



TECNOLOGÍAS INFORMÁTICAS

A

Facultad de Ingeniería
Universidad Nacional de Mar del Plata



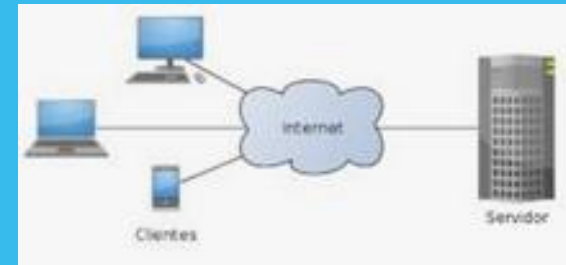
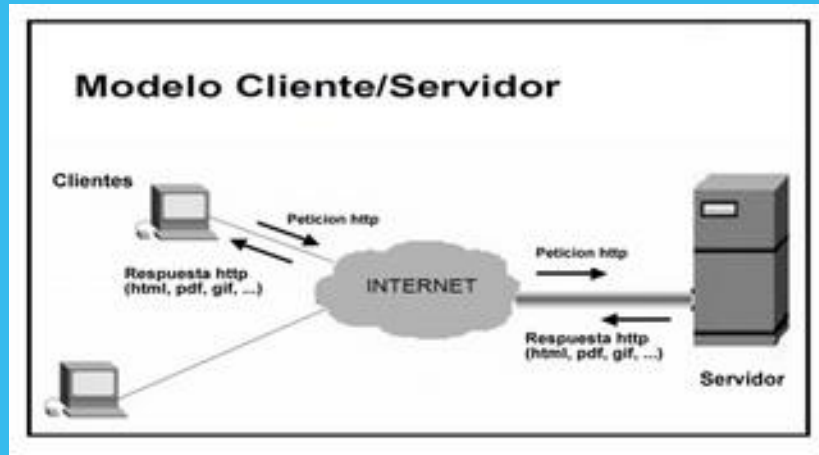
AGENDA

- CONCEPTOS DE CLIENTE SERVIDOR
- *INTRODUCCIÓN AL PROTOCOLO HTTP/HTTPS*
- *CONCEPTO DE HOSTS, DOMINIO, SUBDOMINIO Y URL.*
- *CONCEPTO DE DNS.*
- *REGISTRO DE DOMINIOS DE INTERNET.*
- *SUBIDA DE ARCHIVOS A UN SERVIDOR*

Cliente Servidor

El modelo cliente-servidor es uno de los conceptos de arquitectura más utilizados la tecnología de redes.

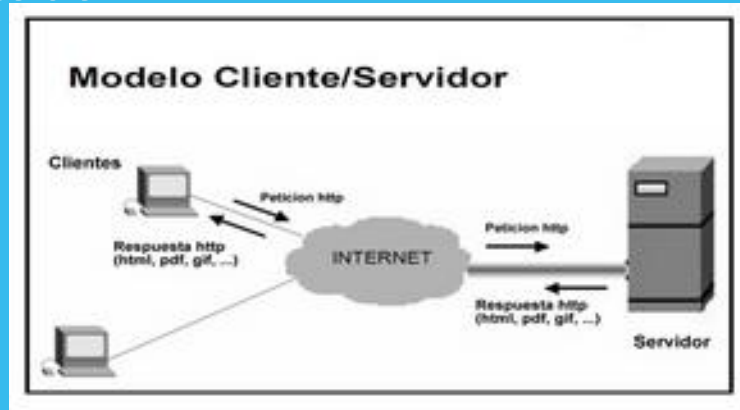
Regula la interacción entre el cliente y el Servidor.



Cliente Servidor

El servidor es un equipo en la red que proporciona los recursos necesarios para otros dispositivos.

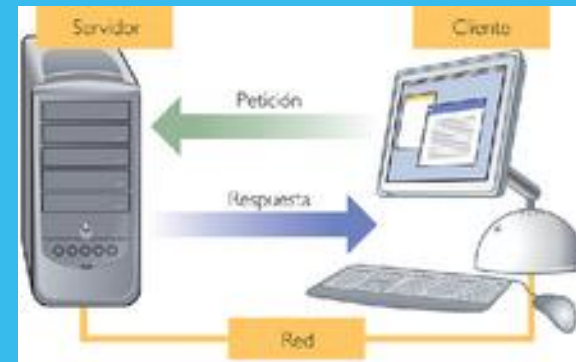
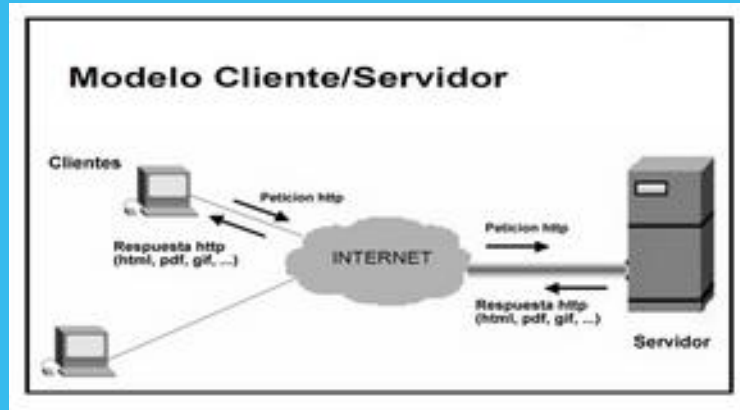
Acepta las peticiones del cliente, las procesa y proporciona la respuesta solicitada



Cliente Servidor

El cliente puede ser cualquier dispositivo en la red que se comunica con el servidor.

Envía solicitudes y recibe respuestas del servidor.

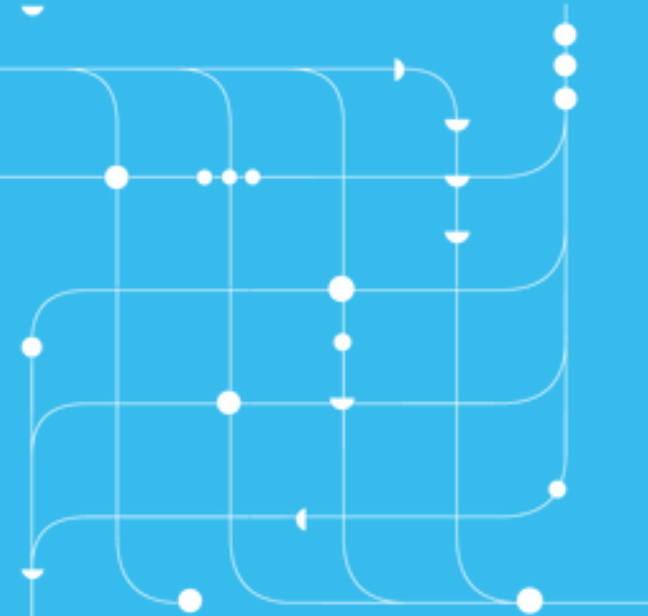


Características modelo cliente-servidor

Distribución de tareas entre los clientes y los servidores.

