

# Práctica 1.2. Instalar Servidor DNS

DESPLIEGUE DE APLICACIONES WEB

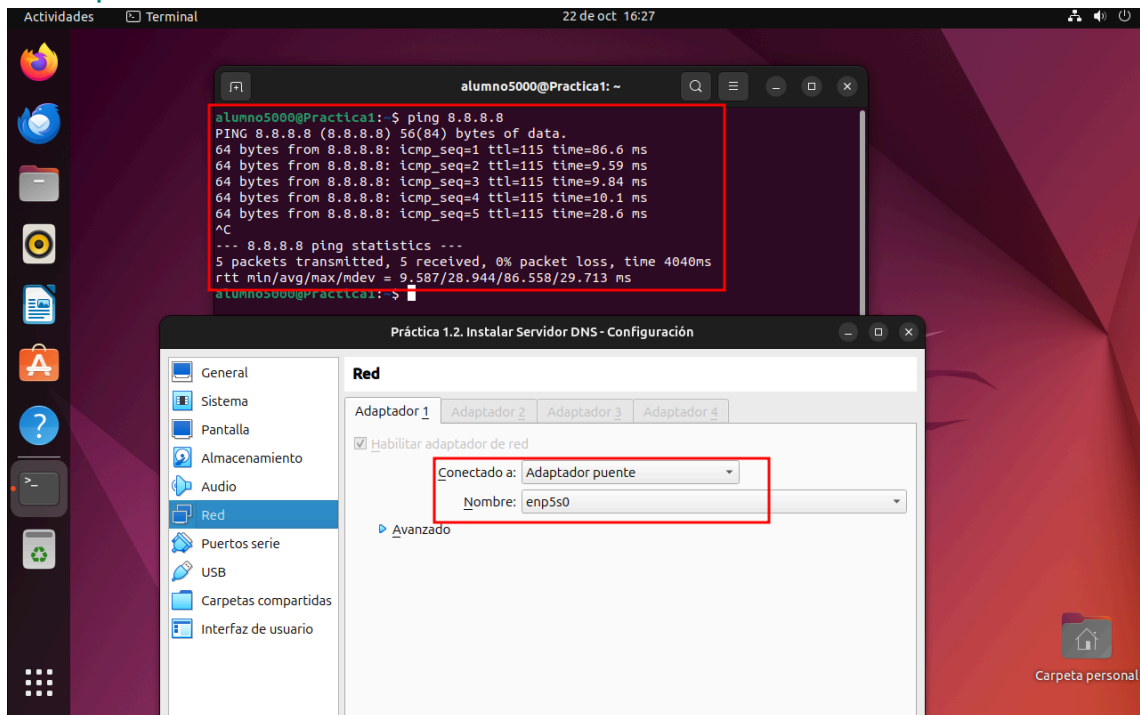
---

Alejandro Leo Carretero  
2º DAW-A

<b>Práctica 1.2. Instalar Servidor DNS</b>	<b>3</b>
Pasos previos	3
Instalación servidor DNS	4
Configuración Servidor Principal	6
Configuración Servidor Primario, Secundario y Comprobaciones	8
Comandos	12
1. ping	12
2. host	13
3. nslookup	14
4. dig	15
Enlace video Explicando la Práctica	16

## Práctica 1.2. Instalar Servidor DNS

### Pasos previos



Ponemos adaptador puente y comprobamos que tenemos internet haciendo un ping a google

```
root@Practica1:~# sudo apt update && upgrade
Obj:1 http://gb.archive.ubuntu.com/ubuntu jammy InRelease
Obj:2 http://gb.archive.ubuntu.com/ubuntu jammy-updates InRelease
Obj:3 http://gb.archive.ubuntu.com/ubuntu jammy-backports InRelease
Obj:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Levendo la información de estado... Hecho
```

