DNS

1.-Previs

Si sabem el nom d'un ordinador de la nostra LAN, podem esbrinar la seva IP (o viceversa) sense la necessitat d'utilitzar cap servidor Dns: qualsevol sistema Linux té un arxiu de texte anomenat **/etc/hosts**, on es pot afegir una parella IP-nom per línia, de manera que el sistema pot saber, si consulta aquest arxiu, la reciprocitat IP<-> nom sense caler cap servidor Dns. No obstant, és evident que el manteniment actualitzat d'aquest arxiu en múltiples ordinadors és una tasca massa farragosa, a més de què la llista de parelles IP<->nom seria ingovernable degut a la seva llargària, a la vegada que del tot incompleta. De fet, en organitzacions amb un parc d'ordinadors elevat, la posada en marxa d'un servidor Dns local pot simplificar molt l'administració de la xarxa.

Es pot provar com a broma escriure a /etc/hosts la ip d'un servidor d'Internet qualsevol, associant-la a un nom d'un altre servidor que no tingui res a veure (o simplement la IP 127.0.0.1). Es podrà comprovar que a l'hora de posar aquest nom (en un navegador, per exemple), s'anirà en realitat a l'altre ordinador, el que poseeix la IP en qüestió. No obstant, abans de fer aquesta prova, caldrà assegurar-se de què un altre arxiu, **/etc/nsswitch.conf**, estigui correctament configurat. Aquest arxiu, entre altres coses, determina l'ordre en el què el nostre ordinador buscarà la informació IP<->nom. És a dir, ens dirà si primer es busca el contingut de l'arxiu /etc/hosts i si no es troba (o no hi ha la parella buscada) llavors es pregunta al servidor Dns que es tingui configurat, o bé al revés, si primer es pregunta al servidor Dns, i si aquest no respon, es la informació desitjada a /etc/hosts. Aquest ordre el configura la línia següent: *hosts: files dns* . Si "files" va abans que "dns", es mirarà primer a l'arxiu hosts, si no, serà al revés. També s'ha de tenir en compte que determinats programes, com per exemple els navegadors, si tenen configurat un proxy, no intenten realitzar cap resolució de noms i deleguen directament aquesta feina al proxy (això té com a conseqüència que l'arxiu /etc/hosts local no s'utilitzi, atenció).

D'altra banda, de forma similar a l'arxiu /etc/hosts, també existeix l'arxiu /etc/networks, el qual estableix una correspondència entre ips de xarxa i noms de xarxa, però no s'utilitza tant.

Per configurar els servidors Dns als que l'ordinador preguntarà primer, històricament només calia editar l'arxiu **/etc/resolv.conf** i afegir tantes línies *nameserver ipservidorDns* com calguessin (com a mínim,2, per si el primer servidor està caigut). Actualment, però, l'edició manualment d'aquest fitxer es desaconsella per no entrar en conflicte amb els possibles mecanismes que hi puguin haver al sistema (com ara el servei "systemd-resolved" o el servei "NetworkManager", entre d'altres) que estan especialment dissenyats per realitzar les actualitzacions de la llista de servidors DNS que hi ha en aquest fitxer de forma desatesa i automàtica

Una altra línia important de l'arxiu /etc/resolv.conf és la línia *search subdomini.domini* Aquesta Línia el que diu és que quan l'usuari escrigui en qualsevol lloc el nom d'un altre ordinador sense cap cua de domini (veure següent apartat, FQDN), automàticament s'afegirà a aquest nom la cua especificada a la línia search, i a més de forma recursiva. És a dir, si un usuari escriu per exemple: *ping pepito* (on pepito se suposa que és el nom d'un ordinador), i al seu arxiu resolv.conf hi ha la línia *search pepsi.cocacola.com* , el que farà la comanda ping és primer buscar l'ordinador pepito.pepsi.cocacola.com, i si no el trobà, provarà amb pepito.cocacola.com, i si no el troba, provarà pepito.com, i si finalment no el troba, provarà només amb el nom tal qual, pepito. A la línia search es poden escriure més d'una cua de domini diferent, però no és aconsellable perquè podeu intuir que la recursivitat en les proves dels diferents noms fa que la intercomunicació entre les màquines sigui bastant lenta.

2.- Conceptes bàsics

Els documents oficials que defineixen el protocol DNS són els RFC (Request-For-Comments) següents: 1033, 1034, 1035,1032(que explica com registrar noms), 1918,1912,1713,1712,2052.Es poden consultar a http://www.rfc-editor.org

