

TEST

1. Los protocolos de la capa de transporte no hacen uso de los servicios que la capa de red les proporciona.
V, F.
2. TCP calcula el temporizador basándose exclusivamente en un valor muestreado y otro estimado previamente.
V, F.
3. La detección de errores en UDP no es necesaria, pues ya lo realizan las capas inferiores.
V, F.
4. En los protocolos de *parada y espera*, la posible existencia de paquetes duplicados obliga a la introducción de números de secuencia.
V, F.
5. Si en una conexión TCP, el número de secuencia de un segmento que viaja de *A* a *B* es *N*, entonces la del siguiente segmento será necesariamente *N + 1*.
V, F.
6. En la cabecera de un segmento TCP se incluye tamaño de la ventana de recepción.
V, F.
7. En el control de congestión de TCP, si el temporizador expira, el valor de la variable *ssthreshold* pasa a la mitad.
V, F.
8. En todas las versiones de TCP, cuando el valor de la ventana de congestión llega a *ssthreshold*, comienza la fase de *evitación de la congestión*.
V, F.
9. El diseño de la capa de transporte tiene un impacto clave sobre la velocidad de transferencia de la red.
V, F.
10. Los protocolos de transporte se ejecutan en los nodos terminales y en los routers.
V, F.
11. El control de congestión en una conexión TCP se lleva a cabo por el emisor ajustando la velocidad de emisión.
V, F.
12. Los protocolos de *parada y espera* pueden emplear reconocimientos positivos y negativos.
V, F.
13. En el protocolo de reconocimiento selectivo, un evento de expiración provoca el re-envío de todos los paquetes no reconocidos.
V, F.
14. Las capas inferiores comunican a la capa de transporte la situación de congestión en la red.
V, F.
15. No existe un campo en la cabecera de un segmento TCP que indique la cantidad de datos transportados por ese segmento.
0V, F.
16. Cuando TCP devuelve un reconocimiento duplicado, ello indica que le había llegado un segmento desordenado.
V, F.
17. Todos los protocolos de transmisión fiable exigen números de identificación para los paquetes:
V, F.
18. TCP es una conexión bidireccional, pero los segmentos de reconocimiento no pueden transportar datos:
V, F.
19. En TCP, el tamaño de la ventana de recepción puede llegar a valer 0:
V, F.

20. El protocolo GBN controla el flujo de datos hacia el receptor:
V, F.
21. Si un cliente intenta iniciar una conexión TCP con un servidor y recibe un segmento de tipo RST puede significar que el nodo servidor está caído:
V, F.
22. El control de flujo está regulado por el emisor y el control de congestión está regulado por el receptor:
V, F.
23. Si el valor de *RTT* que TCP muestrea al recibir un reconocimiento vale 1 segundo, el valor de *RTT* estimado será necesariamente mayor que 1 segundo:
V, F.
24. En la fase de establecimiento de la conexión, se acuerdan los números de secuencia inicial tanto en emisor como en receptor:
V, F.
25. Dos segmentos UDP que tengan como destino el mismo host y el mismo puerto dentro de ese host necesariamente provocan error en el proceso destino:
V, F.
26. El bit PUSH puede ser controlado por el programador de la aplicación:
V, F.

1. Indique los valores por los que se identifica un socket TCP:

- a) Dirección IP origen, puerto origen, dirección IP destino, puerto destino.
- b) Dirección IP origen, dirección IP destino.
- c) Puerto origen, puerto destino.
- d) Dirección IP destino, puerto destino.

2. Indique cuales de los siguientes campos pertenecen a una cabecera UDP:

- a) Puerto origen, puerto destino, flag SYN, CRC.
- b) Puerto origen, puerto destino, longitud, CRC.**
- c) Puerto origen, puerto destino, protocolo aplicación, CRC.
- d) Ninguna de las respuestas anteriores es correcta.

