

Principios de los protocolos de la capa de aplicación.

Aplicaciones de red: jerga

Proceso: programa que se ejecuta en un host.

- ❑ En el mismo host, dos procesos se comunican utilizando **comunicación interproceso** (definidos por un sistema operativo).
- ❑ Los procesos que se ejecutan en diferentes hosts se comunican con un **protocolo de capa de aplicación**.

Agentes de usuario: interfaces con un usuario "por encima" y una red "por debajo".

- ❑ Implementa interfaces de usuario y protocolos a nivel de aplicación.
 - Web: navegador.
 - Correo electrónico: lector de correo.
 - Transmisión de audio/vídeo: reproductor multimedia.

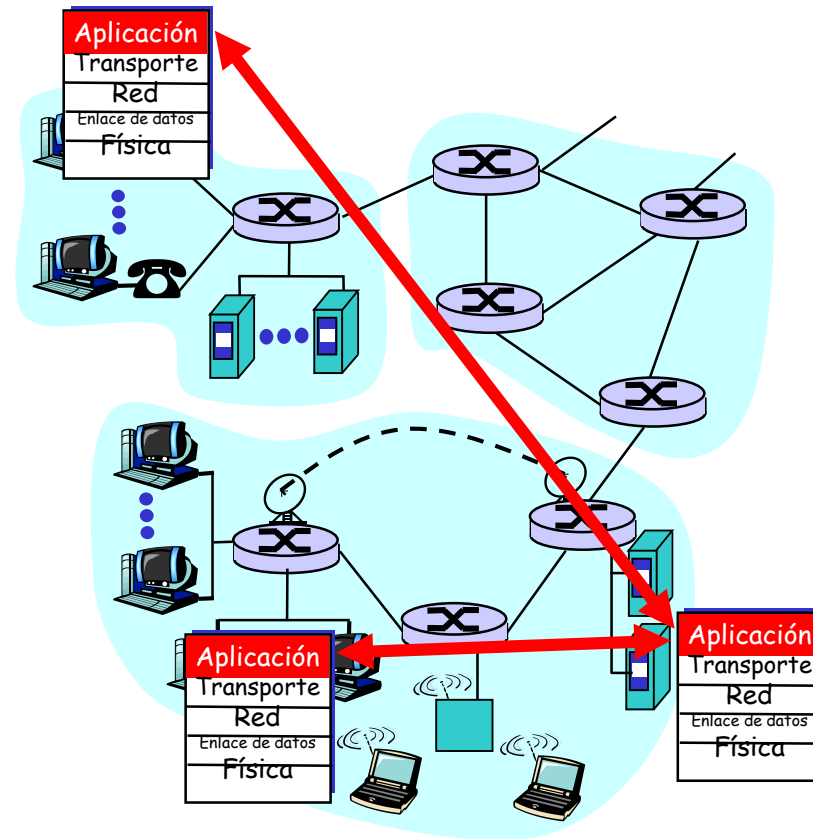
Aplicaciones y protocolos de la capa de aplicación

Aplicación: comunicación, procesos distributivos.

- Por ejemplo: correo electrónico, web, compartición de archivos entre iguales, mensajería instantánea.
- Funcionamiento en sistemas finales (hosts).
- Intercambio de mensajes para la implementación de aplicaciones.

Protocolos de capa de aplicación

- Una "parte" de la aplicación.
- Definición de mensajes intercambiados entre las aplicaciones y las acciones tomadas.
- Uso de servicios de comunicación proporcionados por los protocolos de capas inferiores (TCP, UDP).



Características de los protocolos de capa de aplicación

- ❑ Los tipos de mensajes intercambiados, por ejemplo, mensajes de petición y de respuesta.
- ❑ Sintaxis de los tipos de mensajes: qué campos hay en los mensajes y cómo están delineados estos campos.
- ❑ Semántica de los campos, por ejemplo, significado de la información en los campos.
- ❑ Reglas que determinan cuándo y cómo los procesos envían y responden a los mensajes.

Protocolos de dominio público:

- ❑ Definidos en RFC.
- ❑ Permiten interoperabilidad.
- ❑ Por ejemplo: HTTP, SMTP.

Protocolos de propietarios:

- ❑ Por ejemplo: KaZaA, Skype.

