Redes de Computadores

Tema 7 – Capas superiores

Juan Segarra, Natalia Ayuso y Jesús Alastruey

Departamento de
Informática e Ingeniería
de Sistemas

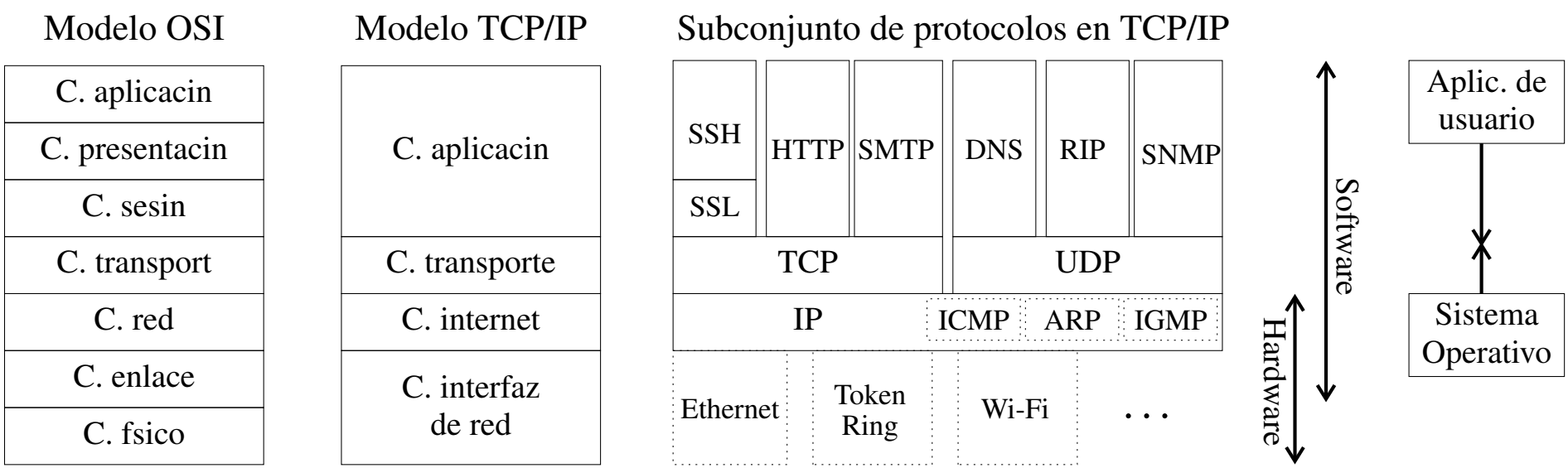
Universidad Zaragoza

Índice

1 Índice

1 Introducción: capas superiores

- Sesión: control/coordinación de comunicaciones
- Presentación: formato de los datos intercambiados
- Aplicación: protocolos específicos para aplicaciones concretas



2 Capa de sesión OSI

- Se encarga del establecimiento, gestión y terminación de comunicaciones entre aplicaciones
- Sesión: establecimiento comunicaciones + secuencia de transacciones + terminación comunicaciones
- Servicios: autentificación, permisos, restablecimiento de sesión, etc.
- En el modelo TCP/IP la gestión de la sesión se realiza mediante mecanismos de la aplicación
  - HTTP cookies permiten suspender una sesión y reanudarla después
  - X Window System coordina flujos de pantalla, teclado, ratón...
- Especialmente importante en voz sobre IP (VoIP)
  - Session Description Protocol (SDP), Session Initiation Protocol (SIP), Control de llamada (H.323), etc.

3 Índice

3 Capa de presentación OSI

- Gestiona la representación, compresión y cifrado de datos
- Existen distintos formatos, que dependen de:

  Hardware: *big/little-endian*, tamaño de registros (tamaño de enteros por defecto)...

  Sistema operativo: repertorio y codificación de caracteres (ASCII 7/8 bits, ISO, Unicode+UTF), salto de línea...

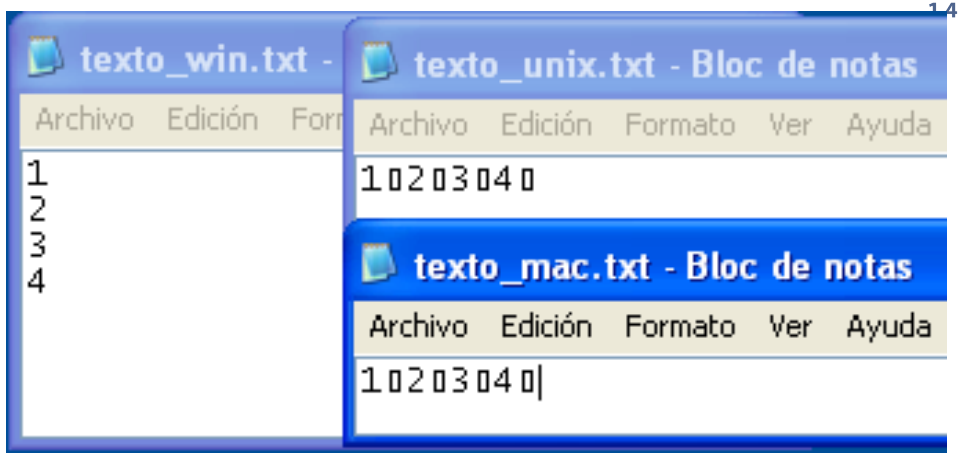
  Compilador: disposición de *structs* en memoria (distintos *paddings* para alinear campos)

