

Clase 24

# E-Mail

Tema 6.- Nivel de aplicación en Internet

*Dr. Daniel Morató  
Redes de Computadores  
Ingeniero Técnico de Telecomunicación  
Especialidad en Sonido e Imagen  
3º curso*

# Temario

- 1.- Introducción
- 2.- Nivel de enlace en LANs
- 3.- Interconexión de redes IP
- 4.- Enrutamiento con IP
- 5.- Nivel de transporte en Internet
- 6.- Nivel de aplicación en Internet
- 7.- Ampliación de temas

# Temario

1.- Introducción

2.- Nivel de enlace en LANs

3.- Interconexión de redes IP

4.- Enrutamiento con IP

5.- Nivel de transporte en Internet

**6.- Nivel de aplicación en Internet**

» Nivel de aplicación: WWW

» FTP. Telnet. Ejemplos

» E-mail

» DNS. P2P. Mensajería...

7.- Ampliación de temas

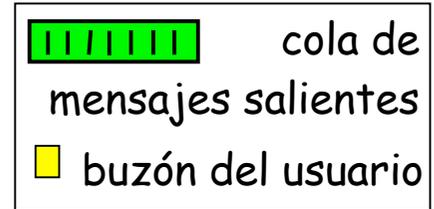
# Contenido

- » Arquitectura
- » SMTP
- » Formato de los mensajes
- » Protocolos de acceso al mail

# Contenido

- » **Arquitectura**
- » SMTP
- » Formato de los mensajes
- » Protocolos de acceso al mail

# Electronic Mail

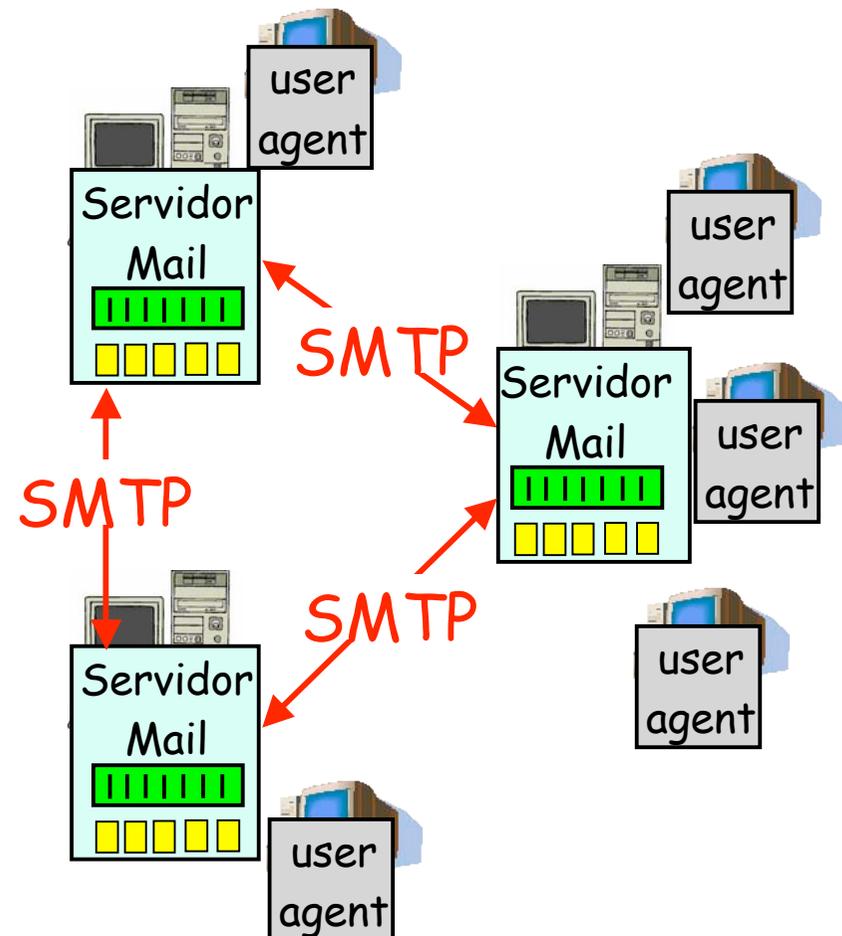


## Tres elementos principales:

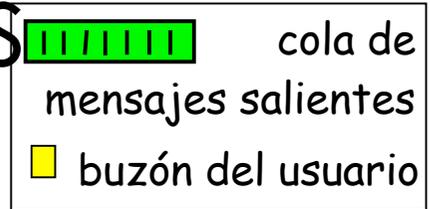
- » Agentes de usuario (*user agents*)
- » *Mail servers*
- » Simple Mail Transfer Protocol: **SMTP**

