15. DNS

15. 1.- Introducción

DNS (**D**omain **N**ame **S**ystem) es el procotolo de resolución de nombres de redes TCP/IP. Es un sistema globalmente distribuido, escalable y jerárquico que ofrece una base de datos dinámica que asocia direcciones IP con información de diverso tipo, facilitando el acceso a los servicios de Internet por nombre en lugar de por IP.

Se utiliza el puerto 53 tanto TCP como UDP. Desde un cliente se hace una consulta a un servidor DNS por UDP y el servidor responde también por UDP. Las conexiones TCP se utilizan cuando el tamaño de la respuesta excede de 52 bytes, p.e. transferencia de zona (los datos del maestro se replican a los esclavos). En las comunicaciones DNS, se utiliza casi siempre UDP ya que requiere menos recursos de proceso y de red pero al no haber control sobre los datos transmitidos, la respuesta siempre es recibida y esto supone múltiples vulnerabilidades.

Existen varias implementaciones de DNS, las más importantes:

- *Bind*: con el uso más extendido, fue desarrollada en los 80 por la universidad de Berkeley y actualmente mantenida por el ISC.
- *Unbound*: desarrollado en 2007, ha comenzado a reemplazar a Bind como implementación DNS por defecto instalada en los proyectos Open Source ya que es más ligero, moderno y seguro que Bind.
- *DNSmasq*: para configurar forwarders DNS ligeros proporcionando además de servicio DNS, servicio DHCP y TFTP a redes pequeñas.

15.1.1.- Conceptos básicos

- ***FQDN*** (**F**ully **Q**ualified **D**omain **N**ame): nombre que especifica la posición absoluta de un nodo en el árbol jerárquico de DNS. Lleva un **.** al final. Tiene una longitud de hasta 255 caracteres (alfanuméricos y el carácter guión), sin distinción entre mayúsculas y minúsculas, donde la primera ha de ser una letra (se permiten caracteres unicode desde 2004) y con una limitación de 63 caracteres para cada etiqueta de dentro del dominio.
- ***Zona:*** división del espacio de nombres de un dominio donde un servidor de nombres en particular es responsable de mantener la información, se dice que es autoritativo para esa zona. Cada zona almacena información de nombres sobre todo un dominio o sólo una parte. Existen dos tipos de zonas: de resolución directa o reenvío (de nombre de dominio a IP) y de resolución inversa (de IP a nombre de dominio).

- ***Registro de recurso*** (RR) es una entrada en una zona DNS con información específica. Un registro de recurso tiene: nombre (del dominio al que pertenece), tipo (tipo de dato que almacena), un TTL (**T**ime **t**o **L**ive, segundos que deben pasar para que el registro caduque), una clase (IN), longitud en bytes del dato que almacena y el dato que almacena.

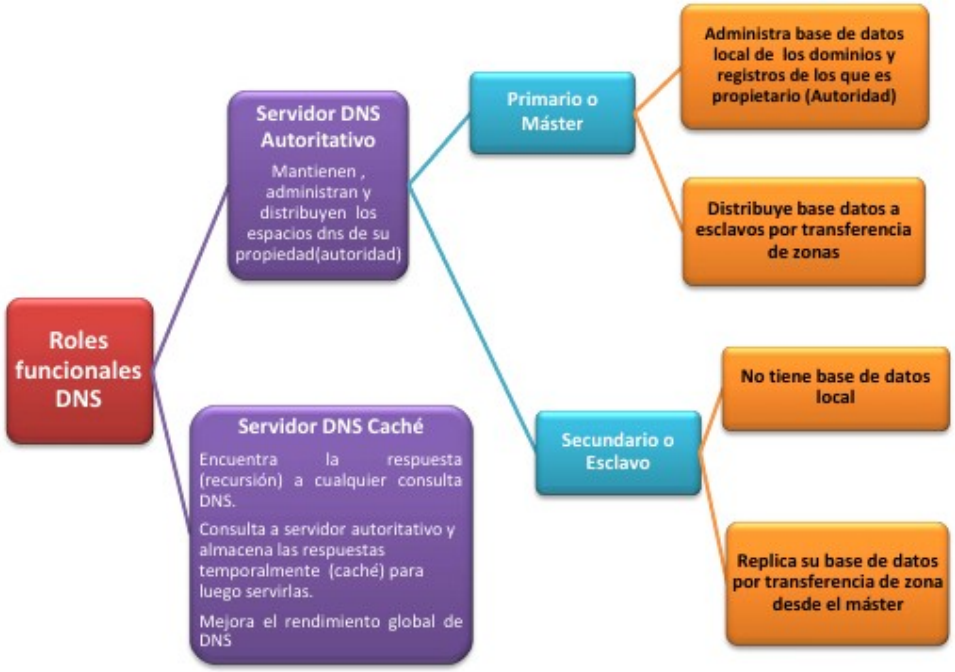
  Tipos de registros DNS más comunes:

