

DEAW -DW2

Reto #2

Aketza Egusquiza
Iker Navarro
Aritz Romo

DNS
20/10/2020

Contenido

Introducción:.....	2
Esquema de red:.....	3
Servidor DNS	4
Configuración	4
DNS.....	5
RESOLUCIÓN DIRECTA:.....	5
RESOLUCIÓN INVERSA:	6
COMPROBACIÓN DE LA CONFIGURACIÓN:.....	7
COMPROBACIONES:.....	8

Introducción:

En este reto se busca configurar un router como dns con el fin de acceder a diferentes servidores por nombre.

Para ello, utilizaremos el sistema de red del anterior reto, añadiendo al router de la red dicha funcionalidad.

Para configurar las diferentes máquinas, hemos utilizado el software de virtualización VirtualBox.

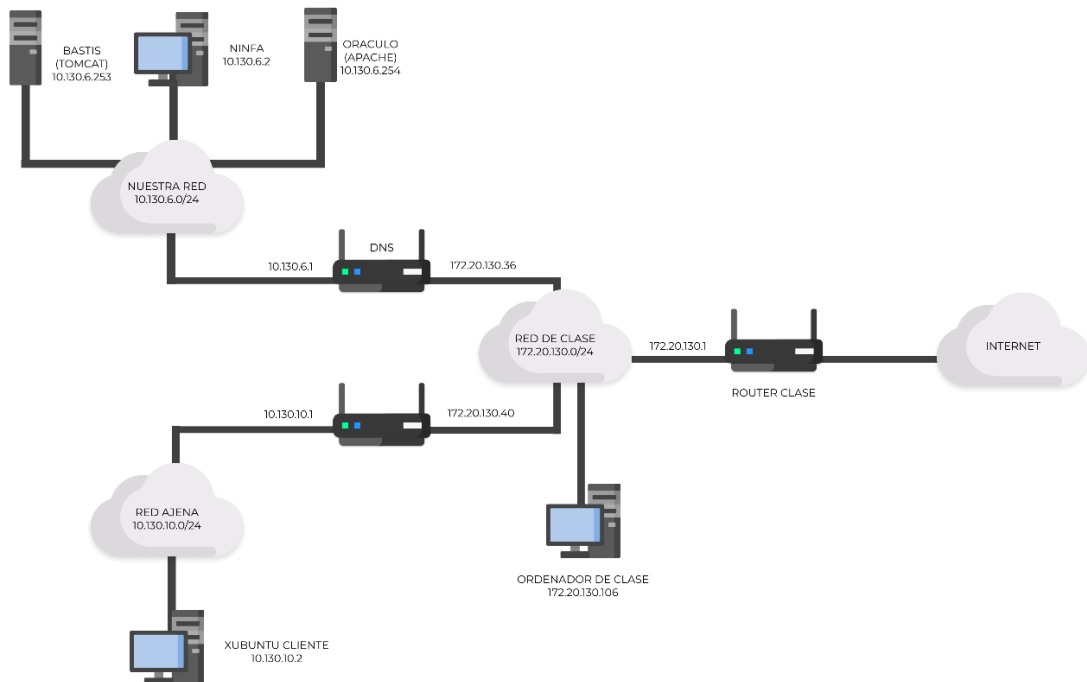
Sistemas operativos utilizados:

Router: Debian 9

Tomcat: Debian 10

Apache: Debian 10

Esquema de red:



Servidor DNS

Para el servidor de DNS hemos utilizado una máquina con Debian 9, la misma que utilizamos en la anterior práctica que hacía las veces de router, y el paquete BIND, el cual hemos tenido que instalar.

BIND es el servidor DNS más utilizado en sistemas Unix y es el que también utilizaremos nosotros. Para la instalación de BIND simplemente nos identificamos como root y lo instalamos a través de apt-get. Seguiremos logueados como root para continuar configurando e instalando los servicios.

```
su -  
apt-get install bind9
```

Configuración

```
/etc/bind/named.conf
```

En este archivo se guarda la configuración de las zonas generadas por defecto en el momento de la instalación y no ha sido necesario modificarlo.

```
/etc/bind/named.conf.local
```

Este archivo lo editamos para definir la zona y la zona inversa del servidor. Para editar el archivo, escribimos en consola el comando:

```
/etc/bind/named.conf.local  
zone "atenea.olimpo.god" {  
    type master;  
    file "/etc/bind/db.atenea.olimpo.god";  
};  
  
zone "6.130.10.in-addr.arpa" {  
    type master;  
    file "/etc/bind/db.10.130.6";  
};
```

Este archivo lo editamos y definimos los servidores DNS de Internet (forwarders) que se encargarán de resolver los nombres de dominio que nuestro servidor local es incapaz de resolver. Esto es necesario, ya que si no las máquinas de nuestra red no serían capaces de salir a Internet.

