Índice

1. Instalación de BIND

El servidor DNS más utilizado en Linux (y en Internet) es BIND (*Berkeley Internet Name Domain*). En Ubuntu Server, BIND puede instalarse desde el asistente de instalación inicial del sistema operativo. Si no lo hicimos en su momento, podemos instalarlo con posterioridad desde los repositorios haciendo:

```
sudo apt-get update
```

Este comando actualiza la lista local de paquetes desde las fuentes definidas en el fichero **/etc/apt/sources.list**, pero no realiza ningún cambio sobre los paquetes instalados en el sistema (si queremos saber los paquetes que podemos actualizar, debemos ejecutar **sudo apt-get upgrade** y a continuación confirmar o no el proceso de actualización).

Para instalar BIND desde los repositorios ejecutamos:

```
sudo apt-get install bind9 bind9-doc
```

1.1 Gestión de BIND

El servicio (*daemon*) de BIND se llama **named**.

Durante la instalación de BIND, se creó en **/etc/init.d/bind9** un script de gestión del servicio. También se crearon una cuenta de usuario y una cuenta de grupo llamados **bind**, que se usan para ejecutar el servicio.

El script **bind9** podemos usarlo desde la consola para ver y gestionar el estado del servicio BIND.

```
sudo service bind9 status | reload | restart | start | stop
```

Es muy útil la opción **reload**, que permite recargar la configuración desde los ficheros sin necesidad de detener el servicio.

2. Configuración de BIND mediante Webmin

2.1 Configuración básica de BIND

El módulo de gestión de BIND que nos presenta Webmin tiene el siguiente aspecto.

En la zona superior tenemos opciones de configuración, y a continuación la lista de zonas DNS configuradas en el servidor.

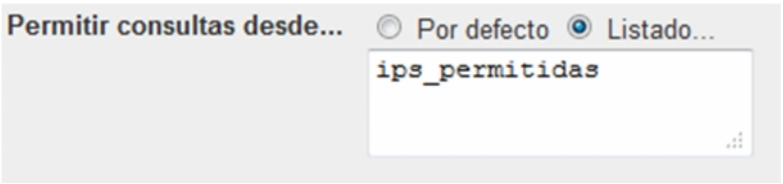
La configuración predeterminada una vez finalizada la instalación de BIND es la correspondiente a un **servidor caché de libre acceso**, esto es, responderá a las peticiones provenientes de cualquier dirección IP que lleguen al puerto 53.

Es muy importante asegurarse de tener la hora correcta en los sistemas que ejecuten un servidor DNS. Si no es el caso, el servidor devolverá un mensaje de error (*SERVFAIL*).

Limitar el acceso al servicio DNS

Una de nuestras primeras modificaciones puede ser limitar el conjunto de direcciones IP que puedan realizar solicitudes de resolución de nombres. Para lograrlo, el primer paso es crear una nueva **Lista de Control de Acceso** (**ACL**, *Access Control List*) indicando el conjunto de direcciones IP a las que permitiremos el acceso al servicio DNS, y dándole un nombre.

Una vez creada la ACL, la usaremos en la configuración de la zona o zonas a las que queramos limitar el acceso, o bien en la configuración por defecto de las zonas (en la casilla "**Permitir consultas desde…**".



Antes de aplicar la configuración (***Apply Configuration***), es conveniente comprobar que la configuración es correcta (***Check BIND Config***).

Si una vez limitado el acceso, intentamos acceder al servicio desde una IP no permitida, obtendremos un mensaje de rechazo (*query refused*).

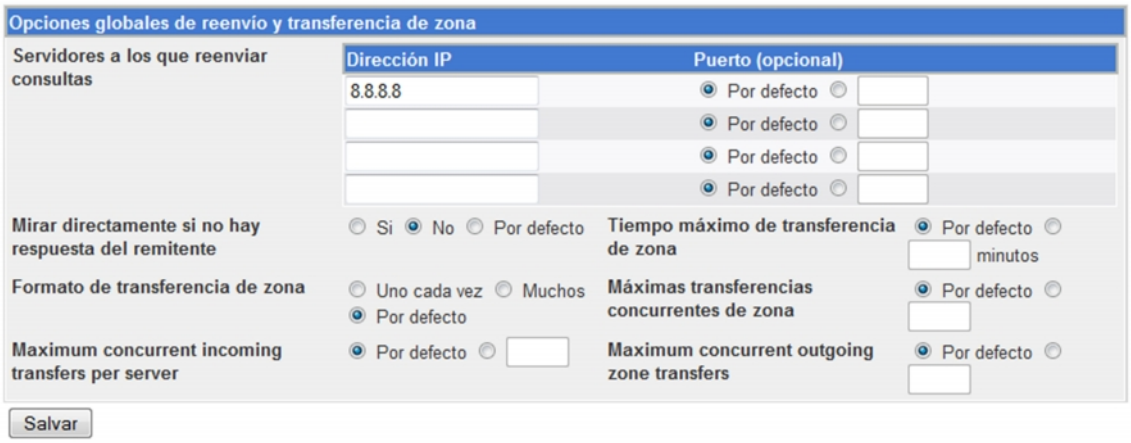
```
> www.iessanclemente.net
Servidor:  UnKnown
Address:  192.168.1.34

*** UnKnown no encuentra www.iessanclemente.net: Query refused
```

