Configuración de servicio DNS en Linux

BIND9

Franz Josué Ramírez Villca
2DAW-A

Indice

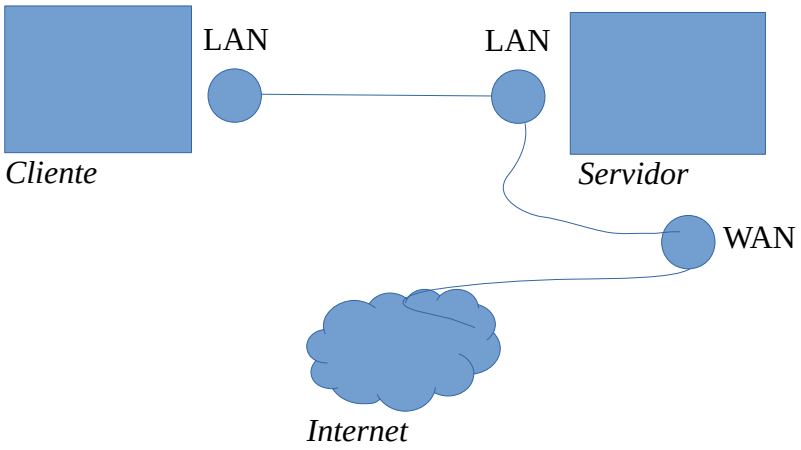
Introducción a la configuración de un servidor DNS.

Configuraremos un servidor DNS en Ubuntu Server usando BIND9.

El servidor de DNS tendrá que poder resolver los nombres tanto por la IP o el nombre del dominio.

Configuración de adaptadores de red VirtualBox.

Tendremos que configurar los adaptadores de nuestras maquinas virtuales tal y como lo vimos en clase, el cliente y el servidor tendrán un adaptador de red en modo “Red Interna” y el servidor también deberá tener un adaptador de red en modo “NAT”.



Configuración adaptadores de red.

Realizaremos la configuración de la conexión de la maquina cliente con la maquina servidor, las dos estarán conectadas con una Red interna llamada “lanjosue”, ademas que el servidor tendrá una segunda tarjeta de red en NAT.

La finalidad de esta configuración es tener una red cerrada, sin salida a internet, simulando una red local que se conectaría a un “router” que en este caso seria la Red NAT.

El nombre de la red interna tiene bastante importancia ya que con esta es como se identifican las redes internas, las que tengan este mismo nombre formaran una única red.

Espacio de direccionamiento IP.

Aquí veremos que direcciones Ip usaremos para configuraras en nuestras maquinas como estáticas.

Red Interna

Dirección Ip de tipo C (192.168.0.0).

Mas que suficiente para la actividad que estamos realizando.

Red NAT

Usaremos la red que nos da por defecto el DCHP del adaptador de red NAT, que es la 10.0.2.15.

Usaremos esta ya que necesitamos salida a internet.

Cliente.

Red Interna

La configuración seria la siguiente:

IP: 192.168.0.160

Mascara de Subred: 255.255.255.0

Puerta de Enlace: 192.168.0.150

Nota: Es importante que la puerta de enlace apunte a la dirección ip del servidor (Ip de la Red Interna), para tener salida a internet.

DNS: 192.168.0.150, 192,168,0,151

Nota: Es importante que de servidor DNS pongamos el nuestro para así poder resolver los nombres de dominio.

Servidor Primario.

Red NAT

La configuración seria la siguiente:

IP: 10.0.2.15

Mascara de Subred: 255.255.255.0

Puerta de Enlace: 10.0.2.2

Red Interna

La configuración seria la siguiente:

IP: 192.168.0.150

Mascara de Subred: 255.255.255.0

Servidor Secundario.

Red Interna

La configuración seria la siguiente:

IP: 192.168.0.151

Mascara de Subred: 255.255.255.0

Enrutamiento en el Servidor DNS.

