Configurar Servidor DNS con Bind9 en Debian 11

¿Alguna vez han tenido la necesidad de configurar Bind9 como **Servidor DNS Recursivo/Cache** y como **Servidor DNS Autoritario** para su red local? Lo que te demuestro con éste artículo es para que puedas aplicar estas configuraciones.

Servidor DNS Recursivo/Cache

Un **DNS Recursivo**, tambien conocido como solucionador recursivo, es el que reenvia todas las consultas o solicitudes DNS hacia los **Servidores DNS Externos** y éstos servidores externos son tres: **Servidores DNS Raices**, **Servidores DNS de Primer Nivel (TLD)**, y **Servidores DNS Autoritarios**. Un **Servidor DNS Recursivo** actua como **intermediario** entre un cliente y un servidor de nombres DNS. Cuando en los **Servidores DNS Externos** es encontrada la solicitud **(Dirección IP)** del dominio, el **DNS Recursivo** envía de regreso al cliente la consulta peticionada.

Un **DNS Cache**, es un registro en dónde se almacena los sitios web visitados. Cuando un usuario visita un sitio web en el navegador, el cliente **(ordenador)** realiza la consulta del nombre de dominio que se está solicitando. Si el sitio web **(nombre de dominio)** consultada ya se encuentra registrada en el **Cache del Servidor DNS**, éste le regresa inmediatamente la solicitud DNS al cliente; pero si el dominio consultada por el cliente no se encuentra registrada en el Cache, entonces ahí es cuando entra en función del **DNS Recursivo** hasta que la solicitud DNS es encontrado y enviado de vuelta hacia al **Cache del Servidor DNS Local** y también hacia al cliente quién realizó la consulta. Todo este proceso sucede en milisegundos. **Otro factor importante es que la Cache del DNS también lo mantiene el navegador y el sistema operativo del cliente.**

Servidor DNS Autoritario

Un **DNS Autoritario**, es aquí donde se almacena en múltiples base de datos diferentes tipos de registros DNS **(NS, SOA, A, AAAA, MX, TXT, SRV, PTR, etc.)**. Estas base de datos son zonas directas y reversas de múltiples dominios públicos y/o privados. Este tipo de servidor DNS es el último en ser consultado durante el proceso de un **DNS Recursivo** cuando el cliente solicita la Dirección IP de un dominio.

Instalación del BIND DNS

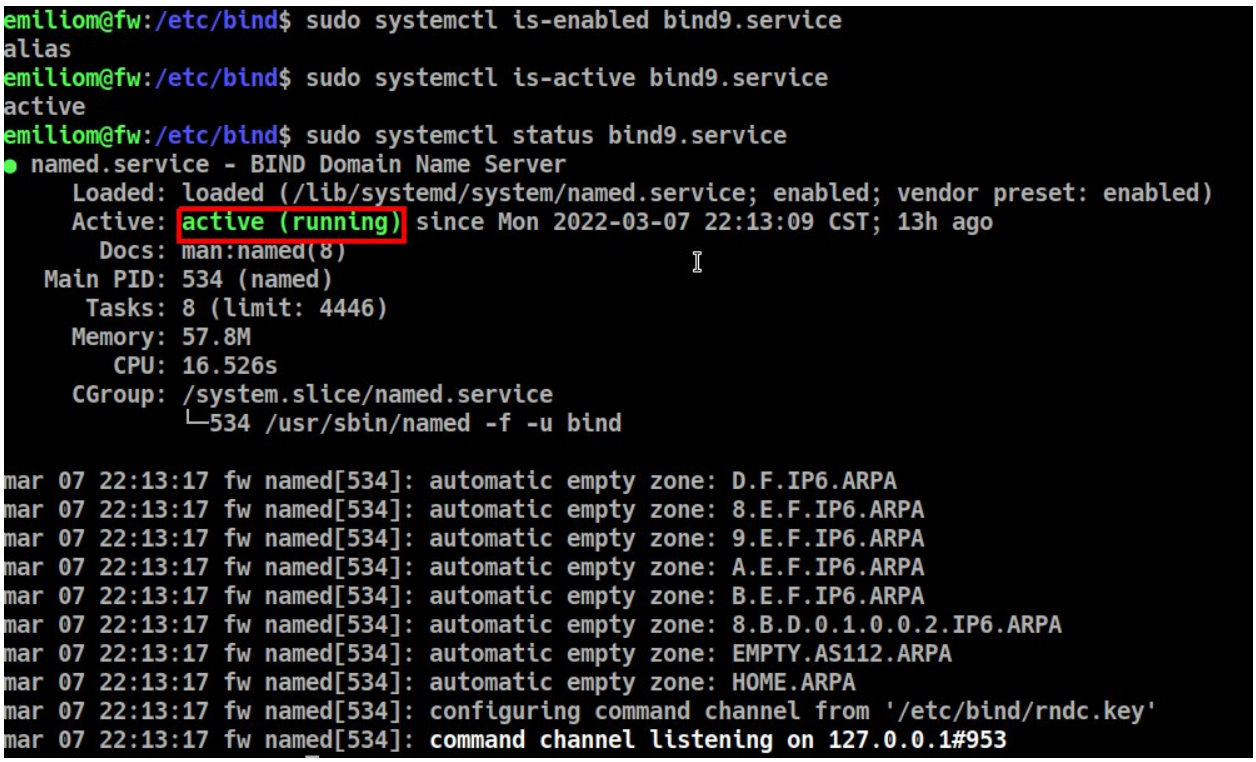
Para la instalación del Servidor BIND DNS, debemos ejecutar el siguiente comando:

```
sudo apt install dnsutils bind9-utils bind9
```