También actualizamos los paquetes

## Instalación servidor DNS

```

root@Practica1: ~
root@Practica1:~# sudo apt update && sudo apt install bind9 -y
Obj:1 http://gb.archive.ubuntu.com/ubuntu jammy InRelease
Obj:2 http://security.ubuntu.com/ubuntu jammy-security InRelease
Obj:3 http://gb.archive.ubuntu.com/ubuntu jammy-updates InRelease
Obj:4 http://gb.archive.ubuntu.com/ubuntu jammy-backports InRelease
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se pueden actualizar 23 paquetes. Ejecute «apt list --upgradable» para verlos.
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  bind9-utils
Paquetes sugeridos:
  bind-doc resolvconf
Se instalarán los siguientes paquetes NUEVOS:
  bind9 bind9-utils
0 actualizados, 2 nuevos se instalarán, 0 para eliminar y 23 no actualizados.
Se necesita descargar 422 kB de archivos.
Se utilizarán 1.666 kB de espacio de disco adicional después de esta operación.
Des:1 http://gb.archive.ubuntu.com/ubuntu jammy-updates/main amd64 bind9-utils amd64
1:9.18.28-0ubuntu0.22.04.1 [161 kB]
Des:2 http://gb.archive.ubuntu.com/ubuntu jammy-updates/main amd64 bind9 amd64 1:9.18

```

Vamos a instalar bind9

```

root@Practica1: ~
root@Practica1:~# service bind9 status
● named.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/named.service; enabled; vendor preset: enable)
   Active: active (running) since Tue 2024-10-22 16:51:18 CEST; 4min 39s ago
     Docs: man:named(8)
  Process: 24317 ExecStart=/usr/sbin/named $OPTIONS (code=exited, status=0/SUCCESS)
 Main PID: 24318 (named)
    Tasks: 14 (limit: 4607)
   Memory: 8.2M
      CPU: 72ms
   CGroup: /system.slice/named.service
           └─24318 /usr/sbin/named -u bind

oct 22 16:51:18 Practica1 named[24318]: network unreachable resolving './DNSKEY/IN':>
oct 22 16:51:18 Practica1 named[24318]: network unreachable resolving './NS/IN': 200>
oct 22 16:51:18 Practica1 named[24318]: network unreachable resolving './DNSKEY/IN':>
oct 22 16:51:18 Practica1 named[24318]: network unreachable resolving './NS/IN': 200>
oct 22 16:51:18 Practica1 named[24318]: network unreachable resolving './DNSKEY/IN':>
oct 22 16:51:18 Practica1 named[24318]: network unreachable resolving './NS/IN': 200>
oct 22 16:51:18 Practica1 named[24318]: network unreachable resolving './DNSKEY/IN':>
oct 22 16:51:18 Practica1 named[24318]: network unreachable resolving './DNSKEY/IN':>
oct 22 16:51:18 Practica1 named[24318]: managed-keys-zone: Initializing automatic tr>
oct 22 16:51:18 Practica1 named[24318]: resolver priming query complete: success
lines 1-22/22 (END)

```

Vemos el estado del servicio bind9

```

root@Practica1:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:87:92:f5 brd ff:ff:ff:ff:ff:ff
    inet 172.30.0.92/24 brd 172.30.0.255 scope global dynamic noprefixroute enp0s3
        valid_lft 19502sec preferred_lft 19502sec
    inet6 fe80::4db5:7fe:4f8:8937/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
root@Practica1:~#

```

Vemos la dirección ip

```

root@Practica1:~# nslookup google.com 172.30.0.92
Server:          172.30.0.92
Address:         172.30.0.92#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.201.78
Name:   google.com

```

```

^Croot@Practica1:~# dig @172.30.0.92 google.com

; <<>> DiG 9.18.28-0ubuntu0.22.04.1-Ubuntu <<>> @172.30.0.92 google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 63313
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:;; udp: 1232
; COOKIE: c3d0a664da308ddf010000006717becfa9022de722a821c1 (good)
;; QUESTION SECTION:
;google.com.                IN      A

;; ANSWER SECTION:
google.com.                 197     IN      A      142.250.201.78

;; Query time: 0 msec
;; SERVER: 172.30.0.92#53(172.30.0.92) (UDP)
;; WHEN: Tue Oct 22 17:03:43 CEST 2024
;; MSG SIZE rcvd: 83

