Jorge Ángel Vázquez y
Juan Antonio Peregrina

Práctica 2 - Domain Name System (DNS)

1. **Comenzamos con la máquina VM1. Realizar peticiones DNS utilizando las utilidades dig y/o nslookup, tanto en resolución directa como inversa, sobre servidores de DNS locales y remotos (servidores de Internet). Llevar a cabo manualmente el proceso iterativo de búsqueda de un recurso de Internet desde los servidores raíz hasta el servidor con autoridad en el dominio a consultar (también puede usar la opción +trace de dig).**

La máquina VM1 será Ubuntu (192.168.160.135).

En primer lugar, se han visualizado los paquetes a actualizar con: apt update y a continuación, con apt upgrade se realiza la actualización de estos. Sucesivamente, se emplea el comando apt install bind9 para instalar el servidor de DNS bind9. Estos pasos de instalación realizan, ya que sin ellos no podemos realizar las operaciones deseadas de búsqueda.

Así pues, se hará una consulta a la dirección que especifica el enunciado de la práctica (www.isc.org.) desde el servidor local de la máquina Debian:

```
root@server:~# dig @localhost www.isc.org.

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost www.isc.org.
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 48526
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: bdacd48e14162aed0100000065e229052c96122da5844ab8 (good)
;; QUESTION SECTION:
;www.isc.org.                   IN      A

;; ANSWER SECTION:
www.isc.org.            300     IN      CNAME   isc.map.fastlydns.net.
isc.map.fastlydns.net.  30      IN      A       151.101.134.217

;; Query time: 896 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Fri Mar 01 19:14:13 UTC 2024
;; MSG SIZE  rcvd: 119

root@server:~#
```

Como se puede observar se recibe una resolución en dos pasos:

Primero, se consulta www.isc.org y se descubre que es un alias para isc.map.fastlydns.net debido a su registro de tipo CNAME para **www.isc.org** con un tiempo de vida (TTL) de 300 segundos.

Luego, isc.map.fastlydns.net. resuelve a la dirección IP 151.101.1.195, con lo que el registro A para el dominio **isc.map.fastlydns.net.** con un TTL de 30 segundos, es el registro de la dirección IPv4 real.

Ahora se pasa a realizar peticiones DNS sobre un servidor de internet (Google). Mediante resolución inversa con un servidor remoto:

```
root@server:~# dig @8.8.8.8 -x 8.8.8.8

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @8.8.8.8 -x 8.8.8.8
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 61873
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;8.8.8.8.in-addr.arpa.          IN      PTR

;; ANSWER SECTION:
8.8.8.8.in-addr.arpa.   18892   IN      PTR     dns.google.

;; Query time: 40 msec
;; SERVER: 8.8.8.8#53(8.8.8.8) (UDP)
;; WHEN: Fri Mar 01 19:53:18 UTC 2024
;; MSG SIZE  rcvd: 73
```

Como se puede ver, en el comando aparece –x lo cual indica que se trata de una resolución inversa al tratar con direcciones IP (8.8.8.8) en la consulta, utilizando ese mismo servidor (8.8.8.8) como el servidor de nombres perteneciente a los DNS públicos de Google.

En cuanto a su resultado:

- No hubo errores en la consulta (**NOERROR**).
- La respuesta proporcionada es el nombre asociado a la dirección IP 8.8.8.8, es decir, **dns.google**.
- El tiempo de consulta fue de 40 milisegundos.
- La consulta se hizo a través de UDP al servidor de nombres 8.8.8.8.
- El tamaño del mensaje de respuesta recibido fue de 73 bytes.

Ahora se mostrará una consulta de resolución inversa a través del servidor local:

```
root@server:~# dig @localhost -x 8.8.8.8

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost -x 8.8.8.8
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 18318
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: eec454206f9c4aa60100000065e237fb703fc1ea55b3b109 (good)
;; QUESTION SECTION:
;8.8.8.8.in-addr.arpa.          IN      PTR

;; ANSWER SECTION:
8.8.8.8.in-addr.arpa.   86400   IN      PTR     dns.google.

;; Query time: 1715 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Fri Mar 01 20:18:03 UTC 2024
;; MSG SIZE  rcvd: 101
```

- **8.8.8.8.in-addr.arpa**: Este es el dominio especial que representa la dirección IP 8.8.8.8 en una consulta inversa de DNS.
- **IN PTR**: El tipo de registro, PTR, indica que se trata de una consulta de puntero DNS inverso.
- **dns.google.**: Este es el nombre de dominio asociado con la dirección IP 8.8.8.8 según la respuesta del servidor DNS.

Como se puede apreciar, el tiempo de respuesta de la “query” es superior desde el servidor local de la máquina V1 hacia el servidor de Google que si empleamos el servidor remoto, ya que su proximidad es más cercana y puede haber latencia de la red.

En cuanto a la Resolución directa con el servidor local:

```
root@server:~# dig @localhost www.google.com.

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost www.google.com.
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 58375
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: dec8458ebc0ed4bb0100000065e233fe2879f3a7c89f7311 (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         295     IN      A       142.250.200.132

;; Query time: 4 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Fri Mar 01 20:01:02 UTC 2024
;; MSG SIZE  rcvd: 87
```

El *dig* desde el servidor local a www.google.com devuelve un registro tipo A, que corresponde a la dirección IP 142.250.200.132 para el sitio web de Google. La consulta se resolvió muy rápidamente en solo 4 milisegundos. El TTL (Time To Live) del registro es de 295 segundos, indicando que se puede almacenar esta respuesta en su propia caché por un tiempo antes de tener que realizar otra consulta para actualizar la información.