El servidor es el responsable de proporcionar los servicios. Se encarga de procesar las peticiones solicitados y entrega la respuesta.

El cliente solicita los servicios proporcionados. Recibe la respuesta del servidor.

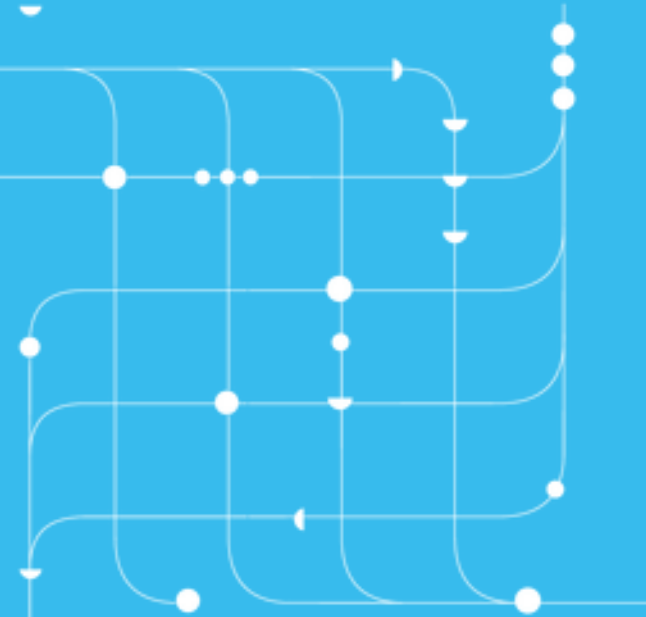


Características modelo cliente-servidor

Un servidor sirve a varios clientes y procesa múltiples peticiones. de diferentes clientes.

Para ello, presta su servicio de forma permanente y pasiva.

Un cliente solicita activamente los servicios del servidor e inicia las tareas del servidor.



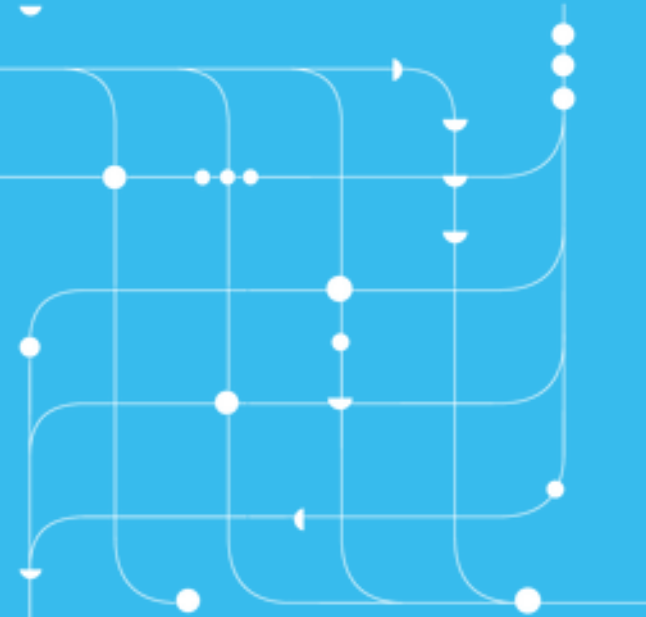
Características modelo cliente-servidor

Ventajas:

- Administración central
- Acceso controlados globalmente
- Un solo servidor para muchos clientes

Desventajas:

- Caída del servidor
- Recursos de un servidor



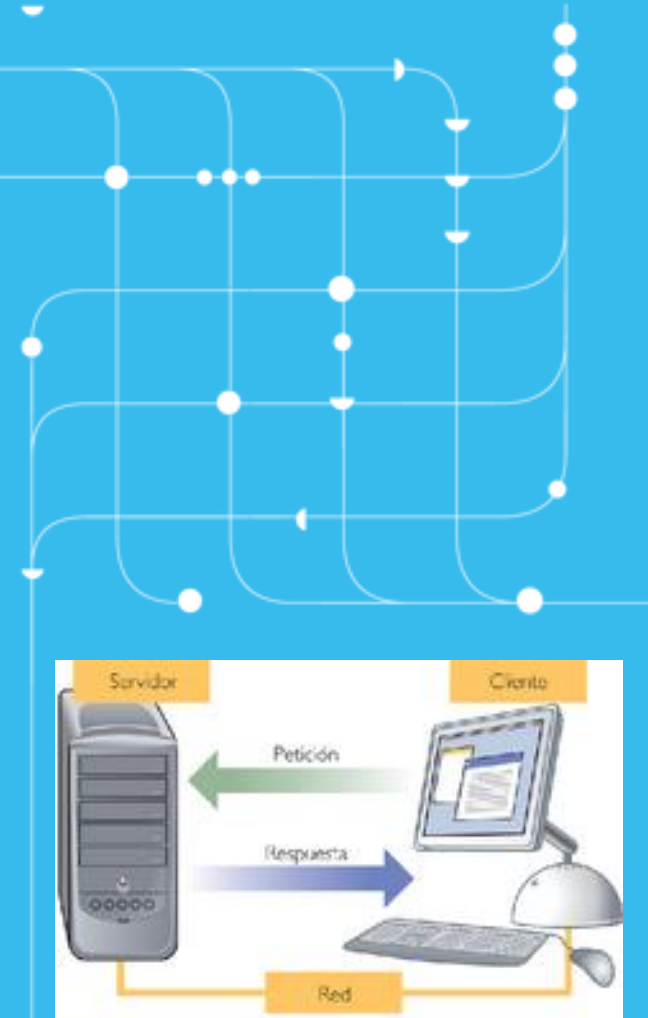
Ejemplos modelo cliente-servidor

Servidor web:

El cliente envía una petición al servidor web para abrir una página web concreta. El servidor devuelve al cliente los datos solicitados.

En el cliente la página web se muestra en el navegador del cliente.

Para enviar peticiones **HTTP** se utiliza el Hypertext Transfer Protocol.