Una vez instalado, podemos verificar su instalación de la siguiente manera.

Habilitar y Activar BIND DNS

Verificamos que el servicio Bind9 esté habilitado y activo.



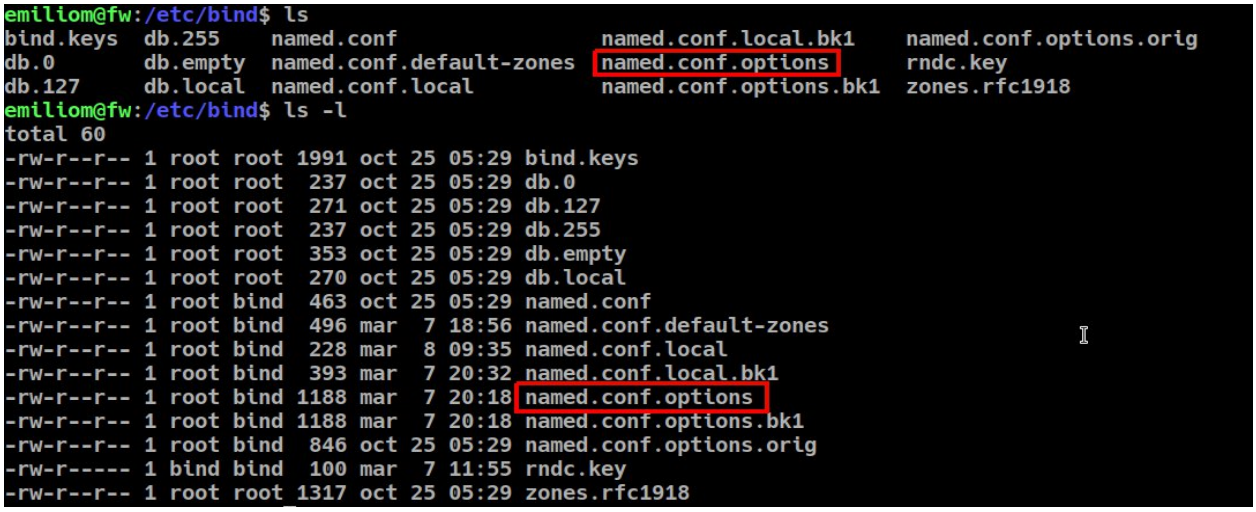
Si el servicio Bind9 no están habilitado ni tampoco activado, se deben ejecutar los siguientes comandos:

```
sudo systemctl enable bind9.service

sudo systemctl start bind9.service
```

Configurar BIND como DNS Recursivo/Cache

Para configurar el **DNS Recursivo y Cache** con **BIND DNS**, lo primero que deben hacer es conocer la ruta en dónde se almacena todos los archivos de configuración. La ruta es la siguiente: **/etc/bind**. Dentro de éste directorio, encontraremos los siguientes archivos de configuración.