2.2 Servidor DNS reenviador (o proxy)

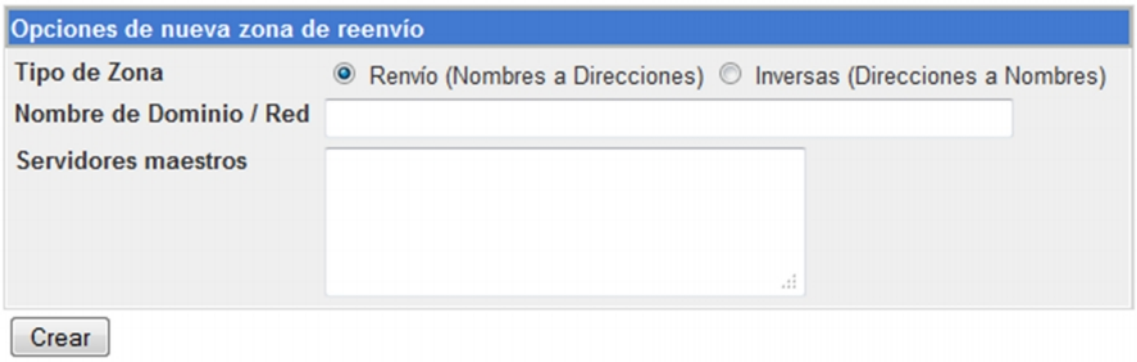
En Webmin, la configuración de un servidor de reenvío se realiza en la opción **Reenvío y Transferencias**.

Rellenaremos al menos una dirección IP de la lista "**Servidores a los que reenviar consultas**", y opcionalmente indicaremos si queremos o no que resuelva directamente las solicitudes recibidas cuando no haya respuesta del servidor al que se reenvía ("**Mirar directamente si no hay respuesta del remitente**").



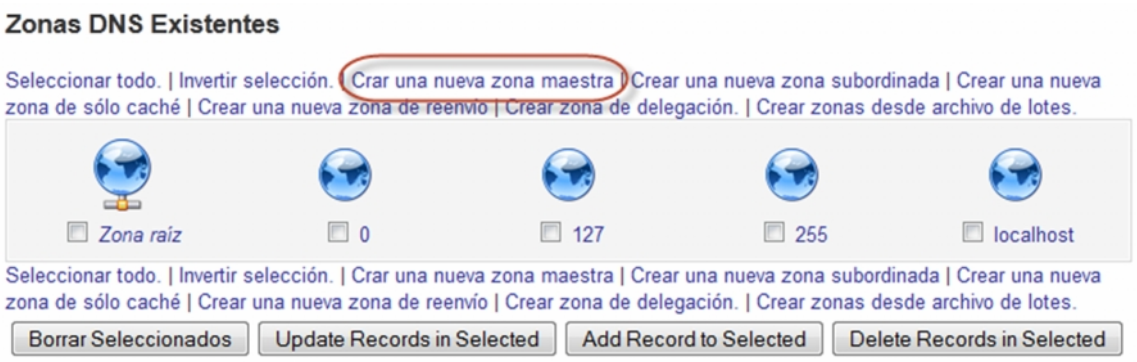
Si solo queremos que nuestro servidor reenvíe las consultas recibidas correspondientes a una zona determinada, el procedimiento es distinto. En este caso deberemos definir la zona para la cual actuará como reenviador, utilizando el enlace "**Crear una nueva zona de reenvío**", bajo la lista de "Zonas DNS Existentes".

Deberemos especificar el tipo (de búsqueda directa o inversa) y el nombre de la zona, así como el servidor o servidores maestros a los que se realizará el reenvío.



2.3 Zonas de búsqueda directas e inversas

Para crear una nueva zona de búsqueda, bien sea directa o inversa, utilizamos la opción "**Crear una nueva zona maestra**".



En la pantalla siguiente escogemos entre otros el tipo de zona (para resolución directa o inversa), su nombre, el archivo en que se almacenará su configuración, y los parámetros del registro SOA.

Una vez creada una zona, aparece en la lista de "**Zonas DNS Existentes**". Pulsando en su enlace, accedemos a una pantalla dividida en tres partes.

