

UD 5. SERVICIOS DE RED IMPLICADOS EN EL DESPLIEGUE DE UNA APLICACIÓN WEB.

1. SISTEMA DE NOMBRES

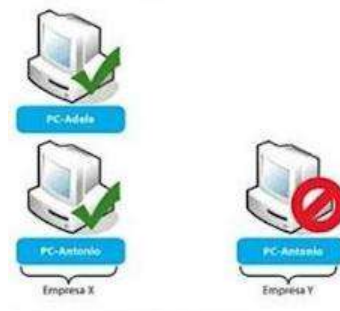
En una red TCP/IP, las máquinas se identifican mediante su dirección de red o número IP. Sin embargo es mucho más fácil recordar un nombre de dominio que una sucesión de números.

Si alguien nos pide que visitemos la página 216.642.10.28 a los pocos segundos ni la recordamos, sin embargo si nos solicita entrar en cocacola.com lo recordaremos mucho mejor.

Hace años en 1983 los servidores eran tan pocos que cada host tenía una lista de **nombres de dominio - ip** y la actualizaban diariamente descargándose la información actualizada de un servidor que centralizaba esta información. Esto era un **sistema de nombres plano**.

En un espacio de nombres planos, todos los nombres deben ser absolutamente únicos: no puede haber 2 máquinas con el mismo nombre.

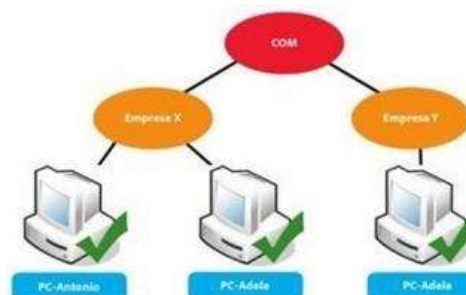
Para organizaciones grandes, esto no sirve, pues podría haber conflictos de nombres y sería imposible poder manejar los más de 50.000 nombres nuevos diarios.



SOLUCIÓN: SISTEMA DE NOMBRES JERÁRQUICO Y DISTRIBUIDO

El sistema de nombres jerárquico, tiene estructura de árbol de forma que cada nodo del árbol tiene un significado. Por ejemplo el espacio de nombres de los ficheros en disco (se pueden crear ficheros con el mismo nombre siempre que estén en otra carpeta)

disco\carpeta\subcarpeta\+...+nombre.



El sistema de nombres de dominio (Domain Name System) se encarga de esta gestión, traduce un nombre en una dirección IP, consultando en los servidores DNS.

Sistema de Nombres de Dominio (Domain Name System)

Desarrollado en noviembre de 1983 por Paul Mockapetris.

Se encarga de convertir direcciones formadas por cadenas de caracteres llamadas nombres DNS en direcciones IP, que serían más difíciles de recordar por los usuarios.

Podemos hacer referencia a los ordenadores a través de nombres DNS que lógicamente serán traducidos por “alguien” a direcciones IP que, al final, son lo único que entienden los sistemas TCP/IP. Ese alguien es un ordenador que ejecuta un programa servidor DNS y que, por tanto, se llama SERVIDOR DNS.

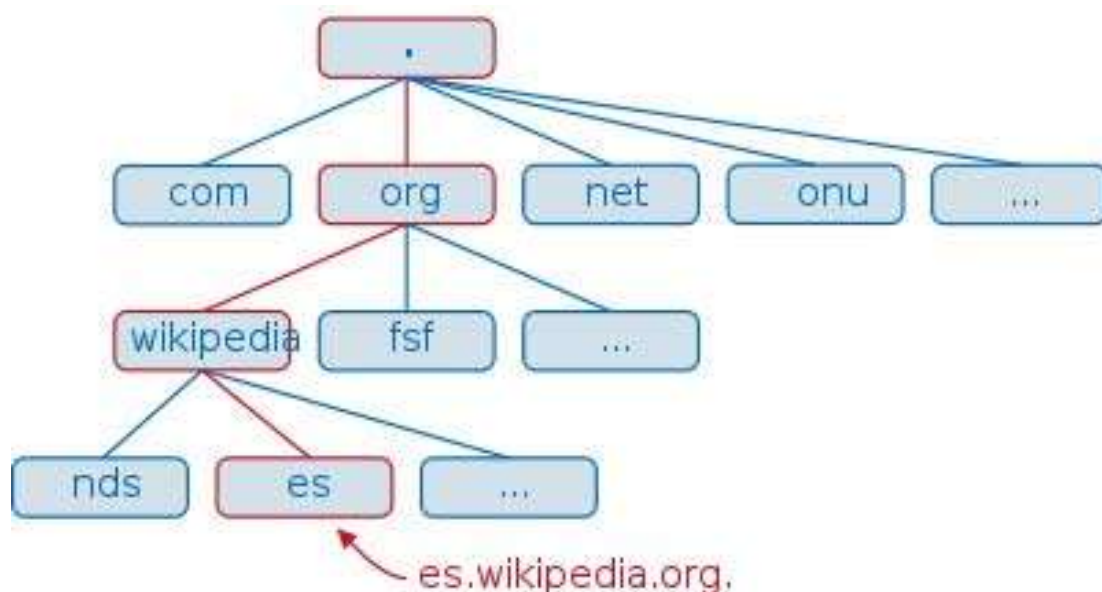
El **sistema de nombres de dominio** es un sistema de nombres jerárquico y distribuido. La estructura del sistema de nombres de dominio es una estructura en forma de árbol. Cada nodo del árbol se llama **nombre de dominio** y consiste en una etiqueta de hasta 63 caracteres.

