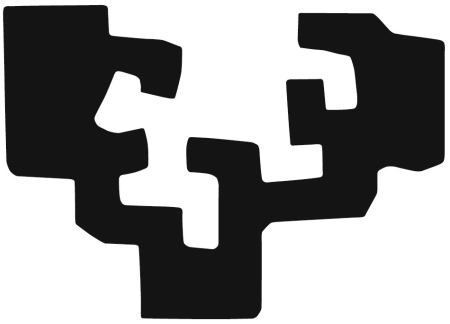
Práctica : Lab DNS

Grupo 01



UPV EHU

Alex Rivas Machín , Borja Gómez Calvo y Martín López de Ipiña Muñoz

Pasos Previos

Para realizar los ejercicios 6 (Delegación Segura) y 7 (NSEC3) del laboratorio, **asumimos que se han seguido los ejercicios 1 y 2** que establecen la configuración base del servidor DNS primario. Estos ejercicios posteriores se construyen sobre dicho entorno ya operativo.

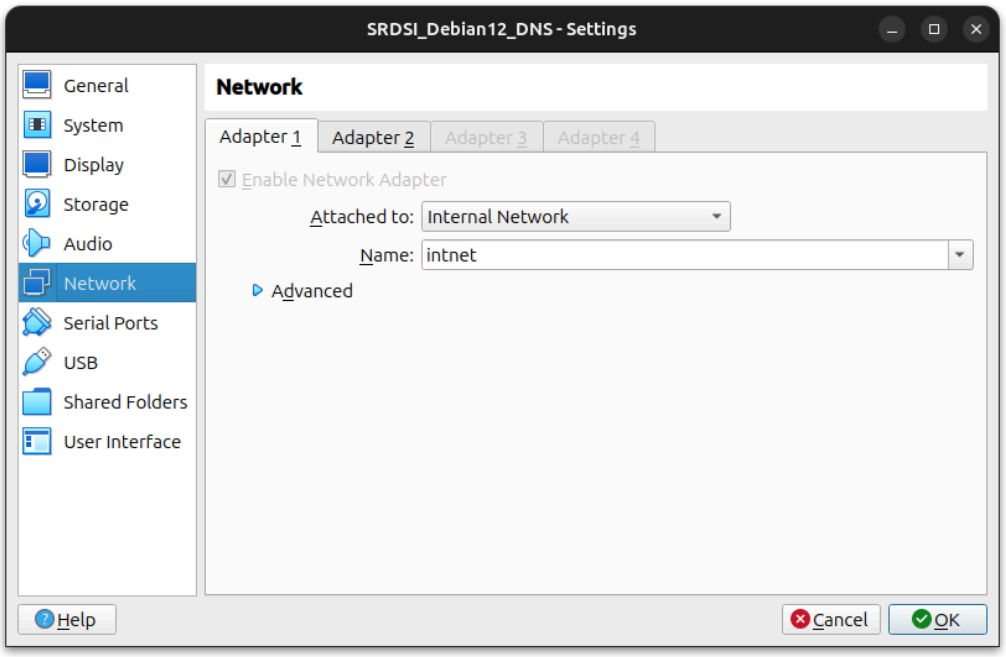
La configuración del laboratorio utiliza la dirección IP estática `172.16.0.136` para la máquina virtual que actúa como servidor DNS primario (`dns1.srdsi.lab`). Es necesario asignar esta IP estática, si no lo has hecho, aquí tienes los pasos para lograrlo:

La configuración por defecto del adaptador de red de la máquina virtual es de tipo NAT, por lo que necesitaremos añadir uno adicional de tipo interno para la ip estática. Se podría realizar sobre el adaptador que ya tenemos, pero perdemos el acceso al exterior del servidor.

Configuración de la Interfaz Primaria:

Esta interfaz (usualmente `enp0s3`) la convertiremos en red privada y tendrá una IP estática.

- **En VirtualBox/VMware:** Configure el "Adaptador 1" de la VM en modo "Red Interna"



- **Dentro de la máquina virtual (usando `/etc/network/interfaces`):** Configuraremos la dirección IP estática `172.16.0.136` para esta interfaz.
- Abre el fichero `/etc/network/interfaces`
- Busca la línea que configura tu interfaz (`enp0s3`) por DHCP (probablemente `iface enp0s3 inet dhcp`).
- Comenta o elimina esa línea y añade la configuración estática

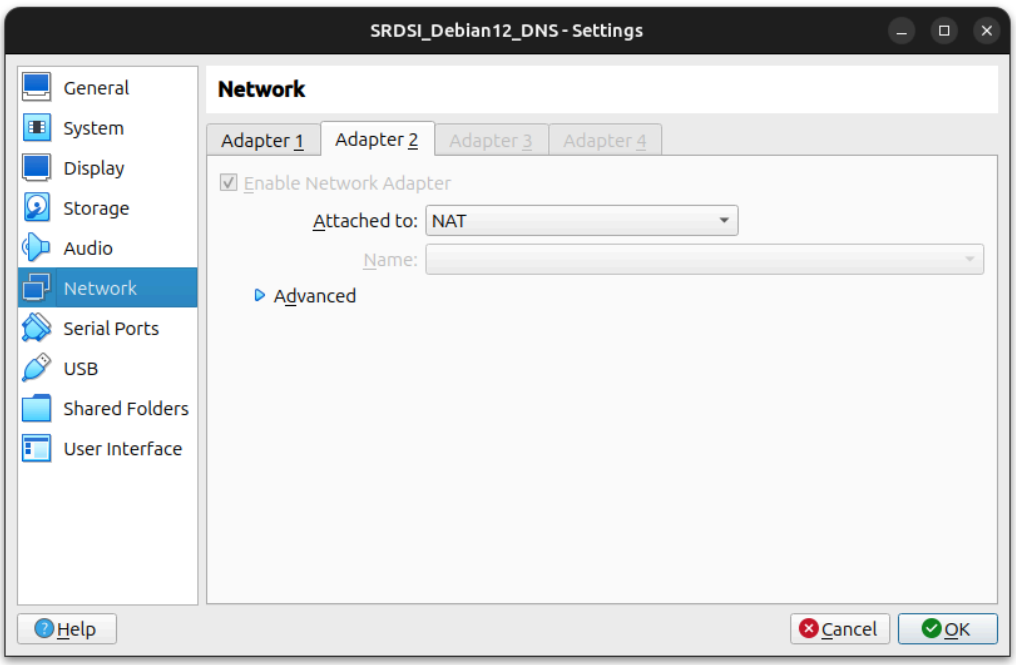
```
# This file describes the network interfaces available
# and how to activate them. For more information, see

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug enp0s3
iface enp0s3 inet static
    address 172.16.0.136
    netmask 255.255.255.0
```