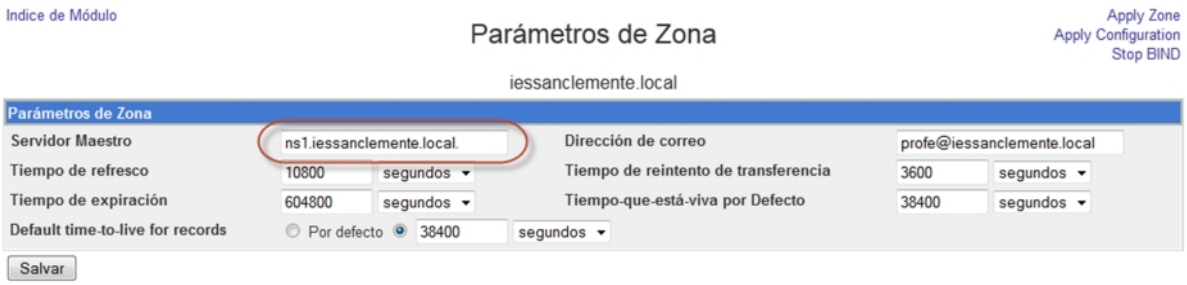
En la parte superior podemos gestionar (crear, modificar y borrar) los registros de la zona. Los enlaces de la parte intermedia nos permiten gestionar las características de la zona.

Por último en la parte inferior de la pantalla tenemos unos botones para realizar acciones concretas sobre la zona. Cada vez que hagamos modificaciones a una zona es impor-

tante comprobar si su configuración es correcta (botón "***Check Records***") antes de aplicar los cambios (enlace "***Apply Zone***", en la esquina superior derecha).

Es importante destacar que para que el "***Apply Zone***" funcione correctamente, previamente debemos haber configurado la utilidad RNDC para que pueda gestionar BIND. Para ello, en la pantalla de Webmin correspondiente a BIND, pulsamos en el enlace "***Setup RNDC***".

El primer cambio que deberemos realizar para la zona, es crear un registro de tipo A para el servidor maestro, y actualizar la información del registro NS correspondiente y el nombre del servidor maestro en los parámetros de la zona.



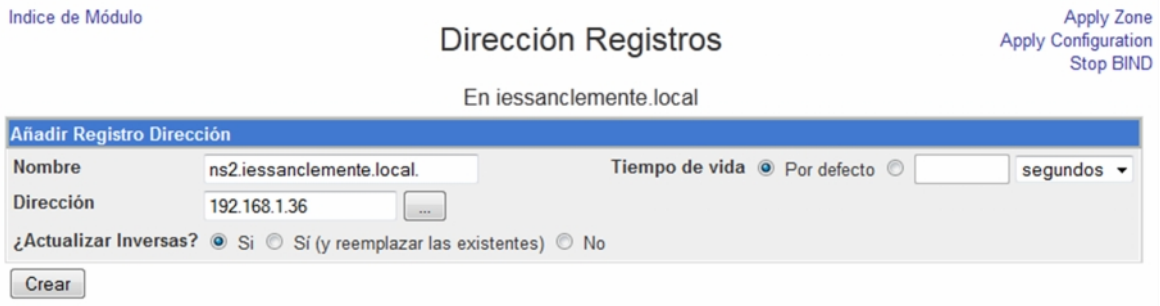
2.4 Servidor esclavo para una zona

Si queremos que nuestro servidor funcione como esclavo para una zona, en la lista de "Zonas DNS Existentes" tenemos el enlace "**Crear una nueva zona subordinada**". En este caso habrá que cubrir los siguientes datos:

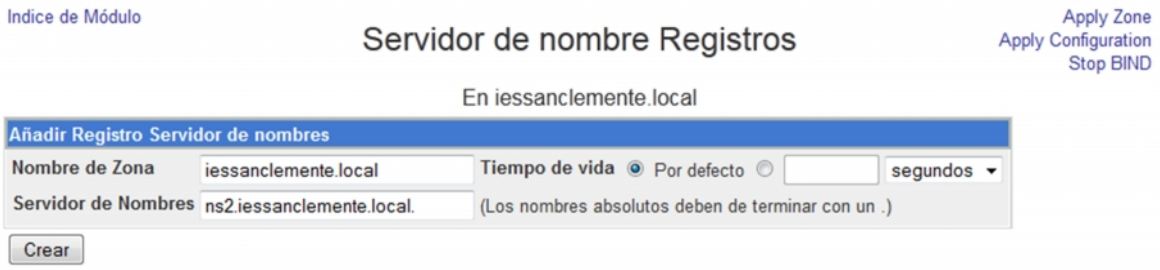
- Tipo de zona, de búsqueda directa o inversa.
- Nombre del dominio.
- Nombre del archivo en que se guardará la información de la zona.
- Lista de servidor(es) maestro(s) para esa zona.

En las propiedades de la zona subordinada nos informa de la última vez que se ha realizado una transferencia desde el maestro, y tenemos una opción para comprobar si ésta se realiza correctamente.