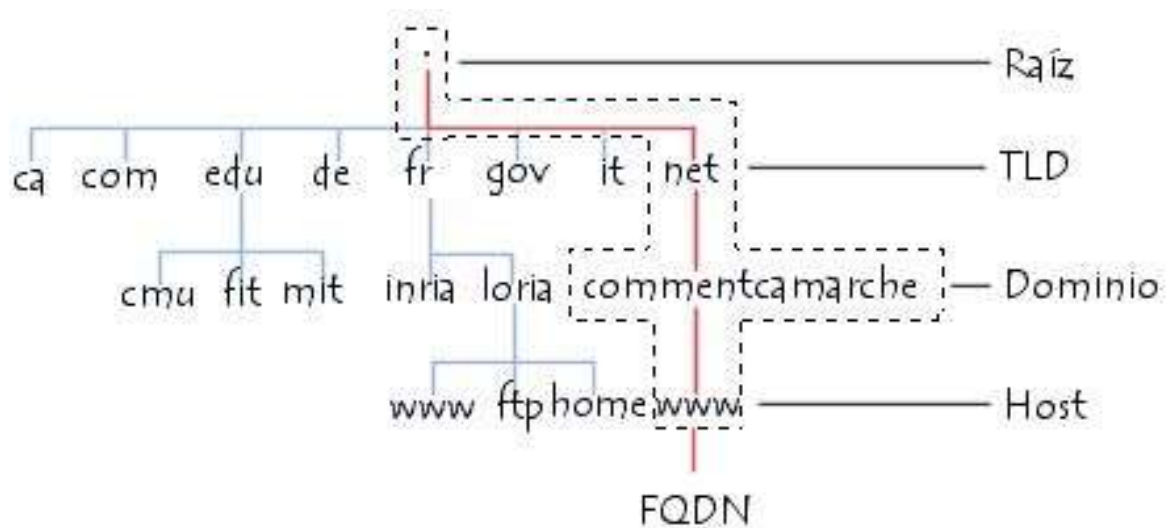
La raíz del árbol siempre es un PUNTO.

A la etiqueta de segundo nivel se le llama **dominio de nivel superior** o **Top Level Domain (TLD)**.

El extremo de la bifurcación se denomina **host** y corresponde a un equipo en la red.

Una **dirección FQDN** (*Nombre de Dominio totalmente calificado*) corresponde con todas las etiquetas de nodo de una estructura arbórea, separadas por puntos y que termina con un punto final. La profundidad máxima de una estructura arbórea es 127 niveles y la longitud máxima para un nombre FQDN es 255 caracteres.





La Corporación para la Asignación de Nombres y Números de Internet (ICANN) es responsable de la administración y coordinación del Sistema de nombres de dominio (DNS), a fin de garantizar que cada dirección sea única y que todos los usuarios de Internet puedan encontrar todas las direcciones válidas. Esto se logra mediante supervisión de la distribución de direcciones IP y nombres de dominio únicos. También garantiza que cada nombre de dominio se asocie a la dirección IP correcta.

ICANN: <http://www.icann.org>

ICANN opera el sitio web de InterNIC para brindar información pública sobre los servicios de registro de los nombres de dominio de Internet.

Inicialmente, los dominios superiores se han asignado a organizaciones y países. Estos nombres de dominio son los más habituales y siguen un estándar internacional. Para los países se usan abreviaturas de dos y de tres letras (es, it, de...), y se han reservado varias abreviaturas para que las usen las organizaciones, como se muestra en los siguientes ejemplos:

.com → comercial	.int → org. Internacional
.mil → militar	.org → org. no gubernamental
.edu → educación	.gov → institución gubernamental
.net → centros soporte de red	

Sin embargo en los últimos años se han gestionado otra serie de dominios superiores, dominios comerciales, como .google, .tienda, .barcelona, .madrid, .gal. O dominios patrocinados como .jobs, .mil, .museum, etc. Puedes ver un listado de dominios superiores en la url:

<https://www.icann.org/resources/pages/listing-2012-02-25-en>

2. RESOLUTORES DE NOMBRES.

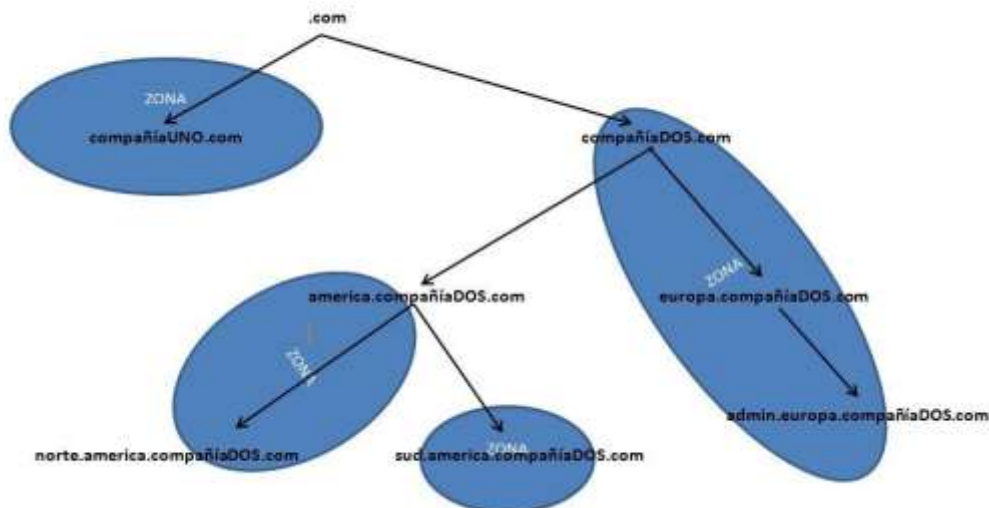
Cada nodo del árbol de una base de datos DNS, junto con todos los nodos por debajo del mismo, se llama dominio.

