



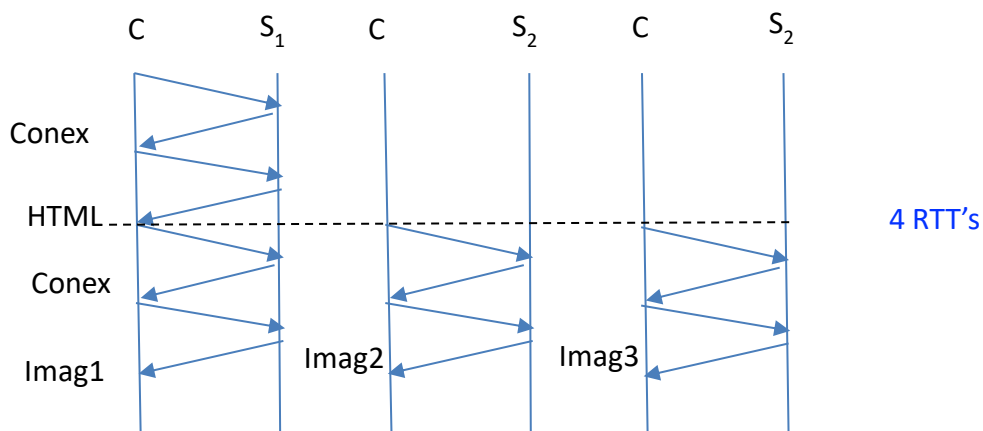
APELLIDOS: \_\_\_\_\_ NOMBRE: \_\_\_\_\_

1. ¿Cuántos RTT's son necesarios para obtener una página web consistente en un documento HTML y 3 imágenes incrustadas (dos de ellas ubicadas en otro servidor) en cada uno de los siguientes casos?
  - a. **(0,5 puntos)** Empleando conexiones no persistentes con un máximo de 3 conexiones simultáneas (concurrentes) por servidor.
  - b. **(0,5 puntos)** Empleando conexiones persistentes con *pipelining* y sin conexiones simultáneas (concurrentes) al mismo servidor.
  - c. **(0,5 puntos)** Empleando conexiones persistentes sin *pipelining* con un máximo de 3 conexiones simultáneas (concurrentes) por servidor.

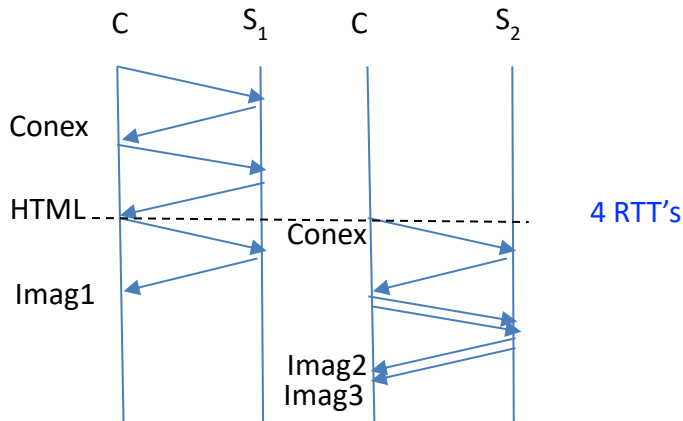
**Nota:** justifica brevemente tu respuesta.

Las respuestas sin justificación, gráfica o de texto, no se tendrán en cuenta.

