



**Uleam**  
UNIVERSIDAD LAICA  
ELOY ALFARO DE MANABÍ

# TECNOLOGÍA DE LA INFORMACIÓN

## **ESTUDIANTE:**

Vera Soledispa Mary Mayerli

## **TEMA:**

Informe parcial Sistemas Distribuidos

## **CURSO:**

8vo "A"

## **FECHA:**

29/05/2024

## **DOCENTE:**

Ing. Cesar Sinchiguano

## **Arquitectura de los sistemas distribuidos**

### **Introducción**

La tecnología se ha vuelto cada vez más interconectada y distribuida. Los sistemas distribuidos son una pieza fundamental de esta interconexión, permitiendo que dispositivos y servicios se comuniquen y colaboren entre sí de manera efectiva. La arquitectura de estos sistemas es fundamental para su diseño y funcionamiento, definiendo cómo se organizan y comunican los diversos componentes de software que los componen. Desde arquitecturas en capas hasta modelos cliente-servidor y más allá, la elección de una arquitectura adecuada puede influir en la escalabilidad, eficiencia y seguridad de un sistema distribuido.

La complejidad inherente a estos entornos distribuidos presenta desafíos significativos, pero también oportunidades para la innovación y el avance tecnológico. Comprender y abordar estos desafíos es esencial para construir sistemas robustos y eficientes que impulsen el desarrollo continuo de la informática distribuida.

### **Arquitectura de los sistemas distribuidos**

La organización y estructura de los sistemas distribuidos dependen en gran medida de la disposición de los componentes de software que constituyen el sistema. Estas arquitecturas de software definen cómo se organizan y comunican varios componentes del sistema entre sí. La implementación de un sistema distribuido implica la división e identificación de estos componentes de software, así como su despliegue en máquinas físicas reales. La configuración final de estos componentes, su interacción y su despliegue se conocen como arquitectura de software.

Existen varios estilos arquitectónicos que definen la estructura de los sistemas distribuidos. Algunos de los más importantes son:

### ***Arquitecturas en capas***

El estilo arquitectónico en capas organiza los componentes en capas jerárquicas, donde cada capa puede acceder a la capa inmediatamente inferior. Este estilo facilita el flujo de control de arriba hacia abajo y de abajo hacia arriba. Por ejemplo, un servidor de aplicaciones web puede constar de una capa de presentación, una capa de lógica empresarial y una capa de acceso a datos.

### ***Arquitecturas basadas en objetos***

Las arquitecturas basadas en objetos conectan componentes a través de mecanismos de invocación remota, donde cada objeto encapsula funcionalidades específicas. Este estilo se adapta bien al modelo cliente-servidor y permite una mayor flexibilidad en la comunicación entre componentes.

### ***Arquitecturas centradas en datos***

Las arquitecturas centradas en datos giran en torno a un repositorio central o medio común a través del cual los diferentes procesos se comunican. Por ejemplo, en un sistema distribuido que utiliza un sistema de archivos compartido, la comunicación entre los nodos se establece mediante el acceso a este archivo compartido.

### ***Implementación de aplicaciones en tres capas***

Una forma común de organizar los sistemas distribuidos es mediante una arquitectura de tres capas, que incluye:

- **Interfaz de usuario:** Esta capa proporciona la interfaz a través de la cual los usuarios interactúan con la aplicación. Puede ser tan simple como una pantalla de texto o tan compleja como una interfaz gráfica de usuario (GUI).
- **Procesamiento:** En esta capa se encuentran las aplicaciones y la lógica empresarial que procesan las solicitudes del usuario y realizan operaciones en los datos.
- **Datos:** Esta capa gestiona los datos con los que opera la aplicación y proporciona acceso a ellos para las capas superiores.

En muchas aplicaciones cliente-servidor, estas tres capas se utilizan para separar claramente las diferentes responsabilidades y facilitar el desarrollo y mantenimiento del sistema.

### *Arquitecturas Cliente-Servidor*

El modelo cliente-servidor es uno de los enfoques más comunes para la construcción de sistemas distribuidos. En este modelo, los procesos se dividen en dos grupos: los servidores, que proporcionan servicios específicos, y los clientes, que solicitan y consumen estos servicios. La interacción entre un cliente y un servidor sigue el patrón de solicitud-respuesta, donde el cliente envía una solicitud al servidor y espera una respuesta.

### **Seguridad de los sistemas distribuidos**

Los sistemas distribuidos, aunque ofrecen numerosas ventajas en términos de escalabilidad, eficiencia y flexibilidad, también presentan desafíos significativos en términos de seguridad. Es esencial comprender y abordar estas preocupaciones para garantizar la integridad, confidencialidad y disponibilidad de los datos en un entorno distribuido.

Una de las principales preocupaciones en los sistemas distribuidos es la seguridad de la comunicación. La transmisión de datos a través de redes puede ser vulnerable a la interceptación

y manipulación por parte de terceros no autorizados. Por lo tanto, implementar protocolos de cifrado robustos, tanto simétricos como asimétricos, es fundamental para proteger la confidencialidad de la información transmitida.

