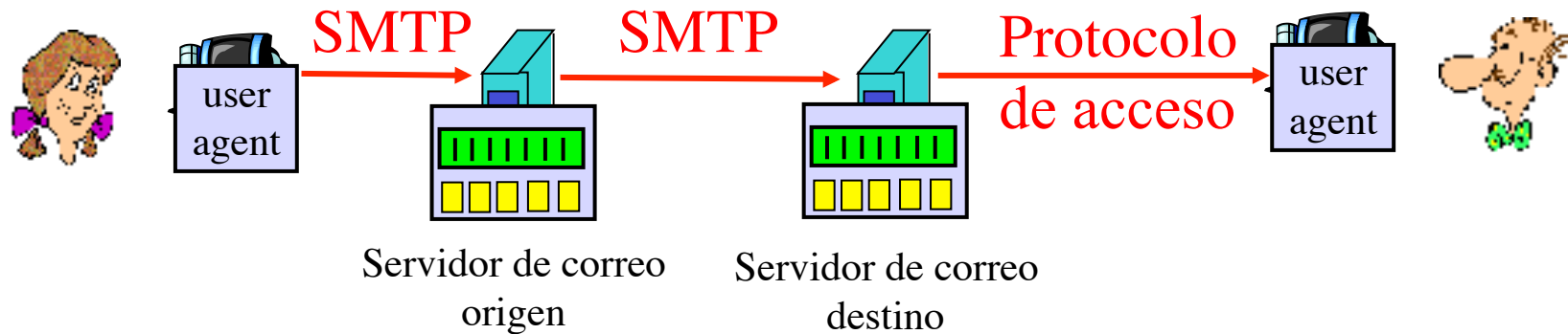
- Otros datos que deben ser procesados por el lector antes de que sean visibles
- Ejemplos de subtipos:
`msword, octet-stream`

6.2 Multipart Type

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

```
--StartOfNextPart
Dear Bob, Please find a picture of a crepe.
--StartOfNextPart
Content-Transfer-Encoding: base64
Content-Type: image/jpeg
base64 encoded data .....
.....base64 encoded data
--StartOfNextPart
Do you want the recipe?
```

6.2 Mail access protocols



- SMTP: entrega/almacenaje en el servidor receptor
- Protocolo de acceso de correo: recogida del servidor
 - ❑ POP: *Post Office Protocol* [RFC 1939]
 - Autorización (agente <-->servidor) y descarga
 - ❑ IMAP: *Internet Mail Access Protocol* [RFC 1730]
 - Más características (más complejo)
 - Manipulación de mensajes almacenados en el servidor
 - ❑ HTTP: Google, Hotmail , Yahoo! Mail, etc.

6.2 Protocolo POP3

Fase de autorización

- Comandos de cliente:
 - ☐ `user`: declare username
 - ☐ `pass`: password
- Respuestas del servidor:
 - ☐ `+OK`
 - ☐ `-ERR`

Fase de transacción,

- Comandos de cliente:
 - ☐ `list`: lista número de mensajes
 - ☐ `retr`: recupera por número de mensaje
 - ☐ `dele`: borrar
 - ☐ `quit`: salir

```
S: +OK POP3 server ready
C: user bob
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
```

6.2 POP3 e IMAP

POP3

- El ejemplo precedente usa el modo “descarga y borra”.
- Bob no podrá releer el correo si cambia de cliente.
- “Descarga y mantiene”: copia de los mensajes en diferentes clientes.
- POP3 no mantiene el estado entre sesiones.

IMAP

- Mantiene todos los mensajes en un lugar: el servidor.
- Permite la organización de mensajes en el carpetas.
- IMAP mantiene el estado del usuario entre sesiones:
 - ❑ Los nombres de las carpetas y relación entre las carpetas y los mensajes.

6.2 Bibliografía

- [1] Tanenbaum, A. S., Computer Networks, 4ª Ed Pearson 2003, apartado 7.2.
- [2] J.F Kurose y K.W. Ross, Redes de Computadores: un enfoque descendente basado en Internet, 2002.