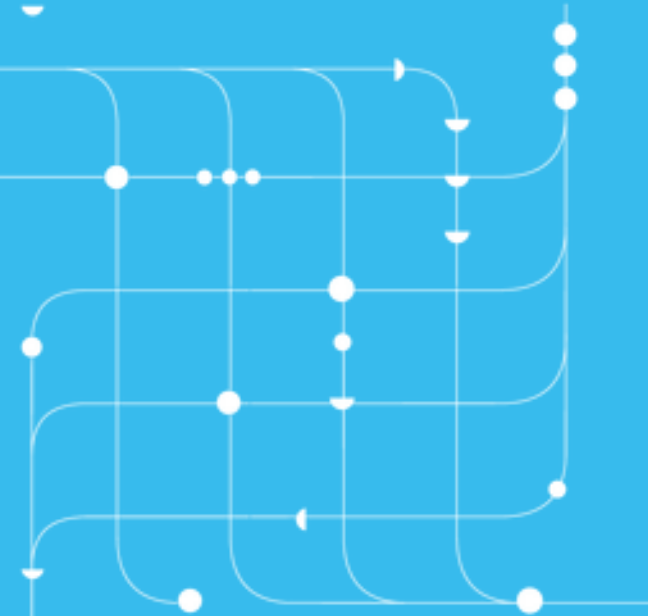
Ejemplos modelo cliente-servidor

Servidor de Correo electrónico:

Cuando un cliente de correo electrónico se comunica con un servidor, el cliente solicita y recupera los correos electrónicos que están en el servidor.

El servidor pone los correos electrónicos a disposición del cliente.

Los protocolos utilizados son SMTP, IMAP o POP y TLS.

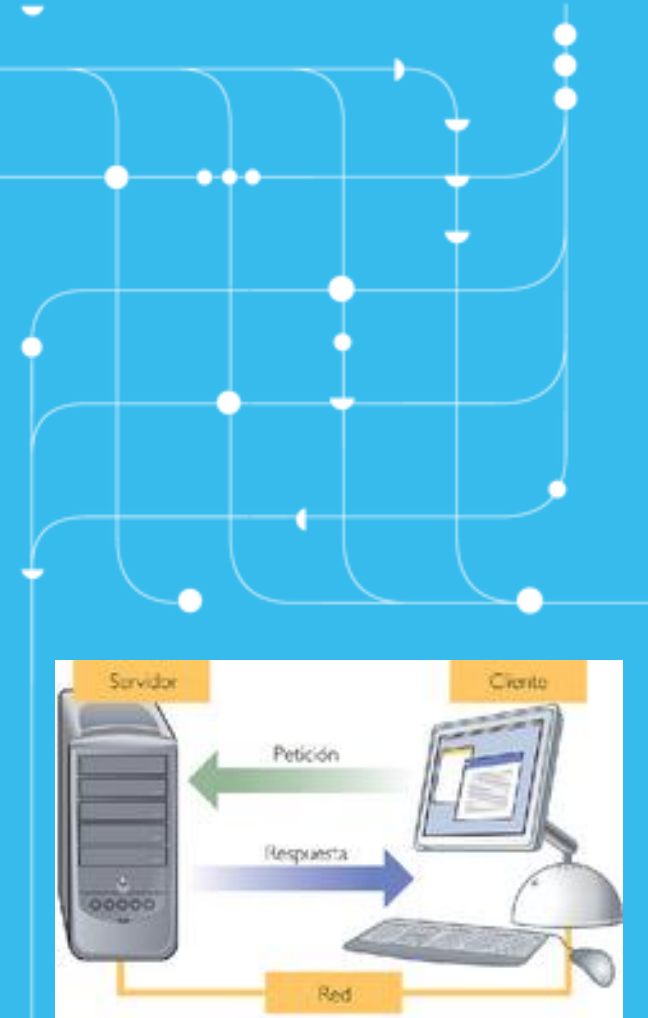


Ejemplos modelo cliente-servidor

Servidor de FTP:

Aplicación muy común es la transferencia de datos entre un cliente y un servidor web mediante File Transfer Protocols (FTP).

Este protocolo permite subir y bajar archivos.



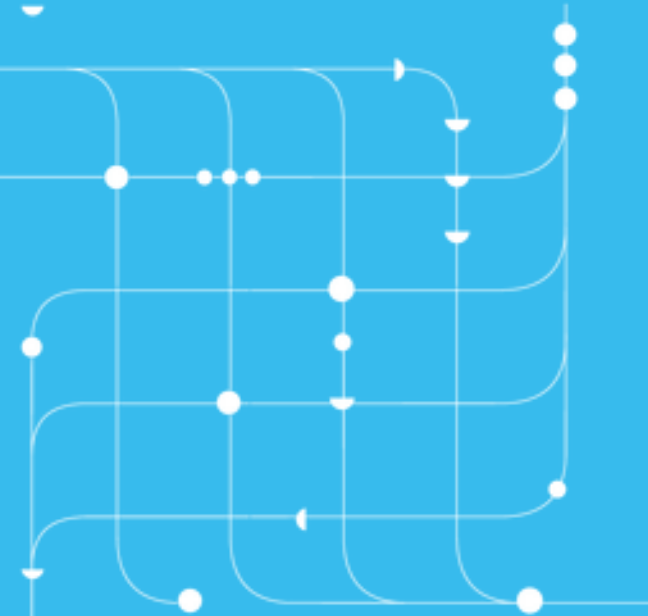
Algo sobre Protocolos

Que entendemos por Protocolo?

En informática es un conjunto formal de estándares y normas.