FQDN (Fully Qualified Domain Name)	És el nom complet d´una màquina, incloent-t'hi tota la cua de dominis a la què pertany (per exemple: hostname.subdomini1.subdomini2.domini). L'anomenat "domini d'últim nivell" és el que s'escriu a la dreta del punt de més a la dreta del FQDN.
Zone	Básicamente es un archivo con una lista de parejas <nombre>/<dirección IP> . Puede contener la información sobre un dominio completo (y posiblemente, sus subdominios, si es que no se delegan éstos en otros servidores), o bien sobre más de uno (rafa.com, dani.org...) o bien sobre parte de uno.
Resolver	Es la parte cliente del servicio DNS. Su función es la de realizar las peticiones solicitando la dirección IP de un nombre. Prácticamente todas las aplicaciones de red (desde un ping hasta un gestor de correo, etc,etc) incluyen una librería/componente que realiza esta función.
Nameserver	Es la parte servidora del DNS. Su función es la de contestar a las peticiones de nombres recibidas. Escucha normalmente en el puerto 53 Udp. Sólo en determinados casos (en una transferencia de zona, o bien cuando la petición supera los 512 bytes -extraño-), se utilitzará el puerto 53 Tcp.
Authoritative nameserver	Servidor responsable de una zona (el que contiene físicamente los archivos de zona, y donde el administrador los modificará si es el caso). Para cada una de las zone que puedan existir debe haber siempre como mínimo un servidor que se encargue de su gestión. Las respuestas relacionadas con su zona que dará a los clientes se llaman respuestas "autoritativas", porque son respuestas "de primera mano"
Recursive nameserver	Si un nameserver recibe una petición de resolución y no conoce la respuesta porque no posee el archivo de zona con la información pedida, realiza a su vez todas las peticiones que sean necesarias a otros servidores Dns para acceder a la zone correspondiente que contenga la respuesta, de manera que le pueda dar al cliente la información aunque no disponga de ella en un primer momento. Esta capacidad recursiva es configurable, de forma que no siempre los servidores realizarán todo este proceso de búsqueda completa. Si no lo hace, el servidor en cuestión al menos deberá indicar al cliente a qué otro servidor puede preguntar (es decir, "se lava las manos",en lo que se conoce como "mecanismo de delegación"), con lo que todo el trabajo corre a cargo del cliente, ya que éste tendrá que volver a preguntar al nuevo servidor indicado. Este tipo de consulta no recursivas se llaman consultas **iterativas**.
Cache nameserver	Servidor DNS que no definen ninguna zona: solamente contiene la caché de consultas anteriores a otros servidores DNS, realizadas (casi) siempre de forma recursiva. Por tanto, sus respuestas nunca serán autoritativas. Pero entonces, ¿para qué interesa tener sólo un servidor Dns de caché en una red local? Para que se agilice el acceso a Internet: gracias a que la mayoría de respuestas ya estarán dentro de la caché del servidor local, no se tendrá que salir a fuera para cada petición Dns que se realice,. a duración de la información en la caché del servidor viene definida por un campo de su configuración llamado TTL (Time To Live). Por otro lado, el "resolver" del cliente también tiene una caché (en Windows se puede consultar y borrar respectivamente con: *ipconfig /displaydns* y *ipconfig /flushdns* ; en Linux ya lo veremos).
Forwarding nameserver	Servidor DNS que simplement passa totes les peticions a un altre servidor DNS remot (possiblement de catxé). La gràcia està en que els servidors forwarding guarden les respostes (és a dir, funcionen també com a servidors de catxé) però no realitzen les consultes recursives sinò que li "passen el marró" al servidor DNS remot.
Root server	El sistema de DNS proporciona un conjunto básico de servidores (también llamados ".") de forma que a partir de ellos se pueda resolver cualquier petición. Esta lista de servidores básicos es "fija" (consta de 13 servidores por todo el mundo, nombrados mediante letras del abecedario), y está permanentemente almacenada en todos los softwares servidores Dns existentes (Bind, Dnsmasq,etc) Se puede consultar su existencia en http://www.root-servers.org
gTLD & ccTLD	El Top Level Domain es el servidor que contiene toda la información referente a un dominio de último nivel, los cuales pueden ser de dos tipos: geográficos (Country Code TLD) o genéricos

(Global TLD). Un ejemplo de los primeros sería el ".es" y uno de los segundos, el ".com". Existe un número limitado de TLD para cada uno de los dominios, generalmente mantenidos por gobiernos o entidades reconocidas sin ánimo de lucro (respectivamente). La lista se puede consultar aquí: http://www.iana.org/domains/root/db

Zone transfer Transferencia de la información que contiene un servidor **maestro (primario)** en sus ficheros de zona, a otro llamado **esclavo (secundario)**. El servidor maestro es el que mantiene la zona (lee la información del fichero y puede hacer modificaciones en ella), y el servidor secundario simplemente coge los datos periódicamente del primario y los mantiene en su memoria RAM.El servidor secundario no puede hacer modificaciones en la zona, pero tiene la misma autoridad que el maestro: la única diferencia es que en el servidor maestro probablemente los archivos de zona están guardados físicamente y pueden editarse de forma manual mientras que en el esclavo no existen los ficheros físicos porque su información ha sido descargada directamente a su memoria RAM.

El motivo de la existencia de los servidores esclavos es la disponibilidad del servicio Dns: si en algún momento uno de los dos servidores cae, el otro puede seguir trabajando sin que el servicio se resienta. De hecho, el RFC 2182 requiere que existan al menos dos servidores DNS por cada zona

No confundir servidor secundario con servidor de caché: no tienen nada que ver. El primero tiene tanta autoridad como el primario (sólo que sus zonas las obtiene de éste), y el segundo sólamente puede responder por la información que anteriormente obtuvo de otras peticiones a servidores autoritativos.

Resource record Además de la traducción <nombre> / <dirección IP> que ya hemos comentado, los nameservers también pueden proporcionar otros tipos de información útil al sistema. Ejemplos de este aspecto puede ser la obtención de los servidores de correo de un dominio (registros MX) u otros.

Per una explicació més detallada podeu consultar les entrades https://www.digitalocean.com/community/tutorials/a-comparison-of-dns-server-types-how-to-choose-the-right-dns-configuration i https://www.digitalocean.com/community/tutorials/an-introduction-to-dns-terminology-components-and-concepts

3.-Procés d´una petició de resolució de noms

Para ejemplificar el funcionamiento del DNS realizaremos una petición de "ping" a www.rediris.es, por ejemplo.

1.-Lo primero que realizará nuestro ordenador es enviar una petición a nuestro servidor de nombres preferido solicitándole la dirección IP de www.rediris.es . Obviamente para nuestro ejemplo debemos asumir que ninguna petición se encuentra en la memoria caché (lugar dónde se guardan las peticiones ya realizadas) de los servidores. Esto nos permitirá analizar en detalle todo el proceso.

2.-Una vez solicitada la petición a nuestro servidor de nombres, éste comprueba que no conoce la respuesta y consulta entonces su base de datos interna de Root servers (todas las aplicaciones servidoras Dns la llevan incorporada de serie) para elegir uno al azar. Este mecanismo de selección aleatorio tiene como objetivo evitar el colapso de un único servidor balanceando las peticiones entre los distintos elementos del conjunto. Una vez seleccionado uno de los servidores genera una petición para reenviar la consulta sobre el nombre de www.rediris.es. El Root server que recibe la petición comprueba que no conoce la respuesta y entonces responde proporcionando un servidor GTLD que gestiona el dominio solicitado (".es" en nuestro caso).

