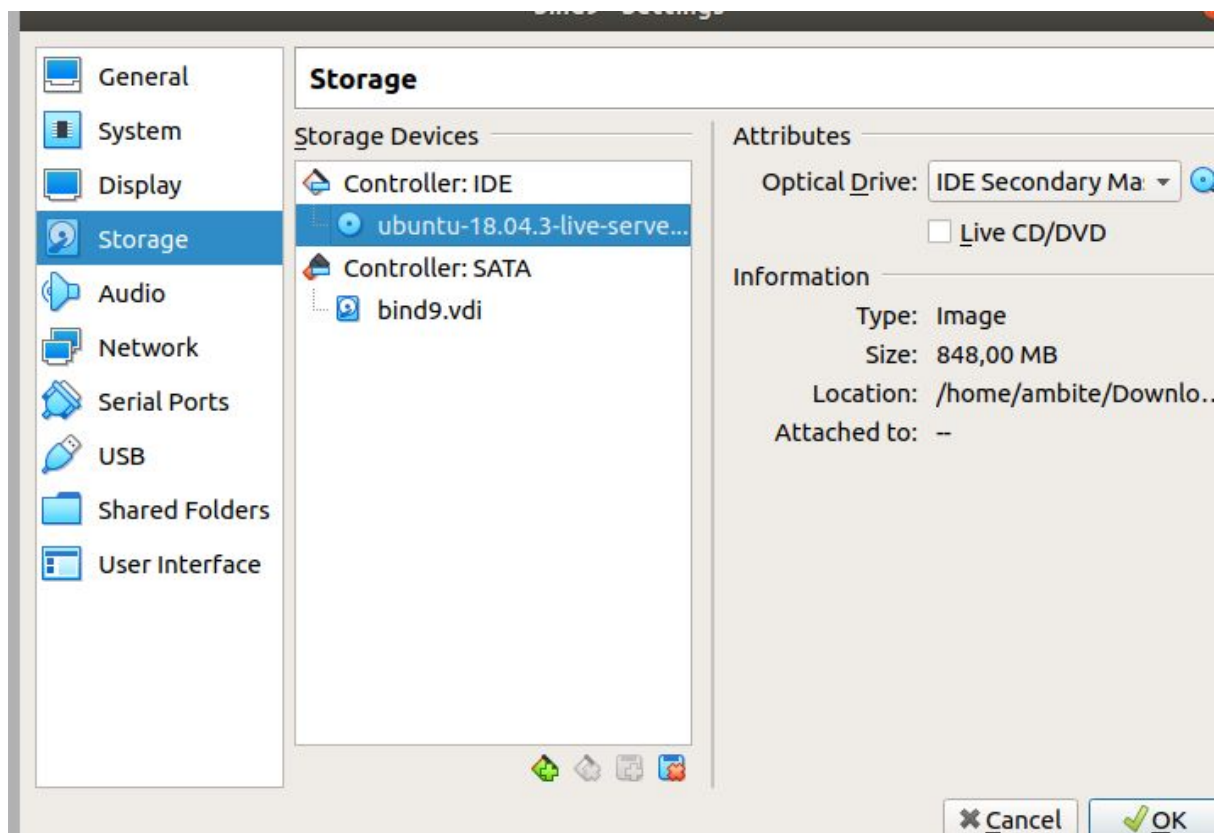


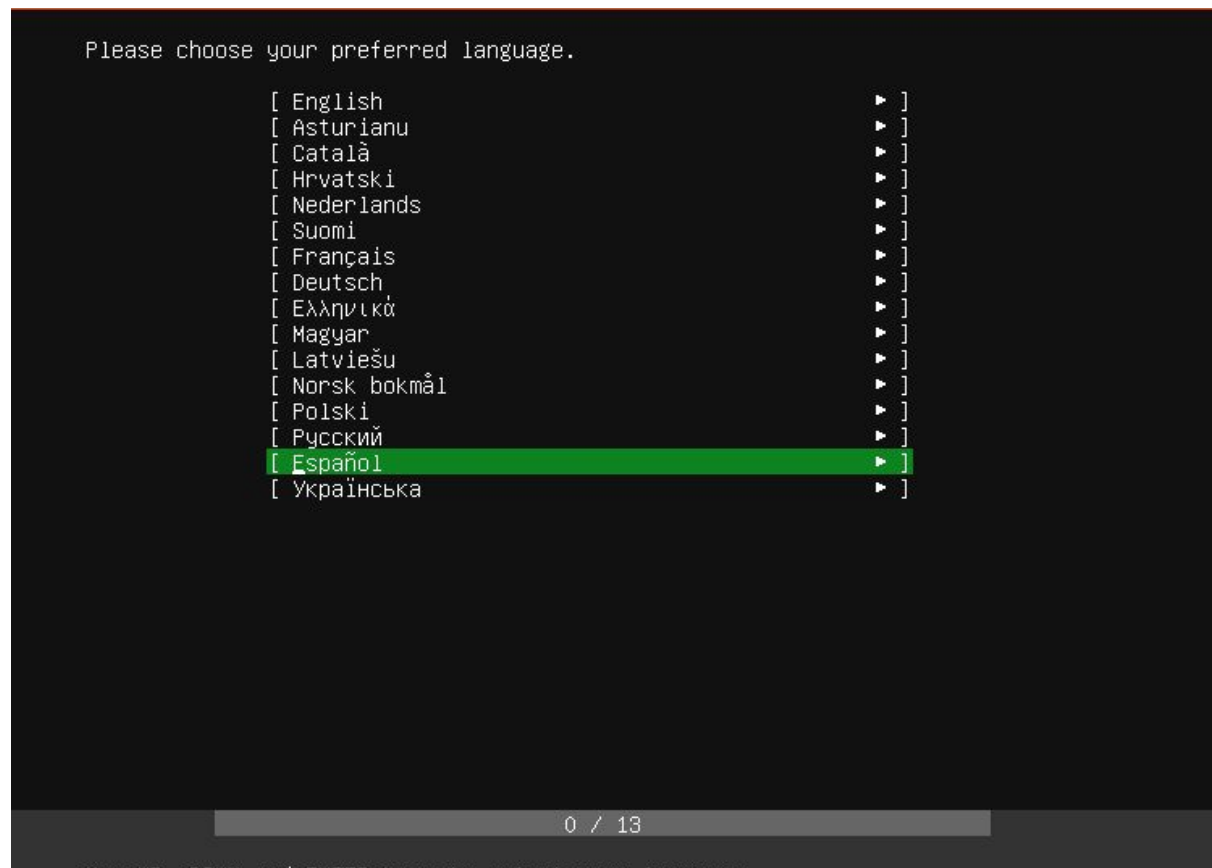
INSTALACIÓN Y CONFIGURACIÓN DE DNS

By Kessusa

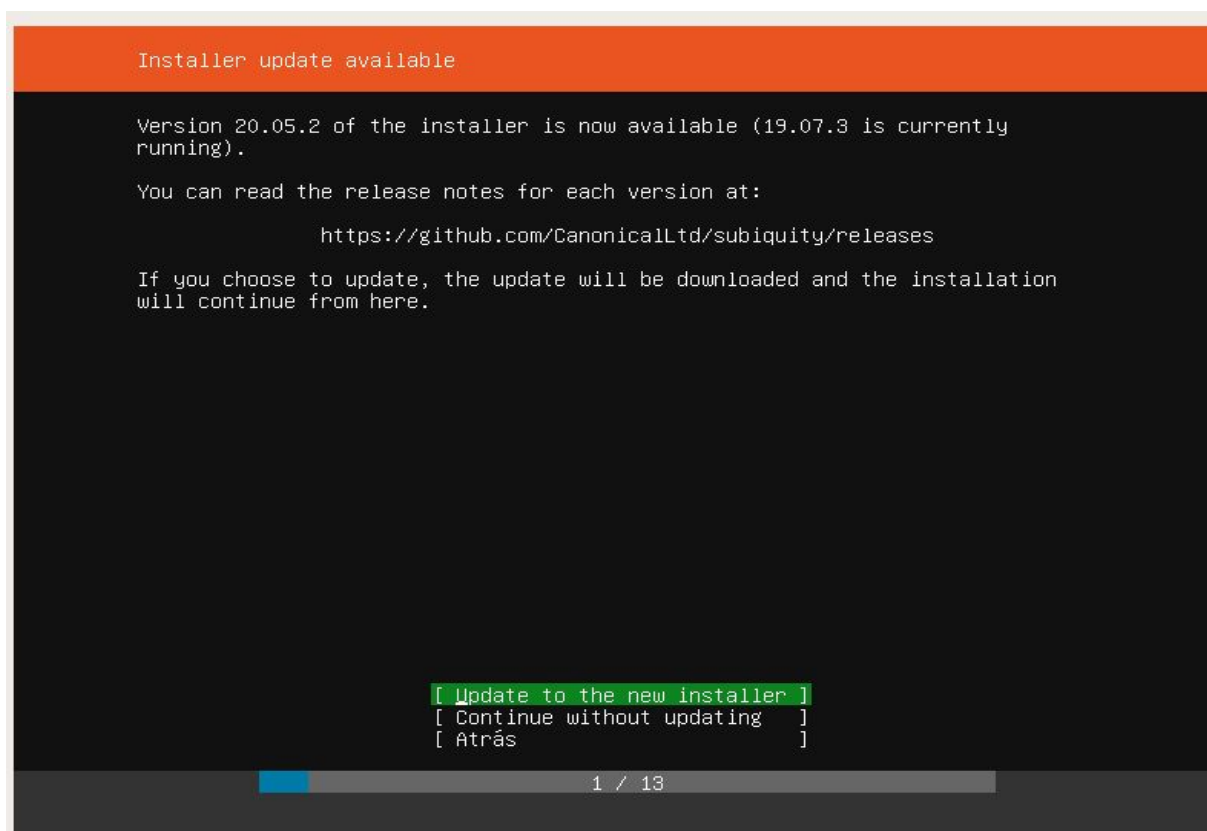
Vamos a configurar dos máquinas virtuales con ubuntu server para que funcionen como un dns, con zona de maestra y zona de resolución inversa. Primero vamos a crear