Como pueden observar, el nombre del archivo que hay que configurar se llama **named.conf.options** y dentro de éste archivo, agregamos los siguientes parámetros.

acl mgnetwork { 172.16.20.128/25; };

```
options {
        directory "/var/cache/bind";
        dnssec-validation auto;
        listen-on port 53 { 127.0.0.1; 172.16.20.254; };
        listen-on-v6 { none; };
        querylog yes;
        max-cache-size 512m;
        max-cache-ttl 60;
        max-ncache-ttl 60;
        cleaning-interval 30;
        recursion yes;
        allow-recursion { 127.0.0.1; mgnetwork; };
        allow-query { 127.0.0.1; mgnetwork; };
        allow-transfer { none; };
        forwarders { 186.2.137.13; 186.2.141.133; 8.8.8.8; 8.8.4.4; 1.1.1.1; 1.0.0.1; };
        forward first;
        pid-file "/run/named/named.pid";
        session-keyfile "/run/named/session.key";
        managed-keys-directory "/var/cache/bind";
        bindkeys-file "/etc/bind/bind.keys";
};
```

Lo que observan arriba, son un montón de parámetros de configuración que obviamente ustedes van a tener que cambiarlos en base a sus requerimientos y necesidades. Los parámetros que ustedes van a tener que cambiar son los siguientes:

- **ACL (Lista de Control de Acceso)**, es decir, el nombre y la red del mismo.
- Las Direcciones IP por medio de dónde se escucharán todas las peticiones DNS por el puerto 53. **(listen-on porte 53 { direcciones ip de su servidor })**.
- Servidores DNS Externos hacia dónde se reenviarán las peticiones DNS recursivamente. **(forwarders)**.

- Los permisos de recursión y petición DNS en base a su ACL. **(allow-recursion y allow-query)**.
- El tamaño máximo para el almacenamiento cache. **(max-cache-size)**. Lo sugerible es que sea **512 megabytes o 1024 megabytes (1 gigabyte)** pero esto depende mucho de la **RAM** del servidor.
- El límite de tiempo para mantener los registros DNS en el Cache. **(max-cache-ttl y max-ncache-ttl)**.

Configurar BIND como DNS Autoritario

Aprovechando la misma ruta del Servidor BIND DNS **(/etc/bind)**, para configurarlo como DNS Autoritario, debemos configurar el archivo **named.conf.local** para registrar las bases de datos (zonas directas y reversas de los dominios).

```
emiliom@fw:/etc/bind$ ls
bind.keys  db.255    named.conf                named.conf.local.bk1    named.conf.options.orig
db.0       db.empty  named.conf.default-zones  named.conf.options      rndc.key
db.127     db.local  named.conf.local          named.conf.options.bk1  zones.rfc1918
emiliom@fw:/etc/bind$ ls -l
total 60
-rw-r--r-- 1 root root 1991 oct 25 05:29 bind.keys
-rw-r--r-- 1 root root  237 oct 25 05:29 db.0
-rw-r--r-- 1 root root  271 oct 25 05:29 db.127
-rw-r--r-- 1 root root  237 oct 25 05:29 db.255
-rw-r--r-- 1 root root  353 oct 25 05:29 db.empty
-rw-r--r-- 1 root root  270 oct 25 05:29 db.local
-rw-r--r-- 1 root bind  463 oct 25 05:29 named.conf
-rw-r--r-- 1 root bind  496 mar  7 18:56 named.conf.default-zones
-rw-r--r-- 1 root bind  228 mar  8 09:35 named.conf.local
-rw-r--r-- 1 root bind  393 mar  7 20:32 named.conf.local.bk1
-rw-r--r-- 1 root bind 1212 mar  8 12:23 named.conf.options
-rw-r--r-- 1 root bind 1188 mar  7 20:18 named.conf.options.bk1
-rw-r--r-- 1 root bind  846 oct 25 05:29 named.conf.options.orig
-rw-r----- 1 bind bind  100 mar  7 11:55 rndc.key
-rw-r--r-- 1 root root 1317 oct 25 05:29 zones.rfc1918
```

En mi caso solo se registraron dos zonas **(una directa y otra reversa)** para dos dominios privados.

```
zone "mgnetwork.home" {

        type master;

        file "/var/lib/bind/db.mgnetwork.home.hosts";

        allow-update { none; };

};

zone "20.16.172.in-addr.arpa" {

        type master;

        file "/var/lib/bind/db.172.16.20.rev";

        allow-update { none; };

};
```

Si observan detalladamente, ambas zonas son de **tipo maestro** y ambas base de datos fueron almacenados en el directorio **/var/lib/bind**.

