

Practica 3

File Systems

1. ¿Qué es un file system?

Es una estructura jerárquica (árbol de archivos) de archivos y directorios que se parece a un árbol invertido con las raíces arriba y las ramas abajo. Utiliza directorios para organizar los datos y los programas en grupos, lo que permite la gestión de varios directorios y archivos a la vez. Permite la creación, eliminación, modificación y búsqueda de archivos y su organización en directorios. Básicamente es la parte del SO que se encarga del manejo de los archivos.

2. Describa las principales diferencias y similitudes entre los file systems: ext(2,3,4) y XFS.

Característica	Ext2	Ext3	Ext4	XFS
Tamaño máximo archivo individual	2TB	2TB	16TB	8EB
Tamaño máximo File System	32TB	32TB	1EB	16EB
Cantidad máxima subdirectorios	32,000	32000	64000 (extensible, pudiendo ser ilimitado)	Ilimitado

<i>Tamaño del inodo por defecto</i>	128 bytes	128 bytes	256 bytes	512 bytes
<i>Desfragmentación</i>	No	No	Sí	Sí
<i>Indexación de directorios</i>	Deshabilitado	Deshabilitado	Activado	Activado
<i>Delay allocation</i>	No	No	Sí	Sí
<i>Multiblock allocation</i>	No	No	Sí	No
<i>Journaling</i>	No	Si	Si (se puede desactivar)	Si
<i>Extents</i>	No	No	Si	Si
<i>Nombre del archivo se almacena en el inodo</i>	No	No	No	No
<i>Reducir tamaño File System</i>	Si	Si	Si	No

3. En ext4, describa las siguientes características: extents, multiblock allocation, delay allocation y persistent pre-allocation (<https://kernelnewbies.org/Ext4>).

- Extents:
 - o Un extents es un conjunto de bloques físicos contiguos que indican la ubicación de los datos. Por ejemplo, un archivo de 100 MB puede

asignarse a una única extensión en lugar de 25600 bloques individuales.

- Los archivos grandes se dividen en múltiples extents.
- Las extents mejoran el rendimiento y reducen la fragmentación al fomentar diseños continuos en el disco.

- Multiblock allocation:

- Se asignan muchos bloques en una sola llamada, en lugar de un solo bloque por llamada (mballoc asignador multibloque)
- Aumenta la eficiencia y el rendimiento, especialmente útil con asignaciones y extensiones retrasadas.
- No afecta el formato del disco.

- Delay allocation:

- Característica de rendimiento presente en sistemas de archivos modernos como Ext4, XFS, ZFS, btrfs y Reiser 4.
- Posterga la asignación de bloques tanto como sea posible.
- Tradicionalmente, los sistemas de archivos asignan bloques inmediatamente cuando un proceso escribe datos, incluso si estos datos se mantienen en caché por un tiempo. Delay allocation no asigna bloques de inmediato, sino que retrasa la asignación mientras los datos están en caché, hasta que realmente se escriben en el disco.

- Persistent pre-allocation:

- Característica disponible en Ext3 en las últimas versiones del kernel y emulada por glibc en otros sistemas de archivos.
- Permite a las aplicaciones reservar espacio en disco antes de necesitarlo realmente.
- Las aplicaciones indican al sistema de archivos que preasigne el espacio, y este reserva los bloques y estructuras de datos correspondientes.
- No hay datos en estas áreas preasignadas hasta que la aplicación los escriba en el futuro.

- Esta función es similar a lo que hacen las aplicaciones P2P al reservar espacio para descargas futuras, pero implementada de manera más eficiente a través del sistema de archivos y con una API genérica.
- Tiene varios usos:
 - Evita que las aplicaciones llenen archivos con ceros de manera ineficiente.
 - Mejora la fragmentación al asignar bloques de manera contigua
 - Garantiza que las aplicaciones siempre tengan el espacio necesario..
- Importante para aplicaciones en tiempo real, ya que sin preasignación, el sistema de archivos podría quedarse sin espacio en medio de una operación crítica.
- Esta función se accede a través de la interfaz libc `posix_fallocate()`.

4. ¿Es siempre necesario tener un file system para acceder a un disco o partición?

No, no siempre es necesario tener un sistema de archivos para acceder a un disco o partición. Algunos sistemas operativos y aplicaciones pueden acceder a un disco o partición sin un sistema de archivos, tratándolo como un bloque de datos sin estructura específica.

5. ¿Qué es el área de swap en Linux? ¿Existe un área similar en Windows?

El área de swap es una partición especial del disco duro o un archivo reservado que se utiliza como memoria virtual. Cuando la RAM física de un sistema se agota, el sistema operativo Linux puede transferir datos de la RAM a esta área de swap para liberar espacio en la memoria RAM y mantener el sistema en funcionamiento. En Windows, existe un concepto similar llamado "archivo de paginación" o "archivo de intercambio". Al igual que en Linux, este archivo se utiliza como memoria virtual cuando la RAM física se agota. Windows utiliza el archivo de paginación para realizar el mismo propósito que el área de swap en Linux: liberar espacio en la memoria RAM moviendo datos menos utilizados a un archivo en el disco duro.

6. ¿Qué función cumple el directorio lost+found en Linux?

Almacena los archivos que se han encontrado en el sistema de archivos durante el proceso de verificación de integridad o reparación del sistema de archivos.

Cuando el sistema de archivos se corrompe o se daña de alguna manera, el sistema operativo puede intentar reparar el sistema de archivos durante el próximo inicio. Durante este proceso de reparación, el sistema de archivos puede encontrar archivos que no están asociados con ningún directorio o que tienen algún otro tipo de problema de integridad.

Estos archivos "perdidos" se pueden recuperar (tras una revisión del file system a través de la herramienta fsck) y mover al directorio "lost+found" para que el administrador del sistema pueda revisarlos manualmente y decidir qué hacer con ellos.

7. En Linux, ¿dónde se almacenan el nombre y los metadatos de los archivos?

El nombre del fichero se almacena en el inodo de la carpeta padre y los metadatos en sus propios inodos.

8. Seleccione una de sus particiones y conteste usando el comando dumpe2fs (si está usando la MV de la cátedra, como root, dumpe2fs /dev/sda1):

- ¿Qué información muestra el comando dumpe2fs?

Muestra la configuración y estado del sistema de archivos de la partición deseada. Imprime la información del grupo de superbloques y bloques para el sistema de archivos presente en la partición seleccionada.

- ¿Cuál es el tamaño de bloque del file system?

El tamaño de bloque es 4096. Hay 3681792 bloques en total.

- ¿Cuántos inodos en total contiene el file system? ¿Cuántos archivos más se podrían crear con el estado actual del file system?

Hay 922080 inodos en total. Hay 619524 inodos libres, por lo que se podrían crear esa cantidad de archivos.

- ¿Cuántos grupos de bloques existen?

Hay 112 grupos

- ¿Cómo haría para incrementar la cantidad de inodos de un file system?
- Crear de un nuevo file system con más inodos: es la opción más segura. Se debería realizar una copia de seguridad de los datos, crear un file system con la cantidad deseada de inodos y después restaurar los datos en el nuevo file system.
- Usar herramientas de ajuste avanzado: existen herramientas avanzadas que permiten ajustar la cantidad de inodos en un file system existente, como `resize2fs` o `tune2fs`.
 - Estas herramientas tienen limitaciones y riesgos asociados. Es recomendable leer cuidadosamente la documentación y realizar pruebas en un entorno de prueba antes de aplicar cambios en producción.
- Convertir el file system a otro: aunque no aumentará directamente la cantidad de inodos en el file system, convertirlo a otro tipo que permita más inodos podría ser una solución alternativa. Esto también conlleva riesgos:
 - Iniciar la PC desde un Live USB.
 - Utilizar el comando `lsblk` para listar las particiones.
 - Crear las carpetas `/mnt/fuente` y `/mnt/destino` con `mkdir`.
 - Montar la partición a aumentar en `/mnt/fuente` y la que almacenará temporalmente una copia de los datos en `/mnt/destino` con `mount`.
 - Copiar todos los archivos de `/mnt/fuente` a `/mnt/destino`, utilizando `cp` o `rsync`.
 - Desmontar `/mnt/fuente` con `umount`.
 - Crear una nueva partición con más inodos, reemplazando el sistema de archivos existente con `mkfs`.

- Montar esta partición en /mnt/fuente con mount.
- Copiar todos los archivos de /mnt/destino a /mnt/fuente, utilizando cp o rsync.
- Reiniciar la PC, arrancando el sistema operativo original.
- <https://www.alibabacloud.com/help/en/ecs/use-cases/resolve-the-issue-of-insufficient-disk-space-on-a-linux-instance#c9cce4b36a6je>

9. ¿Qué es el file system procfs? ¿Y el sysfs?

Procfs es un pseudo file system montando en el directorio /proc que reside en la memoria RAM. Este presenta información sobre procesos y otra información del sistema en una estructura jerárquica de “files”. Por cada proceso se tiene un directorio. La mayoría de los “files” son solo lectura, aunque algunos pueden ser modificados.

Con el paso del tiempo, /proc se convirtió en un desorden. Como solución a esto se usa sysfs, que es como procfs pero mejor organizado. Sysfs está montado en /sys y exporta información sobre varios subsistemas del kernel, dispositivos de hardware y sus controladores (drivers), módulos cargados, etc. desde el espacio del kernel hacia el espacio del usuario.

Ambos file systems conviven para mantener retrocompatibilidad.

10. Usando el directorio /proc, contestar:

- ¿Cuál es la versión de SO que tiene instalado?

```
cat /proc/version
```

Se tiene la version 6.8.0

- ¿Cuál es procesador de su máquina?

```
cat /proc/cpuinfo | grep "model name"
```

AMD Ryzen 3 3200G with Radeon Vega Graphics

- ¿Cuánta memoria RAM disponible tiene?

```
cat /proc/meminfo | grep "MemFree"
```

6608508 KiB → 6.3 GiB

- ¿Qué archivo debería consultar si se quiere ver el mismo resultado que el comando lsmod?

Se debería usar cat /proc/modules

11. Usando el comando stat, contestar

- ¿Cuándo fue la última vez que se modificó el archivo /etc/group?

```
File: /etc/group
Size: 879          Blocks: 8          IO Block: 4096   regular file
Device: 8,1      Inode: 130894       Links: 1
Access: (0644/-rw-r--r--)  Uid: (   0/   root)   Gid: (   0/   root)
Access: 2024-05-04 13:37:04.339999932 -0300
Modify: 2024-04-01 14:14:57.199999998 -0300
Change: 2024-04-01 14:14:57.267999998 -0300
Birth: 2024-04-01 14:14:57.199999998 -0300
```

La ultima vez que se modifiko fue el 01/04/2024 a las 14:14 hs.