Paradigma del cliente-servidor

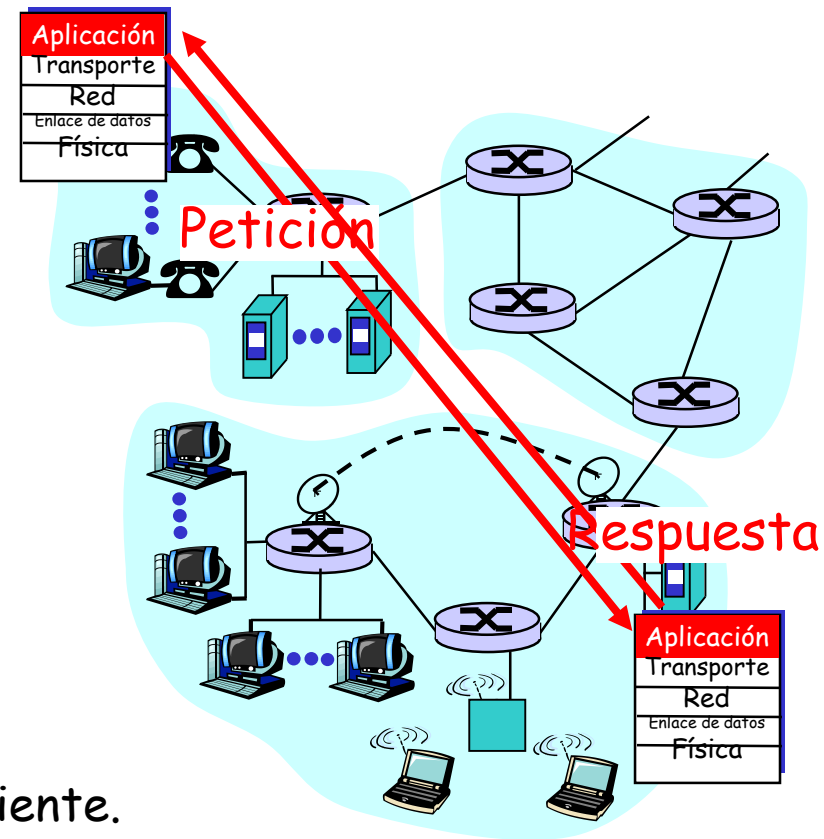
La aplicación de red típicamente tiene dos partes: *el cliente* y *el servidor*

Cliente:

- ❑ Inicia el contacto con el servidor ("habla primero").
- ❑ Normalmente solicita un servicio del servidor.
- ❑ Web: cliente implementado en el navegador; correo electrónico: en el lector de correo.

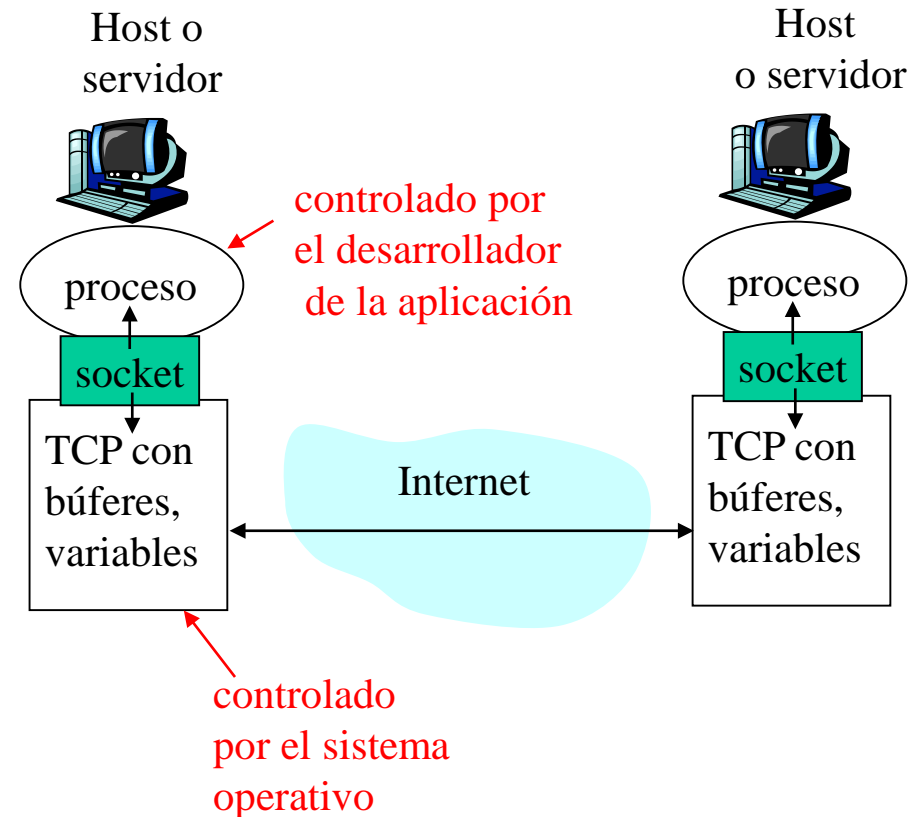
Servidor:

- ❑ Proporciona el servicio solicitado al cliente.
- ❑ Por ejemplo, el servidor de Web envía la página Web solicitada, el servidor de correo entrega el correo electrónico.