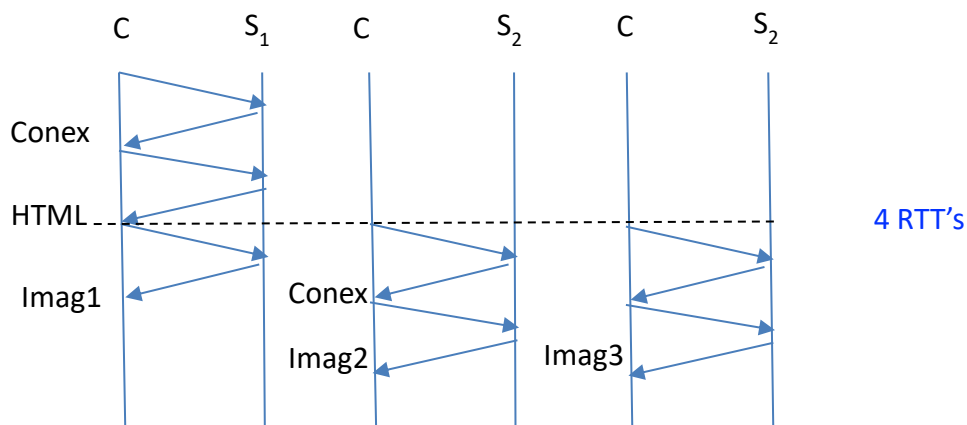
a) 4 RTT's



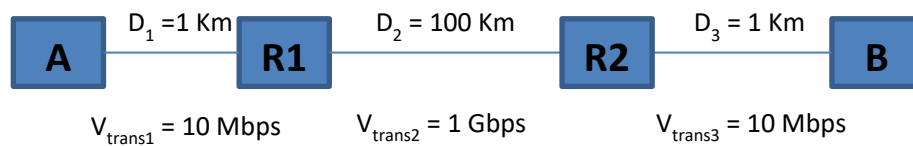
b) 4 RTT's



c) 4 RTT's

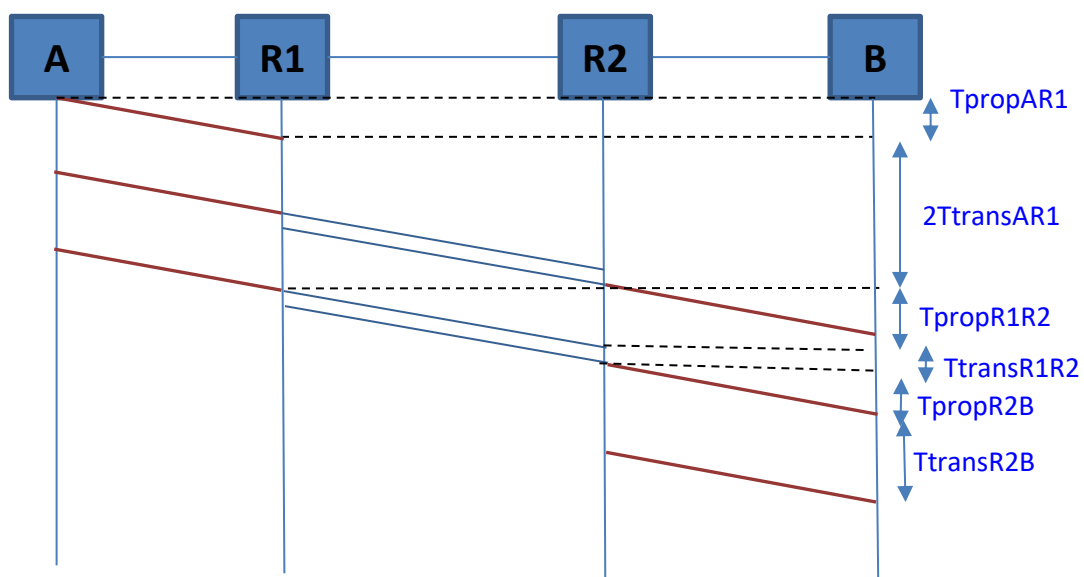


2. El host **A** se dispone a enviar un mensaje de 4000 bytes al host **B** usando la red de conmutación de paquetes del dibujo. El tamaño máximo de paquete es de 2000 bytes. Se supone que los tiempos de procesamiento en los hosts y routers son despreciables. También se consideran despreciables las cabeceras del paquete. La velocidad de propagación en todos los enlaces es de  $2 \times 10^8$  m/s.