## User Agent

- » alias “programa de correo”
- » Componer, editar, leer mensajes de correo
- » ej., Eudora, Outlook, elm, Netscape Messenger
- » Mensajes salientes y entrantes en el servidor

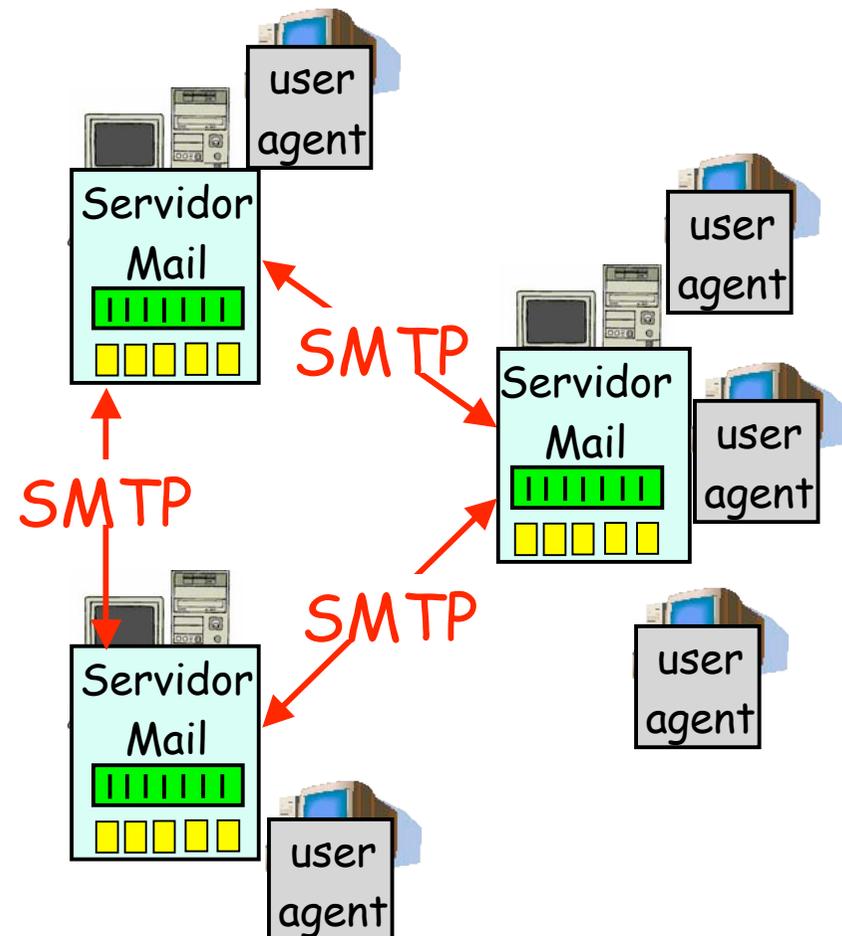


# E-Mail: Servidores



## Servidores de Mail:

- » **Mailbox** contiene los mensajes entrantes para el usuario
- » **Cola de mensajes** salientes (a enviar)
- » **Protocolo SMTP** entre servidores de correo para enviar mensajes
  - cliente: el servidor de correo que envía
  - "servidor": el servidor de correo que recibe