- ¿Cuál es la diferencia entre los datos Cambio (Change) y Modificación (Modify)?

Modify – la última vez que el archivo fue modificado (el contenido)

Change – la última vez que los metadatos del archivo fueron modificados (ejemplo, los permisos).

- ¿Cuál es el inodo que ocupa? ¿Cuántos bloques ocupa?

Ocupa el inodo 130894. Ocupa 8 bloques.

- ¿Qué número de inodo ocupa el directorio raíz?

```
root@SistemasOperativos:~# stat /
  File: /
  Size: 4096          Blocks: 8          IO Block: 4096   directory
Device: 8,1    Inode: 2              Links: 19
Access: (0755/drwxr-xr-x)  Uid: (   0/   root)   Gid: (   0/   root)
Access: 2024-04-15 21:09:54.511999993 -0300
Modify: 2024-04-15 21:09:54.427999993 -0300
Change: 2024-04-15 21:09:54.427999993 -0300
 Birth: 2024-04-01 14:07:23.000000000 -0300
```

Ocupa el inodo 2.

- ¿Es posible conocer la fecha de creación de un file en ext4? ¿Cómo lo haría?

Si, a través del timestamp mostrado en “Birth” con el comando “stat”

12. ¿Qué es un link simbólico? ¿En qué se diferencia de un hard-link?

Link simbólico: apunta otro archivo o directorio mediante su ruta absoluta o relativa. Si el archivo original se elimina, el enlace simbólico queda roto. Cada enlace simbólico dispone de su propio número de inodo y es diferente al del archivo original. Se pueden crear para directorios.

Hard-link: entrada de directorio adicional que apunta al mismo nodo de i-nodo (estructura de datos que almacena información sobre un archivo) que el archivo original. Si se elimina el archivo original, el hard-link sigue apuntando al contenido del archivo. El espacio del disco no se libera hasta que todos los enlaces duros se eliminan. No se puede crear para directorios.

13. Al crear un hard-link, ¿se ocupa un nuevo inodo? ¿Y con un link simbólico?

Explicado en punto 12.

14. Si se tiene un archivo llamado prueba.txt y se le genera un link simbólico, ¿qué sucede con el link simbólico si se elimina el archivo prueba.txt? ¿Y si el link fuese hard-link? (Ver el comando ln para la creación de links)

Si se genera un link simbólico y se elimina el archivo prueba.txt el enlace simbólico queda roto (explicado en punto 12). Si fuera un hard-link, este seguirá apuntando al contenido de prueba.txt (explicado en punto 12).

```
agusnfr@SistemasOperativos:~/kernel$ rm prueba.txt
agusnfr@SistemasOperativos:~/kernel$ ls -l
total 145844
-rw-r--r-- 1 agusnfr agusnfr 115343360 Apr  1 15:26 btrfs.image
drwxr-xr-x 26 agusnfr agusnfr    4096 Apr 15 20:30 linux-6.7
-rw-r--r-- 1 agusnfr agusnfr 141406528 Apr  1 14:36 linux-6.7.tar.xz
-rw-r--r-- 1 agusnfr agusnfr   6929868 Apr  1 14:36 patch-6.8.xz
lrwxrwxrwx 1 agusnfr agusnfr      10 May  4 14:45 pruebaDirecto -> prueba.txt
```

15. Crear un archivo llamado prueba2.txt. Si ahora se genera un hard-link sobre ese archivo llamado pruebahd.txt, ¿cómo se refleja la creación de ese hard-link?

```
agusnfr@SistemasOperativos:~/kernel$ ln prueba2.txt pruebaHard
agusnfr@SistemasOperativos:~/kernel$ ls -l
total 145844
-rw-r--r-- 1 agusnfr agusnfr 115343360 Apr  1 15:26 btrfs.image
drwxr-xr-x 26 agusnfr agusnfr    4096 Apr 15 20:30 linux-6.7
-rw-r--r-- 1 agusnfr agusnfr 141406528 Apr  1 14:36 linux-6.7.tar.xz
-rw-r--r-- 1 agusnfr agusnfr   6929868 Apr  1 14:36 patch-6.8.xz
-rw-r--r-- 2 agusnfr agusnfr      0 May  4 14:48 prueba2.txt
-rw-r--r-- 2 agusnfr agusnfr      0 May  4 14:48 pruebaHard
agusnfr@SistemasOperativos:~/kernel$ echo "hola" > pruebaHard
agusnfr@SistemasOperativos:~/kernel$ cat p
patch-6.8.xz prueba2.txt pruebaHard
agusnfr@SistemasOperativos:~/kernel$ cat pruebaHard
hola
agusnfr@SistemasOperativos:~/kernel$ cat prueba2.txt
hola
agusnfr@SistemasOperativos:~/kernel$
```

Se refleja la creación como si fueran 2 ficheros separados.

16. Elimine el archivo prueba2.txt, ¿es posible acceder al archivo pruebahd.txt? ¿Cómo se refleja la eliminación de ese archivo?

```

agusnfr@SistemasOperativos:~/kernel$ rm prueba2.txt
agusnfr@SistemasOperativos:~/kernel$ ls -l
total 145848
-rw-r--r-- 1 agusnfr agusnfr 115343360 Apr  1 15:26 btrfs.image
drwxr-xr-x 26 agusnfr agusnfr      4096 Apr 15 20:30 linux-6.7
-rw-r--r-- 1 agusnfr agusnfr 141406528 Apr  1 14:36 linux-6.7.tar.xz
-rw-r--r-- 1 agusnfr agusnfr  69298688 Apr  1 14:36 patch-6.8.xz
-rw-r--r-- 1 agusnfr agusnfr      5 May  4 14:50 pruebaHard

```

Sigue siendo válido el enlace, por lo tanto se puede seguir accediendo al archivo.

RAID

1. ¿Qué es un RAID? Explique las diferencias entre los distintos niveles de RAID

Es una técnica que permite usar múltiples discos en forma conjunta con el fin de construir un sistema de discos más rápido, más grande y confiable.

Nivel	Confiabilidad	Rendimiento	Disponibilidad
RAID 0	<ul style="list-style-type: none"> No proporciona tolerancia a fallos. 	<ul style="list-style-type: none"> Mejora la tasa de transferencia y el tiempo de acceso a los datos. 	<ul style="list-style-type: none"> El sistema deja de funcionar si hay una unidad de disco en falla.
RAID 1	<ul style="list-style-type: none"> Protege la información en caso de falla. 	<ul style="list-style-type: none"> Mejora la lectura de los datos. 	<ul style="list-style-type: none"> Evita interrupciones por fallas en las unidades.
RAID 2	<ul style="list-style-type: none"> El uso del código Hamming permite detectar y corregir errores. 	<ul style="list-style-type: none"> Mejora la operación de aplicaciones con alta tasa de transferencia. 	<ul style="list-style-type: none"> Usa múltiples discos dedicados que permiten redundancia de datos.
RAID 3	<ul style="list-style-type: none"> El disco de paridad permite reconstruir la información. 	<ul style="list-style-type: none"> Elevada tasa de transferencias secuenciales. 	<ul style="list-style-type: none"> Si falla un disco el sistema puede seguir en funcionamiento.
RAID 4	<ul style="list-style-type: none"> Es ideal para almacenar ficheros de gran tamaño. 	<ul style="list-style-type: none"> Durante las operaciones de lectura-escritura las unidades de disco son accedidas de forma individual. 	<ul style="list-style-type: none"> Es tolerante a fallos ya que se puede recuperar los datos de un disco averiado en tiempo real.
RAID 5	<ul style="list-style-type: none"> Distribuye los datos de paridad entre todas las unidades de disco. 	<ul style="list-style-type: none"> La velocidad de transferencia de datos es alta. 	<ul style="list-style-type: none"> Es tolerante a fallos con una unidad de disco averiada.
RAID 6	<ul style="list-style-type: none"> Cada dato de paridad es redundante y distribuido en dos unidades de disco diferentes. 	<ul style="list-style-type: none"> Las operaciones de escritura resultan más lentas que las de lectura de datos. 	<ul style="list-style-type: none"> Es tolerante a fallos con dos unidades de discos averiadas.

Nivel	Ventajas	Desventajas
RAID 0	<ul style="list-style-type: none"> Mayor capacidad de almacenamiento, por lo que no duplica los datos. Proporciona mayor velocidad de lectura y escritura. Mejora el rendimiento de transferencia lineal. Se utiliza toda la capacidad del disco. 	<ul style="list-style-type: none"> En caso de que una unidad de disco falle, no es posible recuperar los datos. No posee tolerancias a fallos si, falla una unidad de disco el sistema queda fuera de servicio. No posee redundancia de datos. Las aplicaciones convencionales de escritorio no muestran mejora notable en el rendimiento.
RAID 1	<ul style="list-style-type: none"> Si una unidad de disco falla, es posible recuperar los datos mediante el disco espejo. Es tolerante a fallos, en caso de que una unidad falle el sistema puede seguir en servicio. Mayor rendimiento en la lectura de datos. 	<ul style="list-style-type: none"> Es más costoso puesto que se necesita el doble del espacio para almacenar los datos. La escritura de los datos es más lenta ya que se debe realizar en dos localizaciones. Si se escriben datos corruptos en una unidad estos son duplicados en la otra.
RAID 2	<ul style="list-style-type: none"> Mejora la demanda y la velocidad de transferencia. Realiza detección y correcciones de errores con código de Hamming. 	<ul style="list-style-type: none"> Es una solución costosa por la cantidad de discos que se necesitan para guardar los códigos de error. Tiempo de escritura bastante lento. No puede usarse con discos SCSI. No hay implementaciones comerciales de este nivel.
RAID 3	<ul style="list-style-type: none"> Implementa un disco de paridad para corrección de errores. Puede fallar un disco y es posible recuperar la información. Alto rendimiento para acceso secuencial de grandes archivos. 	<ul style="list-style-type: none"> Al fallar el disco de paridad se pierde la redundancia de los datos. El disco de paridad puede convertirse en un cuello de botella para el sistema. No puede atender múltiples peticiones de lectura-escritura.
RAID 4	<ul style="list-style-type: none"> Puede atender peticiones simultáneas de lectura-escritura. Las operaciones de lectura se realizan más rápido. 	<ul style="list-style-type: none"> El disco de paridad es un punto único fallo que puede producir la pérdida de redundancia sobre los datos. La escritura de datos pequeños resulta más lenta.
RAID 5	<ul style="list-style-type: none"> Tiene un mejor desempeño al trabajar con múltiples transacciones pequeñas. Ideal para aplicaciones con procesamiento transaccional. 	<ul style="list-style-type: none"> Si fallan 2 unidades de disco simultáneamente el sistema deja de funcionar. No aumenta el rendimiento de las aplicaciones.
RAID 6	<ul style="list-style-type: none"> Es posible recuperar los datos hasta con dos unidades de disco en falla. 	<ul style="list-style-type: none"> Es ineficiente con pocos discos.

