

SESIÓN 7: RESUMEN DE SERVIDORES

Un **servidor** es un ordenador especialmente configurado para ofrecer servicios a otros ordenadores llamados clientes.

Puede estar en tu red local o en otro país.

Un servidor:

- suele estar encendido 24 horas,
- no necesita pantalla ni teclado,
- atiende peticiones constantemente,
- responde a miles (o millones) de clientes.

¿Qué hace realmente un servidor?

Un servidor puede:

- entregar páginas web,
- guardar archivos,
- procesar datos,
- gestionar correos,
- ejecutar aplicaciones.

Un servidor no “ve” ni “abre” páginas: simplemente procesa peticiones y responde datos.

Ejemplo cotidiano

Tú entras en Instagram.

Tu móvil (cliente) pide:

- el perfil
- las fotos
- la descripción
- los seguidores

Los servidores de Instagram te responden con esos datos en milisegundos.

2. ¿Qué es una IP?

Una **IP (Internet Protocol Address)** es un número único que identifica a un dispositivo en una red.

Es como la dirección de una casa en una ciudad: si no conoces la dirección, no puedes enviarle nada.

¿Para qué sirve una IP?

- Para localizar a un dispositivo.
- Para enviarle información.
- Para diferenciarlo de todos los demás.

Tipos de IP

1. **IP pública** : visible desde Internet.
2. **IP privada** : solo válida dentro de tu red local (como tu casa o instituto).
3. **localhost (127.0.0.1)** : el propio ordenador.

Ejemplo de vida real

Cuando escribes:

<https://www.google.com>

tu navegador pregunta:

“¿Cuál es la IP de google.com?”

Y el DNS responde algo como:

142.250.185.14

Tu navegador usa esa IP para conectarse.

Sin IP, es imposible encontrar un servidor. Es como intentar enviar una carta sin dirección.

3.PUERTO

¿Qué es un puerto?

Un puerto es un número que identifica qué servicio específico quieras usar en un servidor.

Una misma IP puede ofrecer muchos servicios.

Los puertos evitan confusiones.

Analogía

La IP es la dirección del edificio.

El puerto es el número de puerta o el despacho al que llamas.

Una IP puede tener:

- puerto 80 : servidor web
- puerto 22 : acceso remoto (SSH)
- puerto 3306 : base de datos
- puerto 3000 : una app en Node.js

¿Qué ocurre si no se pone puerto?

El navegador asume uno automáticamente:

- `http://` : puerto 80
- `https://` : puerto 443

El puerto indica qué servicio quieras dentro del servidor.

1 edificio = muchas puertas = muchos servicios.

4. HTTP y HTTPS

¿Qué son realmente?

HTTP y HTTPS son protocolos, es decir, *formas estándar de comunicación*.

Son el idioma que hablan:

- el navegador (cliente),
- y el servidor web.

Diferencias esenciales

- HTTP: comunicación normal, sin cifrar.
- HTTPS: comunicación cifrada con certificados SSL/TLS.

¿Para qué sirven?

Para pedir y recibir:

- páginas web
- imágenes
- datos
- formularios
- APIs

Ejemplo real

Tu navegador envía una petición así:

- GET /videos
- Host: youtube.com

El servidor responde:

- 200 OK

HTTP/HTTPS son las “reglas del idioma” que permiten que cliente y servidor se entiendan.

5. PROXY

Un **proxy** es un intermediario entre un cliente y un servidor.

Actúa como un “representante”: recibe las peticiones del cliente, las gestiona, y devuelve la respuesta.

¿Por qué existen los proxys?

Porque permiten:

- controlar el tráfico,
- filtrar contenido,
- mejorar seguridad,
- optimizar rendimiento,
- ocultar información,
- repartir carga.

Ejemplo de la vida real

El profesor recoge las dudas de todos los alumnos y va a secretaría a resolverlas.
Los alumnos nunca hablan directamente con secretaría.

El profesor = *proxy*

Secretaría = *servidor*

Alumnos = *clientes*

El proxy siempre está en medio. El cliente no ve al servidor real.

6. PROXY DIRECTO

¿Qué es?

Un proxy directo es un proxy colocado del lado del cliente.

El cliente *pasa primero por él* antes de salir a Internet.

¿Dónde se usa?

- Institutos
- Empresas
- Bibliotecas
- Cibers

¿Para qué sirve?

- Bloquear páginas
- Registrar historial de navegación
- Ahorrar ancho de banda
- Filtrar virus o spam
- Ocultar la IP del usuario

Ejemplo real

En un instituto:

1. El alumno intenta abrir “facebook.com”.
2. La petición pasa por el proxy del centro.
3. El proxy decide:
 - permitir
 - bloquear
 - registrar

El proxy directo “representa al cliente” y controla lo que puede hacer.

PROXY INVERSO

¿Qué es un proxy inverso?

Un proxy inverso está colocado del lado de los servidores, delante de ellos.

Los clientes *siempre* se comunican con el proxy inverso, nunca directamente con los servidores internos.

¿Por qué se llama “inverso”?

Porque el flujo es al revés que el proxy normal:

- en el normal : controla clientes
- en el inverso : controla servidores

¿Para qué sirve?

1. Seguridad

Oculta servidores internos, filtra ataques.

2. Rendimiento

Caché, compresión, optimización.

3. Escalabilidad

Balancea carga entre muchos servidores backend.

4. HTTPS

El proxy inverso maneja los certificados.

Ejemplo real

Una tienda online con miles de usuarios:

- 1 servidor no sirve.
- En lugar de exponer 10 servidores al exterior...
- Se expone solo uno: Nginx (proxy inverso).
- Nginx reparte peticiones entre los 10.

El proxy inverso es la “única puerta pública” para acceder a toda la infraestructura interna.

TABLA 1: Cliente vs Servidor

| Concepto | Cliente | Servidor |
|-----------|---------------------|------------------|
| Función | Pide | Responde |
| Ejemplo | Navegador | Nginx, Apache |
| Actividad | Ocasional | 24/7 |
| Ubicación | Hogar, insti, móvil | Centros de datos |

TABLA 2: Tipos de IP

| Tipo | Ejemplo | Uso | Visible desde fuera |
|-----------|-----------------|------------------|---------------------|
| Pública | 142.250.184.132 | Internet | Sí |
| Privada | 192.168.1.15 | Red local | No |
| Localhost | 127.0.0.1 | Pruebas en tu PC | No |

TABLA 3: Puertos comunes

| Puerto | Servicio | Descripción |
|-------------|----------|---------------|
| 80 | HTTP | Web normal |
| 443 | HTTPS | Web segura |
| 22 | SSH | Acceso remoto |
| 3306 | MySQL | Base de datos |
| 8080 / 3000 | Apps | Programación |

TABLA 4: HTTP vs HTTPS

| Protocolo | Seguridad | Uso |
|-----------|-----------|-------------|
| HTTP | Ninguna | Web normal |
| HTTPS | Cifrado | Web moderna |

TABLA 5: Proxy Directo vs Proxy Inverso

| Característica | Proxy Directo | Proxy Inverso |
|----------------|----------------------|---------------------------|
| Ubicación | lado del cliente | lado del servidor |
| Oculta | al cliente | a los servidores |
| Función | controlar navegación | proteger infraestructura |
| Caché | para usuarios | para tráfico entrante |
| Seguridad | filtra webs | filtra ataques |
| Uso típico | institutos | webs grandes |
| Ejemplo | proxy de empresa | Nginx delante de backends |