- Guarda los cambios y cierra el archivo

Configuración de la Interfaz Secundaria (Acceso Externo):

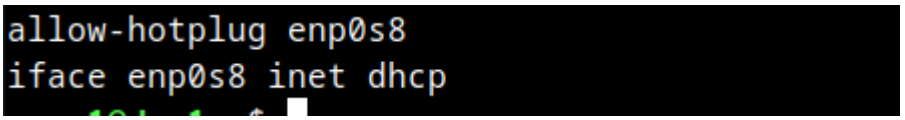
Esta interfaz proporcionará conectividad a Internet al servidor para que BIND pueda usar reenviadores (forwarders).

- **En VirtualBox/VMware:** Habilita el "Adaptador 2" de la VM y configúralo en modo **NAT**.



- **Dentro de la máquina virtual:** Esta segunda interfaz (que podría aparecer como `enp0s8` u otro nombre) debería obtener su configuración IP automáticamente vía DHCP.

- Comprueba que en `/etc/network/interfaces` contiene está configuración para dicha interfaz. Si no es el caso, agregala.



- Reinicia el servicio de red: `sudo systemctl restart networking` o reinicia la VM.
- Verifica la IP con `ip a`

¿verificación?

Ejercicio 6: Delegación segura

6.1 Crear subzona delegada

subG01.srdsi.lab

Crearemos la subzona delegada para el dominio `sub.srdsi.lab` como otra zona gestionada por el propio servidor `dns1.srdsi.lab`

- Edita `/etc/bind/named.conf.local`
- Añade una nueva definición de zona para `sub.srdsi.lab`

Contenido del fichero `/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";

// zona de resolución directa "srdsi.lab"
zone "srdsi.lab" {
     type primary;
     file "/etc/bind/db.srdsi.lab";
};
// zona de resolución inversa "0.16.172.in-addr.arpa"
zone "0.16.172.in-addr.arpa" {
     type primary;
     file "/etc/bind/db.0.16.172";
};
// zona de resolución directa "sub.srdsi.lab"
zone "sub.srdsi.lab" {
     type primary;
     file "/etc/bind/db.sub.srdsi.lab";
};
```

- Añadir en la zona padre los registros relativos a la subzona delegada

`/etc/bind/db.srdsi.lab`

```
$ORIGIN     srdsi.lab.
$TTL        2h
@     IN    SOA   dns1.srdsi.lab. admin.srdsi.lab. (
      20140401    ; Serial
      604800      ; Refresh
      86400       ; Retry
      2419200     ; Expire
      604800      )       ; Negative Cache TTL
```

```
;
srdsi.lab.      IN      NS      dns1.srdsi.lab.
;
dns1            IN      A       172.16.0.136
host1           IN      A       172.16.0.138
www             IN      CNAME   host1
;
sub     IN      NS      dns1.srdsi.lab.
sub     IN      A       172.16.0.136
```

- Crear el fichero de la subzona. `/etc/bind/db.sub.srdsi.lab` (el nombre que pusimos en `named.conf.local`).

`/etc/bind/db.sub.srdsi.lab`

```
$TTL    2h
@       IN      SOA     dns1.srdsi.lab. admin.srdsi.lab. (
                20140407        ; Serial
                604800          ; Refresh
                86400           ; Retry
                2419200         ; Expire
                604800 )        ; Negative Cache TTL
;
sub.srdsi.lab.  IN      NS      dns1.srdsi.lab.
;
host1   IN      A       172.16.145.45
host2   IN      A       172.16.145.46
```

- Verificar configuración

```
$ sudo named-checkconf /etc/bind/named.conf
$ sudo named-checkzone sub.srdsi.lab /etc/bind/db.sub.srdsi.lab
```

```
user1@dns1:~$ sudo named-checkconf /etc/bind/named.conf
user1@dns1:~$ sudo named-checkzone sub.srdsi.lab /etc/bind/db.sub.srdsi.lab
zone sub.srdsi.lab/IN: loaded serial 20140407
OK
user1@dns1:~$
```

- Recargar la configuración

```
$ sudo rndc reload
$ sudo rndc status
```

```
user1@dns1:~$ sudo rndc reload
server reload successful
user1@dns1:~$ sudo rndc status
version: BIND 9.18.33-1~deb12u2-Debian (Extended Support Version) <id:>
running on localhost: Linux x86_64 6.1.0-33-amd64 #1 SMP PREEMPT_DYNAMIC Debian
6.1.133-1 (2025-04-10)
boot time: Tue, 15 Apr 2025 16:20:14 GMT
last configured: Tue, 15 Apr 2025 17:06:19 GMT
configuration file: /etc/bind/named.conf
CPUs found: 2
worker threads: 2
UDP listeners per interface: 2
number of zones: 106 (98 automatic)
debug level: 0
xfers running: 0
xfers deferred: 0
soa queries in progress: 0
query logging is OFF
recursive clients: 0/900/1000
tcp clients: 0/150
TCP high-water: 0
server is up and running
```

- Pruebas con el resolver