```

Y comprobamos con estos dos comandos que tenemos conexión a google y que el servidor esta bien

## Configuración Servidor Principal

```
GNU nano 6.2 /etc/netplan/01-cfg-static.yaml
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: false
      addresses:
        - 172.30.0.92/24
      routes:
        - to: default
          via: 172.30.0.92
      nameservers:
        addresses: [172.30.0.92]
```

Editamos este fichero para tener una ip estática

```
root@Practica1:~# sudo netplan apply
```

Y aplicamos los cambios

```
usuario@usuario-VirtualBox:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:5e:d5:94 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.134/24 brd 192.168.1.255 scope global noprefixroute enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe5e:d594/64 scope link
        valid_lft forever preferred_lft forever
```

Ya tenemos nuestra ip estatica

```
usuario@usuario-VirtualBox:~$ sudo ufw allow bind9
Reglas actualizadas
Reglas actualizadas (v6)
usuario@usuario-VirtualBox:~$
```

Añadimos una regla de firewall a nuestro servidor bind



```
GNU nano 6.2 named.conf.options *
options {
    directory "/var/cache/bind";

    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk. See http://www.kb.cert.org/vuls/id/800113

    // If your ISP provided one or more IP addresses for stable
    // nameservers, you probably want to use them as forwarders.
    // Uncomment the following block, and insert the addresses replacing
    // the all-0's placeholder.

    // forwarders {
    //     8.8.8.8;
    //     8.8.4.4;
    // };

    //=====  

    // If BIND logs error messages about the root key being expired,  

    // you will need to update your keys. See https://www.isc.org/bind-keys  

    //=====
    dnssec-validation auto;

    listen-on { any; };
    listen-on-v6 { any; };
    allow-query{
        any;
    };
};
```

Configuramos el fichero named.conf.options con los parámetros que salen en la imagen.

```
Microsoft Windows [Versión 10.0.19045.5011]
(c) Microsoft Corporation. Todos los derechos reservados.

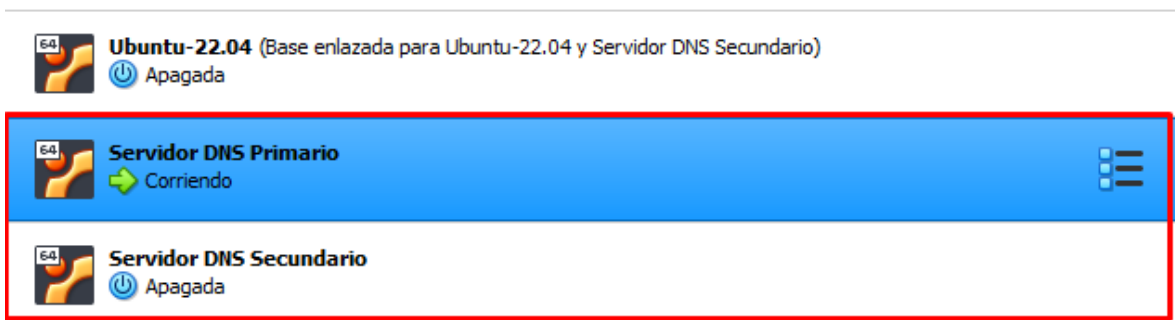
C:\Users\aleja>nslookup google.com 192.168.1.134
Servidor: UnKnown
Address: 192.168.1.134

Respuesta no autoritativa:
Nombre: google.com
Addresses: 2a00:1450:4003:80c::200e
          142.250.184.174

C:\Users\aleja>
```

Comprobamos desde nuestra máquina real windows funciona el servicio

## Configuración Servidor Primario, Secundario y Comprobaciones



Clonamos 2 veces el servidor y arrancamos el primario

```
GNU nano 6.2 /etc/bind/named.conf.local *
//
// Do any local configuration here
//
// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "asir-sri.com" {
    type master;
    file "/etc/bind/db.asir-sri.com";
    allow-transfer { 192.168.1.134; };
    also-notify { 192.168.1.134; };
};
```

