

Control de Redes de Computadores (XC), Grado en Ingeniería Informática		20/11/2014	otoño 2014
Nombre:	Apellidos:	DNI	

Duración: 1h15m. El test se recogerá en 30 minutos. Responder en el mismo enunciado.

Test. (4 puntos) Todas las preguntas son multi-respuesta: Cada pregunta vale la mitad si hay un error, 0 si más.

1. Indica qué expresiones son ciertas para UDP:

- ☐ La cabecera UDP tiene un campo TTL que se decrementa a cada salto para limitar su duración.
- ☒ La cabecera UDP tiene un campo "checksum" para proteger el paquete de cambios entre origen y destino.
- ☒ Un paquete UDP puede ser fragmentado por el protocolo IP.
- ☐ UDP tiene control de flujo pero no control de congestión.

2. En la fase de "slow start" la ventana de congestión puede crecer:

- ☒ Un segmento más por cada ACK que confirma nuevos datos.
- ☐ Un segmento más por cada RTT.
- ☐ Se duplica cada ACK que confirma nuevos datos.
- ☒ Se duplica cada RTT.

3. En el régimen permanente de TCP (congestion avoidance) la ventana de congestión puede crecer:

- ☐ 1 MSS cada ACK que confirma nuevos datos.
- ☒ 1 MSS cada RTT.
- ☐ 1 MSS cada segundo.
- ☐ 1 MSS cada RTO.

4. El tamaño de la ventana anunciada por el receptor:

- ☒ Cambia a medida que se reciben datos.
- ☒ Cambia a medida que un proceso lee del buffer.
- ☐ No cambia durante la conexión.
- ☐ Cambia según la congestión de la red.

5. En un cierto instante durante una transferencia TCP la ventana anunciada del receptor (awnd) es de 5000 bytes, la ventana de congestión es de 8000 bytes y la variable ssthresh vale 4500. ¿Qué afirmaciones son correctas?

- ☐ El último RTO se ha producido cuando la ventana era de 4500 bytes.
- ☒ El último RTO se ha producido cuando la ventana era de 9000 bytes.
- ☐ Estamos en la fase de slow start.
- ☒ Estamos en la fase de congestion avoidance.

6. El número de secuencia en TCP:

- ☐ Es un valor que comienza en cero y cuenta segmentos enviados.
- ☐ Es un valor que comienza en un valor aleatorio y cuenta segmentos enviados.
- ☐ Es un valor que comienza en cero y cuenta bytes enviados.
- ☒ Es un valor que comienza en un valor aleatorio y cuenta bytes enviados.

7. Cuando el host origen y destino se comunican sin que se produzcan pérdidas en el camino, se cumple que:

- ☐ La fase de slow start acaba cuando hay una reducción de la ventana del receptor (awnd).
- ☒ La velocidad de transferencia queda limitada por la ventana anunciada por el receptor si el RTT es grande.
- ☐ La fase de slow start no existe pues se toma la ventana anunciada por el receptor directamente.
- ☐ La fase de slow start se repite al menos dos veces.

8. Cuando hay varias conexiones TCP con algunos saltos comunes en el camino, en promedio:

- ☐ La capacidad se reparte en proporciones iguales.
- ☐ La capacidad se reparte con preferencia por conexiones TCP de mayor RTT.
- ☒ La capacidad se reparte con preferencia por conexiones TCP de menor RTT.
- ☒ La capacidad se reparte con preferencia por conexiones TCP de mayor ventana anunciada del receptor.

9. Cuando un segmento se pierde y se recibe el siguiente en TCP el receptor puede reaccionar:

- ☒ Enviando un ACK del último recibido en orden para que se retransmita el segmento perdido.
- ☒ Enviando un ACK duplicado para que se retransmita el segmento perdido.
- ☒ Esperando RTO sin enviar ningún ACK para que se retransmita el segmento perdido.
- ☐ Enviando un RESET para que se ese retransmita el segmento perdido.

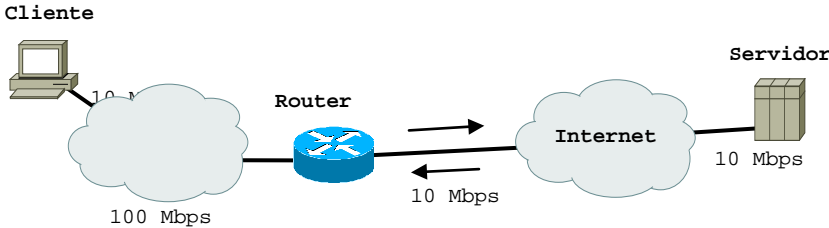
Correcto con 2 o 3 resultados pues diferentes versiones de TCP tienen comportamientos diferentes.

Primer control de Xarxes de Computadors (XC), Grau en Enginyeria Informàtica		31/03/2014	Primavera 2014
NOM:	COGNOMS	DNI	

Duració: 1h15m. El test es recollirà en 30 minuts. Respondre en el mateix enunciat.

Pregunta 1.

El volcado de la figura presenta el intercambio de mensajes TCP entre el servidor y el cliente que aparecen en la figura. En este volcado no aparecen todos los mensajes (se ha indicado con ... las partes eliminadas). Contestar a las siguientes preguntas. Motivar brevemente las respuestas.