```
$ dig host1.sub.srdsi.lab
$ dig host2.sub.srdsi.lab
```

```
user1@dns1:~$ dig @172.16.0.136 host1.sub.srdsi.lab

; <<>> DiG 9.18.33-1~deb12u2-Debian <<>> @172.16.0.136 host1.sub.srdsi.lab
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 11621
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 2879a880831cc2b50100000067fe929e1c7a373d6a9740fe (good)
;; QUESTION SECTION:
;host1.sub.srdsi.lab.           IN      A

;; ANSWER SECTION:
host1.sub.srdsi.lab.    7200    IN      A       172.16.145.45

;; Query time: 0 msec
;; SERVER: 172.16.0.136#53(172.16.0.136) (UDP)
;; WHEN: Tue Apr 15 19:08:46 CEST 2025
;; MSG SIZE  rcvd: 92
```

6.2 Firmar la subzona

- Generar Claves (KSK y ZSK) para la Subzona (en /etc/bind):

```
$ cd /etc/bind/
$ sudo dnssec-keygen -a RSASHA256 -b 1024 -f KSK -n ZONE sub.srdsi.lab.
$ sudo dnssec-keygen -a RSASHA256 -b 1024 -n ZONE sub.srdsi.lab.
```

```
user1@dns1:/etc/bind$ sudo dnssec-keygen -a RSASHA256 -b 1024 -f KSK -n ZONE sub
.srdsi.lab.
Generating key pair.................................+++++++++++++++++++++++++
++++++++++++++++++++++++++++++++++ ..........................+++++++++
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Ksub.srdsi.lab.+008+57662
user1@dns1:/etc/bind$ sudo dnssec-keygen -a RSASHA256 -b 1024 -n ZONE sub.srdsi.
lab.
Generating key pair.....++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++ .++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Ksub.srdsi.lab.+008+20974
user1@dns1:/etc/bind$ ls
bind.keys     db.srdsi.lab                    named.conf.default-zones
db.0          db.sub.srdsi.lab                named.conf.local
db.0.16.172   Ksub.srdsi.lab.+008+20974.key    named.conf.options
db.127        Ksub.srdsi.lab.+008+20974.private  rndc.key
db.255        Ksub.srdsi.lab.+008+57662.key    zones.rfc1918
db.empty      Ksub.srdsi.lab.+008+57662.private
db.local      named.conf
```

- Añadir los registros DNSKEY de las claves ZSK y KSK al chero de la zona `/etc/bind/db.sub.srdsi.lab` (actualizar serial)

```
  GNU nano 7.2              /etc/bind/db.sub.srdsi.lab *
$TTL    2h
$INCLUDE Ksub.srdsi.lab.+008+20974.key
$INCLUDE Ksub.srdsi.lab.+008+57662.key
@       IN              SOA     dns1.srdsi.lab. admin.srdsi.lab. (
        20140408                ; Serial
        604800                  ; Refresh
        86400                   ; Retry
        2419200                 ; Expire
        604800 )                ; Negative Cache TTL
;
sub.srdsi.lab.  IN      NS      dns1.srdsi.lab.
;
host1   IN      A       172.16.145.45
host2   IN      A       172.16.145.46
```

- Firmar la Subzona

Reemplaza el nombre de las claves privadas por las que has generado

```
$ cd /etc/bind/
$ sudo dnssec-signzone -t -o sub.srdsi.lab. -k Ksub.srdsi.lab.+008+57662.private
db.sub.srdsi.lab Ksub.srdsi.lab.+008+20974.private
```

```
user1@dns1:/etc/bind$ sudo dnssec-signzone -t -o sub.srdsi.lab. -k Ksub.srdsi.la
b.+008+57662.private db.sub.srdsi.lab Ksub.srdsi.lab.+008+20974.private
Verifying the zone using the following algorithms:
- RSASHA256
Zone fully signed:
Algorithm: RSASHA256: KSKs: 1 active, 0 stand-by, 0 revoked
                      ZSKs: 1 active, 0 stand-by, 0 revoked
db.sub.srdsi.lab.signed
Signatures generated:                       9
Signatures retained:                        0
Signatures dropped:                         0
Signatures successfully verified:           0
Signatures unsuccessfully verified:         0
Signing time in seconds:                0.012
Signatures per second:                750.000
Runtime in seconds:                     0.028
```

- Actualizar la declaración de la zona. Remplazar la zona “sub.srdsi.lab” en /etc/bind/named.conf.local por:

```
zone "srdsi.lab" {
        type primary;
        file "/etc/bind/db.sub.srdsi.lab.signed";
};
```

- Verificar y recargar la configuración del servidor

```
$ sudo named-checkconf /etc/bind/named.conf
$ sudo named-checkzone sub.srdsi.lab /etc/bind/db.sub.srdsi.lab.signed
```

```
user1@dns1:/etc/bind$ sudo named-checkconf /etc/bind/named.conf
user1@dns1:/etc/bind$ sudo named-checkzone sub.srdsi.lab /etc/bind/db.sub.srdsi.
lab.signed
zone sub.srdsi.lab/IN: loaded serial 20140408 (DNSSEC signed)
OK
```