la máquina virtual con ubuntu server, creamos una nueva máquina y nos vamos a settings y en optical drive cargamos la iso de ubuntu live server.



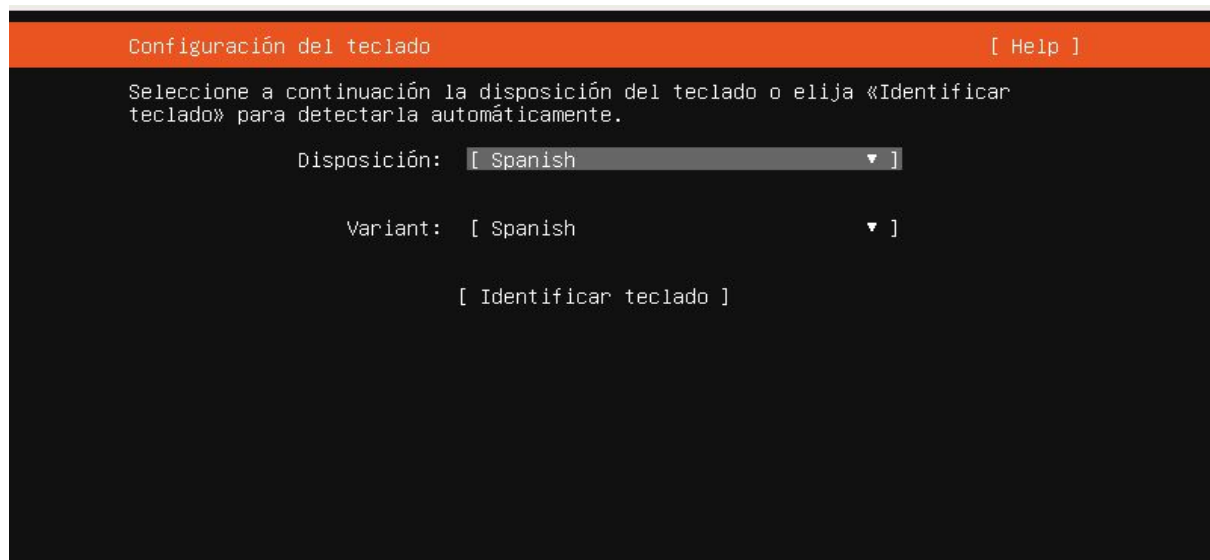
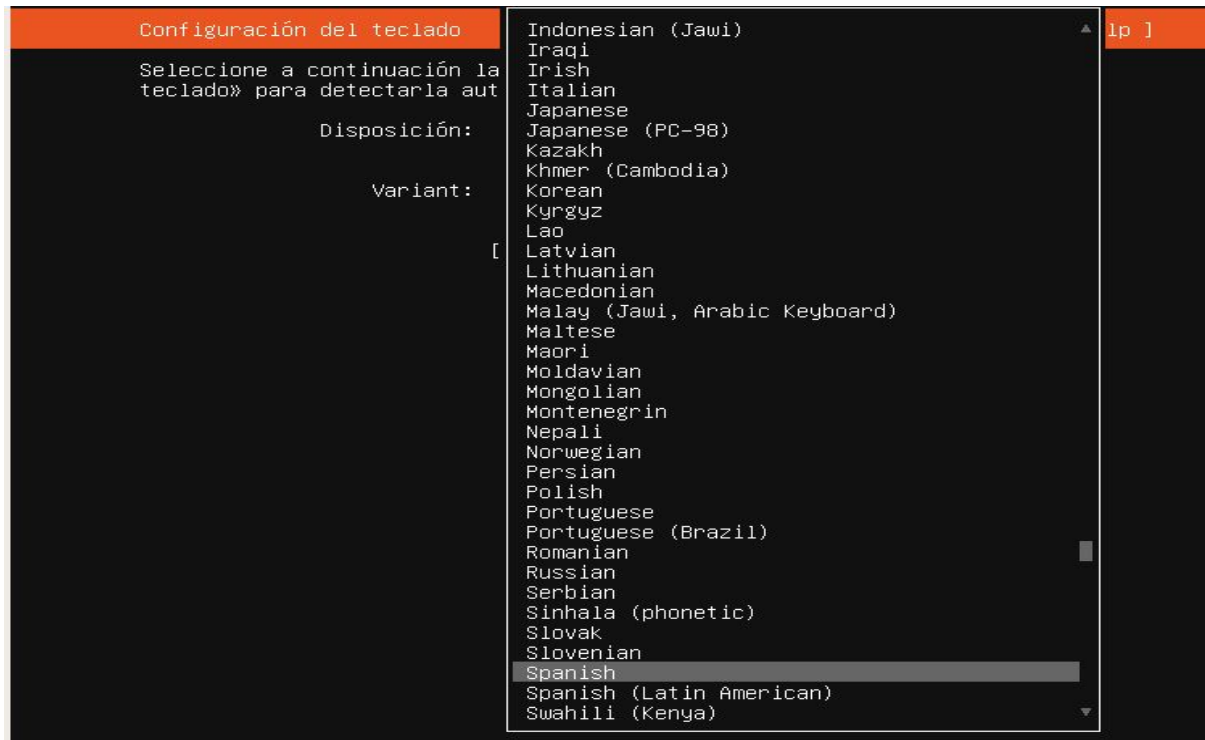
Lo primero que nos pide es el idioma he elegido Español en este caso aunque recomiendo hacerlo en inglés.



