

Laboratorio: Instalación de CentOS a partir de Debian

Requisitos previos

Material necesario:

- Máquina virtual con SO GNU/Linux
- ISO de instalación de CentOS

Instalación de paquetes:

- ntfs-3g
- xfsprogs

Planteamiento

Creamos un disco duro nuevo de VirtualBox y lo añadimos a una máquina que ya tengamos operativa. Esto nos va a permitir diseñar el esquema de particionado del disco desde nuestro SO con las herramientas que conocemos.

La idea es crear un esquema de particionado en el que dispongamos de 5 particiones con las siguientes características:

Partición	Tamaño	Sistema de ficheros	Etiqueta	Punto de montaje
1	3G	ext2	ARRANQUE	/boot
2	20G	xfs	SISTEMA	/
3	2G	Swap	SWAP	N/A
5	5G	xfs	PERSONAL	/home
6	10G	ext4	DATOS	/srv

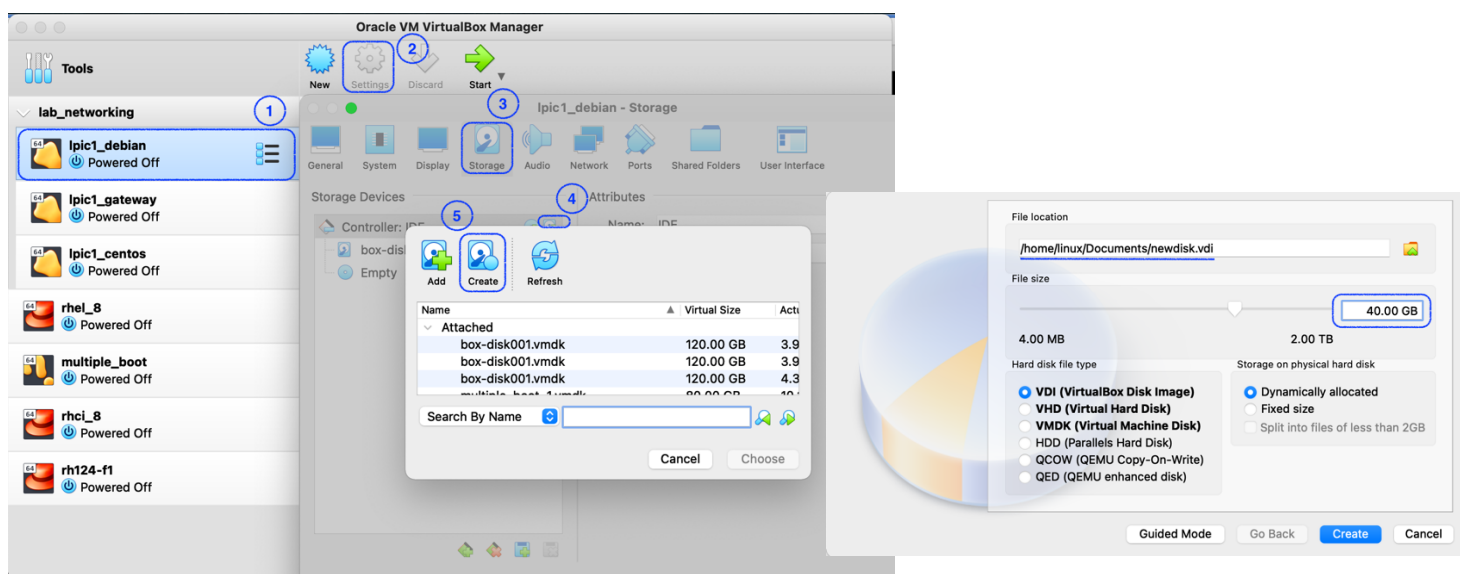
Una vez tenemos esta configuración de disco, lo desconectamos de la máquina en la que lo hemos puesto y creamos una nueva máquina con él y con la ISO de CentOS. El SO del CD deberá detectar nuestro disco y su estructura y permitirnos hacer la instalación según lo diseñado.