  Lenguaje de programación: tipos de datos (e.g. booleano) y funciones predefinidas (e.g. serialización)

  Aplicación: datos a enviar
- Para intercambiar datos se necesita un formato común
  - Ejemplos: XDR, ASN.1, MIME, HTML, XML

3.1 Ejemplo: salto de línea

- Fichero de texto: el salto de línea tiene distintas representaciones en sistemas Unix (LF, 0x0a), Mac (CR, 0x0d) y Windows (CRLF, 0x0d0a)



SO Windows: visualización en bloc de notas de ficheros de texto con saltos de línea Windows, Unix y Mac.

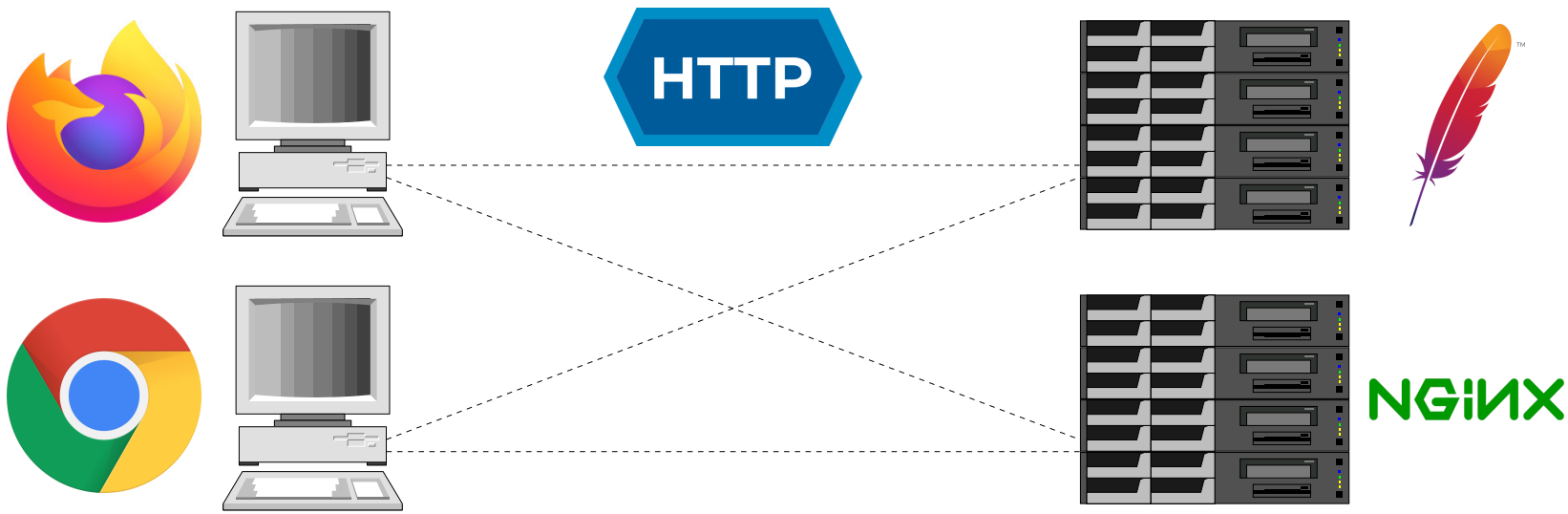
```
$ hexdump -C texto_unix.txt
31 0a 32 0a 33 0a 34 0a                        |1.2.3.4.|
$ hexdump -C texto_mac.txt
31 0d 32 0d 33 0d 34 0d                        |1.2.3.4.|
$ hexdump -C texto_win.txt
31 0d 0a 32 0d 0a 33 0d 0a 34 0d 0a   |1..2..3..4..|
```

- Las aplicaciones de transferencia de ficheros de texto convierten el formato de los saltos de línea

4 Índice

4 Capa de aplicación OSI

- Especifica los protocolos de transmisión de datos específicos de cada servicio
- Única capa que interacciona directamente con el usuario
- Abstrae a las aplicaciones de sus comunicaciones
  - E.g. distintos navegadores usan el mismo protocolo de comunicación con los servidores web