- a. **(0,75 puntos)** Representa en un esquema gráfico espacio-temporal el tránsito de los paquetes por la red desde **A** hasta **B**. Indica en el mismo qué tiempo corresponde al  $T_{trans}$  y al  $T_{prop}$  en cada enlace.
- b. **(0,75 puntos)** Calcula el tiempo total transcurrido desde el comienzo de la transmisión en **A** hasta que el mensaje se encuentra disponible en **B**.

**Nota:** Justifica adecuadamente los cálculos realizados y expresa el resultado en milisegundos.



$$T_{propAR1} = T_{propR2B} = D_1 / V_{prop} = 10^3 / 2 \times 10^8 = 0,5 \times 10^{-5} \text{ s} = 0,005 \text{ ms}$$

$$T_{propR1R2} = D_2 / V_{prop} = 100 T_{propAR1} = 0,5 \text{ ms}$$

$$T_{transAR1} = T_{transR1B} = 8 \times 2000 / V_{trans1} = 16000 / 10^7 = 16 \times 10^{-4} \text{ s} = 1,6 \text{ ms}$$

$$T_{transR1R2} = T_{transAR1} / 100 = 0,016 \text{ ms}$$

Otra forma de calcularlo:

$$T_{transR1R2} = 8 \times 2000 / V_{trans2} = 16000 / 10^9 = 16 \times 10^{-6} \text{ s} = 0,016 \text{ ms}$$

$$T_{total} = T_{propAR1} + 2T_{transAR1} + T_{propR1R2} + T_{transR1R2} + T_{propR2B} + T_{transR2B}$$

Teniendo en cuenta que en los enlaces AR1 y R2B los valores coinciden:

$$T_{total} = 2T_{propAR1} + 3T_{transAR1} + T_{propR1R2} + T_{transR1R2}$$

$$T_{total} = (2 \times 0,005 + 3 \times 1,6 + 0,5 + 0,016) \text{ ms} = 5,326 \text{ ms}$$



3. Al hacer una consulta a nuestro servidor DNS local de la UPV obtenemos la siguiente información de respuesta:

```
Trying "gmail.com"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52140
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 0, ADDITIONAL: 10

;; QUESTION SECTION:
;gmail.com.      IN      MX

;; ANSWER SECTION:
gmail.com.      836     IN      MX      5 gmail-smtp-in.l.google.com.
gmail.com.      836     IN      MX      40 alt4.gmail-smtp-in.l.google.com.
gmail.com.      836     IN      MX      10 alt1.gmail-smtp-in.l.google.com.

;; ADDITIONAL SECTION:
gmail-smtp-in.l.google.com. 142     IN      A       74.125.133.27
gmail-smtp-in.l.google.com. 142     IN      AAAA    2a00:1450:400c:c07::1b
alt4.gmail-smtp-in.l.google.com. 221    IN      A       74.125.195.26
alt4.gmail-smtp-in.l.google.com. 221    IN      AAAA    2607:f8b0:400e:c09::1a
alt1.gmail-smtp-in.l.google.com. 174    IN      A       209.85.233.27
alt1.gmail-smtp-in.l.google.com. 174    IN      AAAA    2a00:1450:4010:c03::1a
```

- a. **(0.5 puntos)** ¿Qué tipo de contenido se almacena en los registros de tipo MX? Indica que representan gmail.com y gmail-smtp-in.l.google.com en la primera línea de la sección de respuestas.

El campo *nombre* del registro contiene un dominio de correo (gmail.com en este ejercicio) y el campo *valor* contiene el nombre de dominio del servidor SMTP que atiende (acepta mensajes) ese dominio de correo. En nuestro ejemplo, gmail-smtp-in.l.google.com es el contenido del campo valor del registro, por tanto, es uno de los servidores SMTP del dominio gmail.com.

- b. **(0.5 puntos)** ANSWER SECTION: ¿Por qué hemos obtenido 3 líneas en esta sección? ¿Qué significa  
c. el quinto campo (con valores 5, 40 y 10) de cada una de las líneas?