Los dominios pueden contener host (equipos) y otros dominios (subdominios).

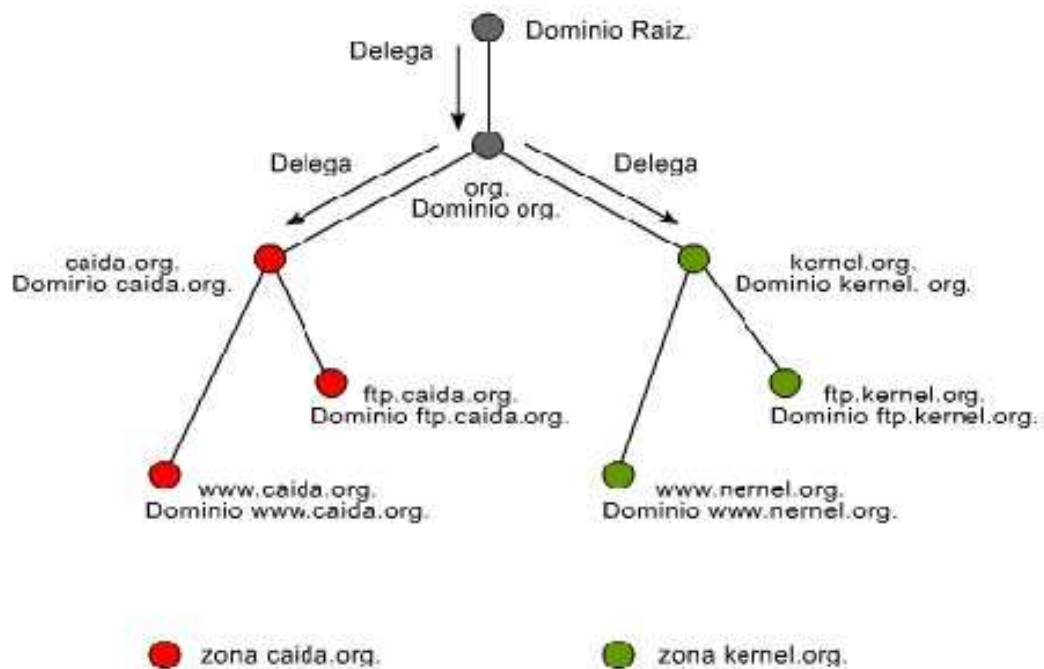
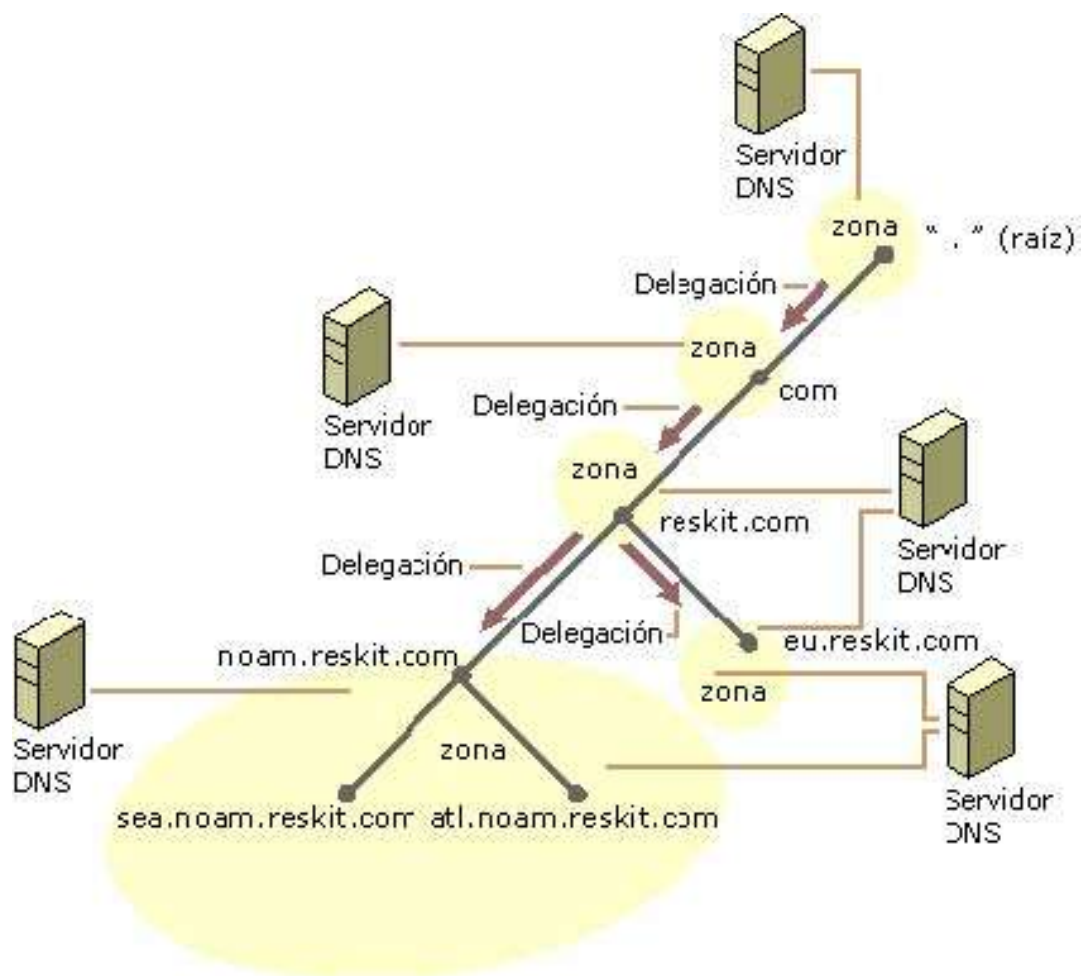
Por ejemplo, el dominio Midominio (midominio.com), podría contener a la vez equipos, como host.midominio.com, y subdominios, como subdom.midominio.com, que a su vez podría contener host, como por ejemplo host.subdom.midominio.com.

