Instalación del servidor DNS maestro con Bind9

UT2: Servicio de Nombres de Dominio (DNS)

DESPLIEGUE DE APLICACIONES WEB

Si con las posibilidades que nos ofrece dnsmasq no son suficientes para nuestra red y necesitamos un servidor DNS más completo, podemos utilizar el paquete **bind9**. Para instalarlo, podemos hacerlo con apt-get desde una consola de root:

```
// Instalación del servidor DNS bind
# apt-get install bind9
```

De esta forma instalaremos los programas necesarios para disponer de un completo servidor DNS con bind. Tan solo será necesario configurarlo y ponerlo en marcha.

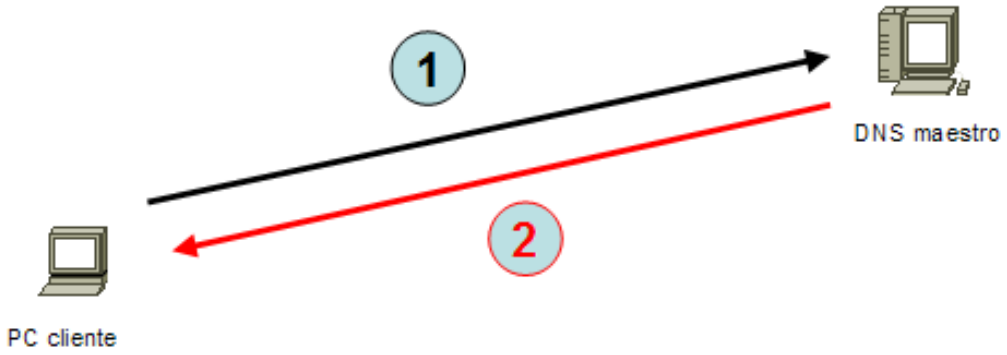
Configuración del servidor DNS

El servidor DNS bind admite tres modos de funcionamiento:

- Servidor DNS maestro
- Servidor DNS esclavo
- Servidor caché DNS

Servidor DNS maestro

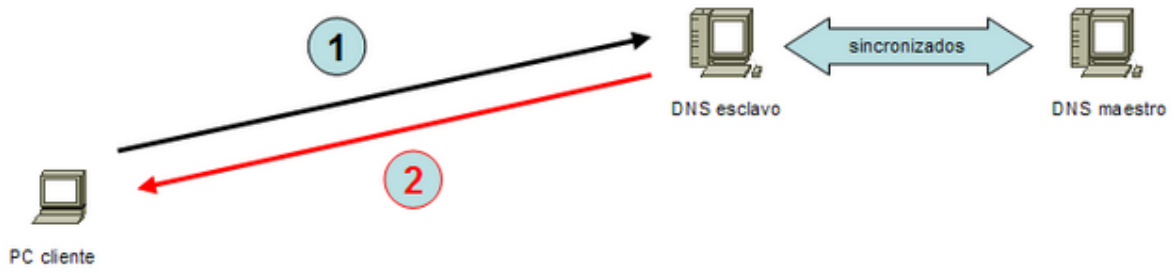
En este modo de funcionamiento, nuestro servidor se comporta como un auténtico servidor DNS para nuestra red local. Atenderá directamente a las peticiones de resolución de direcciones pertenecientes a la red local y reenviará a servidores DNS externos las peticiones del resto de direcciones de Internet.



Consulta a un DNS maestro

Servidor DNS esclavo

Un servidor esclavo actuará como un servidor espejo de un servidor DNS maestro. Permanecerá sincronizado con el maestro. Se utilizan para repartir las peticiones entre varios servidores aunque las modificaciones solo se realicen en el maestro. En redes locales salvo por razones de disponibilidad, es raro que exista la necesidad de tener dos servidores DNS ya que con uno será suficiente.