Significa que el dominio de correo "gmail.com" está atendido por, al menos, 3 servidores de correo (SMTP) distintos. Los valores 5, 40 y 10 indican el orden con el que deberían intentar utilizarse los servidores. A menor valor mayor prioridad. Por tanto, debe intentar utilizarse primero el de valor 5, si este no respondiese se intentaría el siguiente (valor 10) y así sucesivamente.

- d. **(0.5 puntos)** ¿Por qué obtenemos en la sección de información adicional registros de tipo A y de tipo AAAA para el mismo nombre de dominio?

Porque cada uno de los servidores de correo indicados tiene asociadas dos direcciones IP, una IP v4 (registro de tipo A) y otra IP v6 (registro de tipo AAAA).

4. **(1 punto)** ¿Qué utilidad tiene la cabecera HTTP “If-modified-since”? **Justifica** las respuestas que puede enviar el servidor (suponiendo que no devuelve un mensaje de error).

Sirve para comprobar si un objeto almacenado en la caché (del cliente http inicial o de un servidor proxy) sigue siendo válido o, por el contrario, se ha modificado en el servidor.

Si el objeto ha sido modificado, el servidor devolverá una respuesta completa incluyendo en el cuerpo del mensaje el objeto solicitado (código de respuesta “HTTP/1.X 200 OK”). Por el contrario, si la copia en caché aún es válida, la respuesta del servidor no tendrá cuerpo, y el servidor no enviará el objeto solicitado. (Código de respuesta en este caso: “HTTP/1.X 304 Not Modified”).

5. **(1,5 puntos)** Tras el establecimiento de la conexión TCP, A comienza enviando a B los segmentos TCP indicados en la tabla adjunta. Cada segmento enviado por A incluye 1000 bytes de datos. Suponemos que B emplea **reconocimientos retrasados** y que está disponible el uso de reconocimientos selectivos (SACK). El **NSI(B)** al inicio de la conexión era **0**.

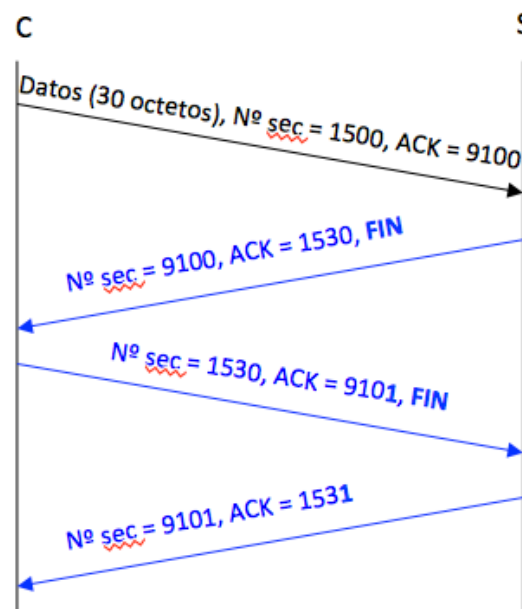
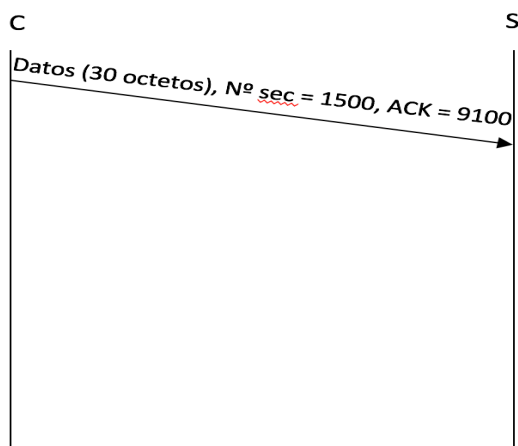
Completa la tabla indicando los **segmentos de reconocimiento que envía B ( que no llevan datos)**. Para cada segmento enviado indica su número de secuencia, nº de ACK y en el caso de SACK los valores habituales que utiliza la opción. Justificaciones: si B no envía segmento de reconocimiento indica la causa. En caso de que lo envíe justifica el valor del nº de secuencia y del ACK del segmento.