- Usando el comando fdisk (fdisk /dev/sda) crear una nueva partición de tipo extendida, si es que no existe previamente. Para evitar tener que crearla nuevamente crear esta partición con el tamaño máximo posible.

```

Miscelánea
m muestra este menú
u cambia las unidades de visualización/entrada
x funciones adicionales (sólo para usuarios avanzados)

Script
I carga la estructura del disco de un fichero de script sfdisk
O vuelca la estructura del disco a un fichero de script sfdisk

Guardar y Salir
w escribe la tabla en el disco y sale
q sale sin guardar los cambios

Crea una nueva etiqueta
g crea una nueva tabla de particiones GPT vacía
G crea una nueva tabla de particiones SGI (IRIX) vacía
o crea una nueva tabla de particiones DOS vacía
s crea una nueva tabla de particiones Sun vacía

Orden (m para obtener ayuda): n
Tipo de partición
p primaria (2 primaria(s), 0 extendida(s), 2 libre(s))
e extendida (contenedor para particiones lógicas)
Seleccionar (valor predeterminado p): e
Número de partición (3,4, valor predeterminado 3):
Primer sector (79124480-335544319, valor predeterminado 79124480):
Último sector, +/-sectores o +/-tamaño{K,M,G,T,P} (79124480-335544319, valor predeterminado 335544319):
Crea una nueva partición 3 de tipo 'Extended' y de tamaño 122,3 GiB.

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

root@so2022:~# _

```

3. Dentro de esta partición extendida crear 3 nuevas particiones de 300MB cada una.
Para esto utilizar nuevamente el comando fdisk, pero ahora las particiones deben ser de tipo logical. Reiniciar.

```

root@so2022:~# sudo fdisk /dev/sda

Bienvenido a fdisk (util-linux 2.33.1).
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.
Tenga cuidado antes de utilizar la orden de escritura.

Orden (m para obtener ayuda): n
Se está utilizando todo el espacio para particiones primarias.
Se añade la partición lógica 5
Primer sector (79126528-335544319, valor predeterminado 79126528):
Último sector, +/-sectores o +/-tamaño{K,M,G,T,P} (79126528-335544319, valor predeterminado 335544319): +300M

Crea una nueva partición 5 de tipo 'Linux' y de tamaño 300 MiB.

Orden (m para obtener ayuda): n
Se está utilizando todo el espacio para particiones primarias.
Se añade la partición lógica 6
Primer sector (79742976-335544319, valor predeterminado 79742976):
Último sector, +/-sectores o +/-tamaño{K,M,G,T,P} (79742976-335544319, valor predeterminado 335544319): +300M

Crea una nueva partición 6 de tipo 'Linux' y de tamaño 300 MiB.

Orden (m para obtener ayuda): n
Se está utilizando todo el espacio para particiones primarias.
Se añade la partición lógica 7
Primer sector (80359424-335544319, valor predeterminado 80359424):
Último sector, +/-sectores o +/-tamaño{K,M,G,T,P} (80359424-335544319, valor predeterminado 335544319): +300M

Crea una nueva partición 7 de tipo 'Linux' y de tamaño 300 MiB.

```

```

root@so2022:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda         8:0    0 160G  0 disk
├─sda1      8:1    0  37,3G  0 part /
├─sda2      8:2    0  488M  0 part [SWAP]
├─sda3      8:3    0    1K  0 part
├─sda5      8:5    0  300M  0 part
├─sda6      8:6    0  300M  0 part
└─sda7      8:7    0  300M  0 part
sr0         11:0    1 1024M  0 rom

root@so2022:~# fdisk -l
Disco /dev/sda: 160 GiB, 171798691840 bytes, 335544320 sectores
Modelo de disco: VBOX HARDDISK
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: dos
Identificador del disco: 0xfa0715a1

Disposit.  Inicio Comienzo      Final Sectores Tamaño Id Tipo
/dev/sda1  *          2048  78125055  78123008  37,3G 83 Linux
/dev/sda2              78125056  79124479   999424  488M 82 Linux swap / Solaris
/dev/sda3              79124480 335544319 256419840 122,3G  5 Extendida
/dev/sda5              79126528  79740927   614400  300M 83 Linux
/dev/sda6              79742976  80357375   614400  300M 83 Linux
/dev/sda7              80359424  80973823   614400  300M 83 Linux

```

- Utilizar el comando mdadm para crear un RAID 5 utilizando las 3 particiones lógicas que se generaron en el punto anterior (fdisk -l para ver el nombre de las particiones que generaron):

```
mdadm --create --verbose /dev/md0 --level=5 --raid-devices=3 /dev/sda5
/dev/sda6 /dev/sda7
```

(Obs.: md0 es el nombre que le dará al nuevo RAID)

```

root@so2022:~# mdadm --create --verbose /dev/md0 --level=5 --raid-devices=3 /dev/sda5 /
dev/sda6 /dev/sda7
mdadm: layout defaults to left-symmetric
mdadm: layout defaults to left-symmetric
mdadm: chunk size defaults to 512K
mdadm: size set to 305152K
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.

```

- ¿Qué significan los valores sda5, sda6 y sda7?

Son las particiones lógicas a utilizar para crear el RAID.

- Ejecutar la siguiente consulta y contestar:

```
mdadm --detail /dev/md0
```

```

root@so2022:~# mdadm --detail /dev/md0
/dev/md0:
    Version : 1.2
    Creation Time : Sun May  5 15:18:59 2024
    Raid Level : raid5
    Array Size : 610304 (596.00 MiB 624.95 MB)
    Used Dev Size : 305152 (298.00 MiB 312.48 MB)
    Raid Devices : 3
    Total Devices : 3
    Persistence : Superblock is persistent

    Update Time : Sun May  5 15:19:02 2024
    State : clean
    Active Devices : 3
    Working Devices : 3
    Failed Devices : 0
    Spare Devices : 0


    Layout : left-symmetric
    Chunk Size : 512K

Consistency Policy : resync

        Name : so2022:0 (local to host so2022)
        UUID : 53aee208:41d294e6:04af71a3:ealf826d
        Events : 18

    Number Major Minor RaidDevice State
       0     8     5        0     active sync  /dev/sda5
       1     8     6        1     active sync  /dev/sda6
       3     8     7        2     active sync  /dev/sda7

```

a) ¿Cuál es el tamaño del RAID?

596.00 MiB.

b) ¿Qué significa Used Dev Size?

Es tamaño de disco que no es para datos.

7. Analizar el contenido del siguiente comando:

```
cat /proc/mdstat
```

Obs.: puede suceder que al reiniciar la VM el RAID se vea como de solo lectura y con el número 127. Para solucionar esto deben ejecutar los comandos `mdadm --stop /dev/md127` para parar el RAID y `mdadm --assemble --scan` para volverlo a generar como md0 y de lectura/escritura. Esto se debe hacer cada vez que se inicia la VM. Si quiere que quede en forma persistente a través de los reboots debe guardar la configuración en el archivo `mdadm.conf`, `mdadm --assemble --scan >> /etc/mdadm/mdadm.conf` y luego `update-initramfs -u` (esto último puede tardar un poco de tiempo)

```
root@so2022:~# cat /proc/mdstat
Personalities : [raid6] [raid5] [raid4]
md0 : active raid5 sda7[3] sda6[1] sda5[0]
      610304 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/3] [UUU]

unused devices: <none>
```

El comando `cat /proc/mdstat` muestra el estado del software RAID.

Personalities: indica los tipos de RAID soportados por el Kernel.

md0: muestra información sobre el RAID llamado md0. Este mismo está activo, es un raid5 y los dispositivos físicos involucrados son sda7, sda6 y sda5.

La siguiente línea proporciona algunos datos básicos sobre el tamaño y la disposición fija: indica que el tamaño utilizable del conjunto en bloques es 610304; el conjunto utiliza un superbloque 1.2 y confirma que es de nivel 5 con un tamaño de fragmento de 512k usando el algoritmo 2.

[3/3] significa que idealmente el RAID tiene 3 dispositivos y los 3 están en uso.

Obviamente, cuando $m \geq n$, las cosas están bien. [UUU] representa el estado de cada dispositivo, ya sea U activo o _ para caído. Están todos activos en este caso.

<https://raid.wiki.kernel.org/index.php/Mdstat>

8. Ahora se va a probar la funcionalidad del RAID 5. Para esto completar los siguientes pasos:

- a) Crear un file system de tipo ext4 en el RAID 5 recién generado

```
mkfs.ext4 /dev/md0
```

- b) Montar la partición con el file system generado en el directorio /mnt/rd5
- c) Crear un directorio con dos archivos


```
root@so2022:~# mkfs.ext4 /dev/md0
mke2fs 1.44.5 (15-Dec-2018)
Creating filesystem with 152576 4k blocks and 38160 inodes
Filesystem UUID: f442d8ac-50b3-411f-a023-9171a85c7d52
Superblock backups stored on blocks:
    32768, 98304
```

```
Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done
```

```
root@so2022:~# mkdir /mnt/rd5
root@so2022:~# mount /dev/md0 /mnt/rd5
root@so2022:~# cd /mnt/rd5/
root@so2022:/mnt/rd5# mkdir directorio
root@so2022:/mnt/rd5# cd directorio/
root@so2022:/mnt/rd5/directorio# touch archivo1 archivo2
root@so2022:/mnt/rd5/directorio# ls
archivo1  archivo2
```

- d) Quitar una de las particiones del RAID. Para esto ponemos uno de los componentes en falla:

```
mdadm --fail /dev/md0 /dev/sda7
```

```
root@so2022:/mnt/rd5/directorio# mdadm --fail /dev/md0 /dev/sda7
mdadm: set /dev/sda7 faulty in /dev/md0
```

- e) Observar el estado del RAID y contestar

```
md0 : active raid5 sda5[0] sda7[3](F) sda6[1]
    610304 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/2] [UU_]
```

- 1) ¿Cuál es el estado del RAID? ¿Cuántos dispositivos activos existen?

El RAID sigue activo. Actualmente hay 2 dispositivos activos de 3.

- 2) El tamaño del RAID, ¿se modificó?

No, no se modificó.

- 3) ¿Qué sucedería si se ejecuta el comando anterior sobre una de las particiones restantes?

Entonces el RAID en este caso dejaría de estar activo, puesto que un RAID nivel 5 puede tener como máximo 1 disco fallido para poder seguir funcionando.

- f) Quitar del RAID el componente puesto en falla en el paso anterior

```
mdadm --remove /dev/md0 /dev/sda7
```

- g) Observar nuevamente el estado del RAID y contestar:

- 1) ¿Se puede acceder al directorio /mnt/rd5? ¿Están los archivos creados anteriormente?

Si, se puede seguir accediendo, y también siguen estando los archivos.