Para los servidores DNS de Internet, hemos elegido el de Cloudflare (1.1.1.1) y el de Google (8.8.8.8).

/etc/bind/named.conf.options

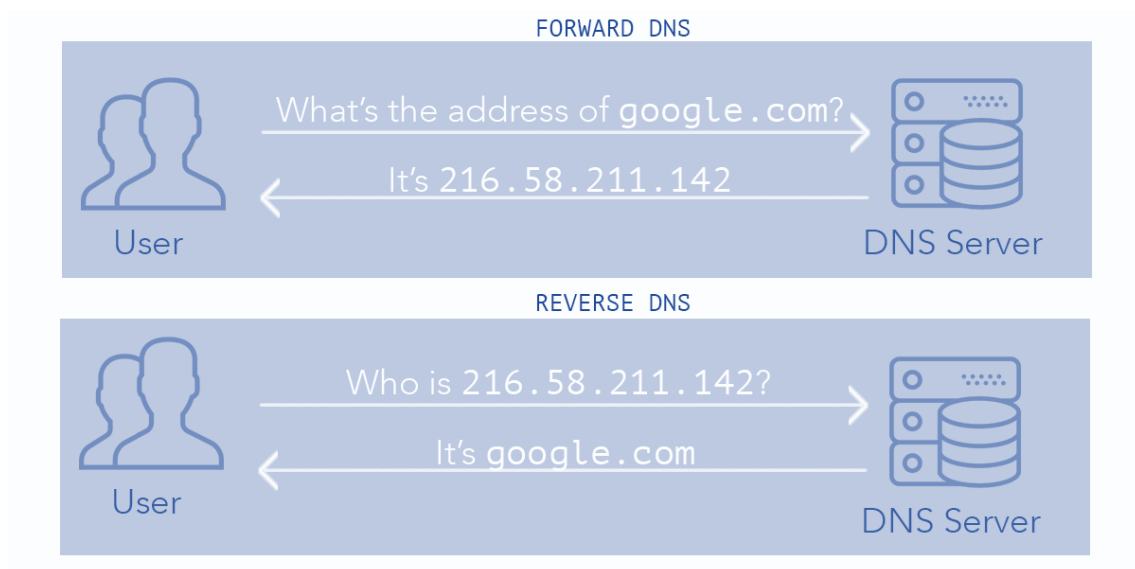
```
options {  
  
    forwarders {  
        1.1.1.1;  
        8.8.8.8;  
    };  
  
    dnssec-validation auto;  
  
    auth-nxdomain no; # conform to RFC1035  
    listen-on-v6 { any; };  
};
```

DNS

Con esto ya tendremos configuradas las zonas y los servidores que redirigirán las peticiones en caso que nuestro DNS desconozca algún nombre.

Pero ahora tendremos que configurar todos los nombres de las máquinas que resolverá nuestro DNS; para ello tendremos que crear y editar 2 archivos. Uno para la zona de resolución directa y otro para la zona de resolución inversa.

En resumen:



RESOLUCIÓN DIRECTA:

/etc/bind/db.atenea.olimpo.god

```
$TTL 604800
```

```
atenea.olimpo.god. IN SOA atenea.olimpo.god. router.atenea.olimpo.god. (
```

```
2006081401;
28800;
3600;
604800;
38400)
atenea.olimpo.god. IN NS router.atenea.olimpo.god.
router IN A 10.130.6.1
ninfa IN A 10.130.6.2
bastis IN A 10.130.6.253
oraculo IN A 10.130.6.254
dns IN CNAME router
```

El primer apartado de este archivo se encarga de configurar ciertas variables para la resolución, como el TTL (Time To Live), la tasa de refrescado y otros parámetros similares.

Tras esto, especificaremos mediante una entrada "IN NS" quien será el Name Server de nuestra zona (atenea.olimpo.god), que en nuestro caso será nuestro router (se pueden especificar múltiples servidores de nombres para una única zona).

Hecho esto, tendremos todas las relaciones entre nombre y dirección IP para todas las máquinas que nuestro servidor DNS será capaz de resolver mediante entradas "IN A".