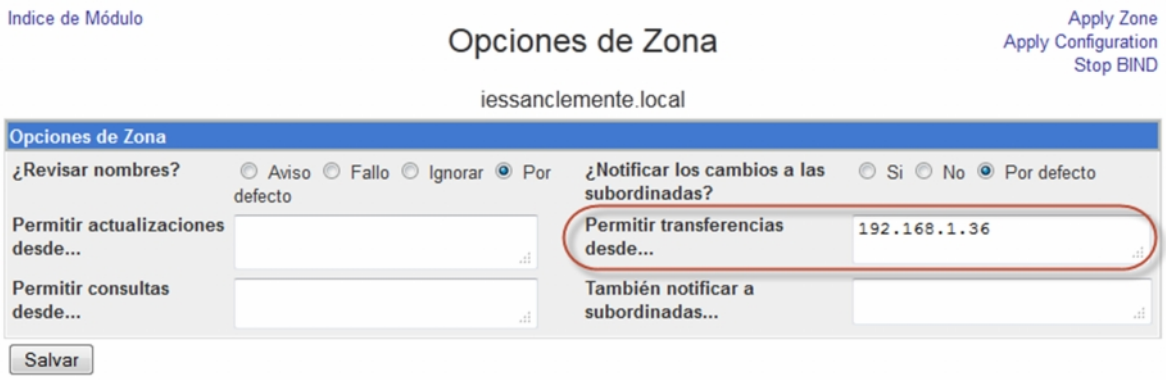
En el maestro correspondiente a esa zona, tenemos que incluir los registros correspondientes al nuevo servidor esclavo. El registro de tipo A.



Y el registro de tipo NS, para indicar que es un servidor de nombres.



Y por último ajustar las opciones de la zona para asegurarnos de que permitimos las transferencias desde ese equipo (por defecto se permiten desde cualquier equipo; es preferible denegarlas usando ***none*** en las opciones por defecto, y permitirlas solo para aquellas zonas en que sea necesario).



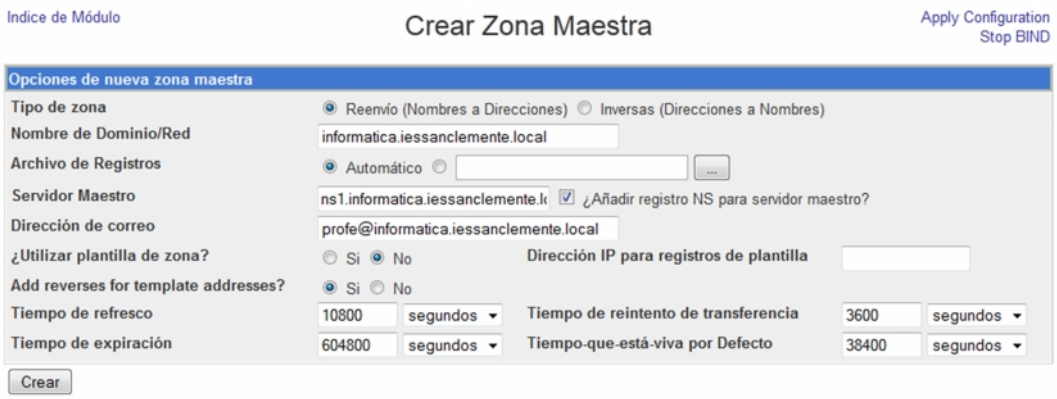
Si ahora probamos la transferencia de zona desde el esclavo, debería funcionar sin problemas.

En los servidores maestros de una zona, es habitual habilitar las notificaciones a los esclavos, marcando **Si** en la casilla "**¿Notificar los cambios a las subordinadas?**". Cuando se activa, el maestro avisará a los esclavos cada vez que se produzca un cambio en los registros de la zona, de forma que estos comprueben si deben actualizar su información sin esperar a que se cumpla el plazo marcado en el registro SOA.

2.5 Delegación de subdominios

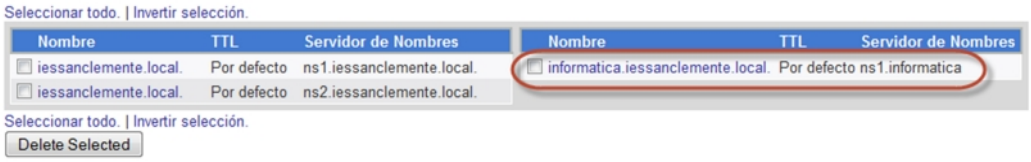
Para crear una delegación de una sub-zona o subdominio, debemos seguir los siguientes pasos.

- Crear la nueva zona como maestra en el servidor DNS en que se va a delegar la administración. Por ejemplo, si queremos delegar el subdominio **informática.iessanclemente.local**:



- En esa zona maestra correspondiente al subdominio, crearemos tantos registros de dirección como queramos (al menos uno para el servidor de nombres maestro).

- Ya tenemos la zona maestra correspondiente al subdominio preparada. Ahora falta indicar en el servidor maestro de la zona **iessanclemente.local**, que vamos a delegar el subdominio **informatica** (creando un registro de tipo **NS**).



