

Capa Aplicación: File Transfer Protocol

ELO322: Redes de Computadores Agustín J. González

Este material está basado en:

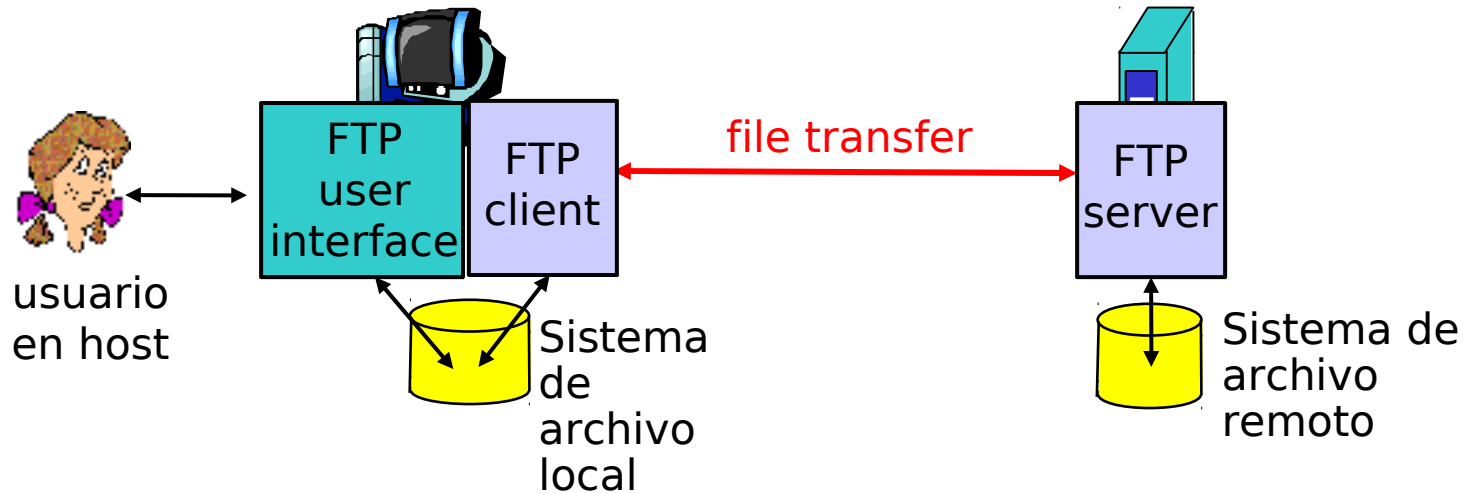
- Material de apoyo al texto *Computer Networking: A Top Down Approach Featuring the Internet 3rd* edition. Jim Kurose, Keith Ross Addison-Wesley, 2004.

Capítulo 2: Capa Aplicación

- 2.1 Principios de la aplicaciones de red
- 2.2 Web y HTTP
- **2.3 FTP**
- 2.4 Correo Electrónico
 - SMTP, POP3, IMAP
- 2.5 DNS
- 2.6 P2P
Compartición de archivos
- 2.7 Programación de Socket con TCP
- 2.8 Programación de socket con UDP
- 2.9 Construcción de un servidor WEB

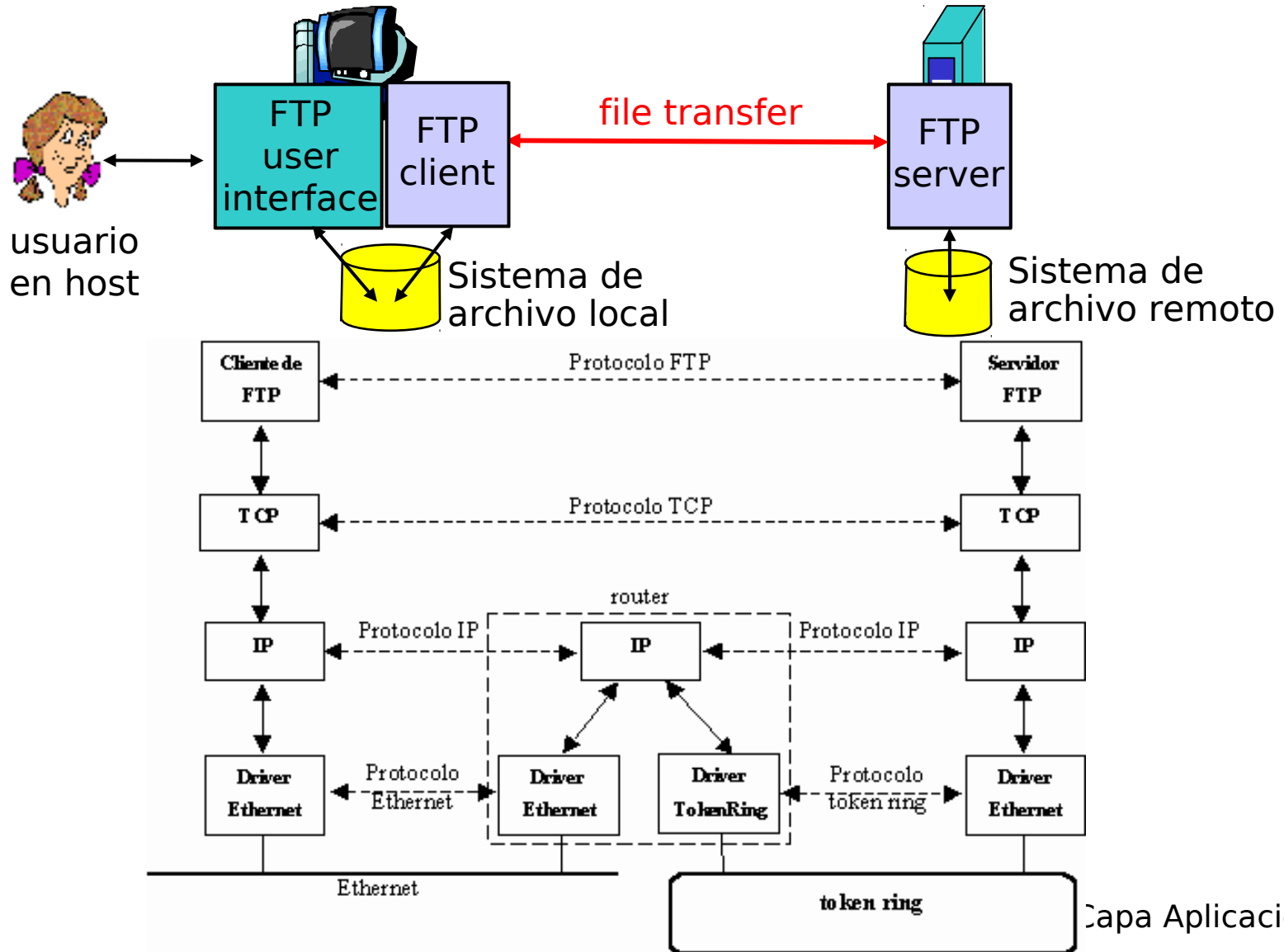
Obs: Veremos FTP como ejemplo de protocolo aún cuando hoy ya no lo usamos

