

PREGUNTAS EXÁMEN TEORIA TEMAS 4 Y 5

1. **¿Cómo se recupera un extremo que usa TCP ante la recepción de tres confirmaciones (ACK) iguales?**

Reenviando el segmento TCP perdido

2. **En una red P2P totalmente distribuida, ¿cómo es el proceso de unirse a la red?**

3. **¿Qué diferencia hay entre el modo activo y pasivo de FTP?**

Activo: el cliente de FTP actúa como servidor de la conexión de datos, creando él la conexión (enviando el comando PORT al servidor)

Pasivo: el servidor de FTP también actúa como servidor en la conexión de datos, creando éste la conexión de envío de datos (el cliente envía el comando PASV al servidor)

4. **Explique por qué el RTO en TCP es dinámico y en base a qué elementos se calcula**

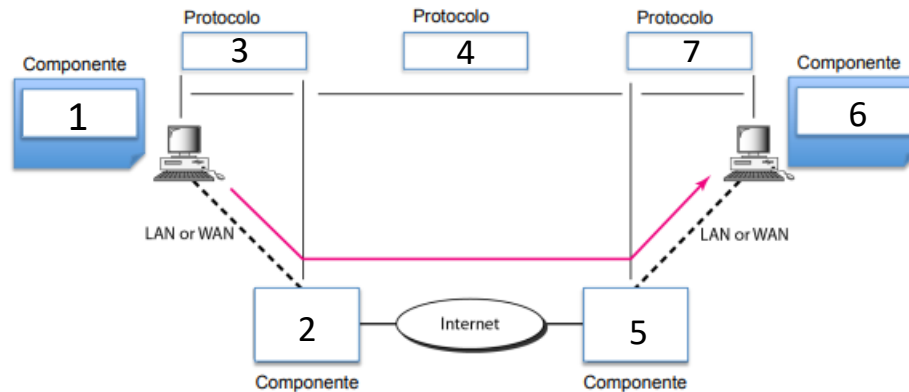
Es dinámico ya que no es fijo, cambia en el tiempo. Se actualiza en base al tiempo de ida y vuelta (RTT): usa su media (SRTT) y varianza (RTTVAR)

5. **El siguiente trozo de código, ¿utiliza TCP o UDP? y ¿corresponde a un servidor o a un cliente? Justifique la respuesta**

```
DatagramSocket s = new DatagramSocket(PUERTO);
while (true) {
    DatagramPacket recibido = new DatagramPacket(new byte [M],M);
    s.receive(recibido);
    String message=...
    msg=message.getBytes();
    DatagramPacket paquete=new DatagramPacket(msg, msg.length,
                                                recibido.getAddress(),recibido.getPort());
    s.send(paquete);
}
```

Corresponde a un servidor UDP, ya que su ejecución es un bucle infinito y directamente espera un mensaje del cliente, sin iniciar una conexión antes con el cliente ni cerrándola después.

6. Indique en la siguiente figura los componentes de una aplicación de correo y con qué protocolos se comunican



- 1.- Sender (cliente MTA con UA) 2.- Mail server (servidor MTA)
 3.- SMTP 4.- SMTP 5.- Mail server (servidor MTA y MAA)
 6.- Receiver (cliente MAA con UA) 7.- POP3 e IMAP4

7. ¿Para qué se utiliza el protocolo SIP?

Este protocolo se utiliza para el establecimiento de conexiones multimedia entre dos usuarios (muy usado en VoIP). No se encarga de transmitir los datos, ya que para eso se utiliza RTP/UDP, UDP o TCP. También permite localizar la IP de un usuario gracias a un sistema de registro.