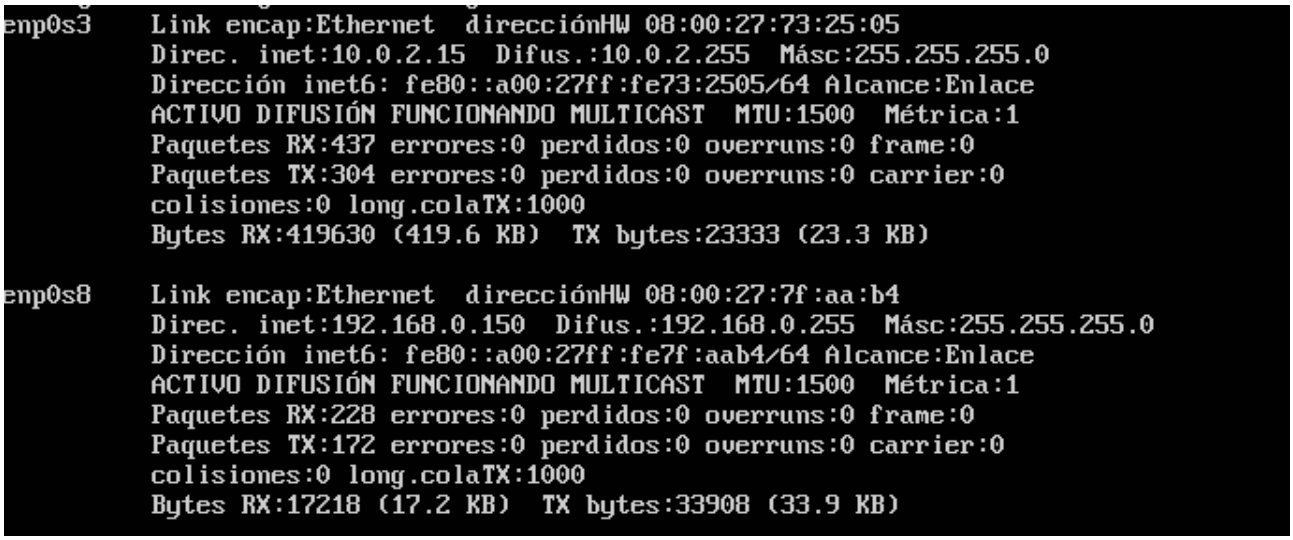
Para poder tener salida a internet desde la red interna, necesitamos poder usar nuestro adaptador de red NAT como un “router”.

Necesitamos configurar nuestra red interna para que se conecte a internet a través de la red NAT cuando lo necesite.

Creación de un script para activar el enrutamiento.

Para facilitarnos el trabajo a la hora de estar usando el enrutamiento, crearemos un script con todas las instrucciones necesarias para su correcto funcionamiento, al cual llamaremos activar-enrutamiento.sh y contendra las siguientes instrucciones.

Esta captura es para ver las interfaces y su direccionamiento.



Script para activar enrutamiento.

Con esto activaremos el enrutamiento en Linux, aceptaremos el reenvio de paquetes de dentro hacia fuera de nuestra red mediante NAT y le indicaríamos que rango de direccionamiento queremos configurar y por que interfaz saldrá.

Instalación y configuración de BIND9 en nuestro servidor.

Instalación de bind9.

Para instalarlo simplemente ejecutaremos el comando “sudo apt-get install bind9”.

Configurar servidor DNS.

Una vez tengamos instalado bind nos dirigiremos a nuestro primer fichero de configuración que tendremos que modificar **/etc/bind/named.conf.options** aquí solo tendremos que quitar el comentario y añadir un servidor DNS que vayamos a usar en caso de que el nuestro no encuentre respuesta a alguna petición. Nosotros usaremos el servidor 8.8.8.8.

Configuración de zonas de búsqueda directa e inversa.

Editaremos el fichero **/etc/bind/named.conf.local** en este archivo especificaremos las zonas de búsqueda directa e inversa del servicio DNS. El dominio de nuestra zona directa y la subred del la zona inversa. También tendremos que incluir qué tipo de servicio es (maestro o esclavo) y en que archivos hará la búsqueda de nombres.