Si en un futuro quisiéramos añadir una nueva máquina a nuestra red simplemente tendríamos que añadir una nueva entrada para dicho servidor/equipo.

Finalmente tenemos configurado un alias para que nuestro equipo encargado con las labores de routing y de resolución de nombres sea accesible mediante dos nombres: ROUTER y DNS.

RESOLUCIÓN INVERSA:

```
/etc/bind/db.10.130.6
```

```
$TTL 604800
@ IN SOA atenea.olimpo.god. router.atenea.olimpo.god. (
2006081401;
28800;
3600;
```

```
604800;  
38400)  
@ IN NS router.atenea.olimpo.god.  
1 IN PTR router.atenea.olimpo.god.  
2 IN PTR ninfa.atenea.olimpo.god.  
253 IN PTR bastis.atenea.olimpo.god.  
254 IN PTR oraculo.atenea.olimpo.god.
```

Al igual que en el archivo de la resolución directa, el primer apartado del archivo simplemente contiene parámetros de configuración para la resolución de nombres.

Tras estas configuraciones iniciales pasaremos a configurar al igual que en la configuración de la zona directa la relación entre direcciones IP y nombres, pero en este caso de manera inversa.

Primero, especificaremos el Name Server de nuestra red (en este caso nuestra máquina router) y hecho esto, configuraremos el resto de máquinas de nuestra red, especificando primero la IP dentro de la red en la zona (10.130.6) seguido del nombre de la máquina (todos ellos con entradas IN PTR).

En este archivo "@" hace referencia a la ubicación actual.

COMPROBACIÓN DE LA CONFIGURACIÓN:

Bind ofrece algunos comandos para realizar comprobaciones de sintaxis en los archivos de configuración que hemos creado y editado.

Primero, para la comprobación del archivo named.conf tendremos que utilizar el siguiente comando:

```
# named-checkconf
```

Si este comando no devuelve nada significará que la sintaxis de dicho archivo es correcta.

Para comprobar los archivos de zona tendremos que utilizar los siguientes comandos:

```
# named-checkzone 10.130.6 /etc/bind/db.10.130.6  
# named-checkzone atenea.olimpo.god /etc/bind/db.atenea.olimpo.god
```


Para que este comando funcione tendremos que especificar primero la zona seguido de la ruta del archivo donde hemos realizado la configuración de dicha zona.

Si la sintaxis del archivo es correcta, este comando devolverá la siguiente línea en la terminal.

```
zone nombre.de.la.zona/IN: loaded serial 1 OK
```

Habiendo hecho todos los pasos anteriores nuestro DNS ya estará operativo, pero como siempre podremos controlar el estado de este servicio mediante los siguientes comandos:

```
# systemctl start bind9  
# systemctl stop bind9  
# systemctl restart bind9  
# systemctl status bind9
```

COMPROBACIONES:

Para comprobar que nuestro servidor DNS efectivamente funciona realizaremos diversas pruebas desde diversos clientes de nuestra red. Si nuestro servidor DNS está funcionando correctamente deberá reconocer todas las máquinas de nuestra red interna que hayamos configurado tanto por nombre como por dirección IP, además de funcionar con direcciones y nombres externos a nuestra red, para asegurarnos que los forwarders configurados funcionan.

Para realizar estas pruebas utilizaremos el comando nslookup, un programa utilizado para saber si el DNS está resolviendo correctamente nombres e IPs.

Es un programa que suele venir instalado por defecto en múltiples distribuciones, pero en el caso de no estar instalado simplemente tendríamos que hacerlo utilizando el siguiente comando:

Distribuciones basadas en debian:

```
# apt-get install dnsutils
```

Distribuciones basadas en RedHat

```
# yum install bind-utils
```

Antes de realizar estas pruebas tendremos que configurar nuestro servidor DNS como predeterminado en todos nuestros equipos. Este paso podemos realizarlo mediante la interfaz gráfica de configuración de red.

Nombre de la conexión: Conexión cableada 1

General | Cableada | Seguridad 802.1x | DCB | Proxy | Ajustes de IPv4 | Ajustes de IPv6

Método: Manual

Dirección

Dirección	Máscara de red	Puerta de enlace
10.130.6.2	24	10.130.6.1

Servidores DNS: 10.130.6.1

Dominios de búsqueda:

ID del cliente DHCP:

☐ Requiere dirección IPv4 para que esta conexión se complete

Rutas...

