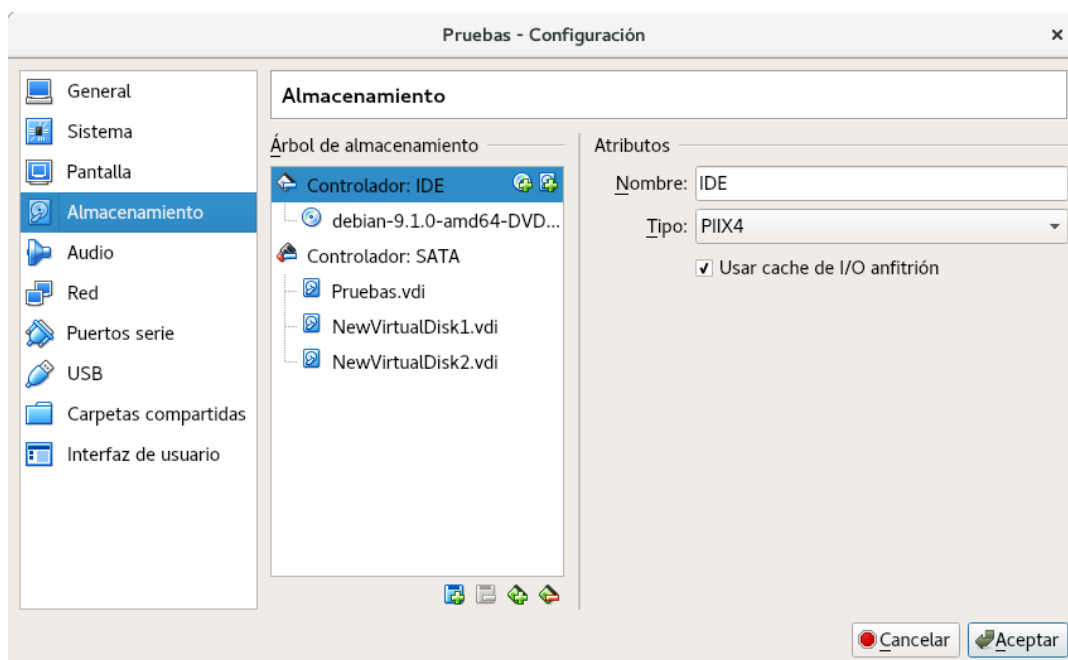


## Actividad con Btrfs

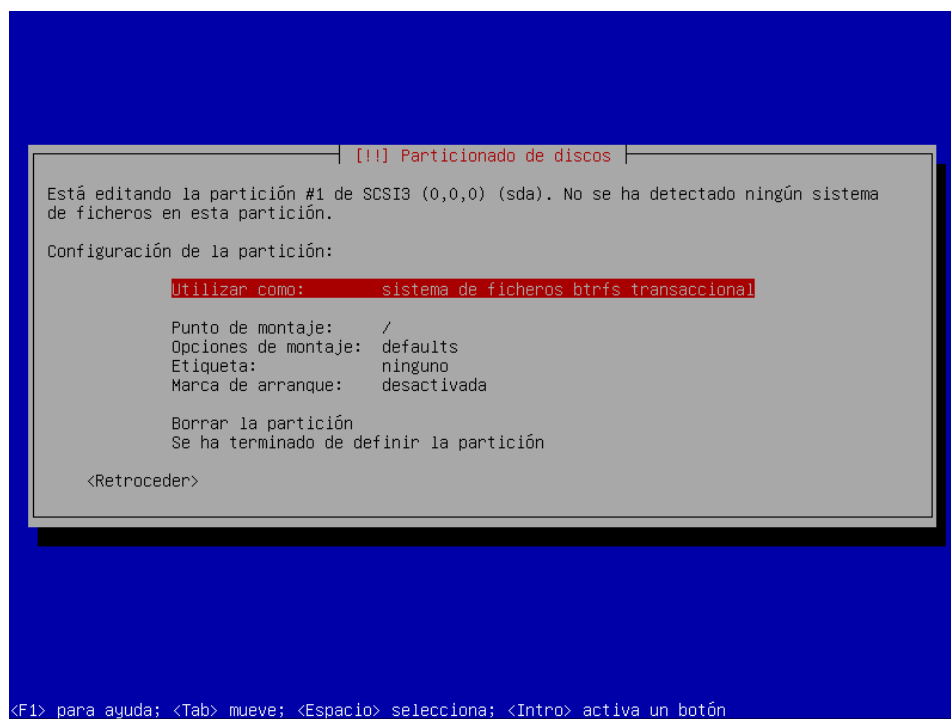
En esta actividad probaremos varias de las características mas interesantes de Btrfs, a través de pequeños ejercicios

