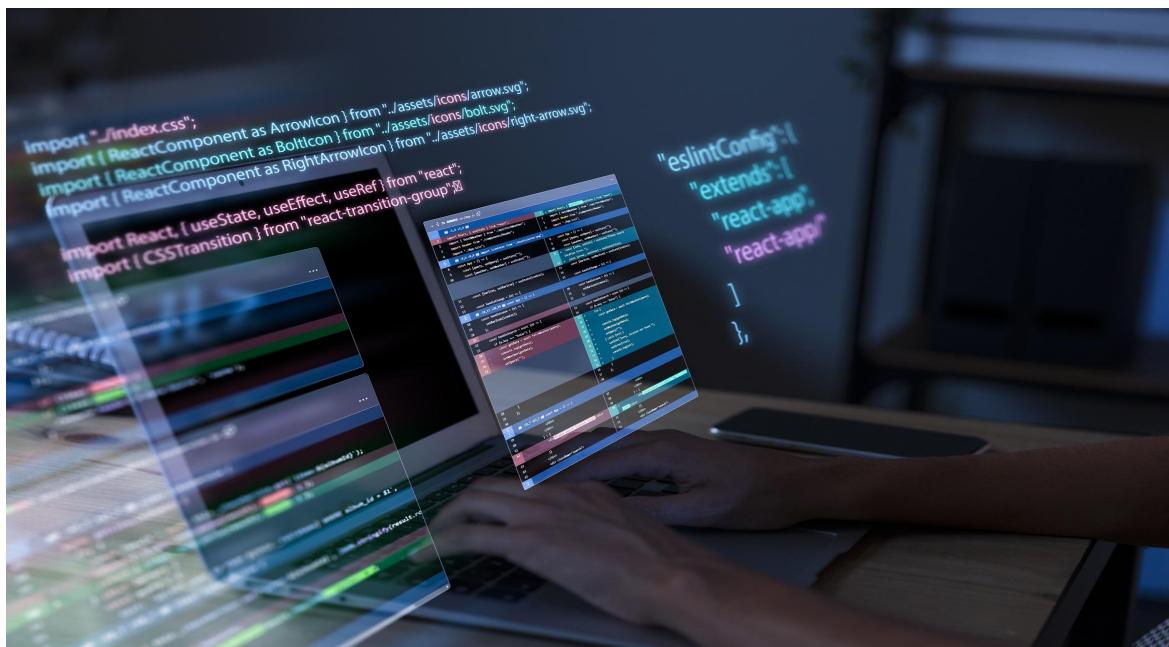


UNIDAD 4

Servicios de Red para el Despliegue



Autor: Luz María Álvarez Moreno

Fecha: 20/11/2025

SESIÓN 8 – DNS: registros A, CNAME, MX. Resolución de nombres El "libro de direcciones" de Internet

1. ¿Qué es el DNS?

El DNS es el **"libro de direcciones" de Internet**.

- Las personas recordamos mejor **nombres**: www.google.com, www.miempresa.es
- Los ordenadores solo entienden **direcciones IP**: 142.250.184.100, 185.42.24.10, etc.
- El DNS es el sistema que **traduce** un nombre de dominio a una IP (y viceversa, en algunos casos).

Sin DNS, tendríamos que escribir la IP de cada web en el navegador.
Impracticable.

2. Estructura del DNS

Plantea el DNS como un **árbol jerárquico**:

1. **Servidores raíz (root)**
 - Son la “cima” del árbol.
 - Saben dónde están los servidores de cada TLD (.com, .es, .org, ...).
2. **Servidores de dominio de nivel superior (TLD)**
 - Ejemplos: .com, .es, .org.
 - Cuando preguntas por miempresa.com, el root te manda a los servidores de .com.
3. **Servidores autoritativos del dominio**
 - Son los que “mandan” sobre miempresa.com.
 - Aquí están los registros DNS concretos: A, CNAME, MX, TXT, etc.
4. **Servidores recursivos (o resolvers)**

- Son los que consulta tu PC o tu móvil (los del proveedor de Internet, el router, etc.).
- Preguntan “hacia arriba” (root, TLD, autoritativo), **guardan la respuesta en caché** y te la devuelven.

3. Registros DNS esenciales: A, CNAME, MX

3.1. Registro A (Address record)

- **Qué hace:** Asocia **un nombre de dominio** a **una dirección IPv4**.
- Ejemplo típico:
 - miempresa.com --> 185.42.24.10
- Es el registro que “dice” dónde está físicamente tu web.

