



Universidad de Granada

Fundamentos de Redes

3º del Grado en Ingeniería
Informática



Dept. Teoría de la Señal,
Telemática y Comunicaciones

Ejercicios – Tema 3

1. Suponga que el cliente A inicia una sesión Telnet con el servidor S. Aproximadamente en el mismo instante, el cliente B inicia otra conexión con el servidor.
 - a. Proporcione los posibles números de puerto origen y destino para los segmentos enviados de A a S.
Puerto origen $\in [1024,65535]$
Puerto destino=23,
 - b. Proporcione los posibles números de puerto origen y destino para los segmentos enviados de S a B.
Puerto origen=23,
Puerto destino $\in [1024,65535]$
 - c. Si A y B son hosts diferentes, ¿es posible que el número de puerto origen en los segmentos que van de A a S sea el mismo que en los segmentos de B a S?
Sí.
 - d. ¿Y si A y B son el mismo host?
No.
2. ¿Por qué TCP y UDP usan los números de puerto para identificar las entidades de transporte en lugar de usar el identificador del proceso?

El identificador de proceso podría usarse cuando las dos entidades de transporte que se comunican están en el mismo host. En el caso general, las entidades de transporte se encuentran en hosts diferentes, entonces no es posible conocer el identificador de proceso de la entidad contraria, y por lo tanto, no puede iniciarse la comunicación. Por ejemplo, no se puede conocer el identificador del proceso encargado de un servicio web, que además cambia cada vez que se reinicia el servidor. Para solucionar este problema, se usan los puertos bien conocidos (en el ejemplo del puerto 80)

3. Un proceso en un host 1 se le asigna el puerto p, y a un proceso en un host 2 se le asigna el puerto q. ¿Sería posible establecer 2 o más conexiones TCP simultáneas entre estos puertos?

No, únicamente podría darse una conexión. En primer lugar, al menos uno de los procesos debe iniciar la conexión mandando un paquete SYN. De acuerdo al diagrama de estados de TCP, en ese momento su puerto pasaría al estado SYN_SENT y se podría establecer la primera conexión siguiendo el saludo en tres pasos (three-way handshake) Una vez establecida, el puerto del iniciador no aceptaría nuevas conexiones. La única posibilidad alternativa sería el envío simultáneo de mensajes SYN por parte de los procesos, que se resolvería en una sola conexión, como se discute en el ejercicio 6.



Universidad de Granada

Fundamentos de Redes

3º del Grado en Ingeniería
Informática



Dept. Teoría de la Señal,
Telemática y Comunicaciones

4. Además del campo de 32 bits para el acuse de recibo en los segmentos TCP hay un bit o flag ACK. ¿Realmente este bit sirve para algo? Justificar la respuesta.

Sí, el bit ACK sirve para diferenciar los segmentos que llevan confirmaciones de los que no las llevan. Por tanto, si el flag ACK no está activo, el campo de acuse de recibo no debe tenerse en cuenta.

5. Calcule la suma de comprobación en UDP y TCP de las siguientes palabras de 8 bits (observe que aunque UDP y TCP utilicen palabras de 16 bits, en este ejercicio se pide el mismo cálculo sobre palabras de 8 bits): 01010011, 01010100, 01110100.

La suma en complemento a uno de se realiza de dos en dos palabras, y al derivarse un acarreo se suma adicionalmente:

$$01010011 + 01010100 = 10100111$$

$$10100111 + 01110100 = 00011011 + 1(\text{acarreo}) = 00011100$$

El complemento a uno de 00011100 es 11100011.

- a. ¿Por qué UDP/TCP utilizan el complemento a uno de la suma complemento a uno, en lugar de directamente la suma en complemento a uno?

El checksum calculado es añadido al paquete. Cuando se re-calcula en el receptor, al incluir el complemento a uno del checksum del resto de las palabras, el resultado final es una palabra todo a 1s. Si ése no es el resultado, se detecta un error. Esta estrategia es más rápida computacionalmente que incluir una comparación, que sería lo necesario en el caso de no usar el complemento a 1.

- b. ¿cómo detecta el receptor los errores?

Si obtiene algún 0 en el checksum del receptor.

- c. ¿se detectan todos los errores de 1 bit?

Sí, todos los errores de 1 sólo bit generarían un 0 en el checksum del receptor, por lo que serían detectados.

- d. ¿se detectan todos los errores que afectan simultáneamente a 2 bits?

No todos. Por ejemplo, si variamos el mismo bit en la 1ª y 2ª palabras, coincidiendo que estos bits tienen valores distintos, este error no se detecta ya que su suma es la misma.

