DNS

Servicio de Nombres de Dominio

Fernando Raya, Carlos Martín

Versión 0.8, 2022-10: IES Zaidín Vergeles

Tabla de Contenido

Capítulo 1. DNS

El servicio de resolución de nombres usado en las redes TCP/IP es el servicio *DNS Domain Name System* o Sistema de Nombres de Dominio. Este servicio permite identificar de una forma más sencilla a un equipo mediante un nombre, en lugar de usar la identificación numérica de la dirección IP.

1.1. Características

En las redes TCP/IP, como es Internet, no es fácil recordar las direcciones IP de los equipos. Es mucho más cómodo usar y recordar nombres que secuencias de números.

Para facilitar el uso de los servicios, recursos y equipos de una red se creó un sistema de nombres que mediante un servicio de resolución de nombres permite asociar nombres con direcciones numéricas.

Simplificadamente un servicio de nombres almacena direcciones y sus nombres correspondientes.

La operación que hay que realizar para conocer la dirección IP de un equipo a través de su nombre se denomina **resolución de nombre**.

1.1.1. Sistemas de nombre planos y jerárquicos

Los sistemas de nombres se clasifican en planos y jerárquicos.

Sistemas de nombres planos

Los nombres se usan sin ninguna estructura u organización que permita clasificarlos de alguna manera.

Ejemplo 1. Matrículas de alumnos

Por ejemplo, los números de matrícula de los alumnos de un centro y su correspondencia con los nombres de dichos alumnos. (Los nombres NetBIOS que usa Windows para identificar equipos son un espacio de nombres plano). El mayor inconveniente es que los nombres deben ser únicos, sin repetición.

Sistemas de nombres jerárquicos

Los sistemas de nombres jerárquicos usan nombres que se agrupan o clasifican bajo algún criterio. Esta cualidad permite una gestión más simple, permitiendo que los nombres se puedan repetir y con la posibilidad de realizar una gestión de forma distribuida.

Ejemplo 2. Sistema de archivos

Los nombres en un sistema de archivos y también como veremos, el servicio de resolución de

nombres DNS.

1.2. Servicio DNS

El servicio DNS es un servicio de registro y consulta de información, que se almacena en una base de datos distribuida en numerosos equipos.

Los equipos que almacenan una parte de dicha información se denominan **servidores de nombres**.

La información a la hora de distribuirla entre los diversos servidores de nombres se organiza mediante un esquema de nombres jerárquico. Este esquema jerárquico es lo que se denomina **espacio de nombres de dominio**.

Así cada servidor de nombre gestiona sólo una parte del espacio de nombres de dominio. Dicha parte se denomina **dominio** y es un **subárbol del espacio de nombres de dominio**.

Los clientes DNS se dedican a preguntar a los servidores de nombres, los cuales responden usando para la comunicación entre ellos el protocolo DNS.

El servicio DNS no sólo permite asociar un nombre de dominio con una dirección IP, además permite almacenar otras informaciones, como por ejemplo qué equipos ofrecen un determinado servicio, cuál es el servidor de correo del dominio o qué equipos son fuentes de malware o de spam.

1.3. Componentes

El servicio DNS se basa en el modelo cliente-servidor y está formado por los componentes siguientes:

Espacio de nombres de dominio

Lo forma la totalidad del conjunto de nombres, estructurados de forma jerárquica, que permite identificar equipos o servicios de red.

Base de datos DNS

Es una base de datos distribuida que contiene la información del espacio de nombres del dominio. La base de datos DNS se organiza en zonas, en las cuales los datos se registran mediante los llamados **registros de recursos** (RR).

Servidores de nombres

Los servidores DNS guardan parte de la base de datos DNS en las llamadas zonas y son capaces de responder a las preguntas relativas a la información que almacena en sus zonas. Normalmente un servidor de nombres guarda información de una sola zona, es decir la información correspondiente a esa parte del espacio de nombres o dominio, pero también puede registrar la información de varios dominios, cada una en su zona correspondiente.

Clientes DNS o resolvedores

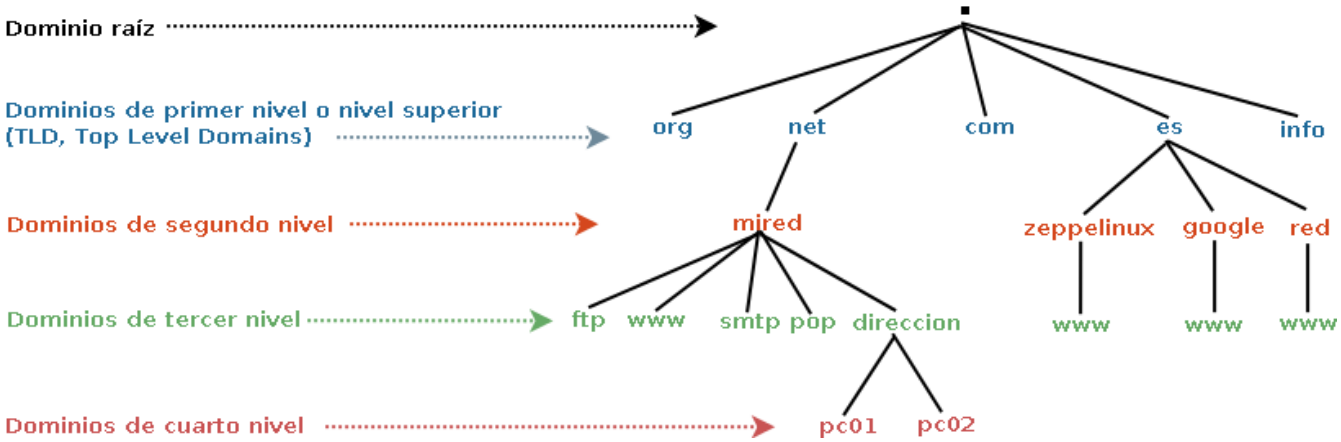
Son los encargados de realizar las preguntas a los servidores de nombres y ofrecer las respuestas a las aplicaciones que las solicitan.

Protocolo DNS

Conjunto de reglas y normas que usan los servidores y clientes DNS para dialogar.

1.3.1. Espacio de nombres de dominio

Como hemos comentado, el espacio de nombres de dominio lo forma el conjunto de nombres que permite identificar equipos o servicios de red, estructurados de forma jerárquica.



Estos nombres se denominan nombres de dominio y están formados por una serie de caracteres separados por puntos.

Ejemplo 3. Ejemplos de nombres de dominio

```
google.es., info., servidor.aula.izv.
```

El espacio de nombres de dominio se puede representar mediante una estructura jerárquica arborescente, donde cada nodo del árbol se separa de otro mediante el carácter punto.

Los nombres de dominio no pueden superar los 255 caracteres distribuidos en como máximo 127 niveles, los cuales pueden contener como máximo 63 caracteres.

Los nombres de dominio siempre terminan con un punto, ya que el árbol de nombres de dominio empieza en el dominio “.” o dominio raíz.

Los dominios que cuelgan del dominio raíz se denominan **dominios de primer nivel** o **dominios de nivel superior** (TLD, *Top Level Domains*). Los que cuelgan de los dominios de primer nivel se denominan dominios de segundo nivel. Normalmente a los dominios de tercer nivel e inferiores se les suelen denominar subdominios, aunque sólo es cuestión de nomenclatura.

Nombres de dominio relativos y absolutos

Cuando se hace referencia a un dominio, podemos usar su nombre relativo o absoluto.

Si usamos nombres relativos debemos conocer cuál es el dominio superior para saber a qué nombre nos estamos refiriendo exactamente. Si uso como nombre `puesto7` a no ser que el dominio de contexto sea por ejemplo `aula.izv.` no sabré exactamente a qué equipo estoy haciendo referencia.

Los nombres absolutos están formados por todos los nombres de nodos separados por puntos desde el nodo correspondiente hasta el nodo raíz "inclusive". Por ejemplo: `puesto7.aula.izv.`

Los nombres absolutos se denominan nombres de dominio perfectamente cualificados (FQDN, *Fully Qualified Domain Names*).

Los nombres de dominio no sólo sirven para hacer referencia a un equipo en el espacio de nombres de dominio. También sirven para hacer referencia a un subárbol del espacio de nombres del dominio, es decir para hacer referencia a un nodo, y a todos los nodos descendientes que cuelgan de éste.

Por ejemplo, el dominio es. hace referencia a todos los nodos (subdominios y equipos) que cuelgan del dominio de primer nivel es. El dominio `aula.izv.` hace referencia a todos los nodos que cuelgan del domino de segundo nivel `aula.izv.`

Por lo tanto un dominio permite hacer referencia a un conjunto de equipos y subdominios que se agrupan según un criterio. Uno de los criterios más usados, y que suele causar confusión, es el de los equipos que pertenecen a una misma red. Por ejemplo, en nuestro caso el dominio `aulaxxx.izv.` agrupa a todos los equipos del aula `xxx`, los cuales están en la misma red, pero esto no quiere decir que todos los equipos que pertenezcan a una misma red deben pertenecer al mismo dominio. De hecho se pueden usar otros criterios de agrupación: equipos de una misma empresa (aunque los equipos estén en redes distintas...), equipos que prestan servicios, equipos que estén en una misma ubicación física, etc.

1.3.2. Administración de nombres de dominio

En Internet, la administración y la organización del espacio de nombres de dominio se realiza a través de diversas empresas y organizaciones coordinadas por la ICANN (*Internet Corporation for Assigned Names and Numbers* `http://www.icann.org/`).

El ICANN tiene la misión de que Internet sea funcional y entre otras cosas, de administrar el dominio raíz, asignar bloques de direcciones IP y mantener un registro de los dominios de nivel superior (TLD).

Antiguamente, InterNIC (`http://www.internic.net/`) asumía las responsabilidades que ahora gestiona el ICANN; pero actualmente como organización asociada al ICANN, es la encargada proporcionar al público información relativa a los servicios de registro de nombres de dominio de Internet.

1.3.3. Clasificación de dominios de primer nivel

Los dominios de **primer nivel** (TLD) se clasifican en:

- Genéricos (gTLD)
- Geográficos (ccTLD)
- arpa
- Reservados

Explicaremos cada uno de ellos a continuación.

Genéricos (gTLD)

Usan un nombre relacionado con el propósito o el tipo de organización que lo va a utilizar.

Éstos a su vez se clasifican en:

Patrocinados (sTLD)
: *Sponsored TLD*. Existe una organización que lo patrocina.

Ejemplo 4. TLDs patrocinados

> Algunos ejemplos de TLDs patrocinados son `.info`, `asia`, `edu`, `gov`, etc.

No patrocinados (uTLD)
: *Unsponsored TLD*. Operan con unas reglas comunes establecidas por el ICANN.

Ejemplo 5. TLDs no patrocinados

> Como ejemplo de TLDs no patrocinados tenemos `.com`, `.net`, `.org`, `.mil`, `.gov`, etc.

Geográficos (ccTLDs)

Usan dos letras en función del país. La gestión de estos dominios es delegada por el ICANN a organizaciones propias de cada uno de los países denominadas operadoras de registro.

Ejemplo 6. Ejemplo de dominios geográficos

> `.es`, `.uk`, `.fr`

En España, Red.es (`http://www.red.es`) es la organización delegada por el ICANN para la gestión del dominio `.es.`, y del registro de dominios de segundo nivel dentro del dominio `es.`.

arpa

El dominio *arpa* lo gestiona directamente la ICANN y sirve a través de los subdominios `in-addr.arpa` y `ip6.arpa` para poder efectuar la **resolución inversa** de direcciones IP.