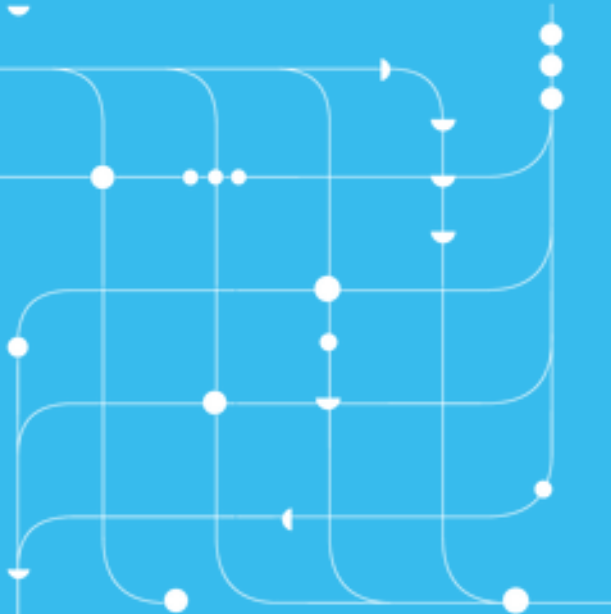
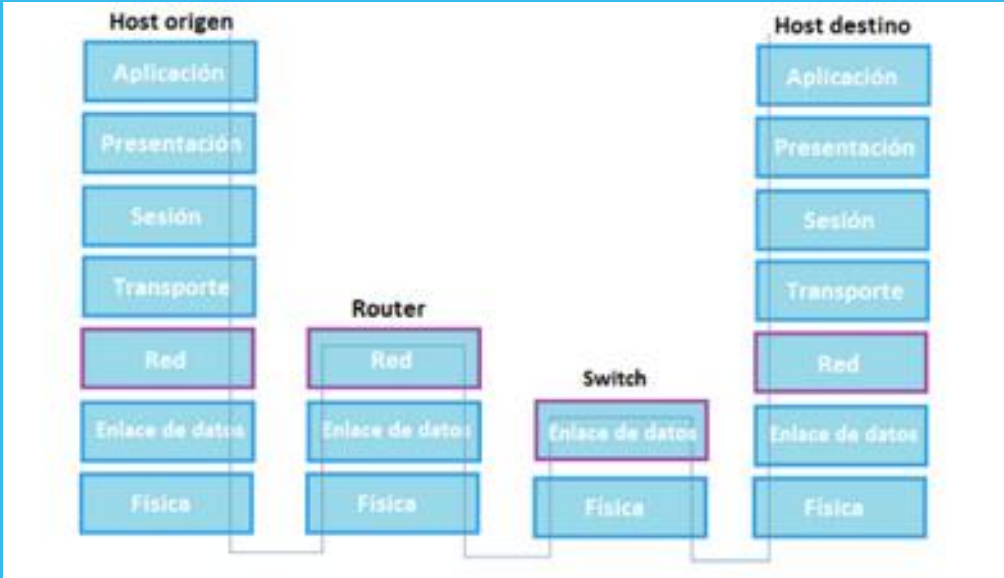
Estos rigen tanto el formato como el control de la interacción entre los distintos dispositivos dentro de una red o sistema de comunicación.

Los protocolos son como un lenguaje franco que los ordenadores pueden utilizar, para conversar de forma similar a como dos personas. Que previamente se ponen de acuerdo.



Algo sobre Protocolos

Modelo OSI:



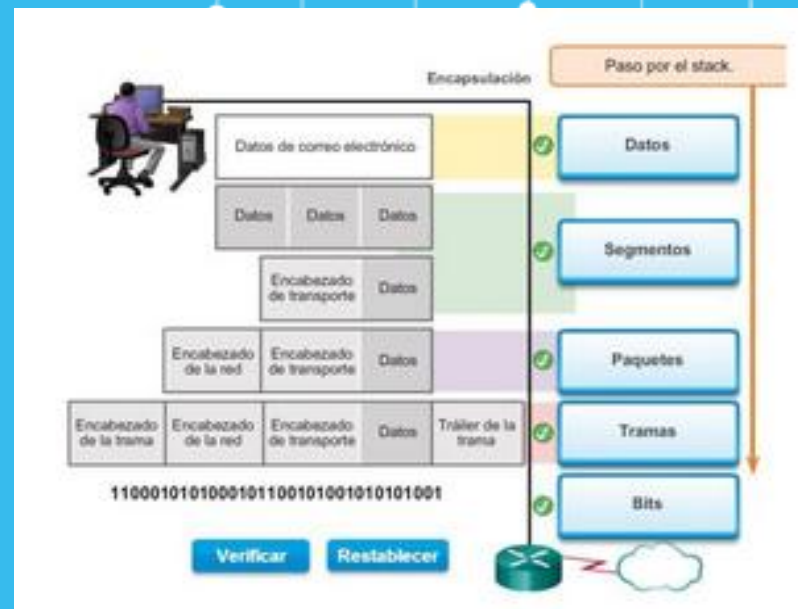
Algo sobre Protocolos

OSI vs TCP/IP:

Modelo TCP/IP

4	Aplicación	Aplicación 7	Presentación 6	Sesión 5
3	Transporte	Transporte 4	Red 3	Enlace de Datos 2
2	Internet	Enlace de Datos 2	Física 1	
1	Enlace			

Modelo OSI



Algo sobre Protocolos

Algunos Protocolos x capa:



Algo sobre Protocolos

**En nuestro caso para las páginas de web
Utilizamos:**

HTTP

(Protocolo de Transferencia de
Hipertexto).



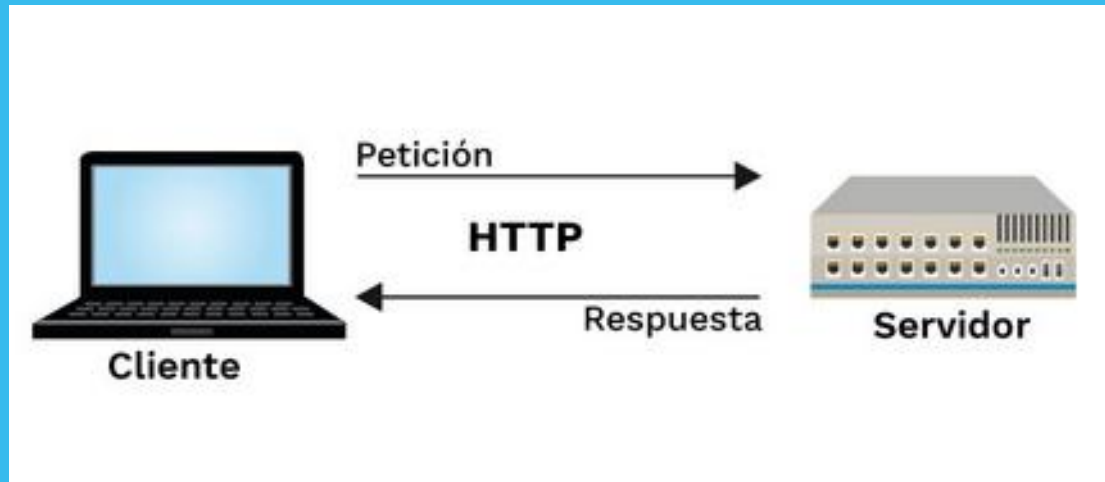
HTTPS

(Protocolo seguro de transferencia de
hipertexto)



Peticiones del Cliente al Servidor

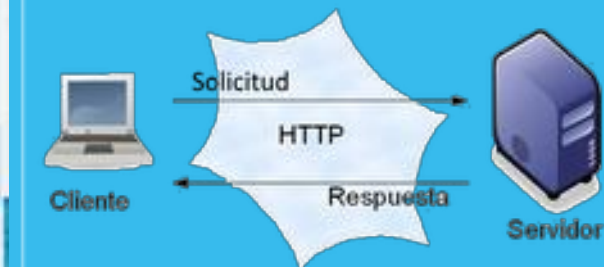
Los Métodos: Métodos de petición para indicar la acción que se desea realizar para un recurso determinado. Aunque estos también pueden ser sustantivos, estos métodos de solicitud a veces son llamados HTTP verbs.