Existen equipos que actúan como **servidores de nombres de dominio** que permiten establecer la relación entre los nombres de dominio y las direcciones IP de los equipos de una red.

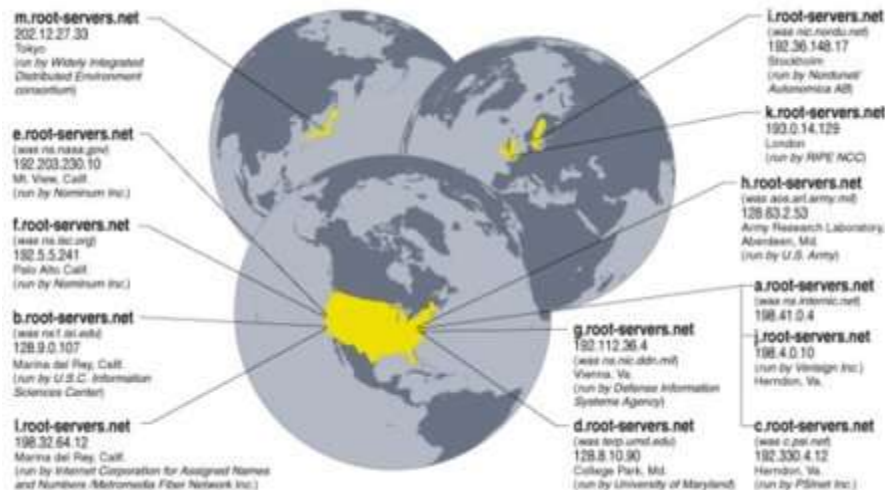
El sistema de servidores de nombres de dominio es una **arquitectura jerárquica y distribuida**, y permite su administración descentralizada mediante la **delegación de dominios**. Es decir como no habría un ordenador tan potente como para traducir todos los nombres de todos los ordenadores de Internet, lo que se hizo es que existen varios servidores DNS que trabajan de forma jerárquica y delegándose tareas unos en otros. Cada servidor DNS almacena información acerca de algunas partes del espacio de nombres del dominio. Cada una de esas partes se llama **zona**. Se dice el servidor DNS tiene **autoridad sobre una o varias zonas**.



Por otra parte cada servidor de nombre de dominio está especificado en el servidor de nombre de dominio en el nivel superior inmediato, lo que significa que la autoridad sobre los dominios puede delegarse implícitamente, a esto se le llama **delegación de dominios**, es decir un dominio puede ser dividido en subdominios delegando el control de otras **zonas** a otro servidor.



Los servidores relacionados con los dominios de nivel superior (TLD) se llaman "**servidores de dominio de nivel superior**". Son 13, están distribuidos por todo el mundo y sus nombres van desde "a.root-servers.net" hasta "m.root-servers.net".



3. CLASIFICACIÓN DE TIPOS DE SERVIDORES DNS

Hay 3 tipos de servidores DNS.

Servidores primarios (maestros)

La información de una zona la obtienen de sus archivos locales. Todas las modificaciones sobre una zona (añadir dominios, modificarlos, etc) se llevan a cabo en el servidor primario.

Servidores secundarios (esclavos)

Contiene una copia de sólo lectura de los archivos de zona de los servidores maestros con autoridad sobre esas zonas.

Cuando la información del servidor primario cambia, el esclavo la copia para actualizarse. Al proceso mediante el cual se obtiene información actualizada de la zona a través de internet se le llama "transferencia de zona".

Servidores caché

Cuando se les hace una consulta, si no tiene la respuesta, la reenvían a los servidores que puedan responder. Al recibir la respuesta la almacenan en caché para poder usarla en sucesivas peticiones.

Si se almacena la información en el servidor caché por mucho tiempo, la info puede estar desactualizada, pero si se almacena muy poco tiempo, mayor carga para el servidor de nombres. Hay que mirar lo que sea mejor para cada caso. Esto es el TTL o Time To Live (Tiempo de vida).

4. PROCESO DE RESOLUCIÓN DE UN NOMBRE DE DOMINIO

Cuando escribimos una dirección Web, el navegador debe resolver ese nombre en la dirección IP del servidor Web para establecer una conexión con ese servidor. Puede resolver el nombre de 2 formas:

Consulta a fichero de texto

Cuando un programa necesita resolver un nombre, normalmente, su primera parada suele ser el fichero HOSTS teniendo éste precedencia sobre DNS. En este fichero es donde comprueba si aparece o no una entrada para el nombre DNS especificado.

Este fichero contiene una línea por cada equipo que tenga asociado un nombre dns.