- 2) ¿Qué hubiese sucedido si teníamos otra partición como “hot-spare”?

Si se tuviera otra partición como “hot-spare” esta hubiera reemplazado al componente puesto en falla. La partición "hot-spare" se habría activado y sincronizado con el resto del RAID para mantener la redundancia y la integridad de los datos. No se habrías tenido que sacar manualmente el componente en falla, ya que el RAID se habría reconstruido automáticamente con la partición "hot-spare".

- h) Por último, remover la partición permanentemente del RAID (Obs.: esto es muy importante para que el próximo booteo mdadm no intente usar a esta partición como parte del RAID, lo que provocaría la pérdida de todos los datos)

```
mdadm --zero-superblock /dev/sda7
```

A partir de este momento la partición /dev/sda7 se puede utilizar como una partición común

9. Para evitar la pérdida de datos es fundamental volver al RAID a un estado estable (sacarlo del estado degradado). Para esto se agregará nuevamente la partición /dev/sda7 que se quitó en el paso anterior
- a) Ejecutar el comando `mdadm --add /dev/md0 /dev/sda7`
 - b) Ejecutar el comando `mdadm --detail /dev/md0`

```
root@so2022:/home/so# mdadm --add /dev/md0 /dev/sda7
mdadm: added /dev/sda7
root@so2022:/home/so# mdadm --detail /dev/md0
/dev/md0:
    Version : 1.2
  Creation Time : Sun May  5 15:18:59 2024
    Raid Level : raid5
    Array Size : 610304 (596.00 MiB 624.95 MB)
  Used Dev Size : 305152 (298.00 MiB 312.48 MB)
    Raid Devices : 3
    Total Devices : 3
 Persistence : Superblock is persistent

    Update Time : Sun May  5 16:09:29 2024
      State : clean
    Active Devices : 3
    Working Devices : 3
    Failed Devices : 0
    Spare Devices : 0

    Layout : left-symmetric
    Chunk Size : 512K

Consistency Policy : resync

        Name : so2022:0 (local to host so2022)
        UUID : 53aee208:41d294e6:04af71a3:ealf826d
        Events : 48

   Number   Major   Minor   RaidDevice State
     0         8         5           0  active sync  /dev/sda5
     1         8         6           1  active sync  /dev/sda6
     3         8         7           2  active sync  /dev/sda7
```

- c) ¿Qué hace el RAID con la nueva partición recientemente agregada? ¿Qué significa el estado "Rebuild Status"?

Lo que hará será re-construir la información de la partición a partir del contenido de las demás.

El "Rebuild Status" indica en qué punto se encuentra el proceso de reconstrucción. Puede tener diferentes estados, como "idle" (inactivo), "rebuilding" (reconstruyendo), "resyncing" (resincronizando) o "check" (verificación). Cada estado indica qué acción se está llevando a cabo en el proceso de reconstrucción del conjunto RAID.

"Idle": Indica que no se está llevando a cabo ninguna reconstrucción en el momento.

"Rebuilding": Significa que el proceso de reconstrucción está activo y en curso.

"Resyncing": Se utiliza para indicar que se está sincronizando el nuevo disco con los datos existentes en el conjunto RAID.

"Check": Puede indicar que se está realizando una verificación de integridad en el conjunto RAID.

- d) ¿Es posible ingresar al recurso /mnt/rd5? ¿Se encuentran disponibles los datos creados en el punto anterior?

Si, es posible ingresar al recurso. Se encuentran disponibles los datos creados en el punto anterior.

10. Como los datos que mantiene el RAID son muy importantes es necesario tener un disco (partición en nuestro ejemplo) de respaldo. Para esto se va a agregar una partición como "hot-spare".

11. Usando el comando fdisk o parted generar una nueva partición, /dev/sda8, con igual tamaño a las anteriores.

```
root@so2022:/mnt/rd5/directorio# fdisk /dev/sda

Bienvenido a fdisk (util-linux 2.33.1).
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.
Tenga cuidado antes de utilizar la orden de escritura.

Orden (m para obtener ayuda): n
Se está utilizando todo el espacio para particiones primarias.
Se añade la partición lógica 8
Primer sector (80975872-335544319, valor predeterminado 80975872):
Último sector, +/-sectores o +/-tamaño{K,M,G,T,P} (80975872-335544319, valor predeterminado 335544319): +300M

Crea una nueva partición 8 de tipo 'Linux' y de tamaño 300 MiB.

Orden (m para obtener ayuda): w
Se ha modificado la tabla de particiones.
Se están sincronizando los discos.
```

12. Agregar la nueva partición al RAID:

```
mdadm --add /dev/md0 /dev/sda8
```

a) ¿Cómo se agregó la nueva partición? ¿Por qué?

Number	Major	Minor	RaidDevice	State	
0	8	5	0	active sync	/dev/sda5
1	8	6	1	active sync	/dev/sda6
3	8	7	2	active sync	/dev/sda7
4	8	8	-	spare	/dev/sda8

Se agrego como spare. Esto sucede porque el RAID fue creado para funcionar con 3 dispositivos los cuales ya están activos. El nuevo dispositivo agregado no va a tener datos hasta que alguno de los otros falle.

13. Volver a poner en falla a la partición /dev/sda7. Ver el estado del RAID y contestar

```
mdadm --detail /dev/mda0
```

a) ¿Qué hace el RAID con la partición que estaba como spare?

```
Name : so2022:0 (local to host so2022)
UUID : 53aee208:41d294e6:04af71a3:eal1f826d
Events : 113
```

Number	Major	Minor	RaidDevice	State	
0	8	5	0	active sync	/dev/sda5
1	8	6	1	active sync	/dev/sda6
4	8	8	2	active sync	/dev/sda8
3	8	7	-	faulty	/dev/sda7

Ahora la partición que estaba como spare va a ser de datos. Esta misma reemplaza a la partición que fallo.

14. Por último, se eliminará el RAID creado en los pasos anteriores:

a) Desmontar el RAID (comando umount)

b) Para cada una de las particiones del RAID ejecutar los pasos realizados cuando se quitó una partición del RAID (mdadm con las opciones --fail y --remove). Por cada partición que se quita ir mirando el estado del RAID para ver como se comporta

- c) Remover los superbloques de cada una de las particiones

```
mdadm --zero-superblock /dev/sda5 /dev/sda6 /dev/sda7
```

- d) Remover el RAID

```
mdadm --remove /dev/md0
```

Obs.: si existe, quitar la línea ARRAY... del archivo /etc/mdadm/mdadm.conf

```
root@so2022:/mnt# mdadm --fail /dev/md0 /dev/sda5
mdadm: set /dev/sda5 faulty in /dev/md0
root@so2022:/mnt# mdadm --remove /dev/md0 /dev/sda5
mdadm: hot removed /dev/sda5 from /dev/md0
root@so2022:/mnt# mdadm --fail /dev/md0 /dev/sda6
mdadm: set /dev/sda6 faulty in /dev/md0
root@so2022:/mnt# mdadm --remove /dev/md0 /dev/sda6
mdadm: hot removed /dev/sda6 from /dev/md0
root@so2022:/mnt# mdadm --fail /dev/md0 /dev/sda7
mdadm: set /dev/sda7 faulty in /dev/md0
root@so2022:/mnt# mdadm --remove /dev/md0 /dev/sda7
mdadm: hot removed /dev/sda7 from /dev/md0
root@so2022:/mnt# mdadm --fail /dev/md0 /dev/sda8
mdadm: set /dev/sda8 faulty in /dev/md0
root@so2022:/mnt# mdadm --remove /dev/md0 /dev/sda8
mdadm: hot removed /dev/sda8 from /dev/md0
root@so2022:/mnt# cat /proc/mdstat
Personalities : [raid6] [raid5] [raid4] [linear] [multipath] [raid0] [raid1] [raid10]
md0 : active raid5 super 1.2 level 5, 512k chunk, algorithm 2 [3/0] [____]

unused devices: <none>
root@so2022:/mnt# mdadm --zero-superblock /dev/sda5 /dev/sda6 /dev/sda7
root@so2022:/mnt# mdadm --remove /dev/md0
```

- e) Reiniciar y comprobar que el RAID ya no existe

```
mdadm --stop /dev/md0
```

```
mdadm --remove /dev/md0
```

Para eliminar RAID

```
root@so2022:/# mdadm --stop /dev/md0
mdadm: stopped /dev/md0
root@so2022:/# sudo mdadm --detail /dev/md0
mdadm: cannot open /dev/md0: No such file or directory
```

LVM (Logical Volumen Management)

1. ¿Qué es LVM? ¿Qué ventajas presenta sobre el particionado tradicional de Linux?

Logical Volume Management (LVM) provee un método más flexible que los convencionales esquemas de particionamiento para alocar espacio en los dispositivos de almacenamiento masivo.

Crea una capa de abstracción en el almacenaje físico, permitiendo crear volúmenes lógicos de almacenaje ofreciendo más flexibilidad respecto al uso directo de almacenaje físico.

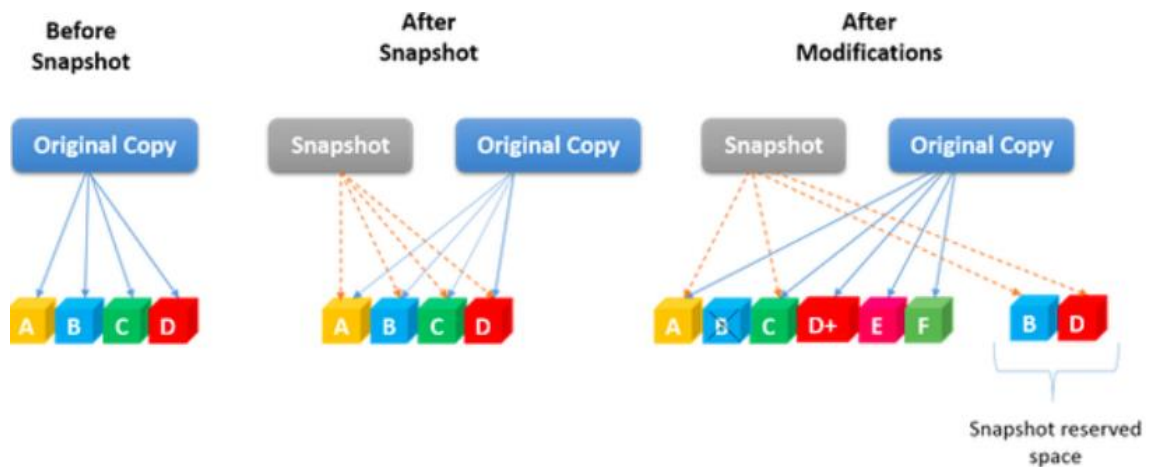
Con un volumen lógico no hay restricción física de espacio del disco. Además, la configuración de almacenaje del hardware se oculta del software permitiendo así redimensionar y desplazar sin tener que detener la aplicación o desmontar el sistema de archivos. Esto puede reducir costos operacionales:

- Capacidad flexible:
 - Los sistemas de archivos pueden extenderse a lo largo de varios discos, ya que se pueden agregar discos y particiones en un único volumen lógico.
- Grupos de almacenamiento dimensionables:
 - Se pueden extender los volúmenes lógicos o reducir los volúmenes lógicos con comandos de software sin necesidad de reformatar o crear particiones en los discos subyacentes.
- Asignación de datos en línea:
 - Se pueden trasladar datos mientras el sistema está activo para implementar subsistemas de almacenamiento más modernos o resilientes. Los datos pueden ser reorganizados en discos mientras los discos están siendo utilizados
- Nombres de dispositivos convenientes:
 - Los volúmenes de almacenaje lógico pueden ser administrados en grupos definidos por el usuario.
- Entrelazado de disco:
 - Posibilidad de crear volúmenes lógicos que entrelazan datos a través de múltiples discos, aumentando la velocidad de transferencia.
- Volúmenes en espejo:
 - Los volúmenes lógicos ofrecen una forma conveniente de configurar copias de seguridad para los datos.
- Instantáneas del volumen:

- Se pueden crear instantáneas del dispositivo con volúmenes lógicos para obtener copias de seguridad consistentes o probar cambios sin afectar los datos reales.

