

Tema 4

Nivel de Transporte

1

Mecanismos del protocolo de la capa de transporte orientado a conexión

- ⌘ Conexión lógica
- ⌘ Establecimiento
- ⌘ Mantenimiento y cierre
- ⌘ Seguridad y Control de Flujo
- ⌘ Ejemplo: TCP

Servicio de red de secuenciamiento seguro

- ⌘ Acepta mensajes con un tamaño arbitrario
- ⌘ Envía los mensajes en secuencia al destino con una seguridad virtual del 100%
- ⌘ El protocolo de transporte se utiliza como un protocolo extremo-a-extremo entre dos sistemas finales conectados en la misma red

Elementos de un protocolo de transporte sencillo

- ⌘ Direccionamiento
- ⌘ Multiplexación
- ⌘ Control de flujo y de errores
- ⌘ Establecimiento/cierre de la conexión

Identificación

Destino <-----> Proceso de Aplicación

Problemas

- ☒ Procesos que se crean y se destruyen
- ☒ Reemplazamiento de procesos
- ☒ Identificación de destino basado en funciones
- ☒ Procesos multifunción

Solución ==> PUERTOS DEL PROTOCOLO

Sistema Operativo realiza la función de interfaz entre

PROCESO <====> PUERTO DEL PROTOCOLO

5

TCP y UDP

⌘ Protocolo de Control de la Transmisión (TCP)

- ☒ Orientado a conexión
- ☒ Especificado en el RFC 793

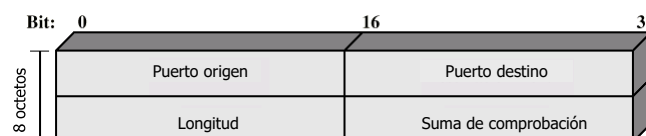
⌘ Protocolo Datagrama de Usuario (UDP)

- ☒ No orientado a conexión
- ☒ Especificado en el RFC 768

UDP

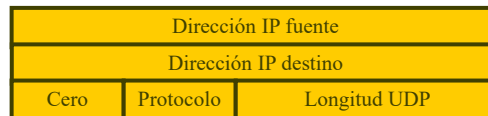
- ⌘ Protocolo de datagrama de usuario
- ⌘ Servicio no orientado a conexión para procedimientos de la capa de aplicación
 - ☑ Servicio no seguro
 - ☑ La entrega y la seguridad no están garantizadas
- ⌘ Se reduce la información suplementaria
- ⌘ Añade la capacidad de distinguir entre múltiples destinos de un Sistema Informático

Cabecera UDP

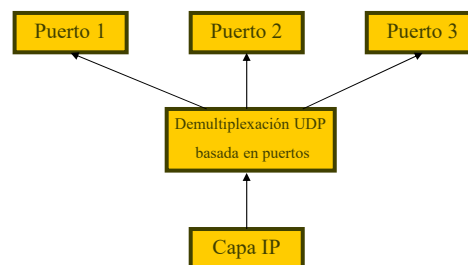


UDP

Pseudoencabezamiento



Multiplexado



9

Servicios TCP

- ⌘ Comunicación segura entre pares de procesos
- ⌘ A través de una gran variedad de redes seguras e inseguras, así como sobre un conjunto de redes interconectadas
- ⌘ Dos facilidades para etiquetar datos
 - ☑ Cargar flujo de datos
 - ☑ Indicación de datos urgentes

Transport Control Protocol (TCP)

Propiedades

- ☒ Orientado a Ristras
- ☒ Conexión por Circuito Virtual
- ☒ Transferencia "Buffereada"

Fiabilidad

- ☒ Control de Errores
- ☒ Secuenciación
- ☒ Mecanismo de Ventanas Deslizantes

11

Transport Control Protocol (TCP)

Puertos, Conexiones y Puntos Finales

- ☒ TCP usa la conexión, no el puerto del protocolo, como abstracción fundamental

⌘ Las Conexiones se identifican mediante un par de puntos finales

⌘ Punto Final: Par de números compuestos por
(Dirección IP, Puerto)

Ejemplo Punto Final
(128.10.2.3, 25)

Ejemlo de Conexión
(128.9.0.32, 1184) y (128.10.2.3, 53)

12

Transport Control Protocol (TCP)

Aperturas de Conexión

- ☒ **Pasiva:** Disponibilidad para aceptar una conexión
- ☒ **Activa:** Solicitud de establecimiento de una conexión

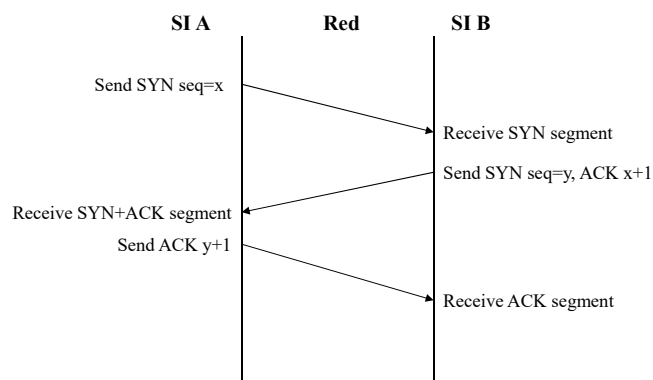
Puertos Bien Conocidos

- ☒ Las aplicaciones que proporcionan servicios estandarizados (correo, ftp, etc.) suele establecer conexiones pasivas sobre puertos establecidos a priori para tal fin.
- ☒ En prácticamente todos los S.O. existe un fichero denominado services que indica el número de puerto donde escuchan las aplicaciones anteriormente citadas.