Peticiones del Cliente al Servidor

Los Métodos:

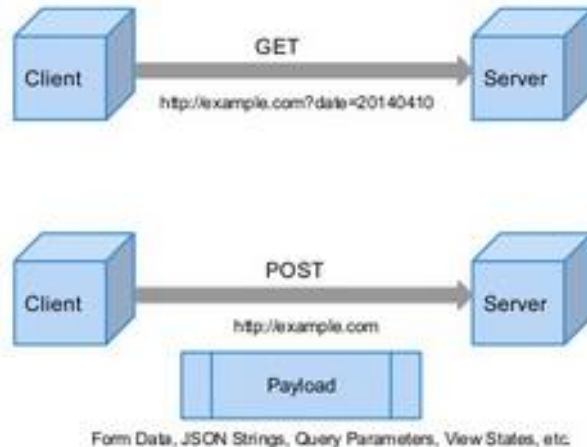
MÉTODOS HTTP	
Método HTTP	Significado en Restful Web Services
GET	Se utiliza para operaciones de sólo lectura. No generan ningún cambio en el servidor.
DELETE	Elimina un recurso en específico. Ejecutar esta operación múltiples veces no tiene ningún efecto.
POST	Cambia la información de un recurso en el servidor. Puede o no regresar información.
PUT	Almacena información de un recurso en particular. Ejecutar esta operación múltiples veces no tiene efecto, ya que se está almacenando la misma información sobre el recurso.
HEAD	Regresa solo el código de respuesta y cualquier encabezado HTTP asociado con la respuesta.
OPTIONS	Representa las opciones disponible para establecer la comunicación en el proceso de petición/respuesta de una URI.



Peticiones del Cliente al Servidor

Los Métodos: (Existen varios solo analizaremos 2 por ahora)

GET vs. POST



Peticiones del Cliente al Servidor

Los Métodos: (GET)

Petición

Verbo Recurso Versión

```
GET /index.html HTTP/1.1
Host: wikipedia.org
Accept: text/html
```

Primera línea

Encabezados

Cuerpo

Respuesta

Versión Código respuesta

```
HTTP/1.1 200 OK
Server: wikipedia.org
Content-Type: text/html
Content-Lenght: 2026
```

```
<html>
...
</html>
```

Como se comunican los Dispositivos?

Señales: Codificación y decodificación.

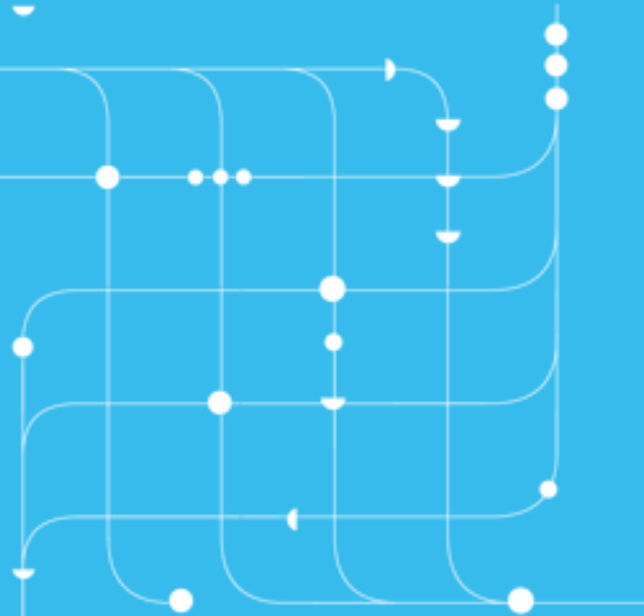
DIRECCIONES MAC: Dirección física del Dispositivo.

Direccionamiento Plano

Ejemplos: 3c:95:09:05:f9:4b; "00:00:00:00:00:00" –
"FF:FF:FF:FF:FF:FF"

DIRECCIONES IP: Internet Protocol.

Ejemplos: IPV4 200.0.182.2;
IPV6 2800:3f0:4002:80a::200e



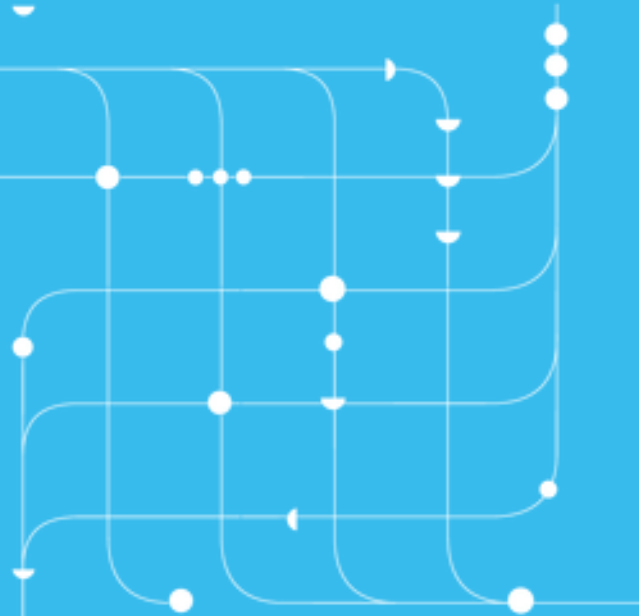
Resolución direcciones x Nombres

Problema: Utilizamos nombres para no recordar números...

¿En sus aplicaciones se manejan por nombres o por números?

Imaginen el **whatsapp** sin nombres de contactos...

No confundir : HOST – DOMAIN – SUBDOMAIN
– URL –



Hosts- Dominio -SubDominio -URL

Host: Es un dispositivo conectado a una red su nombre es el Nombre del Host. Esta asociado a una dirección IP. Compuesta por subdominio.dominio

Nombre de dominio: Es la parte personalizada y reconocible del dominio que identifica a una entidad específica en Internet. Por ejemplo, en "gidi.com", "gidi" es el nombre de dominio.

URL: (Uniform Resource Locator) y también se conoce como dirección de Internet. Suele referirse a una ruta específica o a un directorio de un sitio web.



Estructura y componentes de una URL

The diagram shows the URL `http://www.youtube.com/watch?v=QhcwLyyEjOA` with its components numbered 1 through 7. The components are: 1. Protocol (`http://`), 2. Host (`www.youtube.com`), 3. Subdomain (`www.`), 4. Domain (`youtube.com`), 5. Top-Level-Domain (`.com`), 6. Path (`/watch`), 7. Query parameter (`?v=QhcwLyyEjOA`).