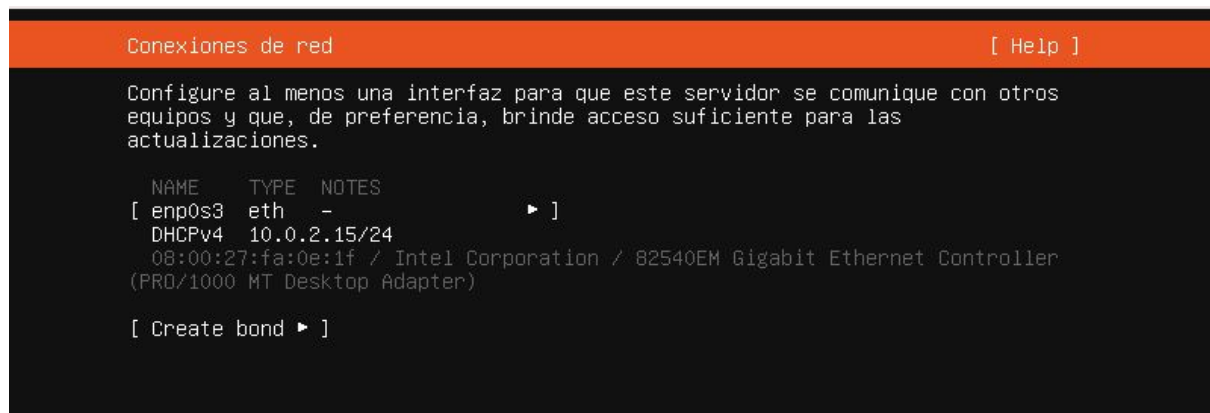
Elijo actualizar el instalador a la nueva versión.



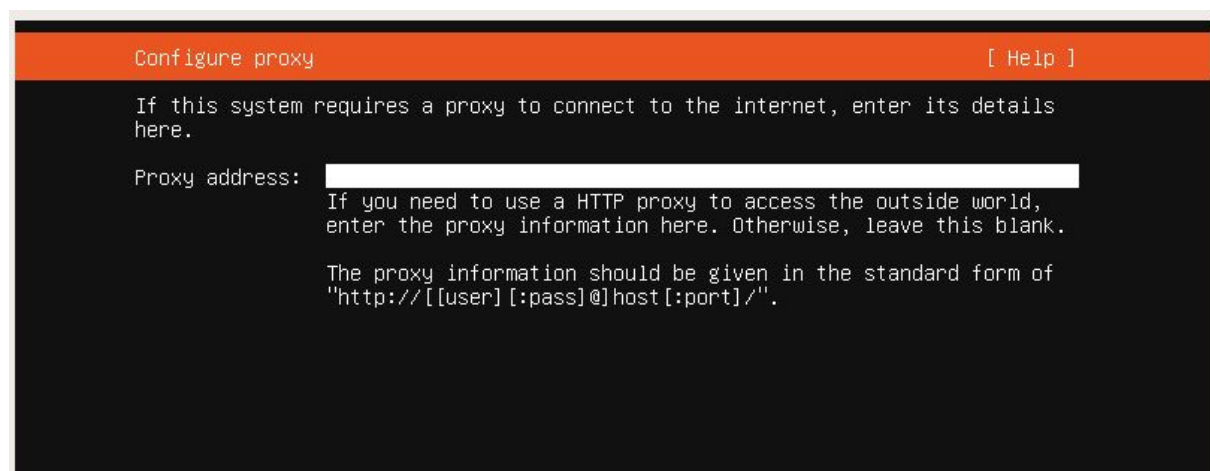
Descargar unos paquetes y puede tardar un rato .



Configurar el adaptador de red.



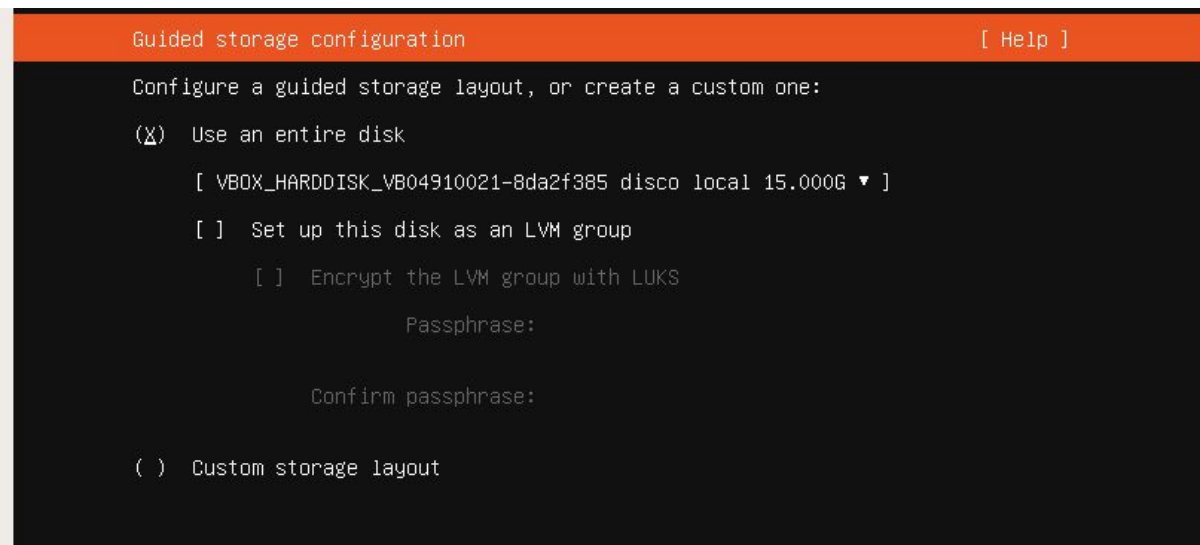
Configuramos el proxy en nuestro caso no vamos a poner ninguno por ahora.