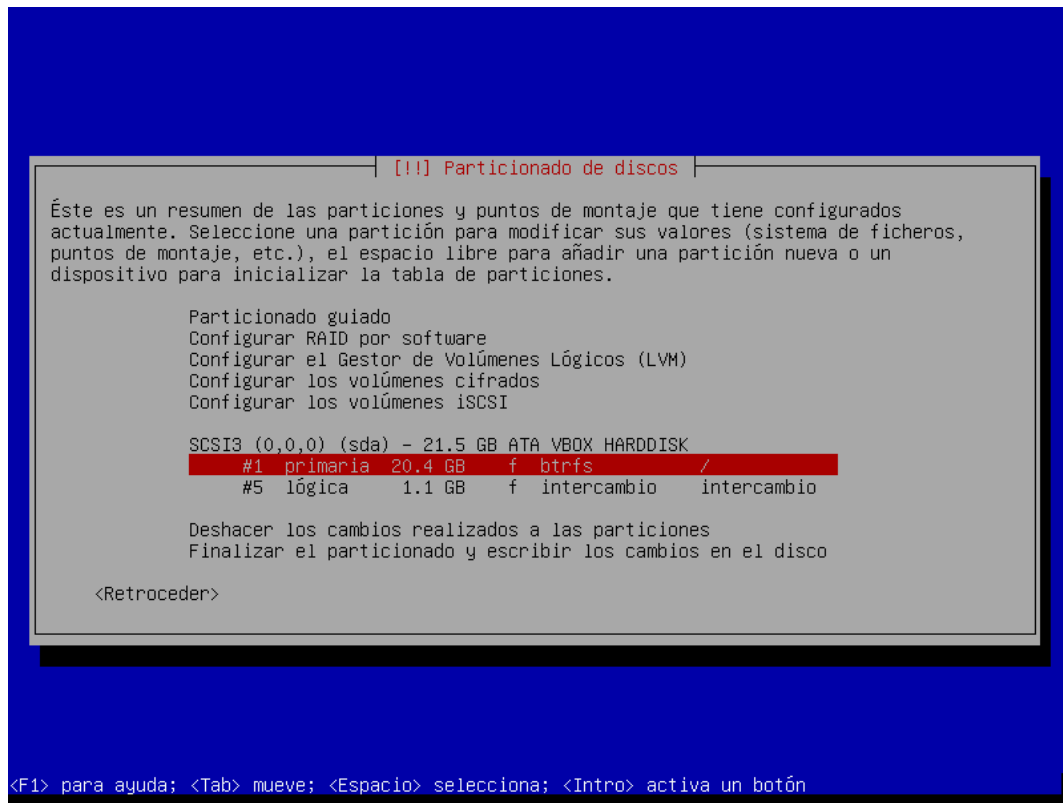
### 1. Realizar una nueva instalación de una máquina virtual, usando como sistema de ficheros Btrfs

Creamos una nueva máquina virtual, con varios discos duros.