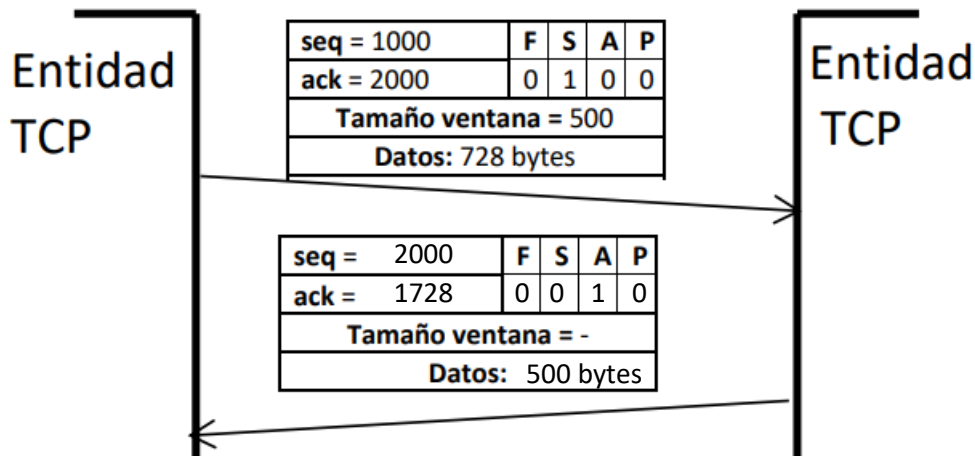
8. ¿Qué puerto utiliza cada uno de los siguientes comandos y qué diferencias hay entre ellos?

- telnet localhost -> 23/TCP. Empieza a escuchar en el puerto estándar de Telnet
- telnet localhost 80 -> 80/TCP. Se conecta a la aplicación en la propia máquina alojada en el puerto 80

9. ¿Cómo detecta UDP la pérdida de mensajes?

UDP no detecta la pérdida de mensajes

10. Complete el segmento TCP de respuesta donde va la confirmación a los datos recién recibidos junto a nuevos datos (quiere enviar 1024 bytes).



11. Sea un servidor que se comunica con tres clientes. ¿Cuántos sockets hacen falta como mínimo en el servidor si en la comunicación se usa TCP? ¿Y si se usa UDP?

Serán necesarios 3 sockets para TCP y 1 para UDP.

En TCP tendremos un socket/conexión ("variable") por cada cliente (3 en este caso).

En UDP tendremos un único socket (una cola de recepción y una cola de emisión) independientemente del número de clientes.

12. ¿Qué es y cuándo se produce la retransmisión rápida en TCP?

Cuando se reciben 3 ACKs duplicados para el mismo número de secuencia, se reenvía el segmento perdido inmediatamente.

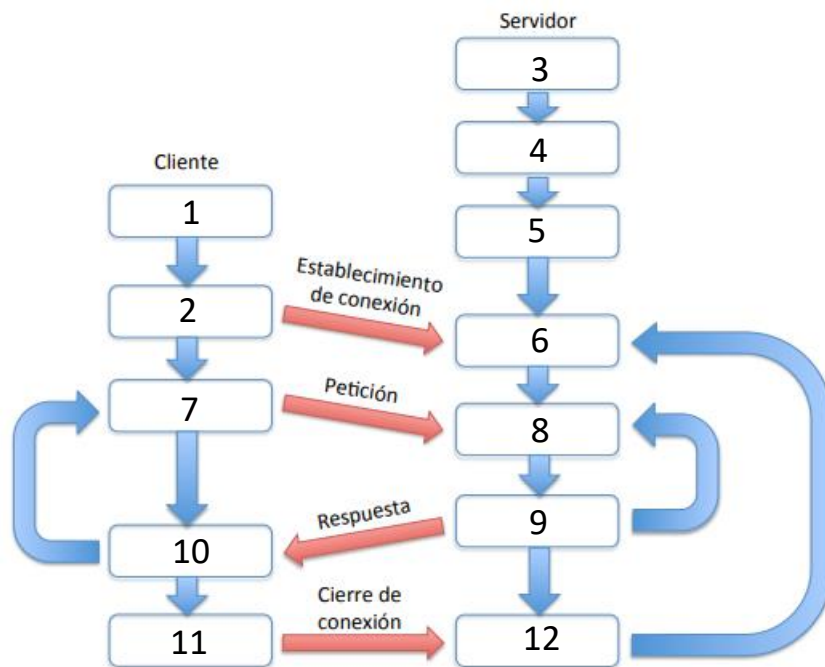
13. ¿Qué pasa cuando un datagrama UDP llega a la máquina destino y no existe una cola asociada al puerto destino?