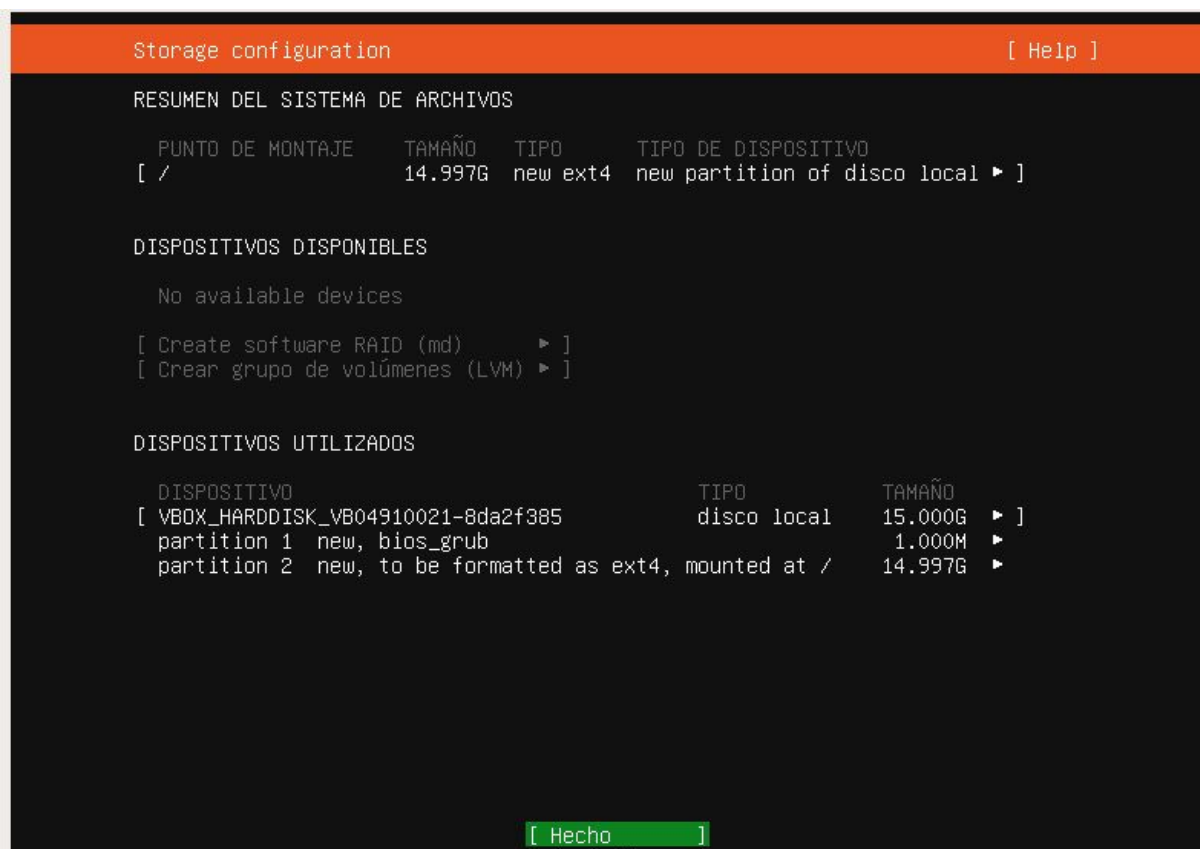
Configurar el mirror por defecto dejamos el que viene.



Donde vamos a instalar el so dejamos la opción de disco entero por defecto.



Confirmamos la instalación y revisamos el sistema de archivos u las particiones que se van a realizar.



Creamos el nombre del servidor , y el usuario con la contraseña que usaremos.

Configuración de perfil [Help]

Proporcione el nombre de usuario y la contraseña que utilizará para acceder al sistema. Puede configurar el acceso SSH en la pantalla siguiente, pero aun se necesita una contraseña para sudo.

Su nombre:

El nombre del servidor:
El nombre que utiliza al comunicarse con otros equipos.

Elija un nombre de usuario:

Elija una contraseña:

Confirme la contraseña:

Instalamos el servidor ssh para poder gestionar los servidores en remoto.

Configuración de SSH [Help]

You can choose to install the OpenSSH server package to enable secure remote access to your server.

☒ Instalar servidor OpenSSH

Importar identidad SSH:
Puede importar sus claves SSH desde GitHub o Launchpad.

Importar nombre de usuario:

☒ Permitir autenticación con contraseña por SSH

No seleccionamos ningún servidor ni aplicación más e iniciamos la instalación.

```
Se ha completado la instalación. [ Help ]

Ha finalizado la instalación.
running '/snap/bin/subiquity.subiquity-configure-run'
running '/snap/bin/subiquity.subiquity-configure-apt
/snap/subiquity/1874/usr/bin/python3 true'
curtin command apt-config
curtin command in-target
running 'curtin curthooks'
curtin command curthooks
configuring apt configuring apt
installing missing packages
configuring iscsi service
configuring raid (mdadm) service
installing kernel
setting up swap
apply networking config
writing etc/fstab
configuring multipath
updating packages on target system
configuring pollinate user-agent on target
updating initramfs configuration
finalizing installation
running 'curtin hook'
curtin command hook
executing late commands
final system configuration
configuring cloud-init
installing openssh-server
restoring apt configuration
downloading and installing security updates |

[ View full log ]
[ Cancelar actualización o reiniciar ]
```

En mi caso se actualizó después de instalar y me llevo un buen rato.

```
Se ha completado la instalación. [ Help ]

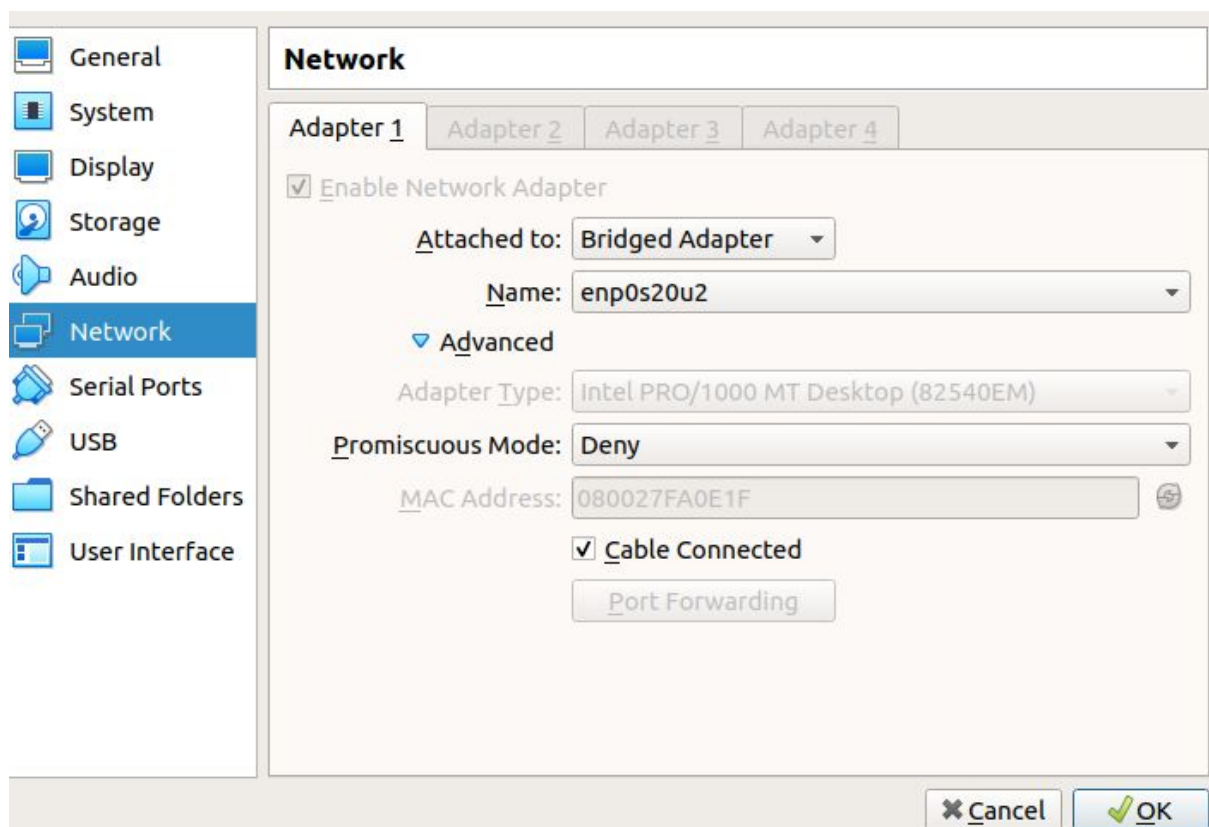
Ha finalizado la instalación.
running '/snap/bin/subiquity.subiquity-configure-run'
running '/snap/bin/subiquity.subiquity-configure-apt
/snap/subiquity/1874/usr/bin/python3 true'
curtin command apt-config
curtin command in-target
running 'curtin curthooks'
curtin command curthooks
configuring apt configuring apt
installing missing packages
configuring iscsi service
configuring raid (mdadm) service
installing kernel
setting up swap
apply networking config
writing etc/fstab
configuring multipath
updating packages on target system
configuring pollinate user-agent on target
updating initramfs configuration
finalizing installation
running 'curtin hook'
curtin command hook
executing late commands
final system configuration
configuring cloud-init
installing openssh-server
restoring apt configuration
downloading and installing security updates

[ View full log ]
[ Reiniciar ]
```

