

# FUNDAMENTOS DE REDES, P2P VS CLIENTE-SERVIDOR

ISAAC VIEYRA SANTOS

ING. EN SOFTWARE

ANDRES LUNA JAIMES

CLIENTE SERVIDOR

UNIDAD 3

UNIVERSIDAD POLITÉCNICA DE LÁZARO CÁRDENAS

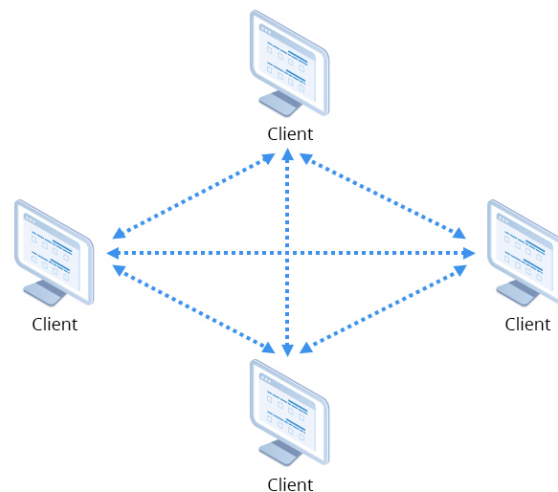


## CONCEPTO Y FUNCIONAMIENTO

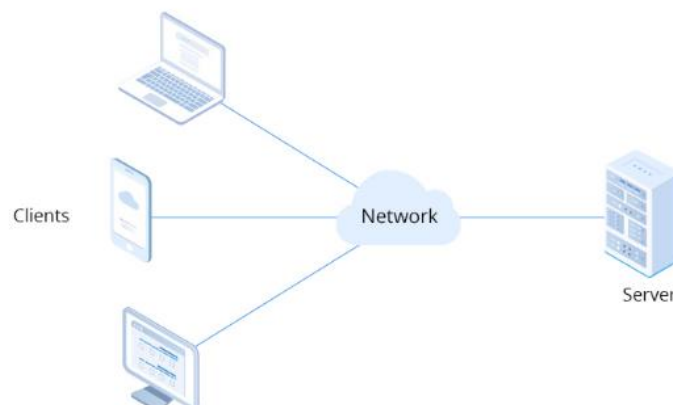
Cuando se habla de cómo se organizan las computadoras para comunicarse y compartir recursos en una red, es común escuchar los términos "cliente-servidor" y "peer-to-peer" (P2P). Ambas son formas de topologías de red, es decir, maneras en que los dispositivos (como ordenadores, teléfonos o servidores) se conectan entre sí para intercambiar información, acceder a programas o compartir archivos. Aunque cumplen el mismo objetivo de permitir el intercambio de recursos, sus métodos de funcionamiento son fundamentalmente diferentes.

En una arquitectura cliente-servidor, existe una o más computadoras dedicadas, conocidas como servidores, que actúan como proveedores centralizados de recursos y servicios. Los demás dispositivos en la red, llamados clientes, se conectan a estos servidores para solicitar y acceder a dichos recursos.

- CLIENTE SERVIDOR



- P2P



## **VENTAJAS Y DESVENTAJAS**

### **(C-S)**

#### **1. VENTAJAS**

- La gestión centralizada de los servidores facilita al personal de TI la actualización de los archivos de datos porque pueden realizar todos los cambios en una ubicación central. También pueden supervisar los datos desde un único servidor para anticiparse a posibles problemas.
- La información crítica está más protegida contra amenazas externas al almacenarse en un único servidor en lugar de múltiples clientes.
- Se pueden añadir nuevas funciones a un servidor sin tener que interrumpir el funcionamiento normal de otros dispositivos.
- Los datos compartidos a través de una red cliente-servidor pueden ser utilizados por todos los clientes y pueden recuperarse más fácilmente que a través de una red de servidores descentralizada.

#### **2. DESVENTAJAS**

- El precio de compra y mantenimiento de los servidores centrales puede ser elevado. Consecuentemente, el coste global de configuración de la red cliente-servidor también tiende a ser alto.
- Esta arquitectura requiere un gran nivel de experiencia de los técnicos de red para la configuración y la gestión del hardware y el software del servidor.
- En caso de fallo en el servidor central, probablemente se interrumpa la conexión en todos los ordenadores o dispositivos de la red.