3 Sobre los protocolos de bit alternante:

- a) Reciben su nombre del hecho de emplear un bit aleatorio como número de secuencia.
- b) Emplean checksum, reconocimientos, números de secuencia y temporizadores.**
- c) Emplean checksum, reconocimientos y temporizadores, pero no números de secuencia.
- d) Emplean checksum, reconocimientos y números de secuencia, pero no temporizadores.

4. El protocolo de parada y espera es más ineficiente conforme:

- a) Aumentan las ventanas de emisor y receptor.
- b) Disminuyen las ventanas de emisor y receptor.
- c) Aumenta el RTT.**
- d) Disminuye el RTT.

5. Sobre el valor del temporizador de retransmisión de TCP:

- a) Debe ser inferior al RTT para que funcione correctamente TCP.
- b) Emplea el valor de RTT medido para darle valor, dando mayor peso a las muestras recientes que a las antiguas.**

- c) Emplea el valor de RTT medido para darle valor, dando mayor peso a las muestras antiguas que a las recientes.
- d) No considera la desviación de los retardos medidos.

6. Sobre el algoritmo de control de congestión en TCP Vegas:

- a) **Actúa cuando estima la pérdida inminente de segmentos a partir de observar los valores de RTT.**
- b) Actúa a posteriori, una vez que se detecta un evento de pérdida, o presunta pérdida de segmento, independientemente del valor RTT.
- c) Actúa a posteriori, una vez que se detecta un evento de pérdida, o presunta pérdida de segmento, cuando se detecta que el RTT es muy bajo.
- d) Ninguna de las anteriores es correcta.

7. En el modo de recuperación rápida en TCP:

- a) **A la recepción del ACK del segmento que falta, pasa al modo "evitación de la congestión".**
- b) A la recepción del ACK del segmento que falta, pasa al modo "arranque lento".
- c) A la recepción de un nuevo ACK pasa al modo "arranque lento".
- d) A la recepción de un nuevo ACK permanece en el modo "recuperación rápida".

8. Control de Congestión TCP, modo "arranque lento":

- a) **A la recepción de tres ACK's duplicados se pasa al modo "recuperación rápida", si éste está habilitado.**
- b) A la recepción de tres ACK's duplicados se pasa al modo "evitación de la congestión", si éste está habilitado.
- c) A la recepción de tres ACK's duplicados se pasa al modo "recuperación rápida", aunque no esté habilitado.
- d) Ninguna de las anteriores es correcta.

9. En la fase de establecimiento de conexión TCP:

- a) El segmento SYN inicial puede contener datos.
- b) El segmento SYNACK puede contener datos.
- c) El segmento FIN puede contener datos.
- d) **Ninguna de las anteriores es correcta.**

10. El control de flujo en TCP:

- a) **Procedimiento de adaptación de velocidades entre emisor y receptor TCP, que conlleva al control de la velocidad de emisión de segmentos en base a una variable que caracteriza al receptor TCP.**
- b) Procedimiento de adaptación de velocidades entre emisor y receptor TCP, que conlleva al control de la velocidad de emisión de segmentos en base a una variable que caracteriza al emisor TCP.
- c) No es un proceso habitual en el protocolo TCP.
- d) Ninguna de las anteriores es correcta.

11. Temporizador de retransmisión TCP (timeout):

- a) **Su valor se calcula haciendo una estimación del tiempo medio de los RTT, y la desviación de las muestras RTT a partir de este valor medio, dando mayor peso a los valores más recientes de las muestras RTT.**
- b) Su valor se calcula haciendo una estimación del tiempo máximo de los RTT, y la desviación de las muestras RTT a partir de este valor medio, dando mayor peso a los valores más recientes de las muestras RTT.
- c) Su valor se calcula haciendo una estimación del tiempo medio de los RTT, y la desviación de las muestras RTT a partir de este valor medio, dando mayor peso a los valores más antiguos de las muestras RTT.
- d) Ninguna de las anteriores es correcta.

12. Empleando un protocolo de parada y espera, suponga que el host A envía dos segmentos al host B. El host B envía ACKs para cada segmento, el primer ACK se pierde pero el segundo llega al emisor antes de que expire el temporizador del primer paquete.