Simplificando mucho, imagina que la "zona" que controla un Root Server es simplemente una lista de ips de los distintos servidores Dns GTLD que conoce, a los cuales preguntará cuando reciba una petición que corresponda con su GTLD correspondiente,asi.Por ejemplo:

.com	14.164.201.78
´	32.56.143.234
	89.67.122.236
.org	67.45.111.132
	154.213.213.2
.net	56.34.213.143

3.-Una vez recibido la IP de un servidor autoritativo GTLD del dominio ".es" nuestro Recursive nameserver le envía una nueva petición de solicitud de resolución de nombre para www.rediris.es. El servidor GTLD comprueba que no dispone de la respuesta y entonces busca en su base de datos qué servidores se encargan del dominio de segundo nivel de la petición (rediris.es en nuestro caso).

Simplificando mucho, imagina que la "zona" que controla un servidor GTLD es simplemente una lista de ips de los distintos servidores Dns de segundo nivel para ese GTLD, a los cuales preguntará cuando reciba una petición que corresponda con ese dominio correspondiente,asi. Por ejemplo, para un servidor GTLD .edu:

pepito.edu	14.164.201.78
	32.56.143.234
	89.67.122.236
manolito.edu	67.45.111.132
	154.213.213.2
jaimito.edu	56.34.213.143

4.-Una vez más nuestro Recursive nameserver recibe una contestación negativa a su pregunta junto con un nuevo servidor al que debe preguntar. De esta forma vuelve a realizar la petición de resolución de nombre para www.rediris.es y la envía al servidor encargado de gestionar el dominio "rediris.es". Finalmente recibe una contestación afirmativa dónde se explicita la dirección IP asociada a la máquina con nombre www.rediris.es (130.206.1.22).

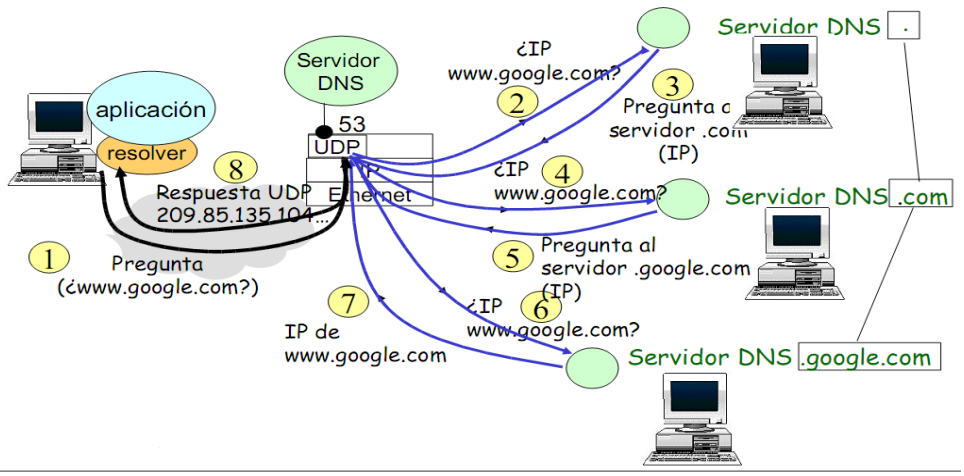
Simplificando mucho, imagina que la "zona" que controla el servidor cocacola.com es simplemente una lista de ips de las distintas máquinas que forman ese dominio. Cuando reciba una consulta preguntando sobre una máquina concreta de su dominio, responderá devolviendo esa ip. Por ejemplo:

www.cocacola.com	14.164.201.78
smtp.cocacola.com	67.45.111.132
unpc.cocacola.com	154.213.213.2
otropc.cocacola.com	56.34.213.143

5.-Una vez resuelta la petición original por el Recursive nameserver este envía la respuesta al usuario inicial, que realizará a su vez una conexión a la dirección IP proporcionada.

6.-Cabe destacar que el sistema de caché de DNS se encarga de almacenar las últimas peticiones que se realizan por los usuarios de forma que no sea necesario realizar este proceso incontables veces cada segundo para cada usuario. Pero por otra parte, el sistema de DNS marca cada una de estas respuestas con un período de validez finito de forma que los nuevos cambios que se pudieran realizar en alguna zona puedan ser "visibles" al cabo de un período de tiempo concreto.

7.-En el caso del envío de correo electrónico, el proceso es similar: una vez que el correo a emitir ha llegado al servidor SMTP predefinido en el cliente, ¿cómo se le entrega al destinatario?. Pues el servidor SMTP solicita al servidor DNS que tenga configurado el valor del registro MX del dominio del receptor del mensaje; cuando lo recibe, se conectará al puerto 25 y lo entregará, aunque el proceso puede continuar si el receptor del mensaje no es el último de la cadena, ya que existen servidores que actúan como gateways entre dominios.



4-Informació que pot haver emmagatzemada en una zona

Els arxius de zona no només hi ha informació sobre què nom correspon a cada ip, hi ha més informació. Per identificar el tipus d'informació, s'especifica el seu "tipus" amb un codi clau ("A", "MX",etc), que indica quina informació és la que hi ha emmagatzemada en concret.

Registre A	Tradueix noms de hosts del domini a IPs. És la informació "per antonomàsia" que esperem trobar en qualsevol arxiu de zona.
Registre AAAA	El mateix que l´anterior però per IPv6
Registre CNAME	Indica un alias para un host determinado (definido por A)
Registre MX	Indica el servidor SMTP encargado de recibir el correo elect para ese dominio(ver punto 7 anterior)
Registre NS	Indica el/los servidor/es de nombres autoritativos de un dominio (los servidores primarios y secundarios que mantienen la zona). Permite así que los diferents servidores Dns se reconozcan entre sí cuando se realicen consultas recursivas entre ellos, ya que lo que mirarán será el valor de este registro para saber a quién tienen que preguntar. Este registro también permite la delegación de dominios, ya que si se produce, existirá (además de los servidores primario y secundarios ya comentados del dominio superior) un registro NS por cada responsable de cada subdominio delegado.
Registre HINFO	Añade información textual adicional relativa a un host del dominio, como su CPU o su SS.OO
Registre TXT	Añade información textual adicional (se puede usar por ej. para almacenar claves de cifrado, o para filtrar spam mediante servidores de listas negras DNSBL, o para usar VPNs)
Registre PTR	Traduce IP a nombres de hosts del dominio (registro inverso)
Registre SOA	Indica vàries dades administratives importants, com ara la direcció de correu de l´administrador, el número de sèrie del domini, paràmetres d´actualització de la zona,etc). La seva existència per si sola ja indica que el servidor que conté l´arxiu de zona on apareix és un servidor autoritatiu.
Registre SRV	Informa de serveis específics disponibles a través del domini. Ho fan servir protocols com SIP o XMPP per a què els clients descobreixin de forma automàtica els serveis disponibles al domini.
Registre ANY	Tots els registres (utilitzat per dig)