2. ¿Cómo funcionan los “snapshots” en LVM?

Cuando se realiza una snapshot LVM, tanto la snapshot como la ubicación original tienen los mismos metadatos y datos (al crearse el snapshot solo se copian los metadatos). Recién cuando haya una modificación (en la primera nada más) de datos la snapshot copiara un “Point-In-Time” de la ubicación original a otro volumen lógico temporal y ahora la snapshot apuntara a esa copia. Este volumen lógico temporal solo va a contener los metadatos y los bloques de datos de la LV origen que hayan sido modificados desde que se generó la snapshot. Los que no se modificaron, no estarán. Para reestablecer la snapshot, se copian los bloques de datos a la ubicación original. Si se desea mantener los cambios realizados, entonces se borra la snapshot.



3. Instalar la herramienta lvm2: `apt-get install lvm2`
4. Ejecutar el siguiente comando, ¿qué es lo que realiza?

```
pvccreate /dev/sda5 /dev/sda6
```

```
root@so2022:~# pvccreate /dev/sda5 /dev/sda6
Physical volume "/dev/sda5" successfully created.
Physical volume "/dev/sda6" successfully created.
```

Crea dos volúmenes físicos a partir de las particiones /dev/sda5 y /dev/sda6.

5. Mediante el comando `pvdisk` observar el estado del volumen físico recientemente creado

```
root@so2022:~# pvdisk
"/dev/sda5" is a new physical volume of "300,00 MiB"
--- NEW Physical volume ---
PV Name           /dev/sda5
VG Name
PV Size           300,00 MiB
Allocatable       NO
PE Size           0
Total PE          0
Free PE           0
Allocated PE      0
PV UUID           C1ItfG-hVzc-nee1-5MjW-YUop-y8TE-xNuqTq

"/dev/sda6" is a new physical volume of "300,00 MiB"
--- NEW Physical volume ---
PV Name           /dev/sda6
VG Name
PV Size           300,00 MiB
Allocatable       NO
PE Size           0
Total PE          0
Free PE           0
Allocated PE      0
PV UUID           skRka5-Kkpt-xc0i-7oEL-9www-2ucu-tcNOLi
```

6. Crear un grupo de volúmenes (volume group, VG) llamado "so".

```
vgcreate so /dev/sda5 /dev/sda6
```

```
root@so2022:~# vgcreate "so" "/dev/sda5" "/dev/sda6"
Volume group "so" successfully created
```

7. Utilizar el comando `vgdisplay` para ver el estado del VG

```
root@so2022:~# vgdisplay so
--- Volume group ---
VG Name           so
System ID
Format            lvm2
Metadata Areas     2
Metadata Sequence No 1
VG Access          read/write
VG Status          resizable
MAX LV            0
Cur LV            0
Open LV            0
Max PV             0
Cur PV            2
Act PV             2
VG Size            592,00 MiB
PE Size            4,00 MiB
Total PE           148
Alloc PE / Size    0 / 0
Free PE / Size     148 / 592,00 MiB
VG UUID            sECdqR-ptfW-FDNB-qre6-1KGL-00Gq-THpB0o
```

- ¿Cuál es tamaño total del VG?

592 MiB

- ¿Qué significa PE? ¿Qué define?

Physical Extent (PE) son las unidades direccionables en las que se divide cada volumen físico. Un PE es la unidad de asignación básica en un Volumen de Grupo (Volume Group) de LVM. El número de PEs total indica la cantidad total de PEs que existen en el VG, mientras que el número de PEs utilizadas representa la cantidad de PEs que se han asignado y utilizan actualmente en el VG para los Logical Volumes (LVs).

Cuando se crea un grupo de volúmenes (VG) en LVM, los discos físicos o las particiones se dividen en PEs, y luego estos PEs se pueden asignar a los volúmenes lógicos (LV) como bloques de almacenamiento.

8. Crear dos volúmenes lógicos (logical volume, LV) de 8MB y 117MB respectivamente

```
lvcreate -l 2 -n lv_vol1 so
lvcreate -L 117M -n lv_vol2 so
```

```
root@so2022:~# lvcreate -l 2 -n lv_vol1 so
Logical volume "lv_vol1" created.
root@so2022:~# lvcreate -L 117M -n lv_vol2 so
Rounding up size to full physical extent 120,00 MiB
Logical volume "lv_vol2" created.
```

9. ¿Cuál es la diferencia entre los dos comandos utilizados en el punto anterior?

-l se usa para especificar el tamaño del volumen lógico en extents

-L se usa para especificar el tamaño del volumen lógico en bytes. Esta misma se calcula de la siguiente manera:

$$\text{ceil}\left(\frac{\text{NumBytes}}{\text{TamañoPE}}\right) * \text{TamañoPE}$$

<https://linux.die.net/man/8/lvcreate>

10. ¿Con qué tamaño se generó el LV lv_vol2? ¿Por qué?

Se genero con 120 MiB.

$$\text{ceil}\left(\frac{120 \text{ MiB}}{4 \text{ MiB}}\right) * 4 \text{ MiB} = 120 \text{ MiB}$$

Esto se realiza de dicha manera para respetar el tamaño de los PE dado que a cada volumen logico se le asigna una cantidad de PEs garantizando que se utilice de manera eficiente el espacio asignado en los PEs.

11. Formatear los dos LV generados en el paso anterior con un file system de tipo ext4:

```
mkfs.ext4 /dev/so/lv_vol1
```

12. Crear dos directorios, vol1 y vol2, dentro de /mnt y montar ambos LVs en estos directorios (montar el LV lv_vol1 en el directorio vol1 y lv_vol2 en vol2)

```
root@so2022:/mnt# lvs
--- Logical volume ---
LV Path                /dev/so/lv_vol1
LV Name                 lv_vol1
VG Name                 so
LV UUID                 EAqMgD-P1DN-8Fmq-mRNA-H03e-UwQ5-GL9oat
LV Write Access         read/write
LV Creation host, time  so2022, 2024-05-06 12:30:59 -0300
LV Status                available
# open                  0
LV Size                 8.00 MiB
Current LE               2
Segments                1
Allocation               inherit
Read ahead sectors      auto
- currently set to      256
Block device            253:0

--- Logical volume ---
LV Path                /dev/so/lv_vol2
LV Name                 lv_vol2
VG Name                 so
LV UUID                 P2pmi3-bnsk-y8rx-b0k9-bqwT-NEZV-DxMFkf
LV Write Access         read/write
LV Creation host, time  so2022, 2024-05-06 12:31:06 -0300
LV Status                available
# open                  0
LV Size                 120.00 MiB
Current LE               30
Segments                1
Allocation               inherit
Read ahead sectors      auto
- currently set to      256
Block device            253:1

root@so2022:/mnt# mount /dev/so/lv_vol1 /mnt/vol1
root@so2022:/mnt# mount /dev/so/lv_vol2 /mnt/vol2
```

13. Ejecutar el comando proof (Puede tomar un rato su ejecución. Este comando estará disponible en la plataforma y deberán copiarlo a la VM)

```
root@so2022:/home/so/Descargas# ./proof
```

14. Crear un nuevo archivo en /mnt/vol1. ¿Es posible? ¿Por qué? ¿Qué debería hacerse para solucionarlo?

No, no se puede. No hay mas espacio en el dispositivo. El problema es que la tabla de inodos de la partición está llena. Para solucionar esto se tendrían que eliminar archivos o aumentar el tamaño del file system lo que implicaría aumentar el tamaño del LV (lv_vol1) para que haya suficiente espacio disponible para crear un nuevo archivo en /mnt/vol1.

15. Para aplicar la solución propuesta en el punto anterior, para esto primero se debe incrementar el tamaño del LV correspondiente:

a) Extender el LV lv_vol1 en 20M

```
lvextend -L +20M /dev/so/lv_vol1
```

b) Ejecute el comando df -h, ¿se refleja en la salida del comando el incremento del espacio? ¿Por qué?

No, no se refleja. Esto se debe a que se tiene que incrementar también el tamaño del file system, si no este sigue siendo el mismo.

c) Incrementar el tamaño del file system

```
resize2fs /dev/so/lv_vol1
```

```
root@so2022:/mnt# resize2fs /dev/so/lv_vol1
resize2fs 1.44.5 (15-Dec-2018)
Filesystem at /dev/so/lv_vol1 is mounted on /mnt/vol1; on-line resizing required
old_desc_blocks = 1, new_desc_blocks = 1
The filesystem on /dev/so/lv_vol1 is now 28672 (1k) blocks long.
```

```
root@so2022:/mnt# df -h
```

S.ficheros	Tamaño	Usados	Disp	Uso%	Montado en
udev	3,9G	0	3,9G	0%	/dev
tmpfs	798M	9,2M	789M	2%	/run
/dev/sda1	37G	4,7G	30G	14%	/
tmpfs	3,9G	0	3,9G	0%	/dev/shm
tmpfs	5,0M	0	5,0M	0%	/run/lock
tmpfs	3,9G	0	3,9G	0%	/sys/fs/cgroup
tmpfs	798M	7,7M	791M	1%	/run/user/1000
/dev/mapper/so-lv_vol1	26M	61K	25M	1%	/mnt/vol1
/dev/mapper/so-lv_vol2	111M	14K	103M	1%	/mnt/vol2

16. Después de la operación previa, ¿siguen estando los datos disponibles?

Si, siguen estando disponibles.