Información adicional para la instalación del SO

- Zona horaria: Europa/Madrid
- Configuración regional y de teclado: Catalán
- Idioma soportado: Inglés
- Nombre de host: centos.lab.local

Creación del disco duro con VirtualBox

Para poder crear un disco virtual nuevo la opción más rápida es agregarlo a una máquina ya existente. Para ello debemos abrir el menú “Settings” de la máquina, escoger la opción “Storage” y seleccionar la opción añadir nuevo disco en la controladora que ya tiene la MV, tal y como se ve en la siguiente imagen:



Particionado del disco

Una vez hemos creado el disco arrancamos la máquina y comprobamos que nuestro sistema lo detecta:

```
# fdisk -l

Disk /dev/sda: 120 GiB, 128849018880 bytes, 251658240 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x000a6b8b

Device      Boot      Start          End  Sectors  Size Id Type
/dev/sda1                2048   83888127  83886080   40G 83 Linux
/dev/sda2  *    83888128  146802687  62914560   30G 83 Linux
/dev/sda3        193064960  251656191  58591232   28G 83 Linux

Disk /dev/sdb: 40 GiB, 42949672960 bytes, 83886080 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

Vemos que tenemos el disco “/dev/sdb” con 40GB y sin particiones. Entramos en él y realizamos la configuración que queremos:

```
# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.33.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x4025ecb1.
```

Como vemos el sistema nos indica que este disco no dispone de ninguna tabla de particiones, por lo que necesitamos crear una (opción “o”). NOTA: Esta operación implica perder cualquier dato que haya en el disco

```
Command (m for help): o
Created a new DOS disklabel with disk identifier 0x92924fac.

Command (m for help): p
Disk /dev/sdb: 40 GiB, 42949672960 bytes, 83886080 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x92924fac
```

El siguiente paso es crear las particiones (opción “n”). Nos va a pedir el tipo de partición, su número y su principio y final (para calcular el tamaño).

Partición 1

```
Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-83886079, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-83886079, default 83886079): +3G

Created a new partition 1 of type 'Linux' and of size 3 GiB.
```

Partición 2

```
Command (m for help): n
Partition type
   p   primary (1 primary, 0 extended, 3 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (2-4, default 2): 2
First sector (6293504-83886079, default 6293504):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (6293504-83886079, default 83886079): +20G

Created a new partition 2 of type 'Linux' and of size 20 GiB.
```

Partición 3

```
Command (m for help): n
Partition type
   p   primary (2 primary, 0 extended, 2 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (3,4, default 3): 3
First sector (48236544-83886079, default 48236544):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (48236544-83886079, default 83886079): +2G

Created a new partition 3 of type 'Linux' and of size 2 GiB.
```

En este caso, al tratarse de la partición de swap, no nos interesa el ID que se la ha configurado (Linux), por lo que se lo cambiaremos (opción “t”)

```
Command (m for help): t
Partition number (1-3, default 3): 3
Hex code (type L to list all codes): 82

Changed type of partition 'Linux' to 'Linux swap / Solaris'.)
```

Partición 4

Para poder crear las particiones 5 y 6 (lógicas) necesitamos crear una partición especial de tipo “extendida” que las contenga. Para ello seguimos el mismo proceso:

```
Command (m for help): n
Partition type
   p   primary (3 primary, 0 extended, 1 free)
   e   extended (container for logical partitions)
Select (default e): e

Selected partition 4
First sector (52430848-83886079, default 52430848):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (52430848-83886079, default 83886079):

Created a new partition 4 of type 'Extended' and of size 15 GiB.
```

Partición 5

```
Command (m for help): n
All primary partitions are in use.
Adding logical partition 5
First sector (52432896-83886079, default 52432896):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (52432896-83886079, default 83886079): +5G