Los registros DNS para la base de datos de la zona directa **(mgnetwork.home)**, son los siguientes:

```
$ORIGIN mgnetwork.home.
$TTL    604800
@       IN      SOA     mgnetwork.home.    root.localhost. (
                        2022030701              ; Serial
                             604800             ; Refresh
                              86400             ; Retry
                            2419200             ; Expire
                             604800 )       ; Negative Cache TTL
;
@       IN      NS      mgnetwork.home.
@       IN      A       172.16.20.254
fw      IN      A       172.16.20.254
rmpc    IN      A       172.16.20.140
```

Los registros DNS para la base de datos de la zona reversa **(20.16.172.in-addr.arpa)**, son los siguientes:

```
$ORIGIN 20.16.172.in-addr.arpa.
$TTL    604800
@       IN      SOA     mgnetwork.home.    root.localhost. (
                        2022030702              ; Serial
                             604800             ; Refresh
                              86400             ; Retry
                            2419200             ; Expire
                             604800 )       ; Negative Cache TTL
;
@       IN      NS      mgnetwork.home.
254     IN      PTR     mgnetwork.home.
254     IN      PTR     fw.mgnetwork.home.
140     IN      PTR     rmpc.mgnetwork.home.
```

Es muy importante tomar en cuenta que los **números seriales** de cada zona deben ser muy diferentes y colocar al final de cada dominio o FQDN un **punto (.)**.

Configuración del Cortafuego

Para que los clientes puedan realizar las consultas DNS, es necesario permitir dichas solicitudes DNS y esto se hace mediante un cortafuego o firewall. En mi caso, estoy utilizando Shorewall como cortafuego y para esto se debe agregar una regla similar al que demuestro abajo, dentro del archivo de configuración **/etc/shorewall/rules**.

?COMMENT Permitir a la Red Local resolver Nombres de Dominios mediante Servidor DNS Local

DNS(ACCEPT) loc:172.16.20.128/25 fw:172.16.20.254

Sin embargo, si el cortafuego solo fuera con Iptables Puro (sin Shorewall, sin UFW, sin FirewallD, etc.), se debe agregar algunas reglas tanto en la cadena INPUT como también en la cadena FORWARD. **En la cadena INPUT**, las reglas serían para permitir a los clientes enviar las consultas DNS con el Servidor DNS Local y **en la cadena FORWARD**, las reglas serían para permitir al Servidor DNS Local reenviar las consultas DNS hacia los Servidores DNS Externos.

Entonces en la cadena INPUT, las reglas serían los siguientes maneras:

```
sudo iptables -A INPUT -i interfaz LAN -s red local -d ip del servidor dns -m
state --state NEW,ESTABLISHED -p udp --dport 53 -j ACCEPT

sudo iptables -A INPUT -i interfaz LAN -s red local -d ip del servidor dns -m
state --state NEW,ESTABLISHED -p tcp --dport 53 -j ACCEPT
```

Entonces en la cadena FORWAD, las reglas serían los siguientes maneras:

```
sudo iptables -A FORWARD -i interfaz LAN -o interfaz WAN -s red local -m
state --state NEW,ESTABLISHED -p udp --dport 53 -j ACCEPT

sudo iptables -A FORWARD -i interfaz LAN -o interfaz WAN -s red local -m
state --state NEW,ESTABLISHED -p tcp --dport 53 -j ACCEPT
```

Con éstas configuraciones en el cortafuego debe funcionar las consultas DNS adecuadamente.

Diagnósticos DNS

Para los diagnosticos DNS, realizamos dos consultas DNS, una directa y otra reversa.

Consulta DNS Directa:

```
emiliom@fw:~$ dig mgnetwork.home

; <<>> DiG 9.16.22-Debian <<>> mgnetwork.home
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 14683
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 017ee3c66e17b31f010000006227ba288c5b05ccdab1a8d5 (good)
;; QUESTION SECTION:
;mgnetwork.home.                        IN      A

;; ANSWER SECTION:
mgnetwork.home.         604800  IN      A       172.16.20.254

;; Query time: 4 msec
;; SERVER: 172.16.20.254#53(172.16.20.254)
;; WHEN: Tue Mar 08 14:18:48 CST 2022
;; MSG SIZE  rcvd: 87
```

Consulta DNS Reversa:

```
emiliom@fw:~$ dig -x 172.16.20.254

; <<>> DiG 9.16.22-Debian <<>> -x 172.16.20.254
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1997
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: d5dbe55c529b83ad010000006227baa44f52119b83d16c41 (good)
;; QUESTION SECTION:
;254.20.16.172.in-addr.arpa.    IN      PTR

;; ANSWER SECTION:
254.20.16.172.in-addr.arpa. 604800 IN   PTR     mgnetwork.home.
254.20.16.172.in-addr.arpa. 604800 IN   PTR     fw.mgnetwork.home.

;; Query time: 4 msec
;; SERVER: 172.16.20.254#53(172.16.20.254)
;; WHEN: Tue Mar 08 14:20:52 CST 2022
;; MSG SIZE  rcvd: 128
```

Como pueden observar, todo funciona perfectamente y adecuadamente.

Saludos.