```
$ sudo rndc reload
$ sudo rndc status
```

```
user1@dns1:/etc/bind$ sudo rndc reload
server reload successful
user1@dns1:/etc/bind$ sudo rndc status
version: BIND 9.18.33-1~deb12u2-Debian (Extended Support Version) <id:>
running on localhost: Linux x86_64 6.1.0-33-amd64 #1 SMP PREEMPT_DYNAMIC Debian
6.1.133-1 (2025-04-10)
boot time: Tue, 15 Apr 2025 16:20:14 GMT
last configured: Tue, 15 Apr 2025 17:29:08 GMT
configuration file: /etc/bind/named.conf
CPUs found: 2
worker threads: 2
UDP listeners per interface: 2
number of zones: 106 (98 automatic)
debug level: 0
xfers running: 0
xfers deferred: 0
soa queries in progress: 0
query logging is OFF
recursive clients: 0/900/1000
tcp clients: 0/150
TCP high-water: 0
server is up and running
```

- Comprobación

dig sub.srdsi.lab. +dnssec	respuesta negativa. Faltaría una prueba de respuesta positiva.

```
user1@dns1:/etc/bind$ dig @localhost sub.srdsi.lab. +dnssec

; <<>> DiG 9.18.33-1~deb12u2-Debian <<>> @localhost sub.srdsi.lab. +dnssec
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 47783
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 4, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 1232
; COOKIE: 505316d407fbf6cc0100000067fe981ba444b44fd77ff3c2 (good)
;; QUESTION SECTION:
;sub.srdsi.lab.                 IN      A

;; AUTHORITY SECTION:
sub.srdsi.lab.          7200    IN      SOA     dns1.srdsi.lab. admin.srdsi.lab. 20140408 6
04800 86400 2419200 604800
sub.srdsi.lab.          7200    IN      RRSIG   SOA 8 3 7200 20250515162310 20250415162310
20974 sub.srdsi.lab. Kkoq7nOaNn7ko2yZiqZkHidU5adBFa8lQNlNzLOMOI+pGCVWSI9P8hGt JtYsP+J7GABcU
w41NKrFS/IXKGm5eN2L9kKcxI3Af5sUX7+Dp42rZMVP nAQTP1Is39Cr+QD1x4sOclCpnMMyh+/lgGpQmlL7URl/8Ae
ahJe5SIQY SAo=
sub.srdsi.lab.          7200    IN      NSEC    host1.sub.srdsi.lab. NS SOA RRSIG NSEC DNSK
EY
sub.srdsi.lab.          7200    IN      RRSIG   NSEC 8 3 7200 20250515162310 20250415162310
 20974 sub.srdsi.lab. GUQUVjQO7jMiSTL9s4rIZTTwkFFJoJM4q0vuDvFXurbSzmtEjjO49sXQ WvTkSGmf7rJo
1eU+nSkNL+A5Ttylr1TXtOKSzy4faMZB2623+0B7xcvh QuzLt7FgmsU9kjvSgd+RT/Fw2sFG9rBgyAQI1ERTOcHNzP
od8A3AntDS CFc=

;; Query time: 0 msec
;; SERVER: ::1#53(localhost) (UDP)
;; WHEN: Tue Apr 15 19:32:11 CEST 2025
;; MSG SIZE  rcvd: 505
```

6.3 Agregar DS-RR de la subzona a la DB de la zona padre

Obtener el Registro DS

El comando `dnssec-sigzone` ejecutado anteriormente creó un fichero llamado `dsset-sub.srdsi.lab.` . Este archivo contiene el registro DS(Delegarion Signer) de nuestra subzona.

Añadir el Registro DS a la DB de la Zona Padre

- Copia el contenido del archivo `dsset-sub.srdsi.lab.`
- Pégalo en el archivo de la zona padre `/etc/bind/db.srdsi.lab`. Debería quedar junto a los registros NS de la delegación. (importante incrementa el número de serie)

```
  GNU nano 7.2                        db.srdsi.lab
$ORIGIN         srdsi.lab.
$TTL            2h
@               IN              SOA     dns1.srdsi.lab. admin.srdsi.lab. (
                20140402                ; Serial
                604800                  ; Refresh
                86400                   ; Retry
                2419200                 ; Expire
                604800          )       ; Negative Cache TTL
;                                .
srdsi.lab.      IN              NS      dns1.srdsi.lab.
;
dns1            IN              A       172.16.0.136
host1           IN              A       172.16.0.138
www             IN              CNAME   host1
;
sub             IN              NS      dns1.srdsi.lab.
sub             IN              A       172.16.0.136
;
sub.srdsi.lab.  IN      DS      7662 8 2 B67DBD0A5E467E2CD2A1F4D60169D16A5BC841>
```

- Comprobamos la configuración

```
$ sudo named-checkzone srdsi.lab /etc/bind/db.srdsi.lab
```

```
user1@dns1:/etc/bind$ sudo named-checkzone srdsi.lab /etc/bind/db.srdsi.lab
zone srdsi.lab/IN: loaded serial 20140402
OK
```

Re-firmar la zona padre y reiniciar servidor

Si no has creado previamente las claves para la zona padre, utiliza estos comandos:

```
$ cd /etc/bind/
$ sudo dnssec-keygen -a RSASHA256 -b 1024 -f KSK -n ZONE srdsi.lab.
$ sudo dnssec-keygen -a RSASHA256 -b 1024 -n ZONE srdsi.lab.
```

Añadir el nombre de las llaves públicas al fichero /etc/bind/db.srdsi.lab

```
  GNU nano 7.2                        db.srdsi.lab
$ORIGIN         srdsi.lab.
$TTL            2h
$INCLUDE        Ksrdsi.lab.+008+55987.key
$INCLUDE        Ksrdsi.lab.+008+62389.key
@               IN              SOA     dns1.srdsi.lab. admin.srdsi.lab. (
                20140402                ; Serial
                604800                  ; Refresh
```

Firma la zona padre (reemplaza el nombre de las llaves privadas por las que has generado):