# Contenido

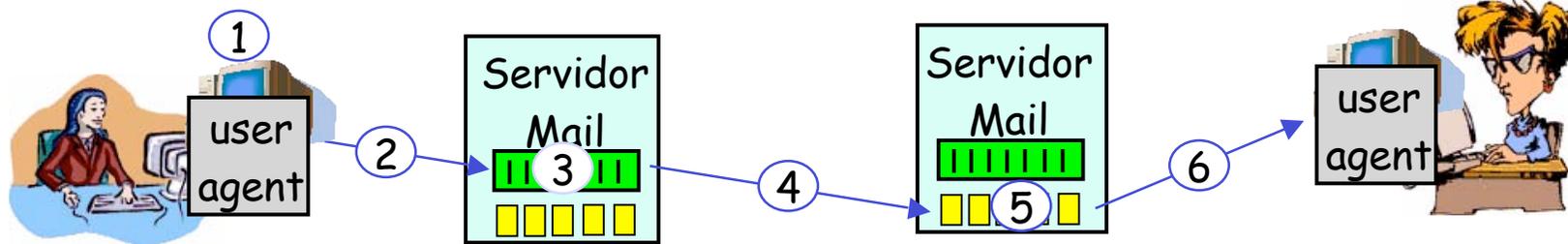
- » Arquitectura
- » **SMTP**
- » Formato de los mensajes
- » Protocolos de acceso al mail

# E-Mail: SMTP [RFC 2821]

- » Emplea **TCP** para entregar de forma fiable los mensajes entre el cliente y el servidor
- » Puerto **25**
- » Transferencia directa: del servidor del emisor al servidor del receptor
- » **Tres fases** en la transferencia
  - handshaking (el saludo)
  - transferencia de mensajes
  - cierre
- » Interacción mediante comandos y respuestas
  - comandos: texto ASCII
  - respuestas: código de estado y frase de estado
- » Los mensajes deben estar en **ASCII** de 7 bits

# Ejemplo: Usuario 1 envía mensaje a Usuario 2

- 1) Usuario 1 emplea un UA para crear el mensaje para usuario2@micasa.com
- 2) El programa envía el mensaje a su servidor de correo y lo coloca en una cola de mensajes
- 3) El Servidor de Mail, como cliente, abre una conexión TCP con el Servidor de Usuario 2
- 4) Envía el mensaje de Usuario 1 empleando SMTP sobre esa conexión TCP
- 5) El servidor de mail de Usuario 2 coloca el mensaje en su buzón
- 6) Usuario 2 lanza su UA para leer el mensaje (volveremos a esta parte)



# Ejemplo de SMTP

```
» 220 unavarra.es ESMTP Sendmail 8.9.3/8.9.1 (IRIS 3.0); Fri, 29 Apr 2005 14:00:19 +0200
(MET DST)
» HELO daniel.flm.unavarra.es
» 250 unavarra.es Hello s169m159.unavarra.es [130.206.169.159], pleased to meet you
» MAIL FROM: <daniel.moraro@unavarra.es>
» 250 <daniel.moraro@unavarra.es>... Sender ok
» RCPT TO: danielmorato@yahoo.com
» 250 danielmorato@yahoo.com... Recipient ok
» DATA
» 354 Enter mail, end with "." on a line by itself
» Hola
» Aqui, saludandome a mi mismo
» .
» 250 OAA24057 Message accepted for delivery
» QUIT
» 221 unavarra.es closing connection
```

# Probando SMTP

- » **telnet servername 25**
- » Pruebe los comandos HELO, MAIL FROM, RCPT TO, DATA, QUIT
- » Con esos comandos puede enviar un email sin emplear un programa de email

# Algo más sobre SMTP

- » Conexiones persistentes
- » Requiere que el mensaje (cabecera y contenido) esté en ASCII de 7 bits
- » El servidor de SMTP emplea **CRLF.CRLF** para reconocer el final del mensaje