Veure http://www.iana.org/assignments/dns-parameters

5-Comandes clients: dig i host

Són comandes del paquet bind-utils que fan de resolvers pur i durs, i ens permeten per tant obtenir informació emmagatzemada als servidors Dns que preguntem (sobre el domini que preguntem) d´una forma directa.

La comanda dig funciona així: *dig [@servidorDNS] nom.maqu.ina TIPUS [opcions]*

on @servidorDns és el servidorDns al que es preguntarà (es pot fer servir qualsevol servidor DNS públic disponible com ara els del nostre ISP, els de Google -8.8.8.8 o 8.8.4.4- o qualsevol altre dels llistats a http://public-dns.info: són tots equivalents) ; si no s´especifica cap, s´utilitzarà el primer definit a resolv.conf). "nom.maqu.ina" és el nom de la màquina del qual es vol saber la informació especificada amb TIPUS (on TIPUS pot valer qualsevol dels tipus de registre Dns -A,AAAA, CNAME,..- si no es posa cap, s´interpreta un tipus A). Finalment, es poden indicar una sèrie d´opcions per modificar la recerca. Per exemple:

+tcp : s´utilitza Tcp i no Udp per fer la petició
+norecurse: es vol fer una petició no recursiva
+short : es vol obtenir un resum de la informació rebuda (similar a +nostats, +nocomments i algun més)
+multiline: el contrari de +short
+trace : mostra tots els passos recursius que el servidor utilitzat farà per donar-nos la resposta
+nocomments : no es vol obtenir comentaris extres a la informació rebuda
+nocmd : no es vol obtenir el comentari inicial que indica la versió de dig
+noquestion : no es vol obtenir la secció de Question a la informació rebuda
+nostats : no es vol obtenir estadístiques sobre la informació rebuda
+noall +answer : només s´obté la secció d´Answer a la informació rebuda, i res més.
+time=3 : s´esperarà tres segons a rebre la resposta abans de tornar-ho a intentar (o desistir)
+tries=3 : es farà un màxim de tres intents de consulta

En el cas de voler realitzar consultes inverses (és a dir, preguntar sobre els registres PTR), la sintaxis és: *dig -x [@servidorDNS] ip [opcions]* (fixeu-vos en la presència del *paràmetre -x*).

En el cas de voler obtenir informació sobre els servidors DNS autoritatius d'un domini determinat, la sintaxis és: *dig [@servidorDNS] dom.ini NS [opcions]* . Per exemple, si es vol obtenir la llista de Root Servers, es podria fer: *dig . NS* . Si es vol obtenir la llisat de servidors GTLD d´un domini concret es pdoria fer: *dig net. NS* (una altra manera seria: *dig +norec +noques +nostats +nocmd dom.ini @A.ROOT-SERVERS.NET*)

També es pot especificar com a tipus de registre "AXFR", obtenint així una transferència de zona.

Intentem desxifrar la resposta que ens dóna la comanda. Per exemple, si preguntem pel host www.ignside.net, obtenim això:

```
; <<>> DiG 9.4.0-(Hawk)-8.02 <<>> www.ignside.net
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1580
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;www.ignside.net.            IN      A
;; ANSWER SECTION:
www.ignside.net.       2886    IN      CNAME   irvnet.nexenservices.com.
irvnet.nexenservices.com. 6486  IN      CNAME   sauterne.nexen.net.
sauterne.nexen.net.    486     IN      A       217.174.203.10
;; Query time: 15 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Thu Jul 28 22:33:36 2005
;; MSG SIZE  rcvd: 116
```

Las respuestas que comienzan por ; son comentarios introducidos por dig, no provienen del servidor Dns. En las dos primeras líneas, dig se limita a informar de la versión del programa en ejecución y del dominio objeto de consulta. La línea `;; global options: printcmd` se refiere a las opciones generales usadas en la consulta. Puedes evitar estas dos lineas utilizando la sintaxis de consulta dig +nocmd nombredominio.com

La siguiente seccion `Got Answer` nos ofrece detalles de la consulta recibida, entre ellos, el número de respuestas recibidas, y si ésta nos la ha dado o no una servidor con autoridad en el dominio solicitado. Las banderas (flags) nos dan detalles de la consulta y respuesta: `QR` (Query/Response) sirve para diferenciar la consulta de la respuesta. `RD` (Recursion Desired), es una modalidad de la consulta, que es replicada o no en la respuesta con la bandera `RA` (Recursion Allowed), y significa que pedimos al server que si no puede resolver la respuesta por si mismo, consulte recursivamente a otro server. La aceptación de la petición por el server es opcional. `AA` significaría que la respuesta es de un server autoritativo. Otras flags son: `TC` (Truncated Response), que significa que la respuesta se ha fraccionado por ser de mayor tamaño del permitido, AD (Authentic Data) y CD (Checking Disabled).

La tercera sección nos da detalles de la consulta; además como es obvio del dominio consultado, nos informa que estamos consultando en los registros A (si no se indica ningún registro concreto, será éste el que se solicitará). Como ya sabemos, si indicase MX en su lugar querria decir que estamos consultando una dirección de email.