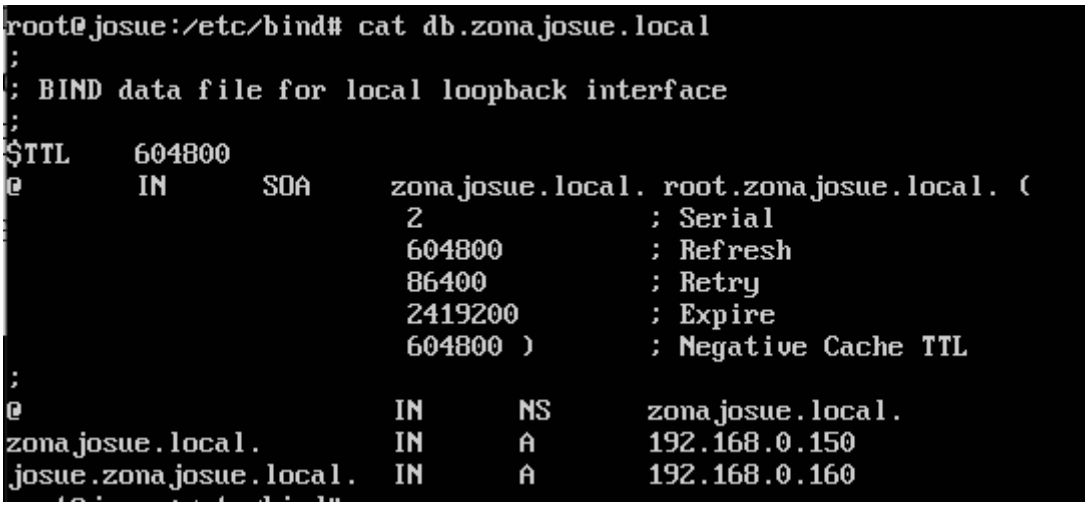
```
root@josue:/home/josue# cat /etc/bind/named.conf.local
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

//zona directa
zone "zonajosue.local" {
        type master;
        file "/etc/bind/db.zonajosue.local";
        notify yes;
};

//zona inversa
zone "0.168.192.in-addr.arpa"{
        type master;
        file "/etc/bind/db.192.168.0.rev";
        notify yes;
};
```

Podemos comprobar que no tenemos errores de sintaxis en al fichero, gracias al comando **named-checkconf /etc/bind/named.conf.local**. Si no nos devuelve nada sabremos que todo esta bien.

Tabla de búsqueda directa DNS.

Dentro del directorio **/etc/bind/** se encuentra un fichero llamado **db.local** copiaremos este fichero dentro de la misma ruta con el nombre que especificamos en el fichero de zonas (**db.zonajosue.local**).

Este fichero sera nuestra tabla de busqueda directa. Las primeras líneas son unos parámetros relacionados con la actualización del DNS (número de serie y periodos de actuación). La siguiente línea indica quién es el servidor primario (NS = Name Server). Las siguientes líneas especifican las @IP's de los diferentes PC's componentes del dominio (A = Address).

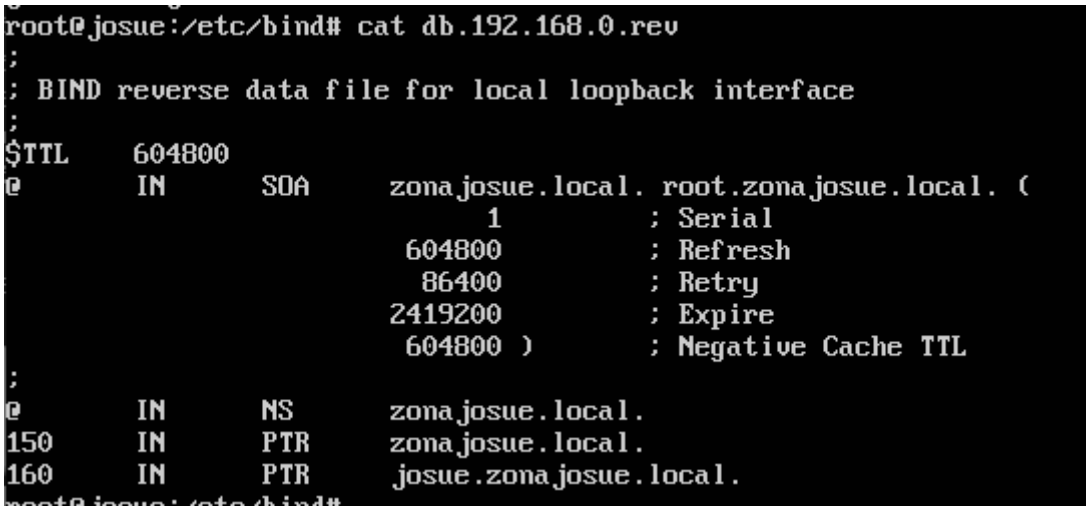


En la ultima linea tenemos añadido a nuestro cliente.