- a) El host A no retransmitirá ningún paquete.
- b) El host A retransmitirá el primer paquete.
- c) No puede darse la situación descrita.**
- d) El host A retransmitirá el segundo paquete.

13. Empleando un protocolo de repetición selectiva, suponga que el host A envía dos segmentos al host B. El host B envía ACKs para cada segmento, el primer ACK se pierde pero el segundo llega al emisor antes de que expire el temporizador del primer paquete.

- a) No puede darse la situación descrita.
- b) El host A retransmitirá el primer paquete.**
- c) El host A retransmitirá ambos paquetes.
- d) El host A retransmitirá el segundo paquete.

14. Un cliente TCP, para establecer una conexión envía un segmento:

- a) Con el flag SYN activo, N° secuencia inicial y primeros datos.
- b) Con los flags SYN y ACK activos, N° secuencia inicial y sin datos.
- c) Con el flag ACK activo, N° secuencia inicial y sin datos.
- d) Con el flag SYN activo, N° secuencia inicial y sin datos.**

15. Una entidad TCP ha recibido un segmento correctamente pero no tiene datos que enviar al emisor. ¿Qué hace en ese caso?

- a) No envía un ACK hasta que no tenga datos que enviar.
- b) Espera hasta 500 ms a tener datos antes de contestar con un ACK. Si en este tiempo no los tiene y no le llega otro segmento correcto envía el ACK.**
- c) La entidad se queda bloqueada, hay que resetearla enviando un segmento con el flag RST activado.
- d) Espera un tiempo igual a 2 veces el RTT estimado, y contesta con un ACK.

16. El límite de la velocidad máxima de envío en una conexión TCP:

- a) Lo fija el receptor con la ventana de recepción, que es función de la capacidad del receptor.
- b) Lo fija con la ventana de congestión, que es función de la capacidad ruta entre ambos.
- c) Lo fija la ventana de recepción y de congestión, con el máximo de ambas.
- d) Lo fija la ventana de recepción y de congestión, con el mínimo de ambas.**

17. TCP ajusta su velocidad de emisión haciendo lo siguiente:

- a) La incrementa cuando llega un ACK que indica que no hay congestión y la decrementa cuando se pierde un segmento.**
- b) La mantiene cuando llega un ACK que indica que no hay congestión y la decrementa cuando se pierde un segmento.
- c) La mantiene cuando llega un ACK que indica que no hay congestión y la mantiene cuando se pierde dos segmentos.
- d) Ninguna de las anteriores es correcta.

18. El control de congestión en TCP:

- a) Busca un compromiso entre producir congestión por transmitir demasiado rápido e infrautilizar recursos por ser demasiado cauteloso.**
- b) Se inicia al recibir tres ACKs duplicados.
- c) El modo de arranque lento es opcional.
- d) El modo de evitación de la congestión es opcional.

19. Sobre el formato de cabecera de TCP:

- a) El campo longitud de cabecera especifica el tamaño de la cabecera medida en octetos.
- b) El campo longitud de cabecera especifica el tamaño de la cabecera medida en palabras de 16 bits.
- c) La cabecera TCP tiene un tamaño fijo.
- d) Se emplean dos octetos para especificar el puerto destino.**

20. Sobre la comparación entre TCP y UDP:

- a) TCP y UDP son protocolos orientados a la conexión.
- b) TCP y UDP incorporan detección de errores, aunque su uso en UDP es opcional.**
- c) UDP no retransmite los segmentos perdidos, pero sí los erróneos.
- d) UDP, al igual que TCP, realiza un control de flujo del tipo parada y espera.

21. Indique los valores por los que se identifica un socket UDP:

- a) Dirección IP origen, puerto origen, dirección IP destino, puerto destino.
- b) Dirección IP origen, dirección IP destino.
- c) Puerto origen, puerto destino.
- d) Dirección IP destino, puerto destino.**

22. Entre los protocolos de aplicación que utilizan UDP podemos señalar:

- a) HTTP, FTP, NTP.
- b) SNMP, SMTP, FTP.
- c) NTP, TFTP, DNS.**
- d) FTP, HTTP, MIME, TFTP.