  - *A* (IPv4 **A**ddress): Resuelve nombres de dominio a direcciones IPv4. Tanto en servidores de zona como en los otros tipos, habrá uno o más registros A.
  - *AAAA* (IPv6 **A**ddress): Resuelve nombres de dominio a direcciones IPv6. Tanto en servidores de zona como en los otros tipos, habrá uno o más registros AAAA.
  - CNAME (**C**annonical **NAME**): crea nombres adicionales (alias) de recurso. Tanto en servidores de zona como en los otros tipos, habrá cero o más registros CNAME.
  - *MX* (**M**ail **E**xchange): lista de servidores de correo para el dominio, con balanceo de carga y prioridad. No debe apuntar a un CNAME, debe apuntar a un A o AAAA. En un servidor DNS de zona, habrá al menos un registro MX.
  - *PTR* (**P**oin**t**er **r**ecord): resuelve una dirección IPv4 a nombre de dominio, esto se llama resolución inversa. Tanto en servidores de zona como en los otros tipos, habrá uno o más registros PTR, uno por dirección IP.
  - *NS* (**N**ame **S**erver): lista de servidores de nombres que almacenan la información de la zona o los servidores de nombre de las zonas delegadas. No debe apuntar a un CNAME, debe apuntar a un A o AAAA. En un servidor DNS de zona, habrá un registro NS por servidor autoritativo.
  - *SOA* – (**S**tart **O**f **A**uthority): especifica cómo trabaja la zona, sólo un registro de este tipo en el servidor DNS primario de la zona.
  - *SRV* (**S**ervice): información de servicios disponibles en el dominio. En un servidor DNS de zona, habrá 0 o más registros SRV.
  - *TXT* (**T**ext): Registros de texto. En un servidor DNS de zona, habrá 0 o más registros TXT.
- ***DNSSEC:*** ampliación del protocolo DNS. El protocolo UDP es propenso a suplantación de identidad, con DNSSEC los recursos se validan con certificados digitales antes de ser cacheados evitando una de las vulnerabilidades más comunes de DNS llamada *envenenamiento de caché*.

15.1.2.- Componentes principales DNS:

DNS tiene tres componentes principales:

- *Espacio de nombres de dominio*: estructura jerárquica de árbol donde cada nodo contiene de cero a N registros con información del dominio. El nivel más alto de la jerarquía es el dominio *root* **.** que contiene el espacio de nombres DNS completo. Cada nodo o dominio representa una parte del árbol siendo TLD (top-leven domain) el nivel que está justo debajo del root, sólo tienen un componente, p.e., .com, .edu, .org, ...
- *Servidores de nombres*: servidores encargados de mantener y/o proporcionar la información del espacio de nombres a los clientes. Son de dos tipos: autoritativos o de caché. Para cada zona deben existir al menos tres servidores autoritativos (RFC-2182) para permitir que la información esté siempre disponible y sea confiable; normalmente, un maestro y dos esclavos. Los servidores de caché obtienen respuestas a consultas DNS consultando a los servidores autoritativos y las almacenan en caché, mejorando el rendimiento al reducir el tráfico DNS en Internet y reduciendo la carga sobre los servidores autoritativos, especialmente sobre los root servers.