(Solución en azul en la tabla original del enunciado).

En respuesta al segmento de nº de secuencia	Nº de secuencia	Nº ACK	Nº SACK	Breve justificación del valor del nº de secuencia y del ACK del segmento
3000 (1er. segm. de datos)	-	-	-	No se reconoce este segmento porque se están utilizando ACKs retrasados. Se reconocerá el siguiente segmento de datos.
4000	1	5000	-	NSec: El establecimiento de la conexión consumió un número de secuencia. Luego se ha pasado del 0 al 1. ACK: expresa el NSec del siguiente segmento de datos que B está esperando.
5000 (no se recibe en B)	-	-	-	-
6000	1	5000	6000-7000	NSec: no se incrementa pq B no envía datos. ACK: los reconocimientos son globales y el segmento 5000 sigue pendiente (no puede reconocerse ningún segmento posterior).
7000	1	5000	6000-8000	Misma justificación que el apartado anterior. Además, estos ACKs no se retrasan, se generan cada vez que se recibe un segmento fuera de orden.



6. **(0,5 puntos)** Tras recibir el segmento de datos mostrado en la figura (de la izquierda), el servidor S inicia el cierre de la conexión TCP. Suponiendo que no hay errores de transmisión y que ninguno de los dos extremos envía ya más datos, completa la secuencia de segmentos TCP hasta que la conexión quede cerrada en los dos extremos. Indica para cada segmento: flags activados, valor del campo ACK (sólo si el flag ACK está activado) y número de secuencia del segmento. *(Solución en azul en la gráfica de la derecha).*



7. **(1,5 puntos)** Tras establecer una conexión entre un proceso en el host A y otro en el host B, la tabla refleja la evolución de **la ventana de recepción TCP de B** en cada RTT. Suponiendo que A tiene infinitos segmentos para enviar, que en el RTT=6 recibe 3 ACK's duplicados y que en el RTT=11 se produce un *TimeOut* (eventos que se detectan al final del RTT, y por tanto afectan al siguiente RTT), completa la tabla siguiente indicando los valores solicitados al comienzo de cada RTT. No se producen otros errores ni se utilizan reconocimientos retardados. Excepto los RTT, las restantes variables se expresan en segmentos.

RTT	1	2	3	4	5	6	7	8	9	10	11	12	13	14
V_rec(B)	32	30	30	32	30	30	28	20	0	20	28	30	32	32
Umbral (A)	32													
V_cong(A)	2													
V_trans(A)														

**Solución:**

RTT	1	2	3	4	5	6	7	8	9	10	11	12	13	14
V_rec(B)	32	30	30	32	30	30	28	20	0	20	28	30	32	32
Umbral (A)	32	32	32	32	32	32	15	15	15	15	15	9	9	9
V_cong(A)	2	4	8	16	32	33	15	16	17	17	18	1	2	4
V_trans(A)	2	4	8	16	30	30	15	16	0	17	18	1	2	4

8. **(1 punto)** ¿Qué es una función hash criptográfica? Indica las principales propiedades que debería tener una función de este tipo.

Una función hash  $H()$  toma como entrada un mensaje de longitud arbitrario  $m$  y obtiene un resumen del mensaje dando como salida una cadena de longitud fija:  $H(m)$ .

$H()$  es una función de “muchos a 1” (múltiples entradas pueden dar la misma salida).

Propiedades:

- Salida de longitud fija.
- Fácil de calcular (computacionalmente).
- Irreversible: no se puede hallar  $m$  a partir de  $H(m)$ .
- Resistencia a la colisión: hace referencia a que la probabilidad de que dos conjuntos de datos diferentes generen el mismo código hash ha de ser altamente improbable. No debe ser posible (computacionalmente) encontrar dos mensajes que generen el mismo resumen.