Ejemplo:

$$01010011 + 01010100 + 01110100 = 00011100$$

$$01010010 + 01010101 + 01110100 = 00011100 \text{ ¡Error en dos bits no detectado!}$$



Universidad de Granada

Fundamentos de Redes

3º del Grado en Ingeniería
Informática



Dept. Teoría de la Señal,
Telemática y Comunicaciones

6. Suponga un establecimiento de conexión TCP en el que ambos host mandan un mensaje SYN casi simultáneo. ¿Sería posible un bloqueo del sistema? Justificar la respuesta con un ejemplo.

No sería posible. El Establecimiento de Conexión TCP está diseñado también para el caso en el que los dos host realizan una apertura activa. En este caso, ambos hosts pasarían iterativamente por los estados:

CLOSED→SYN_SENT→SYN_RCVD→ESTABLISHED.

En todo caso, ante la incidencia de una posible desincronización de la conexión debido a la latencia variable que experimentan los paquetes en Internet, se mandarían paquetes RESET, para reiniciar la conexión.

7. Describir dos formas de llegar al estado *SYN RCVD* para TCP.

CLOSED→LISTEN→SYN_RCVD
CLOSED→SYN_SENT→SYN_RCVD

8. Comparando UDP con TCP:

- a. ¿Con qué protocolo de transporte tiene una aplicación más control sobre qué datos se envían en un segmento/datagrama?

A nivel general en UDP, ya que al no implementar un servicio de conexión fiable le da mayor flexibilidad a la capa de aplicación para decidir el envío de los datos.

- b. ¿Con qué protocolo de transporte tiene una aplicación más control sobre cuándo se envía un segmento/datagrama?

Igual que a)

9. Se desea transferir con protocolo TCP un archivo de L bytes usando un MSS de 536.

- a. ¿Cuál es el valor máximo de L tal que los números de secuencia de TCP no se agoten?

Los números de secuencia, y por tanto de acuse de recibo, se agotan tras 2^{32} bytes, ya que cada número está asociado a un byte de la capa de aplicación. Si consideramos además los 2 números de secuencia consumidos correspondientes al SYN inicial y al FIN final, el número máximo de L es $2^{32} - 2$ bytes.

- b. Considerando una velocidad de transmisión de 155 Mbps y un total de 66 bytes para las cabeceras de las capas de transporte, red y enlace de datos, e ignorando limitaciones debidas al control de flujo y congestión, calcule el tiempo que se tarda en transmitir el archivo en A.



Universidad de Granada

Fundamentos de Redes

3º del Grado en Ingeniería Informática



Dept. Teoría de la Señal,
Telemática y Comunicaciones

En primer lugar, es necesario calcular el número de segmentos necesarios para la transmisión del fichero completo:

$$n_{seg} = \left\lceil \frac{L}{MSS} \right\rceil seg. = 8012999 seg.$$

donde $\lceil A \rceil$ significa el número entero inmediatamente superior a A.

A partir de este número, se calcula el número de bits transmitidos, que incluye para cada segmento la parte de datos (MSS Bytes) y las cabeceras (66 B):

$$n_{bit} = \left\lceil \frac{L}{MSS} \right\rceil \times (66 + MSS) \times 8 bits$$

En este caso, se está asumiendo que todos los segmentos tienen el mismo tamaño de datos, cuando en realidad el último segmento puede ser menor. No obstante, considerando que el número de segmentos es tan grande, las implicaciones en el resultado final serían despreciables. En todo caso siempre se puede calcular el tamaño de datos del último segmento a partir de L y el número de segmentos (366 Bytes).

Por último, el tiempo de transmisión (sin incluir inicio y cierre de conexión) es:

$$t_{transmisión del fichero} = \frac{\#bits transmitidos}{velocidad de transmisión} \approx 249 sg.$$

10. Los hosts A y B se están comunicando a través de una conexión TCP y B ya ha recibido y confirmado todos los bytes hasta el byte 126. Suponga que a continuación el host A envía dos segmentos seguidos a B que contienen, respectivamente, 70 y 50 bytes de datos. El envío de A es ordenado, el número de puerto origen en dichos segmentos es 302 y el de destino el 80. El host B envía una confirmación inmediata a la recepción de cada segmento de A, sin esperar el retardo de 500 ms del estándar.

