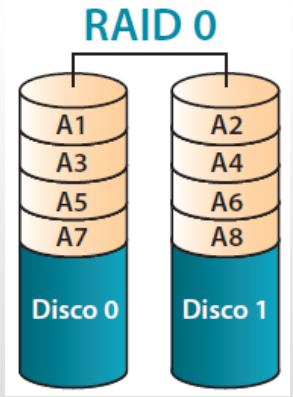
**RAID**

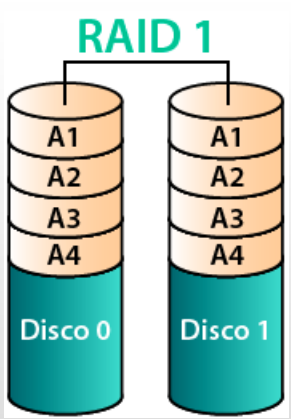
*En este punto se va a crear un RAID Level 5 por software. Para simular los 3 discos necesarios para un RAID de este nivel se deberán generar 3 particiones del mismo tamaño. Una vez realizado este paso se generará el RAID. Para esto se utilizará la herramienta mdadm que deberá instalarla en la VM provista por la cátedra usando el siguiente comando: apt-get install mdadm.*

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

RAID - Redundant Array of Independent Disks (originalmente llamado Redundant Array of Inexpensive Disks).

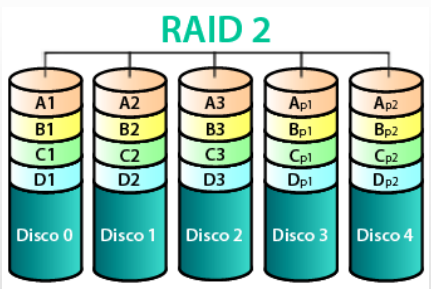
RAID es una técnica que permite usar múltiples discos en forma conjunta con el fin de construir un sistema de discos más rápido, más grande y más confiable.

**Level 0(Striping):** No es un nivel de RAID en absoluto. No existe redundancia. Array de discos con “striping” a nivel de bloque. Necesita de 2 o más discos para conformarse. La capacidad es la sumatoria de la capacidad de los discos participantes. Si falla un disco, los datos de todos los discos se vuelven inaccesibles. Sirve como “upper-bound” en cuanto a performance y capacidad. Disk Striping: proceso de dividir datos en bloques y esparcirlos en múltiples dispositivos de almacenamiento.



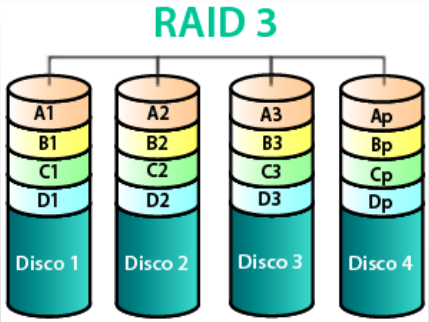
**Level 1 (Mirroring):**

Asegura la redundancia mediante el mirroring (espejado) de datos. No hay striping de datos. Almacena datos duplicados en discos separados o independientes. Se requiere un mínimo de 2 discos. Trabaja con discos de a pares. Ineficiente por la escritura en espejos que provoca un desperdicio del 50 % de la capacidad total. Desperdicio del 50 % de la capacidad total.

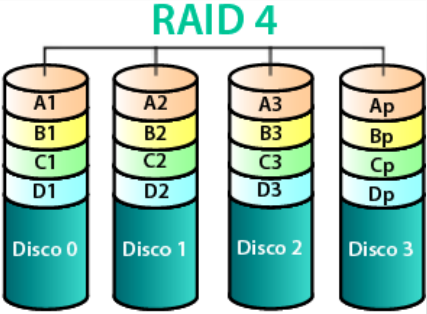


**Level 2 :**

Striping a nivel de bit y paridad Hamming dedicada. Se requieren al menos 3 discos como mínimo. Buena protección de datos en caso de fallas de disco. Tasa de transferencia de datos puede llegar a ser muy elevada. Disco dedicados para almacenar información de verificación y corrección de errores. No muy aceptado por la industria.

