TRANSPORTE:

FUNCIONES:

Protocolos:

* UDP
* TCP

Similitudes:

• Multiplexación: Sobre un mismo IP, se poseen puertos en los cuales se pueden tener muchos servicios.

• Conexión/desconexión:

• Intercambio de datos y control de flujo

• Casos de baja eficiencia en TCP

• Control de congestión

Diferencias:

Funciones:

• Entrega datos de extremo a extremo (host to host)

• Realiza la comunicación de forma transparente al medio físico

• Multiplexa tráfico de diversas instancias del nivel de aplicación. Hay una sola instancia del nivel de transporte en el host. (Usa diferentes puertos, puede correr varios servicios en simultáneo por el puerto)

• Ofrece:

* Servicio orientado a conexión
* Servicio no orientado a conexión

PROTOCOLO UDP:

• No orientado a conexión. Hace intercambio de mensajes entre host y host

Se utiliza en:

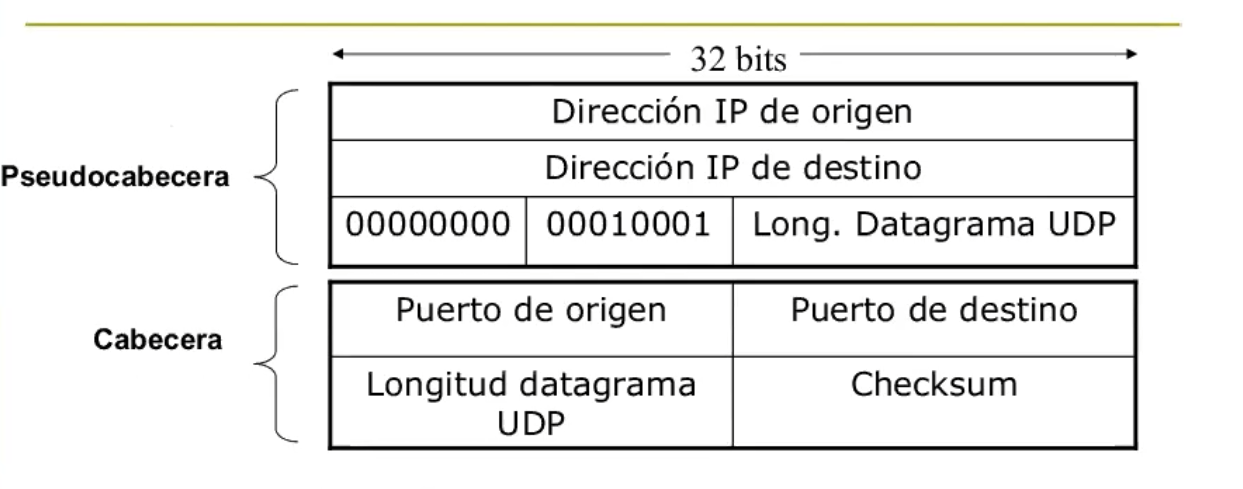
* Intercambio de mensajes escasos (DNS)
* Aplicación a tiempo real, no puede esperar confirmaciones (videoconferencia, voz sobre IP)
* Los mensajes se producen regularmente y no importa si se pierde alguno (NTP, SNMP)
* El medio de transmisión es altamente fiable y sin congestión (LAN’s)
* Se envía tráfico broadcast/multicast

• TPDUs en UDP se llaman mensajes o datagramas UDP

• UDPO multiplexa los datos de las aplicaciones y efectúa opcionalmente una comprobación de errores, pero no controla el flujo ni la congestión, ni retransmite datos perdidos. Tampoco tiene conexión y desconexión ya que es un solo mensaje. Se realiza del puerto 0 al 65535. Del 0 al 1023 están reservados para well known ports.

PARTES:

Cabecera UDP:



SOCKET: Relación puerto – ip.

Ej:

147.156.135.22.1038

PROTOCOLO TCP:

• Orientado a conexión.

• Ofrece transporte fiable sobre un servicio no fiable del nivel de red (dado por el IP)

• Las TPDUs de TCP se llaman segmentos.

• Especificado por RFC 793

• Requiere un procedimiento de establecimiento y terminación de la comunicación

• Durante una conexión, mantienen info sobre la misma, conocida como información de estado (contadores de bytes, espacio libre en buffers, etc)

• Se suele separar en servidor y cliente. El cliente hace peticiones y el servidor las espera para conectarse.