Created a new partition 5 of type 'Linux' and of size 5 GiB.
```

Partición 6

```
Command (m for help): n
All primary partitions are in use.
Adding logical partition 6
First sector (62920704-83886079, default 62920704):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (62920704-83886079, default 83886079):

Created a new partition 6 of type 'Linux' and of size 10 GiB
```

Una vez hechas todas las particiones comprobamos que sean correctas (opción “p”) y aplicamos los cambios (opción “w”)

```
Command (m for help): p
Disk /dev/sdb: 40 GiB, 42949672960 bytes, 83886080 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x92924fac

Device      Boot    Start        End    Sectors  Size Id Type
/dev/sdb1                2048    6293503    6291456    3G 83 Linux
/dev/sdb2           6293504    48236543   41943040   20G 83 Linux
/dev/sdb3           48236544   52430847    4194304    2G 82 Linux swap / Solaris
/dev/sdb4           52430848   83886079   31455232   15G  5 Extended
/dev/sdb5           52432896   62918655   10485760    5G 83 Linux
/dev/sdb6           62920704   83886079   20965376   10G 83 Linux

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

Si todo ha ido bien, el comando “fdisk -l /dev/sdb debería devolver la misma información

```
# fdisk -l /dev/sdb
Disk /dev/sdb: 40 GiB, 42949672960 bytes, 83886080 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x92924fac

Device      Boot    Start        End    Sectors  Size Id Type
/dev/sdb1                2048    6293503    6291456    3G 83 Linux
/dev/sdb2           6293504    48236543   41943040   20G 83 Linux
/dev/sdb3           48236544   52430847    4194304    2G 82 Linux swap / Solaris
/dev/sdb4           52430848   83886079   31455232   15G  5 Extended
/dev/sdb5           52432896   62918655   10485760    5G 83 Linux
/dev/sdb6           62920704   83886079   20965376   10G 83 Linux
```

Creación de sistemas de ficheros

Ahora nos falta aplicar el sistema de ficheros que queremos en cada partición. Para ello realizamos lo siguiente:

Partición 1

```
# mkfs.ext2 /dev/sdb1
mke2fs 1.44.5 (15-Dec-2018)
Creating filesystem with 786432 4k blocks and 196608 inodes
Filesystem UUID: 7ff3ad8e-2dcc-4ca5-9ab3-0e4ed1a83a37
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done
```

Partición 2

```
root@debian:~# mkfs.xfs /dev/sdb2
meta-data=/dev/sdb2            isize=512    agcount=4, agsize=1310720 blks
                               =          sectsz=512   attr=2, projid32bit=1
                               =          crc=1        finobt=1, sparse=1, rmapbt=0
                               =          reflink=0
data      =                    bsize=4096   blocks=5242880, imaxpct=25
                               =          sunit=0      swidth=0 blks
naming    =version 2           bsize=4096   ascii-ci=0, ftype=1
log       =internal log        bsize=4096   blocks=2560, version=2
                               =          sectsz=512   sunit=0 blks, lazy-count=1
realtime  =none                extsz=4096   blocks=0, rtextents=0
```

Partición 3

```
# mkswap /dev/sdb3
Setting up swapspace version 1, size = 2 GiB (2147479552 bytes)
no label, UUID=238e79b0-c3e8-4e03-987c-27de77545114
```

Partición 5

```
# mkfs.xfs /dev/sdb5
meta-data=/dev/sdb5            isize=512    agcount=4, agsize=327680 blks
                               =          sectsz=512   attr=2, projid32bit=1
                               =          crc=1        finobt=1, sparse=1, rmapbt=0
                               =          reflink=0
data      =                    bsize=4096   blocks=1310720, imaxpct=25
                               =          sunit=0      swidth=0 blks
naming    =version 2           bsize=4096   ascii-ci=0, ftype=1
log       =internal log        bsize=4096   blocks=2560, version=2
                               =          sectsz=512   sunit=0 blks, lazy-count=1
realtime  =none                extsz=4096   blocks=0, rtextents=0
```