Comprobamos que la instalación está completada guiándonos y viendo que somos el usuario que hemos creado y el so que tenemos instalado.

```
airon@terminator:/$ whoami
airon
airon@terminator:/$ uname -a
Linux terminator 4.15.0-101-generic #102-Ubuntu SMP Mon May 11 10:07:26 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
airon@terminator:/$ _
```

Una vez que hemos visto que está bien instalada creamos un clon de la máquina, le cambio la mac antes de arrancarla.



En mi caso al arrancar el servidor dhcp me dió la misma ip para las dos máquinas para solventarlo hice un `sudo dhclient -r -v` para renovar la solicitud de ip y me dio otra distinta hice ping de una a la otra para comprobar la conectividad y lo hacían sin problemas así que ya estaban listas ambas para poder empezar a trabajar con ellas.

Por último vamos a instalar bind9 para poder configurar nuestros servidores de DNS en la próxima clase.

Para ello simplemente ejecutaremos el siguiente comando en ambas máquinas

- **`sudo apt install bind9 bind9utils bind9-doc`**


```
Processing triggers for ufw (0.36-0ubuntu0.18.04.1) ...
airon@terminator:~$ sudo apt install bind9 bind9utils bind9-doc _
```

Vamos a comprobar que están funcionando los servidores .

- ***sudo lsof -i -P -n***

```
Processing triggers for ufw (0.36-0ubuntu0.18.04.1) ...
airon@terminator:~$ sudo lsof -i -n -P
COMMAND  PID  USER      FD  TYPE DEVICE SIZE/OFF NODE NAME
systemd-n 682  systemd-network 20u IPv4  16582    0t0  UDP 192.168.42.111:68
systemd-r 701  systemd-resolve 12u IPv4  16555    0t0  UDP 127.0.0.53:53
systemd-r 701  systemd-resolve 13u IPv4  16556    0t0  TCP 127.0.0.53:53 (LISTEN)
sshd      1064  root       3u  IPv4  19992    0t0  TCP *:22 (LISTEN)
sshd      1064  root       4u  IPv6  20009    0t0  TCP *:22 (LISTEN)
named     2637  bind       21u IPv6  26472    0t0  TCP *:53 (LISTEN)
named     2637  bind       22u IPv4  26476    0t0  TCP 127.0.0.1:53 (LISTEN)
named     2637  bind       23u IPv4  26482    0t0  TCP 192.168.42.111:53 (LISTEN)
named     2637  bind       24u IPv4  26491    0t0  TCP 127.0.0.1:953 (LISTEN)
named     2637  bind       25u IPv6  26492    0t0  TCP [::1]:953 (LISTEN)
named     2637  bind       512u IPv6  26471    0t0  UDP *:53
named     2637  bind       513u IPv4  26475    0t0  UDP 127.0.0.1:53
named     2637  bind       514u IPv4  26481    0t0  UDP 192.168.42.111:53
airon@terminator:~$
```

Como podemos ver están escuchando en el puerto tcp y udp 53, y vamos a comprobar estado del servicio con el siguiente comando.

- ***sudo systemctl status bind9.service***

```
airon@terminator:~$ sudo systemctl status bind9.service
[sudo] password for airon:
• bind9.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/bind9.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2020-06-07 16:01:08 UTC; 18min ago
     Docs: man:named(8)
   Main PID: 2637 (named)
    Tasks: 4 (limit: 1108)
   CGroup: /system.slice/bind9.service
           └─2637 /usr/sbin/named -f -u bind

Jun 07 16:01:14 terminator named[2637]: network unreachable resolving 'E.ROOT-SERVERS.NET/AAAA/IN':
Jun 07 16:01:15 terminator named[2637]: DNS format error from 198.41.0.4#53 resolving './NS: non-impr
Jun 07 16:01:15 terminator named[2637]: FORMERR resolving './NS/IN': 198.41.0.4#53
Jun 07 16:01:16 terminator named[2637]: DNS format error from 192.33.4.12#53 resolving './NS: non-imp
Jun 07 16:01:16 terminator named[2637]: FORMERR resolving './NS/IN': 192.33.4.12#53
Jun 07 16:01:17 terminator named[2637]: DNS format error from 192.36.148.17#53 resolving './NS: non-i
Jun 07 16:01:17 terminator named[2637]: FORMERR resolving './NS/IN': 192.36.148.17#53
Jun 07 16:01:17 terminator named[2637]: network unreachable resolving './NS/IN': 2001:500:12::d0d#53
Jun 07 16:01:17 terminator named[2637]: network unreachable resolving './NS/IN': 2001:500:a8::e#53
Jun 07 16:01:18 terminator named[2637]: resolver priming query complete
```

Una de las máquinas la vamos a configurar como un servidor dns primario y la otra como secundario.

Vamos primero con la máquina que será el servidor maestro o primario.

Vamos a ponerle un nombre a la máquina que podamos reconocer, por ejemplo dns1.kessusa a la primaria y dns2.kessusa a la secundaria.

Para ello cambiamos el nombre de la máquina a dns1.kessusa y el fichero host ponemos la ip de nuestra máquina con ese nombre.

```
GNU nano 2.9.3 /etc/hostname Mod
dns1.kessusa
```

```
GNU nano 2.9.3 /etc/hosts
127.0.0.1 localhost
127.0.1.1 terminator1
192.168.42.111 dns1.kessusa
```

Con esto ya cambiado vamos a crear una zona en el archivo named.conf.local podríamos configurarlos también en named.conf.default-zones.

- ***sudo nano named.conf.local***

Vamos a configurar lo primero la zona de resolución directa, yo he llamado a la zona así para saber el dominio que voy a estar gestionando pero se le puede poner el nombre que queramos.

Lo importante aquí es poner type master y el archivo del que va a coger los datos.

```
//
zone "www.aambite-example.org"{
    type master;
    file "/etc/bind/db.aambite-example.org";
};
```

También he creado una zona de resolución inversa.

Tenemos las mismas consideraciones tipo maestro y el archivo del que va a coger la información.

```
zone "0.42.168.192.in-addr.arpa"{
    type master;
    file "/etc/bind/db.42.168.192";
};
```

Cuando tenemos esto creado lo guardamos y vamos a crear los archivos de las zonas.

Para crear estos archivos voy a explicar para qué sirve cada campo dentro del mismo.

TTL:Time to live número de segundos que puede estar el registro en caché antes de ser borrado.

SOA:start of authority es el primer registro de una zona y define las opciones generales de la zona.

