

# Actividad de Cierre III

---

## SISTEMAS CONCURRENTES Y DISTRIBUIDOS

Ignacio David Vázquez Pérez

218292866

---

### Objetivo:

Recordar los conocimientos previos y responder las preguntas correctamente.

### Introducción:

La actividad aborda aspectos cruciales en el mundo de las comunicaciones y las redes, centrándose en protocolos, el modelo OSI, ATM (Asynchronous Transfer Mode), el modelo cliente-servidor, y el funcionamiento de primitivas y RPC (Remote Procedure Call). Estos conceptos desempeñan un papel fundamental en la infraestructura tecnológica de hoy en día y son esenciales para comprender cómo funcionan las redes de datos y la comunicación en entornos informáticos.

### Actividades a realizar

#### 1. ¿Qué es un protocolo?

Un protocolo es un conjunto de reglas y convenciones que establecen cómo deben comunicarse y interactuar dos o más dispositivos o sistemas en una red. Define el formato de los mensajes, el orden de las operaciones y las acciones a tomar en diversas situaciones para garantizar una comunicación efectiva y fiable.

#### 2. Cuáles son las capas de modelo OSI en orden

Las capas del modelo OSI en orden son: 1. Física, 2. Enlace de Datos, 3. Red, 4. Transporte, 5. Sesión, 6. Presentación, 7. Aplicación.

#### 3. Define X.25 y su modo de conexión

X.25 es un antiguo protocolo de comunicaciones de nivel de enlace de datos utilizado para la transmisión de datos a través de redes de conmutación de paquetes. Su modo de conexión implica el establecimiento de un circuito virtual entre los dispositivos conectados antes de la transmisión de datos, lo que garantiza la entrega secuencial y confiable de paquetes.

#### 4. ¿Cuál capa del modelo OSI se encarga del significado de los datos?

La capa de Aplicación del modelo OSI es la que se encarga del significado de los datos. En esta capa, los datos se interpretan y se brinda la interfaz para que las aplicaciones interactúen con la red.

## **5. ¿Qué es ATM?**

ATM (Asynchronous Transfer Mode) es un protocolo de comunicación de alta velocidad que utiliza conmutación de celdas y es utilizado para la transmisión de datos, voz y video en redes de alta velocidad.

## **6. ¿Qué es SONET y por cuantos bytes está formado?**

SONET (Synchronous Optical Network) es un estándar de transmisión de datos a través de fibras ópticas. Está formado por tramas que tienen 810 bytes de longitud, conocidas como tramas SONET STS-1.

## **7. ¿Qué es latencia?**

La latencia es el tiempo que transcurre desde que se inicia una solicitud o acción hasta que se obtiene una respuesta o se completa la acción. Se mide en milisegundos y puede afectar la velocidad de comunicación en redes y la capacidad de respuesta de los sistemas.. Representa el tiempo que tarda un paquete de datos en viajar desde el emisor al receptor.

## **8. Menciona dos ventajas del modelo Cliente-Servidor**

Dos ventajas del modelo Cliente-Servidor son la escalabilidad, ya que es posible agregar más clientes y servidores según sea necesario, la modularidad, que permite modificar o actualizar partes del sistema sin afectar a otras y la centralización de recursos y servicios, lo que facilita la administración y el control.

## **9. Menciona las capas del modelo cliente servidor y su relaciona con el modelo OSI**

Las capas del modelo Cliente-Servidor suelen relacionarse con el modelo OSI de la siguiente manera:

1. Capa de Aplicación (Cliente-Servidor) con Capa de Aplicación (OSI)
2. Capa de Presentación (Cliente-Servidor) con Capa de Presentación (OSI)
3. Capa de Transporte (Cliente-Servidor) con Capa de Transporte (OSI)
4. Capa de Red (Cliente-Servidor) con Capa de Red (OSI)

## **10. ¿Cuáles son los aspectos de diseño del modelo cliente-servidor?**

Los aspectos de diseño del modelo Cliente-Servidor incluyen la definición de roles y responsabilidades de clientes y servidores, el diseño de protocolos de comunicación, la gestión de conexiones y sesiones, la seguridad, la escalabilidad y la administración de recursos compartidos.