Además, el control de acceso es crucial en un entorno distribuido donde múltiples usuarios pueden acceder a recursos compartidos. La implementación de firewalls y mecanismos de autenticación sólidos ayuda a garantizar que solo las personas autorizadas puedan acceder a los datos y recursos del sistema.

Otro aspecto importante es la integridad de los datos. En un sistema distribuido, donde múltiples nodos pueden contribuir o modificar información, es fundamental contar con mecanismos para garantizar que los datos permanezcan intactos y no sean alterados por usuarios no autorizados. La implementación de firmas digitales y registros de cambios puede ayudar a rastrear y verificar la integridad de los datos a lo largo del tiempo.

Además de estas medidas específicas, es importante abordar las vulnerabilidades específicas de los sistemas distribuidos, como la falta de un reloj global para sincronizar las operaciones y la posible latencia en la propagación de la información. Esto puede requerir el desarrollo de algoritmos y protocolos adaptados específicamente para entornos distribuidos para garantizar un rendimiento óptimo y una seguridad sólida.

### **Desafíos de los sistemas distribuidos**

Los sistemas distribuidos son un caos en constante cambio, donde cada decisión lleva consigo una serie de desafíos. La complejidad a estos sistemas se manifiesta en múltiples formas, desde la gestión de la latencia y la escalabilidad hasta la comprensión de la comunicación entre servidores y la complejidad de los algoritmos utilizados.

Un aspecto fundamental en los sistemas distribuidos es la gestión de los errores. La mera posibilidad de fallas en cualquier nivel de la red complica enormemente la tarea de los ingenieros. En un sistema distribuido, cada componente puede fallar de forma independiente, lo que significa que los errores pueden surgir en cualquier momento y de diversas formas impredecibles.

La gestión de errores en los sistemas distribuidos no solo implica prevenir y solucionar fallas, sino también comprender y mitigar su propagación. Un error en un servidor puede propagarse rápidamente a otros componentes, afectando la integridad y el funcionamiento del sistema en su conjunto. Este fenómeno se asemeja a una epidemia, donde un error inicial puede desencadenar una cascada de problemas en toda la red.

Para abordar estos desafíos, es crucial adoptar un enfoque proactivo en la detección y resolución de errores. La detección temprana de fallas, preferiblemente durante las etapas de desarrollo y pruebas, puede evitar problemas más graves en producción. Además, es fundamental comprender la naturaleza compleja de la comunicación en red y diseñar sistemas que sean resilientes a fallos y capaces de manejar situaciones imprevistas.

La complejidad de los sistemas distribuidos puede resultar abrumadora, pero también presenta oportunidades para la innovación y el avance tecnológico. Al comprender y abordar los desafíos inherentes a estos sistemas, podemos construir infraestructuras más robustas y eficientes que impulsen el desarrollo y la evolución de la informática distribuida.

## **Conclusiones**

La diversidad de estilos arquitectónicos, como las arquitecturas en capas, basadas en objetos y centradas en datos, ofrece a los diseñadores de sistemas distribuidos flexibilidad para

adaptar la estructura de acuerdo con los requisitos específicos de cada aplicación. Esta variedad permite optimizar la comunicación y el rendimiento de los sistemas distribuidos según las necesidades del proyecto.

La seguridad en los sistemas distribuidos es un desafío constante que requiere la implementación de medidas como el cifrado de datos, el control de acceso y la garantía de integridad de la información. La complejidad de estos entornos demanda un enfoque integral que aborde tanto las vulnerabilidades tradicionales como los desafíos únicos que surgen de la naturaleza distribuida de la infraestructura.

Los desafíos inherentes a los sistemas distribuidos, como la gestión de errores y la comprensión de la comunicación entre servidores, son obstáculos significativos, pero también oportunidades para la innovación. Al adoptar un enfoque proactivo en la detección y resolución de problemas, podemos construir sistemas más robustos y eficientes que impulsen el desarrollo continuo de la informática distribuida.

## Bibliografía

- Arízaga-Gamboa, J., Alvarado-Unamuno, E., & Chicala- Arroyave, J. (2021). Seguridad en Sistemas Distribuidos caso proyecto FCI. *Serie Científica de la Universidad de las Ciencias Informáticas*, 14(4), 1-15.  
<https://publicaciones.uci.cu/index.php/serie/article/download/849/724/#:~:text=La%20Seguridad%20en%20Sistemas%20Distribuidos,m%C3%A1quina%20que%20requiere%20de%20un>
- Gabrielson , J. (2019). *Amazon Builders' Library*". AWS: <https://aws.amazon.com/es/builders-library/challenges-with-distributed-systems/>
- LOPEZ FUENTES, F. D. (2015). *Universidad Autónoma Metropolitana*.  
<http://ilitia.cua.uam.mx:8080/jspui/handle/123456789/173>
- ZETTLER, K. (s.f.). *Atlassian*. <https://www.atlassian.com/es/microservices/microservices-architecture/distributed-architecture>