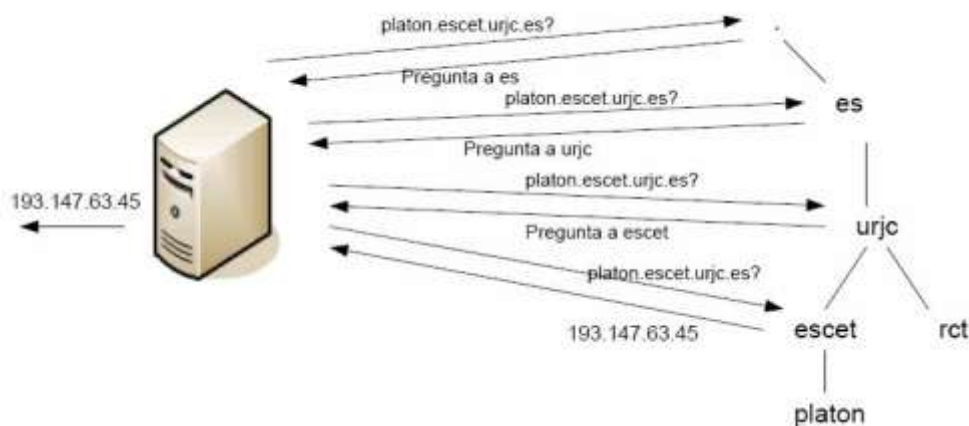
```
exemple# cat /etc/hosts
127.0.0.1 localhost
201.24.31.87 pc1.uu.vi.com pc1
201.24.31.105 pc2.uu.vi.com pc2
201.24.31.106 pc3.uu.vi.com pc3
```

Contacto con un servidor DNS

Cuando una aplicación (cliente) necesita resolver un FQDN envía un requerimiento al servidor de nombres configurado en el sistema (normalmente, el provisto por el ISP). A partir de entonces se desencadena el proceso de resolución del nombre. Existen 2 mecanismos:

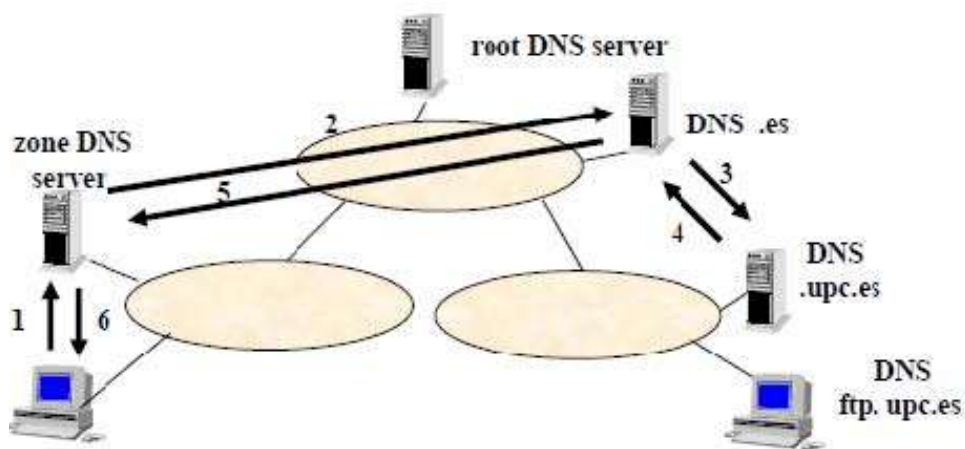
Resolución iterativa

1. El servidor de nombres inicial consulta a uno de los servidores raíz. (Actualmente existen 13 servidores raíz especificados, con los nombres de la forma letra.root-servers.net, donde letra va desde la A a la M.
2. Este devuelve el nombre del servidor a quien se le ha delegado la sub-zona.
3. El servidor inicial interroga al nuevo servidor.
4. El proceso se repite nuevamente a partir del punto 2 si es que se trata de una sub-zona delegada.
5. Al obtener el nombre del servidor con autoridad sobre la zona en cuestión, el servidor inicial lo interroga.
6. El servidor resuelve el nombre correspondiente, si este existe.
7. El servidor inicial informa al cliente el nombre resuelto.



Resolución recursiva

1. (1) - Host pregunta por ftp.upc.es al servidor DNS de su zona (dominio)
2. (2) - El servidor DNS de la zona pregunta al DNS server con dominio .es
3. (3) - El servidor DNS .es pregunta al servidor DNS con dominio .upc.es
4. (4) - El servidor DNS .upc.es le devuelve la @IP del servidor ftp.upc.es al dominio .es
5. (5) (6) - Se devuelve la @IP del servidor ftp.upc.es al cliente.



5. PARÁMETROS DE CONFIGURACIÓN Y REGISTROS DEL SERVIDOR DE NOMBRES AFECTADOS EN EL DESPLIEGUE.

Los archivos del DNS.

Un archivo de base de datos del servidor de nombres (o archivo DNS) es un 'archivo de zona' que contiene los registros (líneas del archivo) para la parte del dominio de la que es responsable la zona.

Registro	Función
A	Dado un nombre devuelve la IP correspondiente.
NS	Indica la dirección IP de un servidor DNS para el dominio dado. Pueden existir varios.
MX	Indica la dirección IP de un servidor de mail para el dominio dado. Pueden existir varios.
PTR	Indica el nombre correspondiente a una dirección IP dada. Sólo puede existir uno por dirección IP.
CNAME	Es un alias. Si un host tiene varios nombres se especifica uno con un registro A y el resto con CNAME, haciendo referencia al nombre en el registro A. Pueden existir varios.
SOA	Es un registro especial que provee información importante sobre la configuración de una zona en un DNS. Es básicamente utilizado para el funcionamiento interno del servicio. Sólo puede haber uno.
SPF	Es un registro especial cuya función principal es la de ayudar a combatir el SPAM. Lo que indica es qué servidores están autorizados para enviar correos para el dominio en cuestión.