- a. **Especifique los números de secuencia de ambos segmentos.**
127 y 197
- b. **Si el primer segmento llega antes que el segundo ¿cuál es el número de acuse y los puertos origen y destino en el primer ACK que se envía?**

Confirmación del primer segmento:

- número de acuse: 197
- puerto origen: 80
- puerto destino: 302

- c. **Si el segundo segmento llega antes que el primero ¿cuál es el número de acuse y los puertos origen y destino en el primer ACK que envía?**



Universidad de Granada

Fundamentos de Redes

3º del Grado en Ingeniería Informática



Dept. Teoría de la Señal,
Telemática y Comunicaciones

Se produce una recepción fuera de orden, por lo tanto se envía una confirmación duplicada (RFC 1122, 2581) con las siguientes características:

- número de acuse=127
- puerto origen=80
- puerto destino=302

d. **Imagine que los segmentos llegan en orden pero se pierde el primer ACK.**

Si se pierde el primer ACK no pasa nada, pues TCP utiliza confirmaciones acumulativas, es decir, la recepción del ACK del segundo segmento sirve para confirmar también el primer segmento. Si llegara a expirar el temporizador del primer segmento y éste fuera retransmitido, sería descartado en el receptor que ya guarda una copia válida de este segmento.

11. Considere el diagrama de tiempo de la Figura 1 donde se ilustra el envío y recepción de tres segmentos de 100 bytes de datos desde el Host A al Host B. Especifique las confirmaciones que realizaría el Host B. Vuelva a repetir el ejercicio considerando que estos son los únicos paquetes de datos en la conexión y que A calcula ISN=40210.

Las confirmaciones se muestran en la Figura 2. La primera confirmación (ACK=101) se enviaría tras esperar 500 ms, de acuerdo al estándar. Si el tiempo entre la llegada del primer y segundo segmento fuera inferior a 500 ms, simplemente esta primera confirmación no se enviaría y el resto sería igual. La segunda confirmación sería también ACK=101, debido a la llegada de un segmento fuera de orden, y se enviaría de manera inmediata de acuerdo al estándar. La tercera confirmación, ACK=301, sería acumulativa, y se enviaría de manera inmediata de acuerdo al estándar.

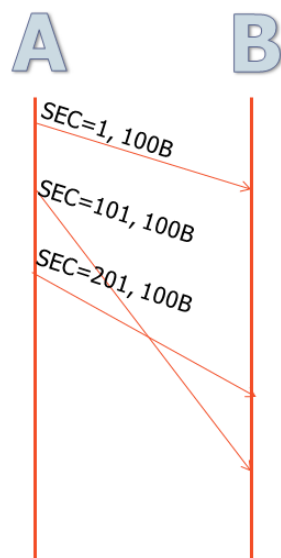


Figura 1

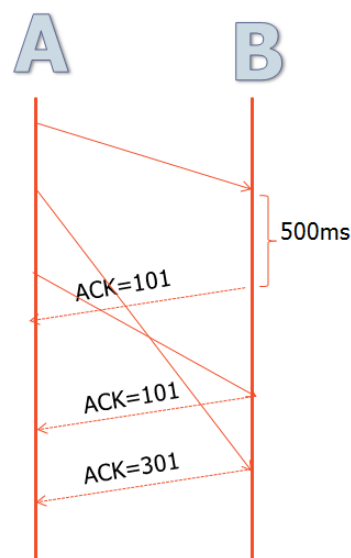


Figura 2



Universidad de Granada

Fundamentos de Redes

3º del Grado en Ingeniería
Informática



Dept. Teoría de la Señal,
Telemática y Comunicaciones

12. Los hosts A y B están directamente conectados mediante un enlace de 100 Mbps. Existe una conexión TCP entre los dos hosts y el host A está transfiriendo al host B un archivo de gran tamaño. El host A puede enviar los datos de la capa de aplicación a su socket a 120 Mbps pero el host B sólo puede leer los datos de su socket a 60 Mbps. Describa el efecto del control de flujo TCP.

El control de flujo TCP hará que el host A envíe a su socket datos a una velocidad no superior a la que se leen los datos en el socket del host B (60 Mbps).

13. En una conexión TCP, el nodo A tiene en su ventana los bytes 200, 201 y 202 enviados pero no confirmados, recibe un mensaje ACK con un valor de acuse de 202 y una ventana útil de 9. Si a continuación envía los bytes 203, 204 y 205 y si el valor de la ventana de congestión es 20.

- a. Muestre la nueva ventana en el nodo A.

La nueva ventana se muestra en la Figura3.