En el ejemplo que hemos puesto mas arriba, la respuesta tiene varias lineas. La línea `www.ignside.net. 2886 IN CNAME irvnet.nexenservices.com.` nos dice que el nombre por el que preguntamos es un *alias* (CNAME) de irvnet.nexenservices.com; La siguiente linea nos indica que irvnet.nexenservices.com es un alias de sauterne.nexen.net, y la tercera linea nos indica la IP de sauterne.nexen.net.
El que ninguna de las respuestas que hemos obtenido sea AUTHORITY no dice nada sobre la fiabilidad de la respuesta, simplemente que el server que nos la ha dado no es responsable del dominio. Finalmente podemos encontrarnos con una sección de respuestas adicionales, que típicamente nos informaría de las ips de los nameserves devueltos en la sección AUTHORITY, si los hubiera.

La última sección nos explica el tiempo que ha tardado en resolverse la consulta, el número en bytes de la respuesta, la fecha y el servidor dns consultado.

Per altra banda, la comanda host funciona així: *host [-t tipusregistre] domini [servDns]*, i el seu significat és similar als paràmetres del dig. També té el paràmetre -l, que serveix per llistar la informació de tots els registres

6.-Servidor Dnsmasq

Un servidor DNS de catxé és un servidor DNS que no conté cap informació autoritzativa ell per sí mateix sinó que es limita a remetre les peticions que li fan a un altre servidor DNS (configurat com a "forwarder" ja sigui dins de la pròpia configuració del servidor DNS catxé o bé directament indicat dins de /etc/resolv.conf), el qual sí serà qui respongui. La gràcia llavors és que el servidor DNS de catxé, com diu el seu nom, es guardarà la resposta obtinguda a la seva memòria de forma que una altra petició posterior ja la podrà respondre ell mateix sense haver de tornar a preguntar. Les respostes es guarden en memòria catxé un temps configurable.

El programa Dnsmasq (http://www.thekelleys.org.uk/dnsmasq/doc.html) és un servidor **DHCP**(+TFTP/PXE) **+DNS de catxé** (preguntant per defecte als servidors DNS indicats a /etc/resolv.conf) encara que **també pot funcionar com servidor DNS autoritatiu del contingut present a l'arxiu /etc/hosts local.**

Dnsmasq es gestiona com un dimoni típic (*systemctl status dnsmasq*) i la seva configuració es troba per defecte a l'arxiu /etc/dnsmasq.conf (encara que la seva ruta i/o nom es pot canviar amb el paràmetre –conf-dir i/o –conf-file del binari, respectivament) ; allà podem trobar les següents línies relacionades amb el servei DNS (de catxé+autoritatiu):

NOTA: Si volguéssim deixar només funcionant el servei DNS (de catxé+autoritatiu) de Dnsmasq deshabilitant el seu servei DHCP, només cal que a l'arxiu /etc/dnsmasq.conf no aparegui cap línia de les que comencen per "dhcp-*" i que la línia port valgui 53.

listen-address=IPfixaServidorDnsmasq *bind-interfaces*	La directiva listen-address= indica la direcció IP fixa associada a la tarja de xarxa a través de la qual escoltarà les peticions dels clients DNS (i les respondrà). Es poden indicar més d'una línia listen-address= . Acompanyant a aquesta directiva cal afegir sempre la directiva bind-interfaces. Si no s'indica cap d'aquestes directives, Dnsmasq escolta per defecte a través de totes les tarjes que tinguin IP fixa associada. Alternativament a listen-address= (però indicant igualment bind-interfaces) es pot fer servir la directiva interface=nomTarja . Es poden indicar vàries tarjes escrivint més d'una línia interface= o bé usant el comodí '*'
port=53	Indica el port per on escoltarà el servidor DNS les peticions entrants. El valor estàndar és 53.
resolv-file=/ruta/fitxer	Opcional. Indica la ruta del fitxer que s'utilitzarà en comptes de /etc/resolv.conf per "buscar" els servidors DNS "forwarder". Aquest fitxer ha d'incloure línies amb el format "nameserver x.y.z.w"
no-resolv	Si està present, indica que no es farà servir l'arxiu /etc/resolv.conf (ni cap altre indicat a resolv-file=) per "buscar" els servidors DNS "forwarder". Aquesta directiva hauria de venir acompanyada de la directiva server= si es volgués poder seguir utilitzant Dnsmasq com un servidor DNS de catxé.
server=ip.serv.DNS	Opcional. Indica directament la IP del servidor DNS "forwarder" que Dnsmasq farà servir. Si es vol indicar més d'ún cal escriure una línia *server=...* per cadascun És possible especificar un domini concret així: server=/dom.ini/IP; d'aquesta manera, només s'usarà el servidor DNS indicat quan la consulta DNS en qüestió pertanyi al domini especificat. Això pot fer-se servir per apuntar directament a servidors DNS que sabem que són autoritatius d'un determinat domini.
addn-hosts=/ruta/fitxer	Opcional. Indica la ruta del fitxer que s'usarà com a font de "registres A/AAAA" en comptes de l'arxiu /etc/hosts (o a més de, segons si està present o no la directiva no-hosts, respectivament)
no-hosts	Si està present, indica que no es farà servir l'arxiu /etc/hosts com a font de "registres A/AAAA". Aquesta directiva hauria de venir acompanyada de la directiva addn-hosts= (o bé de la directiva address=) si es volgués poder seguir utilitzant Dnsmasq com un servidor DNS autoritatiu.
address=/nom/nom.dom.ini/IP	Opcional. Indica directament un "registre A" (en concret, l'associació nom + nom.domini + IP per una màquina concreta) sense haver de tenir-lo escrit a /etc/hosts.

	És possible no especificar cap nom concret i només indicar un nom de domini, així: address=/nom.dom.ini/IP ; d'aquesta manera, qualsevol nom que acabi amb el domini indicat serà "col.lapsat" a la IP indicada. Això pot fer-se servir com un sistema artesanal de bloqueig de dominis.
domain=dom.ini	Opcional. El domini DNS del qual Dnsmasq és autoritatiu. Per xarxes LAN sense connexió a Internet es pot configurar lliurement.
local=/dom.ini/	Opcional. Similar a la directiva anterior. És recomanable especificar les dues
expand-hosts	Si està present, no caldrà afegir manualment a cada nom present dins de /etc/hosts el domini indicat a la directiva domain=
domain-needed	Si està present, no reenvia als servidors DNS "forwarders" cap petició de noms que vagin sense domini (és a dir, no reenvia nom de l'estil "pepe" però sí "pepe.domini.com"). D'aquesta manera s'eviten consultes inútils.
bogus-priv	Si està present, no reenvia als servidors DNS "forwarders" cap petició inversa corresponents a direccions IP privades. D'aquesta manera s'eviten consultes inútils.
cache-size=nº	Indica el número de noms de màquina que catxejarà. Per defecte val 150.