```
$ cd /etc/bind/
$ sudo dnssec-signzone -o srdsi.lab. -k Ksrdsi.lab.+008+55987.private db.srdsi.lab
Ksrdsi.lab.+008+62389.private
```

```
user1@dns1:/etc/bind$ sudo dnssec-signzone -o srdsi.lab. -k Ksrdsi.lab.+008+5598
7.private db.srdsi.lab Ksrdsi.lab.+008+62389.private
Verifying the zone using the following algorithms:
- RSASHA256
Zone fully signed:
Algorithm: RSASHA256: KSKs: 1 active, 0 stand-by, 0 revoked
                      ZSKs: 1 active, 0 stand-by, 0 revoked
db.srdsi.lab.signed
user1@dns1:/etc/bind$
```

Si previamente habias firmado la zona se te creará una nueva versión firmada del archivo de zona (probablemente con un nombre como `db.srdsi.lab.signed.new`).

Sustituye el archivo de zona actual que BIND está utilizando por el nuevo archivo firmado. Esto generalmente implica renombrar archivos:

```
$ sudo mv /etc/bind/db.srdsi.lab.signed /etc/bind/db.srdsi.lab.signed.bak
$ sudo mv /etc/bind/db.srdsi.lab.signed.new /etc/bind/db.srdsi.lab.signed
```

Asegúrate que en el fichero /etc/bind/named.conf.local la zona padre apunta al fichero firmado.

```
// zona de resolución directa "srdsi.lab"
zone "srdsi.lab" {
        type primary;
        file "/etc/bind/db.srdsi.lab.signed";
};
```

Indica a BIND que recargue la configuración y los archivos de zona

```
$ sudo rndc reload srdsi.lab
```

Verificar

Usaremos dig para verificar que la cadena de confianza se valida correctamente

```
$ dig sub.srdsi.lab. +dnssec
$ dig srdsi.lab. +dnssec
```

¿Qué resultados se obtienen?

En la salida del comando verás el flag *ad*

Necesitas validar la cadena de confianza con delv

Ejercicio 7: NSEC3

Generación de las clave KSK y ZSK para la zona inversa

Con este comando creamos una clave pública/privada para DNSSEC utilizando el algoritmo RSASHA256, con un tamaño de clave de 1024 bits, marcada como KSK (Key Signing Key), para la zona inversa `0.16.172.in-addr.arpa`.

```
sudo dnssec-keygen -a RSASHA256 -b 1024 -f KSK -n ZONE 0.16.172.in-addr.arpa
```

- `-a RSASHA256`: Especifica el algoritmo de firma (RSASHA256).
- `-b 1024`: Define el tamaño de la clave (1024 bits).
- `-f KSK`: Marca esta clave como una clave de firma de zona (KSK).
- `-n ZONE`: Define que esta clave será utilizada para una zona DNS.

```
user1@dns1:~$ cd /etc/bind
user1@dns1:/etc/bind$ sudo dnssec-keygen -a RSASHA256 -b 1024 -f KSK -n ZONE 0.1
6.172.in-addr.arpa

Generating key pair.....................................................
..................+++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++ ..........................................+++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++
K0.16.172.in-addr.arpa.+008+07525
```

Generamos otra clave para firmar los registros reales de la zona. sta clave será de uso diario para firmar los registros de la zona. No se marca como KSK, ya que se utiliza para firmar los registros DNS.

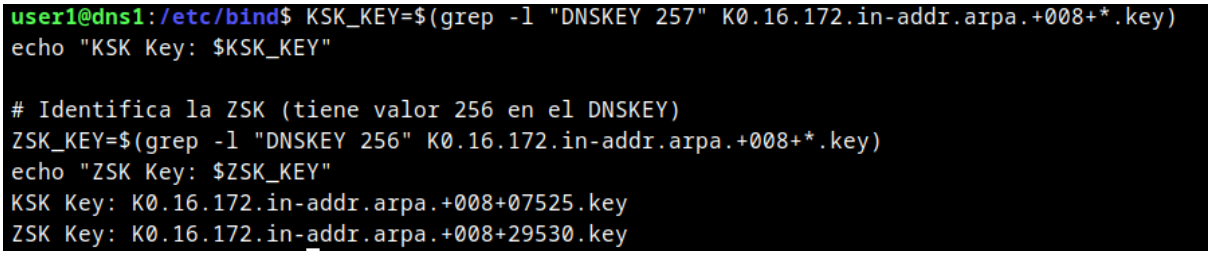
```
sudo dnssec-keygen -a RSASHA256 -b 1024 -n ZONE 0.16.172.in-addr.arpa
```

```
user1@dns1:/etc/bind$ sudo dnssec-keygen -a RSASHA256 -b 1024 -n ZONE 0.16.172.i
n-addr.arpa

Generating key pair.....................................................
.................................................+++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++ ..........................................
.........................+++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++
K0.16.172.in-addr.arpa.+008+29530
```

Identificación de las claves
Estos comandos identifican automáticamente los nombres de los archivos de claves generados, que contienen números aleatorios. Guardan en variables de entorno los nombres de los ficheros con las claves.

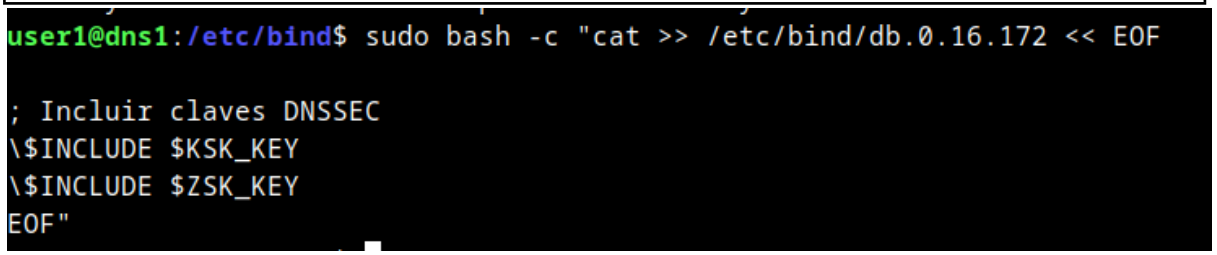
El identificador 257 corresponde a una KSK en el archivo DNSKEY. El comando almacena el nombre del archivo en la variable KSK_KEY, de la misma forma, la clave ZSK, que tiene el identificador 256.