Reservados

Son dominios de primer nivel reservados sólo para pruebas y documentación.

Ejemplo 7. Dominios reservados

`test`, `example` y `localhost`.

1.4. Delegación de dominios

Ya comentamos que el servicio DNS se basa en la administración distribuida de la información del espacio de nombres de dominio entre múltiples servidores de nombres. Esto se consigue mediante la delegación.

La delegación permite que una organización que administra un dominio, ceda la administración de sus subdominios (de uno, de varios, o de todos) a otras organizaciones. Por ejemplo, la ICANN, administradora del dominio raíz, delega en otras organizaciones los dominios de primer nivel (TLD) y se denominan operadoras de registro (*registry operators*). En España Red.es administra el dominio de primer nivel `es.`, el cual delega a otras organizaciones los dominios de segundo nivel.

Ejemplo 8. Ejemplo de delegación

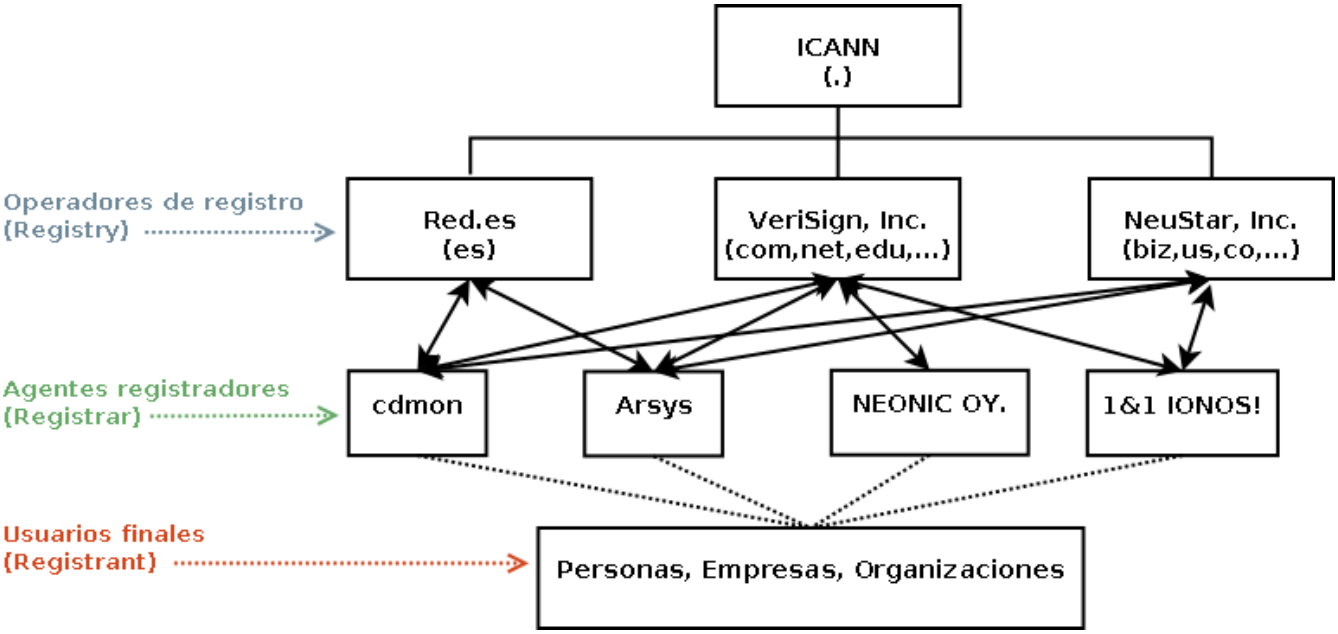
El dominio `ugr.es.` es delegado por `Red.es` a la Universidad de Granada, la cual puede a su vez delegar o no sus subdominios a otras organizaciones. Así el subdominio `etsiit.ugr.es.` es delegado por la Universidad de Granada a la ETS de Informática y Telecomunicaciones.

El hecho de que un dominio se divida en subdominios no implica que tenga que ser delegado. Por ejemplo la UGR puede crear el subdominio `derecho.ugr.es.` pero no delegar su administración a la Facultad de Derecho de la UGR.

1.5. Registros de dominio

El registro de un dominio consiste en reservarle un nombre de dominio durante un tiempo para poder crear subdominios, y asociar al dominio y a sus subdominios direcciones IP.

El registro de nombres de segundo nivel lo realizan empresas u organizaciones acreditadas denominados agentes registradores (*registrant*) los cuales asesoran a los clientes que quieren registrar un dominio, y tramitan las solicitudes operando como intermediarios entre los clientes y las operadoras de registro de primer nivel.



Así, en España, `Red.es` es la operadora de registro de primer nivel para el dominio `es.`, y aunque puede registrar directamente dominios de segundo nivel a los clientes, éstos utilizan normalmente los servicios de los agentes registradores debido a que suelen ofrecer otros servicios complementarios (alojamiento web, servidores de correo, etc,) y a precios mucho más competitivos.

En los sitios web pertenecientes a ICANN e InterNIC podemos obtener la lista de empresas registradoras acreditadas que ofrecen servicios de registro de dominios genéricos; y en `Red.es` las empresas registradoras acreditadas para el dominio `es.`.

1.6. Servidores de nombres

Los servidores de nombres o servidores DNS almacenan una parte de la base de datos del espacio de nombres guardando información sobre nombres de dominio y responden a las peticiones de los clientes DNS u a otros servidores DNS. Los servidores escuchan por dos puertos: `53/TCP` y `53/UDP`.

El puerto `53/UDP` se usa para responder rápidamente a las peticiones de resolución. El `53/TCP` se dedica para realizar las transferencias de zona entres servidores DNS para mantener copias de las zonas.

La parte de información del espacio de nombres de dominio que mantienen los servidores de nombres se denomina **zona**.

La información de una zona se almacena en archivos de texto o en bases de datos, dependiendo del tipo de servidor.

Cuando un servidor de nombres contiene información de una zona se dice que es **autorizado** (*authoritative*) para esa zona.

Los ficheros de zona almacenan básicamente sus datos mediante **registros de recursos**. Dependiendo del tipo de información que se asocie con un nombre de dominio se utiliza un tipo u otro de registro.

Ejemplo 9. Servidor DNS de `izv`

Un servidor DNS para los equipos de un aula almacenaría la información de la zona, que podríamos denominarla `aula.izv.` y en dicha zona, mediante registros, se definirían los nombres que cuelgan de `aula.izv.` como por ejemplo `puesto1.aula.izv.`, `puesto2.aula.izv.`, etc.

Bind

Aunque posteriormente lo examinaremos con más detalle, el servidor BIND (*Berkeley Internet Name Domain*), el servidor DNS más comúnmente usado en Internet y un estándar de facto, registra los datos en formato texto.

Los registros de recursos más usuales son:

NS

El registro NS (*Name Server*) indica los servidores de nombres de dominio DNS que tienen autoridad para una zona.

Ejemplo 10. Registro NS

El registro NS del dominio `aula.izv` que indica que el servidor de nombres es `dns0.aula.izv` se escribiría:

```
aula.izv. IN NS dns0.aula.izv.
```

A

El registro A (*Address*) asocia un nombre de dominio con su dirección IP.

Ejemplo 11. Registro A

El registro A de `dns0.aula.izv` es:

```
dns0.aula.izv. IN A  192.168.210.211
```

Toda zona debe tener como mínimo un registro NS y un registro A de ese servidor de nombres.

CNAME

El registro CNAME (*Canonical Name*) asigna otro nombre de dominio o alias a un nombre de dominio ya existente.

Ejemplo 12. Registro CNAME

El nombre `www.aula.izv` *apunta* a `dns0.aula.izv.` (el *canonical name*):

```
www.aula.izv. IN CNAME dns0.aula.izv.
```

Ejemplo 13. Archivo de zona de `aula.izv.`

Como ejemplo más completo mostraremos parte de un archivo de zona de resolución directa del dominio `aula.izv.`, que se almacena en un servidor DNS con IP `192.168.210.1`, y cuyo nombre de dominio es `puesto1.aula.izv.`.

```
aula.izv.          IN    NS      puesto1.aula.izv. ①
puesto1.aula.izv.  IN    A       192.168.210.1     ②
puesto2.aula.izv.  IN    A       192.168.210.2     ③
puesto3.aula.izv.  IN    A       192.168.210.3
www.aula.izv.      IN    CNAME   puesto2.aula.izv. ④
```

① El archivo de zona del dominio `aula.izv.` indica mediante el registro `NS` que el equipo `puesto1.aula.izv.` es un servidor DNS autorizado para el dominio `aula.izv`.

② Con el primer registro de tipo `A` establece que dicho equipo, `puesto1.aula.izv.` usa la dirección IP `192.168.210.1`.

③ A continuación los otros registros tipo `A` asocian un nombre de dominio con una dirección IP.

④ Con `CNAME` indicamos otro nombre de dominio (`www.aula.izv.`) para el equipo `puesto2.aula.izv.`.

Ejemplo 14. Subdominio delegado

Podemos continuar el ejemplo anterior añadiendo un subdominio delegado para `sri.aula.izv.` que estará almacenado en otro servidor denominado `puesto10.sri.aula.izv` y cuya dirección IP es `192.168.210.10`.

```
; subdominio sri.aula.izv. delegado                    ①

sri.aula.izv.           IN  NS  puesto10.sri.aula.izv. ②
puesto10.sri.aula.izv.  IN  A   192.168.210.10         ③
```

① Comentario

② Este servidor (`puesto10.sri.aula.izv`) será autorizado para el subdominio `sri.aula.izv.` y almacenará el archivo de zona de dicho subdominio.

③ El registro A del servidor de nombres, obligatorio ponerlo.

El archivo de zona del subdominio delegado `sri.aula.izv.` podría estar en otro fichero distinto:

```
sri.aula.izv.           IN  NS  puesto10.sri.aula.izv.
```

```
puesto10.sri.aula.izv.  IN  A   192.168.210.10
puesto11.sri.aula.izv.  IN  A   192.168.210.11
puesto12.sri.aula.izv.  IN  A   192.168.210.12
```

Ejemplo 15. Subdominio no delegado

Podemos añadir al primer ejemplo del archivo de resolución directa del dominio `aula.izv.` un subdominio no delegado.

```
; subdominio bd.aula.izv. no delegado

puesto15.bd.aula.izv.   IN  A   192.168.210.15
puesto16.bd.aula.izv.   IN  A   192.168.210.16
puesto17.bd.aula.izv.   IN  A   192.168.210.17
```

El subdominio `bd.aula.izv.` no se ha delegado ya que la zona `aula.izv.` también almacena los RR del subdominio. Por lo tanto el servidor DNS queda autorizado para los dominios `aula.izv` y `bd.aula.izv`.

Es común el error de considerar los términos zona y dominio como sinónimos. Un dominio es un subárbol del espacio de nombres del dominio. Los datos asociados a los nombres de dominio puede estar almacenados en una o varias zonas, las cuales pueden estar distribuidas en uno o varios servidores DNS.