4 Capa de aplicación OSI (II)

- No confundir la aplicación completa con la capa de aplicación de la arquitectura de red
  - Aplicación: firefox, chrome, apache, etc.
  - Protocolo en capa aplicación: HTTP
- Modelo TCP/IP: nivel de aplicación puede incluir nivel de presentación, es decir, especificación del formato de los datos que intercambian
  - E.g. correo electrónico (SMTP + MIME), navegación web (HTTP + HTML)

4.1 Protocolo HTTP

- HyperText Transport Protocol
- Dos tipos de mensaje: petición, respuesta
- Objetos web (páginas, imágenes, etc.) referenciables mediante URL (*Uniform Resource Locator*):
  - esquema://dirección[:puerto]/camino/objeto
  - http://www.unizar.es/academico/unizar.html
- Versiones
  - HTTP/1.0 (RFC 1945): por defecto un objeto por conexión
  - HTTP/1.1 (RFC 7230): por defecto conexión persistente
  - HTTP/2 (RFC 7540): reduce la latencia de comunicación mediante compresión de cabeceras y priorización de peticiones (pueden responderse en orden distinto al que fueron cursadas). Codificación binaria.
  - HTTP/3 (RFC 9114): HTTP sobre QUIC (*Quick UDP Internet Connections*).

4.1.1 Petición HTTP

1. Línea de petición (negrita) + salto de línea
2. Cabeceras de petición + salto de línea
3. Salto de línea
4. Cuerpo, vacío si no se transmiten datos

Nota: salto de línea = <CR><LF> (0x0d 0x0a)

GET /index.html HTTP/1.1
Host: www.unizar.es
User-Agent: Mozilla/5.0
Accept: text/html,application/xhtml+xml,application/xml
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive

4.1.1 Petición HTTP (II)

- Línea de petición: tipo objeto versión
  - E.g. `GET /index.html HTTP/1.1`
- Tipos de petición (*method*):

GET	Solicita un elemento (HTTP/1.0)
POST	Envía datos a un *script* (HTTP/1.0)
HEAD	Solicita sólo la cabecera (HTTP/1.0)
PUT	Reemplaza el elemento (HTTP/1.1)
DELETE	Borra el elemento indicado (HTTP/1.1)
otros:	TRACE, OPTIONS, CONNECT, PATCH

- GET y HEAD nunca modifican contenido
- Se pueden enviar datos en el URL (GET, `miurl?var=25&var2=30`) o en el cuerpo de la petición (POST)

4.1.1 Petición HTTP (III)

Cabeceras más importantes para la petición:

Host	Nombre del equipo al que se solicitan datos
User-Agent	Identificador del software cliente
Accept	Tipos de datos aceptados
Accept-Language	Idiomas aceptados
Accept-Encoding	Métodos de compresión aceptados
Accept-Charset	Codificaciones de texto aceptadas
Connection	Comportamiento de conexión deseado
Keep-Alive	Tiempo a mantener la conexión abierta
Referrer	URL que nos ha llevado a esta petición
If-Modified-Since	Petición condicionada
Cookie	Identificador proporcionado por el servidor

4.1.2 Respuesta HTTP

1. Línea de respuesta (negrita) + salto de línea
2. Cabeceras de respuesta + salto de línea
3. Salto de línea
4. Cuerpo (texto normal)

Nota: salto de línea = <CR><LF> (0x0d 0x0a)

```
HTTP/1.1 200 OK
Date: Tue, 04 Dec 2012 09:48:53 GMT
Server: Apache
Set-Cookie: JSESSIONID=45C0E390670416F258729C7464522923; Path=/
ETag: W/"13322-1354614320000"
Last-Modified: Tue, 04 Dec 2012 09:45:20 GMT
Transfer-Encoding: chunked
Content-Type: text/html; charset=ISO-8859-1
Content-Language: es

1ff8

<!DOCTYPE HTML PUBLIC //W3C//DTD HTML 4.01 Transitional//EN"…
```

4.1.2 Respuesta HTTP (II)

- Línea de respuesta comprensible por máquinas y humanos: versión código texto (`HTTP/1.1 200 OK`)
  - 2xx Success
    - 200 OK
  - 3xx Redirection
    - 301 Moved permamently
    - 304 Not modified (ante petición condicional)
  - 4xx Client Errors
    - 400 Bad request
    - 401 Authorization required
    - 404 Not found
  - 5xx Server errors
    - 500 Internal error
    - 502 Service overloaded
    - 505 Version not supported

4.1.2 Respuesta HTTP (III)

Cabeceras más importantes para la respuesta:

Server	Software del servidor
Date	Fecha actual
Last-Modified	Fecha de modificación
Expires	Fecha de expiración
Transfer-Encoding	Codificación de la transferencia
Content-Type	Tipo de datos enviados
Content-Length	Longitud de datos enviados
Content-Encoding	Método de compresión usado
ETag	Identificador de recurso
Set-cookie	Envía un identificador al usuario
Cache-Control	Especificaciones de cacheo de contenido

4.2 Protocolo SMTP

- Simple Mail Transfer Protocol, RFC 5321
- Servicio de *envío* de correo electrónico
- Mensajes de texto de petición/respuesta
  - En principio sólo admite texto ASCII (7 bits)
  - Pueden usarse otras codificaciones para enviar texto no ASCII o contenido binario
- Formato del mensaje separado del protocolo
  - RFC822, MIME

☞ Experimentar con SMTP:
`lab000$ nc localhost 25`

☞ Experimentar con POP3 (*Post Office Protocol*) para descargar correo: `nc posta.unizar.es 110`

4.2 Protocolo SMTP (II)

```
S: 220 www.example.com ESMTP Postfix
C: HELO mydomain.com
S: 250 Hello mydomain.com
C: MAIL FROM:<sender@mydomain.com>
S: 250 Ok
C: RCPT TO:<friend@example.com>
S: 250 Ok
C: DATA
S: 354 End data with <CR><LF>.<CR><LF>
C: Subject: test message
C: From: sender@mydomain.com
C: To: friend@example.com
C:
C: Hello,
C: This is a test.
C: Goodbye.
C: .
S: 250 Ok: queued as 12345
C: QUIT
S: 221 Bye
```

4.3 Protocolo DNS

- Domain Name System / Sistema de Nombres de Dominios
- Un dominio de Internet es un nombre jerárquico asociado a un grupo de dispositivos o equipos conectados a Internet
- DNS proporciona servicios de nombres de dominio
  - Traducción entre nombre y dirección IP:

```
$ host hendrix01.cps.unizar.es
hendrix01.cps.unizar.es has address 155.210.152.183
$ dig -x 155.210.152.183 +short
hendrix01.cps.unizar.es.
```

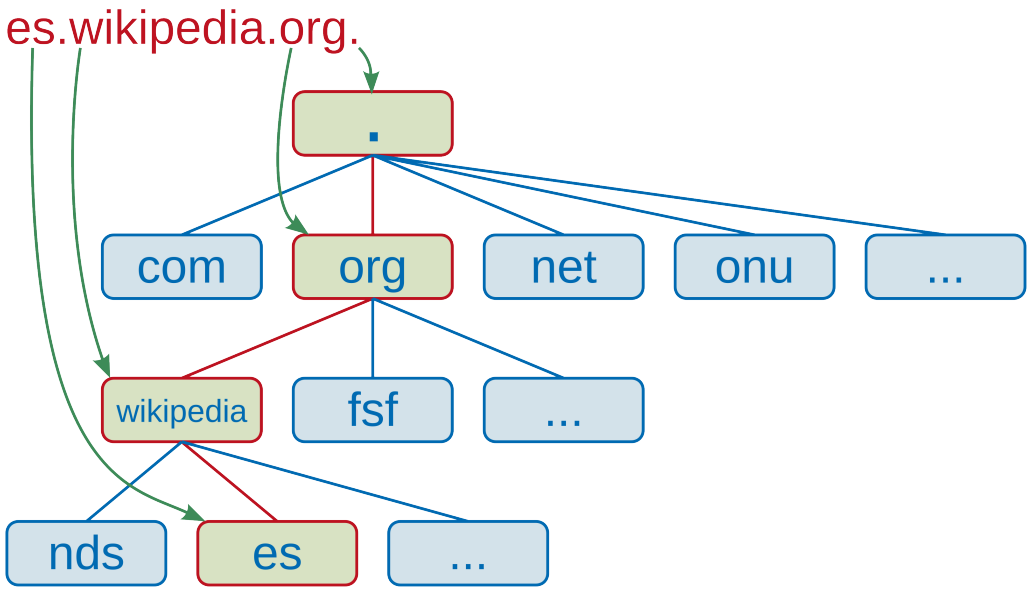
  - Servidor de correo asociado al dominio:

```
$ host -t mx unizar.es
unizar.es mail is handled by 10 mx01.puc.rediris.es.
unizar.es mail is handled by 10 mx02.puc.rediris.es.
```

- Servicio importante: mínimo 2 servidores por dominio

4.3 Protocolo DNS (II)

- Organización jerárquica en dominios
- Nivel superior (*top level domain*, TLD) gestionado por NIC
- *Full Qualified Domain Name* (FQDN) o nombre absoluto: toda la jerarquía separada por puntos y acabada en punto



4.3 Protocolo DNS (III)

- En general, estrategia de resolución iterativa por parte del DNS local, por ejemplo: www.wikipedia.org