Se descarta el datagrama y se pide a ICMP que envíe el mensaje de puerto no alcanzable al emisor

14. ¿Qué hace TCP cuando llega un segmento fuera de orden?

Se almacenan temporalmente y se marca como "fuera de orden" (al ser marcados no se entregan al proceso destino) hasta que llega el segmento perdido

15. Usa funciones de sockets en C para rellenar los recuadros del siguiente diagrama que representa una comunicación entre un cliente y un servidor TCP



```

1.- serviceSocket = new Socket(serverIP, serverPort);
2.- in = new BufferedReader(new InputStreamReader(serviceSocket.getInputStream()));
   Out = new PrintWriter(serviceSocket.getOutputStream(), true);
3.- SocketPasivo = new ServerSocket(port, 1);
4.- Innecesario
5.- Innecesario
6.- socketConectado = socketPasivo.accept();
   In = new BufferedReader(new InputStreamReader(socketConectado.getInputStream()));
   PrintWriter(socketConectado.getOutputStream, true);
7.- out.println(userInput);
8.- linea = in.readLine();
9.- out.println(linea);
10.- linea = in.readLine();
11.- in.close();    Out.close();    serviceSocket.close();
12.- in.close();    out.close();    socketConectado.close();

```

16. Indique tres mensajes de SMTP y describa su función

HELO<SP><domain><CRLF>: inicio de conexión de transmisión con el servidor especificado en domain

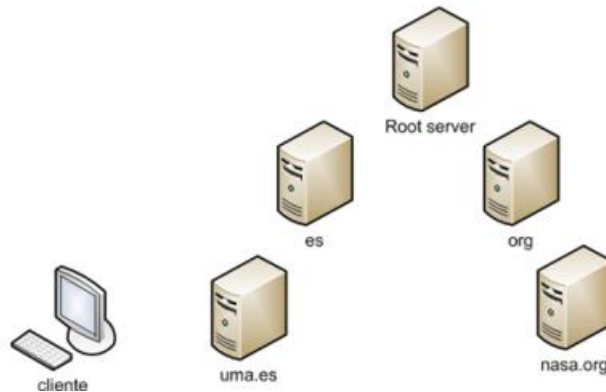
MAIL<SP>FROM:<sender><CRLF>: especifica el remitente/emisor del mensaje

RCPT<SP>TO:<receiver><CRLF>: especifica el receptor del mensaje

DATA<CRLF>: especifica el contenido del mensaje, con su cabecera y cuerpo. Termina en <CRLF>.<CRLF>

QUIT<CRLF>: cierre de conexión de transmisión con el servidor

17. Indique cuántas peticiones DNS tiene que realizar la máquina del cliente del siguiente dibujo para resolver la dirección www.nasa.org si la resolución es iterativa. ¿Y si la resolución es recursiva?



Iterativa: 5 peticiones. Cliente -> uma.es; Cliente -> es; Cliente -> Root server; Cliente -> org; Cliente -> nasa.org

Recursiva: 1 petición. Cliente -> uma.es; uma.es -> es; es -> Root server; Root server -> org; org -> nasa.org

18. Indique dos mecanismos/protocolos para acceder al buzón de correo electrónico y describa una característica de cada uno de ellos.

POP3 (protocolo de la oficina de correos): permite la descarga de correos electrónicos almacenados en el buzón del usuario. Protocolo TCP/110. Dos modos de funcionamiento: borrado y mantenimiento

IMAP4 (protocolo de acceso a correo de Internet): Más complejo que POP3, para usuarios nómadas. Protocolo TCP/143. Permite gestionar el buzón (consultar mensajes, buscar, descarga parcial, crear carpetas, crear, borrar o renombrar buzones)

Correo basado en la Web: permite el acceso desde un navegador al buzón de correo electrónico. Combina el uso de SMTP con HTTP.