Consulta a un DNS esclavo

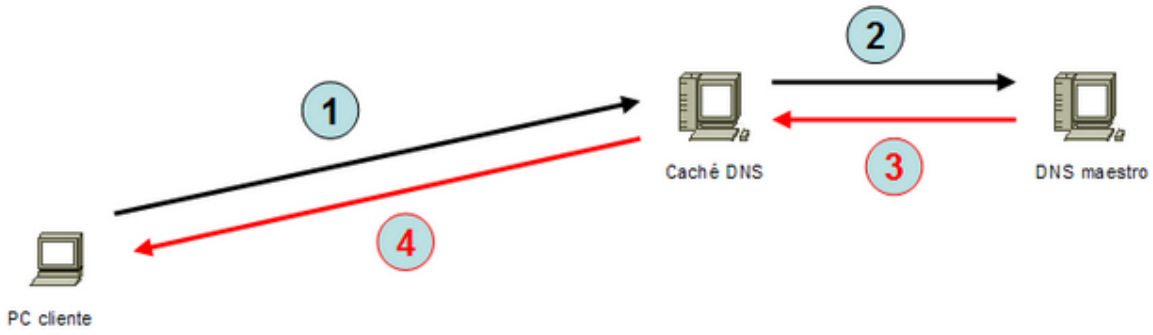
Servidor caché DNS

En este modo de funcionamiento, nuestro servidor se comporta como si fuera un auténtico servidor DNS para nuestra red local aunque realmente no sea un servidor DNS propiamente dicho. Cuando recibe una petición de DNS por parte de un cliente de nuestra red, la trasladará a un DNS maestro que puede estar en nuestra

red o fuera, almacenará en una memoria caché la respuesta y a la vez la comunicará a quien hizo la petición. Si un segundo cliente vuelve a realizar la misma petición, como nuestro servidor tiene la respuesta almacenada en su memoria caché, responderá inmediatamente sin tener que cursar la petición a ningún servidor DNS de Internet.

Disponer de un servidor caché DNS en nuestra red local aumenta la velocidad de la conexión a Internet pues cuando navegamos por diferentes lugares, continuamente se están realizando peticiones DNS. Si nuestro caché DNS almacena la gran mayoría de peticiones que se realizan desde la red local, las respuestas de los clientes se satisfarán prácticamente de forma instantánea proporcionando al usuario una sensación de velocidad en la conexión.

Es un modo de funcionamiento de sencilla configuración ya que prácticamente lo único que hay que configurar son las direcciones IP de un DNS primario y de un DNS secundario. Muchos routers ADSL ofrecen ya este servicio de caché, tan solo hay que activarlo y configurar una o dos IPs de servidores DNS en Internet. En los PCs de nuestra red local podríamos poner como DNS primario la IP de nuestro router y como DNS secundario una IP de un DNS de Internet.



Consulta a un caché DNS. En caso de fallo, se redirecciona hacia un DNS maestro

Archivos de configuración del DNS

El archivo de configuración del DNS es el archivo /etc/bind/named.conf, pero este hace referencia a otros cuantos archivos como por ejemplo:

- Archivo named.conf: Archivo principal de configuración
- Archivo named.conf.options: Opciones genéricas
- Archivo named.conf.local: Especificación particular de este servidor DNS
- Archivo db.127:Especificación dirección de retorno
- Archivo db.root: DNS de nivel superior
- Otros archivos: db.0, db.255, db.empty, db.local, rndc.conf, rndc.key, zones.rfc1918

Configuración como caché DNS

Por defecto, al instalar el paquete bind está preconfigurado como servidor caché DNS. Tan solo será necesario editar el archivo /etc/bind/named.conf.options y en la sección forwarders añadir las IPs de dos servidores DNS donde redirigir las peticiones DNS:

```
// Configuración como caché DNS
// Añadir IPs de los DNS de nuestro proveedor en /etc/bind/named.conf.options
options {
forwarders {
80.58.0.33; 80.58.32.97;
};
};
```

Configuración DNS maestro

Por razones de accesibilidad y organizativas, deseamos asignar un nombre a todos los equipos de nuestra red, para lo que instalaremos un servidor DNS privado con un dominio ficticio, por ejemplo '**aula129.org**'. Todos los PCs de nuestra red pertenecerán a dicho dominio ficticio que funcionará solo en nuestra red interna, no en Internet. En tal caso el nombre completo de los PCs terminará con '**aula129.org**', por ejemplo: **equipo12.aula129.org**. Lo ideal en una situación así es disponer de un servidor DNS que sea maestro de nuestro dominio, es decir, maestro del dominio interno '**aula129.org**'.