En cambio, la Resolución directa con servidor remoto:

```
root@server:~# dig @8.8.8.8 www.google.com

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @8.8.8.8 www.google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 64869
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         235     IN      A       142.250.200.68

;; Query time: 39 msec
;; SERVER: 8.8.8.8#53(8.8.8.8) (UDP)
;; WHEN: Fri Mar 01 20:11:19 UTC 2024
;; MSG SIZE  rcvd: 59
```

Esta consulta es similar a la anterior pero dirigida directamente al servidor DNS de Google (8.8.8.8), en lugar de a través del servidor DNS local (localhost). La respuesta obtenida también es para un registro tipo A de **www.google.com**, sin embargo, la dirección IP devuelta

es diferente (142.250.200.68), lo cual es normal ya que tienen grandes servicios distribuidos con múltiples direcciones IP para tener alta disponibilidad.

El tiempo de respuesta de la consulta es un poco más lento que la consulta local anterior, 39 milisegundos, que es de esperar al hacer la consulta a través de la red a un servidor DNS remoto, en lugar de a uno local.

El TTL para esta entrada es de 235 segundos, que es un poco más corto que el TTL que vimos en la consulta previa. Esto sugiere que la respuesta puede estar menos tiempo en la caché antes de una actualización. La diferencia en el TTL y la dirección IP puede ser una indicación de cómo Google gestiona su infraestructura de DNS, con posibles variaciones geográficas o basadas en el tráfico.

En seguida, usamos la opción +trace (esto significa que dig seguirá el proceso de resolución de la consulta DNS desde la raíz del sistema de nombres de dominio (DNS) hasta el nombre de dominio especificado mostrando cada paso en el proceso.). Se usa el comando dig @localhost www.google.com +trace para resolución directa:

```
com.                    172800  IN      NS      i.gtld-servers.net.
com.                    172800  IN      NS      j.gtld-servers.net.
com.                    172800  IN      NS      k.gtld-servers.net.
com.                    172800  IN      NS      l.gtld-servers.net.
com.                    172800  IN      NS      m.gtld-servers.net.
com.                    86400   IN      DS      19718 13 2 8ACBB0CD28F41250A80A491389424D341522D946B0DA0C0291F2D3D7 71D7
805A
com.                    86400   IN      RRSIG   DS 8 1 86400 20240314170000 20240301160000 30903 . jOsNBJmuR+E2rE2br1iHz
zEXnMULEK3I6Amcpcq/K4wTZ58oNpgodQLI Je5xUqtHysareALkC7g78IVwHM7+DGkJmqaf2bcK1fV8HLcyBFwGnubC AZT+PmwBvUOOPNXM9KChP9kscm4
2k0GMdcLByOSzw5wNNvMUSmPmqT5L klirqH5xO7J3/cXncqWAgjKZRqh3fFxOZ3PR3m/1h6zD3OEJu0IBnxkD kw5DgRDLNkJESuN5Qpq+hiUq16QUzPfRr
km7RsIg1SfsEVb5Nk66YPuT J6OhvAe4hYiMoMMG/vCEi7Ho7ttd7xsIl6LuyjfofpNCGNk3q/HG5p+b 85KaFw==
;; Received 1174 bytes from 199.7.83.42#53(l.root-servers.net) in 31 ms

google.com.             172800  IN      NS      ns2.google.com.
google.com.             172800  IN      NS      ns1.google.com.
google.com.             172800  IN      NS      ns3.google.com.
google.com.             172800  IN      NS      ns4.google.com.
CK0POJMG874LJREF7EFN8430QVIT8BSM.com. 86400 IN NSEC3 1 1 0 - CK0Q2D6NI4I7EQH8NA30NS61O48UL8G5 NS SOA RRSIG DNSKEY NSEC3P
ARAM
CK0POJMG874LJREF7EFN8430QVIT8BSM.com. 86400 IN RRSIG NSEC3 13 2 86400 20240305052632 20240227041632 4534 com. +8QYWgkjye
iGnpSFZNEfI++IUfN0kqQfZDJ3BFQNPREYKEvnGH0QmLzv E8BhzQ3+3UkDuQTkpq7rNoEL1XFZ+A==
S84BKCIBC38P58340AKVNFN5KR9O59QC.com. 86400 IN NSEC3 1 1 0 - S84BR9CIB2A20L3ETR1M2415ENPP99L8 NS DS RRSIG
S84BKCIBC38P58340AKVNFN5KR9O59QC.com. 86400 IN RRSIG NSEC3 13 2 86400 20240306053717 20240228042717 4534 com. FbStRDmTdr
I3HOc9Qi1gFCLUfsUYXQbmefxH7ikJygIUvW9tAb0qusfE pJ1FpBvbQgJ7UgiR2J1qktbQOdHoHQ==
;; Received 648 bytes from 192.52.178.30#53(k.gtld-servers.net) in 43 ms

www.google.com.         300     IN      A       142.250.200.132
;; Received 59 bytes from 216.239.38.10#53(ns4.google.com) in 47 ms
```

- La traza comienza en la zona raíz (**.**), con los servidores **.com** siendo listados (los servidores ***.gtld-servers.net**).
- Después, muestra la delegación del dominio a los servidores de nombres de Google (**ns*.google.com**).
- Posteriormente, se obtienen los registros NS (servidores de nombres) para **google.com**, que indican qué servidores son responsables de la zona de google.com y, por último, al final de la traza, se muestra un ping a www.google.com, que indica que la dirección IP resuelta responde a la conexión.