7.-Servidor Bind

*Per instal.lar-lo a Ubuntu: *apt install bind9* .També es pot instal.lar *bind9utils* , que són eines client per depurar el comportament del servidor, i *bind9-doc*, que és la documentació.

*Per instal.lar-lo a Fedora: dnf *install bind* (inclou la documentació). També es pot instal.lar *bind-utils* i el paquet *bind-sdb*, que és una versió del servidor compilada per suportar gestió de zones dins de bases de dades PostgreSQL ó SqlLite i de servidors LDAP

*Per iniciar/aturar el servidor, com sempre: *sudo systemctl {start|stop} bind9* A Fedora seria: *sudo systemctl {start|stop} named*

*L'arxiu /etc/bind/named.conf és l'arxiu principal de configuració del servidor Bind. Si observem el seu contingut a Ubuntu, veurem que només consta de tres línies “include” que apunten a tres arxius respectius. La línia “include” el que fa és “copiar-pegar” el contingut dels arxius referenciats com si estigués escrit directament sobre l'arxiu “mare”. Això es fa per separar diferents seccions de configuració en diferents fitxers i tenir-ho tot una mica més ordenat. En concret, a named.conf s'apunten als fitxers:

NOTA: Els comentaris a tots els fitxers de configuració poden ser estil C (// ó /* */) o estil Unix (#).

/etc/bind/named.conf.options	Conté una secció “options” amb opcions generals del servidor, que ara veurem. Allà es pot configurar per a què actuï com a servidor de catxé o com un simple servidor "forwarding"
/etc/bind/named.conf.local	Fitxer buit que està reservat per a què nosaltres escriguem les nostres zones. És a dir, allà es pot configurar per a què actuï com a servidor autoritatiu.
/etc/bind/named.conf.default-zones	Fitxer amb unes quantes zones ja predefinides necessàries per a què Bind funcioni correctament i que no caldrà modificar mai.

8.-Conifguració com a servidor de catxé

Primer llistarem els paràmetres més importants que podem escriure dins de la seció "options" del fitxer /etc/bind/named.conf.options, i a partir d'aquí realitzarem la configuració desitjada.

port nº ;	Port on escoltarà el servidor Dns
listen-on {llista_ips;};	Especifica quina tarja de xarxa farà servir (si en tingués vàries) per escoltar peticions.
directory "/ruta/arxiu" ;	Ruta per defecte (si no s´especifiquen rutes absolutes en les directives d'aquest arxiu), També és la ruta on es troba la catxé
recursion yes | no ;	Indica si realitzarà recerques recursives a tot l´arbre Dns fins trobar la resposta (si no, ho ha de fer el client). El valor per defecte és "yes"
forwarders {llista_ips;};	Indica una llista de servidors Dns als que es reenviarà una petició (a través d'una interfície i port concret si s'indica amb la directiva **query-source{}**) si no se sap la resposta per a què ells facin la recerca recursiva (és a dir, anar als root-server, etc) en comptes de fer-la hom mateix
forward first| only ;	Indica si consultarà als reenviadors abans de mirar a la seva pròpia configuració, o si només preguntarà als reenviadors (per defecte el seu valor és "first")
allow-query {llista_ips;};	Limita els ordinadors o xarxes que poden realitzar peticions al Dns actual. Es pot usar acls ("any", "none"...) -explicades més endavant-
allow-recursion {llista_ips;};	Limita els ordinadors o xarxes que poden utilitzar el DNS actual per fer consultes recursives (només vàlid si recursion és "yes"). Es pot usar acls ("any", "none"...) -explicades més endavant-
allow-transfer {llista_ips;};	Limita els servidors DNS que poden ser esclaus d'hom (és a dir, que poden demanar una transferència de zona). Es pot usar acls ("any", "none"...)
blackhole {llista_ips;};	Especifica les direccions IP d´equips als què el servidor no contestarà.
cleaning-interval nº ;	Especifica l´interval de temps en el que les correspondències obtingudes romanen a la catxé del servidor abans de ser eliminades.

Fora de la secció "options" podem afegir l'opció "acl", així:

acl miacl { !192.168.1.25; 192.168.1/24 ; 192.168.2.13 };

Permet definir llistes de direccions per ser utilitzades posteriorment. Per ex. (el ! indica que no està inclosa):
Ja existeixen acls predefinides: "any", "none", "localhost", "localnets".
Aquestes acl es poden posar a directives del tipus allow-recursion, allow-transfer, etc. P. ex: *allow-query { miacl ; } ;*

Per més informació, consultar el Administrator's Reference Manual, disponible a la web de Bind. També es pot consultar http://www.zytrax.com/books/dns/

A partir d'aquí, per tenir un servidor de catxé cal que el fitxer named.conf.options tingui un contingut similar a aquest (l'acl no és obligatòria però molt recomanable per limitar l'abús en la realització de consultes recursives...de fet, en realitat, no caldria realitzar cap configuració especial perquè ja d'entrada el servidor Bind ja funciona com a servidor de catxé sense haver de fer res):

```
acl bonsclients { 17.85.0.0/16;  localhost;  localnets;  };
options {
    directory "/var/cache/bind";
    recursion yes;  #Encara que ja ho sigui per defecte, ho explicitem
    allow-recursion {bonsclients;};
    ...
}
```

És important, abans de reiniciar el servidor, comprovar que la configuració està ben escrita amb la comanda *sudo named-checkconf*. Per provar que el servidor funcioni, simplement caldrà executar des d'una màquina client qualsevol una comanda semblant a *dig @ip.nostre.servidor.DNS un.nom.dns.qualsevol* . Es pot confirmar que el nom indicat es troba a la catxé del servidor DNS amb les comandes r*ndc dumpdb ; grep "un.nom.dns.qualsevol" /var/cache/bind/named_dump.db* (veure apartat nº11).