## (P2P)

### 1. VENTAJAS

- Es fácil compartir archivos. Una red peer-to-peer avanzada permite compartir archivos a larga distancia y acceder a ellos en cualquier momento.
- No es necesario invertir en servidores centrales, además, esta arquitectura de red no requiere un administrador dedicado a jornada completa.
- Se pueden añadir fácilmente nuevos clientes a la red, lo que contribuye a su flexibilidad y escalabilidad.
- En caso de fallo de uno de los ordenadores, el resto seguirá en funcionamiento. Además, como el tráfico se distribuye entre varios ordenadores, se evitan los cuellos de botella.
- Permite la colaboración entre distintos clientes que realizan diferentes funciones, lo que beneficia a toda la red.

### 2. DESVENTAJAS

- La gestión de la seguridad en los ordenadores individuales en lugar del conjunto de la red suele implicar un menor nivel de seguridad.
- Cada ordenador necesita su propio sistema de copias de seguridad y software antivirus, lo que puede aumentar el coste global de la gestión de la red.
- El rendimiento y los servicios prestados en todo el sistema pueden ser más lentos, ya que cada ordenador realiza más de una tarea y tiene el acceso abierto al resto de peers.

## COMPARACION DE RENDIMIENTO, SEGURIDAD Y ESCALABILIDAD

Característica	Cliente-Servidor	Peer-to-Peer (P2P)
Escalabilidad	Limitada por la capacidad del servidor, pero gestionable.	Altamente escalable, con la red creciendo al añadir pares.
Rendimiento	Depende del servidor, puede ser alto pero propenso a cuellos de botella.	Distribuido, puede ser muy eficiente para ciertas tareas (ej. descargas masivas) pero inconsistente.
Seguridad	Centralizada, más fácil de implementar y controlar, pero con un punto único de fallo.	Distribuida, más resistente a ataques a un solo punto, pero más difícil de gestionar y susceptible a riesgos individuales.

## **EJEMPLOS REALES**

### Compartir archivos:

Programas como BitTorrent utilizan P2P para que los usuarios compartan archivos directamente entre sí.

### Criptomonedas:

Blockchain, la tecnología subyacente de Bitcoin, es una red P2P donde las transacciones son validadas por los "peers".

### Algunos juegos en línea:

Ciertos juegos multijugador conectan a los jugadores directamente entre sí sin un servidor central para el juego en sí.

## ANÁLISIS COMPARATIVO

CRITERIO	C-S	P2P
Definición	Un servidor centralizado provee recursos y servicios a múltiples clientes que los solicitan.	Un servidor centralizado provee recursos y servicios a múltiples clientes que los solicitan.
Ejemplos reales	Sitios web (tu navegador es cliente, el sitio es servidor), correo electrónico, bancos en línea, Google Drive.	Sitios web (tu navegador es cliente, el sitio es servidor), correo electrónico, bancos en línea, Google Drive.
Seguridad	Centralizada y controlada, facilitando la implementación de firewalls, autenticación y cifrado. Sin embargo, el servidor es un punto único de fallo.	Centralizada y controlada, facilitando la implementación de firewalls, autenticación y cifrado. Sin embargo, el servidor es un punto único de fallo.
Escalabilidad	Escalable vertical u horizontalmente añadiendo o mejorando servidores, pero puede encontrar cuellos de botella si la demanda excede la capacidad del servidor.	Escalable vertical u horizontalmente añadiendo o mejorando servidores, pero puede encontrar cuellos de botella si la demanda excede la capacidad del servidor.
Mantenimiento	Más complejo, requiere personal especializado para configurar, mantener y actualizar los servidores y la infraestructura central.	Más complejo, requiere personal especializado para configurar, mantener y actualizar los servidores y la infraestructura central.
Rendimiento en redes pequeñas	Alto rendimiento debido al control centralizado y conexiones directas cliente-servidor. Puede ser muy eficiente.	Alto rendimiento debido al control centralizado y conexiones directas cliente-servidor. Puede ser muy eficiente.
Rendimiento en redes grandes	Puede sufrir de cuellos de botella en el servidor si no se escala adecuadamente. La latencia puede aumentar si el servidor está distante.	Puede sufrir de cuellos de botella en el servidor si no se escala adecuadamente. La latencia puede aumentar si el servidor está distante.

**BIBLIOGRAFÍA:** <https://www.fs.com/>