Editamos este fichero con los parámetros que se ven en la foto

```
usuario@usuario-VirtualBox:~$ sudo cp /etc/bind/db.local /etc/bind/db.asir-sri.com
usuario@usuario-VirtualBox:~$
```

Ahora para crear el archivo de la zona utilizamos la plantilla de "db.local" y la pegamos en la nuestra



```
GNU nano 6.2 /etc/bind/db.asir-sri.com *
;
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA      ns.asir-sri.com. usuario.asir-sri.com. (
        2024103000      ; Serial
        10hr           ; Refresh
        15m           ; Retry
        48h           ; Expire
        604800 )      ; Negative Cache TTL
;

@         IN      NS       ns.asir-sri.com.
@         IN      NS       ns2.asir-sri.com.
@         IN      A        192.168.1.134
ns        IN      A        192.168.1.135
ns2       IN      A        192.168.1.136
router    IN      A        192.168.1.1
www       IN      CNAME    router
```

Ahora editamos el siguiente fichero con la configuración que sale en la captura de pantalla

```
Símbolo del sistema
C:\Users\aleja>nslookup asir-sri.com 192.168.1.135
Servidor: UnKnown
Address: 192.168.1.135

Nombre: asir-sri.com
Address: 192.168.1.134

C:\Users\aleja>nslookup ns.asir-sri.com 192.168.1.135
Servidor: UnKnown
Address: 192.168.1.135

Nombre: ns.asir-sri.com
Address: 192.168.1.135

C:\Users\aleja>nslookup ns2.asir-sri.com 192.168.1.135
Servidor: UnKnown
Address: 192.168.1.135

Nombre: ns2.asir-sri.com
Address: 192.168.1.136

C:\Users\aleja>nslookup router.asir-sri.com 192.168.1.135
Servidor: UnKnown
Address: 192.168.1.135

Nombre: router.asir-sri.com
Address: 192.168.1.1

C:\Users\aleja>nslookup www.asir-sri.com 192.168.1.135
Servidor: UnKnown
Address: 192.168.1.135

Nombre: router.asir-sri.com
Address: 192.168.1.1
Aliases: www.asir-sri.com

C:\Users\aleja>
```

Hacemos todas estas comprobaciones desde nuestra máquina real y vemos que está todo correcto

```
GNU nano 6.2 /etc/bind/named.conf.local *
//
// Do any local configuration here
//
// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "asir-sri.com" {
    type slave;
    file "/etc/bind/db.asir-sri.com";
    masters { 192.168.1.135; };
};
```

Ahora abrimos el servidor secundario y ponemos la configuración de este fichero

```
C:\Users\aleja>nslookup asir-sri.com 192.168.1.136
Servidor: UnKnown
Address: 192.168.1.136

Nombre: asir-sri.com
Address: 192.168.1.134

C:\Users\aleja>
```

Si probamos la conexión con nuestra máquina real veremos que todo está bien

```
GNU nano 6.2 /etc/bind/named.conf.local *
//
// Do any local configuration here
//
// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "asir-sri.com" {
    type master;
    file "/etc/bind/db.asir-sri.com";
    allow-transfer { 192.168.1.136; };
    also-notify { 192.168.1.136; };
};

zone "1.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/db.1.168.192.in-addr.arpa";
};
```

Cerramos el servidor secundario y volvemos al primario donde vamos a editar este fichero de la forma en la que sale en la captura de pantalla para la configuración inversa

```
usuario@usuario-VirtualBox:~$ sudo cp /etc/bind/db.asir-sri.com /etc/bind/db.1.168.192.in-addr.arpa
```

