

# Redes y Comunicaciones

Ulises J. Cornejo Fandos

Abril 2017

## PRÁCTICA 5

---

### 1.1 Capa de transporte - Parte 1

#### 1. ¿Cuál es la función de la capa de Transporte?

Proporciona directamente servicios de comunicación a los procesos de aplicación que se ejecutan en diferentes hosts. Amplía el servicio de entrega de la capa de red entre dos sistemas terminales a un servicio de entrega entre dos procesos de la capa de aplicación. La capa de transporte entonces proporciona comunicación lógica entre procesos en distintos hosts mientras que la capa de red proporciona comunicación lógica entre hosts.

Es la que transporta mensajes entre los puntos terminales de una red.

#### 2. Describe la estructura del segmento TCP y UDP.

La cabecera de un segmento UDP solo tiene 4 campos y cada uno de ellos tiene 2 bytes:

- puerto de origen,
- puerto de destino,
- longitud,
- suma de comprobación.

Los números de puertos permiten dirigir el segmento al socket correspondiente, es decir, el socket UDP es identificado por un puerto origen y un puerto destino.

#### 3. ¿Cuál es el objetivo del uso de puertos en el modelo TCP/IP?

La identificación de procesos.

#### 4. Compare TCP y UDP en cuanto a:

##### (a) Confiabilidad

**Asegura la entrega de todos los paquetes enviados, permitiendo mantener en el receptor el orden en que le fueron enviados (utilizando confirmación de recepción, timeout, reenvío de paquetes y números de secuencia) y detectar errores en la recepción (mediante checksum).**

TCP: sí, utiliza establecimiento de conexión de tres pasos.

UDP: No, sin conexión, solo ofrece el best effort IP.

##### (b) Multiplexación

**Extender la comunicación host-host que ofrece IP permitiendo diferenciar los procesos en el receptor y en el emisor, de modo que varios procesos en un host se comuniquen con varios procesos en otros hosts, sin confundir los paquetes (utilizando direcciones IP y números de puerto que identifican a los procesos).**

TCP: Socket (IP Orig, Port Orig, IP Dest, Port Dest)

UDP: Socket (IP Dest, Port Dest)

- (c) Orientado a la conexión  
**Controlar la velocidad de transmisión del emisor para no sobrecargar la red.**  
 TCP: SI  
 UDP: NO
  - (d) Controles de congestión  
**Controlar la velocidad de transmisión del emisor para no sobrecargar al receptor (según el tamaño de su buffer, ventana de recepción).**  
 TCP: SI  
 UDP: NO
  - (e) Utilización de puerto  
**Los puertos destino y origen son utilizados tanto en TCP como en UDP por fines de multiplexación / desmultiplexación. UDP sólo precisa el puerto, mientras que TCP también precisa IP origen y destino.**  
 TCP: Port origen y destino  
 UDP: Port origen y destino
5. ¿Cuál es el campo del datagrama IP y los valores que se utilizan en este para diferenciar que se transporta TCP o UDP? (Ayuda: buscar en /etc/protocols y contrastarlo con una captura de tráfico).
6. La PDU de la capa de transporte es el segmento. Sin embargo, en algunos contextos suele utilizarse el término Datagrama, indique cuándo. Datagrama = ¿ PDU de la capa de Transporte
- Un datagrama es una cabecera + datos, tal que la cabecera ofrece la información necesaria para que los datos puedan ser encaminados y alcancen un destino.
- Se suele hablar de Datagrama en referencia a los segmentos en el protocolo UDP -Protocolo de Datagrama de Usuario-. Sin embargo, como datagrama es el nombre formal de los PDU de la capa de Red -protocolo IP- es preferible evitar esta denominación.
7. Describa el saludo de tres vías de TCP.
- El cliente realiza una conexión enviando un paquete SYN al servidor, en el servidor se comprueba si el puerto está abierto (si existe un proceso escuchando por ese puerto), si el puerto no está abierto se le envía al cliente un paquete de respuesta RST, esto significa un rechazo de intento de conexión. Si el puerto está abierto, el servidor responde con un paquete SYN/ACK. Entonces el cliente respondería al servidor con un ACK, completando así la conexión.
- **ACK o “acknowledge” (1 bit):** Si está activo entonces el campo con el número de acuse de recibo es válido (si no, es ignorado).  
 Es un mensaje que se envía para confirmar que un mensaje o un conjunto de mensajes han llegado.
  - **SYN o “synchronize” (1 bit):** Activa/desactiva la sincronización de los números de secuencia.  
 Se usa para sincronizar los números de secuencia en tres tipos de segmentos: petición de conexión, confirmación de conexión (con ACK activo) y la recepción de la confirmación (con ACK activo).
  - **RST o “reset” (1 bit):** Si llega a 1, termina la conexión sin esperar respuesta.  
 Es un bit que se encuentra en el campo del código en el protocolo TCP, y se utiliza para reiniciar la conexión. Un ejemplo práctico de utilización es el que realiza un servidor cuando le llega un paquete a un puerto no válido: este responde con el RST activado.
8. Utilice el comando ss (reemplazo de netstat) para obtener la siguiente información de su PC:

- (a) Para listar las comunicaciones TCP establecidas.
    - `ss -t`
    - `ss -A tcp`
  - (b) Para listar las comunicaciones UDP establecidas.
    - `ss -u`
    - `ss -A udp`
  - (c) Obtener sólo los servicios TCP que están esperando comunicaciones
    - `ss -l -t`
    - `ss -l -A tcp`
  - (d) Obtener sólo los servicios UDP que están esperando comunicaciones.
    - `ss -l -u`
    - `ss -l -A udp`
  - (e) Repetir los anteriores para visualizar el proceso del sistema asociado a la conexión.  
**ejecuto los mismos comandos con el parametro -p (process).**
  - (f) Obtenga la misma información planteada en los items anteriores usando el comando `netstat`.  
**Mismo parametros que ss.** Es la misma consulta solo que *netstat* devuelve una respuesta más limpia.
9. **¿Qué sucede si llega un segmento TCP a un host que no tiene a ningún proceso esperando en el puerto destino de dicho segmento?**  
 Se retorna un response TCP con el flag RESET seteado en 1 indicando que dicha conexión no debe ser utilizada.
- (a) **Utilice `hping3` para enviar paquetes TCP al puerto destino 22 de la máquina virtual con el flag SYN activado.**  
`hping -s 8080 localhost -S -p 22,`  
 donde -s indica el puerto origen ,-p indica el puerto y -S es el flag SYN en 1.
  - (b) **Utilice `hping3` para enviar paquetes TCP al puerto destino 40 de la máquina virtual con el flag SYN activado.**  
`hping -s 8080 localhost -S -p 40,`  
 donde -s indica el puerto origen ,-p indica el puerto y -S es el flag SYN en 1.
10. **¿Qué sucede si llega un datagrama UDP a un host que no tiene a ningún proceso esperando en el puerto destino de dicho datagrama (es decir, que dicho puerto no está en estado LISTEN)?**  
 UDP, al no tener flag ACK, no responde de ninguna forma, ignorando así el mensaje.
- (a) **Utilice `hping3` para enviar datagramas UDP al puerto destino 68 de la máquina virtual.**  
`hping -s 8080 localhost -udp -p 68,`  
 donde -s indica el puerto origen ,-p indica el puerto.
  - (b) **Utilice `hping3` para enviar datagramas UDP al puerto destino 40 de la máquina virtual.**  
`hping -s 8080 localhost -udp -p 40,`  
 donde -s indica el puerto origen ,-p indica el puerto.
11. **Investigue qué es multicast. ¿Sobre cuál de los protocolos de capa de transporte funciona? ¿Se podría adaptar para que funcione sobre el otro protocolo de capa de transporte? ¿Por qué?**  
 Multidifusión IP, en inglés IP Multicast, es un método para transmitir datagramas IP a un grupo de receptores interesados. Funciona sobre el protocolo UDP, y no en TCP dado que este está implementado para la conexión de únicamente dos host, y la implementación de un Multicast en TCP se vería afectada por esto.