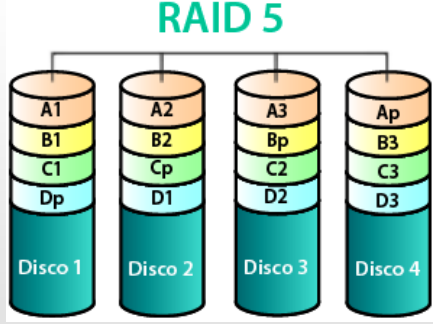
**Level 3:**

Striping a nivel de byte y paridad dedicada. Disco “dedicado” para almacenar la información de paridad. Datos se escriben en paralelo entre los disco del array. 3 discos como mínimo. Alta tasa de transferencia en escritura y lectura. El disco de paridad puede convertirse en cuello de botella. No ofrece solución a fallo simultáneo de discos.



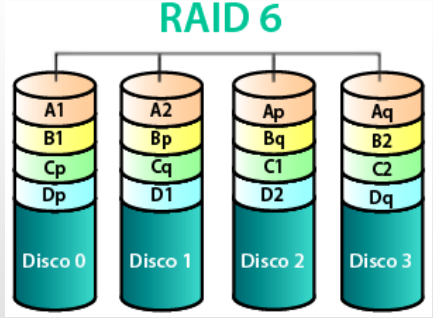
**Level 4:**

Striping a nivel de bloque y paridad dedicada. 3 disco requeridos como mı́nimo. Alta tasa de transferencia de datos. No muchas implementaciones disponibles.



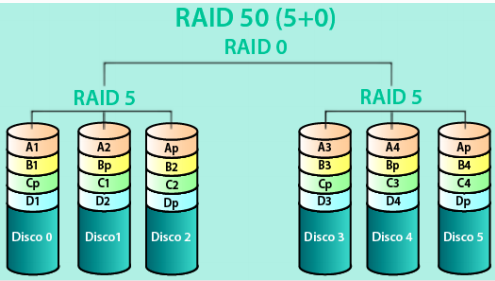
**Level 5:**

Striping a nivel de bloque y paridad distribuida. Es una de las implementaciones más utilizadas. Distribuye la información de paridad entre todos los discos del array. Se necesitan 3 discos como mínimo. Alto rendimiento. No hay cuello de botella. No ofrece solución al fallo simultáneo de discos.



**Level 6:**

Striping a nivel de bloque y doble paridad distribuida. Recomendado cuando se tienen varios discos. Requiere 4 discos mínimo. Alta tolerancia a fallos (hasta dos discos).Operaciones de escrituras más lentas debido al cálculo de la doble paridad.

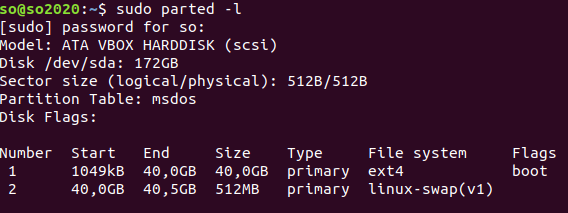


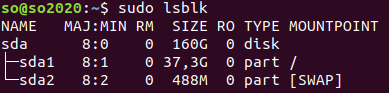
Además de estos niveles de RAID es posible utilizar niveles híbridos los cuales consisten en la combinación entre los ya nombrados.