Copiamos la plantilla del anterior fichero en este

```
GNU nano 6.2 /etc/bind/db.1.168.192.in-addr.arpa
;
; BIND data file for local loopback interface
;
$TTL 604800
@ IN SOA ns.asir-sri.com. usuario.asir-sri.com. (
    2024103000 ; Serial
    10h ; Refresh
    15m ; Retry
    48h ; Expire
    604800 ) ; Negative Cache TTL
;
@ IN NS ns.asir-sri.com.
@ IN NS ns2.asir-sri.com.
134 IN PTR asir-sri.com.
135 IN PTR ns.asir-sri.com.
136 IN PTR ns2.asir-sri.com.
```

Configuramos el fichero como sale en la imagen

```
C:\Users\aleja>nslookup 192.168.1.134 192.168.1.135
Servidor: ns.asir-sri.com
Address: 192.168.1.135

Nombre: asir-sri.com
Address: 192.168.1.134

C:\Users\aleja>nslookup 192.168.1.135 192.168.1.135
Servidor: ns.asir-sri.com
Address: 192.168.1.135

Nombre: ns.asir-sri.com
Address: 192.168.1.135

C:\Users\aleja>nslookup 192.168.1.136 192.168.1.135
Servidor: ns.asir-sri.com
Address: 192.168.1.135

Nombre: ns2.asir-sri.com
Address: 192.168.1.136

C:\Users\aleja>
```

Comprobamos que funciona las resoluciones inversa en la máquina real y como vemos en la imagen funcionan sin problema

## Comandos

### 1. ping

Definición: ping es una herramienta que verifica la conectividad entre el host local y un destino (IP o nombre de dominio) enviando paquetes de eco ICMP. Mide el tiempo que tarda en recibir una respuesta.

Ejemplos:

- Ejemplo 1: ping google.com

```
C:\Users\aleja>ping google.com

Haciendo ping a google.com [142.250.185.14] con 32 bytes de datos:
Respuesta desde 142.250.185.14: bytes=32 tiempo=11ms TTL=118
Respuesta desde 142.250.185.14: bytes=32 tiempo=11ms TTL=118
Respuesta desde 142.250.185.14: bytes=32 tiempo=11ms TTL=118
Respuesta desde 142.250.185.14: bytes=32 tiempo=11ms TTL=118

Estadísticas de ping para 142.250.185.14:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 11ms, Máximo = 11ms, Media = 11ms

C:\Users\aleja>
```

- Ejemplo 2: ping 8.8.8.8

```
C:\Users\aleja>ping 8.8.8.8

Haciendo ping a 8.8.8.8 con 32 bytes de datos:
Respuesta desde 8.8.8.8: bytes=32 tiempo=12ms TTL=118
Respuesta desde 8.8.8.8: bytes=32 tiempo=12ms TTL=118
Respuesta desde 8.8.8.8: bytes=32 tiempo=12ms TTL=118
Respuesta desde 8.8.8.8: bytes=32 tiempo=12ms TTL=118

Estadísticas de ping para 8.8.8.8:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 12ms, Máximo = 12ms, Media = 12ms

C:\Users\aleja>
```

- Ejemplo 3: ping localhost

```
C:\Users\aleja>ping localhost

Haciendo ping a DESKTOP-KIHD5BU [::1] con 32 bytes de datos:
Respuesta desde ::1: tiempo<1m
Respuesta desde ::1: tiempo<1m
Respuesta desde ::1: tiempo<1m
Respuesta desde ::1: tiempo<1m

Estadísticas de ping para ::1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
      (0% perdidos),
  Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 0ms, Media = 0ms

C:\Users\aleja>
```

## 2. host

Definición: host es una herramienta para realizar consultas DNS y obtener información sobre nombres de dominio, como direcciones IP y registros DNS.