13

Transport Control Protocol (TCP)

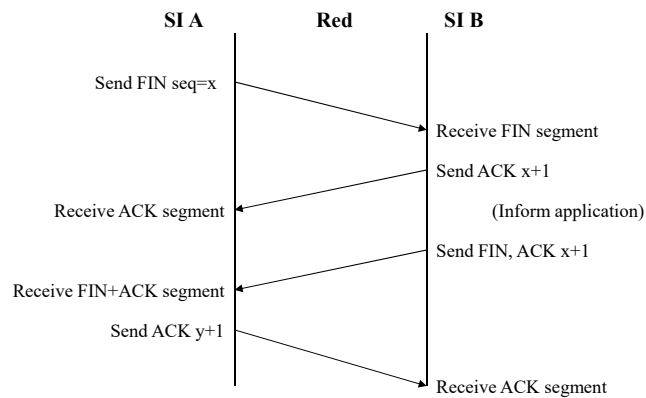
Establecimiento de la Conexión (3 way handshake)



14

Transport Control Protocol (TCP)

Liberación de la Conexión



15

Transport Control Protocol (TCP)

Mecanismo de Ventanas Deslizantes

- ☒ Eficiencia y Control de Flujo: El mecanismo de ventanas deslizantes en TCP opera a nivel de octeto y no a nivel de segmento



- ☒ La apertura de la ventana viene controlada por el protocolo
- ☒ Campo Ventana: Especifica el número de octetos adicionales que el receptor está preparado para recibir

16

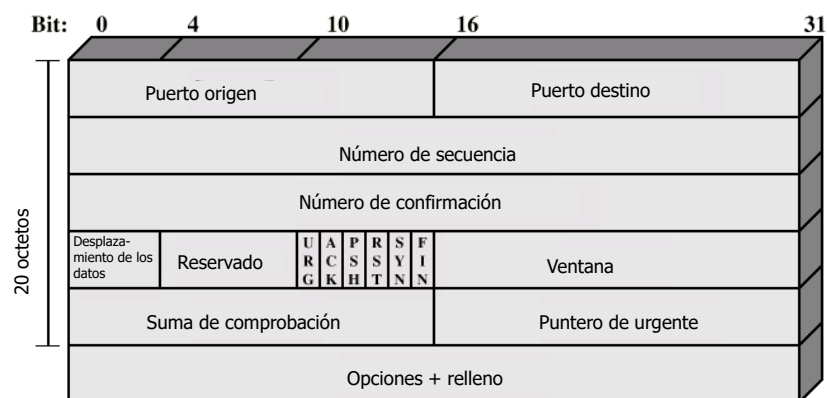
Transport Control Protocol (TCP)

⌘ Segmento TCP

- ☒ Es la unidad de transferencia entre el software tcp de dos máquinas
- ☒ Los segmentos se intercambia para
 - ☒ Establecer conexiones
 - ☒ Transferir datos
 - ☒ Enviar acuses de recibo (reconocimientos)
 - ☒ Anunciar tamaños de ventana
 - ☒ Cerrar conexiones

17

Cabecera TCP



Transport Control Protocol (TCP)

Tamaño máximo del segmento

- ☒ El campo de Opciones sirve para negociar el tamaño máximo de los mensajes entre ambos extremos de la conexión
- ☒ Norma
 - ☒ Sistemas Informáticos en la misma red

Tamaño máximo se elige de tal forma que los datagramas IP resultantes coincidan con el tamaño del paquete de la red (MTU)
 - ☒ Sistemas Informáticos en diferentes redes

Tamaño máximo recomendado : $536 = 576 - \text{Cabecera IP+TCP}$

Valor IP por defecto 40 Octetos
- ☒ Problemática
 - ☒ Tamaño demasiado corto => Baja Eficiencia

$D = D/(D+H)$

19

Transport Control Protocol (TCP)

- ☒ Tamaño demasiado largo => Baja Eficiencia

Fragmentación + Posibilidad de pérdida de fragmentos
- ☒ Tamaño Óptimo = Aquel con el mayor tamaño de forma que los datagramas IP que lo portan no necesiten fragmentación
- ☒ Dificultad
 - ☒ TCP no incluye mecanismo para ello
 - ☒ Las pasarelas pueden cambiar las rutas dinámicamente
 - ☒ Depende de los encabezamientos de los protocolos de nivel inferior

Cálculo del Checksum

Dirección IP fuente		
Dirección IP destino		
Cero	Protocolo	Longitud TCP

20

Transport Control Protocol (TCP)

Determinación del intervalo de timeout

Método: Algoritmo de retransmisión adaptativo

$$RTT = (\alpha * Old_RTT) + ((1 - \alpha) * New_RTT_Sample)$$

$\alpha \rightarrow 1 \Rightarrow$ cambios suaves

$\alpha \rightarrow 0 \Rightarrow$ cambios bruscos

$$TimeOut = \beta * RTT \quad (\beta \approx 2)$$

Problema: Segmentos retransmitidos

RTT = Tiempo transcurrido entre el envío de un segmento y la recepción de su reconocimiento

Transport Control Protocol (TCP)

Control de la Congestión

Congestión \Rightarrow Mayor Retardo \Rightarrow Retransmisión \Rightarrow

Mayor Congestión \Rightarrow Mayor Retardo \Rightarrow Retransmisión \Rightarrow

.

.

Mayor Congestión \Rightarrow Mayor Retardo \Rightarrow **Colapso**

Solución: Reducir la velocidad de transmisión, mediante la apertura y cierre de la ventana de transmisión