Estructura y componentes de la URL: `http://www.youtube.com/watch?v=QhcwLyyEjOA`

- 1- protocolo utilizado, en este caso HTTP (Hypertext Transfer Protocol). También existen otros tipos de protocolos HTTPS, FTP, etc.
- 2- host o hostname: `www.youtube.com`
- 3- subdominio: `www.`
- 4- Nombre del dominio (dominio): `youtube.com`
- 5- Top-Level-Domain (terminación de la dirección web): `.com` (abreviado como TLD)
- 6- Ruta: `/watch`. Una ruta se refiere principalmente a un archivo o una carpeta (directorio).
- 7- Parámetro y valor: `v` (parámetro), `QhcwLyyEjOA` (valor del parámetro). Los parámetros se introducen mediante el carácter “?” dentro de una URL.

HOSTS

Resolver un Nombre a una dirección IP

Una Forma básica: utilizando una tabla de conversión HOSTS

En linux /etc/hosts

En windows → c:\windows\hosts → (95, 98, Me)

En windows → c:\windows\system32\drivers\etc\hosts

La Tabla es del tipo:

Comando host

```
127.0.0.1    localhost
127.0.1.1    carlos-Nitro-AN515-51
127.0.0.1    carlos.gestion
10.0.23.5     esteban
10.0.23.6     leandro
10.0.23.3     impresora_ginfo
200.0.182.70  ingenieria
200.0.182.71  economicas
200.0.182.72  psicologia
200.0.182.73  arquitectura
200.0.182.74  salud
200.0.182.75  humanidades
200.0.182.76  exactas
200.0.182.77  derecho
200.0.182.78  agrarias
200.0.182.79  medicina
127.0.0.1    niproeba.localhost
127.0.0.1    borrame1.localhost
```

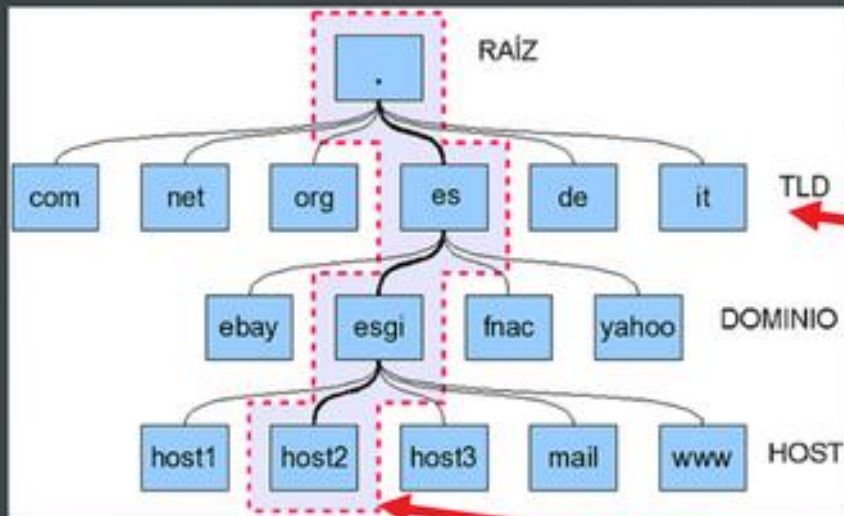
DNS

Nombres de Dominio:

- 1- **Concepto:** Un Nombre de dominio define un área de la red “internet” definida por nombres... **NO ES GEOGRAFICA!!!** como las direcciones o bloques de IP (publicas).
- 2- **Su Formato:** Etiquetas separadas por puntos. La Jerarquia superior se encuentra al extremo derecho. Ejemplo: `www.fi.mdp.edu.ar` ?
- 3- **Longitud:** Cada etiqueta puede tener hasta 63 caracteres. Pero todo debe ser menor igual a 255.
- 4- **Niveles:** Puede contemplar hasta 127. Lógico no?
5. **Tipo de dato:** Solo Alfanuméricos y el signo “-”, este nunca al inicio.
- 5- **Organización:** Un árbol – Las etiquetas concatenan el camino Raiz-Nodo. Ejemplo: `host.sub2.sub1.domain`

DNS

“La estructura de DNS se asemeja a la estructura jerárquica de los sistemas de archivos de UNIX, la cual se representa con un árbol invertido”



Top Level Domain

host2.esgi.es

DNS

Dominios Principales:



¿Que Falta...?

- **com.** Organizaciones comerciales como **sun.com**, **sybase.com**.
 - **edu.** Instituciones educativas como **mit.edu**, **berkeley.edu**
 - **gov.** Agencias gubernamentales como **nasa.gov**
 - **mil.** Organizaciones militares como **navy.mil**
 - **net.** Organizaciones relacionadas con la red como **freshmeat.net**
 - **org.** Organizaciones que no entran en ninguna de las categorías anteriores
-
- **film.** Negocios o filmaciones.
 - **store.** Negocios que ofrecen bienes.
 - **web.** Organizaciones que hace énfasis en el WEB.
 - **arts.** Organizaciones culturales y de entretenimiento.
 - **rec.** Organizaciones recreacionales y de entretenimiento.
 - **info.** Entidades que proveen servicios de información.
 - **nom.** Individuos u organizaciones que desean definir una nomenclatura personal.

Registrar un DNS

Registración de Nombres de Dominios:

Qué es ICANN

La Corporación para la Asignación de Nombres y Números de Internet (ICANN). <https://www.icann.org/>

Responsable de la administración y coordinación del Sistema de nombres de dominio (DNS).

Objetivos y Funciones:

- 1.- Garantizar que cada **dirección** sea **única**.
- 2.- Permitir usuarios de Internet puedan encontrar todas las direcciones válidas.
- 3.- Supervisión de la **distribución** de **direcciones** IP y **nombres** de dominio únicos.
- 4.- Garantiza que cada nombre de dominio se **asocie** a la dirección IP **correcta**.



Registrar un DNS

ICANN : También es responsable de acreditar a los registradores de nombres de dominio.



"Acreditar" significa:

Identificar y establecer estándares mínimos para la ejecución de las funciones de registro.

Reconocer a personas físicas o jurídicas que cumplan con esos estándares.

Celebrar un acuerdo de acreditación que estipule las normas y los procedimientos aplicables

Registrar un DNS

Registración de Nombres de Dominios:

InterNic : Información pública sobre los servicios de registro de nombres de dominio de Internet. <https://www.internic.net>

Entre otros tiene:

- Tiene una queja o disputa

- Información sobre registradores

- Información sobre Whois

- Asignación de Numeros de IP (**IANA**)

InterNIC



Internet Assigned Numbers Authority



IANA: Autoridad de Números Asignados de Internet. Responsable de la coordinación global de la raíz DNS, el direccionamiento IP y otros recursos de protocolo de Internet. <https://www.iana.org>

Registro DNS en Argentina

Registración de Nombres de Dominios:



NicAr : NIC Argentina, la Dirección Nacional del Registro de Dominios de Internet. <https://nic.ar/>

Depende de la Secretaría Legal y Técnica de la Presidencia de la Nación.