Dado que CentOS 6 no soporta Btrfs, usaremos Debian 9 para la prueba. Durante el particionado, indicaremos Btrfs como sistema de ficheros de la partición raíz





Una vez terminada la instalación, arrancamos la máquina, y comprobamos el estado de los discos duros con `lsblk`

```
pruebas@debian:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0   20G  0 disk
├─sda1       8:1    0   19G  0 part /
├─sda2       8:2    0    1K  0 part
└─sda5       8:5    0 1022M  0 part [SWAP]
sdb          8:16    0    1G  0 disk
sdc          8:32    0   20G  0 disk
sr0         11:0    1 1024M  0 rom
pruebas@debian:~$ _
```

## 2. En un nuevo disco duro, formatearlo como ext4, convertirlo a Btrfs, y deshacer la conversión

Usamos el segundo disco que creamos en la máquina virtual, de 1GB de capacidad

- **Creando la partición**

Creamos la partición con `fdisk`, ocupando todo el espacio disponible en el disco

```
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.
Tenga cuidado antes de utilizar la orden de escritura.

El dispositivo no contiene una tabla de particiones reconocida.
Se ha creado una nueva etiqueta de disco DOS con el identificador de disco 0x06a
6bfcb.

Orden (m para obtener ayuda): n
Tipo de partición
  p   primaria (0 primaria(s), 0 extendida(s), 4 libre(s))
  e   extendida (contenedor para particiones lógicas)
Seleccionar (valor predeterminado p): p
Número de partición (1-4, valor predeterminado 1):
Primer sector (2048-2097151, valor predeterminado 2048):
Último sector, +sectores o +tamaño{K,M,G,T,P} (2048-2097151, valor predeterminad
o 2097151):

Crea una nueva partición 1 de tipo 'Linux' y de tamaño 1023 MiB.

Orden (m para obtener ayuda): w
Se ha modificado la tabla de particiones.
Llamando a ioctl() para volver a leer la tabla de particiones.
Se están sincronizando los discos.

root@debian:/home/pruebas# _
```

- **Formateando como ext4**

Usamos `mkfs` para formatear la partición como ext4

```
root@debian:/home/pruebas# mkfs -t ext4 /dev/sdb1
mke2fs 1.43.4 (31-Jan-2017)
Se está creando un sistema de ficheros con 261888 bloques de 4k y 65536 nodos-i
UUID del sistema de ficheros: c5dfbebd-b23b-4bcb-8a20-bf8e01781039
Respaldo del superbloque guardado en los bloques:
    32768, 98304, 163840, 229376

Reservando las tablas de grupo: hecho
Escribiendo las tablas de nodos-i: hecho
Creando el fichero de transacciones (4096 bloques): hecho
Escribiendo superbloques y la información contable del sistema de ficheros: hech
o

root@debian:/home/pruebas# _
```

- **Montando la partición y creando ficheros**

Para poder comprobar si las siguientes pruebas son correctas, montamos la partición y creamos varios ficheros en ella

```
root@debian:/home/pruebas# mount /dev/sdb1 /mnt
root@debian:/home/pruebas# echo "fichero1" > /mnt/f1
root@debian:/home/pruebas# echo "fichero2" > /mnt/f2
root@debian:/home/pruebas# ls /mnt
f1 f2 lost+found
root@debian:/home/pruebas# _
```

- **Convirtiendo a Btrfs**

Una vez creados los ficheros, convertimos la partición usando el comando `btrfs-convert [partición]`

Para ello, previamente desmontamos la partición con `umount`

```
root@debian:/home/pruebas# umount /dev/sdb1
root@debian:/home/pruebas# btrfs-convert /dev/sdb1
create btrfs filesystem:
    blocksize: 4096
    nodesize: 16384
    features:  extref, skinny-metadata (default)
creating ext2 image file
creating btrfs metadatacopy inodes [o] [          4/          13]
conversion complete
root@debian:/home/pruebas# _
```

- **Comprobando la conversión**

Usamos `fdisk` para mostrar el sistema de ficheros actual de la partición

```
root@debian:/home/pruebas# fdisk /dev/sdb1

Bienvenido a fdisk (util-linux 2.29.2).
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.
Tenga cuidado antes de utilizar la orden de escritura.

El dispositivo /dev/sdb1 ya tiene una firma btrfs.
La firma se borrará mediante una orden de escritura.

El dispositivo no contiene una tabla de particiones reconocida.
Se ha creado una nueva etiqueta de disco DOS con el identificador de disco 0x52f12567.

Orden (m para obtener ayuda): _
```