19. Una persona se encuentra en una máquina A y realiza una sesión de telnet a una máquina B. Durante dicha sesión en B, se conecta usando FTP a una máquina C, de la cual se descarga un fichero. ¿En qué máquina(s) se encontrará el fichero después de todo este proceso? Justifique la respuesta

En las máquinas B (se copia a B) y C (sigue en C)

20. Indique las diferencias entre la localización de contenidos centralizada y distribuida en redes P2P.

21. Si implementa un servicio usando UDP, ¿qué información necesita que el cliente le envíe en cada mensaje para poder responderle? ¿Y si fuese TCP?

En UDP, como cada datagrama es independiente, necesitamos la dirección de socket del cliente en cada uno (IP, puerto). En TCP, una vez establecida la conexión, únicamente necesitamos saber de qué cola leer o escribir.

22. ¿Qué protocolo de control de error/flujo visto en la capa de enlace es similar al que utiliza TCP? ¿Qué diferencias hay entre el protocolo visto en la capa de enlace y el utilizado por TCP?

Repetición selectiva.

Algunas diferencias son: Numeración basada en bytes en vez de tramas. Se usan ACK repetidos en vez de NACK para indicar que faltan y forzar la retransmisión. Tamaño de ventana variable.

23. Indica si la siguiente afirmación es cierta o falsa y justifica tu respuesta: “Para realizar una comunicación multicast en UDP todos los procesos receptores deben usar el mismo puerto”.

Falsa. No usan el mismo puerto, deben utilizar la misma dirección IP multicast.

24. Si se quiere enviar un correo con los siguientes datos:

Origen: alumno7@informatica.edu

Destino: profesor5@informatica.es y alumno7@informatica.edu como copia oculta

Asunto: Noticia

Contenido: I think, therefore, I am

Indique qué mensajes SMTP serían enviados por el cliente (en el correo debe indicarse únicamente el “From:”, “To:”, “Subject:” y contenido). ¿Cuántos bytes ocupa el último mensaje SMTP enviado por el cliente?

HELO <servidor>

MAIL FROM: alumno7@informatica.edu

RCPT TO: profesor5@informatica.es

RCPT TO: alumno7@informatica.edu

DATA

From: alumno7@informatica.edu

To: profesor5@informatica.es

Subject: Noticia

I think, therefore, I am

.

QUIT

El último mensaje SMTP ocupa 6 bytes (4 caracteres + 2 bytes por \r\n)

25. Se acaba de recibir un segmento TCP con la siguiente información:

# sec: 110	SYN	FIN	ACK	TAM: 270
# ack: 2460	0	0	1	WIN: 500

Donde TAM indica la longitud del segmento recibido y WIN el tamaño de la ventana de recepción anunciado. Además, se sabe que ningún segmento TCP usa opciones y el MSS negociado es 510. Si después del segmento previo no se ha recibido nada más, ¿cuál será el siguiente segmento enviado si queremos transmitir 2000 bytes?

Sec = 2460. Ack = 360. S = 0. F = 0. A = 1. TAM = 520. WIN = - (viene así para no indicar la ventana)

26. Si la ventana de envío de un extremo de TCP es ... [(34000 - 35399) 35400 – 35999] ... donde entre [] están los número de secuencia incluidos en la ventana y entre () están los número de secuencias enviados, pero en los que no se ha recibido su ACK, indica como se actualiza dicha ventana cuando (se puede suponer que el tamaño de la ventana de envío no cambia):

a. El extremo recibe un segmento con una ACK 20000

La ventana no se actualiza ni cambia su contenido ya que no contiene números de secuencia pendientes de confirmación menores que 20000

b. El extremo recibe un segmento con un ACK 34100

La ventana se actualiza, pasando a ser [(34100 – 35399) 35400 – 36099] ya que el ACK 34100 confirma los números de secuencia que son menores a 34100, sin incluirse a sí mismo

27. Dada la URL [telnet://https.google.edu:22/inicio.html](https://https.google.edu:22/inicio.html) indique qué protocolo de nivel de aplicación se usaría en la comunicación y la dirección socket del servidor (en vez de IP puede usar el nombre):

Recordemos que una URL sigue el esquema: protocolo://host:puerto/path, por lo que el protocolo de nivel de aplicación sería telnet y la dirección de socket (<protocolo transporte, IP, puerto>) del servidor (host) sería <TCP, https.google.edu, 22>.