- Y por última indicar también (mediante un registro de tipo **A**) cuál será la dirección de ese servidor de nombres correspondiente al subdominio.

Es importante indicar que en la delegación de subdominios, el maestro del dominio padre (en nuestro ejemplo **iessanclemente.local**) debe tener acceso a Internet o, en su defecto, se deberá añadir una zona en el servidor DNS para el dominio de primer nivel (**local**).

3. Configuración de BIND mediante ficheros

3.1 Ficheros de configuración de BIND

Los principales archivos que se incluyen en la instalación de BIND, se localizan en **/etc/bind** y son:

- **named.conf**, que junto con los ficheros que incluye normalmente (**named.conf.options**, **named.conf.local** y **named.conf.default-zones**) contiene la configuración del servicio y la lista de ficheros de zona que se usan.
- **db.root**, lista de servidores raíz.
- **db.127**, archivo de zona para resolución inversa local IPv4.
- **db.local**, archivo de zona para resolución directa local IPv4 e IPv6.
- **db.empty**, archivo de zona vacía.

3.2 Configuración básica de BIND

La configuración de BIND en named.conf y los ficheros que éste incluye, contiene tres tipos de entradas:

- **comentarios**. Pueden ser de una línea (comenzando por // o por #), o de más de una (comenzando por /* y terminando en */).
- **sentencias**. Definen el comportamiento del servidor.
- **cláusulas**. Sirven para agrupar conjuntos de sentencias.

Algunas de las **cláusulas** que pueden figurar en la configuración de BIND son:

- **acl**. Define elementos relativos al control de acceso, como una lista de direcciones IP.
- **controls**. Indica desde dónde se permite la administración remota con rndc.
- **include**. Permite incluir en el fichero de configuración el contenido de otros ficheros externos.
- **key**. Define claves secretas utilizadas en operaciones de control, actualizaciones dinámicas o transferencias de zona.
- **options**. Define un grupo de sentencias que controlan el comportamiento global del servidor y el de todas las zonas (salvo que incluyan su propia sentencia equivalente en su propia configuración de zona).
- **zone**. Contiene sentencias que definen la configuración de una zona específica.

Por ejemplo, **named.conf** puede tener el siguiente contenido:

```
include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";
acl IPs_permitidas {192.168.1.0/24;};
```

```
controls {
        inet 127.0.0.1 port 953 allow { 127.0.0.1; } keys { rndc-key; };
        };
key rndc-key {
        algorithm hmac-md5;
        secret "ju9YXtw3BDsig5fxkFIIrQ==";
        };
```

El fichero **named.conf.options** suele contener la cláusula **options** de configuración global.

```
options {
        directory "/var/cache/bind";
        dnssec-validation auto;
        auth-nxdomain no;
        listen-on-v6 { any; };
        allow-query { IPs_permitidas; };
        forwarders { 8.8.8.8; };
        forward only;
};
```

Y la definición de las zonas se reparte entre los ficheros **named.conf.default-zones**, que contiene las zonas incluidas por defecto, y **named.conf.local**, donde se añaden las zonas que se han configurado en el servidor. Por ejemplo:

```
zone "iessanclemente.local" {
        type master;
        file "/var/lib/bind/iessanclemente.local.hosts";
        };
```

3.3 Servidor DNS reenviador (o proxy)

Para configurar BIND como reenviador, debemos añadir dos sentencias a la cláusula **options** que vimos antes:

```
options {
        …
        forwarders { 8.8.8.8; };
        forward only;
};
```

La primera indica a qué servidor se reenviarán las consultas. La segunda indica que no se realice resolución recursiva (solamente reenvío).

3.4 Zonas de búsqueda directas e inversas

Para crear una zona de búsqueda, directa o inversa, es necesario:

- Definir la zona en **named.conf** (preferiblemente en **named.conf.local**) mediante una cláusula **zone**.
- Crear el fichero de la zona con todos los tipos de registro necesarios.

Un fichero para la zona **iessanclemente.local** puede ser, por ejemplo:

```
$ttl 38400
$origin iessanclemente.local.
iessanclemente.local.      IN      SOA    ns1.iessanclemente.local.
profe.iessanclemente.local. (
                           1348098453
                           10800
                           3600
                           604800
                           38400 )
iessanclemente.local.      IN      NS     ns1.iessanclemente.local.
iessanclemente.local.      IN      NS     ns2.iessanclemente.local.
ns1              IN        A       192.168.1.34
ns2              IN        A       192.168.1.36
www              IN        A       192.168.1.38
correo           IN        CNAME   www
```

3.5 Herramientas

Existen muchas herramientas que permiten comprobar la configuración y el correcto funcionamiento de un servidor DNS. Algunas son específicas de BIND, o de sistemas operativos Linux; otras las podemos encontrar también en distribuciones Windows. Muchas de las disponibles para Windows se incluyen con la instalación de BIND para este sistema operativo.