Como por ejemplo:

*2. Utilizar el comando parted -l (debe ejecutarlo con sudo) para ver la tabla de particiones.*

**





*Conteste:*

*a) ¿Cómo llama Linux al dispositivo físico?*

El dispositivo físico se llama: sda, sus particiones sda1 y sda2(swap)

*¿Cuál es su tamaño total?*

El tamaño total es de: 172 GB según el comando parted -l

*b) ¿Cuántas particiones existen?*

Existen 2 particiones, sda1 y sda2

*¿Qué tipo de file-system contiene cada una?*

El tipo de file-system es ext4 para sda1 y linux-swap(v1) para sda2

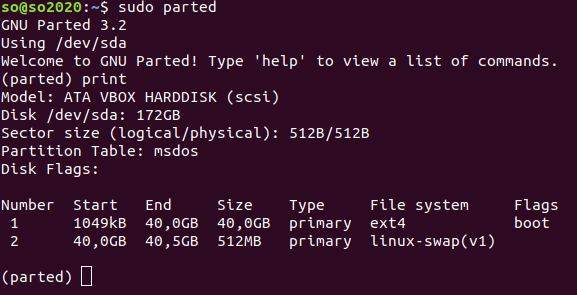
*c) ¿Qué significa la bandera arranque?*

La bandera boot es un valor de 1 byte en un registro de partición no extendido, dentro del master boot record. Aparece al comienzo de un registro de partición, como el valor 0x80. Un valor de 0x00 indica que la partición no tiene establecido el indicador de arranque. Su función principal es indicar al boot loader qué particion arrancar.

*3. Usando el comando parted crear una nueva partición de tipo extendida, si es que no existe previamente (debe seleccionar el dispositivo donde se van a generar las particiones):*

*1) sudo parted*

*2) (parted) print*

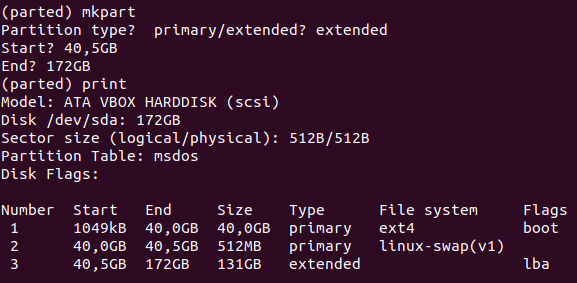


*3) (parted) mkpart*

*4) Tipo de partición: extendida*

*5) Inicio: #MB (igual al tamaño final de la partición existente más alta*

*6) Fin: #MB (igual al tamaño máximo del dispositivo físico)*

**

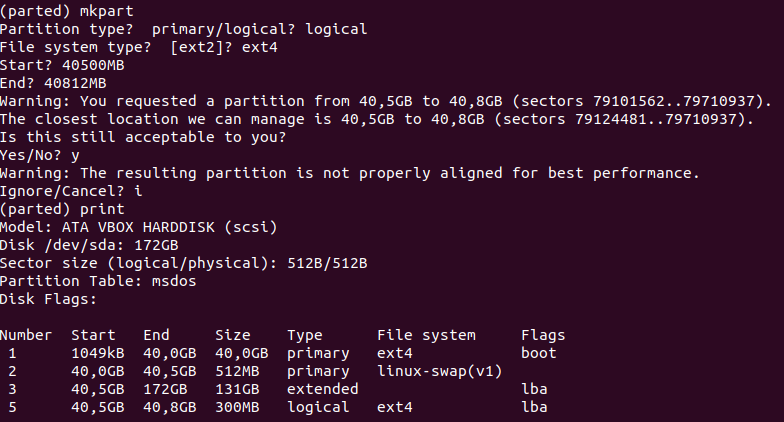
*Dentro de esta partición extendida crear 3 nuevas particiones de 300MB cada una. Para esto utilizar nuevamente el comando mkpart, pero debe seleccionar logical como tipo de partición.*

