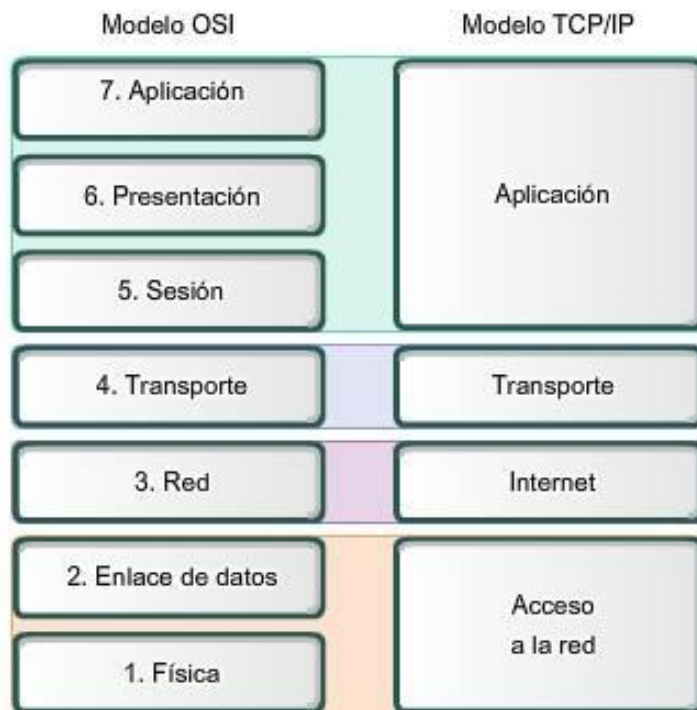


Introducción a Sockets UDP en C

Facultad de Ingeniería
Universidad de Buenos Aires

Modelos de Capas

Comparación del modelo OSI con el modelo TCP/IP



Las semejanzas claves están en la capa de Red y de Transporte.

TCP/IP o UDP/IP

- Capa de Red o Internet (IP)
 - Permite enviar y recibir datos entre máquinas
 - No identifica la aplicación que envía/recibe
- Capa de Transporte
 - Nro de Puerto identifica el destino (aplicación) de la máquina
 - Protocolos:
 - TCP – Transmission Control Protocol
 - UDP – User Datagram Protocol

TCP – Transmission Control Protocol

- Orientado a la **conexión**
- Los paquetes se reciben **TODOS** y en **ORDEN**
 - Se envía un "**acknowledge**" para cada paquete recibido
 - Se valida cada paquete mediante un **Checksum**
- Los datos pueden ser transmitidos **simultáneamente en ambas direcciones**
- “ai.socktype=SOCK_STREAM”



UDP - User Datagram Protocol

- **Sin conexión** (Datagramas). Si **Multicast/Broadcast**
- Los datagramas **pueden perderse** o llegar **duplicados** o **fuera de secuencia**; pero si llegan llegan **completos**.
 - Se utiliza un **Checksum** pero sólo para descartar paquetes (no hay retransmisión)
- Los datos pueden ser transmitidos **simultáneamente en ambas direcciones**
- `"ai.socktype=SOCK_DGRAM"`



Programming: sockets (Berkeley)

TCP	UDP
socket	socket
bind	bind
listen	
accept	
connect	
send/recv	sendto/recvfrom
shutdown/close	close

Programming: Ejemplo Servidor

/ Creamos el socket */*

int Descriptor;

Descriptor = socket (AF_INET, SOCK_DGRAM, 0);

/ Nos linkeamos a un puerto */*

struct sockaddr_in Direccion;

Direccion.sin_family = AF_INET;

Direccion.sin_port = ... ; */* puerto elegido */*

Direccion.sin_addr.s_addr = INADDR_ANY;

bind (Descriptor, (struct sockaddr *)&Direccion, sizeof (Direccion));

/ Recibimos del Remitente*/*

struct sockaddr_in Remitente; */* Datos del Remitente*/*

int longitudRemitente = sizeof(Remitente);

char buffer[100];

recvfrom (Descriptor, buffer, sizeof(buffer), 0, (struct sockaddr *)&Remitente, &longitudRemitente);

/ Respondemos al Remitente*/*

sendto (Descriptor, buffer, sizeof(buffer), 0, (struct sockaddr *)&Remitente, longitudRemitente);

Programming: Ejemplo Cliente

/ Creamos el socket */*

int Descriptor;

Descriptor = socket (AF_INET, SOCK_DGRAM, 0);

/ Nos linkeamos a un puerto */*

struct sockaddr_in Direccion;

Direccion.sin_family = AF_INET;

Direccion.sin_port = 0; */* Dejamos que el sistema elija el puerto */*

Direccion.sin_addr.s_addr = INADDR_ANY;

bind (Descriptor, (struct sockaddr *)&Direccion, sizeof (Direccion));

/ Enviamos al Destinatario*/*

struct sockaddr_in Destinatario; */* Datos del Destinatario*/*

Destinatario.sin_family = AF_INET;

Destinatario.sin_port = ...; */* Puerto del Server */*

Destinatario.sin_addr.s_addr = ...; */* IP del Server */*

int longitudDestinatario = sizeof(Destinatario);

char buffer="hola";

sendto (Descriptor, buffer, sizeof(buffer), 0, (struct sockaddr *)&Destinatario, longitudDestinatario);