## **11. En que consiste el direccionamiento MACHINE.PROCESS**

El direccionamiento MACHINE.PROCESS es un método de direccionamiento en el que los procesos en una red se identifican mediante dos componentes: "machine" (máquina) que se refiere a la dirección de la máquina o nodo en la red, y "process" (proceso) que se refiere a la identificación única del proceso en esa máquina.

## **12. En que consiste el direccionamiento MACHINE.LOCAL-ID**

El direccionamiento MACHINE.LOCAL-ID es similar al direccionamiento MACHINE.PROCESS, pero el "machine" se refiere a la identificación local de la máquina en un contexto específico, en lugar de una dirección global en la red.

## **13. Que es un servidor de nombres**

Un servidor de nombres es un sistema o servicio que se utiliza para asignar y recuperar direcciones o nombres de recursos en una red, lo que facilita la identificación y localización de dispositivos y servicios.

## **14. Describe el funcionamiento de primitivas con bloqueo y sin bloqueo**

- **Primitivas con Bloqueo:** Las primitivas con bloqueo son operaciones que pueden bloquear un hilo o proceso hasta que ciertas condiciones se cumplan. Esto significa que el hilo que realiza una operación bloqueante puede quedar inactivo hasta que se libere el recurso deseado.
- **Primitivas sin Bloqueo:** Las primitivas sin bloqueo son operaciones que permiten que un hilo o proceso continúe su ejecución incluso si no puede completar una operación. Si la operación no puede llevarse a cabo de inmediato debido a conflictos con otros hilos, el hilo que la realiza se reorganiza y vuelve a intentar más tarde.

## **15. Describe el funcionamiento de primitivas no almacenadas**

Las primitivas no almacenadas son mecanismos de sincronización y comunicación en los que los datos o mensajes no se almacenan temporalmente en un buffer intermedio. En lugar de eso, los datos se transfieren directamente entre los procesos emisor y receptor. Esto significa que el proceso emisor debe esperar hasta que el proceso receptor esté listo para recibir los datos.

## **16. Describe el funcionamiento de primitivas almacenadas**

Las primitivas almacenadas son mecanismos de sincronización y comunicación en los que los datos o mensajes se almacenan temporalmente en un buffer o cola antes de ser consumidos por otro proceso. Esto permite que los procesos funcionen de manera asíncrona y evita bloqueos.

## **17. Qué es un buzón**

Un buzón es una estructura de datos que se utiliza para almacenar mensajes o datos de manera temporal hasta que sean recogidos y procesados por otro proceso o hilo.

## 18. Qué es una primitiva confiable y que es una no confiable

Una primitiva confiable es una operación o protocolo que garantiza la entrega y la integridad de los datos transmitidos. Una primitiva no confiable no ofrece estas garantías y asume que pueden ocurrir errores o pérdida de datos.

## 19. Qué es un reconocimiento

Un reconocimiento es un mensaje o señal enviado por el receptor de un mensaje o paquete de datos para confirmar que ha recibido correctamente el mensaje. Es esencial en la comunicación confiable para verificar la entrega exitosa.

## 20. Enlista los tipos de paquetes utilizados en el protocolo clientes-servidor

En un protocolo Cliente-Servidor, se pueden utilizar varios tipos de paquetes, como solicitudes (requests), respuestas (responses), reconocimientos (acknowledgments), y mensajes de control (control messages), entre otros.

## 21. ¿Que es RPC?

RPC significa "Remote Procedure Call" (Llamada a Procedimiento Remoto) y es un protocolo que permite a un programa en una computadora solicitar servicios o procedimientos en otra computadora de manera transparente, como si fueran locales, sin tener que entender los detalles de la red.

## 22. Como funciona RPC

RPC funciona permitiendo que un programa llame a un procedimiento (función) en una máquina remota como si fuera una llamada local. El programa cliente emite una solicitud RPC, que es transmitida a través de la red al servidor. El servidor ejecuta el procedimiento solicitado y devuelve el resultado al cliente.

## 23. Cuantos y cuales son los modos de transferencia de parámetros

- **Por valor (value):** Los parámetros se pasan al servidor como copias de los valores originales.
- **Por referencia (reference):** Los parámetros se pasan al servidor como referencias a los valores originales.
- **Por resultado (result):** Los parámetros se pasan al servidor como valores de retorno y son actualizados por el servidor.