Los principales tipos de registros son:

Registro SOA.

Proporciona información sobre la zona.

- host origen: El host en el que se mantiene el archivo.
- correo electrónico de contacto.
- número de serie: El "número de versión" de este archivo. 1440076612, timestamp de Unix. Cuenta los segundos transcurridos desde el 1 de Enero de 1970.
- tiempo de actualización: El tiempo que esperará un servidor secundario para ver si el archivo de base de datos ha cambiado y si hay que pedir una transferencia de zona al maestro.
- tiempo de reintento: El tiempo antes de volver a intentar una transferencia de zona fallida.
- tiempo de caducidad: El tiempo (en segundos) que un servidor secundario seguirá intentando descargar una zona.
- tiempo de vida (TTL): El tiempo (en segundos) que un servidor DNS tiene permitido acumular en la caché cualquier registro.

Registro NS

Indica los servidores de DNS autorizados (principales y secundarios) para el dominio y significa **Name Server**, y funciona como delegación de DNS de dominio (host).

Permite así que otros servidores vean los nombres de tu dominio.

Normalmente, se configura para subdominios; has de tener en cuenta que la zona de los DNS del Name Server (NS) que has asignado también estén correctamente configurados para evitar errores.

dominio IN NS nombreservidorhost

Ejemplo:

```
telefonica.net. IN NS dns2.terra.es.  
telefonica.net. IN NS dns1.terra.es.
```

El Registro CNAME

Estos registros también reciben el nombre de alias, aunque son conocidos como entradas de "nombre canónico". La utilidad principal de los mismos es la de usar más de un nombre para apuntar a un único host. Esto puede simplificar operaciones como albergar a la vez un servidor FTP y un servidor web en el mismo equipo. Su formato es el siguiente:

```
nombrealiashost IN CNAME nombrehost
```

Ejemplo: Suponga que `www.midominio.com` y `ftp.midominio.com` se encuentran en el mismo equipo. Si éste es el caso, entonces podría tener las siguientes entradas en su archivo de zona:

```
servidorarchivos IN A 157.55.200.41  
ftp IN CNAME servidorarchivos  
www IN CNAME servidorarchivos
```

Registro A

Un Registro de dirección A (Address) sirve para asociar nombres de host a direcciones IP dentro de una zona. Éstos son los registros que componen la mayor parte del archivo de base de datos. Su formato es el siguiente `nombrehost IN A direcciónIPdehost`

Ejemplo:

```
machine1 IN A 157.55.201.143  
nombreservidor2 IN A 157.55.200.2
```

REGISTROS PTR

Los registros 'PTR' o apuntadores, se usan para la resolución inversa. También conocido como 'registro inverso', funciona a la inversa del registro A, traduciendo IPs en nombres de dominio.

Por ejemplo:

```
5.1.168.192.in-addr.arpa. IN PTR pc01.daw1.com.  
6.1.168.192.in-addr.arpa. IN PTR pc02.daw2.com.
```

Ejemplo `$ dig -x 127.0.0.1`

```
;; ANSWER SECTION: 1.0.0.127.in-addr.arpa. 0 IN PTR localhost.  
$ dig -x 213.4.130.95
```

Registro MX (Mail eXchange record)

El Registro de Intercambio de Correo indica qué host procesa el correo de este dominio. Si existen múltiples registros de intercambio de correo, el resolutor de nombres intentará ponerse en contacto con los servidores de correo en orden de preferencia, empezando por los valores inferiores (mayor prioridad) hasta el valor superior (menor prioridad). Su formato es el siguiente:

```
dominio IN MX preferencia servidorcorreohost
```

Ejemplo:

```
netstorming.com.ar.      7200      IN      MX      5 netstorming.com.ar.  
netstorming.com.ar.      7200      IN      MX      0 mail.netstorming.com.ar.
```

El resultado que se ve es que NetStorming tiene dos servidores de mail que son `mail.netstorming.com.ar` y `netstorming.com.ar`. El primero de ellos tiene mayor prioridad (de allí el 0).

EJEMPLO

```
redes.com.          IN SOA  ns.redes.com.
                    admin-redes.gmail.com. (
                    2007030702 ; Número de serie
                    86400      ; Refresco
                    7200       ; Reintento
                    2592000    ; Expiración
                    172800 )   ; Ttl

redes.com.          IN NS   ns.redes.com.
redes.com.          IN MX   mailhost.redes.com.
ns.redes.com.       IN A    193.147.184.6
mercurio.redes.com. IN A    193.147.184.7
venus.redes.com.    IN A    193.147.184.8
tierra.redes.com.   IN A    193.147.184.9
marte.redes.com.    IN A    193.147.184.9
www.redes.com.      IN CNAME mercurio.redes.com.
mailhost.redes.com. IN CNAME venus.redes.com.
gestion.redes.com.  IN NS   ns.gestion.redes.com.
sistemas.redes.com. IN NS   ns.sistemas.redes.com.
ns.gestion.redes.com. IN A    212.135.11.45
ns.sistemas.redes.com. IN A    212.146.13.145
```

6. INSTALACIÓN Y CONFIGURACIÓN DEL SERVIDOR DE NOMBRES AFECTADOS EN EL DESPLIEGUE.

Instalación del servidor BIND9 y configuración de zona

Prácticamente el único software utilizado en los servidores de nombres de Internet es `bind` ("Berkeley Internet Name Domain"), creado originalmente en la Universidad de California, y actualmente propiedad del Internet Systems Consortium.