Partición 6

```
# mkfs.ext4 /dev/sdb6
```

```
mke2fs 1.44.5 (15-Dec-2018)
```

```
Creating filesystem with 2620672 4k blocks and 655360 inodes
```

```
Filesystem UUID: 527b40bd-38a1-4b31-b962-a9ba60b1eba6
```

```
Superblock backups stored on blocks:
```

```
32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632
```

```
Allocating group tables: done
```

```
Writing inode tables: done
```

```
Creating journal (16384 blocks): done
```

```
Writing superblocks and filesystem accounting information: done
```

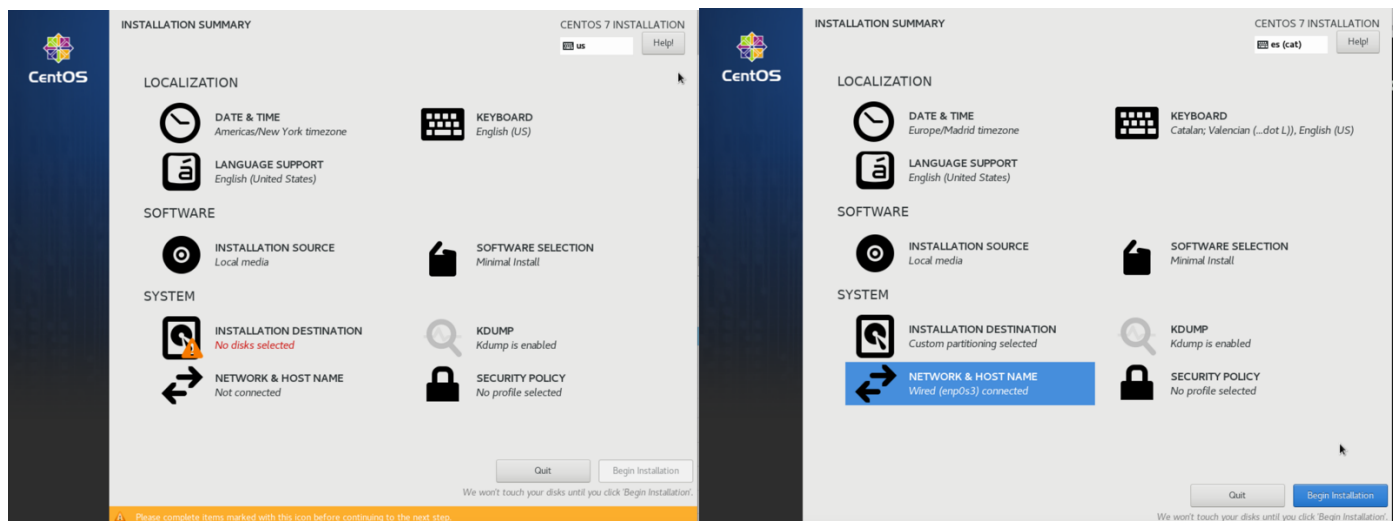
Instalación de CentOS

Paramos la MV y creamos una nueva usando el disco que acabamos de particionar. Ponemos el CD de instalación de CentOS y escogemos la opción "Install CentOS 7"