```
KSK_KEY=$(grep -l "DNSKEY 257" K0.16.172.in-addr.arpa.+008+*.key)
echo "KSK Key: $KSK_KEY"

ZSK_KEY=$(grep -l "DNSKEY 256" K0.16.172.in-addr.arpa.+008+*.key)
echo "ZSK Key: $ZSK_KEY"
```



Incluir las claves públicas en el archivo de zona
Agregamos las claves públicas generadas a la configuración de la zona, permitiendo que DNSSEC pueda validar las respuestas de la zona.

```
# Actualiza el archivo de zona para incluir las claves
sudo bash -c "cat >> /etc/bind/db.0.16.172 << EOF

; Incluir claves DNSSEC
\$INCLUDE $KSK_KEY
\$INCLUDE $ZSK_KEY
EOF"
```



Verificación de la sintaxis del archivo de zona

Verificamos que el archivo de zona modificado esté libre de errores antes de proceder con la firma de la zona.

```
sudo named-checkzone 0.16.172.in-addr.arpa /etc/bind/db.0.16.172
```

```
user1@dns1:/etc/bind$ sudo named-checkzone 0.16.172.in-addr.arpa /etc/bind/db.0
16.172
zone 0.16.172.in-addr.arpa/IN: loaded serial 20140401
OK
```

Generación de un valor aleatorio (salt) para NSEC3

Creamos un valor aleatorio (salt) que se usará para incrementar la seguridad de los registros NSEC3, lo que ayuda a prevenir ataques de diccionario.

```
SALT=$(head -c 16 /dev/random | sha1sum | cut -b 1-16)
echo "Salt generado: $SALT"
```

```
user1@dns1:/etc/bind$ SALT=$(head -c 16 /dev/random | sha1sum | cut -b 1-16)
echo "Salt generado: $SALT"
Salt generado: dee4cb07155103b7
```

Firma de la zona con NSEC3

Firmamos la zona utilizando NSEC3 para añadir protección adicional contra ataques, como los de diccionario. El comando firma todos los registros de la zona con una clave RSA.

```
sudo dnssec-signzone -A -3 $SALT -t -S -o 0.16.172.in-addr.arpa -f db.0.16.172.signed
db.0.16.172
```

- -A: Incluye todas las firmas
- -3: Activa NSEC3 en lugar de NSEC
- -t: Imprime estadísticas
- -S: Firma el SOA por separado (permite pre-publicación)
- -o: Especifica el origen de la zona
- -f: Indica el nombre del archivo de salida

```
user1@dns1:/etc/bind$ sudo dnssec-signzone -A -3 $SALT -t -S -o 0.16.172.in-addr
.arpa -f db.0.16.172.signed db.0.16.172

Fetching 0.16.172.in-addr.arpa/RSASHA256/29530 (ZSK) from key repository.
Verifying the zone using the following algorithms:
- RSASHA256
Zone fully signed:
Algorithm: RSASHA256: KSKs: 1 active, 0 stand-by, 0 revoked
                      ZSKs: 1 active, 0 stand-by, 0 revoked
db.0.16.172.signed
Signatures generated:                       10
Signatures retained:                         0
Signatures dropped:                          0
Signatures successfully verified:            0
Signatures unsuccessfully verified:          0
Signing time in seconds:                 0.016
Signatures per second:                 625.000
Runtime in seconds:                      0.044
```

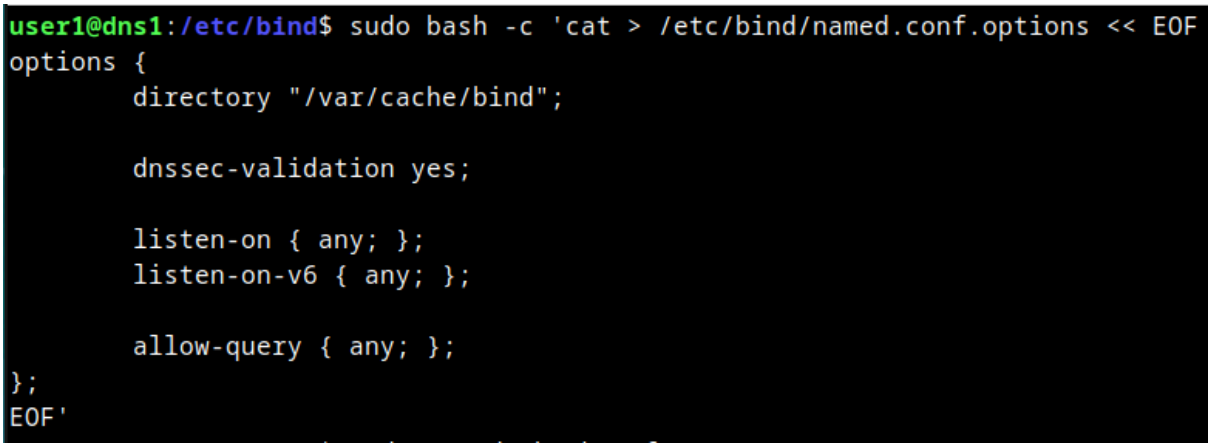
Actualización de BIND para usar el archivo de zona firmado
Actualizamos la configuración de BIND para que apunte al archivo de zona firmado, en lugar del archivo original no firmado.