23. La siguiente figura muestra una captura de tramas utilizando Wireshark. Sobre las tres primeras tramas podemos afirmar:

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.1.102	128.119.245.12	TCP	1161 > http [SYN] Seq=0 Len=0 MSS=1460
2	0.023172	128.119.245.12	192.168.1.102	TCP	http > 1161 [SYN, ACK] Seq=0 Ack=1 win=5840
3	0.023265	192.168.1.102	128.119.245.12	TCP	1161 > http [ACK] Seq=1 Ack=1 win=17520
4	0.026477	192.168.1.102	128.119.245.12	TCP	1161 > http [PSH, ACK] Seq=1 Ack=1 win=17520
5	0.041737	192.168.1.102	128.119.245.12	TCP	1161 > http [PSH, ACK] Seq=566 Ack=1 win=17520
6	0.053937	128.119.245.12	192.168.1.102	TCP	http > 1161 [ACK] Seq=1 Ack=566 win=6780
7	0.054026	192.168.1.102	128.119.245.12	TCP	1161 > http [ACK] Seq=2026 Ack=1 win=17520
8	0.054690	192.168.1.102	128.119.245.12	TCP	1161 > http [ACK] Seq=3486 Ack=1 win=17520
9	0.077294	128.119.245.12	192.168.1.102	TCP	http > 1161 [ACK] Seq=1 Ack=2026 win=8760
10	0.077405	192.168.1.102	128.119.245.12	TCP	1161 > http [ACK] Seq=4946 Ack=1 win=17520
11	0.078157	192.168.1.102	128.119.245.12	TCP	1161 > http [ACK] Seq=6406 Ack=1 win=17520
12	0.124085	128.119.245.12	192.168.1.102	TCP	http > 1161 [ACK] Seq=1 Ack=3486 win=11680
13	0.124185	192.168.1.102	128.119.245.12	TCP	1161 > http [PSH, ACK] Seq=7866 Ack=1 win=17520
14	0.169118	128.119.245.12	192.168.1.102	TCP	http > 1161 [ACK] Seq=1 Ack=4946 win=14600
15	0.217299	128.119.245.12	192.168.1.102	TCP	http > 1161 [ACK] Seq=1 Ack=6406 win=17520
16	0.267802	128.119.245.12	192.168.1.102	TCP	http > 1161 [ACK] Seq=1 Ack=7866 win=20440

- a) Se trata de un establecimiento de conexión TCP.**
- b) Se trata de un cierre de conexión TCP.
- c) Se trata de un intercambio de tramas entre los puertos UDP.
- d) Ninguna de las respuestas anteriores es correcta.

24. Los protocolos de transferencia de datos, para recuperarse de la pérdida de paquetes emplean:

- a) Checksum (también llamado CRC).
- b) Reconocimientos positivos y/o negativos.
- c) Temporizadores (timeouts).**
- d) Ninguna de las anteriores es correcta.

25. En el arranque lento ¿Cuál es el valor por el que se modifica la ventana de transmisión?

- a) Se le añade el valor de la ventana inicial cada vez.
- b) Se incrementa 100 octetos cada RTT.
- c) Se incrementa 1 MSS cada primer ACK de un segmento transmitido.**
- d) Se incrementa 100 bits cada RTT.

26. En el cálculo del valor de *timeout* asociado a la emisión de un segmento se tiene en cuenta:

- a) El valor del último RTT calculado para el último segmento transmitido, pero no el promedio de RTT anteriores.
- b) El valor promedio de valores RTT anteriores sin incluir el último RTT medido.
- c) El valor del último RTT calculado para el último segmento transmitido y el promedio de RTT anteriores.**
- d) Ninguna de las anteriores es correcta.