Tiene el nombre del servidor,correo de contacto de la persona responsable del dominio

Número de serie:indica la versión del archivo y 7 es importante porque los servidores secundarios utilizan este dato para actualizarse cuando cambia.

Actualización:Tiempo que tardan los servidores esclavos en preguntar al maestro por los cambios.

Caducidad:tiempo que el servidor esclavo intenta contactar con el maestro para ver si hay transferencia de zona, es importante porque si este tiempo expira el servidor esclavo o secundario dejará de ser autorizado para la zona.

TTL negativo: tiempo mínimo que se almacenan las respuestas negativas para la zona.

Para generar un nuevo fichero de zona lo más fácil es copiar alguno de los que hay de ejemplo y hacer los cambios que queramos.

- ***sudo cp db.local db.www.aambite-example.org***

Como podemos ver tenemos el fichero genérico .

```
GNU nano 2.9.3 db.www.aambite-example.org
;
; BIND data file for local loopback interface
;
$TTL 604800
@ IN SOA localhost. root.localhost. (
    2          ; Serial
    604800     ; Refresh
    86400      ; Retry
    2419200    ; Expire
    604800 )   ; Negative Cache TTL
;
@ IN NS localhost.
@ IN A 127.0.0.1
@ IN AAAA ::1
```

Lo voy a configurar con cosas muy básicas que resuelva las direcciones de correo un alias para el ftp y la de mi página web.

```

GNU nano 2.9.3                                     db.www.aambite-example.org
;
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA      localhost. root.localhost. (
                        1006201          ; Serial
                        604800           ; Refresh
                        86400            ; Retry
                        2419200          ; Expire
                        604800 )         ; Negative Cache TTL
;
@         IN      NS       localhost.
@         IN      A        127.0.0.1
@         IN      AAAA     ::1
@         IN      MX       10 correo.aambite-example.org.
www.aambite-example.org. IN A      192.168.42.111
correo    IN      A        192.168.42.112
ftp.aambite-example.org. IN CNAME   www.aambite-example.org.

```

Después crea el fichero de la zona inversa copiando también uno de los que tenemos de muestra y modificandolo para mi caso.

- ***cp db.127 db.42.168.192***

```

GNU nano 2.9.3                                     db.42.168.192
; BIND reverse data file for local loopback interface
;
$TTL      604800
@         IN      SOA      localhost. root.localhost. (
                        1006201          ; Serial
                        604800           ; Refresh
                        86400            ; Retry
                        2419200          ; Expire
                        604800 )         ; Negative Cache TTL
;
@         IN      NS       localhost.
111      IN      PTR      www.aambite-example.org.

```

Una vez tengo los archivos y las zonas creadas reinicio el servicio y probamos si funciona.

- ***sudo systemctl start bind9.service***

- ***sudo systemctl status bind9.service***

Para comprobar si el servicio se ha ejecutado correctamente.

```
root@dns1:/etc/bind# systemctl status bind9.service
● bind9.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/bind9.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2020-06-10 08:59:35 UTC; 11s ago
     Docs: man:named(8)
  Process: 1550 ExecStop=/usr/sbin/rndc stop (code=exited, status=0/SUCCESS)
 Main PID: 1553 (named)
    Tasks: 4 (limit: 1108)
   CGroup: /system.slice/bind9.service
           └─1553 /usr/sbin/named -f -u bind

Jun 10 08:59:43 dns1.kessusa named[1553]: network unreachable resolving 'E.ROOT-SERVERS.NET/AAAA/IN'
Jun 10 08:59:43 dns1.kessusa named[1553]: network unreachable resolving 'E.ROOT-SERVERS.NET/AAAA/IN'
Jun 10 08:59:43 dns1.kessusa named[1553]: network unreachable resolving 'G.ROOT-SERVERS.NET/AAAA/IN'
Jun 10 08:59:43 dns1.kessusa named[1553]: network unreachable resolving 'G.ROOT-SERVERS.NET/AAAA/IN'
Jun 10 08:59:43 dns1.kessusa named[1553]: network unreachable resolving 'E.ROOT-SERVERS.NET/AAAA/IN'
Jun 10 08:59:43 dns1.kessusa named[1553]: network unreachable resolving 'E.ROOT-SERVERS.NET/AAAA/IN'
Jun 10 08:59:43 dns1.kessusa named[1553]: network unreachable resolving 'G.ROOT-SERVERS.NET/AAAA/IN'
Jun 10 08:59:45 dns1.kessusa named[1553]: DNS format error from 192.5.5.241#53 resolving ./NS: non-i
Jun 10 08:59:45 dns1.kessusa named[1553]: FORMERR resolving './NS/IN': 192.5.5.241#53
Jun 10 08:59:45 dns1.kessusa named[1553]: resolver priming query complete
lines 1-20/20 (END)
```

Vamos a comprobar que resuelve el servidor la dirección que hemos creado
www.ambite-example.org.

```
root@dns1:/etc/bind# nslookup
> server localhost
Default server: localhost
Address: 127.0.0.1#53
> www.aambite-example.org
Server:          localhost
Address:         127.0.0.1#53

Name:   www.aambite-example.org
Address: 192.168.42.111
> _
```

```

root@dns1:/etc/bind# dig @dns1.kessusa www.aambite-example.org ANY

; <<> DiG 9.11.3-ubuntu1.12-Ubuntu <<> @dns1.kessusa www.aambite-example.org ANY
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 18258
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 3

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 3d83f92a79c057cfe6562ab85ee0aa211f05534b827a3a24 (good)
;; QUESTION SECTION:
;www.aambite-example.org.      IN      ANY

;; ANSWER SECTION:
www.aambite-example.org. 604800 IN      A      192.168.42.111

;; AUTHORITY SECTION:
aambite-example.org.    604800 IN      NS     localhost.

;; ADDITIONAL SECTION:
localhost.              604800 IN      A      127.0.0.1
localhost.              604800 IN      AAAA   ::1

;; Query time: 0 msec
;; SERVER: 192.168.42.111#53(192.168.42.111)
;; WHEN: Wed Jun 10 09:38:41 UTC 2020
;; MSG SIZE rcvd: 163

```

Como vemos el dns ya está resolviendo y ahora vamos a crear el dns secundario para eso debemos ir de nuevo al archivo `named.conf.local` y añadir que permitimos el compartir zona con servidor esclavo y le diremos la ip del servidor que será el secundario. Aquí nos vamos a ir a ese archivo y poner los siguientes cambios.

```

GNU nano 2.9.3      named.conf.local

//
// Do any local configuration here
//
zone "aambite-example.org"{
    type master;
    file "/etc/bind/db.www.aambite-example.org";
    allow-transfer { 192.168.42.111; };
    notify yes ;
};


zone "42.168.192.in-addr.arpa"{
    type master;
    file "/etc/bind/db.42.168.192";
    allow-transfer { 192.168.42.111;};
    notify yes;
};

```


Con esto ya lo que debemos hacer es irnos a la segunda máquina vamos a cambiarle el nombre y a hacer los mismos pasos que en la máquina anterior solo que ahora voy a llamarla dns2.kessus.

Después de modificar el archivo nos queda así, podemos ver la ruta a la que se envían los ficheros de zona de los dns esclavos es distinta a la de los maestros.

Después de esto reiniciamos el servicio.