Nuestro servidor DNS maestro para nuestro dominio ficticio interno 'aula129.org' será capaz de resolver peticiones internas de nombres de este dominio, tanto de forma directa como de forma inversa, es decir, si recibe una consulta acerca de quién es **equipo12.aula129.org** deberá devolver su IP, pongamos por ejemplo 192.168.115.212. Si la consulta es una consulta DNS inversa acerca de quién es 192.168.115.212, deberá responder **equipo12.aula129.org**. Por ello deberemos añadir en el archivo **/etc/bind/named.conf.local** la especificación de maestro para el dominio y para la resolución inversa, por ejemplo:

```
// Añadir en /etc/bind/named.conf.local
// Archivo para búsquedas directas
zone "zona4.org" {
type master;
file "/etc/bind/db.zona4.org";
};
```

```
// Archivo para búsquedas inversas
zone "129.168.192.in-addr.arpa" {
type master;
file "/etc/bind/192.rev";
};
```

Evidentemente será necesario crear los archivos **db.zona4.org** y **192.rev** que especificarán la asociación entre nombres y direcciones IP de nuestra red en un sentido y en otro respectivamente.

Archivo de zona de búsqueda directa

Supongamos que en nuestra red local tenemos un aula con 11 PCs con IPs que van desde la 192.168.115.101 hasta 192.168.115.110 y cuyos nombres van desde equipo01 hasta equipo10, luego un servidor web (ubuntuserver00) que además es servidor DNS con la IP 192.168.115.111. El archivo de configuración DNS de nuestro dominio podría ser así:

```
;
; BIND data file for zona4.org
;
$TTL    1D
@ IN SOA zona4.org. root.zona4.org. (
        1 ; Serial
        604800 ; Refresh
        86400 ; Retry
        2419200 ; Expire
        604800 ) ; Default TTL

; Servidores DNS del dominio
        IN      NS      equipo00.zona4.org.

; Hosts
equipo00 IN A 192.168.129.200
equipo01 IN A 192.168.129.201
equipo02 IN A 192.168.129.202
equipo03 IN A 192.168.129.203
```

```
equipo04 IN A 192.168.1.204
equipo05 IN A 192.168.1.205
equipo06 IN A 192.168.1.206
equipo07 IN A 192.168.1.207
equipo08 IN A 192.168.1.208
equipo09 IN A 192.168.1.209
equipo10 IN A 192.168.1.210

; Alias
www IN A 192.168.1.200
```

Las primeras líneas son unos parámetros relacionados con la actualización del DNS (número de serie y períodos de actuación). Las dos siguientes líneas indican quién es el servidor primario (NS = Name Server) . Las siguientes líneas especifican las IPs de los distintos PCs componentes del dominio (A = Address).

Si olvidamos algún punto y coma, dará errores y no funcionará correctamente. Para revisar los archivos disponemos de los comandos **named-checkconf** y **named-checkzone** que analizan que esté correcta la sintaxis de los mismos.

También se puede consultar el archivo de logs del sistema para comprobar si existe algún error:

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

Para comprobar que el archivo de resolución directa está correctamente, se debe ejecutar:

```
named-checkzone aula129.org /etc/bind/db.aula129.org
```

Y la salida que se obtiene es:

```
zone aula129.org/IN: loaded serial 1
OK
```

Archivo de zona de búsqueda inversa

Para poder realizar consultas inversas (de IP a nombre) será necesario crear el siguiente archivo:

```
;
; BIND reverse data file for 192.168.1.0
;
```

```
$TTL    1D
@ IN SOA 1.168.192.in-addr.arpa. root.aula129.org. (
        1 ; Serial
        604800 ; Refresh
        86400 ; Retry
        2419200 ; Expire
        604800 ) ; Default TTL

        IN      NS      equipo00.aula129.org.

200 IN PTR equipo00.aula129.org.
201 IN PTR equipo01.aula129.org.
202 IN PTR equipo02.aula129.org.
203 IN PTR equipo03.aula129.org.
204 IN PTR equipo04.aula129.org.
205 IN PTR equipo05.aula129.org.
206 IN PTR equipo06.aula129.org.
207 IN PTR equipo07.aula129.org.
208 IN PTR equipo08.aula129.org.
209 IN PTR equipo09.aula129.org.
210 IN PTR equipo10.aula129.org.
211 IN PTR equipo11.aula129.org.
200 IN PTR www.aula129.org.
```

Para comprobar que el archivo de zona inversa está correctamente, se debe ejecutar:

```
named-checkzone 1.168.192.in-addr.arpa /etc/bind/192.rev
```

La salida que nos devuelve la línea anterior será:

```
zone 1.168.192.in-addr.arpa/IN: loaded serial 1
OK
```