Vamos a ver el funcionamiento de las más utilizadas.

3.5.1 nslookup

Es una utilidad en línea de comandos que permite enviar consultas a los servidores DNS. Se puede ejecutar de forma interactiva (para hacer más de una consulta consecutiva) o no interactiva (se realiza una consulta a un servidor DNS y se muestra el resultado obtenido). Existe una versión para plataformas Windows.

Para ejecutar nslookup de **forma no interactiva** debemos indicarle cuál debe ser el contenido de la consulta. Por ejemplo:

```
usuario@ubuntu-profe:~$ nslookup www.iessanclemente.net
Server:         208.67.222.123
Address:        208.67.222.123#53

Non-authoritative answer:
Name:   www.iessanclemente.net
Address: 82.98.132.209
```

Por defecto la consulta se realizará a los servidores DNS que se han configurado en el sistema. Si queremos hacerla a otro servidor, se puede indicar como segundo parámetro.

```
usuario@ubuntu-profe:~$ nslookup www.iessanclemente.net 192.168.1.34
Server:         192.168.1.34
Address:        192.168.1.34#53

Non-authoritative answer:
Name:   www.iessanclemente.net
Address: 82.98.132.209
```

Si no indicamos ningún parámetro, entramos en **modo interactivo** y las consultas se realizarán por defecto a los servidores configurados en el sistema operativo.

```
usuario@ubuntu-profe:~$ nslookup
> www.iessanclemente.net
Server:         208.67.222.123
Address:        208.67.222.123#53

Non-authoritative answer:
Name:   www.iessanclemente.net
Address: 82.98.132.209
>
```

Si quisiéramos hacer la consulta a un servidor distinto, podemos indicarlo en la llamada como segundo parámetro (para que no se confunda con una resolución no interactiva; el primer parámetro será un guión).

```
usuario@ubuntu-profe:~$ nslookup - 192.168.1.34
>
```

O especificarlo como segundo parámetro en cada consulta.

```
usuario@ubuntu-profe:~$ nslookup
> www.iessanclemente.net 192.168.1.34
Server:         192.168.1.34
Address:        192.168.1.34#53

Non-authoritative answer:
Name:   www.iessanclemente.net
Address: 82.98.132.209
>
```

En este modo, además de poder resolver directamente el nombre de un host determinado como en el ejemplo anterior, también tenemos disponibles algunos **comandos** para realizar diversas acciones. Principalmente:

- **server**. Permite cambiar el servidor por defecto al que se efectuarán las consultas posteriores.

  ```
  > server 192.168.1.34
  ```

- **set**. Permite cambiar parámetros relacionados con las consultas posteriores. Por ejemplo:

```
> set all                       # muestra los parámetros y su valor actual
> set type=…                    # establece el tipo de registro a consultar
> set querytype=…               # igual que 'set type=…'
```

Por ejemplo:

```
usuario@ubuntu-profe:~$ nslookup
> server 192.168.1.34
Default server: 192.168.1.34
Address: 192.168.1.34#53
> set type=NS
> set all
Default server: 192.168.1.34
Address: 192.168.1.34#53

Set options:
  novc                  nodebug         nod2
  search                recurse
  timeout = 0           retry = 3       port = 53
  querytype = NS        class = IN
  srchlist = domain.name
> iessanclemente.local
Server:         192.168.1.34
Address:        192.168.1.34#53

iessanclemente.local    nameserver = ns1.iessanclemente.local.
iessanclemente.local    nameserver = ns2.iessanclemente.local.
>
```

Sea cual sea la forma en que iniciemos la utilidad **nslookup**, interactiva o no interactiva, podemos añadir en la llamada (como primer parámetro, y precedidas de un guión) una serie de **opciones** que afectan a su ejecución, como:

```
port=…
type=…   (o querytype)
retry=…
```

3.5.2 dig

Es otra herramienta en línea de comandos similar a nslookup. No tiene un modo interactivo, pero puede utilizarse para ejecutar una serie de peticiones almacenadas en un fichero. La forma más común de utilizar dig es:

```
dig @servidor nombre tipo
```

Significa que se realice al servidor que se indica, una consulta sobre un registro del tipo y nombre especificado.