Roles servidores DNS

- *Resolver o programas cliente:* generan las consultas y obtienen la información solicitada para ofrecerla al usuario.

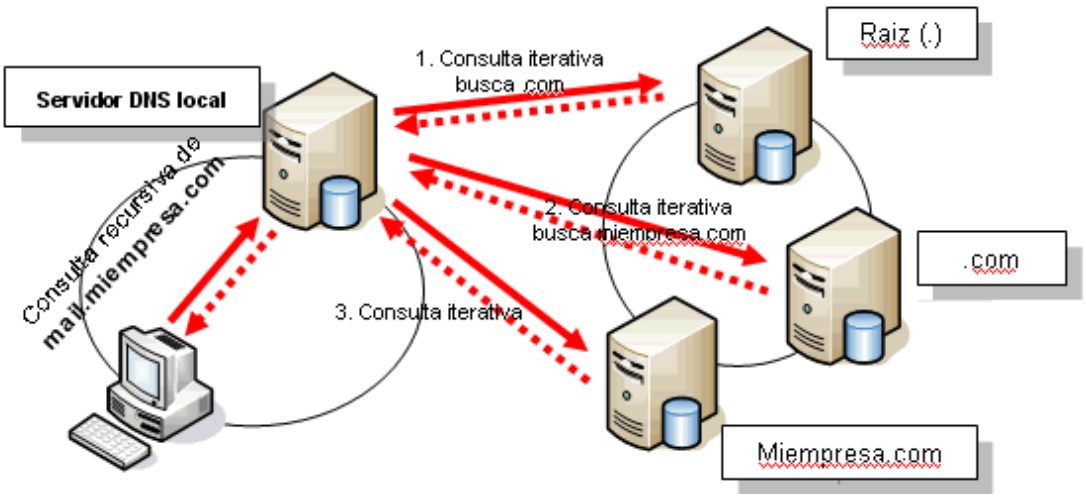
Dada la complejidad de la configuración de un servidor DNS autoritativo (maestro o esclavo), sólo se va a ver cómo configurar un servidor DNS de tipo caché.,

15.2.- Consultas DNS

15.2.1.- Tipos de consultas

Los tipos de consultas a los que responden los servidores de DNS son:

- *Recursivas*: el cliente pregunta y el servidor asume toda la carga de proporcionar respuesta siendo el servidor el que hace varias consultas iterativas.
- *Iterativas o no recursivas*: el cliente pregunta y el servidor responde con la mejor respuesta en la zona local o la caché. Si no puede dar respuesta, le devuelve a qué servidor DNS debe hacer la consulta, así el cliente hará múltiples consultas hasta que llega al servidor que tiene la respuesta.



Tipos de consultas DNS

15.2.2.- Funcionamiento de una consulta DNS

En el sistema cliente, un programa cliente lanza una consulta DNS, el sistema cliente busca en su archivo */etc/resolv.conf* a qué servidores DNS tiene que hacer la consulta, en el orden en el que aparecen.

El servidor recibe la consulta, si está autorizado para la zona busca la información en su archivo de zona local y activa en la cabecera de la respuesta el flag *aa* (**A**uthoritative **A**nswer), respondiendo al cliente con la información. Si no está autorizado para la zona, busca si tiene la información cacheada y la devuelve al cliente sin activar el flag *aa*. Si no tiene la información cacheada, entonces comienza un proceso recursivo donde pregunta al servidor raíz por el servidor autorizado para la zona y se lanza la consulta a ese servidor y recibirá la respuesta o una dirección de a quien

enviar la consulta. Así hasta obtener la respuesta, devolverla al cliente y cachearla. También se cachea a qué servidor pregunto por una zona.