Reglamentación: Reglamento para la Administración de Dominios de Internet en Argentina. Aprobación.

<https://nic.ar/es/nic-argentina/normativa>

Tramites Principales

Registrar.

Delegar

Transferencia

Disputa

Delegación de Dominio

gidi.com.ar			
<input type="checkbox"/> gidi.com.ar (G)	A	31.220.56.61	
<input type="checkbox"/> gidi.com.ar (G)	MX	20:alt1.aspmx.l.google.com	
<input type="checkbox"/> gidi.com.ar (G)	MX	20:alt2.aspmx.l.google.com	
<input type="checkbox"/> gidi.com.ar (G)	MX	10:aspmx.l.google.com	
<input type="checkbox"/> gidi.com.ar (G)	MX	20:aspmx2.googlemail.com	
<input type="checkbox"/> capitulo.gidi.com.ar (G)	A	31.220.56.61	
<input type="checkbox"/> google5a3b13bf229699db.gidi.com.ar (G)	CNAME	google.com	
<input type="checkbox"/> google6aa9d2e1b2e59892.gidi.com.ar (G)	CNAME	ghs.google.com	
<input type="checkbox"/> mail.gidi.com.ar (G)	CNAME	ghs.google.com	
<input type="checkbox"/> pablito.gidi.com.ar (G)	A	34.71.10.29	
<input type="checkbox"/> pablo.gidi.com.ar (G)	A	34.71.10.29	
<input type="checkbox"/> www.gidi.com.ar (G)	A	31.220.56.61	

Es importante tener un
Servidor de DNS

Resumiendo

1. ¿Qué es un *host*?

- Un *host* es cualquier dispositivo conectado a una red (por ejemplo: una computadora, un servidor web, un router).
- Tiene una dirección IP única.

Ejemplo:

`192.168.0.12` es la IP de un host dentro de una red local.

2. ¿Qué es un *dominio*?

- Es un nombre legible por humanos que apunta a un host en Internet.
- Está compuesto por niveles jerárquicos separados por puntos: `www.unmdp.edu.ar`.

Nivel	Nombre
TLD	<code>.ar</code>
2°	<code>edu</code>
3°	<code>unmdp</code>
Subdominio	<code>www</code>

Resumiendo...

3. ¿Qué es un subdominio?

- Es una extensión del dominio principal que puede apuntar a un servidor o recurso específico.

Ejemplo:

`campus.unmdp.edu.ar` → plataforma educativa.

`correo.unmdp.edu.ar` → servicio de correo institucional.

4. ¿Qué es una URL?

- Es la dirección completa para acceder a un recurso.

Ejemplo:

`https://campus.unmdp.edu.ar/cursos/seguridad`

Componente	Ejemplo
Protocolo	<code>https</code>
Subdominio	<code>campus</code>
Dominio	<code>unmdp.edu.ar</code>
Ruta	<code>/cursos/seguridad</code>

Resumiendo...

5. ¿Cómo se registra un dominio?

- Se elige un nombre no registrado y se compra en un **registrador** autorizado.
- En Argentina, por ejemplo, se usa nic.ar.

Pasos básicos:

1. Verificar disponibilidad.
2. Registrar el dominio con tus datos.
3. Asociar el dominio con servidores DNS.

Resumiendo...

6. ¿Qué es la delegación directa (forward)?

- Es la configuración en DNS que indica qué IP corresponde a un nombre.

Ejemplo:

```
dns
campus.unmdp.edu.ar. IN A 190.3.123.45
```

7. ¿Qué es la delegación reversa (PTR)?

- Es lo contrario: se usa para obtener el nombre de dominio asociado a una IP.

Ejemplo:

```
dns
45.123.3.190.in-addr.arpa. IN PTR campus.unmdp.edu.ar.
```

- Esto se configura en los **servidores de zona inversa**, que deben estar autorizados por el proveedor de IP.

Resumiendo...

📷 Analogía útil: dirección postal

- Dominio: Calle y número → `unmdp.edu.ar`
- Subdominio: Departamento → `campus.unmdp.edu.ar`
- URL: Persona o dependencia exacta → `/cursos/seguridad`
- IP: Coordenadas GPS (exactas pero no amigables)
- DNS: Guía telefónica que traduce nombres a coordenadas.

🔍 1. Resolver un dominio a una IP

```
bash
```

```
dig +short unmdp.edu.ar
```

📌 Resultado:

```
190.3.123.45
```

🌐 2. Ver todos los registros del dominio

```
bash
```

```
dig unmdp.edu.ar ANY
```

📌 Resultado:

```
less
```

```
;; ANSWER SECTION:
unmdp.edu.ar.      300 IN A      190.3.123.45
unmdp.edu.ar.      300 IN MX     10 mail.unmdp.edu.ar.
...
```

Resumiendo...

3. Ver el Name Server asignado (delegación)

```
bash
```

```
dig unmdp.edu.ar NS
```

Resultado:

```
yaml
```

```
unmdp.edu.ar. 3600 IN NS ns1.unmdp.edu.ar.  
unmdp.edu.ar. 3600 IN NS ns2.unmdp.edu.ar.
```

4. Ver el servidor de correo (MX)

```
bash
```

```
dig unmdp.edu.ar MX
```

Resultado:

```
yaml
```

```
unmdp.edu.ar. 3600 IN MX 10 mail.unmdp.edu.ar.
```

5. Buscar subdominio

```
bash
```

```
dig campus.unmdp.edu.ar A
```

Resultado:

```
css
```

```
campus.unmdp.edu.ar. 300 IN A 190.3.123.46
```

6. Hacer una consulta inversa (PTR)

```
bash
```

```
dig -x 190.3.123.45
```

Resultado:

```
45.123.3.190.in-addr.arpa. 86400 IN PTR campus.unmdp.edu.ar.
```

Resumiendo...

7. Probar con `host` o `nslookup`

```
bash
```

```
host unmdp.edu.ar
```

Resultado:

```
csharp
```

```
unmdp.edu.ar has address 190.3.123.45  
unmdp.edu.ar mail is handled by 10 mail.unmdp.edu.ar.
```

```
bash
```

```
nslookup campus.unmdp.edu.ar
```

Resultado:

```
yaml
```

```
Server:      8.8.8.8  
Address:     8.8.8.8#53
```

```
Name:   campus.unmdp.edu.ar  
Address: 190.3.123.46
```


Resumiendo...