27. Los procedimientos de control de congestión en las versiones TCP Tahoe y Reno se caracterizan por:

- a) Actuar a posteriori, una vez detectada la pérdida o presunta pérdida de un segmento.**
- b) Actuar con antelación a la ocurrencia de un evento de pérdida o presunta pérdida de segmento.
- c) Actuar con antelación o a posteriori a la ocurrencia de un evento de pérdida o presunta pérdida de segmento, siempre y cuando se esté en el modo Arranque Lento.
- d) Ninguna de las anteriores es correcta.

28. En el modo "recuperación rápida" de TCP:

- a) A la recepción de un ACK duplicado la variable ventana de congestión (VC) se incrementa un MSS**
- b) A la recepción de un ACK duplicado la variable ventana de congestión (VC) se incrementa una fracción de MSS.
- c) A la recepción de un ACK duplicado la variable ventana de recepción (VR) se incrementa un MSS.
- d) A la recepción de un ACK duplicado la variable ventana de recepción (VR) se incrementa una fracción de MSS.

29. Control de Congestión TCP, modo "evitación de la congestión":

- a) A la recepción de tres ACK's duplicados se pasa al modo "recuperación rápida" si se emplea TCP Tahoe.
- b) A la recepción de dos ACK's duplicados se pasa al modo "recuperación rápida", si éste está habilitado.
- c) A la recepción de tres ACK's duplicados permanece en el modo "evitación de la congestión".
- d) Ninguna de las anteriores es correcta.**

30. Control de Congestión TCP:

- a) Cuando vence el temporizador de retransmisión se pasa al modo "arranque lento", y la variable VC se disminuye a la mitad.
- b) Cuando vence el temporizador de retransmisión se pasa al modo "arranque lento", y la variable VC se hace igual a 1 MSS.**
- c) Cuando vence el temporizador de retransmisión se pasa al modo "evitación de la congestión", y la variable VC se hace igual a 1 MSS.
- d) Ninguna de las anteriores es correcta.

31. Control de flujo TCP:

- a) Procedimiento mediante el cual receptor notifica a emisor el estado de su buffer de recepción, por intermedio de la variable "ventana del emisor".
- b) Procedimiento mediante el cual receptor notifica a emisor el estado de su buffer de recepción, por intermedio de la variable "umbral".
- c) Procedimiento mediante el cual receptor notifica a emisor el estado de su buffer de recepción, por medio de la variable "ventana de recepción".**
- d) Ninguna de las anteriores es correcta.

32. En relación con el cierre de una conexión TCP:

- a) Un cierre ordenado de la conexión en ambos sentidos sólo requiere el intercambio de dos segmentos.
- b) El cliente responde con un segmento con el bit ACK activo a los segmentos con el bit FIN activo recibidos.**
- c) El cliente cierra la conexión inmediatamente después de enviar un segmento con el bit ACK activo.
- d) Ninguno de las anteriores es correcta.

33. El algoritmo de retransmisión rápida:

- a) Permite identificar la pérdida de un segmento y retransmitirlo, pero no antes de que venza el timeout.
- b) Se activa con la llegada al emisor de tres segmentos seguidos con el mismo valor de ACK, cuando este ha seguido enviando segmentos de datos.**
- c) Permite evitar la pérdida de un segmento y retransmitirlo, pero no antes de que venza el timeout.
- d) Ninguna de las anteriores es correcta.

34. ¿En el mecanismo de control de congestión TCP?

- a) La red IP subyacente siempre informa a los terminales de la existencia de congestión de red.
- b) La red IP no informa de congestión por lo que los terminales deben deducir cuando hay congestión.**
- c) La velocidad de transmisión la fijará exclusivamente el receptor que es quien puede tener limitaciones en la capacidad de recepción.
- d) Ninguna de las anteriores es correcta.

35. Cuando hay congestión en un router y se descarta un datagrama:

- a) El receptor manda un mensaje especial al emisor para que baje la velocidad.
- b) El emisor recibirá tres ACK duplicados o le vencerá un temporizador y bajará la velocidad.**
- c) El emisor entenderá que hay congestión sólo cuando venza el temporizador asociado al datagrama perdido.
- d) El receptor entiende que hay congestión y reduce la *ventana del receptor* del emisor.