15.2.3.- Comandos para lanzar consultas DNS desde un sistema cliente

- *host <nombre_dominio> <servidor_dns>, host <ip> <servidor_dns>*: dado un nombre de dominio o una ip, hace una consulta al servidor DNS dado. Si no se proporciona *<servidor_dns>*, se hace la consulta a los servidores configurados en */etc/resolv.conf* (no se consulta el archivo */etc/hosts*). Si además se pone *-t <tipo_registro>*, devuelve sólo el registro pedido.
- *dig @<servidor_dns> <nombre_dominio> <tipo_registro>*: hace una consulta al servidor DNS dado (con hostname o IP) del nombre de dominio y registro y devuelve los servidores consultados. Si no se le pasa *@<servidor_dns>*, toma los configurados en */etc/resolv.conf*. Si no se proporciona el tipo de registro, se asume el tipo A.
- *whois <nombre_dominio>, whois <ip>*: consulta a los servidores *whois* (servidores que almacenan toda la información de un conjunto de dominios o IPs) por los datos de un nombre de dominio o IP en Internet: quien lo registró y sus datos de contacto, fecha de creación y expiración, etc.

15.3.- Servidor DNS de caching

Un servidor DNS de caching almacena consultas DNS en caché local con TTL y las borra cuando expira dicho TTL. Se suele configurar uno por red local para aumentar el rendimiento y reducir tráfico en Internet. Aumentan su efectividad con el tiempo, según va creciendo la caché.

15.3.1.-Pasos para configurar un sistema servidor DNS de caching con Unbound

Utilizando Unbound seguiremos los pasos:

1.- Instalar el paquete *unbound.*

2.- Activar y arrancar el servicio *unbound*.

3.- Modificar el archivo de configuración de Unbound */etc/unbound/unbound.conf* con:

- interfaces de red por donde escucha Unbound. Dentro del bloque *server*, se ponen tantas líneas *interface: <IP>* como interfaces de red tenga el sistema configuradas y queramos que se escuche por ellas. Por defecto, sólo estará escuchando en *localhost*. Poniendo *0.0.0.0* hacemos que escuche en todas las interfaces de red que tenga el sistema.
- acceso de los clientes. Por defecto, se rechazan consultas de todos los clientes salvo *localhost*. Para modificar este comportamiento, dentro del bloque *server*, se añaden tantas líneas

access-control: <ip_cliente> <permiso> como clientes se deseen, donde:

<*ip_cliente*>: dirección IP o dirección de red.

<*permiso*>*:* los valores pueden ser: *allow*, *deny*, *refuse*, para permitir, denegar o rechazar (envía mensaje de rechazo al cliente.)

- servidor de DNS donde se van a reenviar las consultas. Se crea un bloque *forward-zone* con las líneas *name: "."* y o *forward-addr: <IP_server>* o *forward-host <hostname_server>*, quedando así:

```
forward-zone:                          forward-zone:
    name: "."              ó               name: "."
    forward-addr: <ip>                     forward-host: <host>
```

- los dominios internos (confiables), que salte la validación DNSSEC, en el bloque *server*, añadimos la línea *domain-insecure: <dominio>*.

4.- Chequear la configuración con *unbound-check* y reiniciar el servicio *unbound*.

5.- Abrir el firewall para el servicio *dns.*

6.- Modificar en la configuración del sistema, si se desea, que el servidor DNS sea el mismo.

15.3.2.- **Operaciones a realizar con la caché de Unbound:**

- Volcar caché a un archivo: *unbound-control dump-cache > <archivo_cache>*
- Cargar la caché de Unbound desde un archivo: *unbound-control load-cache < <archivo_cache>*
- Purgar la caché de recursos anticuados: *unblound-control flush <nombre_dominio>* o *unbound-control flush-zone <dominio>*.

15.4.- Caso práctico

En la instalación de **central** con el archivo de kickstart se configura un servidor DNS con la implementación *bind* para que esté disponible tanto para *localhost* como en la IP de **central** *10.11.1.254*. Es un servidor de forwarding que reenvía las peticiones a la máquina física ubicada en la IP *192.168.122.1*. La zona es *miempresa.com* y que tiene registrados, incluso para resolución inversa, los sistemas **serverX** (0..9) y **central**.