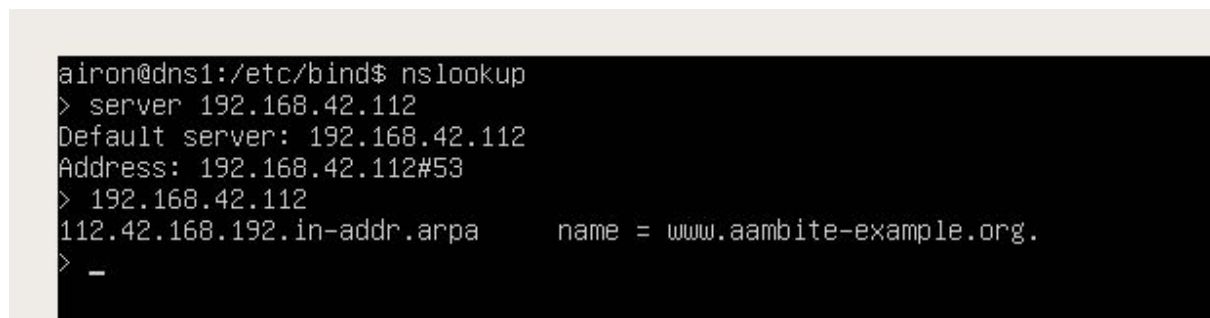
```
bind9segunda [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
GNU nano 2.9.3 /etc/bind/named.conf.local

//
// Do any local configuration here
//
zone "aambite-example.org"{
    type slave;
    file "/var/lib/bind/db.www.aambite-example.org";
    masters { 192.168.42.112; };
};

zone "42.168.192.in-addr.arpa"{
    type slave;
    file "/var/lib/bind/db.42.168.192";
    masters { 192.168.42.112; };
};

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";
```

Como podemos ver esta resolviendo



```
airon@dns1:/etc/bind$ nslookup
> server 192.168.42.112
Default server: 192.168.42.112
Address: 192.168.42.112#53
> 192.168.42.112
112.42.168.192.in-addr.arpa      name = www.aambite-example.org.
> -
```

tenemos que ir a la ruta /var/lib/bind de la máquina esclava para ver si de verdad se ha producido la transferencia de zona los ficheros que ahí se crean no son en texto plano sino binarios.

```
bind9segunda [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
airon@dns2:/var/lib/bind$ ls
bind9-default.md5sum db.42.168.192 db.www.aambite-example.org
airon@dns2:/var/lib/bind$ _
```

Vemos que se han creado y vamos a probar a resolver con el servidor secundario una consulta a ver si funciona.

```
airon@dns2:/var/lib/bind$ dig @dns2.kessusa -t A www.aambite-example.org

; <<>> DiG 9.11.3-ubuntu1.12-Ubuntu <<>> @dns2.kessusa -t A www.aambite-example.org
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 62257
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 3

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:;, udp: 4096
; COOKIE: aee2ee26aea558193bfe02685ee0c441cbd53b5867a78e39 (good)
;; QUESTION SECTION:
;www.aambite-example.org.      IN      A

;; ANSWER SECTION:
www.aambite-example.org. 604800 IN      A      192.168.42.111

;; AUTHORITY SECTION:
aambite-example.org.    604800 IN      NS      localhost.

;; ADDITIONAL SECTION:
localhost.              604800 IN      A      127.0.0.1
localhost.              604800 IN      AAAA   ::1

;; Query time: 0 msec
;; SERVER: 192.168.42.112#53(192.168.42.112)
;; WHEN: Wed Jun 10 11:30:09 UTC 2020
;; MSG SIZE rcvd: 163
```

```
17 File Machine view Input Devices Help
airon@dns2:/var/lib/bind$ nslookup
> server dns2.kessusa
Default server: dns2.kessusa
Address: 192.168.42.112#53
> 192.168.42.111
** server can't find 111.42.168.192.in-addr.arpa: NXDOMAIN
> 192.168.42.112
112.42.168.192.in-addr.arpa      name = www.aambite-example.org.
> 192.168.42.113
113.42.168.192.in-addr.arpa     name = correo.aambite-example.org.
>
```



```

;; MSG SIZE rcvd: 163

airon@dns2:/var/lib/bind$ dig @dns1.kessusa -t A www.aambite-example.org

; <<>> DiG 9.11.3-1ubuntu1.12-Ubuntu <<>> @dns1.kessusa -t A www.aambite-example.org
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1637
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 3

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: ed3737591df8c4ad39ea22045ee0c45f4f8d507eba6cdd3 (good)
;; QUESTION SECTION:
;www.aambite-example.org.      IN      A

;; ANSWER SECTION:
www.aambite-example.org. 604800 IN      A      192.168.42.111

;; AUTHORITY SECTION:
aambite-example.org.    604800 IN      NS      localhost.

;; ADDITIONAL SECTION:
localhost.              604800 IN      A      127.0.0.1
localhost.              604800 IN      AAAA   ::1

;; Query time: 0 msec
;; SERVER: 192.168.42.147#53(192.168.42.147)
;; WHEN: Wed Jun 10 11:30:39 UTC 2020
;; MSG SIZE rcvd: 163

```

Como podemos ver ya resuelven los dos dns tanto el maestro como el esclavo.

Por último decir que para ver los posibles problemas de dns además de la info del status del servidor que suele mostrar los errores más importantes podemos ir al fichero de logs.

/var/log/syslog y aquí veremos los errores por ejemplo voy cuando intente hacer la transferencia de zona que me daba un error de permisos al intentar crear el archivo en la ruta que yo le había especificado y también me descuenta d en problema por cambio de ip de dhcp que no me están cargando la transferencia de zona por supuesto que deberemos poner ip fija nuestros servidores.

Puedes echar un vistazo a los logs de transferencia de zona por ejemplo en el dns2 a ver cuales son las que le han cargado el día 10 de junio.

Esto puede ser interesante para ver cuantas actualizaciones hace el servidor en un día o si alguien está actualizando sin que nosotros queramos.

```

airon@dns2:~$ grep -E "Jun 10.*transferred serial" /var/log/syslog
Jun 10 10:45:28 terminator named[3212]: zone aambite-example.org/IN: transferred serial 2
Jun 10 11:27:36 dns2 named[1806]: zone 42.168.192.in-addr.arpa/IN: transferred serial 1006202
Jun 10 11:28:25 dns2 named[1846]: zone 42.168.192.in-addr.arpa/IN: transferred serial 1006202
airon@dns2:~$

```

Podemos hacer una expresión regular para contar cuantas se han realizado hoy por ejemplo teniendo en cuenta que sean distintas entre sí.

Para ello cojo solo las dos últimas columnas de este comando las ordenó con la opción unique de sort y cuento las líneas que son con wc -l.