Finalmente, para la resolución inversa, se usa el comando dig +trace -x 8.8.8.8:

```
; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> +trace -x 8.8.8.8
;; global options: +cmd
.                       5       IN      NS      i.root-servers.net.
.                       5       IN      NS      l.root-servers.net.
.                       5       IN      NS      d.root-servers.net.
.                       5       IN      NS      a.root-servers.net.
.                       5       IN      NS      g.root-servers.net.
.                       5       IN      NS      e.root-servers.net.
.                       5       IN      NS      h.root-servers.net.
.                       5       IN      NS      c.root-servers.net.
.                       5       IN      NS      j.root-servers.net.
.                       5       IN      NS      k.root-servers.net.
.                       5       IN      NS      b.root-servers.net.
.                       5       IN      NS      m.root-servers.net.
.                       5       IN      NS      f.root-servers.net.
;; Received 519 bytes from 127.0.0.53#53(127.0.0.53) in 2323 ms

in-addr.arpa.           172800  IN      NS      a.in-addr-servers.arpa.
in-addr.arpa.           172800  IN      NS      b.in-addr-servers.arpa.
in-addr.arpa.           172800  IN      NS      c.in-addr-servers.arpa.
in-addr.arpa.           172800  IN      NS      d.in-addr-servers.arpa.
in-addr.arpa.           172800  IN      NS      e.in-addr-servers.arpa.
in-addr.arpa.           172800  IN      NS      f.in-addr-servers.arpa.
in-addr.arpa.           86400   IN      DS      47054 8 2 5CAFCCEC201D1933B4C9F6A9C8F51E51F3B39979058AC21B8DF1B1F2 81CBC
6F2
in-addr.arpa.           86400   IN      DS      54956 8 2 E0E2BF5CFBD66572CA05EC18267D91509BA6A9405AF05C3FD4141DFA 45200
C08
in-addr.arpa.           86400   IN      DS      53696 8 2 13E5501C56B20394DA921B51412D48B7089C5EB6957A7C58553C4D4D 424F0
4DF
8.8.8.in-addr.arpa.     86400   IN      NS      ns1.google.com.
8.8.8.in-addr.arpa.     86400   IN      NS      ns4.google.com.
8.8.8.in-addr.arpa.     86400   IN      NS      ns2.google.com.
8.8.8.in-addr.arpa.     86400   IN      NS      ns3.google.com.
8.8.8.in-addr.arpa.     10800   IN      NSEC    80.8.8.in-addr.arpa. NS RRSIG NSEC
8.8.8.in-addr.arpa.     10800   IN      RRSIG   NSEC 8 5 10800 20240316052827 20240302042827 27033 8.in-addr.arpa. UNXHO
KEhTvMx1PjqTmP8K1wBOhJPrryU011jiYk4wjRbizGMm5plQUSW F2CMIawnsgQK4P1hv0UjuzCwsEGevPakOHGCL6CZSOI3tJ+8SvvHs30T KDWHpjq2JWb
8FTELMAzJtYKqxubWYcLcATZeIEuLW/ejr58+H9BMCO3l /Vg=
;; Received 346 bytes from 192.82.134.30#53(y.arin.net) in 95 ms
```

El comando +trace está mostrando cada servidor DNS consultado en el camino para resolver la dirección IP 8.8.8.8 a un nombre. Esencialmente, está siguiendo la cadena de delegación desde los servidores raíz hasta los servidores autoritativos que pueden responder por la dirección IP en cuestión.

2. Instalar el servidor de DNS Bind 9

Para instalar el servidor de DNS Bind 9 en Ubuntu, se deben seguir los pasos mencionados previamente en el apartado 1, es decir, usar el comando apt install bind9 , tras realizar un update y upgrade sobre los paquetes anticuados.

```
root@server:~# apt install bind9
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
bind9 ya está en su versión más reciente (1:9.18.18-0ubuntu0.22.04.2).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 15 no actualizados.
root@server:~#
```

3. Configuración del servidor Bind. Estudiar el archivo named.conf y el contenido del directorio de configuración de Bind. Arrancar el servicio named. El servidor deberá escuchar en el puerto 53/udp de todas las

direcciones IP del equipo, y atender peticiones de DNS desde cualquier dirección origen. Trabajará de modo recursivo. A continuación, realizar distintas configuraciones sobre el servidor Bind y verificar su correcto funcionamiento:

Lo primero es arrancar el servicio named:

```
root@server:~# systemctl start named
root@server:~#
```

Posteriormente, se debe acceder al fichero ubicado en /etc/bind/named.conf.options donde se debe introducir lo siguiente:

```
        directory "/var/cache/bind";

        // If there is a firewall between you and nameservers you want
        // to talk to, you may need to fix the firewall to allow multiple
        // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

        // If your ISP provided one or more IP addresses for stable
        // nameservers, you probably want to use them as forwarders.
        // Uncomment the following block, and insert the addresses replacing
        // the all-0's placeholder.

        // forwarders {
        //      0.0.0.0;
        // };

        //========================================================================
        // If BIND logs error messages about the root key being expired,
        // you will need to update your keys.  See https://www.isc.org/bind-keys
        //========================================================================
        dnssec-validation auto;

        listen-on-v6 { any; };
        listen-on port 53 { any; };
        recursion yes;
};
```

La sentencia listen-on port 53 { any; }; permite que el servidor escuche en el puerto 53/udp de todas las direcciones IP del equipo. La línea recursion yes; es para atender peticiones de DNS desde cualquier dirección origen.

Es muy importante revisar siempre la sintaxis de los ficheros modificados antes de continuar, ya que, si no podría dar problema:

```
root@server:~# named-checkconf /etc/bind/named.conf.options
root@server:~#
```

Después de revisar que todo está correcto, sin errores, hay que reiniciar el servicio named para que se apliquen los cambios:

```
root@server:~# systemctl restart named
root@server:~#
```

Para apreciar el estado del servidor se puede introducir:

- sudo systemctl status named || sudo rndc status

A) Comprobar que el servidor mantiene una caché de últimos accesos realizados (para eliminar el contenido de la caché se puede utilizar el comando rndc flush).

sudo rndc dumpdb –cache se utiliza para generar un volcado de la caché de un servidor DNS BIND en un formato legible por humanos. Este comando generará un archivo de texto en /var/cache/bind/named_dump.db que contiene la caché completa del servidor DNS.