Ejemplo Web corporativa sencilla

Imagina que contratas un hosting:

- Tu proveedor te da una IP: 91.121.77.25.
- En el panel DNS del dominio miempresa.com creas:

miempresa.com. A 91.121.77.25
 www.miempresa.com. A 91.121.77.25

Ahora, tanto miempresa.com como www.miempresa.com apuntan a la misma máquina.

3.2. Registro CNAME (Canonical Name)

- **Qué hace:** Crea un **alias** hacia otro nombre de dominio.
- No apunta directamente a una IP, sino a **otro nombre**, que a su vez tendrá su registro A o AAAA.
- Sirve para **no duplicar configuraciones** y simplificar.

Ejemplo Alias para www y blog

Supón:

- Tienes la web principal en miempresa.com (con su registro A).

- Quieres que www.miempresa.com y blog.miempresa.com vayan al mismo sitio.

Puedes hacer:

miempresa.com. A 91.121.77.25
www.miempresa.com. CNAME miempresa.com.
blog.miempresa.com. CNAME miempresa.com.

Ventaja:

Si mañana cambias de servidor y la IP pasa a ser 203.0.113.40, solo modificas **uno**.

registro A:

miempresa.com. A 203.0.113.40
y automáticamente www y blog seguirán funcionando, porque son alias (CNAME) de miempresa.com.
Si un nombre tiene CNAME, **no** debe tener a la vez registros A o MX con el mismo nombre: eso rompe la resolución.

3.3. Registro MX (Mail Exchange)

- **Qué hace:** Indica **qué servidor se encarga del correo** del dominio.
- No apunta a una IP, sino a **un nombre de servidor**, que luego tendrá su A o CNAME.
- Incluye una **prioridad** (número): cuanto más bajo, más prioridad.

Ejemplo Correo corporativo

Dominio: empresa.com

Registros:

empresa.com. MX 10 mail.empresa.com.
mail.empresa.com. A 203.0.113.50

Significa:

- Para correos a info@empresa.com, los servidores de Internet enviarán el mail a mail.empresa.com.
- mail.empresa.com se resuelve a la IP 203.0.113.50.

Puedes tener varios servidores de correo:
empresa.com. MX 10 mail.empresa.com.
empresa.com. MX 20 backup.empresa.com.

Si el servidor con prioridad 10 cae, se usa el de prioridad 20.

4. Cómo se resuelve un nombre

Cuando escribes www.miempresa.com en el navegador, pasa algo así:

1. Tu navegador pregunta al **servidor DNS configurado en tu red** (normalmente, el del router o tu ISP).
2. Si ese DNS **ya tiene la respuesta en caché**, te la devuelve directamente.
3. Si **no la tiene**, sigue este camino:
 - o Pregunta a un **servidor raíz**: “¿Quién sabe de .com?”
 - o El root le devuelve los servidores de .com.
 - o Pregunta a los **servidores TLD .com**: “¿Quién es autoritativo para miempresa.com?”
 - o El TLD le responde: “Los DNS autoritativos son ns1.proveedor.com, ns2.proveedor.com...”
 - o Pregunta al **autoritativo**: “Dame el registro A de www.miempresa.com.”
 - o Éste devuelve la IP: por ejemplo, 91.121.77.25.
4. El servidor recursivo guarda esta respuesta en **caché** durante el tiempo marcado por el **TTL (Time To Live)**.
5. Te manda la IP y tu navegador conecta directamente a 91.121.77.25.

Todo esto en **milisegundos**.

5. Caso práctico: la pyme que se quedó sin correo (registro MX mal configurado)

Contexto:

- Una pequeña empresa española migra su web a otro hosting.
- Cambian los registros A para que la web apunte al nuevo servidor : la web funciona.
- Pero... **el correo deja de llegar y de enviarse**: presupuestos que no llegan, clientes que no reciben respuesta, etc.

¿Qué había pasado?

- El registro **MX seguía apuntando al servidor de correo antiguo**, que ya estaba apagado.
- Resultado: los correos enviados a @empresa.com se perdían.

Solución que aplicaron:

1. Revisan los registros usando **DNS Checker** y herramientas como dig empresa.com MX y nslookup -type=MX empresa.com.
2. Crean un **nuevo registro MX** hacia el servidor correcto:
3. empresa.com. MX 10 mail.empresa.com.
4. Verifican que mail.empresa.com tiene su correspondiente registro A al nuevo servidor.
5. Añaden registros TXT para **SPF** y **DKIM** (verificación y reputación del correo).
6. Esperan la **propagación DNS** (hasta 24 h aprox.).