## **Comparación con HTTP:**

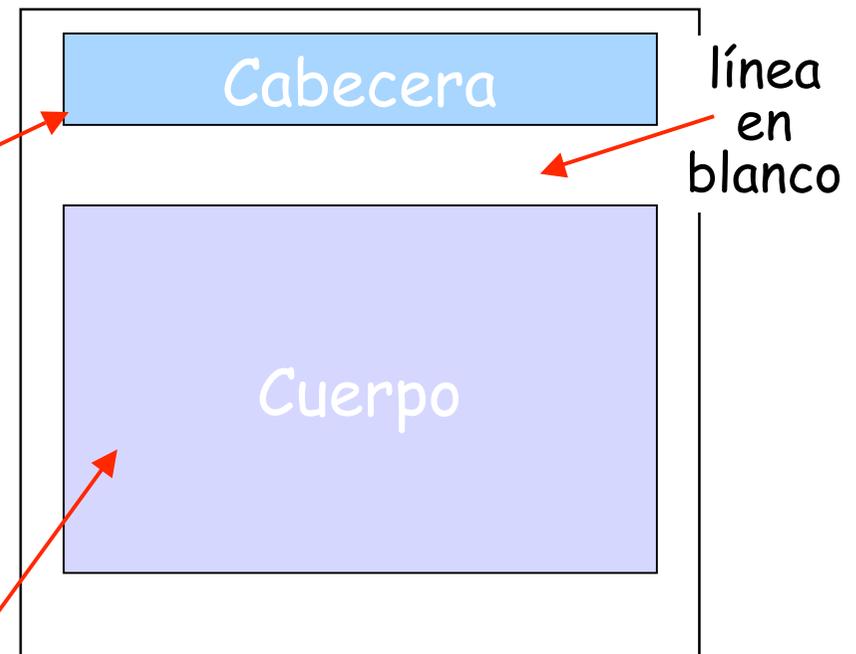
- » HTTP: pull
- » SMTP: push
- » Ambos emplean comandos y respuestas en ASCII

# Contenido

- » Arquitectura
- » SMTP
- » **Formato de los mensajes**
- » Protocolos de acceso al mail

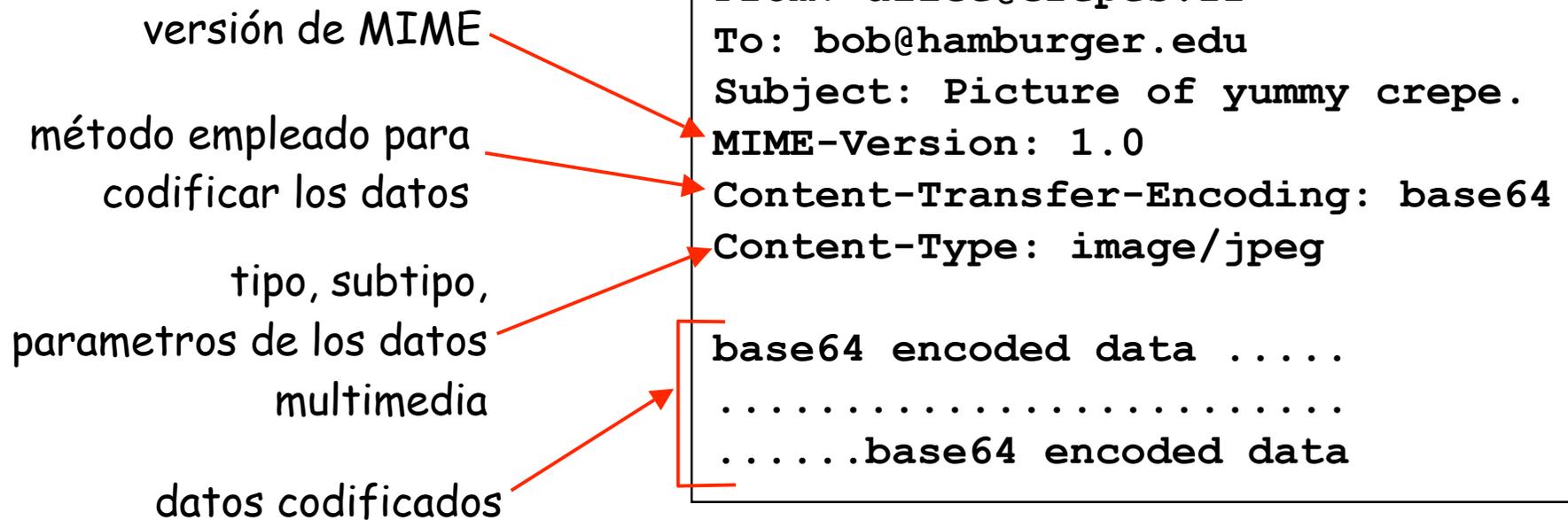
# Formato del mensaje de email

- » SMTP: protocolo para intercambiar mensajes de email (RFC 2821)
- » RFC 822: estándar para el formato del mensaje:
- » Líneas de cabecera, ej.
  - **To:**
  - **From:**
  - **Subject:**Diferentes de los comandos de SMTP
- » Cuerpo
  - el "mensaje", solo caracteres ASCII