Para ver la caché de nuestro servidor, hay que hacerle consultas a este, sino estaría vacío:

```
root@server:~# dig @localhost www.google.com.

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost www.google.com.
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 531
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 1e827d60f630970d0100000065e2eb7789ee538d98ea6fa9 (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         300     IN      A       142.250.185.4

;; Query time: 647 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Sat Mar 02 09:03:51 UTC 2024
;; MSG SIZE  rcvd: 87
```

Se puede ver que el tiempo de query es de 647 msec. Si se hace la consulta otra vez, esta tardará menos:

```
root@server:~# dig @localhost www.google.com.

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost www.google.com.
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 29084
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: e0cc34cf7325550d0100000065e2ec2857e07e46b4820f7d (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         123     IN      A       142.250.185.4

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Sat Mar 02 09:06:48 UTC 2024
;; MSG SIZE  rcvd: 87
```

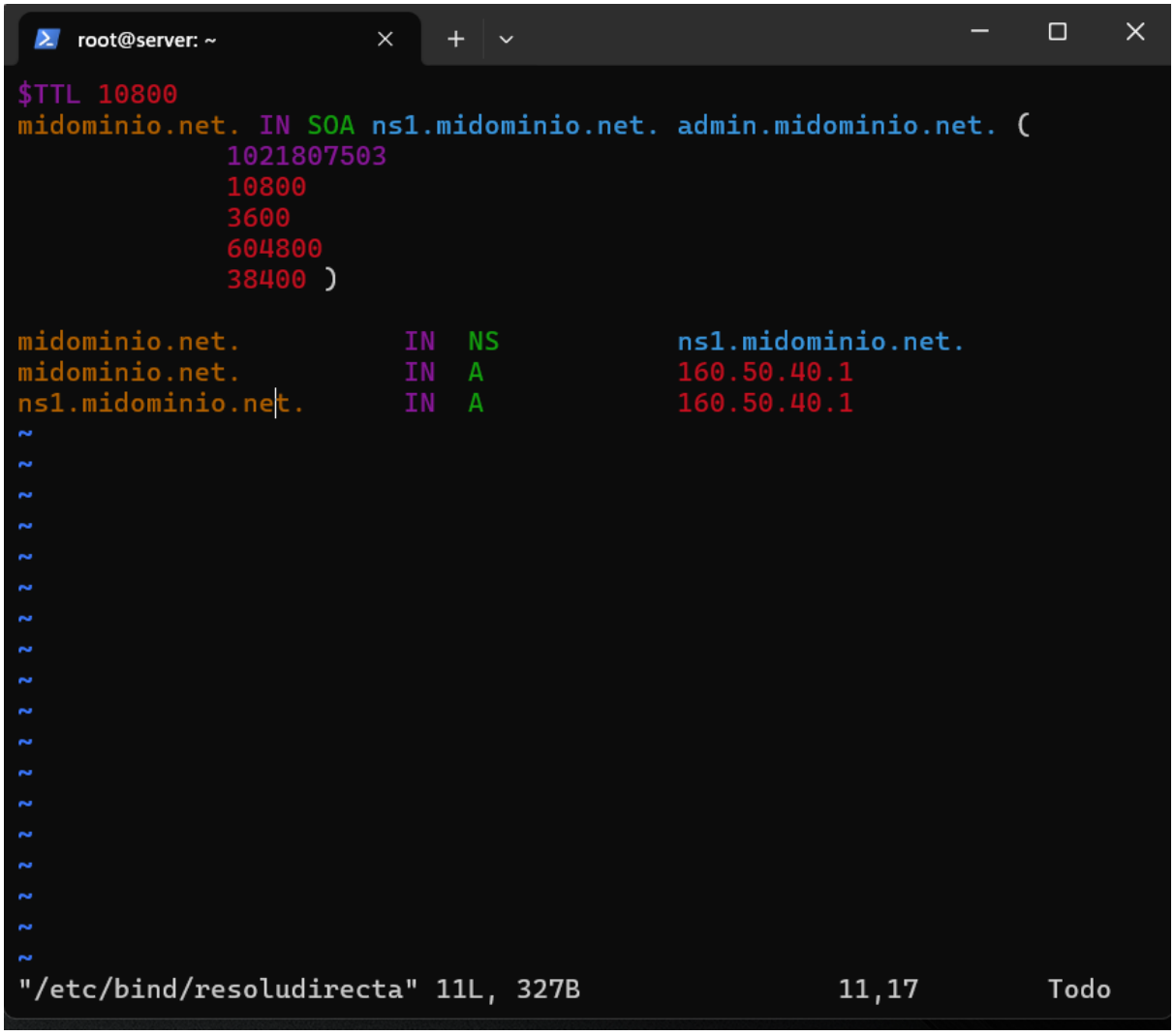
Si se borra la caché con rndc flush podremos observar que la consulta dig @localhost www.google.com. Tardará más (tiempo de query). Esto se debe a que cuando se realiza una consulta de DNS por primera vez después de borrar la caché, el servidor DNS necesita resolver la consulta buscando la información en servidores de nombres remotos y, posiblemente, construyendo su propia caché en el proceso. Este proceso puede llevar más tiempo.

```
root@server:~# dig @localhost www.google.com.

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost www.google.com.
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 46104
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: e6cdde392342ee060100000065e2ec593d6c4de4a135d35b (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         300     IN      A       142.250.185.4

;; Query time: 1308 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Sat Mar 02 09:07:37 UTC 2024
;; MSG SIZE  rcvd: 87
```

Por otro lado, si se realiza una consulta después de que la respuesta ya esté en la caché, el servidor DNS puede devolver la respuesta mucho más rápido, ya que se puede recuperar directamente de la caché local sin necesidad de realizar consultas adicionales a servidores remotos.

b. Creación de zonas primarias. Supondremos un dominio ficticio midominio.net, que administra direcciones en el rango 160.50.40.0/24. Configurar dos zonas primarias para administrar las resoluciones directa e inversa.