En nuestro ejemplo anterior podemos comprobar cómo los nombres de dominio `aula.izv.` se distribuyen en dos archivos de zona, el fichero de la zona `aula.izv` y el fichero de la zona `sri.aula.izv`. Además, un servidor de nombres puede tener autoridad sobre varias zonas. Por ejemplo, un mismo servidor puede ser autorizado para las zonas `aula.izv.` y `taller.izv.`.

1.6.1. Tipos de servidores de nombres

Un servidor autorizado para una zona puede dejar de funcionar, o bien estar colapsado por la carga de trabajo. Para evitar estos problemas y mejorar el servicio de resolución de nombres, DNS permite almacenar una misma zona en varios servidores DNS. Podemos imaginarnos que el servicio de búsquedas de Google no iría tan rápido si hubiera un único servidor autorizado para el dominio `google.com`.

La denominación maestro/esclavo está en deshuso por las connotaciones racistas que tiene en EEUU, sustituyéndose por primario y secundario. Aquí se incluye por razones históricas.

Servidor maestro o primario (*master* o *primary*)

Define una o varias zonas para las que es autorizado. El administrador es el **responsable de los archivos de zona**, añadiendo, modificando o borrando nombres de dominio. Si un cliente DNS le

pregunta por un nombre de dominio para el que está autorizado, consultará en los archivos de su zona y le responderá. Si no está autorizado, buscará la información en otros servidores DNS si así está configurado, o responderá que no sabe la respuesta.

Servidor esclavo/secundario (*slave* o *secundary*)

Define una o varias zonas para las que es autorizado, pero **obtiene los archivos de zona de otro servidor autorizado para la zona** (el maestro u otro esclavo), realizando lo que se denomina **transferencia de zona**. Los archivos de zona del servidor esclavo no se pueden editar, por lo que las modificaciones deben realizarse en el maestro que realiza la transferencia.

Un servidor DNS puede ser maestro para una o varias zonas y a la vez esclavo de otras. Además pueden existir varios servidores esclavos para una misma zona.

Gracias a los servidores esclavos, se puede repartir la carga de trabajo entre varios servidores y disminuir los fallos que se puedan producir en el servicio.

Servidor caché

Si un servidor DNS recibe una pregunta sobre un nombre de dominio para la que no está autorizado, podrá preguntar, si está configurado así, a otros servidores DNS. Si además este servidor actúa como caché, las respuestas obtenidas de los otros servidores las podrá almacenar durante un tiempo (TTL, *Time To Live*). De esta forma, cuando un cliente u otro servidor DNS le formule una pregunta, consultará primero en su memoria caché, por si ya tuviera la respuesta.

Los servidores llamados *solo caché*, son los que **no tienen autoridad sobre ningún dominio** y **siempre realizan las preguntas a otros servidores** para resolver las peticiones, **guardando las respuestas** en su memoria caché. En redes extensas y con muchos equipos, es adecuado tener un servidor DNS que actúe de esta forma ya que se logra disminuir significativamente el tráfico en la red.

Servidor reenviador (*forwarding*)

Cuando un servidor DNS no tiene respuesta a una petición de resolución, puede si así está configurado, realizar la pregunta a otros servidores (en un orden que luego explicaremos), hasta obtener una respuesta; o bien puede trasladar directamente la pregunta a otros servidores DNS (*forwarders*), para que sean éstos los que se encarguen de resolverla.

Por lo tanto, **un servidor configurado como reenviador traslada las preguntas que no puede resolver a otro servidor DNS** al que se le traslada dicha consulta. Así se consigue disminuir el tráfico de peticiones DNS en Internet generado por los equipos de una red (se encargan de resolver los *forwarders*) y se comparte la caché de los servidores DNS a los que se le reenvían las consultas.

Lógicamente las consultas que se reenvían son sólo aquellas para las que el servidor no está autorizado, ni están previamente cacheadas.

Servidor sólo autorizado

Son los servidores que son autorizados para una o varias zonas pero que **no responden a peticiones que no sean para su zona**. Esto implica que no preguntan a otros servidores, no hacen

de reenviador, ni tampoco actúan como caché.

Servidores raíz

En Internet existen una serie de servidores DNS denominados **servidores raíz** (*root servers*) autorizados para el dominio raíz `.` Estos servidores contienen el archivo de la zona `.` en donde se delega a otros servidores DNS autorizados los dominios de primer nivel.

El ICANN es responsable de estos servidores raíz, en concreto 13, de los cuales existen copias repartidas mundialmente. Cada uno de ellos, y sus copias, se identifica con una misma dirección IP, de forma que cuando un cliente u otro servidor DNS les realiza una pregunta, responde la copia más cercana.

Estos servidores raíz deben ser conocidos por todos los servidores DNS para poder empezar el proceso de responder a preguntas sobre nombres de dominio para los que no están autorizados.

En la web `http://root-servers.org/` podemos ver cuáles son los servidores raíz, sus nombres, ubicaciones y las empresas que los administran y en `http://www.iana.org/domains/root/db` los servidores de nombres y las empresas u organizaciones responsables de los dominios de primer nivel.

1.7. Clientes DNS (resolvedores)

Antes de que existieran servidores DNS que resolvieran los dominios, necesariamente se usaba un archivo de texto (denominado *hosts*) donde se guardan las correspondencias entre nombres de dominio y direcciones IP. Este mecanismo se dejó de utilizar cuando Internet empezó a crecer en nombres de dominio, pasándose a usar servidores DNS.

Muchos sistemas operativos están configurados para usar este método de resolución antes de usar el servicio DNS. Así, si mediante la tabla de hosts no se puede resolver, se usa el servicio DNS. En una red local con pocos equipos y que mantengan direcciones fijas puede ser efectivo mantener la tabla de hosts, en otro caso, mantener actualizado y sincronizado el archivo hosts en todos los equipos es muy complicado y genera errores. En la actualidad también se usa para bloquear contenidos de Internet, como la publicidad web.

Bloquear la publicidad

Podemos sustituir el fichero `c:\windows\system32\driver\etc\hosts` en Windows el fichero `/etc/hosts` en Linux, por uno de los contenidos en `https://github.com/StevenBlack/hosts` de forma que todos los sitios web *tóxicos* resuelvan a la dirección `0.0.0.0` que es una IP no enrutable que designa un destino desconocido o inválido.

El formato del archivo hosts es el siguiente:

- En cada línea se debe introducir la dirección IP a la que resolverá, uno o más espacios o tabulaciones y el nombre de dominio a resolver.
- Se pueden introducir más de un nombre de dominio a resolver en la misma línea separados por uno o más espacios o tabulaciones.

- Cada correspondencia de dirección IP y nombre de dominio debe ir en una línea distinta.
- Las líneas que comienzan por # se consideran comentarios

Ejemplo 16. Fichero válido `/etc/hosts`

```
# Definición de localhost
127.0.0.1 localhost

# Correspondencia para equipos de la red local

192.168.210.1    pc1.aula.izv
192.168.210.222  gateway.aula.izv

# Correspondencia para una página web

209.85.229.104   www.google.es

# Dominios de Internet bloqueados mediante una IP inválida (como por
#  ejemplo una máscara de subred···)

255.255.255.0 www.fantomas.com www.patapalo.com
```

La única entrada obligatoria y que aparece siempre por defecto, es la del dispositivo de red loopback. Todas las direcciones del rango `127.0.0.0/8` son direcciones de *loopback*, de la cual la que se utiliza de forma mayoritaria es la `127.0.0.1`, pero se podría utilizar cualquiera otra dirección de dicho rango.

Esta dirección especial se suele utilizar cuando una transmisión de datos tiene como destino el propio equipo de forma que ésta se utiliza para dirigir el tráfico hacia ellos mismos. También es posible hacer ping a la dirección de loopback para probar la configuración de TCP/IP en el host local.

Un **resolvedor** (*resolver*) **es cualquier software que realiza preguntas a un servidor DNS y entiende las respuestas recibidas**. Los sistemas operativos incluyen un conjunto de librerías que realizan estas operaciones y que son invocadas por las aplicaciones cuando usan un nombre de dominio.

Normalmente, se puede configurar si el resolvedor deberá buscar primero en el archivo de hosts antes de hacer la resolución del nombre mediante el servicio DNS.

En general los pasos que sigue el resolver cuando una aplicación quiere resolver un nombre son:

1. Consulta la memoria caché de resolución de nombres del hosts, si está configurada.
2. Si la respuesta no es positiva, realiza la búsqueda en el archivo hosts del equipo local.
3. Si el nombre de dominio tampoco existe en el archivo hosts, se realizará una consulta recursiva al servidor DNS que se tenga configurado y éste suministrará la respuesta a la aplicación.

1.8. Mecanismo de resolución

Las consultas a un servidor DNS pueden ser **recursivas** o bien **iterativas** (también llamadas no recursivas). Como veremos a continuación, a la hora de resolver un nombre, el resolver realiza una consulta recursiva, la cual podrá generar o no en una serie de consultas iterativas.

Consulta recursiva

es aquella en la que el servidor siempre debe dar una respuesta completa o exacta. Esta respuesta podrá ser una de las tres: positiva (es decir, da la información) indicándose si es autorizada o no, negativa (si no se pudo resolver) o de error (por ejemplo por fallo en la red).

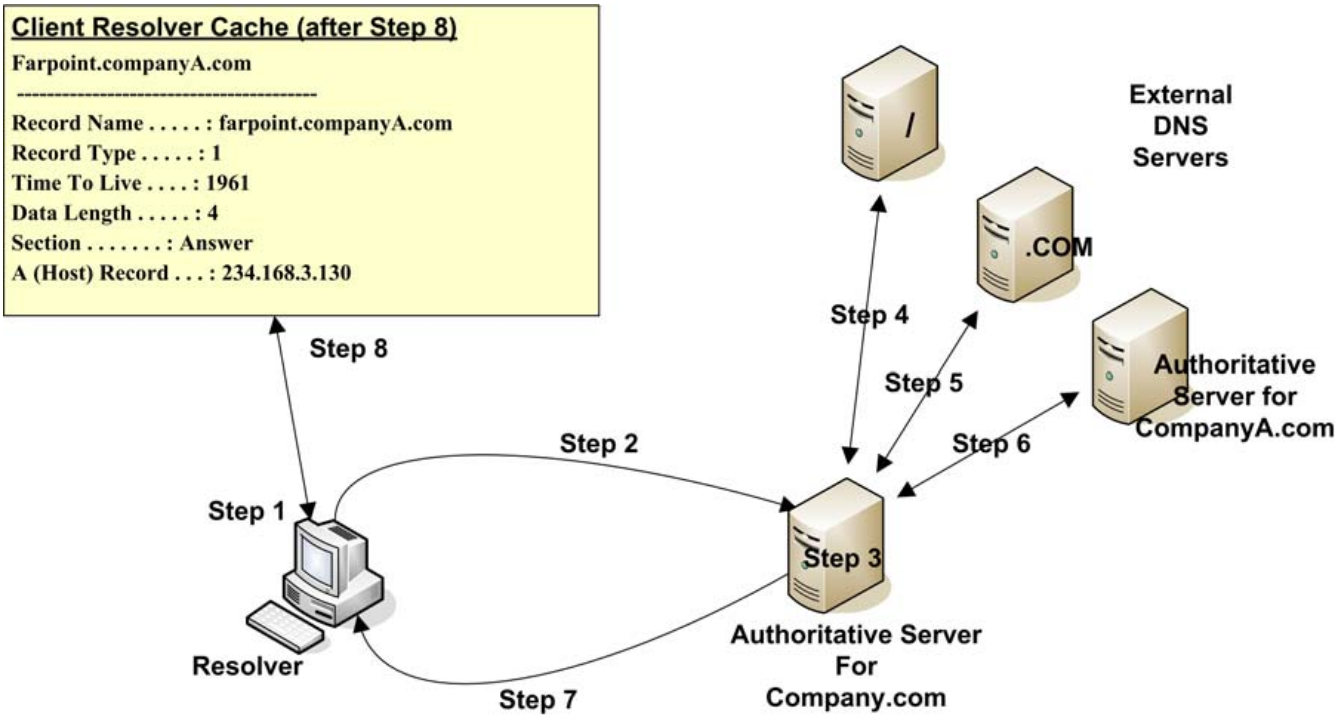
Consulta iterativa

es aquella en la que el servidor DNS puede proporcionar además una respuesta parcial. En este caso, a parte de las respuestas positivas, negativas o de error, puede dar una respuesta incompleta que indique una referencia a otros servidores a los que se les puede preguntar para resolver la pregunta.