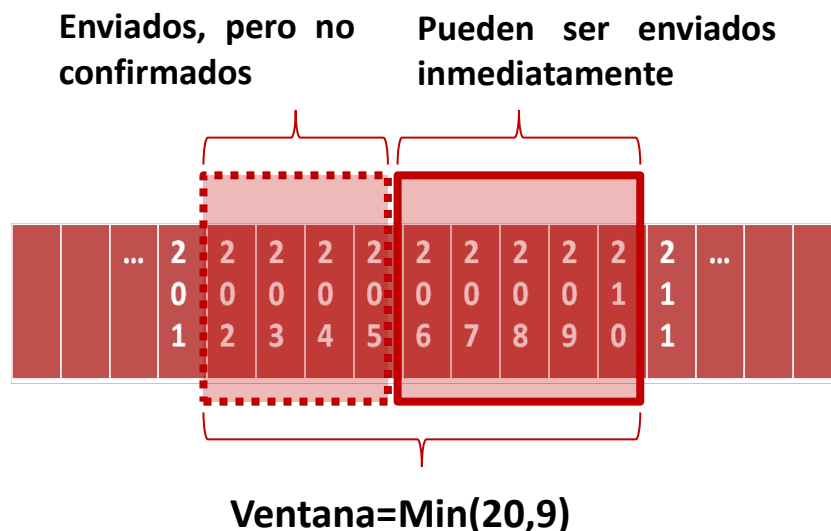


Figura 3

- b. El nodo A envía los bytes 206, 207, 208 y 209; y posteriormente recibe un paquete ACK con acuse 210 y ventana útil de 5. Si el valor de la ventana de congestión sigue siendo 20, muestre la ventana resultante.

La ventana resultante se muestra en la Figura 4.



Universidad de Granada

Fundamentos de Redes

3º del Grado en Ingeniería
Informática



Dept. Teoría de la Señal,
Telemática y Comunicaciones

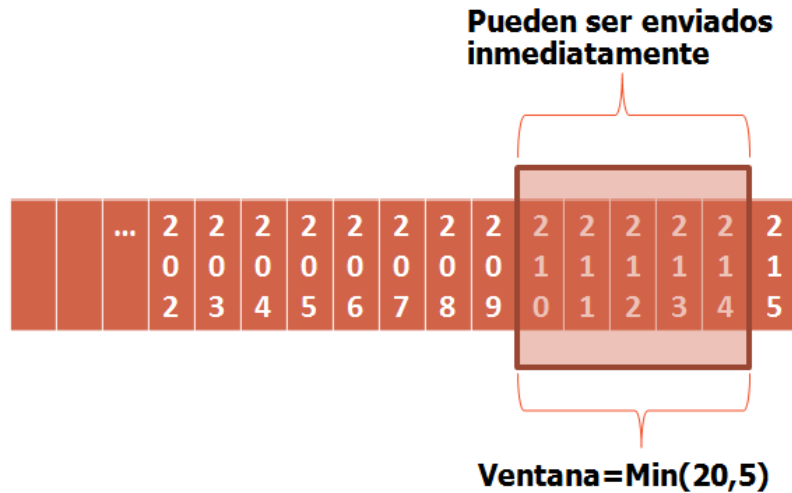


Figura 4

14. Si el RTT es 30 ms, la Desviación es 2 ms y se reciben ACKs tras 26, 32 y 24 ms, ¿Cuál será el nuevo RTT, Desviación y *timeout*? Usar $\alpha=0,125$ y $\beta=0,25$.

-Inicialmente:

$$RTT_{\text{nuevo}} = (1 - \alpha) RTT_0 + \alpha \cdot RTT_{\text{medido}}$$

$$Desv_{\text{nueva}} = (1 - \beta) Desv_0 + \beta \cdot |RTT_{\text{nuevo}} - RTT_{\text{medido}}|$$

$$RTT_0 = 30ms$$

$$Desv_0 = 2ms$$

-Tras la llegada del primer ACK:

$$RTT_1 = (1 - \alpha) * RTT_0 + \alpha * 26 = 29,5ms$$

$$Desv_1 = (1 - \beta) * Desv_0 + \beta * |RTT_1 - 26| = 2,38ms$$

-Tras la llegada del segundo ACK:

$$RTT_2 = 0,875 * RTT_1 + 0,125 * 32 = 29,81ms$$

$$Desv_2 = 0,75 * Desv_1 + 0,25 * |RTT_2 - 32| = 2,33ms$$

-Tras la llegada del tercer ACK:

$$RTT_3 = 0,875 * RTT_2 + 0,125 * 24 = 29,08ms$$

$$Desv_3 = (1 - \beta) * Desv_2 + \beta * |RTT_3 - 24| = 3,02ms$$

$$tout = RTT_3 + 4 * Desv_3 = 41,16ms$$

¿Y si los dos primeros ACKs tienen el mismo número de acuse y se usa el algoritmo de Karn?