Resultado:

- En menos de 24 horas, el correo vuelve a funcionar con normalidad.
- Lecciones que puedes remarcar en clase:
 - Siempre revisar **MX** al cambiar de hosting.
 - La propagación DNS puede tardar **24-48 horas**.
 - Los registros TXT (SPF, DKIM) son claves para que el correo no acabe en spam.

6. TTL y propagación

- Cada registro DNS tiene un **TTL (Time To Live)**.
- Es el tiempo que la respuesta se guarda en caché.

- Hasta que el TTL no expira, otros servidores pueden seguir “viendo” la información antigua.
- Por eso, cuando cambias una IP o un MX, **no se ve el cambio de inmediato** en todo el mundo.

Práctica

Entra en DNS Checker

1. Abre el navegador.
2. Escribe: **dnschecker.org**
3. Llegas a la página con:
 - Un campo para el dominio.
 - Un desplegable para elegir el tipo de registro.
 - Un botón **Search** o **Lookup**.

En el cuadro escribe un dominio conocido: www.google.es

En el desplegable selecciona: A – IPv4 address

Pulsa Search.

¿Por qué Google tiene tantas IPs?

Google tiene muchas IPs por varias razones, todas relacionadas con rendimiento, velocidad y disponibilidad.

- Porque tiene servidores en todo el mundo (infraestructura global)
 - Google no tiene un único servidor: tiene miles de centros de datos repartidos por el planeta.
 - Cada IP suele apuntar a un servidor o grupo de servidores en una región distinta.
- Porque usa Anycast

- Anycast significa que muchos servidores comparten la misma IP.
- Pero también significa que cada región puede tener su propio conjunto de IPs óptimas.
 - Desde España : te asigna una IP cercana.
 - Desde EEUU : otra distinta.
 - Desde Japón : otra distinta.
 - Todo para que cargue más rápido.
- Porque usa balanceo de carga (Load Balancing)
 - Google distribuye el tráfico entre cientos de máquinas.
 - Si solo tuviera una IP:
 - se saturaría
 - sería lento
 - sería vulnerable a ataques
 - Con muchas IP:
 - reparte millones de peticiones por segundo
 - evita colapsos
 - lo mantiene funcionando 24/7
- Porque usa redundancia extrema
 - Si una IP o un servidor cae:
 - entra otro
 - o varias IP nuevas
 - Por eso Google nunca se cae.
- Porque presta muchísimos servicios
 - No solo es google.com:
 - Gmail
 - Maps
 - YouTube
 - Google Photos

- Drive
 - Translate
- Cada uno tiene clusters diferentes que pueden devolver IPs distintas.

Google tiene muchas IPs porque necesita distribuir el tráfico mundial en miles de servidores, evitar saturaciones, ofrecer baja latencia y garantizar que su servicio nunca se caiga.

Cómo hacer la práctica de DNS con Cloudflare

Crear tu dominio gratuito DuckDNS

Ve a: <https://www.duckdns.org>

1. Entra con tu cuenta de Google o GitHub.
2. En la sección *Domains*, crea un dominio:
Ejemplo: **alumno.duckdns.org**

Este dominio será tu “campo de pruebas”.

Crear una cuenta en Cloudflare

1. Entra en <https://dash.cloudflare.com>
2. Crea una cuenta.
3. Pulsa “**Add a site**”.
4. Escribe tu dominio:
alumno.duckdns.org

Cloudflare detectará que es un **subdominio** y te permitirá **crear una zona DNS** solo para practicar.

No necesitas hosting ni pagar nada.

Crear los registros para la práctica

Ahora estamos en el panel DNS de Cloudflare.

Registrar un A

Haz clic en **Add record**:

- **Type:** A
- **Name:** @

- **Value:** 8.8.8.8 (prueba a poner tu ip)
- **TTL:** Auto

Resultado : tu dominio principal apunta a una IP.

Registrar un CNAME

Haz clic en **Add record**:

- **Type:** CNAME
- **Name:** www
- **Target:** alumno.duckdns.org
- **TTL:** Auto

Resultado : www.alumno.duckdns.org es un alias del dominio principal.

Registrar un MX

Haz clic en **Add record**:

- **Type:** MX
- **Name:** @
- **Mail server:** mail.alumno.duckdns.org
- **Priority:** 10

Ahora crea el registro A para el servidor de correo:

- **Type:** A
- **Name:** mail
- **Value:** 1.1.1.1 (por ejemplo)

Ya tienes:

- Registro A
- Registro CNAME
- Registro MX
- Servidor de correo simulado

Comprobar que funciona

Desde la web:

DNS Checker: <https://dnschecker.org>

Probamos con nuestro dominio de alumno.duckdns.org