Tabla de búsqueda inversa DNS.

Dentro del directorio **/etc/bind/** se encuentra un fichero llamado **db.127 o db.0** copiaremos este fichero dentro de la misma ruta con el nombre que especificamos en el fichero de zonas (**db.192.168.0.rev**).

Este archivo contiene las tablas de búsqueda inversa.

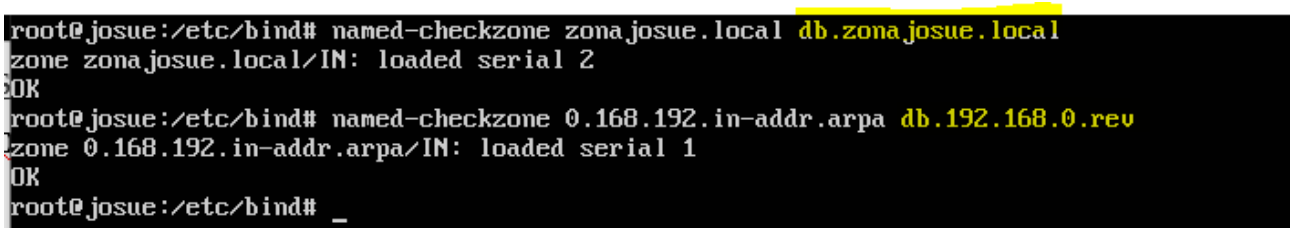


Aquí podemos ver la dirección IP apunta a un dominio, como ejemplo:

*La dirección de IP **160** que apunta a nuestro cliente **josue.zonajosue.local**.*

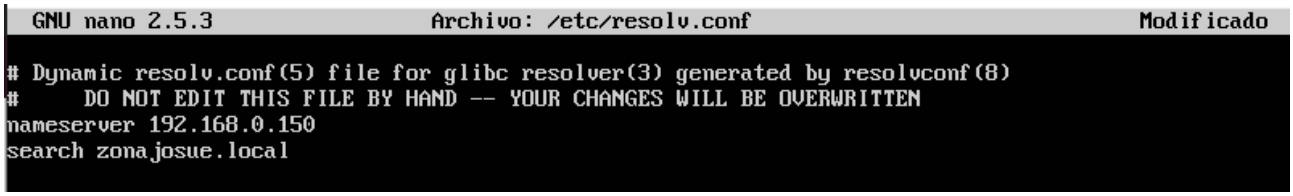
Comprobación de errores se sintaxis en DNS Ubuntu

Ahora comprobaremos que las tablas de búsqueda directa e inversa tengan la sintaxis correcta, con el comando **named-checkzone zonajosue.local /etc/bind/db.zonajosue.local** y **named-checkzone zonajosue.local /etc/bind/db.192.168.0.rev** respectivamente. Si todo esta correcto nos devolverá OK.



Especificar la IP y el dominio donde hacer las peticiones DNS el servidor

Editaremos el fichero **/etc/resolv.conf** que el servidor DNS somos nosotros, indicaremos la IP y el dominio donde se realizaran las busquedas.



Luego de esto reiniciamos bind con la siguiente instrucción.

/etc/init.d/bind9 restart.

Pruebas de funcionamiento del servidor DNS primario.

Comprobaremos el correcto funcionamiento de nuestro servidor con un cliente. Resolviendo tanto por IP como por Nombres.

Servidor.

Primero que anda probaremos el funcionamiento del servidor dns desde nuestro propio servidor. Sobre nosotros mismos.