Para este apartado, es necesario crear dos ficheros dentro de los directorios /etc/bind/ uno de ellos se llama resoludirecta y el otro resoluinversa. En resoludirecta, se capta lo siguiente:



- **ns1.midominio.net.** es el servidor de nombres principal para la zona.
- **admin.midominio.net.** es la dirección de contacto para el administrador de la zona.
- **1021808093** es el número de serie de la zona, que debe incrementarse con cada actualización para que los cambios se propaguen a los servidores esclavos.

- **10800** es el tiempo en segundos que un servidor secundario debe esperar para comprobar si hay cambios en la zona (refresco).
- **3600** es el tiempo en segundos que un servidor secundario debe esperar antes de volver a intentar una solicitud fallida de actualización de la zona.
- **604800** es el tiempo en segundos que un servidor secundario esperará antes de considerar los datos de la zona como no válidos si el servidor maestro no está disponible.
- **38400** es el tiempo en segundos que se debe cachear el registro SOA (tiempo de vida negativo).

El fichero de configuración resoluinversa incluye lo siguiente:

```
$TTL 10800
40.50.160.in-addr.arpa.    IN      SOA     ns1.midominio.net.  admin.midominio.net. (
                             1021808093
                             10800
                             3600
                             604800
                             38400 )
40.50.160.in-addr.arpa.    IN      NS      ns1.midominio.net.
1.40.50.160.in-addr.arpa.  IN      PTR     ns1.midominio.net.
```

1. **Registro NS**: Indica que **ns1.midominio.net.** es el servidor de nombres autorizado para esta zona de dirección IP inversa.
2. **Registro PTR**: Establece un mapeo inverso de la dirección IP **1.40.50.160** al nombre de dominio **ns1.midominio.net.** donde se pregunta por el nombre asociado a una dirección IP específica.

En el fichero /etc/bind/named.conf.local se meterá la configuración de las zonas del DNS primario, indicando que el tipo de servidor es primario y de qué ubicación tomará la información para ambas resoluciones:

```
zone "midominio.net" {
        type master;
        file "/etc/bind/resoludirecta";
};

zone "40.50.160.in-addr.arpa" {
        type master;
        file "/etc/bind/resoluinversa";
};
```

Conviene, comprobar la sintaxis de todos los ficheros que han sido modificados. En el caso del fichero named.conf.local es con el comando named-checkconf

/etc/bind/named.conf.local y en el resto de los ficheros es como figura en la próxima imagen:

```
root@server:~# named-checkzone midominio.net /etc/bind/resoludirecta
zone midominio.net/IN: loaded serial 1021807503
OK
root@server:~# named-checkzone 40.50.160.in-addr.arpa /etc/bind/resoluinversa
zone 40.50.160.in-addr.arpa/IN: loaded serial 1021808093
OK
root@server:~#
```

Se reinicia el servicio named con: systemctl restart named
Para comprobar que todo ha sido exitoso, se realizan consultas:

```
root@server:~# dig @localhost midominio.net

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost midominio.net
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 54372
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 37d2c83b32575a300100000065e32af620d136a0112b9bf5 (good)
;; QUESTION SECTION:
;midominio.net.                 IN      A

;; ANSWER SECTION:
midominio.net.          300     IN      A       104.21.86.60
midominio.net.          300     IN      A       172.67.215.166

;; Query time: 276 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Sat Mar 02 13:34:46 UTC 2024
;; MSG SIZE  rcvd: 102
```

```
root@server:~# dig @localhost -x 160.50.40.1

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost -x 160.50.40.1
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 53344
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 5e1abb444adb2e840100000065e32f0e9560cef355ffd888 (good)
;; QUESTION SECTION:
;1.40.50.160.in-addr.arpa.      IN      PTR

;; ANSWER SECTION:
1.40.50.160.in-addr.arpa. 10800 IN      PTR     ns1.midominio.net.

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Sat Mar 02 13:52:14 UTC 2024
;; MSG SIZE  rcvd: 112
```

c. Mantenimiento de zonas primarias. Añadir los siguientes registros a las zonas creadas: A (zipi, 160.50.40.1), A (zape, .2), A (mortadelo, .4), A (correo, .50), A (dns1, nuestra propia dirección IP), A (www, a las direcciones 160.50.40.200 y 160.50.40.201), NS (dns1), CNAME (aplicaciones ->mortadelo), MX (correo, prioridad20). Todos los registros de tipo A deben tener asociado su correspondiente registro PTR. Verificar su funcionamiento con el cliente de DNS (dig o nslookup)

Las incorporaciones al fichero de la zona de resolución directa lucen tal que así:

```
$TTL 10800
midominio.net.          IN  SOA  dns1.midominio.net.  admin.midominio.net. (
                        1021807503
                        10800
                        3600
                        604800
                        38400 )
midominio.net.              IN  NS    dns1.midominio.net.
dns1.midominio.net.         IN  A    192.168.160.135
zipi.midominio.net.         IN  A     160.50.40.1
zape.midominio.net.         IN  A     160.50.40.2
mortadelo.midominio.net.    IN  A     160.50.40.4
correo.midominio.net.       IN  A     160.50.40.50
www.midominio.net.          IN  A     160.50.40.200
www.midominio.net.          IN  A     160.50.40.201
aplicaciones.midominio.net. IN  CNAME mortadelo.midominio.net.
midominio.net.              IN  MX    20 correo.midominio.net.
midominio.net.              IN  NS    dns2.midominio.net.
dns2.midominio.net.         IN  A     192.168.160.136
```