```
sudo sed -i 's|file "/etc/bind/db.0.16.172";|file "/etc/bind/db.0.16.172.signed";|' /etc/bind/named.conf.local
```

Configuración de BIND para habilitar DNSSEC

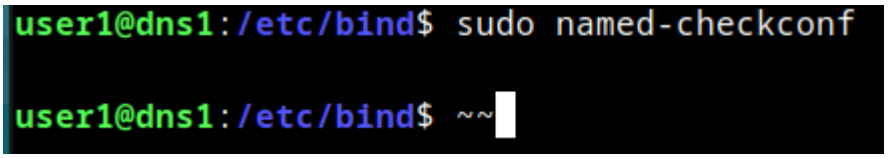
Configuramos BIND para que valide las firmas DNSSEC, lo que es crucial para garantizar la autenticidad de las respuestas de la zona.

```
sudo bash -c 'cat > /etc/bind/named.conf.options << EOF
options {
        directory "/var/cache/bind";

        dnssec-validation yes;

        listen-on { any; };
        listen-on-v6 { any; };

        allow-query { any; };
};
EOF'
```



Verificación de la sintaxis de la configuración de BIND
Verifica que la configuración de BIND no tenga errores antes de reiniciar el servicio.

```
sudo named-checkconf
```

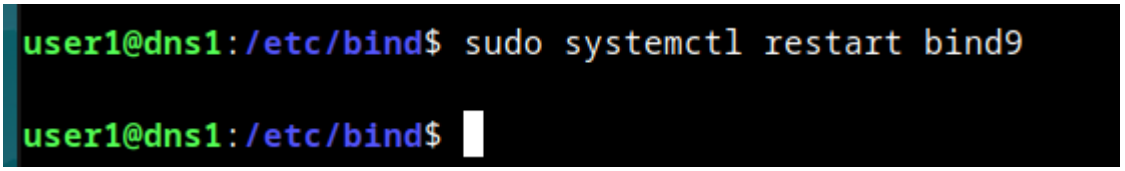


zona

Reiniciar el servidor DNS

Reiniciamos el servidor para aplicar todo y ver que todo funcione correctamente.

```
sudo systemctl restart bind9
```



Creación de un archivo de clave pública KSK como punto de confianza

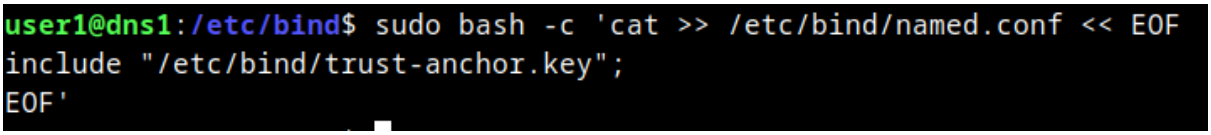
Este archivo contiene la clave KSK para que los clientes puedan validar la autenticidad de la zona.

```
sudo bash -c "cat > /etc/bind/trust-anchor.key << EOF
trusted-keys {
        0.16.172.in-addr.arpa. 257 3 8
\"AwEAAZbit17Gg6YrQMBhdnzdivu1m4caTnrNMav0lU+5K5d0+E4QQjSeEPzGStTIjmg3
Sv8M67XAFcX/HVnc2h6jWVRGPF80uHQVLTJcuX92wbI49EeTQ6O8T9JbBb49mBLQCk
ouWtDXWkKGwmhYyDgdvaUEGsb7pgKi1Ed7IqCBmeFD\";
};
EOF"
```

Añadir referencia al archivo de trust anchor en la configuración de BIND

Configuramos BIND para incluir el archivo de clave pública KSK como un punto de confianza en la validación de DNSSEC.

```
sudo bash -c 'cat >> /etc/bind/named.conf << EOF
include "/etc/bind/trust-anchor.key";
EOF'
```



No habéis probrado la cadena de confianza

Cliente (Verificación)

Verificamos que la zona esté configurada para usar NSEC3 mediante la consulta de registros `NSEC3PARAM`.

dig @127.0.0.1 0.16.172.in-addr.arpa NSEC3PARAM +dnssec

```
user1@dns1:~$ dig @127.0.0.1 0.16.172.in-addr.arpa NSEC3PARAM +dnssec


; <<>> DiG 9.18.33-1~deb12u2-Debian <<>> @127.0.0.1 0.16.172.in-addr.arpa NSEC3PARAM +dnssec
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 10264
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 1232
; COOKIE: 7af2e03b6005cd560100000067fed0d18d075c5d0acf175e (good)
;; QUESTION SECTION:
;0.16.172.in-addr.arpa.         IN      NSEC3PARAM

;; ANSWER SECTION:
0.16.172.in-addr.arpa.  0       IN      NSEC3PARAM 1 0 0 F6EBFD0A40C1181B
0.16.172.in-addr.arpa.  0       IN      RRSIG   NSEC3PARAM 8 5 0 20250515202131 20250415202131 29530 0.16.17
in-addr.arpa. VQ8FgJm5PnKxwlLR+hKuykAsYA1Vh6lXg0RuAZnDIHXdnzptVXWnI6yW IKKCRcY2zZGj6t4Hss4AUeOYS0O+tUmV/F2/o
HX6NROJeiGZf/Omns jLKG+kaDeSEs4kUXpyjlxgok1HzZkMBIYNxmCpyR8Xlutxnslq9DcyF0 vYQ=