Cancelar | Guardar

Haremos esto para todas nuestras máquinas y podremos proceder con las pruebas:

Primero comprobaremos que nuestro servidor DNS resuelve máquinas internas por nombre (bastis.atenea.olimpo.god) y servidores externos (www.google.es) sin mayor problema.

```
Terminal - dw2@dw2-VirtualBox: ~
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda

dw2@dw2-VirtualBox:~$ nslookup www.google.es
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   www.google.es
Address: 216.58.201.163
Name:   www.google.es
Address: 2a00:1450:4003:802::2003

dw2@dw2-VirtualBox:~$ nslookup bastis.atenea.olimpo.god
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   bastis.atenea.olimpo.god
Address: 10.130.6.253

dw2@dw2-VirtualBox:~$
```

Probaremos ahora la resolución inversa para 8.8.8.8 (dns propietario de google) y para 10.130.6.253 (máquina de nuestra red interna).

```
Terminal - dw2@dw2-VirtualBox: ~
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda

dw2@dw2-VirtualBox:~$ nslookup 8.8.8.8
8.8.8.8.in-addr.arpa    name = dns.google.

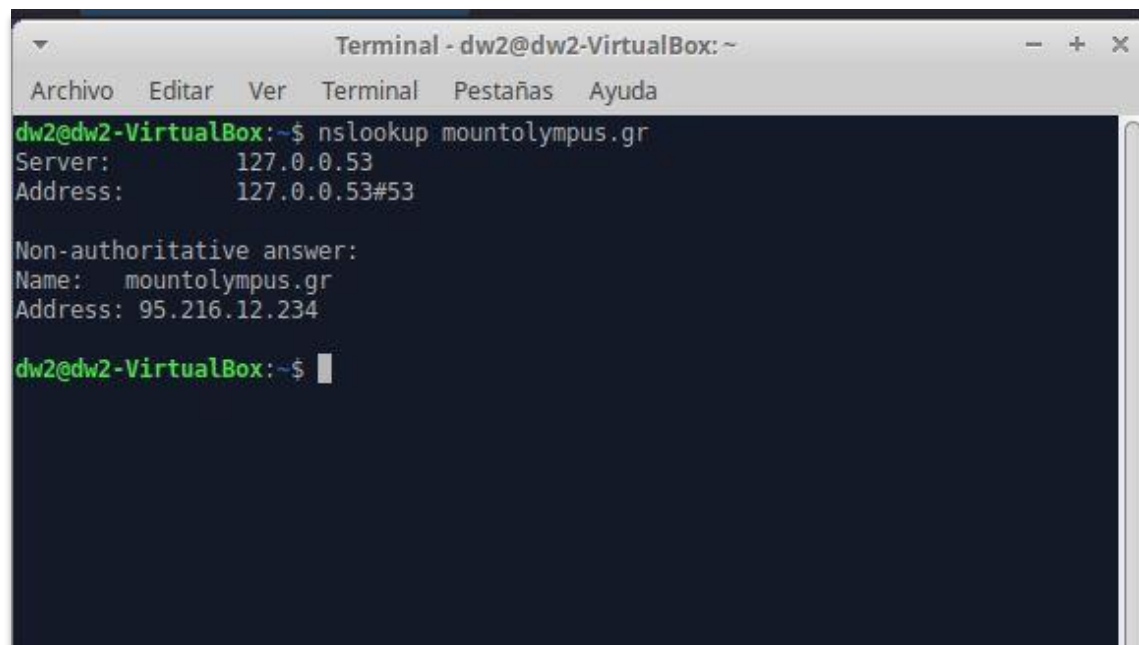
Authoritative answers can be found from:

dw2@dw2-VirtualBox:~$ nslookup 10.130.6.253
253.6.130.10.in-addr.arpa    name = bastis.atenea.olimpo.god.

Authoritative answers can be found from:

dw2@dw2-VirtualBox:~$
```

Tal y como se indica en los requerimientos del reto realizaremos también la resolución directa para "mountolympus.gr".