Agregaciones en el fichero de la zona de resolución inversa:

```
$TTL 10800
40.50.160.in-addr.arpa.    IN     SOA     ns1.midominio.net.  admin.midominio.net. (
                          1021808093
                          10800
                          3600
                          604800
                          38400 )
40.50.160.in-addr.arpa.         IN      NS      ns1.midominio.net.
1.40.50.160.in-addr.arpa.       IN      PTR     zipi.midominio.net.
2.40.50.160.in-addr.arpa.       IN      PTR     zape.midominio.net.
4.40.50.160.in-addr.arpa.       IN      PTR     mortadelo.midominio.net.
50.40.50.160.in-addr.arpa.      IN      PTR     correo.midominio.net.
200.40.50.160.in-addr.arpa.     IN      PTR     www.midominio.net.
201.40.50.160.in-addr.arpa.     IN      PTR     www.midominio.net.
```

Comprobamos la sintaxis de ambos ficheros:

```
root@server:~# named-checkzone midominio.net /etc/bind/resoludirecta
zone midominio.net/IN: loaded serial 1021807503
OK
```

```
root@server:~# named-checkzone 40.50.160.in-addr.arpa /etc/bind/resoluinversa
zone 40.50.160.in-addr.arpa/IN: loaded serial 1021808093
OK
```

Reiniciamos con: systemctl restart named

Ahora realizamos consultas:

```
root@server:~# dig @localhost zipi.midominio.net.

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost zipi.midominio.net.
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 53529
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: e5c24b446fbb335d0100000065e47f7dc9d48148c2873e49 (good)
;; QUESTION SECTION:
;zipi.midominio.net.            IN      A

;; ANSWER SECTION:
zipi.midominio.net.     10800   IN      A       160.50.40.1

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Sun Mar 03 13:47:41 UTC 2024
;; MSG SIZE  rcvd: 91
```

```
root@server:~# dig @localhost correo.midominio.net.

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost correo.midominio.net.
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 64941
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 1864f3ddb9b7a8db0100000065e47fc9913881d9b04b1c31 (good)
;; QUESTION SECTION:
;correo.midominio.net.          IN      A

;; ANSWER SECTION:
correo.midominio.net.   10800   IN      A       160.50.40.50

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Sun Mar 03 13:48:57 UTC 2024
;; MSG SIZE  rcvd: 93
```

```
root@server:~# dig @localhost -x 160.50.40.200

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost -x 160.50.40.200
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15249
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: e5ce6c2e029aaa670100000065e48051a9f75490329f9b5f (good)
;; QUESTION SECTION:
;200.40.50.160.in-addr.arpa.    IN      PTR

;; ANSWER SECTION:
200.40.50.160.in-addr.arpa. 10800 IN    PTR     www.midominio.net.

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Sun Mar 03 13:51:13 UTC 2024
;; MSG SIZE  rcvd: 114
```

```
root@server:~# dig @localhost -x 160.50.40

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost -x 160.50.40
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 18957
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: b4fd79f1f1b858330100000065e480928daa7f11fbf1fab9 (good)
;; QUESTION SECTION:
;40.50.160.in-addr.arpa.                IN      PTR

;; AUTHORITY SECTION:
40.50.160.in-addr.arpa. 10800   IN      SOA     ns1.midominio.net. admin.midominio.net. 1021808093 10800 3600 604800 384
00

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Sun Mar 03 13:52:18 UTC 2024
;; MSG SIZE  rcvd: 138
```

d. Sincronización primaria-secundarias. Vamos a replicar la zona midominio.net en otro servidor de DNS secundario. Arrancar la segunda máquina virtual (VM2), Rocky, e instalar nuevamente el servidor Bind 9. A continuación establecer una zona secundaria del dominio midominio.net. En el servidor maestro añadir otra entrada NS (dns2) apuntando a la dirección IP del nuevo servidor. Configurar en ambos servidores los parámetros que permitirán la transferencia de la zona. Reiniciar ambos servidores y comprobar que el secundario mantiene una copia de la zona. Llevar a cabo algún cambio en la zona maestra incrementando su número de serie. Reiniciar el servidor y verificar que se actualiza la zona secundaria.

Nota: se recomienda configurar la notificación explícita a las secundarias (notify explicit) y la directiva also-notify con la dirección IP del servidor secundario.

La VM2 escogida es la rocky linux (192.168.160.136). En base a ello, lo primero es actualizar los paquetes con dnf update, a continuación, se instala bind poniendo dnf install bind, una vez hecho esto, se usa el comando systemctl enable named para que BIND9 se inicie automáticamente en el arranque del sistema.

En la VM2 en el fichero ubicado en /etc/named.conf añadimos la configuración de zona:

```
zone "midominio.net" {
        type slave;
        masters { 192.168.160.135; };
        file "resoludirecta";
        allow-transfer { none; };
        allow-notify { 192.168.160.135; };
};
```

En la VM1, desde la ruta /etc/bind/named.conf.local se añade:

```
zone "midominio.net" {
        type master;
        file "resoludirecta";
        notify explicit;
        also-notify { 192.168.160.136; };
        allow-transfer { 192.168.160.136; };
};
```

Se han añadido 3 sentencias adicionales al fichero de configuración local. Donde notify explicit se refiere a que notifique explícitamente a la máquina secundaria, also-notify lleva la dirección IP de la VM2 y es para notificar a esta y, por último, allow-transfer especifica qué servidores están autorizados para realizar transferencias de zona, es más define qué servidores pueden obtener una copia completa de la zona desde mi servidor.

Nuevamente en la misma máquina virtual, en el fichero ubicado en: /etc/bind/resoludirecta se añade lo siguiente:

```
$TTL 10800
midominio.net.          IN  SOA  dns1.midominio.net.  admin.midominio.net. (
                        1021807503
                        10800
                        3600
                        604800
                        38400 )
midominio.net.               IN  NS    dns1.midomino.net.
dns1.midominio.net.          IN  A    192.168.160.135
zipi.midominio.net.          IN  A     160.50.40.1
zape.midominio.net.          IN  A     160.50.40.2
mortadelo.midominio.net.     IN  A     160.50.40.4
correo.midominio.net.        IN  A     160.50.40.50
www.midominio.net.           IN  A     160.50.40.200
www.midominio.net.           IN  A     160.50.40.201
aplicaciones.midominio.net.  IN  CNAME mortadelo.midominio.net.
midominio.net.               IN  MX    20 correo.midominio.net.
midominio.net.               IN  NS    dns2.midominio.net.
dns2.midominio.net.          IN  A     192.168.160.136
```

Se han añadido las dos últimas líneas, una entrada NS apuntando a la dirección del nuevo servidor (IP de la secundaria).