En una consulta recursiva, el cliente realiza la pregunta a su servidor DNS, y éste tiene que darle una respuesta completa, realizando consultas a otros servidores DNS si fuera necesario para darle la respuesta. En una consulta iterativa, el servidor puede dar como respuesta información de a qué otros servidores DNS debería preguntar de nuevo el cliente para obtener respuesta a su pregunta. En este caso, el cliente es el encargado de ir haciendo preguntas a diferentes servidores para obtener una respuesta final.

Una consulta recursiva la inicia un cliente DNS a través del *resolvedor*, o bien un servidor DNS que la traslada a otro servidor DNS actuando como *reenviador*. El proceso de resolución cuando un servidor recibe una consulta recursiva es el siguiente:

1. Si el servidor es autorizado para alguna zona, comprueba sus archivos de zona, y si encuentra la respuesta, responde indicando que la respuesta es *autoritativa*.
2. Si no encuentra la respuesta, o no es autorizado y actúa como caché, consulta su caché de respuesta anteriores, y si la encuentra responde que la respuesta es *no autoritativa*.
3. Si en el paso anterior no se encontró respuesta positiva, y tiene configurados reenviadores, entonces reenvía la consulta recursiva a este otro servidor DNS y la respuesta que reciba de éste la traslada al cliente o al servidor que le preguntó.
4. Si **no tiene configurados reenviadores** entonces:
    a. El servidor inicia una serie de consultas *iterativas* a otros servidores DNS empezando la primera de ellas por un servidor raíz.
    b. Los servidores consultados devuelven *referencias* a otros servidores DNS que se usan para realizarles la pregunta. Este proceso de preguntas a distintos servidores finaliza cuando finalmente un servidor autorizado proporciona una respuesta positiva o negativa.



La figura muestra los pasos cuando un cliente realiza una pregunta recursiva por el dominio `farpoint.companyA.com` a un servidor que tiene activada la recursividad (tiene que dar una respuesta completa). Como el servidor DNS no está autorizado para la zona `companyA.com`, no tendrá en sus archivos de zona el nombre de dominio por el que se le ha preguntado.

Como debe dar una respuesta, empezará una serie de consultas iterativas empezando por uno de los servidores raíz. El servidor raíz responde con una referencia al servidor autorizado para el dominio `com`. El servidor DNS envía una consulta iterativa al servidor autorizado del dominio `com`, y éste le responde con una referencia al servidor autorizado para el dominio `companyA.com`. Posteriormente el servidor DNS envía una consulta iterativa al servidor autorizado del dominio `companyA.com`.

El servidor autorizado del dominio `companyA.com` consulta en sus ficheros de zona y como existe el nombre de dominio `farpoint.companyA.com` responde con la información asociada a él, en este caso con la dirección IP (respuesta positiva). Si no existiera el nombre de dominio en sus archivos de zona, la respuesta sería negativa.

Finalmente el servidor que recibió la respuesta recursiva entrega al resolver la información por la que preguntó y la guarda en su caché disponible para futuras consultas.

Por lo tanto, las consultas iterativas son iniciadas por un servidor DNS a otro servidor DNS, cuando en una consulta recursiva no ha encontrado la respuesta en sus archivos de zona o en la caché.

En este proceso de resolución es muy importante la caché. Por ejemplo, si el resolvedor pregunta primero por el dominio `www.granada.es` y luego por `www.alomartes.es` no tendrá que preguntar de nuevo a un servidor raíz, debido a ya que tendrá almacenada en la caché las direcciones de los servidores DNS autorizados para el dominio `es`.

Como hemos podido comprobar, las consultas recursivas son costosas para los servidores DNS ya que deben dar siempre una respuesta; de hecho **los servidores DNS raíz y los de primer nivel**

(TLD) no responden nunca a consultas recursivas.

1.8.1. Respuestas en caché

Las respuestas almacenadas en cache pueden ser positivas o negativas. Es decir se puede almacenar los registros de recursos de nombres de dominio resueltos (respuesta positiva), y la información de que no existe un registro de recursos para el nombre de dominio consultado (respuesta negativa). Cacheando las respuestas negativas se impide la repetición de preguntas para nombres que no existen.

Los archivos de zona almacenan **los tiempos máximos que guardan las respuestas en caché** (TTL), de forma que habrá TTL para las respuestas positivas y TTL para las negativas. Se recomienda que el TTL para las respuestas positivas sea mayor de un día, incluso semanas, y que el TTL para las respuestas negativas sea menor de 3 horas.

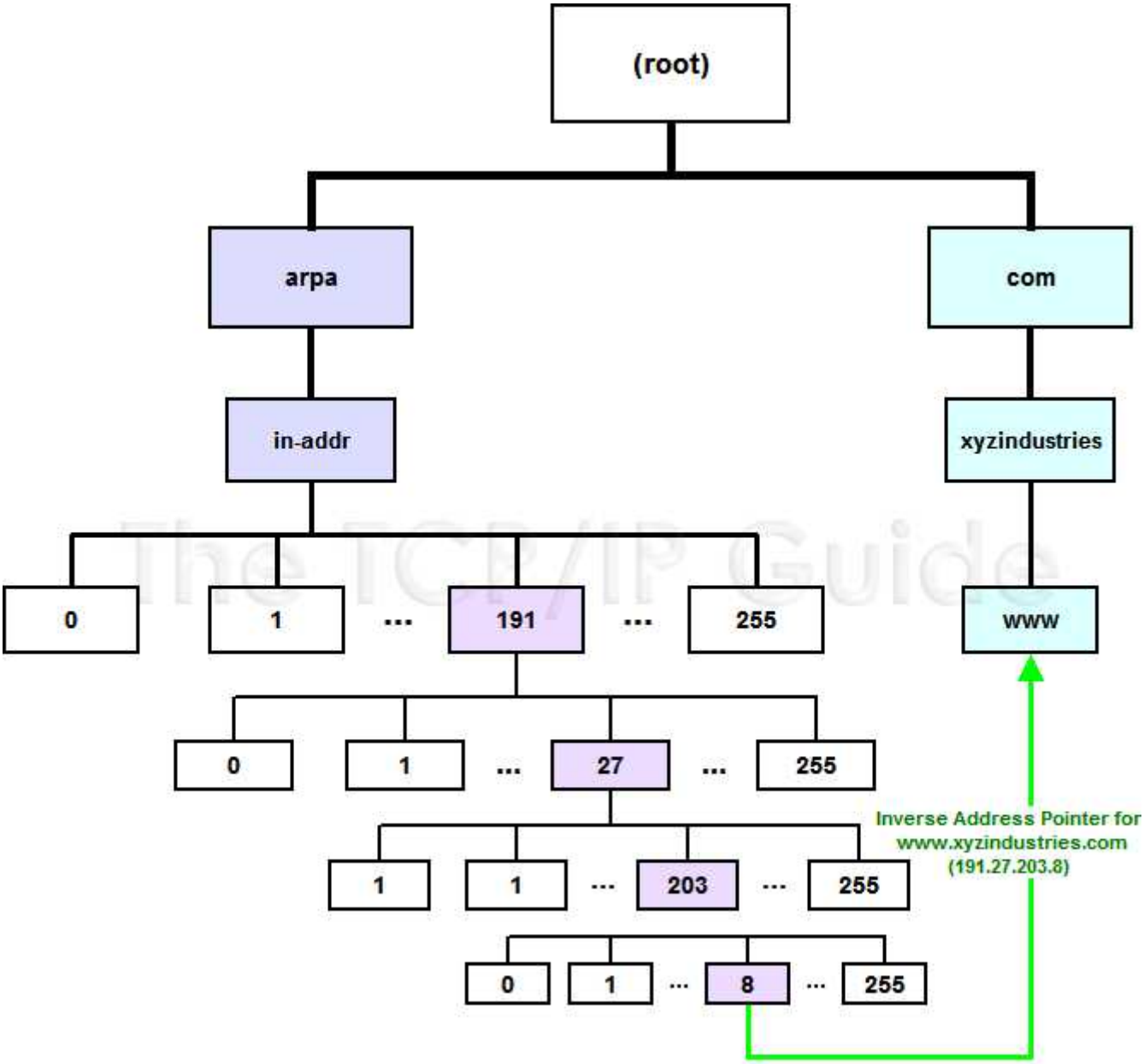
El administrador es el que tiene que valorar estos tiempos en función de la frecuencia con la que cambian los registros de recursos. Si el TTL para las preguntas positivas es pequeño, la información del dominio será más congruente pero a costa de aumentar el trabajo de los servidores y la carga de la red.

1.9. Correspondencias inversas

Las resoluciones de nombre de dominio que hemos comentado hasta ahora se denominan de resolución directa. Las **resoluciones inversas consisten en preguntar por una dirección IP para obtener el nombre o nombres de dominio como respuesta**.

Uno de los principales motivos por el que se realiza una resolución inversa está ligado a la seguridad. Si al realizar una resolución directa para un nombre de dominio obtenemos una dirección IP, la cual se usa para realizar una resolución inversa, se debería obtener el nombre de dominio por el que preguntamos. Si no es así, posiblemente un intruso esté resolviendo nombres a direcciones IP no válidas con fines maliciosos. Los servidores DNS pueden configurarse para que actúen de esta forma para asegurarse de las respuestas recibidas.

También se suelen usar para detectar errores en la configuración de los servidores y equipos, spam en los servidores de correo, o para seguir la traza de un ataque.



En la resolución inversa, las direcciones IP se tratan como nombres donde cada byte de la IP es un dominio que cuelga del dominio `in-addr.arpa`. Así cuando realizamos una pregunta inversa del tipo ¿cuál es el nombre de dominio para la IP `192.168.210.1`?, lo que realmente estamos preguntando es por el nombre de dominio de `1.210.168.192.in-addr.arpa`. Como vemos, la estructura jerárquica de la dirección IP tratada como nombre de dominio es de izquierda a derecha comenzando por el dominio `in-addr.arpa`.

Para la resolución inversa los servidores de nombres deben almacenar zonas de resolución inversa que podrán ser maestras o esclavas.

Las zonas directas e inversas son independientes, y el administrador debe mantener la información de ambas sin discrepancias. Si se administra una zona directa no es obligatorio administrar la zona inversa correspondiente. De hecho si contratamos un dominio asociado a una IP pública a través de un ISP (Proveedor de Servicios Internet), normalmente el ISP no incluirá nuestro dominio en su zona inversa salvo que se lo pidamos expresamente.