• Una conexión puede terminarse por ambas partes

• Se puede usar el modelo igual a igual (peer to peer)

FUNCIONES:

• Puertos

• Controla errores, retransmitiendo segmentos perdidos o erróneos. Elimina duplicados

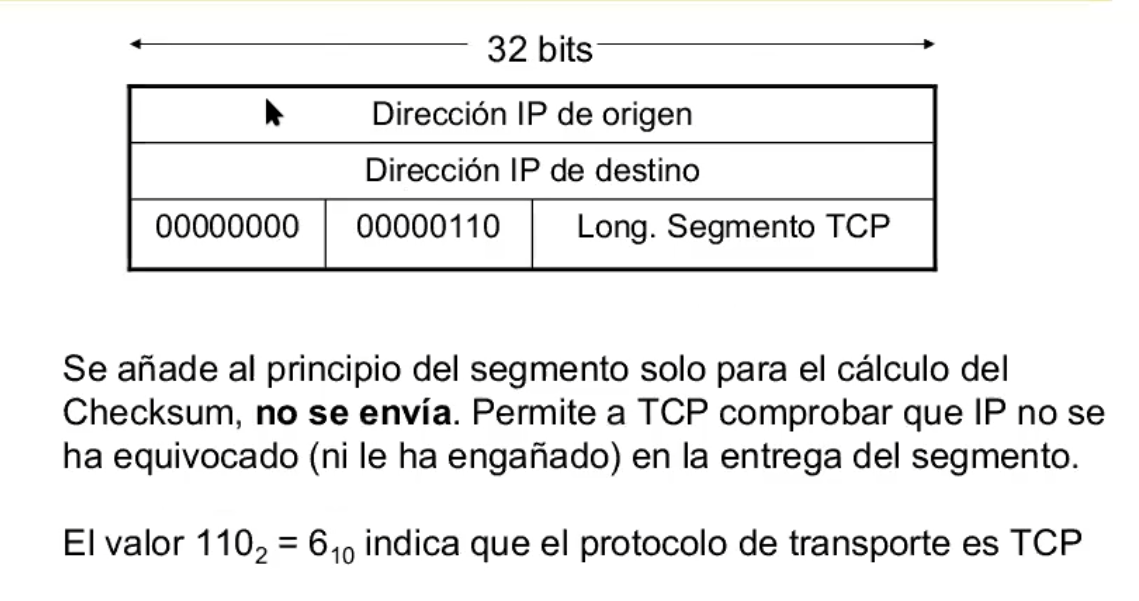
• Establece y termina conexiones

• Gestiona los buffers y ejerce control de flujo de forma eficiente

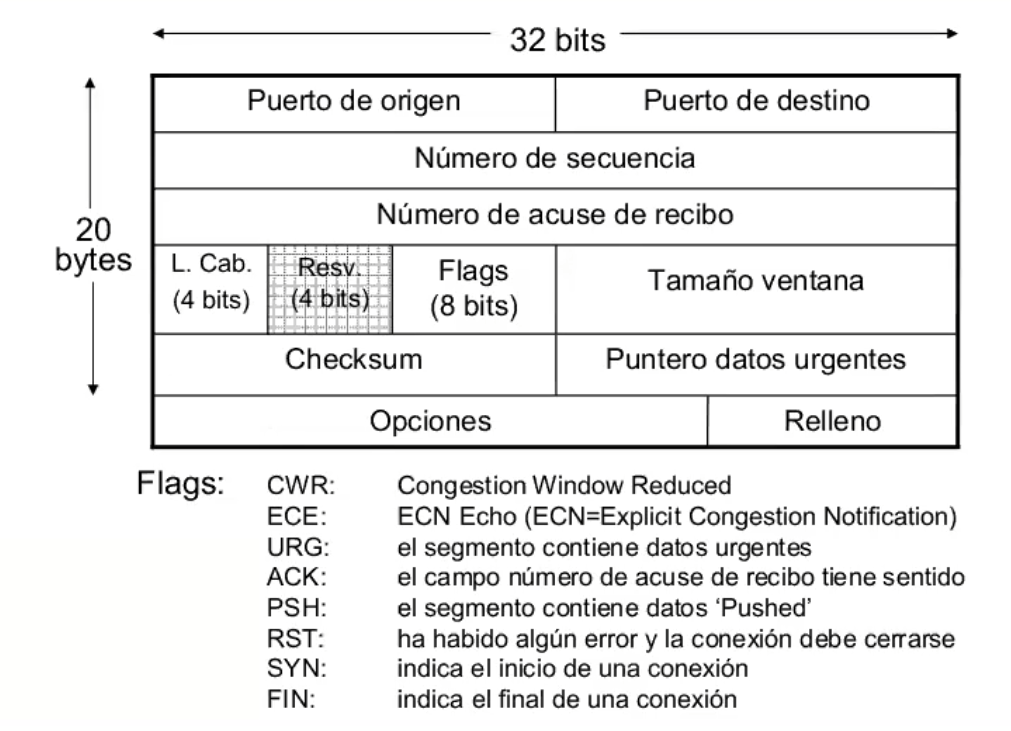
• Gestiona el intercambio de datos con las aplicaciones