Ejemplos:

- Ejemplo 1: host google.com

```
usuario@usuario-VirtualBox:~$ host google.com
google.com has address 142.250.184.174
google.com has IPv6 address 2a00:1450:4003:80c::200e
google.com mail is handled by 10 smtp.google.com.
```

- Ejemplo 2: host www.example.com

```
usuario@usuario-VirtualBox:~$ host www.example.com
www.example.com has address 93.184.215.14
www.example.com has IPv6 address 2606:2800:21f:cb07:6820:80da:af6b:8b2c
```

- Ejemplo 3: host -t mx gmail.com

```
usuario@usuario-VirtualBox:~$ host -t mx gmail.com
gmail.com mail is handled by 5 gmail-smtp-in.l.google.com.
gmail.com mail is handled by 40 alt4.gmail-smtp-in.l.google.com.
gmail.com mail is handled by 10 alt1.gmail-smtp-in.l.google.com.
gmail.com mail is handled by 20 alt2.gmail-smtp-in.l.google.com.
gmail.com mail is handled by 30 alt3.gmail-smtp-in.l.google.com.
```

### 3. nslookup

Definición: nslookup es una herramienta de red que permite consultar servidores DNS para obtener información sobre un nombre de dominio, incluyendo direcciones IP y otros registros.

Ejemplos:

- Ejemplo 1: nslookup google.com

```
C:\Users\aleja>nslookup google.com
Servidor:  one.one.one.one
Address:  1.1.1.1

Respuesta no autoritativa:
Nombre:  google.com
```

- Ejemplo 2: nslookup example.com

```
C:\Users\aleja>nslookup example.com
Servidor:  one.one.one.one
Address:  1.1.1.1

Respuesta no autoritativa:
Nombre:  example.com
```

- Ejemplo 3: nslookup -type=mx gmail.com

```
C:\Users\aleja>nslookup -type=mx gmail.com
Servidor:  one.one.one.one
Address:  1.1.1.1

Respuesta no autoritativa:
gmail.com      MX preference = 20, mail exchanger = alt2.gmail-smtp-in.1.google.com
gmail.com      MX preference = 10, mail exchanger = alt1.gmail-smtp-in.1.google.com
gmail.com      MX preference = 40, mail exchanger = alt4.gmail-smtp-in.1.google.com
gmail.com      MX preference = 30, mail exchanger = alt3.gmail-smtp-in.1.google.com
gmail.com      MX preference = 5, mail exchanger = gmail-smtp-in.1.google.com
C:\Users\aleja>
```

#### 4. dig

Definición: dig (Domain Information Groper) es una herramienta avanzada para consultar DNS. Proporciona detalles sobre el sistema DNS y los registros asociados a un dominio.

Ejemplos:

- Ejemplo 1: dig google.com

```
usuario@usuario-VirtualBox:~$ dig google.com

; <<>> DiG 9.18.28-0ubuntu0.22.04.1-Ubuntu <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60244
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;google.com.                IN      A

;; ANSWER SECTION:
google.com.                 113     IN      A      142.250.184.174

;; Query time: 1 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Wed Oct 30 16:02:22 CET 2024
;; MSG SIZE rcvd: 55

usuario@usuario-VirtualBox:~$
```

- Ejemplo 2: dig example.com mx

```
usuario@usuario-VirtualBox:~$ dig example.com mx

; <<>> DiG 9.18.28-0ubuntu0.22.04.1-Ubuntu <<>> example.com mx
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 63545
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;example.com.                IN      MX

;; ANSWER SECTION:
example.com.                 12946   IN      MX      0 .

;; Query time: 16 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Wed Oct 30 16:02:41 CET 2024
;; MSG SIZE rcvd: 55

usuario@usuario-VirtualBox:~$
```



- Ejemplo 3: dig @8.8.8.8 google.com

```
usuario@usuario-VirtualBox:~$ dig @8.8.8.8 google.com

; <<>> DiG 9.18.28-0ubuntu0.22.04.1-Ubuntu <<>> @8.8.8.8 google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 6264
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;google.com.                IN      A

;; ANSWER SECTION:
google.com.                203     IN      A      142.250.184.174

;; Query time: 10 msec
;; SERVER: 8.8.8.8#53(8.8.8.8) (UDP)
;; WHEN: Wed Oct 30 16:03:22 CET 2024
;; MSG SIZE rcvd: 55

usuario@usuario-VirtualBox:~$
```

## Enlace video Explicando la Práctica

Te dejo aquí un link a youtube sobre mi explicación de la práctica

<https://youtu.be/XXn8Jq22KWo>