El proceso de resolución inversa es análogo al directo. Así, una consulta recursiva de la IP `191.27.203.8` a un servidor DNS, empezaría en buscar en la caché, y si no tiene la respuesta

comenzaría una serie de consultas iterativas a los servidores DNS raíz, después a los servidores autorizados para el dominio `191.in-addr.arpa`, luego a los autorizados para el dominio `27.191.in-ddr.arpa`, hasta realizar la consulta al servidor autorizado para el dominio `203.27.191.in-ddr.arpa`, el cual debe dar un respuesta completa.

Los archivos de zona de resolución inversa usan registros de recursos de tipo NS y PTR. Como veremos, los registros PTR asocian IP a nombres de dominio.

Ejemplo 17. Zona de resolución directa para `aula.izv`

Por ejemplo para el siguiente archivo de zona de resolución directa para el dominio `aula.izv`:

```
aula.izv.          IN NS     puesto1.aula.izv.
puesto1.aula.izv.  IN A      192.168.210.1
puesto2.aula.izv.  IN A      192.168.210.2
puesto3.aula.izv.  IN A      192.168.210.3
www.aula.izv.      IN CNAME  puesto2.aula.izv.
```

Ejemplo 18. Zona de resolución inversa `aula.izv`

Tendríamos el siguiente archivo de zona de resolución inversa `210.168.192.in-addr.arpa.` Este archivo permitiría resolver las consultas inversas sobre direcciones IP de la red `192.168.210.0/24`.

```
210.168.192.in-addr.arpa.   IN  NS  puesto1.aula.izv.
1.210.168.192.in-addr.arpa. IN  PTR puesto1.aula.izv.
2.210.168.192.in-addr.arpa. IN  PTR puesto2.aula.izv.
3.210.168.192.in-addr.arpa. IN  PTR puesto3.aula.izv.
```

1.10. Registros de recursos

Los archivos de zona almacenan la información sobre nombres de dominio mediante **registros de recursos** (RR, *Resource Records*). Estos RR son los que reciben los clientes cuando realizan peticiones de resolución a los servidores DNS.

Los RR se representan en los archivos de zona en formato texto, pero de forma binaria en los mensajes que se envían mediante el protocolo DNS. El formato de los RR en modo texto es:

```
Nombre de Dominio   [TTL]   Clase   Tipo    Dato
```

Ejemplo de RR

```
puesto1.aula.izv.   3600    IN  A   192.168.210.1
```

Lo primero es el nombre de dominio que se asocia al recurso, opcionalmente aparece el tiempo de vida medio (TTL) que indica en segundos el tiempo que puede estar el registro en caché antes de ser descartado. (Veremos que se puede especificar un tiempo global para todos los registros de la zona).

La clase define el tipo de protocolo usado. Aunque se pueden usar otras arquitecturas la habitual es Internet (IN). A continuación el tipo indica el tipo o clase de dato asociado al nombre de dominio y por último se especifica el dato asociado al nombre de dominio.

1.11. Tipos de registros

Comentaremos sólo los tipos de registros de recursos más usados, ya que el protocolo define alrededor de 30 tipos diferentes. En el sitio web del IANA se pueden consultar todos ellos.

1.11.1. Registro SOA (*Star of Authority*)

Es el primer registro de una zona y permite definir el tipo de servidor y las opciones generales de la zona.

El registro tipo SOA comienza con el nombre del dominio de la zona, seguido opcionalmente de un `TTL` (lo normal es usarlo para así establecer de forma global el `TTL` para toda la zona), continua con la clase `IN`, seguido del tipo `SOA`, y finaliza con los datos asociados. Estos datos asociados son:

FQDN
: del servidor de nombres maestro para el dominio. ◦

Contacto
: Dirección de correo del responsable del dominio. (La arroba se sustituye por “.”)

Número de serie (serial)
: Indica la versión del archivo de zona que se irá incrementando cada vez que el archivo se modifique. De esta forma los servidores secundarios conocen si el archivo de zona ha cambiado y deben actualizarse mediante una transferencia de zona.

Actualización (refresh)
: Indica cada cuanto tiempo deben contactar los servidores secundarios con el servidor maestro para comprobar si ha habido cambios en la zona. Dependiendo de la frecuencia de actualización de la zona primaria se usará más o menos tiempo. Se recomiendan valores entre 20 minutos y 212 horas. (Si se usan notificaciones, se puede usar un valor alto).

Reintentos (retry)
: Indica cada cuanto tiempo deben reintentar los servidores secundarios contactar con el servidor maestro si éste no responde a una petición de actualización.

Caducidad (expire)
: Tiempo durante el cual un servidor secundario puede estar sin contactar con el primario para comprobar la zona. Si se supera este tiempo, el secundario descarta los datos que tenía sobre la zona y se declara no autorizado para la zona. (Se recomienda entre 2 y 4 semanas)

TTL negativo

Es el tiempo mínimo en que se almacena las respuestas negativas sobre la zona.

Ejemplo 19. SOA del dominio `aula.izv`

```
aula.izv.  IN  SOA  puesto1.aula.izv. remaster.aula.izv. (
2020102301  ; nº de versión del archivo de zona
604800              ; tiempo de refresco
86400               ; tiempo de reintento
2419200             ; tiempo de expiración
38400 )             ; TTL negativo
```

1.11.2. Registro NS (Name Server)

Permite indicar los servidores de nombres autorizados para una zona. (Cada zona debe contener al menos un registro NS, por ejemplo un maestro; y puede tener uno o más registros NS para indicar los esclavos). Este registro también permite indicar los nombres de los servidores con autoridad de los subdominios que hayan sido delegados.

1.11.3. Registro A (Address)

Establece una correspondencia entre un nombre de dominio FDQN y una dirección IPv4.

```
dns0.aula.izv. IN A 192.168.X.211
```

1.11.4. Registro AAAA (Address)

Establece una correspondencia entre un nombre de dominio FDQN y una dirección IPv6.

```
ftp        IN      AAAA      2001:db8::5
```

1.11.5. Registro CNAME (Canonical Name)

Indica un alias o sobrenombre para los nombres de dominio previamente especificados en registros A y AAAA. Un alias puede apuntar a otro dominio, pero no se pueden usar en la parte derecha de registros MX y NS, ya que estos dos tipos de registros necesitan siempre usar en su parte derecha nombres que aparezcan en registros de tipo A o AAAA.

```
www.aula.izv. IN CNAME dns0.aula.izv.
```

1.11.6. Registro MX (Mail Exchange)

Define equipos encargados del correo en el dominio, para que los servidores de correo sepan a qué

equipo deben entregar los mensajes. Se pueden definir varios servidores de correo donde en el RR de cada uno de ellos se indica mediante un número el orden de preferencia de cada uno de ellos; a número menor le corresponde mayor preferencia.

```
aula.izv. IN MX 10 dns0.aula.izv.
```

1.11.7. Registro SRV (Services Record)

Permite definir equipos que actúan como servidores de algún servicio particular en el dominio. Por ejemplo servidores LDAP, servidores de mensajería, etc.

```
_https._tcp IN SRV 10 10 443 misrv.aula.izv.
```

1.11.8. Registro PTR (Pointer Record)

Establecen correspondencias inversas, es decir, entre direcciones IP y nombres de dominio. Por lo tanto sólo se usan en las zonas de resolución inversas. Si se usan direcciones tipo IPv4 e IPv6 deben aparecer en zonas separadas.

```
211.X.168.192.in-addr.arpa. IN PTR dns0.aula.izv.
```

1.11.9. Registro TXT (Texto)

Permite registrar cualquier texto libre que tenga relación con un equipo.

```
aula.izv. 3600 IN TXT "Aquí pongo lo que quiera"
```

1.12. Registros pegamento (Glue Record)

Sabemos que un servidor de nombres se puede configurar para delegar o no algunos de sus subdominios en otros servidores de nombres. Podemos encontrar dos situaciones.

En el caso de que el servidor de nombres autorizado para el subdominio delegado se encuentre en el mismo dominio es necesario añadir:

1. Un registro NS en el dominio que delega, para indicar cuál es el servidor de nombres para la zona delegada.
2. Un registro A en el dominio que delega que indique la IP del servidor de nombres autorizado del subdominio delegado. Este tipo de registro se denomina *glue record* porque que de alguna forma relaciona o pega la zona hija con la zona padre.
3. OJO. Si el dominio que delega usa reenviadores, cuando le consultemos por un nombre del subdominio delegado, nuestro servidor DNS detectará que no tiene autoridad sobre él, y por lo tanto le preguntará al servidor DNS indicado en el forwarders sin realizar la delegación.

Como veremos, si usamos *bind9*, para solucionar este problema tenemos que anular la función de reenvío pero sólo para la zona que se delega, y para ello en `/etc/bind/named.conf.local`, donde se definen las zonas DNS, configuraríamos en la zona que delega el parámetro: `forwarders { }`;

Por lo tanto, **los servidores de un dominio deben conocer la IP de los servidores de nombres de los subdominios, para que los clientes puedan dirigirse a ellos en las consultas**. (Ver ejemplo pag. 6).

En el caso de que el servidor de nombres autorizado para el subdominio delegado no se encuentre en el mismo dominio, sólo es necesario añadir el registro NS que indique el servidor de nombres de la zona delegada. En este caso no necesita un registro tipo A, ya que los clientes podrán resolver el nombre de dominio de la zona delegada normalmente.

Finalizamos con un ejemplo de zona de resolución directa para el dominio aula.izv. con un subdominio delegado en un subdominio propio y otro delegado en un subdominio ajeno.

```
aula.izv. IN SOA puesto1.aula.izv. carlos.aula.izv. (
             2012041401    ; nº de versión del archivo de zona
             604800        ; tiempo de refresco
             86400         ; tiempo de reintento
             2419200       ; tiempo de expiración
             38400 )       ; TTL negativo

aula.izv.       IN  NS  puesto1.aula.izv.
puesto1.aula.izv.   IN  A   192.168.210.1
puesto2.aula.izv.   IN  A   192.168.210.2
puesto3.aula.izv.   IN  A   192.168.210.3
www.aula.izv.       IN  CNAME   puesto2.aula.izv.
ftp.aula.izv.       IN  CNAME   puesto2.aula.izv.
aula.izv.       IN  MX  10
smtp.aula.izv.      IN  CNAME   puesto3.aula.izv.
_ldap._tcp.aula.izv.    IN  SRV 0 0 389 puesto3.aula.izv.

;subdominio sri.aula.izv.delegado en un subdominio propio
sri.aula.izv.       IN  NS      puesto10.sri.aula.izv.
puesto10.sri.aula.izv.  IN  A   192.168.210.10 ①

;subdominio bd.aula.izv.delegado en un subdominio ajeno
bd.aula.izv.        IN  NS      ns1.informatica.izv.
```

① Glue record

1.13. Transferencias de zona.

Los servidores en los que se declaran zonas esclavas, deben obtener los archivos de zonas, es decir los registros de recursos, de otros servidores (maestros o esclavos) autorizados para dichas zonas.

Este proceso se denomina transferencia de zona y hay que configurar los servidores para que las realicen. Los servidores maestros usan el puerto `53/TCP` para realizar el intercambio de zona la cual

puede ser de dos tipos: *completa e incremental*.

En las transferencias de zonas completas, el servidor maestro envía al esclavo todos los datos de la zona sustituyendo a los datos anteriores. En las transferencias de tipo incremental, el maestro sólo envía los datos que han cambiado desde la última transferencia de zona.