9.-Conifguració com a servidor "forwarding"

L'arxiu /etc/bind/named.conf.options que hauríem de tenir per convertir el nostre servidor de catxé en un servidor "forwarding" són les següents:

```
acl bonsclients { 17.85.0.0/16;  localhost;  localnets;  };
options {
    directory "/var/cache/bind";
    recursion yes;  #Es deixa a "yes" perquè hi ha determinades consultes en les que convé que sigui així
    allow-query {bonsclients;};
    forwarders { 8.8.8.8; 8.8.4.4; };
    forward only;
    ...
}
```

NOTA: Si en provar aquesta configuració els servidors dónen errors en els logs, es pot provar de canviar la línia dnssec-validation auto; que ve per defecte per aquestes dues:

dnssec-enable yes;
dnssec-validation yes;

És important, abans de reiniciar el servidor, comprovar que la configuració està ben escrita amb la comanda *sudo named-checkconf* . Per provar que el servidor funcioni, simplement caldrà executar des d'una màquina client qualsevol una comanda semblant a *dig @ip.nostre.servidor.DNS un.nom.dns.qualsevol* . Es pot confirmar que el nom indicat es troba a la catxé del servidor DNS amb les comandes r*ndc dumpdb ; grep "un.nom.dns.qualsevol" /var/cache/bind/named_dump.db* (veure apartat nº11).

10.-Configuració com a servidor autoritatiu

Primer confirmem que la configuració general del nostre servidor a /etc/bind/named.conf.options sigui la correcta:

```
acl bonsclients { 17.85.0.0/16;  localhost;  localnets;  };
options {
    directory "/var/cache/bind";
    recursion yes;  #Podria ser "no" i llavors tindríem un servidor només autoritatiu (no caldria llavors allow-recursion)
    allow-recursion {bonsclients;};
    allow-transfer {none;}   #Per evitar fuges d'informació
    ...
}
```

Seguidament, hem de definir les zones pròpies dins els arxius de zona respectius . Per fer això, és recomanable utilitzar l´arxiu /etc/bind/named.conf.local. Un exemple:

```
zone "tierramedia.com" IN {
        type master;
        file "/etc/bind/mizona.txt";
};
zone "1.168.192.in-addr.arpa" IN {
        type master;
        file "/etc/bind/mizonainversa.txt";
};
```

Es pot observar la presència de la zona pròpia “tierramedia.com” -zona directa- i “1.168.192.in-addr.arpa”. -zona inversa- . Les zones directes contenen la informació necessària per obtenir, a partir d´un nom de domini conegut, la IP corresponent. Les zones inverses contenen la informació necessària per obtenir, a partir d´una IP coneguda, el nom de domini corresponent. El servidor Bind gestionarà per a cada domini dos fitxers diferents, un per la zona directa i un altre per la inversa.

NOTA: És important saber que el nom que tingui el fitxer que conté la informació en sí de cada zona no és gens rellevant, perquè el nom de la zona ve determinat pel valor escrit després de zone, no pel nom del fitxer.

NOTA: Los nombres de las zonas inversas han de seguir unas normas concretas. Básicamente, no se han de especificar los bytes reservados para número de estación. Esto no tiene nada que ver con máscaras ni nada: simplemente se utilitza el mismo sistema que con los dominios directos: entre cada punto hay un dominio superior de servidores Dns, y lo que aparece a la izquierda del punto más a la izquierda es ya la máquina concreta. De hecho, la IP està al revés para ir de derecha a izquierda de más global a más concreto, tal como se hace en los dominios directos. Tanto el dominio "." como su tld "arpa" como el subdominio "in-addr" son gestionados por la ICANN y siempre aparecerán como la raiz de todos los dominios de zona inversa, a la derecha de todo.

Abans de res, s´ha de tenir unes quantes coses clares a l´hora d´entendre la sintaxis d´aquest fitxers:

*Els fqdn SEMPRE han d´acabar en un punt, i si només s´escriu el nom del host pelat, llavors NO. Això és degut a que Bind afegirà sempre el nom de la zona a qualsevol nom escrit dins del fitxer de zona que no acabi en un punt: per tant, això ens interessarà per poder escriure noms de hosts (sense punts) i tenir-los completament definit, però hem de tenir en compte que si posem un fqdn sense punt, també s´afegirà el nom de la zona al final, cosa que espatllarà les resolucions.

*L'arroba (@) es pot escriure, allà on hagi d'aparèixer, en substitució del nom del domini gestionat per l'arxiu de zona,

*L'asteric (*) es pot escriure en substitució del nom d'un domini qualsevol. Un ús típic es mapejar a una màquina per defecte qualsevol domini que no s'hagi establert explícitament, així: * *IN A 192.168.0.2*)

*Si el valor de més a l'esquerra d'un registre és el mateix en registres consecutius, es poden suprimir tots aquests valors excepte el primer.

*No ha d´haver cap línia en blanc al principi dels fitxers de zona

En el servidor d'exemple, el fitxer de la zona directa *tierramedia.com* pot contenir per exemple això:

```
$ttl 38400
tierramedia.com.    IN    SOA    valinor.tierramedia.com. hostmaster.tierramedia.com. (
                200203194  //anydiamesversio
                10800
                3600
                604800
                38400 )
@           IN    NS     valinor
valinor     IN    A      192.168.1.6
valinor     IN    HINFO  Servidor
moria       IN    A      192.168.1.1
moria       IN    HINFO  Router
galadriel   IN    A      192.168.1.8
proxy       IN    CNAME  galadriel
hobbiton    IN    A      192.168.1.3
@           IN    MX     10  hobbiton  //El 10  indica la prioritat, per si hi ha més servidors de correu  (- nº, + prioritat)
isengard    IN    A      192.168.1.4
mates       IN    A      192.168.1.217
biblioteca1 IN    A      192.168.1.231
www         IN    A      192.168.1.2
```

Se puede ver que el archivo ha de comenzar (esto es igual sea una zona directa o inversa) con una cabecera que especifica algunos parámetros de configuración:

*TTL (*Time To Live*): Periodo de validez para la información contenida en la zona; pasado el tiempo las cachés que hayan obtenido información de ésta debe refrescarse o actualizarse. Se puede afinar para un registro en concreto, si se pone el valor deseado entre el valor de más a la izquierda del registro e "IN".