Procesos que se comunican a través de la red

- ❑ El proceso envía/recibe mensajes a/de su socket.
- ❑ El socket es análogo a una puerta:
 - El proceso emisor empuja el mensaje por su puerta.
 - Este proceso asume la existencia de una infraestructura de transporte al otro lado de la puerta que transportará el mensaje al socket en el proceso receptor.



- ❑ API: (1) elección del protocolo de transporte; (2) posibilidad de fijar algunos parámetros (más adelante se tratará este tema con mucha más profundidad).

Direccionamiento de procesos:

- ❑ Para que un proceso reciba mensajes debe tener un identificador.
- ❑ Cada host tiene una única dirección IP de 32 bits.
- ❑ **Pregunta:** ¿Basta con la dirección IP del host, en el que el proceso se ejecuta, para identificar el proceso?
- ❑ **Respuesta:** No, ya que muchos procesos diferentes pueden estar ejecutándose en el mismo host.
- ❑ El identificador incluye tanto la dirección IP como los **números de puerto** asociados con el proceso del host.
- ❑ Ejemplos de números de puerto:
 - Servidor HTTP: 80.
 - Servidor de correo: 25.
- ❑ **Más adelante se tratará este tema con más profundidad.**

¿Qué servicio de transporte necesita una aplicación?

Pérdida de datos

- ❑ Ciertas aplicaciones (por ejemplo, audio) pueden tolerar algunas pérdidas.
- ❑ Otras aplicaciones (por ejemplo, transferencia de archivos, Telnet) requieren el 100 por ciento de transferencia fiable de datos.

Temporización

- ❑ Algunas aplicaciones (por ejemplo, telefonía de Internet, juegos interactivos) requieren un retardo lento para ser "efectivas".

Ancho de banda

- ❑ Algunas aplicaciones (por ejemplo, multimedia) requieren un mínimo de ancho de banda para ser "efectivas".
- ❑ Otras aplicaciones ("aplicaciones flexibles") hacen uso de cualquier ancho de banda que tengan a su disposición.

Requisitos de los servicios de transporte para aplicaciones comunes

Aplicación	Pérdida de datos	Ancho de banda	Sensible al tiempo
Transferencia de archivos	No pérdida	flexible	No
Correo electrónico	No pérdida	flexible	No
Documentos Web	No pérdida	flexible	No
Audio/vídeo de tiempo real	Tolerante	Audio: 5Kbps-1Mbps Vídeo:10Kbps-	Sí, 100 mseg
Audio/vídeo almacenado	Tolerante	5Mbps	Sí, pocos seg
Juegos interactivos	Tolerante	Igual que el anterior	Sí, 100 mseg
Mensajería instantánea	No pérdida	Pocos Kbps-10Kbps Flexible	Sí y no

Servicios de los protocolos de transporte de Internet

Servicio TCP:

- ❑ *Orientado a la conexión:*
Sistema requerido entre el cliente y el servidor.
- ❑ *Transporte fiable* entre el proceso emisor y el receptor.
- ❑ *Control de flujo:* el emisor no debe sobrecargar al receptor.
- ❑ *Control de congestión:*
regulación del emisor si la red se sobrecarga.
- ❑ *No proporciona:*
temporización, garantías de un ancho de banda mínimo.

Servicio UDP:

- ❑ Transferencia de datos no fiable entre el proceso emisor y el receptor.
- ❑ No proporciona: sistema de conexión, fiabilidad, control de flujo, control de congestión, temporización y garantía de ancho de banda.

PREGUNTA: ¿Por qué tomarse la molestia? ¿Por qué existe un UDP?

Aplicaciones de Internet: aplicación, protocolos de transporte

Aplicaciones	Protocolo de la capa de aplicación	Protocolo de transporte subyacente
Correo electrónico	SMTP [RFC 2821]	TCP
Acceso a terminales remotos	Telnet [RFC 854]	TCP
Web	HTTP [RFC 2616]	TCP
Transferencia de archivos	FTP [RFC 959]	TCP
Flujo de multimedia	Propietario (por ejemplo, Real Networks)	TCP o UDP
Telefonía Internet	Propietario (por ejemplo, Dialpad)	Típicamente UDP