• Efectúa control de congestión

PSEUDOCABECERA:



CABECERA:



\* Tamaño variable en ipv4, en ipv6 es fija.

* Los flags corresponden a la posición. Si tiene un 1 en determinado bit, significa que el flag correspondiente a dicho bit, está activo
* El tamaño de ventana explica cuántos mensajes puedo enviar antes de esperar un acuse de recibo.
* Numero de secuencia: número del paquete enviado, realmente es un corrimiento de los bits pasados mostrados como corrimiento.
* Numero de acuse de recibo: Indica que el anterior segmento fue mandado, por lo que manda el primer bit del segmento que le sigue. Ej, si recibió del 0 al 1500, envía el 1501

• En UDP se manda a sockets, en TCP se manda a un socket en especifico ya que está una comunicación.

• La cantidad máxima de conexiones es igual a la de puertos que posea.

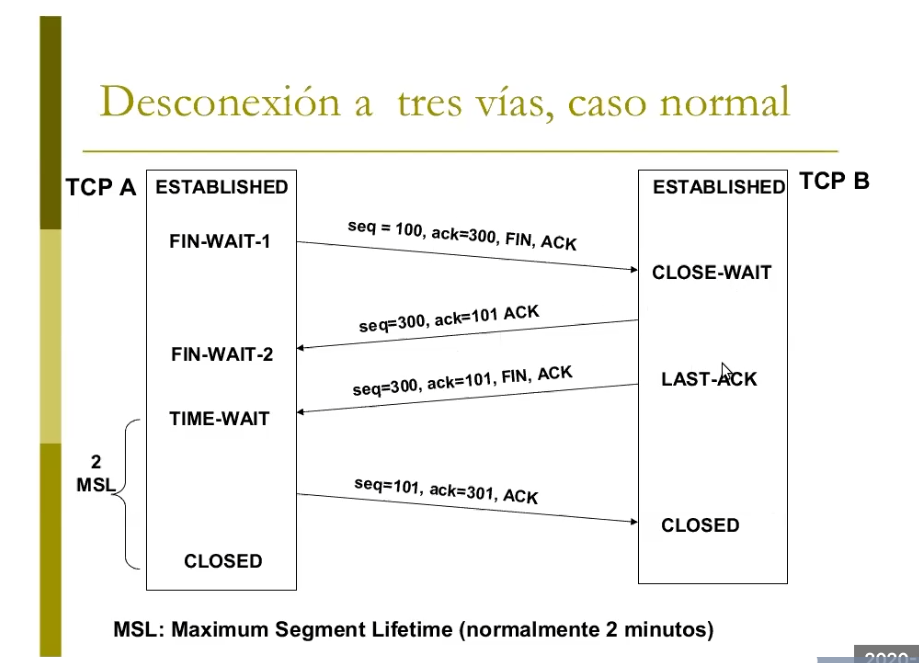
SALUDO DE TRES VÍAS (Handshake)

1. El cliente envía un número único por cada intento de conexión. Se incluye en la petición (CON SYN)
2. El servidor al recibir la petición, elige otro número único y se lo envía al cliente indicándoselo. (CON ACK, el cual no modifica el numero INS)
3. El cliente al recibir la respuesta, considera que ya está hecha la conexión. Envía un tercer mensaje en el que dice que recibió lo anterior. Al recibir este mensaje, el servidor considera que ya ha sido iniciada. (CON ACK )

DESCONEXIONES:

* Simétrica: Hay dos simplex, al cortarse uno se invita a que se corte el otro
* Asimétrica, desconexión en ambos sentidos sin avisarle al otro. Puede causar la pérdida de paquetes.

CASO NORMAL:



a