Una vez comprobada que la partición es btrfs, la montamos y comprobamos su contenido

```
root@debian:/home/pruebas# mount /dev/sdb1 /mnt
root@debian:/home/pruebas# ls /mnt
ext2_saved  f1  f2  lost+found
root@debian:/home/pruebas# ls /mnt/ext2_saved/
image
root@debian:/home/pruebas# ls /mnt/ext2_saved/image
/mnt/ext2_saved/image
root@debian:/home/pruebas# ls /mnt/ext2_saved/image _
```

Vemos que, aparte de los dos ficheros que creamos anteriormente, hay un directorio que contiene la imagen del anterior sistema de ficheros ext4

- **Deshaciendo la conversión**

Deshacemos la conversión de la partición recién convertida, aplicando la opción `-r` a `btrfs-convert`

Deshecha la conversión, volvemos a montar la partición y observamos su contenido

```
root@debian:/home/pruebas# umount /mnt
root@debian:/home/pruebas# btrfs-convert -r /dev/sdb1
rollback complete
root@debian:/home/pruebas# mount /dev/sdb1 /mnt
root@debian:/home/pruebas# ls /mnt
f1  f2  lost+found
root@debian:/home/pruebas# _
```

Vemos que el directorio con la imagen de ext4 ha desaparecido

### **3. Añadir el nuevo disco duro al sistema de ficheros principal y configurarlo como RAID**

Ahora, con el tercer disco duro, vamos a probar la creación de un RAID, incorporando el nuevo disco duro al sistema Btrfs de la partición raíz.

Para ello, seguimos los siguientes pasos:

1. Añadimos el dispositivo a Btrfs, usando el comando  
`btrfs device add [dispositivo] [punto de montaje]`
2. Escaneamos el sistema de ficheros, con:  
`btrfs device scan`
3. Activamos el balanceador de carga en el punto de montaje  
`btrfs filesystem balance [punto de montaje]`

Ejecutamos los comandos en la máquina. Usaremos el segundo disco duro, cuya unidad es sdc

```
root@debian:/home/pruebas# btrfs device add /dev/sdc /mnt
root@debian:/home/pruebas# btrfs device scan
Scanning for Btrfs filesystems
root@debian:/home/pruebas# btrfs filesystem balance /mnt
WARNING:

    Full balance without filters requested. This operation is very
    intense and takes potentially very long. It is recommended to
    use the balance filters to narrow down the balanced data.
    Use 'btrfs balance start --full-balance' option to skip this
    warning. The operation will start in 10 seconds.
    Use Ctrl-C to stop it.
10 9 8 7 6 5 4 3 2 1
Starting balance without any filters.

Done, had to relocate 4 out of 4 chunks
root@debian:/home/pruebas#
root@debian:/home/pruebas# _
```

El balanceador nos indica que ha reubicado 4 bloques

Usando el balanceador, iniciamos la conversión a RAID 1, con el comando:

```
btrfs balance start -dconvert=raid[tipo] -mconvert=raid[tipo]
[punto_de_montaje]
```

```
root@debian:/home/pruebas# btrfs balance start -dconvert=raid1 -mconvert=raid1 /
mnt

Done, had to relocate 3 out of 3 chunks
root@debian:/home/pruebas#
```

```
pruebas@debian:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0   20G  0 disk
├─sda1       8:1    0   19G  0 part /
├─sda2       8:2    0    1K  0 part
└─sda5       8:5    0 1022M  0 part [SWAP]
sdb          8:16   0    1G  0 disk
└─sdb1       8:17   0 1023M  0 part
sdc          8:32   0    2G  0 disk
sr0         11:0    1 1024M  0 rom
```

Vemos un nuevo dispositivo, correspondiente al RAID