Todos los parámetros son opcionales. Por defecto dig realizará una búsqueda de registros tipo A a los servidores configurados en el sistema. Si no se utiliza ningún parámetro en la llamada (ni siquiera el nombre del recurso), dig devuelve la lista de los servidores de nombres raíz. Por ejemplo:

```
usuario@ubuntu-profe:~$ dig @192.168.1.34 iessanclemente.local ns

; <<>> DiG 9.8.1-P1 <<>> @192.168.1.34 iessanclemente.local ns
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 10795
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 2

;; QUESTION SECTION:
;iessanclemente.local.          IN      NS

;; ANSWER SECTION:
iessanclemente.local.   38400   IN      NS      ns1.iessanclemente.local.
iessanclemente.local.   38400   IN      NS      ns2.iessanclemente.local.

;; ADDITIONAL SECTION:
ns1.iessanclemente.local. 38400 IN      A       192.168.1.34
ns2.iessanclemente.local. 38400 IN      A       192.168.1.36

;; Query time: 8 msec
;; SERVER: 192.168.1.34#53(192.168.1.34)
;; WHEN: Tue Sep 4 11:32:30 2012
;; MSG SIZE  rcvd: 106
```

Una de las características más útiles de dig es la posibilidad de probar una trasnferencia de zona:

```
dig @IP_maestro nombre_zona axfr
```

Al contrario que nslookup, dig es capaz de trabajar con DNSSEC.

3.5.3 rndc

Es una aplicación en línea de comandos que permite administrar el servidor BIND (el demonio **named**), tanto desde la propia máquina en que se ejecuta como desde otra remota.

Para asegurar el control remoto de BIND, se utiliza una clave secreta compartida, que debe figurar exactamente igual en los ficheros de configuración de ambos (o almacenada en el fichero **/etc/rndc.key** o **/etc/bind/rndc.key**), tanto en **named.conf** como en **rndc.conf**.

```
key "rndc-key" {
        algorithm hmac-md5;
        secret "ju9YXtw3BDsig5fxkFIIrQ==";
```

```
};
```

Además, en **named.conf** debe haber sección **controls** que indique desde que equipos se permite el control remoto del servidor.

```
controls {
        inet 127.0.0.1 port 953 allow { 127.0.0.1; } keys { rndc-key; };
        };
```

Entre los **comandos** disponibles con rndc, figuran:

- **querylog**. Registra las solicitudes recibidas por el servidor.
- **reload**. Recarga los archivos de zona. Si solo se ha cambiado un fichero de zona, se puede añadir el nombre de la zona después del comando reload para recargar solo esa.
- **status**. Muestra la información de estado del servidor.
- **stop**. Detiene al servidor.

Existen también opciones, como **–s**, que permite realizar las acciones sobre otro servidor BIND distinto al especificado en rndc.conf.

```
usuario@ubuntu-profe:~$ sudo rndc -s 192.168.1.34 reload
server reload successful
```

3.5.4 nsupdate

Permite realizar de forma manual actualizaciones sobre zonas en las que estén permitidas (normales o aseguradas con DNSSEC).

3.5.5 named-checkconf

Es una aplicación que se incluye con BIND. Comprueba la sintaxis de **named.conf**, el fichero de configuración de BIND. Es útil para prevenir posibles errores, comprobando los cambios realizados antes de aplicarlos.

3.5.6 named-checkzone

Similar a la anterior, pero comprueba la sintaxis de una zona concreta. Se le deben indicar como parámetros el nombre de la zona y la ubicación del fichero en que se almacena.

```
named-checkzone sanclemente.local /var/lib/bind/sanclemente.local.hosts
```

3.5.7 dnstracer

Es útil para rastrear las solicitudes que realiza un servidor DNS a la hora de resolver una solicitud. No se incluye en la instalación base de Ubuntu, pero se puede instalar desde los repositorios. Para ejecutarla simplemente se tiene que indicar la consulta a realizar y opcionalmente el servidor DNS con la **opción –s**.

```
usuario@ubuntu-profe:/etc/bind$ dnstracer -s 192.168.1.34 info.sancl.local
```

```
Tracing to info.sancl.local[a] via 192.168.1.34, maximum of 3 retries
192.168.1.34 (192.168.1.34)
  \___ ns1.info.sancl.local [info.sancl.local] (192.168.1.36)
```

3.6 Servidor esclavo para una zona

Para definir un nuevo servidor esclavo para una zona existente, debemos:

- Añadir una sentencia **allow-transfer** a la cláusula **zone** correspondiente a esa zona, en la que se indique la máquina que será esclava para la zona.

```
zone "iessanclemente.local" {
        type master;
        file "/var/lib/bind/iessanclemente.local.hosts";
        allow-transfer { 192.168.1.36; };
        };
```

- Crear esa misma zona en el nuevo servidor, indicando en su sentencia correspondiente que es de tipo esclavo y cuál es el servidor maestro.

```
zone "iessanclemente.local" {
        type slave;
        masters { 192.168.1.34; };
        file "/var/lib/bind/iessanclemente.local.hosts";
        };
```

3.7 Delegación de subdominios

También en este caso son necesarios dos pasos.