Una vez configurado nuestro servidor DNS, debemos indicar a nuestro servidor Linux que el servidor DNS es él mismo, modificando en las líneas necesarias el archivo **/etc/network/interfaces**.

```
// Indicamos que nosotros mismos somos servidores DNS
```

```
// y por defecto buscamos en nuestro dominio
// Editar /etc/network/interfaces del servidor DNS
dns-nameservers 127.0.0.1
dns-search aula129.org
```

En el resto de PCs de la red, indicaremos que el servidor DNS es 192.168.1.200

```
// En el resto de PCs de la red indicamos quién es el DNS
// Editar /etc/network/interfaces del resto de PCs de la red
....
dns-nameserver 192.168.1.200
```

Tan solo nos faltará poner en marcha nuestro servidor de nombres ejecutando en el servidor el script de inicio correspondiente:

```
// Arranque del servidor DNS
# /etc/init.d/bind9 restart
```

y, mediante el comando host, el comando dig o el comando nslookup hacer alguna consulta de prueba.

Comando host:

```
root@equipo00:/home/monterona# host equipo01
equipo01.aula129.org has address 192.168.1.201
```

Comando dig:

```
root@equipo00:/home/monterona# dig equipo07

; <<>> DiG 9.10.3-P4-Ubuntu <<>> equipo07
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 29906
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 1,
ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
```

```
;; QUESTION SECTION:
;equipo07.                  IN    A

;; AUTHORITY SECTION:
.               10800     IN    SOA   a.root-servers.net.
nstld.verisign-grs.com. 2017012301 1800 900 604800 86400

;; Query time: 27 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Mon Jan 23 23:47:13 CET 2017
;; MSG SIZE  rcvd: 112
```

Comando nslookup:

```
root@equipo00:/home/monterona# nslookup 192.168.1.204
Server:         127.0.0.1
Address:   127.0.0.1#53

204.1.168.192.in-addr.arpa  name = equipo04.aula129.org.


root@equipo00:/home/monterona# nslookup equipo04
Server:         127.0.0.1
Address:   127.0.0.1#53

Name: equipo04.aula129.org
Address: 192.168.1.204
```

Configuración DNS esclavo

Si deseamos configurar nuestro servidor DNS para que actúe como esclavo de un servidor DNS maestro, la configuración es mucho más sencilla que en el caso anterior ya que únicamente será necesario indicar en el DNS esclavo quién es el servidor DNS maestro, y en el DNS maestro, la IP del DNS esclavo.

Ejemplo, supongamos que el nombre del DNS maestro es dns.aula129.org (IP 192.168.aula.x1) y que el nombre del DNS esclavo es dns2.aula129.org. En el archivo 'db.aula129.org de zona de búsqueda directa añadiremos la línea del segundo dns justo debajo de donde está la del primero:

```
// Añadir línea en /etc/bind/db.aula129.org del maestro
....
IN NS dns.aula129.org.
IN NS dns2.aula129.org. // Nueva línea
....
```

De esta forma indicaremos que existen más servidores DNS para dicha zona. Lo mismo haremos en el archivo '192.rev' de la zona inversa:

```
// Añadir línea en /etc/bind/192.rev del maestro
....
IN NS dns.aula129.org.
IN NS dns2.aula129.org. // Nueva línea
....
```

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:

```
// Añadir en /etc/bind/named.conf.local del esclavo
zone "aula129.org" {
type slave;
file "/etc/bind/db.aula129.org";
masters { 192.168.1.200; };
};

zone "1.168.192.in-addr.arpa" {
type slave;
file "/etc/bind/192.rev";
masters { 192.168.1.200; };
};
```

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

```
// Archivo /etc/bind/named.conf.local del maestro
zone "aula129.org" {
type master;
file "/etc/bind/db.aula129.org";
also-notify {ip_del_esclavo;}
};

zone "1.168.192.in-addr.arpa" {
```

```
type master;
file "/etc/bind/192.rev";
also-notify {ip_del_esclavo;}
};
```

De esta forma dispondremos en la red de un servidor DNS esclavo que podrá satisfacer las peticiones DNS al igual que lo haría el maestro. Es interesante 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 los archivos **/etc/bind/db.aula129.org** y **/etc/bind/192.rev** 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.

Arranque y parada manual del servidor DNS

El servidor DNS, al igual que todos los servicios en Debian, dispone de un script de arranque y parada en la carpeta /etc/init.d.

```
// Arranque del servidor DNS
sudo /etc/init.d/bind9 start
// Parada del servidor DNS
sudo /etc/init.d/bind9 stop
// Reinicio del servidor DNS
sudo /etc/init.d/bind9 restart
```