Ahora se comprueba la sintaxis de todos los ficheros que se han modificado, tanto en la VM1 como la VM2:

VM1:

```
root@server:~# named-checkconf /etc/bind/named.conf.local
root@server:~# named-checkzone midominio.net /etc/bind/resoludirecta
zone midominio.net/IN: loaded serial 1021807503
OK
root@server:~#
```

VM2:

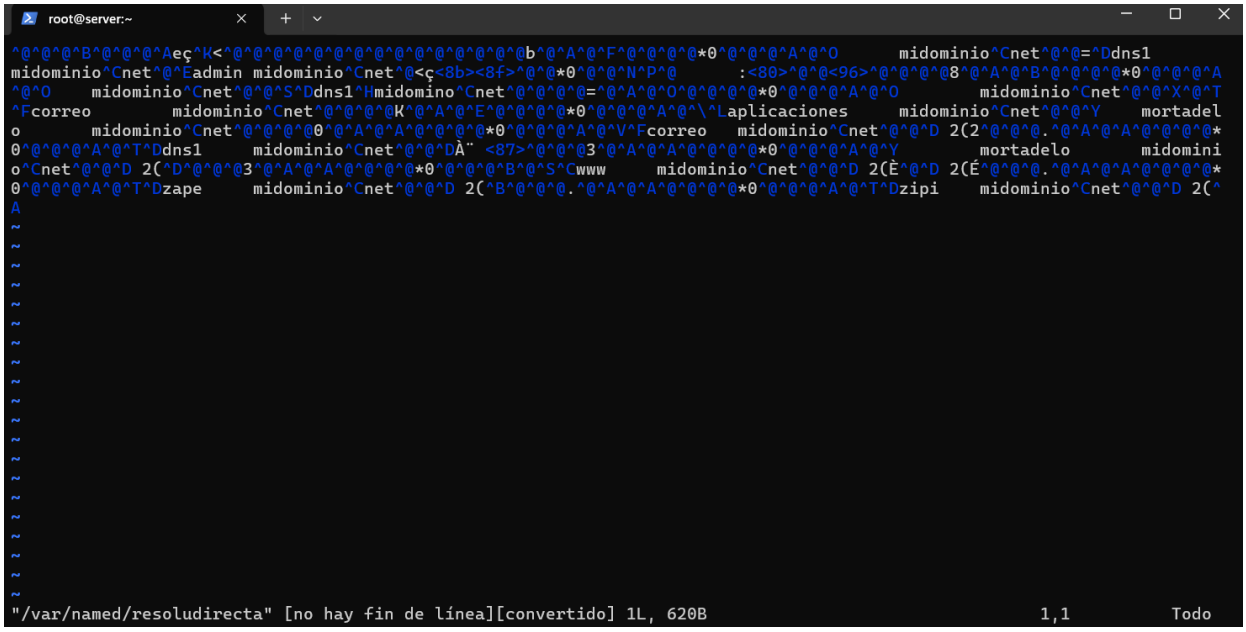
```
[root@server ~]# named-checkconf /etc/named.conf
[root@server ~]#
```

Cabe destacar que hay que reiniciar el servicio named en ambas máquinas con: **systemctl restart named.**

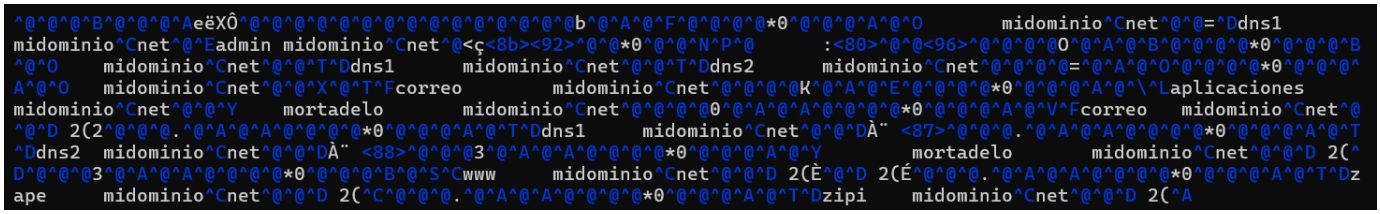
A continuación, se realizan las consultas correspondientes:

```
root@server:~# dig @localhost dns2.midominio.net

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> @localhost dns2.midominio.net
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 22365
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 349e48f9b0124eea0100000065e758de99a37a2041289b62 (good)
;; QUESTION SECTION:
;dns2.midominio.net.            IN      A

;; ANSWER SECTION:
dns2.midominio.net.     10800   IN      A       192.168.160.136

;; Query time: 3 msec
;; SERVER: 127.0.0.1#53(localhost) (UDP)
;; WHEN: Tue Mar 05 17:39:42 UTC 2024
;; MSG SIZE  rcvd: 91
```

En la VM2, se ha tenido que crear un fichero llamado resoludirecta en la ubicación /var/named/ donde se ha guardado la transferencia de zona de la VM1:



En la imagen que acabamos de observar se aprecia el archivo de zona transferido automáticamente con la información encriptada.