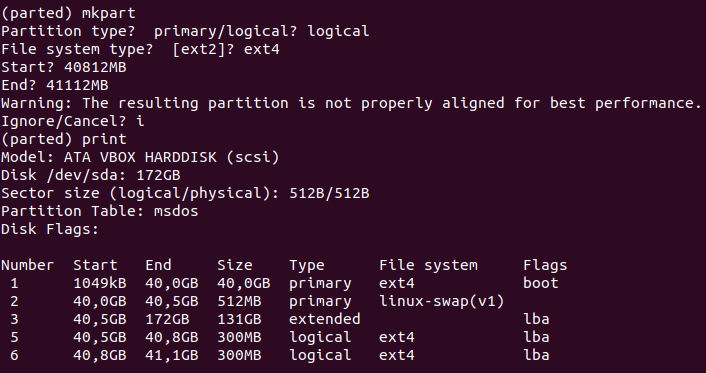
*Como tipo de sistema de archivos elija ext4.*

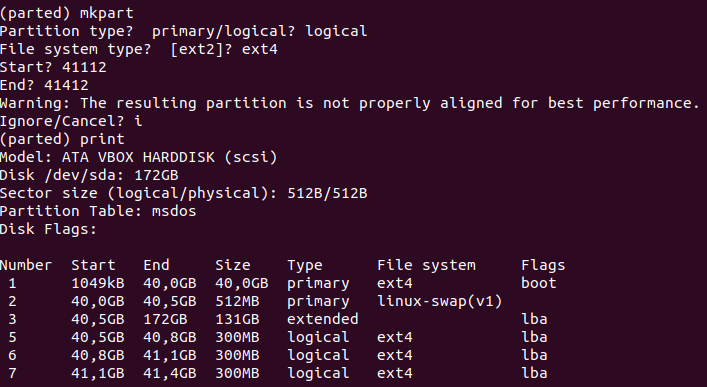
*Obs.: como las particiones lógicas se crean dentro de una partición extendida, los valores*

*de inicio/fin de cada partición lógica deben estar dentro de los valores seleccionados en la*

*partición extendida.*

**





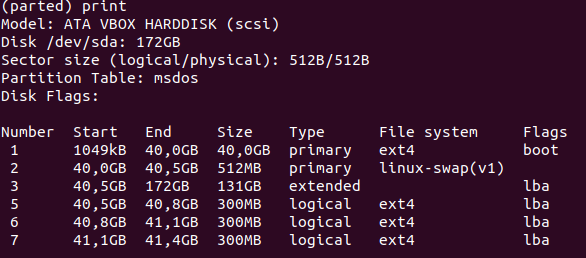
*4. ¿Por qué es necesario crear una partición extendida? Si se usase GPT, ¿sería necesario este tipo de particiones?*

La cantidad de particiones primarias (4) son insuficiente al utilizar **MBR**.

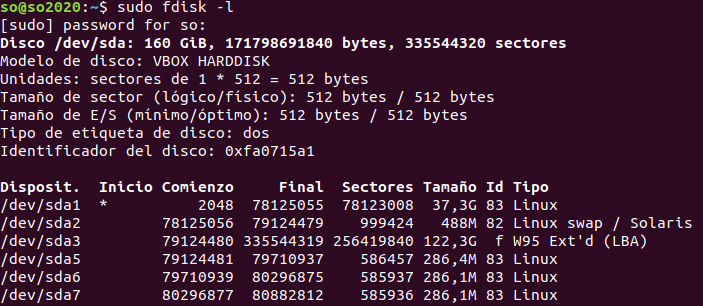
Por esto se crea una partición extendida la cual puede contener varias particiones lógicas. El disco quedaría con 3 primarias y una extendida. Dentro de esta partición extendida puede haber un gran número de particiones lógicas.Y de esta forma romper la limitación de solo 4 primarias.

En **GPT** no son necesarias las particiones lógicas ya que se pueden realizar 128 particiones por defecto. Además gpt soporta discos mucho más grandes qué MBR; cuenta con un backup de la tabla al final del disco y CRC para proteger el contenido de toda la GPT.