Los dos primeros no se utilizan en el cálculo:

$$RTT_1 = 0,875 * RTT_0 + 0,125 * 24 = 29,25ms$$



Universidad de Granada

Fundamentos de Redes

3º del Grado en Ingeniería
Informática



Dept. Teoría de la Señal,
Telemática y Comunicaciones

$$Desv_1 = (1 - \beta) * Desv_0 + \beta * |RTT_1 - 24| = 2,81ms$$

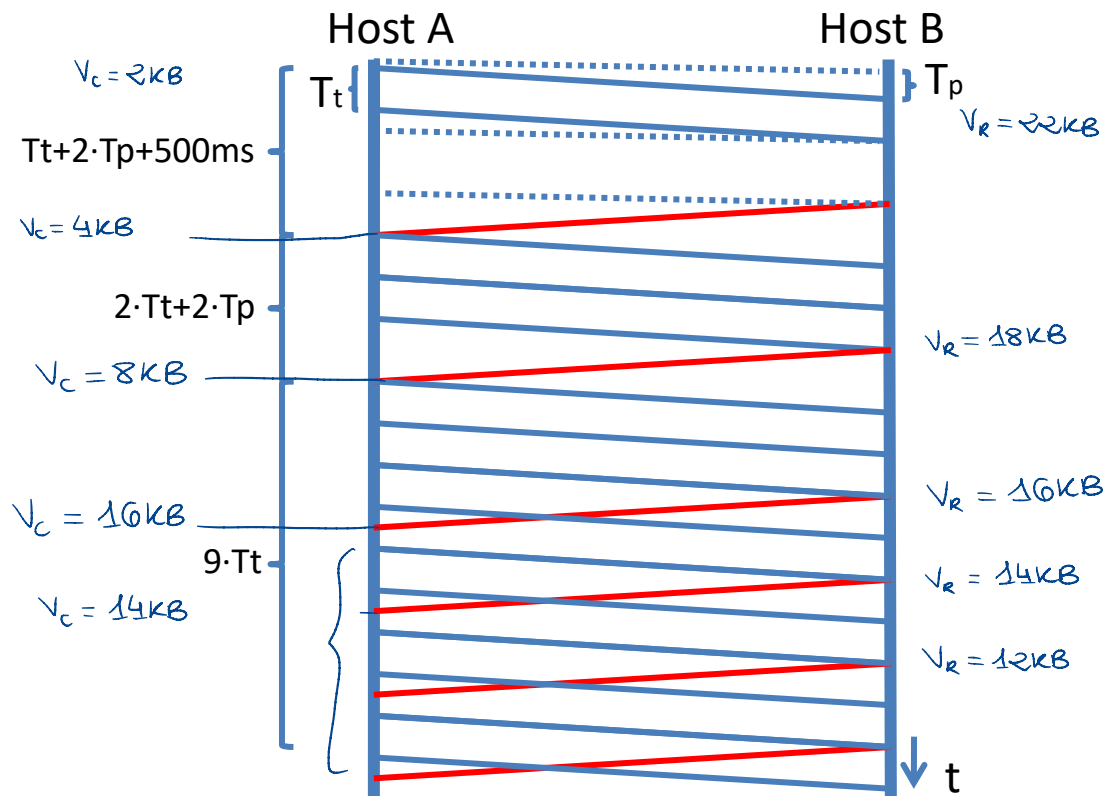
$$tout = RTT_1 + 4 * Desv_1 = 40,49ms$$

15. Teniendo en cuenta el efecto del inicio lento, en una línea sin congestión con 10 ms de tiempo de propagación, 1 Mbps de velocidad de transmisión y un MSS de 2KB, ¿cuánto tiempo se emplea en enviar 24 KB?

A partir de los parámetros, se puede calcular que:

- El número de segmentos, asumiendo siempre MSS datos, a enviar es de $24/2=12$.
- El tiempo de transmisión, despreciando cabeceras, es de $2KB/1Mbps = 16,384$ ms.
- Se desprecia el tiempo de transmisión de los paquetes ACK.

Con estos parámetros, el diagrama de tiempos es el siguiente:



Por tanto, el tiempo total en transmitir todos los datos desde el emisor es:

$$T_{tot} = 12 \cdot T_t + 4 \cdot T_p + 500ms = 736,608$$



Universidad de Granada

Fundamentos de Redes

3º del Grado en Ingeniería Informática



Dept. Teoría de la Señal,
Telemática y Comunicaciones

16. Suponiendo que la ventana de congestión es 18 KB y que se dispara un *timeout* ¿Cuánto será la ventana de congestión si las 4 siguientes ráfagas de transmisiones, donde se envía la ventana completa, son exitosas? Suponer que el MSS es 1 KB.