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1) (UDP)
;; WHEN: Tue Apr 15 23:34:09 CEST 2025
;; MSG SIZE  rcvd: 284
```

Mostramos una consulta exitosa con registros RRSIG que validan criptográficamente la respuesta.

dig @127.0.0.1 -x 172.16.0.136 +dnssec

```
user1@dns1:~$ dig @127.0.0.1 -x 172.16.0.136 +dnssec


; <<>> DiG 9.18.33-1~deb12u2-Debian <<>> @127.0.0.1 -x 172.16.0.136 +dnssec
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 5448
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 1232
; COOKIE: 3e4e5a779a2a791f0100000067fed0d65bcbfd4067f9aa87 (good)
;; QUESTION SECTION:
;136.0.16.172.in-addr.arpa.     IN      PTR

;; ANSWER SECTION:
136.0.16.172.in-addr.arpa. 604800 IN    PTR     dns1.srdsi.lab.
136.0.16.172.in-addr.arpa. 604800 IN    RRSIG   PTR 8 6 604800 20250515202131 20250415202131 29530 0.16.172.in
-addr.arpa. AEeiaedQxz5ym1Q4E0zCey0n3WbVGEkDrC1CWQLHHu7alJbayVJfHc46 Txt5n+IOlFIAUFokQlusYr6CCbP0pCZp0vwchr+m0
3wghnClL9pGCx3A pydDxJN5YdtFQIx4wn4NBGBlwwZ2cHyrTi4kIXiTcKHmaIVUwz1GJSte NaU=

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1) (UDP)
;; WHEN: Tue Apr 15 23:34:14 CEST 2025
;; MSG SIZE  rcvd: 291
```

Cuando consultamos por una IP inexistente (172.16.0.150), se obtiene una respuesta NXDOMAIN con registros NSEC3.

dig @127.0.0.1 -x 172.16.0.150 +dnssec

```
ser1@dns1:~$ dig @127.0.0.1 -x 172.16.0.150 +dnssec


 <<>> DiG 9.18.33-1~deb12u2-Debian <<>> @127.0.0.1 -x 172.16.0.150 +dnssec
 (1 server found)
; global options: +cmd
; Got answer:
; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 55893
; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 4, ADDITIONAL: 1

; OPT PSEUDOSECTION:
 EDNS: version: 0, flags: do; udp: 1232
 COOKIE: 665d1b04582a5cba0100000067fed0e7e1e2e5e79eea2cb7 (good)
; QUESTION SECTION:
150.0.16.172.in-addr.arpa.     IN      PTR

; AUTHORITY SECTION:
.16.172.in-addr.arpa.  604800  IN      SOA     dns1.srdsi.lab. admin.srdsi.lab. 20140401 604800 86400 2419200 604800
.16.172.in-addr.arpa.  604800  IN      RRSIG   SOA 8 5 604800 20250515202131 20250415202131 29530 0.16.172.in-addr.arpa. TUO1w
/P07Hn3MDXr6k2ZMhpiecYFTUtt3I6zG2Y++rItv/5iihMYNTBv tWnCVgllnfQ+yyahHqVGEUnZUnB1OBdEKIdvGHxUe2wtxvKTy6wMrDDv jpjODZgnN+IrImAael
AJ4F1ekiRQ8o4ipLZnWki96o8KuLdqfp6g9zE Pxs=
3HSIFMDJ9JS08OP4SOGHD2MSJVKLAH1.0.16.172.in-addr.arpa. 604800 IN NSEC3 1 1 0 F6EBFD0A40C1181B MRE6V32S5TSHIQP4VCK3O4RO0LEE7PQ6
 NS SOA RRSIG DNSKEY NSEC3PARAM
3HSIFMDJ9JS08OP4SOGHD2MSJVKLAH1.0.16.172.in-addr.arpa. 604800 IN RRSIG NSEC3 8 6 604800 20250515202131 20250415202131 29530 0.
6.172.in-addr.arpa. U/K10uu3ZduCtxovcczzgmg/Mr7TrCoSOOGoIvFCmtdjFU/mDBWcERCe rf+r1/sfPGWcfyuHLCahWbBQV+kM4qzb05EAfLu2nUpizV1ZG
hemeAm itGRMToQYx32BAvOuk+NtwC5aqyDR+JMVdBMP0MUWIGw1dg0cdsT8RZS 4hI=

; Query time: 0 msec
; SERVER: 127.0.0.1#53(127.0.0.1) (UDP)
; WHEN: Tue Apr 15 23:34:31 CEST 2025
; MSG SIZE  rcvd: 588
```

Consultamos específicamente los registros NSEC3 en formato

dig @127.0.0.1 0.16.172.in-addr.arpa NSEC3 +dnssec +multi

```
user1@dns1:~$ dig @127.0.0.1 0.16.172.in-addr.arpa NSEC3 +dnssec +multi


; <<>> DiG 9.18.33-1~deb12u2-Debian <<>> @127.0.0.1 0.16.172.in-addr.arpa NSEC3 +dnssec +multi
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15148
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 4, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 1232
; COOKIE: 222cb4655cab083e0100000067fed105953338a4d77fe791 (good)
;; QUESTION SECTION:
;0.16.172.in-addr.arpa. IN NSEC3

;; AUTHORITY SECTION:
0.16.172.in-addr.arpa.  604800 IN SOA dns1.srdsi.lab. admin.srdsi.lab. (
                              20140401   ; serial
```

Tiempo dedicado

El tiempo dedicado total ha sido de 11 horas.

Pasos previos: 2h
Ejercicio 6: 5h
Ejercicio 7: 4h