17. Intentar crear un nuevo archivo en /mnt/vol1. ¿Es posible? ¿Por qué?

Si es posible, porque se incremento el tamaño del volumen lógico y del file system, y por lo tanto se incrementó también la capacidad de la tabla de inodos

18. Se desea crear un nuevo LV de 500M. ¿Hay suficiente espacio? ¿Cómo lo solucionaría?

```

root@so2022:/mnt/voll# vgdisplay
--- Volume group ---
VG Name                so
System ID
Format                 lvm2
Metadata Areas         2
Metadata Sequence No   4
VG Access               read/write
VG Status               resizable
MAX LV                 0
Cur LV                 2
Open LV                 2
Max PV                 0
Cur PV                 2
Act PV                 2
VG Size                 592,00 MiB
PE Size                 4,00 MiB
Total PE                148
Alloc PE / Size        37 / 148,00 MiB
Free PE / Size          111 / 444,00 MiB
VG UUID                 sECdqR-ptfW-FDNB-qre6-1KGL-00Gq-THpB0o

```

No se podría porque no habría suficiente espacio en el volumen group. Se debería agregar otro volumen físico.

19. Para aplicar la solución indicada en el punto anterior, realizar lo siguiente:

```

pvcreate /dev/sda7
vgextend so /dev/sda7

```

20. Comprobar con los comando correspondientes que se haya extendido el tamaño del VG

```

root@so2022:/mnt/voll# pvcreate /dev/sda7
Physical volume "/dev/sda7" successfully created.
root@so2022:/mnt/voll# vgextend so /dev/sda7
Volume group "so" successfully extended
root@so2022:/mnt/voll# vgdisplay
--- Volume group ---
VG Name                so
System ID
Format                 lvm2
Metadata Areas         3
Metadata Sequence No   5
VG Access               read/write
VG Status               resizable
MAX LV                 0
Cur LV                 2
Open LV                 2
Max PV                 0
Cur PV                 3
Act PV                 3
VG Size                 888,00 MiB
PE Size                 4,00 MiB
Total PE                222
Alloc PE / Size        37 / 148,00 MiB
Free PE / Size          185 / 740,00 MiB
VG UUID                 sECdqR-ptfW-FDNB-qre6-1KGL-00Gq-THpB0o

```

21. Generar el nuevo LV de 500M (llamarlo lv_vol3)

A continuación se mostrará el funcionamiento de los snapshot en LVM

```

root@so2022:/mnt/voll# lvcreate -L 500M -n lv_vol3 so
Logical volume "lv_vol3" created.

```

22. Generar un LV de 100M, nombrarlo lv1, (o usar uno de los generados en pasos anteriores). Montarlo en el directorio /dir1

```
root@so2022:/mnt# mkfs.ext4 /dev/so/lv1
mkfs 1.44.5 (15-Dec-2018)
Creating filesystem with 102400 1k blocks and 25688 inodes
Filesystem UUID: 64668dbc-eb8d-4b3b-ac54-a85a9eeb5dc2
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

root@so2022:/mnt# mount /dev/so/lv1 /mnt/dir1
```

23. Copiar desde /etc todo los archivos y directorios que comiencen con la letra a, b, c y d.

```
root@so2022:/mnt# cp -r /etc/[a-d]* /mnt/dir1
root@so2022:/mnt# ls
dir1  voll  vol2
root@so2022:/mnt# cd dir1
root@so2022:/mnt/dir1# ls
adduser.conf  apparmor.d  ca-certificates  cron.hourly  depmod.d
adjtime       appstream.conf  ca-certificates.conf  cron.monthly  dhcp
aliases       apt          ca-certificates.conf.dpkg-old  crontab       dictionaries-common
alsa          avahi        calendar          cron.weekly   discover.conf.d
alternatives  bash.bashrc  chatscripts       cupshelpers   discover-modprobe.conf
anacrontab    bash_completion  chromium          dbus-1        dleyna-server-service.conf
apache2       bindresvport.blacklist  console-setup    debconf.conf  dpkg
apg.conf      binfmt.d     cracklib          debian_version  lost+found
apm           bluetooth    cron.d            default
apparmor      bogofilter.cf  cron.daily        deluser.conf
```

24. Mediante el siguiente comando generar un snapshot del LV anterior

```
lvcreate -L 30M -s /dev/so/lv1 -n lvcopy (-s indica que este LV será un snapshot)
```

25. Verificar la creación del snapshot con el comando lvs. Montarlo en el directorio /snap. ¿Qué contenido tiene en el snapshot?

```
root@so2022:/mnt/dir1# lvcreate -L 30M -s /dev/so/lv1 -n lvcopy
Rounding up size to full physical extent 32,00 MiB
Logical volume "lvcopy" created.
root@so2022:/mnt/dir1# lvs
LV      VG Attr      LSize   Pool Origin Data%  Meta%   Move Log Cpy%Sync Convert
lv1     so  owi-aos--- 100,00m
lv_voll so -wi-ao---- 28,00m
lv_vol2 so -wi-ao---- 120,00m
lv_vol3 so -wi-a----- 500,00m
lvcopy  so swi-a-s--- 32,00m      lv1      0,04
```

26. ¿Cuánto espacio hay consumido en el snapshot creado? ¿Por qué sucede esto?

No consume lo mismo que lv1, esto sucede debido a que, como no se modificó ningún dato aun, la snapshot lo único que tiene es una copia de los metadatos. Recién cuando se modifique o borre un bloque se van a copiar en la snapshot los valores de los bloques originales y se van a actualizarán los punteros.

27. Para probar el snapshot, elimine una carpeta del LV original (por ej, la carpeta apt). ¿Se eliminó en el LV original? ¿Y qué sucedió en el snapshot?

```

root@so2022:/mnt/dir1# rm -r apt
root@so2022:/mnt/dir1# ls
adduser.conf  apparmor.d      ca-certificates.conf  cron.monthly  dhcp
adjtime       appstream.conf  ca-certificates.conf.dpkg-old  crontab       dictionaries-common
aliases       avahi           calendar              cron.weekly   discover.conf.d
alsa          bash.bashrc     chatscripts           cupshelpers   discover-modprobe.conf
alternatives  bash_completion chromium             dbus-1        dley-na-server-service.conf
anacrontab    bindresvport.blacklist  console-setup        debconf.conf  dpkg
apache2       binfmt.d        cracklib              debian_version  lost+found
apg.conf      bluetooth       cron.d                default
apm           bogofilter.cf   cron.daily            deluser.conf
apparmor      ca-certificates cron.hourly           depmod.d

root@so2022:/mnt/dir1# ls /mnt/snap
adduser.conf  apparmor.d      ca-certificates      cron.hourly  depmod.d
adjtime       appstream.conf  ca-certificates.conf  cron.monthly  dhcp
aliases       apt             ca-certificates.conf.dpkg-old  crontab       dictionaries-common
alsa          avahi           calendar             cron.weekly   discover.conf.d
alternatives  bash.bashrc     chatscripts          cupshelpers   discover-modprobe.conf
anacrontab    bash_completion chromium            dbus-1        dley-na-server-service.conf
apache2       bindresvport.blacklist  console-setup        debconf.conf  dpkg
apg.conf      binfmt.d        cracklib             debian_version  lost+found
apm           bluetooth       cron.d               default
apparmor      bogofilter.cf   cron.daily           deluser.conf

```

Se elimino en la original, pero no en la snapshot. Ahora lvcopy es más grande.

```

root@so2022:/mnt/dir1# lvs
LV      VG Attr      LSize   Pool Origin Data%  Meta%   Move Log Cpy%Sync Convert
lv1     so  owi-aos--- 100,00m
lv_vol1 so -wi-ao---- 28,00m
lv_vol2 so -wi-ao---- 120,00m
lv_vol3 so -wi-a----- 500,00m
lvcopy  so swi-aos--- 32,00m      lv1      0,21

```

28. Si se desea volver el LV a su estado original se debe hacer un “merge” entre el LV y su snapshot. Para esto primero deben desmontar el LV original y su snapshot correspondiente. Luego, realizar el “merge” de ambos LVs:

```
lvconvert --merge /dev/so/lvcopy
```

```

root@so2022:/mnt# umount /dev/so/lv1 /mnt/dir1
umount: /mnt/dir1: no montado.
root@so2022:/mnt# umount /dev/so/lvcopy /mnt/snap
umount: /mnt/snap: no montado.

root@so2022:/mnt# lvconvert --merge /dev/so/lvcopy
Merging of volume so/lvcopy started.
so/lv1: Merged: 100,00%

```

29. Comprobar si el LV original contiene nuevamente los datos eliminados anteriormente (Deberá montarlo nuevamente)

```

root@so2022:/mnt# mount /dev/so/lv1 /mnt/dir1
root@so2022:/mnt# ls /mnt/dir1
adduser.conf  apparmor.d      ca-certificates      cron.hourly  depmod.d
adjtime       appstream.conf  ca-certificates.conf  cron.monthly  dhcp
aliases       apt             ca-certificates.conf.dpkg-old  crontab       dictionaries-common
alsa          avahi           calendar             cron.weekly   discover.conf.d
alternatives  bash.bashrc     chatscripts          cupshelpers   discover-modprobe.conf
anacrontab    bash_completion chromium            dbus-1        dley-na-server-service.conf
apache2       bindresvport.blacklist  console-setup        debconf.conf  dpkg
apg.conf      binfmt.d        cracklib             debian_version  lost+found
apm           bluetooth       cron.d               default
apparmor      bogofilter.cf   cron.daily           deluser.conf

```

30. ¿Qué sucedió con el snapshot? Obs.: en caso que aparezca el error “Can’t merge over...” ejecutar los siguientes comandos para desactivar y activar el LV

```
lvchange -an /dev/so/lv1
lvchange -ay /dev/so/lv1
```

La snapshot fue eliminada (ver lvs o lvs)

BTRFS & ZFS

1. Tanto para BTRFS como para ZFS, responder:

- ¿Cuál es el significado de las siglas?

BTRFS: B-tree FS

ZFS: Zettabyte File System

- ¿Quién los creó? ¿Cuál es su modo de licenciamiento?

BTRFS: Oracle Corporation – GPL

ZFS: Sun Microsystems– the Common Development and Distribution License (CDDL)

- ¿Cuáles son las características más importante de cada uno?

- BTRFS:

- Tamaño máximo de volumen: 16EiB.
- Tamaño máximo de archivo: 16EiB
- Cantidad máxima de archivos: 2^{64}
- Trabaja con:
 - Extents.
 - C-O-W.
 - Alocación dinámica de inodos.
 - Soporte integrado de múltiples dispositivos:
 - RAID0 (stripe) similar al RAID0 tradicional.
 - RAID1 (mirror) similar al RAID1 tradicional.
 - RAID5 y RAID6 similares a los RAID1 y RAID6 tradicionales.
 - Checksum de datos y metadatos y Scrub.
 - Subvolumenes.
 - Snapshots.
- Permite:
 - Agregar o remover dispositivo de bloques.