- i. Inicialmente, VC es de 18 KB.
- ii. Tras el timeout, VC pasa a 1KB y el umbral a 9KB.
- iii. Tras la primera ráfaga de 1KB, VC pasa a 2KB.
- iv. Tras la segunda ráfaga de 2KB, VC pasa a 4KB.
- v. Tras la tercera ráfaga de 4KB, VC pasa a 8KB.
- vi. Tras la cuarta ráfaga de 8KB, VC pasa a 9KB. Esto ocurre de la siguiente forma. Tras el ACK del primer 1KB, VC ya pasa a 9KB y a partir de ese momento, por cada ventana enviada completa, aumentará un MSS. Es decir, tras recibir los siguientes 9KB, pasará a 10KB, pero la cuarta ráfaga sólo llega a 8KB.

17. Dadas dos entidades TCP (A y B) conectadas por una red cuya velocidad de transmisión es 100 Mbps, suponga segmentos de 1024 bytes y un RTT (*Round Trip Time*) constante de 2 mseg. Si A transmite masivamente datos a B ¿Cuánto tiempo tardará en transmitir completamente 8 tramas? Incluya el número de secuencia y de acuse en todos los segmentos TCP. Haga las suposiciones que estime necesarias.

$L \rightarrow$ Segmento TCP \rightarrow 1024 bytes
(+20 bytes de cabecera IP + X bytes de cabeceras de otras capas inferiores)

$V_t = 100 \text{ Mbps}$

$RTT = 2 \text{ ms}$

Tiempo de transmisión = L/V_t

SUPOSICIONES:

- Tiempo ida = Tiempo vuelta ($RTT/2$)
- Tiempo de generación/procesamiento de tramas/ACK ≈ 0
- Consideramos los tamaños de cabecera de TCP=20 bytes e IP=20 bytes.
- Se desprecia el tamaño de las cabeceras y colas de las capas inferiores a IP.
- Consideramos que $RTT = T_{\text{TRANSMISIÓN}} + 2 \times T_{\text{PROPAGACIÓN}}$
- Sólo se considera como cabecera adicional la de la capa de red (cab. IP) ya que aún no conocen la capa de enlace.
- Hay que considerar **el tiempo de establecimiento de conexión de TCP**
- Hay que considerar **control de congestión de TCP**
- Supondremos un tamaño de ventana de congestión inicial de 2, ya que será más eficiente (no habría que esperar los 500ms para confirmar el primer segmento en caso de que el tamaño de ventana de congestión fuese 1).
- **Control de flujo:** La ventana ofertada por el receptor será suficiente para recibir todos los segmentos en cada momento.



Universidad de Granada

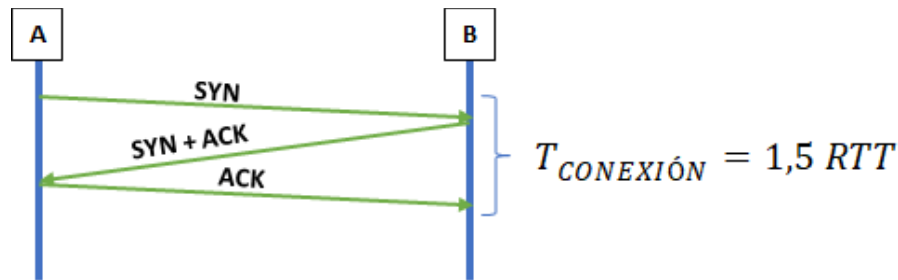
Fundamentos de Redes

3º del Grado en Ingeniería Informática



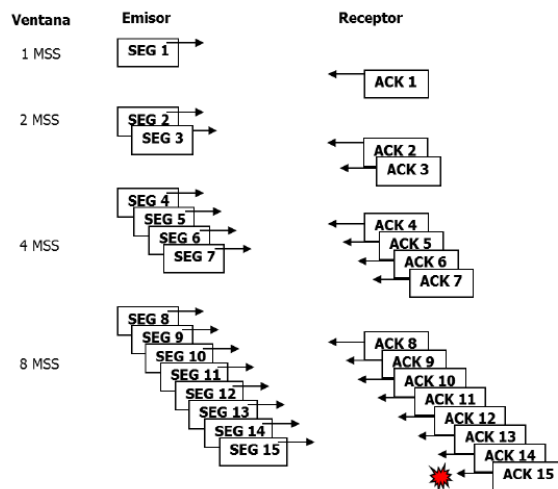
Dept. Teoría de la Señal,
Telemática y Comunicaciones

Establecimiento de conexión:



Envío de mensajes considerando el **inicio lento (slow start)** del control de congestión de TCP:

Inicio lento:



*** Inicialmente la ventana de congestión tiene el tamaño de un MSS (Maximum Segment Size) ***

*** Por cada ACK recibido (por cada segmento enviado con éxito) la ventana se amplía en un MSS ***

Según el protocolo de generación de ACKs:

- El receptor se espera a recibir el siguiente segmento 500ms, si no lo recibe envía el ACK.
- Si llega un segmento ordenado y ya había otro sin confirmar, se genera un ACK acumulado para los dos.