## 24. ¿Cuales son los pasos para generar una llamada a un procedimiento remoto (RCP)?

Los pasos para generar una llamada a un procedimiento remoto en RPC son:

1. El cliente llama a un procedimiento remoto como si fuera local.

2. El cliente genera una solicitud RPC que contiene la información del procedimiento y los argumentos.
3. La solicitud RPC se transmite a través de la red al servidor remoto.
4. El servidor recibe la solicitud y la procesa ejecutando el procedimiento solicitado.
5. El servidor devuelve el resultado al cliente a través de una respuesta RPC.
6. El cliente recibe la respuesta y puede continuar con su ejecución.

## **25. Como es la comunicación en grupo**

La comunicación en grupo se refiere a la interacción entre múltiples participantes o nodos en una red o sistema, donde un mensaje enviado por uno de los participantes puede ser recibido por varios otros participantes al mismo tiempo. Esto permite la difusión de información a varios receptores simultáneamente y se utiliza en aplicaciones como chats grupales, videoconferencias, difusión de mensajes en redes sociales y más.

## **Conclusión**

Hemos abordado conceptos con las redes de comunicación, los modelos de comunicación, los protocolos y las primitivas de comunicación. Algunos de los aspectos destacados incluyen la comprensión de lo que es un protocolo y su importancia en la comunicación, la descripción de las capas del modelo OSI y su relación con el modelo Cliente-Servidor, así como la explicación de diferentes enfoques de direccionamiento y los conceptos de primitivas con bloqueo, sin bloqueo y no almacenadas.

También hemos explorado temas como la confiabilidad en las comunicaciones, el reconocimiento como parte fundamental de la comunicación confiable y los diferentes tipos de paquetes utilizados en el protocolo Cliente-Servidor. Además, hemos discutido el concepto de RPC (Remote Procedure Call) y sus modos de transferencia de parámetros, así como los pasos para generar una llamada a un procedimiento remoto y la comunicación en grupo.

Estos conceptos son fundamentales en el campo de la comunicación y la programación distribuida, ya que permiten la interacción eficiente entre sistemas y la ejecución de procedimientos en máquinas remotas. Comprender estos conceptos es esencial para desarrolladores de software y profesionales de redes que trabajan en entornos distribuidos, y puede ayudar a facilitar la comunicación efectiva y la implementación de servicios remotos en aplicaciones y sistemas.

## **Bibliografía**

- Tanenbaum Andrew. (1995). Sistemas Operativos Distribuidos. España. Prentice-Hall Hisp.
- McIver Ann. (2011). Sistemas Operativos. México. Cengage Learning.
- Tanenbaum, A., & Van Steen M. (2008). Sistemas Distribuidos, Principios y Paradigmas. (Segunda ed.). Prentice Hall.
- Tanenbaum, A. (2011). Redes de Computadoras. (Quinta ed.). Prentice Hall.

- Elmasri, R., Gil Carrick, A., & Levine, D. (2010). Sistemas Operativos, Un enfoque en espiral. McGraw--Hill.
- Mozilla Developer Network. (s.f.). Mensajes HTTP.  
<https://developer.mozilla.org/es/docs/Web/HTTP/Messages>
- Díaz, F. (2006). Sistemas distribuidos: comunicación entre procesos.  
[https://www.infor.uva.es/~fdiaz/sd/2005\\_06/doc/SD\\_TE02\\_20060305.pdf](https://www.infor.uva.es/~fdiaz/sd/2005_06/doc/SD_TE02_20060305.pdf)
- Reyes, A. (s.f.). Pasos de una llamada a un procedimiento remoto.  
<http://aisii.azc.uam.mx/areyes/archivos/licenciatura/sd/U5/PasosRPC.pdf>
- Sosa, V. (2006). RPC: Llamadas a procedimientos remotos.  
[https://www.tamps.cinvestav.mx/~vjsosa/clases/sd/RPC\\_Todo.pdf](https://www.tamps.cinvestav.mx/~vjsosa/clases/sd/RPC_Todo.pdf)
- Microsoft Learn. (s.f.). Data Transfer of Table-Valued Parameters.  
<https://learn.microsoft.com/es-es/sql/relational-databases/native-client-odbc-table-valued-parameters/binding-and-data-transfer-of-table-valued-parameters-and-column-values?view=sql-server-ver16>