BIND es el servidor de nombres de dominio más popular en Internet, que trabaja en todas las plataformas informáticas principales y se caracteriza por su flexibilidad y seguridad.

Al contar con un **DNS** propio (ya sea uno o varios servidores de nombres) es posible definir *zonas locales* (no válidas ni accesibles desde *Internet*) para asignar nombres a cada uno de los *hosts* de la **LAN**. De esta forma es posible, por ejemplo, referirnos a la impresora de red como "*hplaser.mired.local*" en vez de "*192.168.0.2*" y a nuestro servidor de correo interno como "*smtp.mired.local*" en vez de "*192.168.0.3*".

```
sudo apt-get install bind9
```

Vamos a configurar el archivo `/etc/bind/named.conf.local` donde declararemos las zonas.

El dominio que tenemos delegado se va a llamar `despliegue.com`. El archivo debería quedar de la siguiente forma:

```
zone "despliegue.com" {  
    type master;  
    file "/etc/bind/db.despliegue";  
};
```

Explicación:

Línea 1: nombre de la zona

Línea 2: tipo de servidor: La línea `type master` significa que el dns que estamos configurando es primario. Podría ser secundario o esclavo (`slave`), o caché (`hint`).

Línea 3: hace referencia a la ruta del archivo donde se guardarán los registros de los nombres de dominio. Se recomienda que el fichero comience por `db` para hacer ver que se refiere a una base de datos.

Comprobamos que no hay ningún error de sintaxis. Para ello nos situamos en `/etc/bind` y hacemos:

```
named-checkconf
```

Si no pone nada es que está bien sintácticamente (no faltan `“;”`, `“}”` y cosas así)

El siguiente paso es modificar el archivo de configuración de DNS. En este paso se va a crear el fichero que va a almacenar la base de datos del servidor DNS, el nombre de dominio, la dirección IP, etc.

La ruta donde se va a crear es en este caso `/etc/bind/` y el fichero se va a llamar `db.despliegue` tal y como hemos indicado en el anterior apartado.

ATENCIÓN!! , a los puntos al final de cada nombre de dominio.

```
$TTL 38400
despliegue.com.      IN SOA      ns.despliegue.com.      admin.despliegue.com. (
    1317165042
    10800
    3600
    604800
    38400)
```

```
despliegue.com.  IN NS   ns.despliegue.com.
@                IN A    192.168.1.101
ns.despliegue.com.  IN A    192.168.1.101
pc01.despliegue.com.  IN A    192.168.1.2
pc02.despliegue.com.  IN A    192.168.1.6
www.despliegue.com.  IN CNAME ns.despliegue.com.
ftp.despliegue.com.  IN CNAME ns.despliegue.com.
```

Podemos comprobar la correcta sintaxis con la orden:

```
named-checkzone despliegue.com /etc/bind/db.despliegue
zone despliegue.com/IN: loaded serial 1317165042
OK
```

Donde el primer parámetro es el nombre de la zona y el segundo la ruta al archivo correspondiente.

Si todo ha ido bien reiniciamos el server

```
sudo /etc/init.d/bind9 restart
```

Ahora hay que configurar el dns a utilizar que será el nuestro. Configuramos en la interfaz de red nuestro equipo como dns. Para ello en /etc/network/interfaces, **dns-nameservers ip_servidor_dns** (en vez de editar el fichero /etc/resolv.conf) Ahora ya no necesitamos el fichero /etc/hosts para resolver direcciones.

Comprobamos si resuelve las direcciones con los comandos nslookup, dig.

Nota: Para poder utilizar estos comandos tendrás que instalar el paquete dnsutils.

Información del comando nslookup:

<http://eltallerdelbit.com/comando-nslookup-opciones>

Información del comando dig:

<http://rm-rf.es/como-usar-el-comando-dig-ejemplos/>

```
nslookup despliegue.com
nslookup ns.despliegue.com
nslookup www.despliegue.com

dig www.despliegue.com
; <<>> DiG 9.9.5-3-Ubuntu <<>> www.despliegue.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 48640
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.despliegue.com.  IN A
```

```
;; ANSWER SECTION:
www.despliegue.com. 38400 IN CNAME ns.despliegue.com.
ns.despliegue.com. 38400 IN A 192.168.1.101
```

```
;; AUTHORITY SECTION:
despliegue.com. 38400 IN NS ns.despliegue.com.
```

```
;; Query time: 60 msec
;; SERVER: 192.168.1.101#53(192.168.1.101)
;; WHEN: Sun Feb 08 13:36:58 CET 2015
;; MSG SIZE rcvd: 94
```

Por otro lado si queremos resolver nombres de fuera de nuestra red y reenviar las peticiones a otro dns tenemos que configurar los forwarders. (Si no se configuran también se resuelven nombres fuera de nuestra red preguntando por defecto a los servidores raíz)

Lo que debemos hacer ahora es abrir el archivo de opciones.

```
gedit /etc/bind/named.conf.options
```

en él debemos quitar los comentarios de la parte de forwarders y configurarlo con el dns del isp o los que se usen para fuera de nuestra red.

```
forwarders {
    8.8.8.8;
};
```

Si algo falla mirar el log de errores de la máquina, por ejemplo en: `/var/log/syslog`

Otros comandos:

```
dig @servidorDns nombreDominio (Pregunta a un servidor por defecto)
dig +trace nombre (Traza de servidores dns consultados en la resolución)
```

ejemplo: Usando el programa dig, podemos obtener la secuencia de servidores DNS que se sigue para resolver la dirección `www.microsoft.com`

```
dig +trace www.microsoft.com
```

Además:

```
dig -t soa dominio (pregunta sobre el registro soa de un dominio)
dig -t ns dominio (pregunta sobre el registro ns de un dominio)
```

Configuración de zona inversa

Para poder preguntar por el nombre de dominio a partir de la ip necesitamos configurar la resolución inversa.