0.000	202.2.2.1.2104 > 150.214.5.1.995: S 655237:655237(0) win 32738 <mss 1452,nop,nop,wscale 0>
0.002	150.214.5.1.995 > 202.2.2.1.2104: S 16122349:16122349(0) ack 655238 win 4180 <mss 1452,nop,nop,wscale 0>
0.151	202.2.2.1.2104 > 150.214.5.1.995: . ack 1 win 32768
...	
2.410	150.214.5.1.995 > 202.2.2.1.2104: . 29248:30700 (1452) ack 120 win 4180
2.561	202.2.2.1.2104 > 150.214.5.1.995: . ack 26344 win 32768
2.561	150.214.5.1.995 > 202.2.2.1.2104: . 30700:32152(1452) ack 120 win 4180
2.562	150.214.5.1.995 > 202.2.2.1.2104: . 32152:33604(1452) ack 120 win 4180
2.711	202.2.2.1.2104 > 150.214.5.1.995: . ack 26344 win 32768
2.732	202.2.2.1.2104 > 150.214.5.1.995: . ack 26344 win 32768
3.051	150.214.5.1.995 > 202.2.2.1.2104: . 26344:27796(1452) ack 120 win 4180
3.200	202.2.2.1.2104 > 150.214.5.1.995: . ack 32152 win 32768
...	
5.211	150.214.5.1.995 > 202.2.2.1.2104: F 60918:60918(0) ack 120 win 4180
5.360	202.2.2.1.2104 > 150.214.5.1.995: . ack 60919 win 32768
5.366	202.2.2.1.2104 > 150.214.5.1.995: F 120:120(0) ack 60919 win 32768
5.367	150.214.5.1.995 > 202.2.2.1.2104: . ack 121 win 4180
5.872	202.2.2.1.2104 > 150.214.5.1.995: F 120:120(0) ack 60919 win 32768
5.873	150.214.5.1.995 > 202.2.2.1.2104: . ack 121 win 4180

a) (3.5 puntos) Considerando el volcado anterior y sabiendo que las aplicaciones escriben y leen los buffers más rápidamente que la red en transmitir y entregar los datos,

1) Identificar

	Cliente	Servidor
Dirección IP	202.2.2.1	150.214.5.1
Puerto	2104	995
Tamaño del buffer de RX	32738 bytes	4180 bytes
MSS	1452 bytes	1452 bytes
Cantidad de bytes de datos transmitidos durante toda la conexión TCP	119 bytes	60917 bytes

2) Identificar y motivar donde se ha hecho la captura

En el servidor por los timestamps del 3WH

3) Determinar el RTT (tiempo de ciclo) del TCP

Por los timestamps del 3WH se encuentra que son alrededor de 150 ms

4) Si no hubiera perdidas, estimar la velocidad efectiva máxima en régimen estacionario

vef = min (enlace mas lento, wnd/RTT)
Enlace mas lento = 10 Mbit/s
En régimen estacionario, sin perdidas, wnd = awnd = 32738 bytes -> awnd/RTT = 1.75 Mbits/s
Vef = 1.75 Mbit/s

Pregunta 2. (6 puntos)

5) Calcular la velocidad media que se ha conseguido realmente

$$v_{media} = 60917 \cdot 8 / 5.873 \text{ s} = 83 \text{ kbit/s}$$

6) Razonar si ha habido pérdidas y encontrar cuales segmentos se han perdido

Se ha perdido:

- 26344 por la retransmisión de este segmento
- 32152 ya que se ha transmitido en el tiempo 2.562 pero al retransmitir el 26344, el cliente confirma solo hasta 32152
- el primer ack 121 que transmite el servidor ya que el cliente vuelve a enviar su fin

b) (2.5 puntos) Dejando ahora el volcado, suponer que, durante toda la transferencia de datos, cuando la ventana de transmisión llega a 8 MSS, Internet siempre pierde el séptimo segmento. Suponiendo que el $RTO = RTT$, completar la tabla a continuación indicando claramente el valor de la ventana de congestión, si se aplica Slow Start (SS) o Congestion Avoidance (CA) y el umbral $sssthresh$. Suponer que el cliente siempre anuncia su ventana $awnd$ máxima.

Solución si con $cwnd == sssthresh$ se aplica SS

RTT	cwnd (MSS)	SS o CA	sssthresh
0	1	SS	∞
1	2	SS	∞
2	4	SS	∞
3	8	SS	∞
4	14	SS	∞
5	1	SS	7
6	2	SS	7
7	4	SS	7
8	8	SS	7
9	8.75 (aprox.)	CA	7
10	1	SS	4.375
11	2	SS	4.375
12	4	SS	4.375

Solución si con $cwnd == sssthresh$ se aplica CA

RTT	cwnd (MSS)	SS o CA	sssthresh
0	1	SS	∞
1	2	SS	∞
2	4	SS	∞
3	8	SS	∞
4	14	SS	∞
5	1	SS	7
6	2	SS	7
7	4	SS	7
8	7.14	CA	7
9	8 (aprox.)	CA	7
10	8.75 (aprox.)	CA	4.375
11	1	SS	4.375
12	2	SS	4.375