Nslookup

```
root@josue:/etc/bind# nslookup zonajosue.local
Server:         192.168.0.150
Address:        192.168.0.150#53

Name:   zonajosue.local
Address: 192.168.0.150

root@josue:/etc/bind# nslookup 192.168.0.150
Server:         192.168.0.150
Address:        192.168.0.150#53

150.0.168.192.in-addr.arpa      name = zonajosue.local.
```

Host

```
root@josue:/etc/bind# host zonajosue.local
zonajosue.local has address 192.168.0.150
```

Dig

```
root@josue:/etc/bind# dig zonajosue.local

; <<>> DiG 9.10.3-P4-Ubuntu <<>> zonajosue.local
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32975
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;zonajosue.local.               IN      A

;; ANSWER SECTION:
zonajosue.local.        604800  IN      A       192.168.0.150

;; AUTHORITY SECTION:
zonajosue.local.        604800  IN      NS      zonajosue.local.

;; Query time: 0 msec
;; SERVER: 192.168.0.150#53(192.168.0.150)
;; WHEN: Sat Oct 20 17:26:13 WEST 2018
;; MSG SIZE  rcvd: 74
```

Cliente.

Nslookup

```
root@josue:/home/josue# nslookup www.amazon.es
Server:         192.168.0.150
Address:        192.168.0.150#53

Non-authoritative answer:
www.amazon.es   canonical name = www.cdn.amazon.es.
www.cdn.amazon.es       canonical name = www.amazon.es.edgekey.net.
www.amazon.es.edgekey.net       canonical name = e15319.ci.akamaiedge.net.
Name:   e15319.ci.akamaiedge.net
Address: 23.60.210.231

root@josue:/home/josue# nslookup 8.8.8.8
Server:         192.168.0.150
Address:        192.168.0.150#53

Non-authoritative answer:
8.8.8.8.in-addr.arpa    name = google-public-dns-a.google.com.

Authoritative answers can be found from:
.       nameserver = c.root-servers.net.
.       nameserver = f.root-servers.net.
.       nameserver = g.root-servers.net.
.       nameserver = l.root-servers.net.
.       nameserver = a.root-servers.net.
.       nameserver = j.root-servers.net.
.       nameserver = e.root-servers.net.
.       nameserver = b.root-servers.net.
.       nameserver = m.root-servers.net.
.       nameserver = d.root-servers.net.
.       nameserver = k.root-servers.net.
.       nameserver = h.root-servers.net.
.       nameserver = i.root-servers.net.
```

Host

```
root@josue:/home/josue# host www.amazon.es
www.amazon.es is an alias for www.cdn.amazon.es.
www.cdn.amazon.es is an alias for www.amazon.es.edgekey.net.
www.amazon.es.edgekey.net is an alias for e15319.ci.akamaiedge.net.
e15319.ci.akamaiedge.net has address 104.83.80.215
root@josue:/home/josue# host 104.83.80.215
215.80.83.104.in-addr.arpa domain name pointer a104-83-80-215.deploy.static.akamaitechnologies.com.
root@josue:/home/josue#
```

Dig

```
root@josue:/home/josue# dig @192.168.0.150 www.google.es

; <<>> DiG 9.10.3-P4-Ubuntu <<>> @192.168.0.150 www.google.es
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 56359
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 13, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.google.es.                 IN      A

;; ANSWER SECTION:
www.google.es.          262     IN      A       172.217.16.227

;; AUTHORITY SECTION:
.                       38880   IN      NS      g.root-servers.net.
.                       38880   IN      NS      h.root-servers.net.
.                       38880   IN      NS      f.root-servers.net.
.                       38880   IN      NS      l.root-servers.net.
.                       38880   IN      NS      b.root-servers.net.
.                       38880   IN      NS      d.root-servers.net.
.                       38880   IN      NS      i.root-servers.net.
.                       38880   IN      NS      j.root-servers.net.
.                       38880   IN      NS      a.root-servers.net.
.                       38880   IN      NS      m.root-servers.net.
.                       38880   IN      NS      c.root-servers.net.
.                       38880   IN      NS      k.root-servers.net.
.                       38880   IN      NS      e.root-servers.net.

;; Query time: 0 msec
;; SERVER: 192.168.0.150#53(192.168.0.150)
;; WHEN: Sat Oct 20 18:53:56 WEST 2018
;; MSG SIZE  rcvd: 269
```

Configuración de DNS secundario.