Vamos a configurar el fichero named.conf.local para la resolución inversa, para ello hay que modificar el fichero para añadir (a continuación de la zona anterior) la zona inversa identificada como la IP de la red al revés y sin la parte del host acabada en .in-addr.arpa.

```
// Definición de la zona inversa para DNS.
zone "1.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/db.192.168.1.in-addr.arpa";
};
```

Podemos comprobar que la sintaxis sea correcta con la orden que ya vimos. Si es correcta no aparecerá ningún mensaje pero si no lo es nos mostrará el problema.

```
named-checkconf
```

El siguiente paso es crear el archivo de zona inversa.

```
gedit /etc/bind/db.192.168.1.in-addr.arpa
$ttl 38400
@ IN SOA ns.despliegue.com.      admin.despliegue.com. (
    1317165042
    10800
    3600
    604800
    38400)
@ IN NS  ns.despliegue.com.
101     IN     PTR      despliegue.com.
```

Y lo comprobamos

```
named-checkzone 1.168.192.in-addr.arpa /etc/bind/db.192.168.1.in-addr.arpa
```

La @ se podría haber sustituido por el nombre de la zona 1.168.192.in-addr.arpa.

Reiniciamos bind

```
/etc/init.d/bind9 restart
```

Debemos probar que la resolución inversa funciona correctamente.

```
root@debian:/etc/bind# dig -x 192.168.1.101
```

```
; <<>> DiG 9.9.5-3-Ubuntu <<>> -x 192.168.1.101
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 64829
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags;; udp: 4096
;; QUESTION SECTION:
;101.1.168.192.in-addr.arpa. IN PTR

;; ANSWER SECTION:
```


101.1.168.192.in-addr.arpa. 38400 IN PTR despliegue.com.

;; AUTHORITY SECTION:
1.168.192.in-addr.arpa. 38400 IN NS ns.despliegue.com.

;; ADDITIONAL SECTION:
ns.despliegue.com. 38400 IN A 192.168.1.101

;; Query time: 40 msec
;; SERVER: 192.168.1.101#53(192.168.1.101)
;; WHEN: Sun Feb 08 14:03:27 CET 2015
;; MSG SIZE rcvd: 116

EJEMPLOS DE CONFIGURACIÓN:

Configuración donde tenemos un dominio, practica.com y ..

Un servidor de dns ns.practica.com en la máquina 192.168.1.34

Un servidor web responderá en la máquina con IP 192.168.1.35, con la url: www.practica.com

Un servidor ftp que responderá en la misma máquina que el servidor web (192.168.1.35), con la url: ftp.practica.com

```
$TTL 604800
@ IN SOA practica.com. host.practica.com. (
2 ; Serial
604800 ; Refresh
86400 ; Retry
2419200 ; Expire
604800 ) ; Negative Cache TTL
;
@ IN NS ns.practica.com.
ns.practica.com. IN A 192.168.1.34
www IN A 192.168.1.35
ftp IN CNAME www.practica.com.
```

Resolución inversa, resolución de dominio que corresponde a una IP dada.

```
$ttl 38400
@ IN SOA ns.practica.com. host.practica.com(
3
10800
3600
604800
38400
)
@ IN NS ns.practica.com.
34 IN PTR practica.com.
35 IN PTR www.practica.com.
35 IN PTR ftp.practica.com.
```

Otro ejemplo de configuración, incluido dns esclavo:

http://www.ite.educacion.es/formacion/materiales/85/cd/linux/m2/servidor_dns_bind9.html

Información sobre la configuración de DNS en las máquinas cliente.

En Linux (comprobaciones con DEBIAN ó UBUNTU):

Si modificas el archivo `/etc/resolv.conf`, ten en cuenta que cada vez que reinicies el equipo ó una aplicación haga uso del mismo (cómo `dhclient`) los cambios se perdaran.

Así, podemos leer dentro del archivo:

```
Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
# DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
```

Para hacer los cambios permanentes:

Lo habitual es modificar el archivo `/etc/resolv.conf`, utilizando la directiva `nameserver` indicando la dirección IP de nuestro servidor de DNS.

`nameserver 192.168.2.1`, pero como ya hemos dicho antes, estos cambios no serán permanentes.

Puedes hacer que los cambios sean permanentes utilizando alguno de los siguientes métodos:

- 1) Modificar la configuración DHCP, en concreto el archivo `/etc/dhcp/dhclient.conf`

Y actualizando las directivas, `domain-name` y `domain-name-servers`, indicando el dominio y servidor de dominios.

Más información: <https://wiki.debian.org/resolv.conf>

- 2) Editando el archivo `/etc/resolvconf/resolv.conf.d/head`

Añadiendo:

```
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
# DO NOT EDIT THIS FILE BY HAND --
# YOUR CHANGES WILL BE OVERWRITTEN
nameserver 8.8.8.8
nameserver 8.8.4.4
```

En Windows (Windows 10):

<https://www.xataka.com/basics/como-cambiar-o-configurar-los-dns-en-windows-10>

En todos los sistemas operativos Windows la idea es la misma, configurar Protocolo de Internet Versión 4 (TCP/IPv4).