En **server1** se va a configurar un servidor de DNS de caching con la implementación de DNS *unbound* que estará escuchando en todos los interfaces de **server1**. Las peticiones no cacheadas, se reenviarán a **central**. Se permitirá a los sistemas de la subred de **server1**, realizar consultas DNS, siendo el dominio *miempresa.com* considerado seguro.

RESOLUCIÓN

- En **server1**, la interfaz *eth0* está configurada por DHCP, y es el servidor DHCP el que proporciona el servidor DNS al sistema, escribiéndolo Network Manager en el archivo */etc/resolv.conf*.

```
[root@server1 ~]# less /etc/resolv.conf
# Generated by NetworkManager
search miempresa.com
nameserver 10.11.1.254
```

Ejecutamos el comando *host* para ver cuál es el nombre de dominio de esa IP:

```
[root@server1 ~]# host 10.11.1.254
254.1.11.10.in-addr.arpa domain name pointer central.miempresa.com.
```

El servidor DNS que está utilizando **server1** es **central**.

- Configuramos en **server1** un DNS de caching:

- Instalamos el paquete *unbound.*

```
[root@server1 ~]# yum install -y unbound
```

- Activamos y arrancamos el servicio *unbound*.

```
[root@server1 ~]# systemctl start unbound && systemctl enable unbound
```

- Modificamos el archivo de configuración de Unbound */etc/unbound/unbound.conf* con:

  - los interfaces de red por donde escucha Unbound*:* todos, en el bloque *server*, añadimos la línea:

    interface: 0.0.0.0

- el acceso de los clientes: sistemas de la subred *10.11.1.0/24*, en el bloque *server*, añadimos la línea:

  access-control: 10.11.1.0/24 allow

  - servidor de DNS donde se van a reenviar las consultas: **central** que está en *10.11.1.254*:

```
forward-zone:
    name: "."
    forward-addr: 10.11.1.254
```

  - el dominio interno *miempresa.com* es confiable, en el bloque *server*, añadimos:

    domain-insecure: miempresa.com

- Chequeamos la configuración y reiniciamos el servicio:

```
[root@server1 ~]# unbound-checkconf
unbound-checkconf: no errors in /etc/unbound/unbound.conf
[root@server1 ~]# systemctl restart unbound
```

- Abrimos el firewall para el servicio *dns:*

```
[root@server1 ~]# firewall-cmd --permanent --add-service=dns --zone=internal;
[root@server1 ~]# firewall-cmd --reload
```

- Modificamos en la configuración del sistema, si se desea, que el servidor DNS sea el mismo.

```
[root@server1 ~]#  nmcli con show
NAME      UUID                                  TYPE           DEVICE
estatica  fb48045f-6611-4b7f-95d6-7f07101140e6  802-3-ethernet  eth0
eth0      ffbec545-27b6-4c8a-87e5-3ad59fd4b301  802-3-ethernet  --
[root@server1 ~]#  nmcli con mod estatica ipv4.dns 127.0.0.1; reboot
```

Volcamos la caché de Unbound, al no haber utilizado **server1** de servidor de DNS nunca, estará vacía:

```
[root@server1 ~]# unbound-control dump_cache
START_RRSET_CACHE
END_RRSET_CACHE
START_MSG_CACHE
END_MSG_CACHE
EOF
[root@server1 ~]# dig miempresa.com

; <<>> DiG 9.9.4-RedHat-9.9.4-37.el7 <<>> miempresa.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 64838
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
; EDNS  version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;miempresa.com.                         IN      A

;; AUTHORITY SECTION:
miempresa.com.                  3511    IN      SOA     central.miempresa.com.
root.miempresa.com.     20150402 86400 172800 2419200 3600

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Mon Aug 28 14:00:17 CEST 2017
;; MSG SIZE  rcvd: 91
```

Ahora al volcar el contenido de la caché, tiene datos, luego **server1** funciona como servidor DNS.