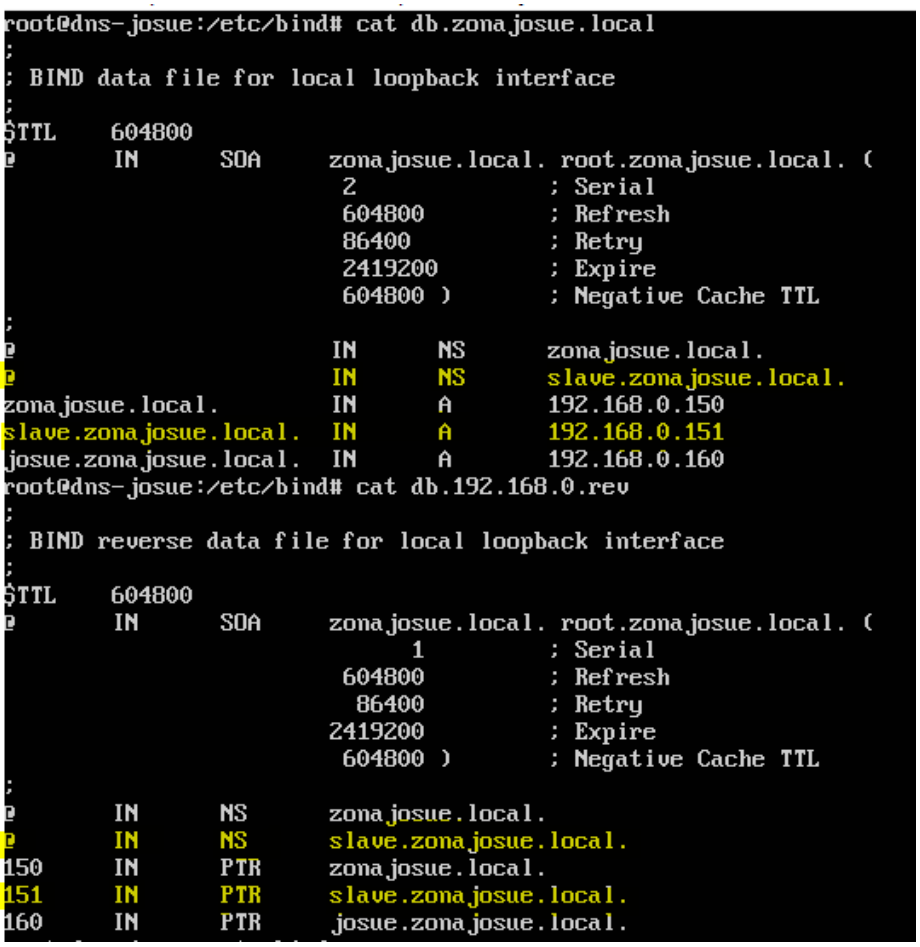
Para esto usaremos otra maquina virtual en la que tendremos que instalar y configurar bind9 para que funciones como un servidor secundario.

Para facilitarnos las cosas clonaremos la maquina del servidor principal que teníamos y asi nos ahorramos configuración y solo tendremos que modificar determinados ficheros.

En el servidor principal.

Aquí solo tendremos que añadir un par de lineas en algunos de ficheros.

En las tablas de configuración tanto los ficheros de la zona directa como inversa añadiremos nuestro segundo servidor DNS. Tal y como lo vemos en las siguientes imágenes.



En el fichero de configuración **/etc/bind/named.conf.local** podemos utilizar **also-notify** para mantener los DNS sincronizados. Con **also-notify** pasamos los cambios de zonas en el maestro al esclavo:

```
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

//zona directa
zone "zonajosue.local" {
        type master;
        also-notify {192.168.0.151;};
        file "/etc/bind/db.zonajosue.local";
};

//zona inversa
zone "0.168.192.in-addr.arpa"{
        type master;
        also-notify {192.168.0.151;};
        file "/etc/bind/db.192.168.0.rev";
};
```

En el servidor secundario.