# Formato del mensaje: Multimedia Extensions

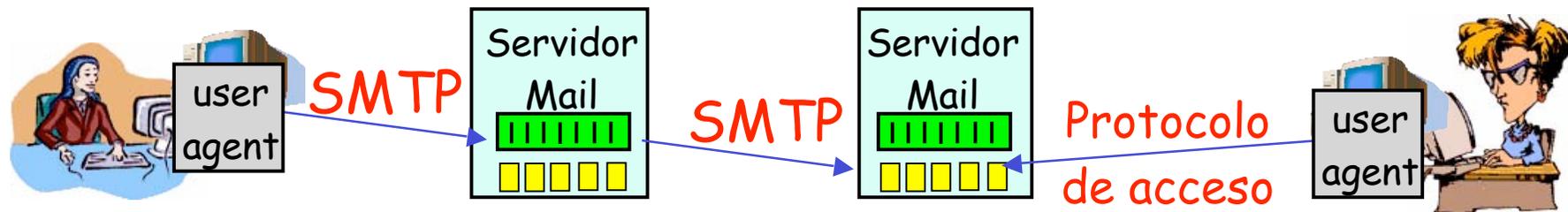
- » MIME: Multimedia Mail Extension, RFC 2045, 2056
- » Permite mandar contenido que no sea texto ASCII
- » Líneas adicionales en la cabecera del mensaje para declarar el tipo del contenido



# Contenido

- » Arquitectura
- » SMTP
- » Formato de los mensajes
- » **Protocolos de acceso al mail**

# Protocolos de acceso al Mail



- » SMTP: entrega/almacena en el servidor del receptor
- » Protocolo de acceso al Mail: obtención de mensajes del servidor
  - POP: Post Office Protocol [RFC 2821]
    - » Autorización (agente ↔ servidor) y descarga
  - IMAP: Internet Message Access Protocol [RFC 3501]
    - » Más funcionalidades (más complejo)
    - » Manipulación de mensajes almacenados en el servidor
  - HTTP: Hotmail , Yahoo! Mail, etc.

# Protocolo POP3

## Autorización

- » Comandos del cliente:
  - **user** : declara el nombre de usuario
  - **pass** : clave
- » Respuestas del servidor:
  - +OK
  - -ERR

## Fase de transacción, cliente:

- » **list** : lista números de mensajes
- » **retr** : descarga mensaje por número
- » **dele** : borrar
- » **quit**

```
+OK Qpopper (version 4.0.5) at si starting.  
user daniel.morato  
+OK  
pass hungry  
+OK daniel.morato has 412 visible messages (0  
hidden) in 35020509 octets.  
list  
1 498  
2 912  
.  
retr 1  
<contenido mensaje 1>  
.  
dele 1  
retr 2  
<contenido mensaje 2>  
.  
dele 2  
quit  
+OK POP3 at si signing off
```

# Más sobre POP3 e IMAP

## Más sobre POP3

- » El ejemplo anterior era “descargar y borrar”
- » Bob no puede volver a leer los mensajes si cambia de cliente
- » “Descargar y mantener”: copia el mensaje pero no lo borra. Permite descargarlos en otro cliente
- » POP3 es sin estado entre sesiones

## IMAP

- » Mantiene todos los mensajes en un lugar: el servidor
- » Permite al usuario organizar los mensajes en carpetas
- » IMAP mantiene el estado entre sesiones:
  - Nombres de carpetas y relación entre ID de mensaje y carpeta en la que está

# Temario

1.- Introducción

2.- Nivel de enlace en LANs

3.- Interconexión de redes IP

4.- Enrutamiento con IP

5.- Nivel de transporte en Internet

**6.- Nivel de aplicación en Internet**

» Nivel de aplicación: WWW

» FTP. Ejemplos

» E-mail

» DNS. P2P. Mensajería...

7.- Ampliación de temas

# Próxima clase

*DNS, P2P, mensajería...*