Zona directa (ejemplo de `db.unmdp.edu.ar`):

```
dns

$TTL 86400
@   IN  SOA ns1.unmdp.edu.ar. admin.unmdp.edu.ar. (
        2025041501 ; serial
        7200      ; refresh
        3600      ; retry
        1209600   ; expire
        86400    ) ; minimum
    IN  NS  ns1.unmdp.edu.ar.
    IN  NS  ns2.unmdp.edu.ar.
@   IN  A   190.3.123.45
campus IN A   190.3.123.46
mail   IN  A   190.3.123.47
       IN  MX 10 mail.unmdp.edu.ar.
```

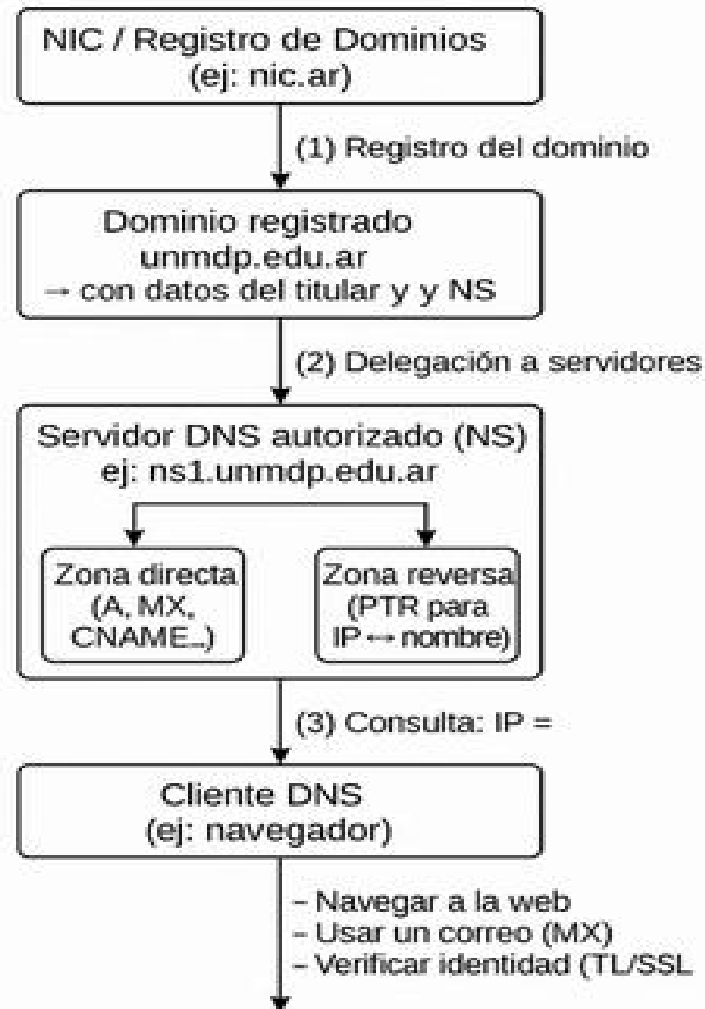
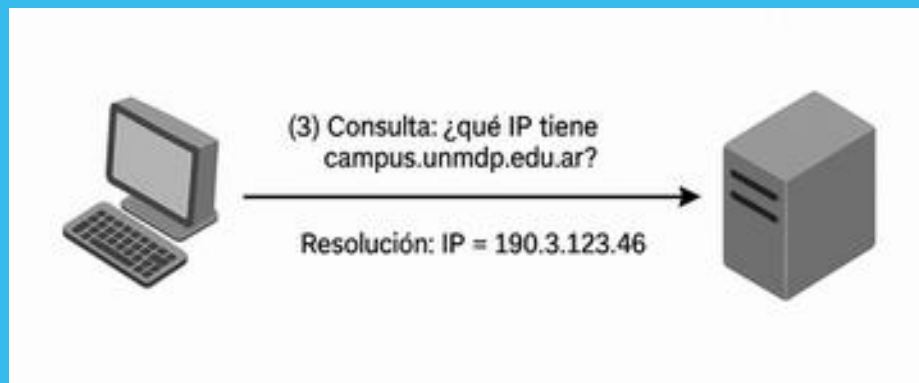
Zona reversa (ejemplo de `db.123.3.190.in-addr.arpa`):

```
dns

$TTL 86400
@   IN  SOA ns1.unmdp.edu.ar. admin.unmdp.edu.ar. (
        2025041501 ; serial
        7200      ; refresh
        3600      ; retry
        1209600   ; expire
        86400    ) ; minimum
    IN  NS  ns1.unmdp.edu.ar.
    IN  NS  ns2.unmdp.edu.ar.

45 IN  PTR www.unmdp.edu.ar.
46 IN  PTR campus.unmdp.edu.ar.
```

Resumiendo...



Herramientas para subir archivos a un hosting web

1. FTP/SFTP/SCP (protocolos de transferencia)

Protocolo	Puerto	Seguro	Descripción breve
FTP	21	✗	Rápido pero inseguro. En desuso para sitios productivos.
FTPS	21/990	✓	FTP sobre SSL/TLS. Agrega seguridad.
SFTP	22	✓	SSH File Transfer Protocol. Muy seguro.
SCP	22	✓	Copia directa vía SSH. Simple y rápida.

Herramientas para subir archivos a un hosting web

Herramientas gráficas

FileZilla (FTP/SFTP/FTPS)

- Muy usado, multiplataforma.
- Permite gestionar múltiples sitios.
- Interfaz de arrastrar y soltar.

WinSCP (Windows)

- Similar a FileZilla, pero exclusivo para Windows.
- Soporta SFTP, SCP y FTP.

Cyberduck (macOS/Windows)

- Interfaz amigable.
- Soporte para FTP, SFTP, WebDAV, Amazon S3, entre otros.

Herramientas para subir archivos a un hosting web

Herramientas por línea de comandos

ftp (comando clásico, no seguro)

```
bash
ftp ftp.ejemplo.com
```

sftp (seguro, basado en SSH)

```
bash
sftp usuario@ejemplo.com
# luego: put archivo.html
```

scp (copia rápida por SSH)

```
bash
scp archivo.html usuario@ejemplo.com:/var/www/html/
```

rsync (para sincronizar carpetas completas)

```
bash
rsync -avz ./sitio/ usuario@ejemplo.com:/var/www/html/
```

Herramientas para subir archivos a un hosting web

Alternativas modernas (para hosting en la nube)

Herramienta	Plataforma	Notas
Git + GitHub/GitLab	Varios	Usado en hosting tipo Netlify o Vercel
<code>curl -T archivo ftp://...</code>	Linux	Subida rápida sin cliente
Panel web (cPanel / Plesk)	Web	Interfaz visual para subir y administrar archivos

Recomendaciones

- Usá **SFTP** o **SCP** siempre que sea posible.
- Evitá FTP sin cifrado (muy inseguro).
- Para automatizar deploys: `rsync` + `cron` o usar `git push` en entornos modernos.

Manos A la Obra...