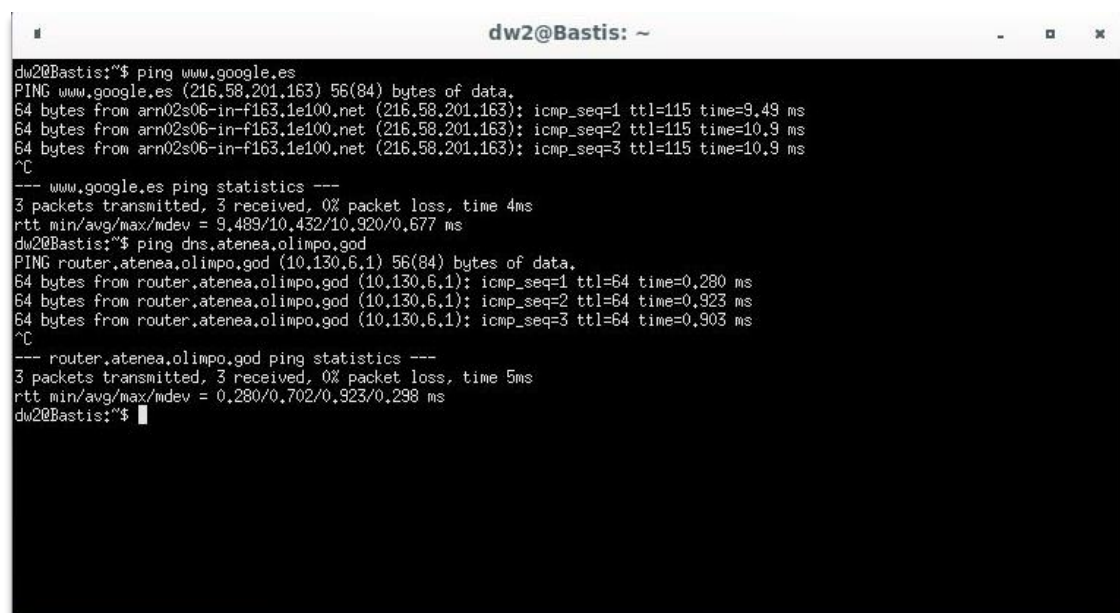
A terminal window titled "Terminal - dw2@dw2-VirtualBox: ~" with a menu bar containing "Archivo", "Editar", "Ver", "Terminal", "Pestañas", and "Ayuda". The terminal shows the command "nslookup mountolympus.gr" and its output: "Server: 127.0.0.53", "Address: 127.0.0.53#53", "Non-authoritative answer:", "Name: mountolympus.gr", and "Address: 95.216.12.234". The prompt "dw2@dw2-VirtualBox:~\$" is visible at the bottom.

```
Terminal - dw2@dw2-VirtualBox: ~
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
dw2@dw2-VirtualBox:~$ nslookup mountolympus.gr
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   mountolympus.gr
Address: 95.216.12.234

dw2@dw2-VirtualBox:~$
```

Finalmente podemos probar como última comprobación que podemos utilizar los nombres configurados con otros métodos a nslookup, por ejemplo realizando ping a las máquinas por nombre.

A terminal window titled "dw2@Bastis: ~" showing two ping commands. The first is "ping www.google.es" which shows three successful pings with times around 9-10 ms and a summary: "3 packets transmitted, 3 received, 0% packet loss, time 4ms". The second is "ping dns.atenea.olimpo.god" which also shows three successful pings with times around 0.28 ms and a summary: "3 packets transmitted, 3 received, 0% packet loss, time 5ms".

```
dw2@Bastis:~$ ping www.google.es
PING www.google.es (216.58.201.163) 56(84) bytes of data:
64 bytes from ann02s06-in-f163.1e100.net (216.58.201.163): icmp_seq=1 ttl=115 time=9.49 ms
64 bytes from ann02s06-in-f163.1e100.net (216.58.201.163): icmp_seq=2 ttl=115 time=10.9 ms
64 bytes from ann02s06-in-f163.1e100.net (216.58.201.163): icmp_seq=3 ttl=115 time=10.9 ms
^C
--- www.google.es ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 4ms
rtt min/avg/max/mdev = 9.489/10.432/10.920/0.677 ms
dw2@Bastis:~$ ping dns.atenea.olimpo.god
PING router.atenea.olimpo.god (10.130.6.1) 56(84) bytes of data:
64 bytes from router.atenea.olimpo.god (10.130.6.1): icmp_seq=1 ttl=64 time=0.280 ms
64 bytes from router.atenea.olimpo.god (10.130.6.1): icmp_seq=2 ttl=64 time=0.923 ms
64 bytes from router.atenea.olimpo.god (10.130.6.1): icmp_seq=3 ttl=64 time=0.903 ms
^C
--- router.atenea.olimpo.god ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 5ms
rtt min/avg/max/mdev = 0.280/0.702/0.923/0.298 ms
dw2@Bastis:~$
```