Las transferencias de zona puede comenzar por una pregunta del servidor esclavo al maestro para comprobar si hay algún cambio en la zona. La primera vez que el servidor esclavo se inicia, realiza esta pregunta, y luego la repite cada cierto tiempo (especificado en el campo refresh del registro SOA).

Por otro lado el servidor maestro puede ser el encargado de notificar a los servidores esclavos de las modificaciones producidas en sus zonas. De esta manera, si se produce una modificación en la zona del maestro, el esclavo queda enterado de forma inmediata y puede comenzar el proceso de transferencia.

En el proceso de transferencia, el maestro envía su registro SOA al esclavo, de forma que éste obtiene su número de serie y lo compara con el que tiene almacenado en su zona. Si el número de serie que obtiene es superior al suyo, entiende que sus datos no están actualizados y se realiza la transferencia.

El que la transferencia sea de tipo completa o incremental depende del tipo de petición que envíe el servidor esclavo al maestro, empleando para ello mensajes de tipo AXFR o IXFR respectivamente.

1.14. DNS dinámico (DDNS, Dynamic DNS)

Hasta ahora hemos comentado que es responsabilidad del administrador mantener actualizada la zona de los servidores de nombres. Esta tarea puede ser muy costosa si la organización dispone de varios dominios o múltiples servidores DNS. Si además si se usan servidores DHCP para asignar direcciones a los equipos, el mantenimiento manual de los registros de las zonas no es una opción viable. Además cada vez que se realizara una modificación en una zona, se tendría que parar y reiniciar el servicio DNS.

Para aliviar esta situación, el RFC 2162 define los procesos para que de forma automática, los registros de recursos de una zona se puedan actualizar de forma externa, sin que el administrador tenga que editar manualmente los archivos de zona y sin necesidad de reiniciar el servicio DNS.

Normalmente estas actualizaciones dinámicas de los registros de una zona las pueden realizar los propios equipos clientes o bien los servidores DHCP. En ambos casos hay que configurar el servidor DNS para que permita las actualizaciones por parte de los clientes o por el servidor DHCP.

Para controlar quién puede realizar estas actualizaciones, se suelen usar ACL (listas de control de acceso) para determinar desde que IPs se pueden realizar las actualizaciones, acompañado de algún protocolo de autentificación simple como es TSIG (Transaction SIGnature) o TKEY (Transaction Key) para aumentar la seguridad.

Otro de los problemas es el acceso desde Internet a diferentes equipos que ofrecen servicios (servidor web, ftp, correo, etc.) en un dominio que no tiene una IP fija en Internet.

Actualmente, la mayoría de los ISP proporcionan direcciones IP públicas por DHCP al router que

nos conecta a Internet. Esto implica que la IP pública cambiará cada cierto tiempo y por lo tanto no podemos asignarle un nombre de dominio estable a esta dirección IP. Los llamados DNS dinámicos que ofrecen algunos sitios web en Internet permiten registrar un nombre de dominio actualizándolo con la dirección IP que realmente se tenga en el momento. Lo más habitual es que sea el propio router el que tenga la posibilidad de actualizar el servidor DNS cuando cambie su dirección IP, aunque también podría ser un programa instalado en algún equipo de la red.

Varias web, como DynDNS, No-ip, EasyDNS, DonDNS, etc. ofrecen estos servicios de forma gratuita sobre un dominio de segundo nivel de su propiedad. Por ejemplo podríamos configurar un servidor web con IP dinámica con el nombre de dominio `micasa.dyndns.org`.

1.15. Seguridad DNS.

El servicio DNS es fundamental para el funcionamiento de una red e imprescindible en Internet, por ello suele ser un objetivo para los atacantes. El hecho de que el protocolo DNS se diseñara inicialmente sin tener en cuenta la seguridad (al igual que otros servicios de red) y que el servicio, al ser distribuido y dependa de muchos equipos, implica ciertas dificultades para su administración y seguridad.

1.15.1. Amenazas del servicio DNS

Las principales amenazas para un servicio DNS son:

- Ataques al servidor DNS usando *exploits* aprovechando agujeros en la seguridad.
- Modificaciones de los archivos de zonas aprovechando una mala configuración de los permisos de usuarios que puedan acceder al equipo servidor de forma remota.
- Ataques DoS (Denial of Service) mediante inundación de múltiples peticiones de resolución.
- Suplantación del servidor DNS, enviando RR incorrectos y envenenamiento de caché de clientes.
- Envenenamiento de la caché de un servidor DNS al preguntar a otro DNS suplantado.
- Suplantaciones del servidor maestro en las transferencias de zona.
- Suplantaciones de los orígenes externos que envían actualizaciones a los DNS dinámicos.

1.15.2. Mecanismos de seguridad

En los servidores DNS

- Actualización a las últimas versiones software,
- Instalación de parches de seguridad,
- Enjaular el servidor en un entorno seguro,
- Configuración de los privilegios de acceso a los archivos de zona,
- Distribuir los servidores DNS en distintas redes y ubicaciones.

En las transferencias de zona y actualizaciones dinámicas

- Establecer ACL (listas de control de acceso por IP o nombres de dominios) con los servidores desde los que se permiten las transferencias de zona y las actualizaciones.
- Utilizar cortafuegos para controlar las transferencias y actualizaciones de zona.
- Mecanismos de autentificación de servidores y equipos mediante generación de claves del tipo TSIG o TKEY.

En las consultas DNS

- Configurar los servidores para minimizar las consultas recursivas.
- Limitar por dirección IP los equipos que pueden preguntar a un servidor.
- Implantación de DNSSEC (DNS Security) basado en algoritmos de cifrado asimétricos para garantizar la autenticidad e integridad en las consultas y respuestas DNS.

Capítulo 2. Servidores DNS en Linux

El servidor DNS más extendido en sistemas operativos basados en UNIX, y de facto un estándar, se llama **bind** (*Berkeley Internet Name Domain*), y actualmente lo desarrolla y mantiene la organización sin ánimo de lucro ISC (*Internet Systems Consortium*).

Durante la configuración utilizaremos una red de ejemplo `192.168.X.0/24`. Deberás cambiar la `X` por tu configuración de red.

2.1. Pasos previos

Antes de realizar la instalación, necesitaremos siempre poder resolver el nombre de nuestro servidor independientemente de que tengamos un servidor DNS. Al inicio del sistema, puede suceder que sea necesario resolver el nombre del equipo, pero que no se logre debido a que todavía no esté operativa la red o el servicio DNS, o bien que no podamos acceder al servidor DNS.

Ejemplo 20. Configurar el nombre del servidor

Escribiremos el nombre del servidor en el fichero `/etc/hostname`.

```
dns0.aula.izv
```

Alternativamente podemos hacerlo:

```
$ sudo hostnamectl set-hostname dns0.aula.izv
```

Luego configuremos la dirección IP con el nombre del servidor en el fichero `/etc/hosts`.

`/etc/hosts`

```
127.0.0.1 localhost
127.0.1.1 dns0.aula.izv dns0 ①
::1       ip6localhost ip6loopback
```

① Da igual el orden y poner `dns0` y luego `dns0.aula.izv`

También, como suele suceder en la mayoría de los servicios, el servidor deberá tener una IP fija.

2.2. Instalación del servicio

El servidor bind se instala mediante el paquete `bind9`

```
apt install bind9 bind9-dnsutils
```

Sus archivos de configuración se ubican en el directorio `/etc/bind`, y el demonio del proceso servidor se denomina `named`.

Los scripts que permiten arrancar, parar y reiniciar el servicio son:

```
systemctl {start|stop|reload|restart|force-reload|status} bind9.service
```

Terminada la instalación debemos comprobar que el proceso named está en ejecución, y que el servicio está a la escucha en los puertos `53/TCP` y `53/UDP`. Para ello ejecutamos:

```
$ ps -ef | grep named
$ ss -ltun
```

Errores IPv6

A veces podemos ver unos errores relacionados con IPv6. En ese caso podemos configurar el fichero `/etc/default/named` con la opción `-4` para que solo utilice IPv4.

```
# run resolvconf?
RESOLVCONF=no
# startup options for the server
OPTIONS="-u bind -4" ①
```

① Añadimos `-4` para que sólo utilice IPv4.

2.3. Ficheros de configuración

Cuando se instala `bind`, se generan unos archivos con una configuración básica, con unas zonas ya preconfiguradas.

`/etc/bind/named.conf`
: es el archivo de configuración principal y mediante la directiva `include`, se incluyen las configuraciones de los tres archivos `named.conf.options`, `named.conf.local` y `named.conf.default-zones`. No nos hace falta modificar este fichero.

`/etc/bind/named.conf.options`
: Contiene las opciones de configuración generales del servidor y define su comportamiento de forma global.

`/etc/bind/named.conf.local`
: Es el archivo de configuración de zonas y donde se declaran las zonas de resolución tanto directas como inversas. Inicialmente el archivo está vacio.

`/etc/bind/named.conf.default-zones`

Contiene la definición de las zonas creadas por defecto.

Los archivos de las zonas por defecto, y referenciados desde `/etc/bind/named.conf.default-zones` son:

`/etc/bind/db.root`

Servidores raíz.

`/etc/bind/db.local`

Resolución directa del bucle local. (Resuelve `localhost`)

`/etc/bind/db.127`

Resolución inversa del bucle local. (Resuelve `127.0.0.1`)

`/etc/bind/db.0`

Resolución directa broadcast.

`/etc/bind/db.255`

Resolución inversa broadcast.

2.4. Diagnóstico

Siempre que se cambie la configuración del servidor hay que reiniciar el servicio para que tengan efecto los cambios.

2.4.1. status

Para comprobar que no tenemos errores primero comprobaremos el estado del servicio con

```
systemctl status bind9
```

2.4.2. Logs

También puede ser conveniente examinar el archivo de logs `/var/log/syslog` del sistema para comprobar que no se hayan producido fallos en el arranque del servidor.

```
tail /var/log/syslog
```

2.4.3. Ficheros de configuración

Para comprobar que no hay errores en los ficheros de configuración `named.conf.options` y `named.conf.local` utilizaremos el programa `named-checkconf`. Si la salida no muestra nada es que no hay errores.

```
named-checkconf <fichero-configuración>
```

2.4.4. Ficheros de zona

Para comprobar que no hay errores (de sintáxis) en los ficheros de zona utilizaremos el programa `named-checkzone`. Si la salida es `OK` es que no hay errores.

```
named-checkzone <nombre-zona> <fichero-zona>
```

Ejemplo 21. Comprobar errores en zona directa `aula.izv`

Suponemos que la zona directa `aula.izv.` ha sido guardada en el fichero `/var/lib/bind/db.aula.izv`. Comprobaremos los errores con:

```
named-checkzone aula.izv. /var/lib/bind/db.aula.izv
```

Ejemplo 22. Comprobar errores en zona inversa `192.168.207/24`

Suponemos que la zona inversa ha sido guardada en el fichero `/var/lib/bind/db.192.168.207`, comprobaremos si tiene errores con:

```
named-checkzone 207.168.192.in-addr.arpa. /var/lib/bind/db.192.168.207
```

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

Como el contenido del archivo de configuración `/etc/bind/named.conf.options` que se genera por defecto es muy simple, como documentación le vamos a añadir algunas opciones más, cuyo significado aparecen como comentarios.