- Agrandar o achicar volúmenes.
- Desfragmentación online.
- Compresión (zlib, LZO, etc.).

- ZFS

- Tamaño máximo de volumen: 256 ZiB.
- Tamaño máximo de archivo: 16EiB
- Cantidad máxima de archivos: 2^{48}
- Trabaja con:
 - C-O-W.
 - Self-healing.
 - Soporte integrado de múltiples dispositivos
 - RAIDZ es el equivalente de ZFS a RAID5 y RAID6, pero con un esquema de paridad mejorado para evitar el llamado agujero de escritura(write-hole) que afecta al RAID5 y RAID6 tradicional. RAIDZ utiliza una distribución de paridad similar pero con una estrategia de escritura más inteligente para mejorar el rendimiento y la integridad de los datos.
 - RAIDZ2 y RAIDZ3 son configuraciones son similares a RAIDZ, pero con dos o tres discos de paridad respectivamente. Esto proporciona una mayor tolerancia a fallos y capacidad de recuperación en comparación con RAIDZ.
 - Mirrored vdevs es imilar al RAID1, ZFS permite la creación de "vdevs" (dispositivos virtuales) espejados, donde los datos se duplican en múltiples dispositivos para brindar redundancia. Esto se puede combinar con RAIDZ para crear configuraciones híbridas que ofrecen tanto rendimiento como redundancia.
 - Checksum de datos y metadatos y Scrub.
 - Subvolúmenes (datasets)

- Snapshots.
- Permite :
 - Gestión avanzada del almacenamiento,
 - Agrupación de dispositivos (vdevs)
 - Caché de lectura y escritura (L2ARC)
 - Caché adaptativa (ARC).
 - Compresión.
 - No cuenta con desfragmentación.

<https://www.redeszone.net/tutoriales/servidores/sistema-archivos-zfs-servidores/#294949-espacios-de-almacenamiento-virtuales-storage-pools>

- Investigar qué es la técnica copy-on-write

Cuando se quiere realizar una escritura sobre los datos, en vez de modificarse los datos originales, se crea una nueva copia. En lugar de sobrescribir los datos existentes, la modificación se realiza en la copia recién creada y luego se modifican los metadatos para que se apunte a este nuevo bloque. De esta manera, se asegura que los datos originales permanezcan intactos hasta que se complete la operación de escritura en ellos. La técnica Copy-on-Write proporciona ciertos niveles de atomicidad y consistencia en las operaciones de escritura, lo que garantiza que las operaciones se completen de manera segura y sin corrupción de datos.

2. Generar en la MV dos particiones de 3GB cada una. Crear 3 nuevos directorios llamados /disco5, /volumen1 y /volumen2.

```
Orden (m para obtener ayuda): n
Se está utilizando todo el espacio para particiones primarias.
Se añade la partición lógica 9
Primer sector (81592320-335544319, valor predeterminado 81592320):
Último sector, +/-sectores o +/-tamaño{K,M,G,T,P} (81592320-335544319, valor predeterminado 335544319): +3G

Crea una nueva partición 9 de tipo 'Linux' y de tamaño 3 GiB.

Orden (m para obtener ayuda): n
Se está utilizando todo el espacio para particiones primarias.
Se añade la partición lógica 10
Primer sector (87885824-335544319, valor predeterminado 87885824):
Último sector, +/-sectores o +/-tamaño{K,M,G,T,P} (87885824-335544319, valor predeterminado 335544319): +3G

Crea una nueva partición 10 de tipo 'Linux' y de tamaño 3 GiB.
```

3. Tomar una partición, /dev/sdaX (X=número de una de las particiones creadas en el punto anterior), y crearle un file system de tipo BTRFS. Montarla en el directorio disco5. (Sino están instalados los comandos para BTRFS: apt-get install btrfs-progs)

```
root@so2022:/mnt# mkfs.btrfs -t /dev/sda9
btrfs-progs v4.20.1
See http://btrfs.wiki.kernel.org for more information.

Label:              (null)
UUID:               5c5e0712-3622-4a39-b23b-c460b534122d
Node size:          16384
Sector size:        4096
Filesystem size:    3.00GiB
Block group profiles:
  Data:              single                8.00MiB
  Metadata:          DUP                  153.56MiB
  System:             DUP                   8.00MiB
SSD detected:       no
Incompat features:  extref, skinny-metadata
Number of devices:  1
Devices:
  ID        SIZE  PATH
  1         3.00GiB /dev/sda9
```

4. Por defecto, BTRFS, ¿replica los datos? ¿Y los metadatos? ¿Es posible modificar esto? ¿Cómo lo haría? (Hint: usar btrfs fi df -h /disco5 o btrfs df usage /disco5)

```
root@so2022:/mnt# btrfs fi usage /mnt/disco5
Overall:
  Device size:              3.00GiB
  Device allocated:         331.12MiB
  Device unallocated:       2.68GiB
  Device missing:           0.00B
  Used:                     320.00KiB
  Free (estimated):         2.68GiB   (min: 1.35GiB)
  Data ratio:               1.00
  Metadata ratio:           2.00
  Global reserve:           16.00MiB   (used: 0.00B)

Data,single: Size:8.00MiB, Used:64.00KiB
/dev/sda9      8.00MiB

Metadata,DUP: Size:153.56MiB, Used:112.00KiB
/dev/sda9      307.12MiB

System,DUP: Size:8.00MiB, Used:16.00KiB
/dev/sda9      16.00MiB

Unallocated:
/dev/sda9      2.68GiB
```

Por defecto BTRFS replica los metadatos (DUP) pero no los datos (single). Si es posible modificar esto con:

```
btrfs balance start -dconvert=dup /mnt/disco5
```

```
Overall:
  Device size:          3.00GiB
  Device allocated:     963.12MiB
  Device unallocated:   2.06GiB
  Device missing:       0.00B
  Used:                 512.00KiB
  Free (estimated):     1.34GiB      (min: 1.34GiB)
  Data ratio:           2.00
  Metadata ratio:       2.00
  Global reserve:       16.00MiB      (used: 0.00B)
```

```
Data,DUP: Size:320.00MiB, Used:128.00KiB
/dev/sda9      640.00MiB
```

```
Metadata,DUP: Size:153.56MiB, Used:112.00KiB
/dev/sda9      307.12MiB
```

```
System,DUP: Size:8.00MiB, Used:16.00KiB
/dev/sda9      16.00MiB
```

```
Unallocated:
/dev/sda9      2.06GiB
```

5. ¿Cuál es el espacio alocado? ¿Y el ocupado realmente? Utilice los comandos `df -h`, `btrfs fi show`, `btrfs fi df -h /disco5` y `btrfs df usage /disco5`.

```
root@so2022:/mnt# df -h
S.ficheros      Tamaño Usados  Disp Uso% Montado en
udev            3,9G      0  3,9G  0% /dev
tmpfs           798M    9,2M  789M  2% /run
/dev/sda1        37G    4,7G   30G 14% /
tmpfs           3,9G      0  3,9G  0% /dev/shm
tmpfs           5,0M      0  5,0M  0% /run/lock
tmpfs           3,9G      0  3,9G  0% /sys/fs/cgroup
tmpfs           798M    6,5M  792M  1% /run/user/1000
/dev/sda9        3,0G    17M  2,7G  1% /mnt/disco5
root@so2022:/mnt# btrfs fi show
Label: none  uuid: 5c5e0712-3622-4a39-b23b-c460b534122d
      Total devices 1 FS bytes used 256.00KiB
      devid    1 size 3.00GiB used 643.12MiB path /dev/sda9

root@so2022:/mnt# btrfs fi df -h /mnt/disco5
Data, single: total=320.00MiB, used=128.00KiB
System, DUP: total=8.00MiB, used=16.00KiB
Metadata, DUP: total=153.56MiB, used=112.00KiB
GlobalReserve, single: total=16.00MiB, used=0.00B
```

```

root@so2022:/mnt# btrfs fi usage /mnt/disco5
Overall:
  Device size:                3.00GiB
  Device allocated:           643.12MiB
  Device unallocated:         2.37GiB
  Device missing:             0.00B
  Used:                       384.00KiB
  Free (estimated):           2.68GiB   (min: 1.50GiB)
  Data ratio:                 1.00
  Metadata ratio:             2.00
  Global reserve:             16.00MiB   (used: 0.00B)

```

```

Data,single: Size:320.00MiB, Used:128.00KiB
/dev/sda9    320.00MiB

```

```

Metadata,DUP: Size:153.56MiB, Used:112.00KiB
/dev/sda9    307.12MiB

```

```

System,DUP: Size:8.00MiB, Used:16.00KiB
/dev/sda9    16.00MiB

```

```

Unallocated:
/dev/sda9    2.37GiB

```

El alocado es 642.12 MiB, el usado el 384.00 KiB

6. Generar un archivo de 2000MB en el directorio /disco5 (dd if=/dev/zero of=/disco5/so1 bs=100M count=20). Analizar nuevamente el espacio alocado/ocupado. (Obs.: puede que tenga que esperar un tiempo para ver los cambios en la salida de los comandos btrfs....). ¿Cómo quedó el espacio asignado y el utilizado tanto de los datos como de los metadatos?

```

root@so2022:/mnt# btrfs fi usage /mnt/disco5
Overall:
  Device size:                3.00GiB
  Device allocated:           2.82GiB
  Device unallocated:         188.88MiB
  Device missing:             0.00B
  Used:                       1.96GiB
  Free (estimated):           748.31MiB   (min: 653.88MiB)
  Data ratio:                 1.00
  Metadata ratio:             2.00
  Global reserve:             16.00MiB   (used: 0.00B)

```

```

Data,single: Size:2.50GiB, Used:1.95GiB
/dev/sda9    2.50GiB

```

```

Metadata,DUP: Size:153.56MiB, Used:2.33MiB
/dev/sda9    307.12MiB

```

```

System,DUP: Size:8.00MiB, Used:16.00KiB
/dev/sda9    16.00MiB

```

```

Unallocated:
/dev/sda9    188.88MiB

```

Ahora el alocado es 2.82GiB y el usado es 1.96GiB, 2.50GiB alocado 1.95GiB usado por datos y 153.56MiB alocado 2.33MiB usados por metadatos.