Tras hacer una modificación en el fichero resoludirecta de la VM1, y al incrementar el número de serie se actualiza inmediatamente la máquina esclava reconociendo el dominio dns2 en la VM2:

Nuevamente hay que recargar la configuración o reiniciar el servicio BIND para que los cambios surtan efecto. Como ya se ha mencionado en ocasiones anteriores, esto se puede llevar a cabo, tal que así:

Desde el servidor maestro:

- sudo rndc reload

Mientras que en el servidor esclavo:

Se debe utilizar el siguiente comando para hacer inmediata la transferencia y no exista un retardo: **rndc retransfer midominio.net**

Para su comprobación, se puede visualizar el fichero que se creó en la VM2 para confirmar si tiene las modificaciones. O bien, consultar con un **rndc statuszone midominio.net** la zona mirando si el número de serie se ha actualizado a 1021807505, tal y como figura en la siguiente imagen.

```
[root@server ~]# rndc zonestatus midominio.net
name: midominio.net
type: secondary
files: resoludirecta
serial: 1021807505
nodes: 9
last loaded: Tue, 05 Mar 2024 18:29:06 GMT
next refresh: Tue, 05 Mar 2024 21:55:32 GMT
expires: Tue, 12 Mar 2024 19:14:19 GMT
secure: no
dynamic: no
reconfigurable via modzone: no
```

e. Delegación de un subdominio. Crear una nueva zona maestra pruebas.com en el servidor VM2. Vamos a configurar la delegación de un subdominio (por ejemplo, ceu.pruebas.com), cuya zona maestra se ubicará en el otro servidor de DNS (VM1). Para hacer las pruebas añadir algún registro A en el nuevo subdominio y verificar que una petición de resolución dirigida al servidor VM2 y relativa al subdominio, es reenviada y resuelta en el servidor VM1. Comprobar igualmente el efecto de la directiva recursión yes|no en el comportamiento del servidor que aloja el dominio principal (VM2).

A la hora de configurar la delegación de subdominios es conveniente trabajar del subdominio más específico (ceu.pruebas.com), al dominio que engloba al resto de subdominios (pruebas.com).

En cuanto a la máquina virtual primaria:

Si se accede al archivo de configuración local (/etc/bind/named.conf.local)

```
zone "ceu.pruebas.com"{
        type master;
        file "ceupruebas.com";
        allow-query { localhost;192.168.160.136; };
};
```

Salta a la vista que, si se realiza una petición de DNS a un recurso del dominio, el estado de la consulta aparece como denegado, incluso si se incluye la dirección IP concreta de la máquina para que pueda acceder a dicho servidor.

Así pues, si desde la VM2 se realiza una consulta a la VM1 (se usa la IP de la VM1 como servidor):

```
[root@server ~]# dig @192.168.160.135 inventada.ceu.pruebas.com

; <<>> DiG 9.16.23-RH <<>> @192.168.160.135 inventada.ceu.pruebas.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 13552
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: dcd3eb1d978523df0100000065ec514d143b4afa3590c39e (good)
;; QUESTION SECTION:
;inventada.ceu.pruebas.com.     IN      A

;; Query time: 1 msec
;; SERVER: 192.168.160.135#53(192.168.160.135)
;; WHEN: Sat Mar 09 13:08:45 CET 2024
;; MSG SIZE  rcvd: 82
```

El resultado de esta consulta fue un error SERVFAIL, lo que significa que el servidor no pudo procesar la solicitud correctamente. Este error se debe a que el servidor no tiene autoridad para el dominio que se consultó.

A continuación, desde /etc/bind/ceupruebas.com:

```
$TTL 10800
ceu.pruebas.com.    IN  SOA     ns1.ceu.pruebas.com. admin.ceu.pruebas.com. (
                        2022030101
                        3600
                        1800
                        604800
                        38400  )
ceu.pruebas.com.                IN  NS      ns1.ceu.pruebas.com.
ns1.ceu.pruebas.com.            IN  A       192.168.160.135
inventada.ceu.pruebas.com.      IN  A       200.1.1.1
```

Se ha puesto una dirección al azar (inventada.ceu.pruebas.com.) con una dirección IP también aleatoria (200.1.1.1).

Respecto a la máquina VM2:

Considerando el fichero /etc/named.conf de tal forma:

```
zone "pruebas.com" {
        type master;
        file "pruebas.com";
};
```

Se modifica el atributo recursión de yes a no:

```
//
// Provided by Red Hat bind package to configure the ISC BIND named(8) DNS
// server as a caching only nameserver (as a localhost DNS resolver only).
//
// See /usr/share/doc/bind*/sample/ for example named configuration files.
//

options {
        listen-on port 53 { 127.0.0.1; };
        listen-on-v6 port 53 { ::1; };
        directory       "/var/named";
        dump-file       "/var/named/data/cache_dump.db";
        statistics-file "/var/named/data/named_stats.txt";
        memstatistics-file "/var/named/data/named_mem_stats.txt";
        secroots-file   "/var/named/data/named.secroots";
        recursing-file  "/var/named/data/named.recursing";
        allow-query     { localhost; };

        /*
         - If you are building an AUTHORITATIVE DNS server, do NOT enable recursion.
         - If you are building a RECURSIVE (caching) DNS server, you need to enable
           recursion.
         - If your recursive DNS server has a public IP address, you MUST enable access
           control to limit queries to your legitimate users. Failing to do so will
           cause your server to become part of large scale DNS amplification
           attacks. Implementing BCP38 within your network would greatly
           reduce such attack surface
        */
        recursion yes;
```

Se crea un fichero llamado pruebas.com en /var/named/ que incluye la siguiente configuración:

```
$TTL 10800
pruebas.com.    IN  SOA     ns1.pruebas.com. admin.pruebas.com. (
                        2022030101
                        3600
                        1800
                        604800
                        36400
                            )
pruebas.com.            IN  NS      ns1.pruebas.com.
ns1.pruebas.com.        IN  A       192.168.160.136
ceu.pruebas.com.        IN  NS      ns1.ceu.pruebas.com.
ns1.ceu.pruebas.com.    IN  A       192.168.160.135
```

Es importante añadir a la zona un nuevo name server para la dirección ceu.pruebas.com que apunte al servidor donde se aloja el subdominio (especificar la dirección de la VM1).