FTP: El protocolo de transferencia de archivos (File Transfer Protocol)



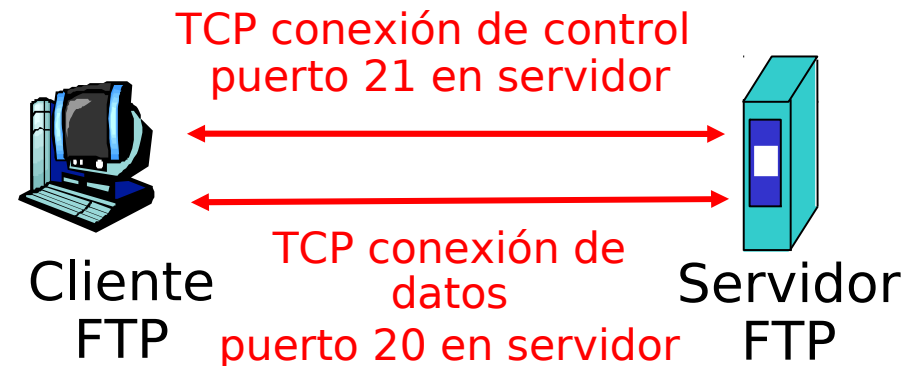
- Transferencia de archivos a/desde el host remoto
- Sigue modelo cliente/servidor
 - *cliente*: sitio que inicia la transferencia (ya sea a/desde sitio remoto)
 - *servidor*: host remoto
- ftp: RFC 959
- Servidor ftp: puerto 21, cliente en cualquier puerto.
Ver netstat -p tcp

FTP: La capa aplicación se apoya en la inferiores. Hacemos abstracción



FTP: Conexiones separadas de control y datos

- ❑ Cliente FTP contacta servidor FTP en puerto 21, especificando TCP como protocolo de transporte
- ❑ El cliente obtiene autorización sobre el control de la conexión
- ❑ El cliente navega el directorio remoto enviando comandos sobre la conexión de control.
- ❑ Cuando el servidor recibe una petición de transferencia de archivo, el servidor abre una conexión de datos hacia el cliente. Éste es Modo Activo.
- ❑ Después de la transferencia de un archivo, el servidor cierra la conexión de datos.



- ❑ El servidor abre una segunda conexión TCP de datos para transferir otro archivo.
- ❑ Conexión de control: **“out of band” (fuera de banda)**
- ❑ Servidor FTP mantiene “estado”: directorio actual, cuenta de usuario conectado.
- ❑ Existe modo activo y pasivo

FTP comandos, respuestas

Muestra de comandos:

- ▣ Son enviados como texto ASCII vía el canal de control
- ▣ **USER *username***
- ▣ **PASS *password***
- ▣ **LIST** retorna la lista de archivos del directorio actual
- ▣ **RETR *filename*** baja un archivo (gets)
- ▣ **STOR *filename*** almacena (puts) archivo en host remoto

Algunos códigos retornados

- ▣ Código estatus y frases (como en HTTP)
- ▣ **331 Username OK, password required**
- ▣ **125 data connection already open; transfer starting**
- ▣ **425 Can't open data connection**
- ▣ **452 Error writing file**

Capítulo 2: Capa Aplicación

- 2.1 Principios de la aplicaciones de red
- 2.2 Web y HTTP
- 2.3 FTP
- 2.4 Correo Electrónico
 - SMTP, POP3, IMAP
- 2.5 DNS
- 2.6 P2P
Compartición de archivos
- 2.7 Programación de Socket con TCP
- 2.8 Programación de socket con UDP
- 2.9 Construcción de un servidor WEB