28. Si usamos HTTP/2.0, ¿Cuántos flujos se establecen para descargar y representar correctamente una página que incluye 7 imágenes y 6 referencias alojadas en el mismo servidor? Justifica la respuesta

Para descargar 7 imágenes y 6 referencias alojadas en el mismo servidor, se deben establecer 7 flujos para la descarga de imágenes (a las referencias no se accederá hasta que cliqueemos sobre ellas, por lo que no se cuentan), sumando el flujo de control (que siempre necesitaremos) y el primer flujo para el acceso a la página en sí, tenemos $7+1+1 = 9$ flujos

29. En la cabecera de TCP tanto el puerto origen como el destino son obligatorios, mientras que en UDP el origen es opcional. ¿Por qué es obligatorio en TCP y no lo es en UDP?

En UDP es posible que el datagrama enviado no necesite respuesta, por lo que el origen es opcional. Mientras que en TCP para crear e identificar la conexión necesito 4 datos (IP origen, IP destino, puerto origen, puerto destino), por lo que tienen que estar siempre definidos

30. ¿Por qué el bit ACK debe estar desactivado siempre que el primer segmento tenga activo el bit SYN?

Porque es el primer segmento a enviar y no hay ninguno anterior a confirmar. Además de que el emisor no sabría cuál debería ser el número de ACK a enviar.

31. ¿Qué hace TCP cuando recibe un intento de conexión y tiene la cola de clientes por conectar llena?

Rechaza la conexión, enviando un segmento con el bit RST

32. Como mecanismo para hacer secretas las peticiones de DNS, una de las propuestas es DNS over HTTPS (DoH). Indique y explique una desventaja de este mecanismo en cuanto a eficiencia.

Mayor carga, ya que DNS, que va sobre UDP, generará dos mensajes (uno en ida y otro de vuelta como respuesta), mientras que DoH, que irá sobre TCP, generará mínimo 8 mensajes (3 para el inicio de la conexión + 1 para la respuesta + 3 para el cierre de la conexión).

33. ¿Qué diferencia importante hay en las respuestas de POP3 respecto a otros protocolos como SMTP, FTP o HTTP?

POP3 no usa códigos de respuesta (ni el mensaje de respuesta que acompañaría al código), solo responde con "OK<CRLF>" y "-ERR<CRLF>"

34. Indique dos tipos de servidores DNS e identifique qué parte resuelve cada uno en www.lcc.uma.es.

Root server: resuelve el "." Final (devolviendo el dominio autorizado uma.es (o lcc.uma.es))

TLD: Resuelve el ".es" (devolviendo el dominio autorizado uma.es (o lcc.uma.es))

ANS: resuelve el "uma.es" (o el "lcc.uma.es", o www.lcc.uma.es) (devolviendo la dirección IP de www.lcc.uma.es)

35. ¿Para qué se usa el puerto 20 en FTP en modo activo? ¿Y en modo pasivo?

En modo activo el puerto 20 es el puerto que utiliza como cliente el servidor para conectarse al cliente que actúa en la conexión de datos como servidor.

En modo pasivo no se utiliza para nada

36. ¿Por qué es necesario indicar los destinatarios de un correo mediante un comando "RCPT TO" si ya vienen en la cabecera "To:" dentro de DATA?

Porque lo que hay dentro de DATA no lo utiliza el protocolo SMTP para las características del envío. El protocolo se fija en lo especificado en "RCPT TO" para identificar el destinatario del correo

37. ¿Qué diferencia hay entre una petición GET y una POST en HTTP?

La diferencia principal es cómo codifican la información que quieren obtener/procesar. El GET envía la información a solicitar dentro de la propia URL, mientras que POST los mete dentro del cuerpo del mensaje HTTP.

Esto implica que con GET no se puede enviar cualquier tamaño de información, ya que el tamaño de las URLs está limitado, mientras que el cuerpo de una petición POST, no.