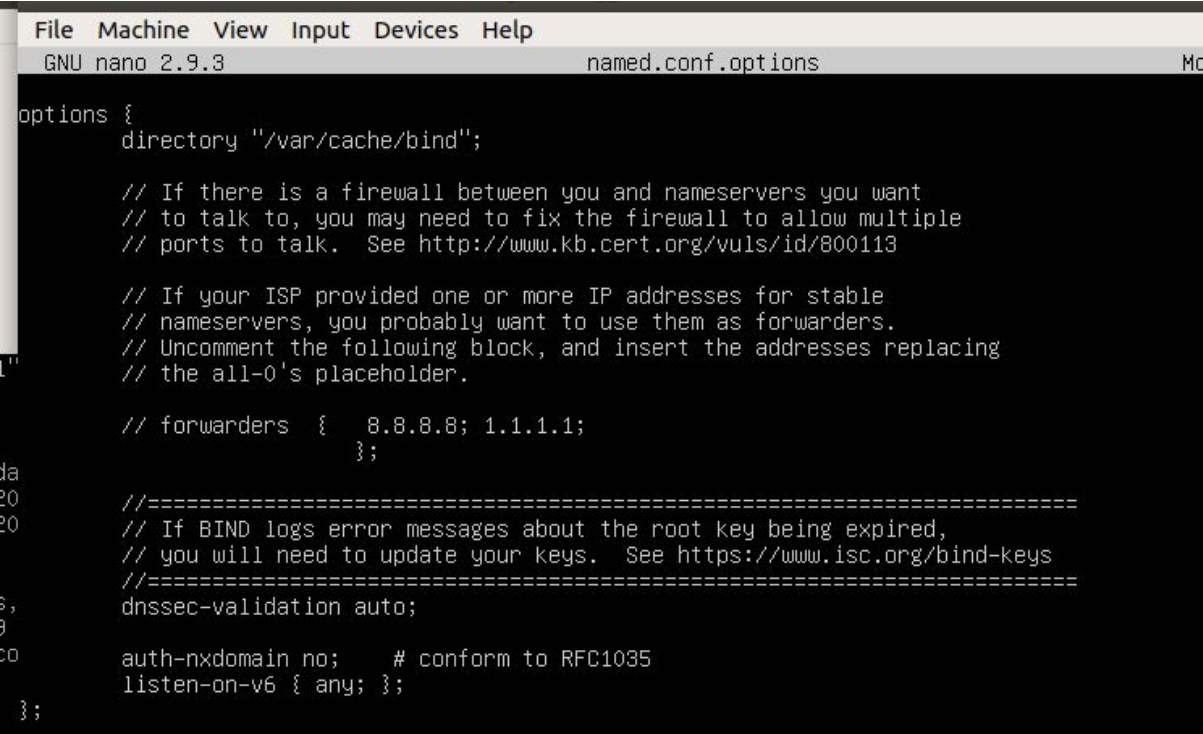
```

airon@dns2:~$ grep -E "Jun 10.*transferred serial" /var/log/syslog | cut -d " " -f 9,10 | sort -u |
wc -l
2
airon@dns2:~$ _

```

Por último vamos a ver cómo se configura como DNS principal el servidor, tendremos que añadir otros DNS como reenviadores ya que si no no tendríamos acceso a internet.

Para ello tenemos que ir al archivo `named.conf.options` y poner cuales vana ser los servidores reenviadores yo he utilizado el de google, podemos poner los que queramos que sepamos que funcionan bien, para resoluciones en internet.



```

File Machine View Input Devices Help
GNU nano 2.9.3 named.conf.options Mo

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; 1.1.1.1;
    // };

    //=====
    // 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;

    auth-nxdomain no;    # conform to RFC1035
    listen-on-v6 { any; };
};

```

Una vez que tenemos el problema de la resolución de nombres fuera de mi red solucionado por los reenviadores debemos ir a al archivo `/etc/resolv.conf` y cambiar para que nuestro servidor sea el que resuelve por defecto todas las consultas. Con poner la dirección ip del servidor que queremos funciona pero hay muchas más posibilidades de configuración recomendando mirar la ayuda de este fichero para entender más las opciones que ofrece.

```
GNU nano 2.9.3 /etc/resolv.conf
# This file is managed by man:systemd-resolved(8). Do not edit.
#
# This is a dynamic resolv.conf file for connecting local clients to the
# internal DNS stub resolver of systemd-resolved. This file lists all
# configured search domains.
#
# Run "systemd-resolve --status" to see details about the uplink DNS servers
# currently in use.
#
# Third party programs must not access this file directly, but only through the
# symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a different way,
# replace this symlink by a static file or a different symlink.
#
# See man:systemd-resolved.service(8) for details about the supported modes of
# operation for /etc/resolv.conf.

nameserver 192.168.42.112
options edns0
```

Con esto ya configurado en mi servidor maestro vamos a probar si de verdad funciona y la diferencia con la otra máquina que no tiene configurado este servidor ni el esclavo como servidores de resolución del sistema.

Aquí ya podremos hacer ping tanto a cualquier página de internet ya que hemos configurado forwarders como a las direcciones de nuestra red local que hemos configurado. Para ello vamos a probar a hacer ping a google y a www.aambite-example.org para ver que resuelve tanto interna como externa.

- ***ping -c 4 google.es***
- ***ping -c 4 aambite-example.org***

```
airon@dns1:/etc/bind$ ping -c 4 google.es
PING google.es (172.217.16.227) 56(84) bytes of data.
64 bytes from mad08s04-in-f3.1e100.net (172.217.16.227): icmp_seq=1 ttl=118 time=18.8 ms
64 bytes from mad08s04-in-f3.1e100.net (172.217.16.227): icmp_seq=2 ttl=118 time=13.9 ms
64 bytes from mad08s04-in-f3.1e100.net (172.217.16.227): icmp_seq=3 ttl=118 time=13.6 ms
64 bytes from mad08s04-in-f3.1e100.net (172.217.16.227): icmp_seq=4 ttl=118 time=22.0 ms

--- google.es ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 13.687/17.103/22.004/3.495 ms
airon@dns1:/etc/bind$ ping -c 4 aambite-example.org
PING aambite-example.org(ip6-localhost (::1)) 56 data bytes
64 bytes from ip6-localhost (::1): icmp_seq=1 ttl=64 time=0.032 ms
64 bytes from ip6-localhost (::1): icmp_seq=2 ttl=64 time=0.060 ms
64 bytes from ip6-localhost (::1): icmp_seq=3 ttl=64 time=0.055 ms
64 bytes from ip6-localhost (::1): icmp_seq=4 ttl=64 time=0.054 ms

--- aambite-example.org ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3069ms
rtt min/avg/max/mdev = 0.032/0.050/0.060/0.011 ms
airon@dns1:/etc/bind$ _
```

Como podemos ver en esta máquina hace la resolución de nombres correctamente pero si nos vamos a la otra que no tiene configurado el servidor no resolverá los de la red interna.

```
airon@dns2:~$ ping -c 4 google.es
PING google.es (216.58.211.227) 56(84) bytes of data:
64 bytes from mad01s24-in-f3.1e100.net (216.58.211.227): icmp_seq=1 ttl=118 time=83.3 ms
64 bytes from mad01s24-in-f3.1e100.net (216.58.211.227): icmp_seq=2 ttl=118 time=11.7 ms
64 bytes from mad01s24-in-f3.1e100.net (216.58.211.227): icmp_seq=3 ttl=118 time=20.9 ms
64 bytes from mad01s24-in-f3.1e100.net (216.58.211.227): icmp_seq=4 ttl=118 time=11.5 ms

--- google.es ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 11.528/31.884/83.339/29.951 ms
airon@dns2:~$ ping -c 4 aambite-example.org
ping: aambite-example.org: Name or service not known
airon@dns2:~$ _
```

Vemos cómo no resuelve las peticiones de nuestro dominio local porque no es el dns que está actuando por defecto en el sistema.

Veremos que si hacemos un nslookup y ponemos de servidor el de esta máquina si que resolverá la consulta.

```
airon@dns2:~$ nslookup
> server dns2.kessusa
Default server: dns2.kessusa
Address: 192.168.42.112#53
> aambite-example.org
Server:      dns2.kessusa
Address:     192.168.42.112#53

Name:   aambite-example.org
Address: 127.0.0.1
Name:   aambite-example.org
Address: ::1
>
```

Bueno con esto ya tenemos una información básica para poder usar un dns de forma local, aunque hay muchísimas más cosas que se pueden hacer y es recomendable empezar a utilizar DNSsec lo dejaré para un próximo documento.