Universidad de Granada

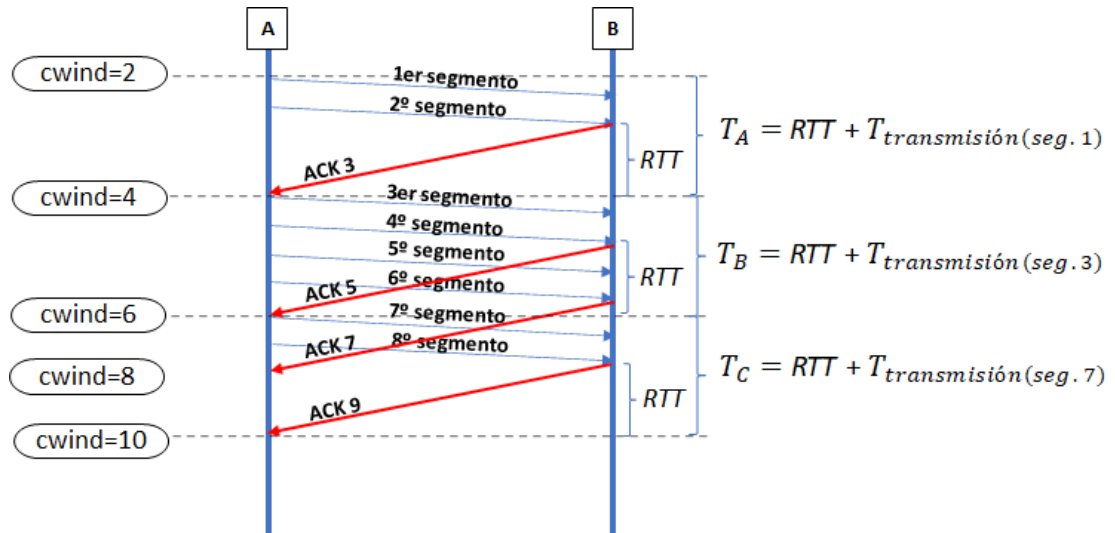
Fundamentos de Redes

3º del Grado en Ingeniería Informática



Dept. Teoría de la Señal,
Telemática y Comunicaciones

**** cwind se refiere al tamaño de la ventana de congestión ****



**** 20 → cabecera IP, 40 → cabeceras IP + TCP ****

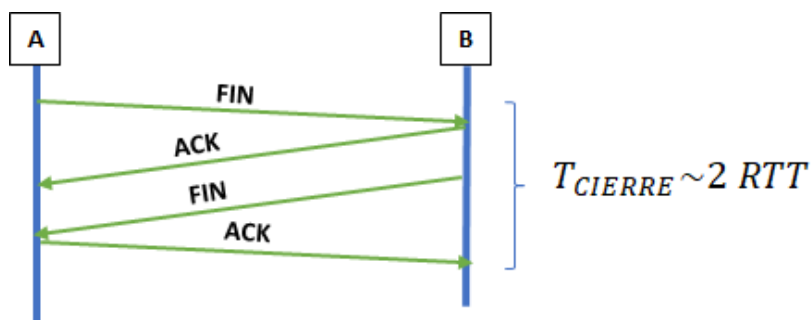
$$T_A = T_B = T_C = T_{ida} + 8 \cdot (1024 + 20) 100 \cdot 106 + T_{vuelta} = RTT + 8 \cdot (1024 + 20) 100 \cdot 106 = 2 \text{ ms} + 83,52 \text{ } \mu\text{s} = 2,1 \text{ ms}$$

$$T_{TOTAL} = T_{CONEXIÓN} + T_A + T_B + T_C = 3,0096 \text{ ms} + 2,1 \text{ ms} + 2,1 \text{ ms} + 2,1 \text{ ms} = 9,4 \text{ ms}$$

- A esto se le podría sumar el tiempo de generación de los segmentos.
- También el tiempo de generación y procesamiento de ACKs. Además, en cada suma de tamaños de cabeceras, habría que considerar el tamaño de las cabeceras de capas inferiores.