En el archivo **/etc/bind/named.conf.local** del servidor DNS esclavo debemos indicar que se trata de un servidor esclavo y también debemos indicar quién es el maestro:

```
root@dns-josue:/home/josue# cat /etc/bind/named.conf.local
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

//zona directa exclavo
zone "zonajosue.local" {
        type slave;
        file "/etc/bind/db.zonajosue.local";
        masters {192.168.0.150;};
};


//zona inversa exlavo
zone "0.168.192.in-addr.arpa" {
        type slave;
        file "/etc/bind/db.192.168.0.rev";
        masters {192.168.0.150;};
};
```

De esta forma dispondremos en la red de un servidor DNS secundario que podrá satisfacer las peticiones DNS al igual que lo haría el maestro.

Tanto como si el número de peticiones es muy elevado y se requiere distribuir la carga entre los dos servidores, o si deseamos disponer de servicio DNS de alta disponibilidad de forma que aunque el servidor maestro deje de funcionar, el servidor esclavo podrá seguir ofreciendo el servicio.
Cada vez que hagamos un cambio en las tablas de registros inversas o directas del maestro, debemos acordarnos de actualizar el parámetro serial (incrementar en una unidad) para que los DNS dependientes del maestro sepan que ha cambiado y actualicen su información para mantenerse perfectamente sincronizados.

Realizando pruebas con el servidor secundario.

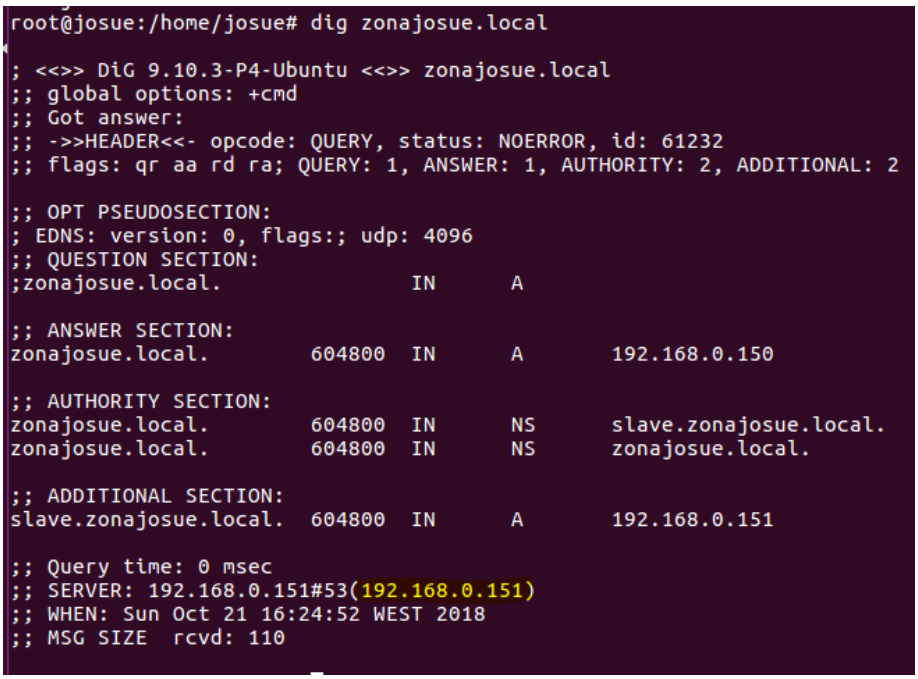
Para ver si el servidor secundario esta funcionando primero que nada veremos si nuestro cliente puede resolverlo usando **nslookup**.

```
root@josue:/home/josue# nslookup slave.zonajosue.local
Server:         192.168.0.150
Address:        192.168.0.150#53

Name:   slave.zonajosue.local
Address: 192.168.0.151
```

Ahora realizaremos la prueba que nos hará ver si realmente funciona nuestro servidor secundario, primero que nada pararemos el servidor primario y luego resolveremos usando el servidor secundario.

```
root@josue:/home/josue# nslookup slave.zonajosue.local
Server:         192.168.0.151
Address:        192.168.0.151#53

Name:   slave.zonajosue.local
Address: 192.168.0.151
```

Con esto vemos que el que nos resuelve es el secundario y no el primario.



Realizando las pruebas con dig, nos da el mismo resultado.