7. Asignar la otra partición a /disco5. ¿Se modificaron los valores con respecto al punto anterior?

```
root@so2022:/mnt# btrfs device add /dev/sda10 /mnt/disco5
```

```
Overall:
  Device size:                6.00GiB
  Device allocated:           2.82GiB
  Device unallocated:         3.18GiB
  Device missing:              0.00B
  Used:                        1.96GiB
  Free (estimated):           3.73GiB   (min: 2.14GiB)
  Data ratio:                  1.00
  Metadata ratio:              2.00
  Global reserve:             16.00MiB   (used: 0.00B)

Data,single: Size:2.50GiB, Used:1.95GiB
/dev/sda9      2.50GiB

Metadata,DUP: Size:153.56MiB, Used:2.33MiB
/dev/sda9      307.12MiB

System,DUP: Size:8.00MiB, Used:16.00KiB
/dev/sda9      16.00MiB

Unallocated:
/dev/sda10      3.00GiB
/dev/sda9      188.88MiB
```

8. Generar otro archivo de 3000MB en el directorio /disco5. ¿Aumenta el espacio alocado? ¿Cuánto espacio se ha ocupado realmente?

```
Overall:
  Device size:                6.00GiB
  Device allocated:           5.69GiB
  Device unallocated:         322.00MiB
  Device missing:              0.00B
  Used:                        2.94GiB
  Free (estimated):           2.75GiB   (min: 2.60GiB)
  Data ratio:                  1.00
  Metadata ratio:              2.00
  Global reserve:             16.00MiB   (used: 0.00B)

Data,single: Size:5.37GiB, Used:2.93GiB
/dev/sda10      3.00GiB
/dev/sda9      2.37GiB

Metadata,DUP: Size:153.56MiB, Used:3.48MiB
/dev/sda9      307.12MiB

System,DUP: Size:8.00MiB, Used:16.00KiB
/dev/sda9      16.00MiB

Unallocated:
/dev/sda10      1.00MiB
/dev/sda9      321.00MiB
```

9. Usando las dos particiones anteriores crear un RAID1 y montarlo en /disco5. ¿Qué partición puede elegir para montar el file system? (Desmontar previamente la partición montada en /disco5.)

```
mkfs.btrfs -d raid1 -m raid1 /dev/sda9 /dev/sda10
```

Se puede elegir cualquiera de las dos particiones del pool ya que va a montar el file system en todas las particiones del mismo.

10. ¿Es posible generar los dos archivos anteriores en ese filesystem? ¿Por qué?

No porque RAID 1 no aumenta la capacidad total.

11. Eliminar todo el contenido de /disco5 y generar dos subvolumenes, llamados vol1 y vol2. ¿Puede ver los subvolumenes creados? ¿Qué ID tiene cada uno? ¿Qué significa el ID 5?

```
root@so2022:~# btrfs subvolume create /mnt/disco5/vol1
Create subvolume '/mnt/disco5/vol1'
root@so2022:~# btrfs subvolume create /mnt/disco5/vol2
Create subvolume '/mnt/disco5/vol2'

root@so2022:~# btrfs subvolume list /mnt/disco5
ID 262 gen 16 top level 5 path vol1
ID 263 gen 17 top level 5 path vol2
```

5 es la id del subvolumen raíz.

```
root@so2022:/mnt# btrfs subvolume show /mnt/disco5
/
      Name:                <FS_TREE>
      UUID:                 3569c181-689b-46c1-bae0-bca6aa2f69c5
      Parent UUID:         -
      Received UUID:       -
      Creation time:       2024-05-06 19:18:20 -0300
      Subvolume ID:        5
      Generation:          17
      Gen at creation:     0
      Parent ID:           0
      Top level ID:        0
      Flags:               -
      Snapshot(s):
```

12. Montar esos volúmenes, vol1 y vol2, en los directorios /volumen1 y /volumen2 respectivamente. ¿Qué espacio disponible tiene cada volumen? ¿Es posible acotar el espacio de un volumen? ¿Es necesario que esté montado el subvolumen top-level para poder montar sus subvolumenes?

```
root@so2022:/mnt# mount -o subvol=vol1 /dev/sda9 /mnt/volumen1
root@so2022:/mnt# mount -o subvol=vol2 /dev/sda9 /mnt/volumen2
```

```

root@so2022:/mnt# df -h /mnt/volumen1
S.ficheros      Tamaño Usados  Disp Uso% Montado en
/dev/sda9        3,0G    17M   2,7G   1% /mnt/volumen1
root@so2022:/mnt# df -h /mnt/volumen2
S.ficheros      Tamaño Usados  Disp Uso% Montado en
/dev/sda9        3,0G    17M   2,7G   1% /mnt/volumen2

```

Si, es posible acotar el espacio de un volumen configurando “cuotas” para limitar el espacio ocupado por los subvolumenes y sus archivos. Se puede hacer usando btrfs quota.

1) Habilitar quota:

- btrfs quota enable /ruta/al/directorio

2) Después de habilitar las cuotas, puedes usar el comando btrfs qgroup para administrarlal:

- btrfs qgroup show /ruta/al/directorio: Muestra información sobre las cuotas de grupo en el directorio especificado.
- btrfs qgroup limit <limite> /ruta/al/subvolumen: Establece un límite en el uso de espacio para el subvolumen especificado.
- btrfs qgroup assign <id_de_cuota> /ruta/al/subvolumen: Asigna un grupo de cuotas a un subvolumen.

1) Habilitar quota: habilitar las cuotas en el file system Btrfs.

- btrfs quota enable /ruta/al/directorio

2) Crear un nuevo grupo de cuotas (opcional): para organizar y gestionar mejor el uso de espacio en disco.

- btrfs qgroup create "IDGrupoNuevo" "/ruta/raíz/grupo/"

3) Asignar subvolumenes al grupo de cuotas (opcional): si se creó un nuevo grupo de cuotas, se puede asignar subvolumenes a este grupo.

- btrfs qgroup assign "IDGrupoNuevo" "/ruta/al/subvolumen"

4) Establecer límites de cuotas: se establece el límite en el uso de espacio para los grupos de cuotas o subvolumenes específicos.

- btrfs qgroup limit <limite> "/ruta/al/subvolumen"

5) Verificar el uso de cuotas: se utiliza el comando btrfs qgroup show para verificar el uso de cuotas en el file system Btrfs.

- btrfs qgroup show /ruta/al/directorio

No es necesario que este montando el subvolumen top-level. Cada subvolumen se trata como una entidad independiente y se puede montar individualmente.

13. Generar un archivo de 300MB en el directorio /disco5/vol1, ¿es posible ver el archivo en /volumen1? Si ejecuta el comando `df -h`, ¿qué espacio se ha consumido disco5, volumen1 y volumen2? ¿Por qué sucede esto?

```
root@so2022:/mnt/disco5# du -h /mnt/volumen1
300M    /mnt/volumen1
```

Si es posible verlo.

```
/dev/sda9      3,0G    317M    2,4G    12% /mnt/disco5
/dev/sda9      3,0G    317M    2,4G    12% /mnt/volumen1
/dev/sda9      3,0G    317M    2,4G    12% /mnt/volumen2
```

Han consumido lo mismo. `df` no es capaz de detectar los subvolumenes BTRFS.

14. Limitar el tamaño del subvolumen volumen2 a 300MB. Intentar copiar un archivo de 400MB, ¿es posible hacerlo? (Hint: `btrfs quota ...`)

```
root@so2022:/mnt# btrfs quota enable /mnt/disco5
root@so2022:/mnt# btrfs qgroup limit 300M /mnt/volumen2

root@so2022:/mnt# btrfs quota enable /mnt/disco5
root@so2022:/mnt# btrfs qgroup limit 300M /mnt/volumen2
root@so2022:/mnt# dd if=/dev/zero of=/mnt/volumen2/sol bs=100M count=4
dd: error al escribir en '/mnt/volumen2/sol': Se ha excedido la cuota de disco
3+0 registros leídos
2+0 registros escritos
314310656 bytes (314 MB, 300 MiB) copied, 0.292081 s, 1.1 GB/s
```

No se puede

15. Elevar el tamaño de la cuota a 450MB, ¿es posible ahora? (Previamente revisar si el volumen está vacío.)

```
root@so2022:/mnt# btrfs qgroup limit 450M /mnt/volumen2
root@so2022:/mnt# rm -r /mnt/volumen2/*
root@so2022:/mnt# dd if=/dev/zero of=/mnt/volumen2/sol bs=100M count=4
4+0 registros leídos
4+0 registros escritos
419430400 bytes (419 MB, 400 MiB) copied, 0.257424 s, 1.6 GB/s
```

Ahora si

16. Realizar un snapshot del subvolumen /disco5/vol1 en /disco5/snap. Antes de esto crear un archivo con el texto **Esto es una prueba de un snapshot** y otro archivo de 100MB. Chequear, antes y después de generar el snapshot, con `df -h` y los comandos de `btrfs` el espacio alocado y consumido. ¿Se incrementó el espacio consumido? ¿Por qué?

```

root@so2022:/mnt# mkdir /mnt/disco5/snap
root@so2022:/mnt# dd if=/dev/zero of=/mnt/disco5/vol1/sol bs=100M count=1
1+0 registros leídos
1+0 registros escritos
104857600 bytes (105 MB, 100 MiB) copied, 0,0724885 s, 1,4 GB/s
root@so2022:/mnt# touch /mnt/disco5/vol1/prueba.txt
root@so2022:/mnt# echo "Esto es una prueba de un snapshot" > /mnt/disco5/vol1/pr
ueba.txt
root@so2022:/mnt# df -h /mnt/disco5/vol1
S.ficheros      Tamaño Usados  Disp Uso% Montado en
-              3,0G   518M   2,3G   19% /mnt/disco5/vol1

root@so2022:/mnt# df -h /mnt/disco5
S.ficheros      Tamaño Usados  Disp Uso% Montado en
/dev/sda9      3,0G   518M   2,3G   19% /mnt/disco5

root@so2022:/mnt# btrfs subvolume snapshot /mnt/disco5/vol1 /mnt/disco5/snap
Create a snapshot of '/mnt/disco5/vol1' in '/mnt/disco5/snap/vol1'

root@so2022:/mnt# df -h /mnt/disco5
S.ficheros      Tamaño Usados  Disp Uso% Montado en
/dev/sda9      3,0G   518M   2,3G   19% /mnt/disco5

```

No se incrementa el espacio consumido por /mnt/disco5 debido a que todavía no se modificó nada, entonces la snapshot y el subvolumen apuntan a los mismos bloques.

17. Modificar el contenido el archivo original agregándole **para sistemas operativos**.

¿Se modifica la copia en el snapshot?

No, no se modificó.

```

root@so2022:/mnt# echo "Para sistemas operativos" >> /mnt/disco5/vol1/prueba.txt
root@so2022:/mnt# cat /mnt/disco5/snap/vol1/prueba.txt
Esto es una prueba de un snapshot

```

18. Si se desea volver al subvolumen original, ¿cómo lo haría? (sin hacer un copy o move de los archivos)

umount /mnt/volumen1

rsync -avz /mnt/disco5/snap/vol1 /mnt/disco5/vol1.

mount -o subvol=vol1 /dev/sda9 /mnt/volumen1