Ejemplo 23. Ejemplo de fichero `/etc/bind/named.conf.options`

```
options {
    directory "/var/cache/bind"; ①
    allow-query { 127.0.0.1; 192.168.210.0/24; }; ②
    recursion yes; ③
    allow-recursion {127.0.0.1; 192.168.210.0/24;}; ④
    notify no; ⑤
    // forwarders {80.58.61.250;}; ⑥
    auth-nxdomain no;    # conform to RFC1035 ⑦
```

```
    listen-on port 53 { 127.0.0.1; 192.168.210.211;}; ⑧
    listen-on-v6 { any; }; ⑨
    dnssec-validation no; ⑩
};
```

① Directorio de trabajo por defecto

② Para aceptar sólo peticiones de resolución de ciertos equipos.

③ Se permiten consultas recursivas (recursion yes;)

④ Pero podemos limitar dichas consultas recursivas solo a los equipos que indiquemos

⑤ En principio no se permite enviar notificaciones de actualización de zona a otros servidores

⑥ Servidores DNS a los que reenviar peticiones de información de otras zonas (comentado)

⑦ No se responderá autoritativamente a peticiones de nombre de dominio inexistentes

⑧ Direcciones y puertos en los que realizar la escucha de peticiones de resolución. Si hay varias interfaces de red podemos indicar las interfaces y puertos de escucha

⑨ También escucha en IPv6 (y además por cualquier interfaz de red)

⑩ No se usa validación entre clientes y servidores usando firmas digitales

Existen muchas más opciones disponibles, algunas de éstas irán apareciendo posteriormente.

2.6. Configuración del servidor DNS como solo caché

Por defecto, el servidor bind está configurado como solo caché ya que no está autorizado para ninguna zona, y es capaz de responder a consultas recursivas (por defecto el parámetro `recursión` es `yes`).

Podemos probarlo configurando un cliente con dicho servidor DNS y preguntar por cualquier nombre de dominio en Internet. Además las consultas repetidas se aceleran al estar resueltas en la caché.

Ejemplo 24. Caché de consultas repetidas

```
usuario@dns0:~$ dig @localhost www.google.es

; <<>> DiG 9.16.15-Debian <<>> @localhost www.google.es
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25754
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 663afa2494360c1a010000006168857c0da86edc77897ce9 (good)
;; QUESTION SECTION:
```

```
;www.google.es.         IN  A

;; ANSWER SECTION:
www.google.es.      113 IN  A   142.250.200.131 ①

;; Query time: 823 msec ②
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: jue oct 14 21:31:08 CEST 2021
;; MSG SIZE  rcvd: 86
```

① Respuesta

② Tiempo de la consulta. Si lo pedimos de nuevo veremos que pone 0 msec ya que estará la respuesta cacheada

No obstante suele ser peligroso tener un servidor que permite recursión y que responda a cualquier ordenador ya que puede ser utilizado para realizar ataques DDOS. Para mejorar la configuración definiremos lo siguiente:

Ejemplo 25. Fichero `/etc/bind/named.conf.options`

```
acl autorizados {   ①
    127.0.0.1;      ②
    ::1;            ③
    192.168.X.0/24; ④
};

options {
    directory "/var/cache/bind";
    recursion yes;
    allow-recursion { autorizados; };
    dnssec-validation no;
};
```

① Definimos una lista de control de acceso (*access control list*) llamado *autorizados*

② Introduzco en mi grupo a mi dirección *localhost* en IPv4

③ Introduzco a mi dirección *localhost* en IPv6

④ Introduzco a todos los ordenadores de la red `192.168.X.0/24`

Comprobaremos que el fichero es correcto con:

```
named-checkconf /etc/bind/named.conf.options
```

2.7. Configuración del servidor DNS como reenviador (forwarding)

Para configurar un servidor para que reenvíe las consultas a otros servidores (forwading) se edita el archivo `named.conf.options` y mediante el atributo `forwarders` se indican las direcciones IP separadas por punto y coma de los DNS que actuarán como reenviadores (`forwarders`).

Recordemos que las consultas que se reenvían son sólo aquellas para las que el servidor no está autorizado y sus respuestas no están en caché.

Ejemplo 26. Ejemplo de configuración de reenviador

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

acl autorizados {
    127.0.0.1;
    192.168.X.0/24;
    ::1;
};

options {
    directory "/var/cache/bind";

    recursion yes;
    allow-query { autorizados; };
    forwarders {
               1.1.1.1; ①
    };
    dnssec-validation no;
};
```

① Reenviamos las consultas al DNS de cloudfare `1.1.1.1`

Si el servidor está configurado como reenviador, cabe la posibilidad de usar la opción `forward` la cual ya tiene como valor por defecto el valor `first`. Con first el servidor consultará a los servidores indicados con `forwarders` en primer lugar, y si no obtiene la respuesta, entonces intentará buscarla por sí mismo. Si se usa `forward only`, entonces el servidor resolverá consultando solamente a los reenviadores.

Ejemplo 27. Configuración como SOLO REENVIADOR

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

acl autorizados {
    127.0.0.1;
    192.168.X.0/24;
    ::1;
```

```
};

options {
    directory "/var/cache/bind";

    recursion yes;
    allow-query { autorizados; };
    forwarders {
              1.1.1.1; ①
    };
    forward only; ②
    dnssec-validation no;
};
```

① Reenviamos las consultas al DNS de cloudfare 1.1.1.1

② Solo reenviamos no buscamos nosotros mismos.

2.8. Configuración del servidor DNS como maestro

Configuraremos el servidor para una zona de resolución directa y una zona de resolución inversa.

Para configurar el servidor DNS como maestro para una zona de resolución directa y una zona de resolución inversa se debe editar el archivo `named.conf.local`.

Aquí se indican entre otras opciones el nombre de la zona, el tipo de zona (*master* o *slave*) y el archivo de zona, donde se almacenarán los registros de recursos de la zona. Por ejemplo, si queremos que el servidor tenga autoridad para el dominio `aula.izv` y para la zona de resolución inversa de la red `192.168.X.0/24` editaríamos el archivo `named.conf.local` y le añadimos las zonas directa e inversa:

Ejemplo 28. Servidor maestro en `/etc/bind/named.conf.local`

```
zone "aula.izv" {
    type master;
    file "/var/lib/bind/db.aula.izv";
};

zone "X.168.192.in-addr.arpa" {
    type master;
    file "/var/lib/bind/db.192.168.X";
};
```

Se podría haber especificado cualquier ruta o nombre de fichero en el parámetro `file` pero escogemos la ruta `/var/lib/bind` ya que el servicio `named` dispone de permisos para leer en esta carpeta, pero no en otras.

> El nombre también es arbitrario y en lugar de `db.aula.izv` se podría haber llamado `zona.aula.izv` o `aula.izv.dns` por ejemplo.

Comprobaremos que el fichero es correcto con:

```
named-checkconf /etc/bind/named.conf.local
```

2.8.1. Zona directa

A continuación deberíamos crear los archivos con los registros de recursos de ambas zonas. Los archivos a crear son los que hemos indicado anteriormente en el parámetro `file`.

Si no hubiéramos especificado la ruta absoluta del fichero de la zona, se tomaría el directorio de trabajo por defecto (`/var/cache/bind`).

Ejemplo 29. Zona directa `/var/lib/bind/db.aula.izv`

```
$ttl 38400 ;
aula.izv.    IN    SOA    dns0.aula.izv.    admin.aula.izv. (
     2016101401 ①
     604800     ②
     86400      ③
     2419200    ④
     38400 )    ⑤

aula.izv.         IN    NS      dns0.aula.izv. ⑥
dns0.aula.izv.    IN    A   192.168.X.211  ⑦
winp.aula.izv.    IN    A   192.168.X.212
debp.aula.izv.    IN    A   192.168.X.210
smtp.aula.izv.    IN    CNAME   debp.aula.izv. ⑧
www.aula.izv.     IN    CNAME   debp.aula.izv.
aula.izv.         IN    MX  10  debp.aula.izv. ⑨
```

① nº de versión del archivo de zona

② tiempo de refresco

③ tiempo de reintento

④ tiempo de expiración

⑤ TTL negativo

⑥ servidor maestro autorizado

⑦ correspondencia FDQN a IP

⑧ alias para nombres de dominio

⑨ quien entrega el correo al dominio

Comprobaremos que la zona es correcta con:

```
named-checkzone aula.izv. /var/lib/bind/db.aula.izv
```

Reiniciaremos el servicio.

2.8.2. Zona inversa

Luego tenemos que registrar los correspondientes PTR de la zona inversa.

Ejemplo 30. Zona inversa `/var/lib/bind/db.192.168.X`*;*

```
$ttl 38400 ;
X.168.192.in-addr.arpa. IN SOA dns0.aula.izv. admin.aula.izv. (
    1316802825
    10800
    3600
    604800
    38400 )

X.168.192.in-addr.arpa.        IN    NS     dns0.aula.izv.
211.X.168.192.in-addr.arpa.    IN    PTR    dns0.aula.izv.
212.X.168.192.in-addr.arpa.    IN    PTR    winp.aula.izv.
210.X.168.192.in-addr.arpa.    IN    PTR    debp.aula.izv.
```

Comprobaremos que la zona inversa es correcta con:

```
named-checkzone X.168.192.in-addr.arpa. /var/lib/bind/db.192.168.X
```

Reiniciaremos el servicio.

```
systemctl restart bind9
```

2.8.3. Directivas

En los archivos de zona se pueden incluir ciertas directivas para simplificar el contenido textual:

`$TTL` tiempo

Ajusta el valor Time to Live (TTL) predeterminado para todos los registros de la zona. Este tiempo indica, el tiempo que un registro de recurso de zona es válido. Cada recurso puede contener su propio valor TTL, ignorando el valor esta directiva.

`$INCLUDE archivo`

Indica al demonio named que incluya otro archivo de zona en el archivo de zona donde se usa la

directiva. Así se pueden almacenar configuraciones de zona suplementarias aparte del archivo de zona principal.

`$ORIGIN dominio`

Permite anexar el nombre del dominio indicado en la directiva a aquellos registros de recursos que aparezcan escritos de una manera no cualificada. Por ejemplo, si usamos `$ORIGIN aula.izv.` cualquier nombre utilizado en los registros de recursos que no terminen en un punto (.) tendrán `aula.izv.` anexado.

2.8.4. Alias de zona

También Se puede usar el carácter `@` como sustituto del nombre de la zona definido en `named.conf.local` para no tener que repetirlo en el archivo.

Si en un registro de recursos no aparece en su parte izquierda un nombre de dominio, se usará como nombre de dominio el especificado en el registro anterior.

Si no aparece en el registro el tipo de zona se supone que es tipo IN.