También podemos considerar que el envío de los 8 segmentos finaliza cuando se han enviado correctamente los 8 segmentos y se cierra la conexión.

Por tanto, se puede sumar al tiempo total el tiempo de cierre de la conexión TCP. Este cierre puede ser igual a la conexión (FIN, FIN-ACK, ACK) o, si consideramos que las entidades tienen que terminar envíos antes de hacer el cierre, se podría calcular como:



$$T_{TOTAL} = T_{CONEXIÓN} + T_A + T_B + T_C + T_{CIERRE} = 3 \text{ ms} + 2,1 \text{ ms} + 2,1 \text{ ms} + 2,1 \text{ ms} + 4 \text{ ms} = 13,3 \text{ ms}$$



Universidad de Granada

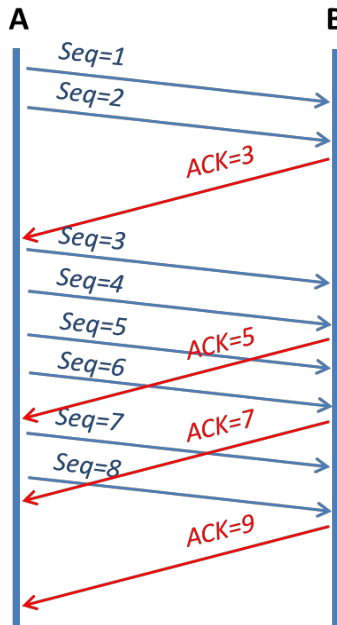
Fundamentos de Redes

3º del Grado en Ingeniería Informática



Dept. Teoría de la Señal,
Telemática y Comunicaciones

Números de secuencia y acuse (numeración simple):



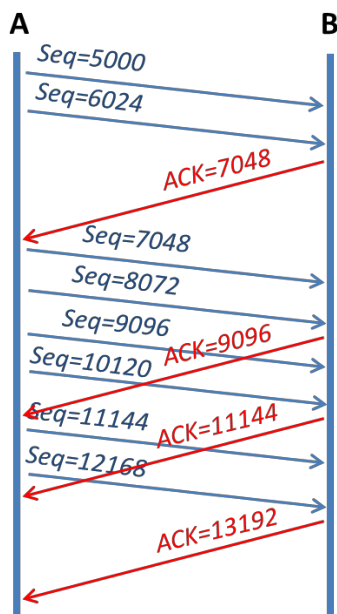
Usando una numeración simple para los números de secuencia y considerando que el receptor B no enviaría datos a A (sólo confirmaciones), el intercambio de segmentos y acuses sería el mostrado en la figura.

(Piense en un servicio de subida de archivos, por ejemplo, en el que el servidor B únicamente confirma que se han subido correctamente los datos)

Debemos recordar que se trata de confirmaciones acumulativas, es decir, un acuse confirma varios segmentos recibidos.

El acuse se referirá al número de secuencia del siguiente segmento a recibir.

Números de secuencia y acuse (numeración realista):



Consideramos que el ISN es actualmente 5000, por lo que el primer segmento se asocia a ese número.

Cada segmento tiene un tamaño de 1024 bytes, según el enunciado.

Consideramos nuevamente que el receptor B no enviaría datos a A (sólo confirmaciones).

Se trata de confirmaciones acumulativas.

El acuse se referirá al número de secuencia (byte de comienzo) del siguiente segmento a recibir.

** Hay que destacar que, tanto este, como el diagrama anterior no son precisos en cuanto a la representación de los tiempos, ya que, por ejemplo, el tiempo de transmisión del ACK será en realidad menor que el del envío de los segmentos de datos. Se ha hecho de esta forma para mayor claridad. **



Universidad de Granada

Fundamentos de Redes

3º del Grado en Ingeniería
Informática



Dept. Teoría de la Señal,
Telemática y Comunicaciones

- 17. Suponga una conexión TCP entre dos entidades ¿Qué ocurre en las dos entidades al detectarse una pérdida?**
- 18. Suponga el envío de un fichero grande sobre una conexión TCP que tiene un RTT constante. La velocidad de transmisión puede suponerse muy elevada.**
- a) Si CongWin es inicialmente 2 MSS, ¿cuánto tiempo como mínimo se necesitará para que CongWin sea 10 MSS? Suponga que no entra en la zona de prevención de congestión.
 - b) ¿Cuál será el throughput medio tras 5 RTTs?
 - c) Si CongWin es 50 MSS y está en la zona de prevención de congestión, ¿cuánto tardará CongWin en llegar a 58 MSS? ¿Cuál será el throughput medio en ese momento?