- En el servidor que va a delegar la administración del subdominio, indicar el nombre del subdominio y la máquina en que se delega, añadiendo las siguientes líneas.

```
informatica       IN       NS      ns1.informatica
$origin informatica.iessanclemente.local.
ns1               IN       A       192.168.1.36
```

- En el servidor en que se va a delegar, crear como zona maestra el nuevo subdominio. Por ejemplo:

```
$ttl 38400
$origin informatica.iessanclemente.local.
informatica.iessanclemente.local.  IN  SOA  ns1. informati-
ca.iessanclemente.local. profe.informatica.iessanclemente.local. (
                        1348178039
                        10800
                        3600
                        604800
                        38400 )
informatica.iessanclemente.local.      IN   NS  ns1.iessanclemente.local.
```

```
ns1             IN      A       192.168.1.36
www             IN      A       192.168.1.36
departamento    IN      CNAME   www
```

3.8 DDNS en Ubuntu

Vamos a ver cómo preparar un servidor DHCP, de tal forma cada vez que se produzca una concesión de dirección IP a un cliente, las modificaciones de los registros A correspondientes a esa máquina se realicen automáticamente en el servidor DNS (BIND) de esa zona (DNS Dinámico – DDNS).

Un requisito fundamental es que el servidor DNS sea autoritativo para la zona (o zonas) en la cual se van a realizar las actualizaciones. Un servidor es autoritativo para una zona si contiene una copia completa del fichero de esa zona (por ejemplo, el servidor primario).

Asegurar las actualizaciones en el servidor DNS

El primer paso es crear una clave criptográfica que sirva para asegurar que únicamente el servidor DHCP va a ser capaz de realizar modificaciones en los archivos de zona del servidor DNS.

La utilidad `ddns-confgen` nos facilita la creación de la clave. Por ejemplo, si hacemos:

```
ddns-confgen -a hmac-md5 -z iessanclemente.local
```

Estamos indicando que queremos una clave que utilice el algoritmo MD5 para la zona `iessanclemente.local`. La ejecución del comando anterior produce el siguiente resultado en pantalla:

```
# To activate this key, place the following in named.conf, and
# in a separate keyfile on the system or systems from which nsupdate
# will be run:
key "ddns-key.iessanclemente.local" {
 algorithm hmac-md5;
 secret "H2fICbsqCAmEG4uU90xPKQ==";
};

# Then, in the "zone" definition statement for "iessanclemente.local",
# place an "update-policy" statement like this one, adjusted as
# needed for your preferred permissions:
update-policy {
   grant ddns-key.iessanclemente.local zonesub ANY;
};

# After the keyfile has been placed, the following command will
# execute nsupdate using this key:
nsupdate -k <keyfile>
```

Es decir, que para activar la actualización dinámica del fichero de la zona deberemos:

- Copiar en un fichero (por ejemplo, `/etc/bind/ddns-key.iessanclemente.local`) la claúsula `key` anterior. Es importante asegurarse de que la lectura de este fichero está restringida al usuario `root` (`chmod og-rw ddns-key.iessanclemente.local`).
- Añadir en `named.conf` la misma claúsula `key` anterior,.
- Añadir en la definición de la zona (fichero `named.conf.local`, dentro de `zone iessanclemente.local`) la claúsula `update-policy`. Fíjate que el fichero al que hace referencia esa claúsula debe ser el mismo que creaste en el primer paso.
- Por último, hay que ejecutar el comando `nsupdate` haciendo referencia al fichero de clave que has creado o (más sencillo) reiniciar el servidor BIND.

Comprobar las actualizaciones dinámicas

Existe una herramienta que permite comprobar que el servidor DNS permite las actualizaciones dinámicas (a los poseedores de la clave). Se trata de `nsupdate`. Es una utilidad interactiva. En la llamada deberemos indicar el fichero de clave que va a utilizar:

```
nsupdate –k /etc/bind/ddns-key.iessanclemente.local
```

Y a continuación, podemos utilizar el comando `update add` para añadir información a la zona. Por ejemplo.

```
update add prueba.iessanclemente.local 3600 A 192.168.56.100
send
quit
```

De la misma forma, con el comando `update delete` podemos eliminar registros.

```
update delete prueba.iessanclemente.local
send
quit
```

Configurar el servidor DHCP para que realice las actualizaciones

Para activar en el servidor DHCP la posibilidad de que realice actualizaciones dinámicas cada vez que otorga una concesión, debemos añadir la siguiente línea al fichero `dhcpd.conf` (o modificar la existente).

```
ddns-update-style interim;
```

Debemos añadir en ese mismo fichero la clave generada anteriormente.

```
key "ddns-key.iessanclemente.local" {
 algorithm hmac-md5;
 secret "H2fICbsqCAmEG4uU90xPKQ==";
};
```

Y por último añadir también en el mismo fichero un claúsula para cada zona que queramos actualizar de forma dinámica, donde debemos indicar la dirección IP del servidor en el que se van a producir las actualizaciones (127.0.0.1 si el servidor DNS y DHCP comparten máquina) y qué clave utilizar (puede haber más de una si actualizamos distintas zonas que se almacenan en distintos servidores DNS).

```
zone iessanclemente.local. {
    primary 127.0.0.1;
    key ddns-key.iessanclemente.local;
}
```