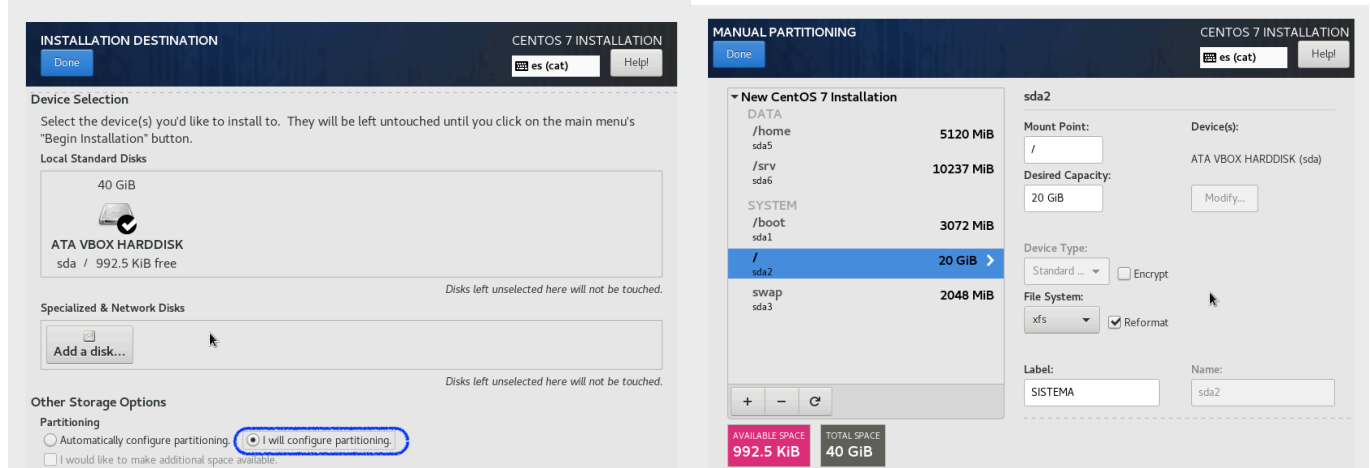
Con los datos indicados anteriormente, cumplimenta los diferentes apartados para conseguir este aspecto:

ANTES

DEPUÉS



Para configurar el apartado "Information Destination" usa como referencia las siguientes imágenes:



El siguiente paso será comenzar la instalación y el asistente nos pide la contraseña de root y la creación de una cuenta de administrador. Configura los datos que consideres.

Si la instalación ha ido bien deberíamos de ver el siguiente resultado:

```
# # df -hT -t ext2 -t ext4 -t xfs
Filesystem      Type  Size  Used Avail Use% Mounted on
/dev/sda2       xfs   20G   1.2G   19G    6% /
/dev/sda5       xfs   5.0G    33M  5.0G    1% /home
/dev/sda1       ext2   3.0G  103M   2.8G    4% /boot
/dev/sda6       ext4   9.8G    37M   9.2G    1% /srv
```

Personalización de /etc/fstab

Como sabemos podemos hacer referencia a las particiones que montamos en el inicio del sistema de diferentes maneras. Para probarlo modifica el fichero /etc/fstab y configúralo de la siguiente manera

- Partición 1: UUID
- Partición 2: UUID
- Partición 5: /dev
- Partición 6: LABEL

El resultado final debería ser algo como esto:

```
# cat /etc/fstab
#
# /etc/fstab
# Created by anaconda on Mon Mar 15 17:04:36 2021
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=e63e024b-43e6-4fb1-9bb6-33beb0eb4242 / xfs defaults 0 0
UUID=286f639c-8484-47d6-9e99-99ea48e4d6ef /boot ext2 defaults 1 2
/dev/sda5 /home xfs defaults 0 0
LABEL=DATOS /srv ext4 defaults 1 2
UUID=8dadcf50-be7d-4b23-94e9-f57cd9012b9f swap swap defaults 0 0
```

NOTA: Considera hacer una copia del fichero original /etc/fstab antes de aplicar las modificaciones

Si reiniciamos la máquina y revisamos veremos que todas las particiones han sido montadas correctamente.

```
# df -hT -t ext2 -t ext4 -t xfs
Filesystem      Type  Size  Used Avail Use% Mounted on
/dev/sda2       xfs   20G   1.2G   19G    6% /
/dev/sda5       xfs   5.0G    33M  5.0G    1% /home
/dev/sda1       ext2   3.0G  103M   2.8G    4% /boot
/dev/sda6       ext4   9.8G    37M   9.2G    1% /srv
```