*Registro SOA (*Start Of Authority*) : Indica que es autoritativo (primario o secundario). El significado de los valores de este registro es:

```
@ IN SOA <Fqdn del servidor DNS responsable de la zona> <correo del admin (cambiando la arroba por . )> (
/*Ha d'incrementar-se amb cada modificació de l'arxiu de zona, d'aquest mode els servidors esclaus podran saber si
cal que demanin una transferència de zona. Es recomana que tinguin un format AAAAMMDDnn */
        <serial-number>
/*Temps que els servidors esclaus deixen passar abans de consultar al servidor mestre per si hi ha canvis a la zona*/
        <time-to-refresh>
/*Tiempo que el esclavo deja pasar antes de reintentar una transferencia de zona si ésta ha fallado.*/
        <time-to-retry>
/*Si un servidor esclavo no consigue actualizar sus zonas mediante la correspondiente transferencia de zona, pasado
este tiempo debe dejar de considerar válida la información de la zona */
        <time-to-expire>
/*Cuando el servidor Dns no puede resolver un dominio lo recuerda el tiempo indicado.*/
        <minimun-TTL>
    )
```

Los valores <time-to-refresh>, <time-to-retry>, <time-to-expire> y <minimun-TTL>, se pueden escribir en segundos o bien en horas (si se añade el prefijo H), en días (si se añade D) o semanas (si se añade W)

A partir d'aquí, la sintaxi general dels registres ja es pot veure que és:

fqdn IN tipoRegistro valor

on recordeu que el fqdn es pot ficar de forma explícita acabant-ho amb un punt, o només el nom del host (sense acabar-lo amb un punt), i valor serà en cada cas diferent, segons el tipus de registre..

D'altra banda, existeixen registres com NS ó MX que tenen una altra sintaxi:

nomzona IN tipoRegistro valor

on nom zona és el mateix que fqdn però sense el nom del host (i recordeu que es pot substituir per @). A més, el registre MX incorpora un camp numèric després de "tipoRegistro" que indica la prioritat si n´hi han més registres MX.

TRUC 1: Si apareix

@	IN	A	192.168.0.2
www	IN	CNAME	@

podem fer referència al servidor 192.168.0.2 bé amb el nom "dominio.com" o bé amb "www.dominio.com"

TRUC 2: Es pot fer un load balancing rudimentari de 3 servidors webs replicats de tal manera:

www	600	IN	A	10.0.0.1
	600	IN	A	10.0.0.2
	600	IN	A	10.0.0.3

TRUC 3: También se puede utilizar dentro de los ficheros de zona el comando $GENERATE, que sirve para crear loops. Por ejemplo: *$GENERATE 0-19 hosts${0,2} A 192.168.0.${10,2}* genera las líneas:

hosts00	A	192.168.0.10
hosts01	A	192.168.0.11
...		
hosts19	A	192.168.0.29

Finamente, a modo de ejemplo, a continuación se incluye un fragmento del fichero correspondiente a la zona inversa de ejemplo anterior:

```
1.168.192.in-addr.arpa. IN    SOA    valinor.tierramedia.com. hostmaster.tierramedia.com. (
        200203177
        10800
        3600
        604800
        38400 )
@    IN  NS    tierramedia.com.
1    IN  PTR   moria  //el nombre de la zona (1.168.192.in...) se añade automáticamente, si no se acaba en un punto!!
3    IN  PTR   hobbiton
```

4 IN PTR isengard
6 IN PTR valinor
217.1.168.192.in-addr.arpa. IN PTR mates.tierramedia.com.
231.1.168.192.in-addr.arpa. IN PTR biblioteca1.tierramedia.com.

donde ahora, @ equivale, como siempre, al nombre de la zona, que en esta caso sería 0.168.0192.in-addr.arpa.

És important, abans de reiniciar el servidor, comprovar que els arxius de zona estiguin ben escrits amb la comanda *sudo named-checkzone tierramedia.com /etc/bind/mizona.txt* (per la zona directa) o bé *sudo named-checkzone 1.168.192.in-addr.arpa /etc/bind/mizonainversa.txt* (per la zona inversa).

11.-Comanda rndc

RNDC stands for Remote Name Daemon Control. It is a name server control utility in bind. This name server control utility allows command line administration of the named service both locally and remotely. It is a command line utility and it controls the operation of a name server. La comanda rndc es pot utilitzar per (executeu la comanda sense paràmetres, o vegeu la pàgina del manual, per saber totes les possibilitats).

-Aturar el servidor (paràmetre stop ó halt)
-Recarregar la configuració (paràmetre reload)
-Veure l'estat del servidor (paràmetre status),
-Volca el contingut actual de la catxé a l'arxiu /var/cache/bind/named_dump.db (paràmetre dumpdb)
-Esborra la catxé sencera (paràmetre flush) o bé un nom concret (paràmetre flushname nom)
-...

Un aspecte interessant és que aquesta comanda pot gestionar també servidors remots des d'un ordinador client normal si s'especifica la IP del servidor amb el paràmetre -s i el port d'administració (53 TCP) a fer servir amb el paràmetre -p . Més informació per aconseguir això es pot trobar a https://tecadmin.net/configure-rndc-for-bind9

El seu fitxer de configuració és /etc/bind/rndc.key. Rndc configuration file specifies which server controls and what algorithm the server should use. To prevent unauthorized users to the named daemon, BIND uses a shared secret key authentication method to grant privileges to particular hosts. It means that an identical key must be present in the configuration file of bind, /etc/bind/named.conf and configuration file, /etc/bind/rndc.key