Usando estas directivas, el archivo de zona de resolución directa del ejemplo anterior podría escribirse de la forma:

Ejemplo 31. Zona directa `aula.izv.` usando alias en `/var/lib/bind/db.aula.izv`

```
$TTL 38400;
$ORIGIN aula.izv. ①
@   IN  SOA        dns0.aula.izv. admin.aula.izv. (
    2016101401
    604800
    86400
    2419200
    38400 )

@       IN  NS     dns0.aula.izv.
dns0    IN  A      192.168.X.211
winp    IN  A      192.168.X.212
debp    IN  A      192.168.X.210
smtp    IN  CNAME  debp
www     IN  CNAME  debp
@       IN  MX  10 debp.aula.izv.
```

① Si no lo ponemos tomará el nombre que aparece en `zone` en el fichero `named.conf.local`

La zona inversa utilizando la notación relativa quedaría:

Ejemplo 32. Zona inversa usando alias en `/var/lib/bind/db.192.168.X`;

```
$ttl 38400 ;
```

```
@  IN   SOA     dns0.aula.izv. admin.aula.izv. (
    1316802825
    10800
    3600
    604800
    38400 )

@       IN      NS      dns0.aula.izv.
211     IN      PTR     dns0.aula.izv.
212     IN      PTR     winp.aula.izv.
210     IN      PTR     debp.aula.izv.
```

2.9. Configuración de servidores DNS como esclavos

Tendremos varias equipos, unos de maestros y otros como esclavos. El esclavo mantiene una copia de los registros de zona del maestro. Cuando un registro cambia en el maestro, el cambio se propaga al esclavo.

Para que se transmitan los cambios del servidor maestro al esclavo, es necesario que aumentemos el número de *serial* de la zona directa e inversa. Si no lo hacemos, el esclavo no se actualizará.

2.9.1. Configuración en el esclavo

En el servidor que se configure como esclavo para una zona, debe editarse el archivo `/etc/bind/named.conf.local` e indicar el nombre de la zona (*zone*), el tipo de servidor (*type slave*), cual es el servidor maestro del que recibirá la transferencia (*masters*), y el archivo donde se almacenarán los registros de recursos que se reciban (*file*).

Por ejemplo, supongamos el dominio `aula.izv` del ejemplo anterior donde el servidor maestro es `dns0.aula.izv` con IP `192.168.X.211`; y el equipo donde se va a configurar un servidor esclavo para dicha zona será `dns1.aula.izv` con IP `192.168.X.213`, escribiríamos en el archivo `/etc/bind/named.conf.local` de este servidor esclavo:

Ejemplo 33. Configuración de servidor DNS esclavo en `/etc/bind/named.conf.local`

```
zone "aula.izv" {
    type slave; ①
    masters { 192.168.X.211; }; ②
    file "/var/lib/bind/db.aula.izv"; ③
};

zone "X.168.192.in-addr.arpa" {
    type slave;
    masters { 192.168.X.211; };
    file "/var/lib/bind/db.192.168.X";
```

```
};
```

① El servidor será de tipo esclavo

② Sólo permitiremos actualizaciones desde la IP de nuestro maestro

③ Fichero donde escribimos los datos (en formato binario)

2.9.2. Configuración en el maestro

En el archivo `/etc/named.conf.local` del servidor maestro de la zona, deberemos indicar que se permitan transferencias de zona hacia el servidor esclavo (*allow-transfer*), y además se puede configurar para que notifique a los servidores esclavos los cambios que se produzcan en el maestro (*notify*).

Ejemplo 34. Configuración en el maestro de `/etc/bind/name.conf.local`

```
zone "aula.izv" {
    type master;
    file "/var/lib/bind/db.aula.izv";
    allow-transfer { 192.168.X.213; }; ①
    notify yes; ②
};
```

① Servidores esclavos a los que se puede transferir la zona

② Se notificará a los servidores esclavos de los cambios

Por último, en el archivo de zona del servidor maestro, se debe indicar la existencia de otro servidor DNS para la zona, añadiendo para ello un nuevo registro `NS` con el nombre de dominio del esclavo, y un registro de tipo `A` que permite resolver su IP.

Ejemplo 35. Configuración de `/var/lib/bind/db.aula.izv`

```
aula.izv.      IN NS dns1.aula.izv.  ①
dns1.aula.izv. IN A  192.168.X.213   ②
```

① Nombre de dominio del servidor esclavo

② Resolución del nombre del servidor esclavo

Las transferencias de zona se pueden realizar manualmente usado el comando `dig`. Para ello hay que indicar como parámetros: el nombre de dominio o IP del servidor maestro de la zona, el nombre de dominio de la zona a transferir, y el tipo de transferencia a realizar (`axfr` para completa).

Para nuestro ejemplo consultaríamos desde el esclavo al maestro por el registro `axfr`:

```
dig @192.168.X.211 aula.izv axfr
```

2.10. Subdominio delegado

Un subdominio delegado es cuando un servidor DNS cede la resolución de direcciones en un subdominio a otro servidor. Por ejemplo el servidor primario de `aula.izv.` puede ceder el subdominio `sri.aula.izv.` a otro servidor.

Para ello en la zona `aula.izv.` de nuestro servidor primario añadiremos que hay un servidor de nombres delegado y su IP.

Ejemplo 36. Zona `aula.izv`

```
sri.aula.izv.         IN NS alumnos.sri.aula.izv. ①
alumnos.sri.aula.izv. IN A  192.168.X.213 ②
```

① Hay un nuevo servidor NS en el pueblo, forastero

② Dirección IP del servidor del subdominio

Desactivar reenviadores **sólo** para la zona `aula.izv.` Si no lo haces, no funcionará la delegación.

Como ahora el servidor maestro DNS de la zona `aula.izv` no está autorizado para la zona `sri.aula.izv.` (que la lleva el equipo `alumnos.sri.aula.izv`) cualquier pregunta relativa a otros dominios (incluido `sri.aula.izv`) la reenviaría al servidor que aparece en `forwarder` en el fichero `/etc/bind/named.conf.options`.

Como globalmente queremos mantener el reenvío y sólo deshabilitarlo para el dominio `aula.izv.` debemos editar el archivo de configuración de la zona `aula.izv.` (`/etc/bind/named.conf.local`) y anular el reenvío para dicha zona usando una directiva `forwarders` vacía de la forma:

Ejemplo 37. `/etc/bind/named.conf.local` del DNS maestro de `aula.izv.`

```
zone "aula.izv" {
    type master;
    file "/var/lib/bind/db.aula.izv";
    forwarders {}; ①
};
```

① Para esta zona no usar reenviadores

Después de salvar y chequear la zona, hay que reiniciar el servicio `bind`.

En el ordenador autorizado para la zona `sri.aula.izv.`, `alumnos.sri.aula.izv.` haríamos una zona

llamada `sri.aula.izv.` de la que él es maestro y añadiriamos los registros necesarios.

Ejemplo 38. `/etc/bind/named.conf.local` del DNS maestro de `sri.aula.izv.`

```
zone "sri.aula.izv" {
    type master;
    file "/var/lib/bind/db.sri.aula.izv"; ①
    ②
};
```

① Añadimos registros de `sri.aula.izv.`

② Aquí no hace falta limitar los reenviadores porque no delegamos

Luego probaríamos que la delegación funciona resolviendo algún ordenador de la zona `sri.aula.izv` tanto desde el servidor que delega como desde el delegado.

2.11. Configuración cliente

Después de reiniciar el servicio y comprobar que no hay errores, deberíamos configurar los clientes para que usen como servidor DNS la máquina en donde hemos instalado y configurado el servidor.

Cliente sería cualquier sistema operativo que usa nuestro DNS como Debian, Ubuntu o Windows.

2.11.1. Configuración automática con un servidor DHCP

Si nuestra infraestructura cuenta con un servidor DHCP, configuraríamos el fichero `dhcpd.conf` con las líneas:

Ejemplo 39. Fichero `/etc/dhcp/dhcpd.conf`

```
option domain-name "aula.izv"; ①
option domain-name-servers 192.168.X.211; ②
```

① Dominio de búsqueda `aula.izv`

② Dirección IP de nuestro servidor DNS

Luego configuraríamos una conexión entre el servidor DHCP y el servidor DNS. Al pedir el cliente una dirección IP al servidor DHCP, le proporcionamos los datos del DNS y apuntamos en la zona directa e inversa del DNS al nuevo ordenador.

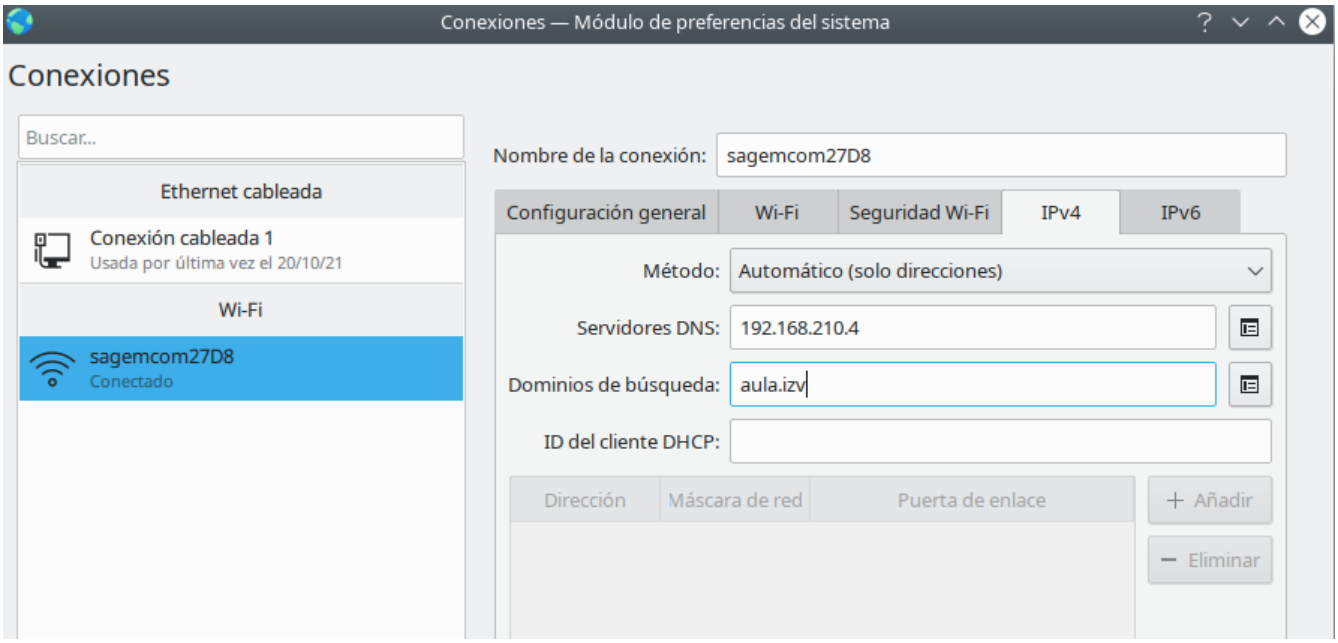
A continuación veremos como hacer la configuración de los clientes de forma manual.

2.11.2. Linux gráfico

El aspecto de la configuración de un Linux gráfico puede cambiar dependiendo de la distribución y del sistema de ventanas.

Generalmente:

1. Pulsaremos en la conexión de red.
2. Pulsaremos en configurar.
3. Escogeremos *Automático sólo direcciones*.
4. En dominios de búsqueda pondremos nuestro dominio.



2.12. Linux texto

Editaremos el fichero `/etc/network/interfaces` y añadiremos

```
dns-nameserver 192.168.X.211 ①
dns-search aula.izv          ②
```

① Suponemos que la dirección IP de nuestro servidor DNS es `192.168.X.211`

② Suponemos que nuestro dominio es `aula.izv`

Derechos de autor

Colaboradores

- De esta edición: Fernando Raya
- De ediciones previas: Carlos Martín

Comentarios y sugerencias

Puede dirigir cualquier clase de comentario o sugerencia acerca de este documento a la lista de correo del equipo de documentación: fraya@ieszaidinvergeles.org