36. El arranque lento TCP termina:

- a) Sólo termina cuando vence el timeout.
- b) Sólo termina cuando se reciben tres ACK duplicados.
- c) Sólo termina cuando la ventana de congestión alcanza el valor del umbral.
- d) Cuando se da cualquiera de las tres causas anteriores.**

37. TCP es un protocolo:

- a) De tipo Retroceder N (GBN).
- b) De tipo Repetición Selectiva (SR).
- c) De tipo Parada y Espera (S&W).
- d) Ninguna de las anteriores es correcta.**

38. En cuanto a la comparación entre TCP y UDP:

- a) Tanto TCP como UDP realizan control de flujo, siendo en este último caso mediante el uso de "parada y espera".
- b) A pesar de que TCP es más complejo que UDP, ambos tienen un tamaño de cabecera de 8 bytes.
- c) UDP no realiza segmentación, pero sí reensamblado.
- d) UDP no es un protocolo orientado a conexión.**

39. Una entidad TCP retransmite un segmento (N° seq = 95; tamaño = 5 octetos) al haber expirado el timeout y mientras los transmite recibe un segmento con ACK = 100.

- a) Se trata de un caso de timeout prematuro. Volverá a recibir un segmento con ACK = 100.**
- b) Debería haber recibido un segmento con ACK = 101.
- c) No debe haber recibido ningún ACK hasta que el segmento llegue al receptor y éste conteste.
- d) Esto indica un bloqueo del protocolo.

40. Sobre el protocolo UDP:

- a) La carga útil puede tener un tamaño de hasta $(2^{16}-1)$ bytes.
- b) El número de puerto máximo es $2^{16}-1$, que debe repartirse entre UDP y TCP.
- c) El puerto origen en UDP se emplea para una posible respuesta.**
- d) Ninguna de las anteriores es correcta.

TEST KUROSE

R14. Verdadero o Falso

- a. El host A está enviando al host B un archivo de gran tamaño a través de una conexión TCP. Suponga que el host B no tiene datos que enviar al host A. El host B no enviará paquetes de reconocimiento al host A porque el host B no puede superponer esos reconocimientos sobre los datos. → **FALSO**
- b. El tamaño de la ventana de recepción de TCP nunca varía mientras dura la conexión. → **FALSO**
- c. Suponga que el host A está enviando al host B un archivo de gran tamaño a través de una conexión TCP. El nº de bytes no reconocidos que A envía no puede exceder el tamaño del buffer del receptor. → **VERDADERO**
- d. Suponga que el host A está enviando al host B un archivo de gran tamaño a través de una conexión TCP. Si el nº de secuencia de un segmento en esta conexión es m, entonces el nº de secuencia del siguiente segmento necesariamente tiene que ser m+1. → **FALSO**
- e. El segmento TCP contiene un campo en su cabecera para VentanaRecepcion. → **VERDADERO**
- f. Suponga que el último RTTMuestra en una conexión TCP es igual a 1 segundo. El valor actual del IntervaloFinDeTemporización para la conexión será necesariamente ≥ 1 segundo. → **FALSO**
- g. Suponga que el host A envía al host B un segmento con el nº de secuencia 38 y 4 bytes de datos a través de una conexión TCP. En ese mismo segmento el nº de reconocimiento necesariamente tiene que ser 42. → **FALSO**

P22. Responda verdadero o falso a las siguientes preguntas y justifique brevemente sus respuestas:

- a. Con el protocolo SR, el emisor puede recibir un ACK para un paquete que se encuentra fuera de su ventana actual. → **VERDADERO**
- b. Con GBN, el emisor puede recibir un ACK para un paquete que se encuentra fuera de su ventana actual. → **VERDADERO**
- c. El protocolo de bit alternante es igual que el protocolo SR, pero con un tamaño de ventana en el emisor y en el receptor igual a 1. → **VERDADERO**
- d. El protocolo de bit alternante es igual que el protocolo GBN, pero con un tamaño de ventana en el emisor y en el receptor igual a 1. → **VERDADERO**