*5. Mirar nuevamente la tabla de particiones para ver las nuevas particiones y salga del entorno parted*

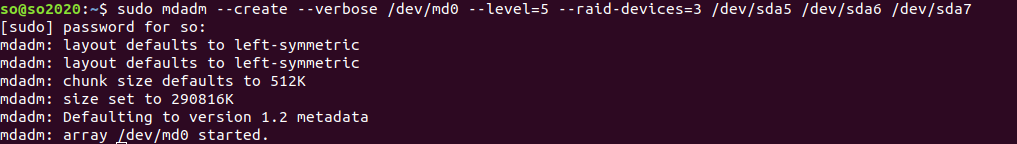
**

*6. 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)*

**

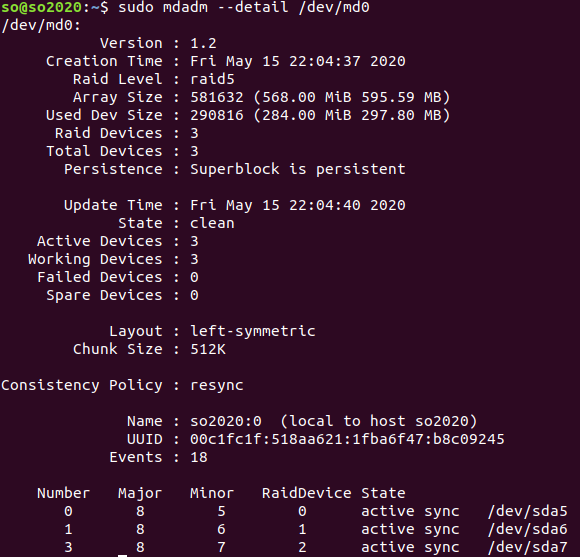
*7. ¿Qué significan los valores sda5, sda6 y sda7?*

Es la nomenclatura para diferenciar las particiones sda es sata y corresponde al

disco, sda5 sda6 sda7 corresponden a las particiones dentro de sda.

*8. Ejecutar la siguiente consulta y contestar:*

*mdadm --detail /dev/md0*

**

*a) ¿Cuál es el tamaño total del RAID?*

Lo que hay disponible para utilizar (595.59MB) más lo que ya fue utilizado para la información de paridad del RAID 5 (297.80MB), por lo tanto el tamaño total del raid es 893.39MB.

*b) ¿Cuál es el tamaño total utilizable para almacenar datos?*

La cantidad real de almacenamiento disponible no es igual a la suma de todas las unidades del raid. La información de paridad del RAID 5 que permite la tolerancia a fallos también se almacena y ocupa el equivalente a una unidad de espacio de almacenamiento, lo que reduce la cantidad de espacio de almacenamiento disponible por el tamaño de un disco duro. Used Dev Size indica el valor utilizado como base para armar el RAID. Lo que tiene disponible para almacenar datos, es de el doble de Used Dev Size

Used dev size \* 2 = 297.80\*2 = 595.60MB como dice en Array Size.

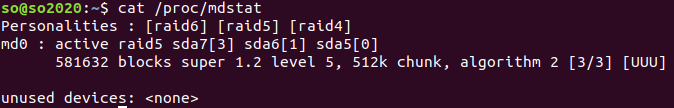
*9. Analizar el contenido del siguiente comando:*

*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).*

Este archivo contiene la información actual sobre las configuración de discos múltiples de RAID. Con /proc/mdstat se ve el estado actual de los dispositivos RAID mdX.

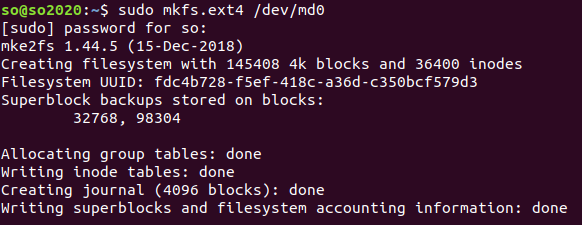
El archivo /proc/mdstat, muestra un sistema con su md0 configurado como un dispositivo RAID 5, mientras está re-sincronizando los discos:

**

*10. 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*

mount /dev/md0 /mnt/rd5/

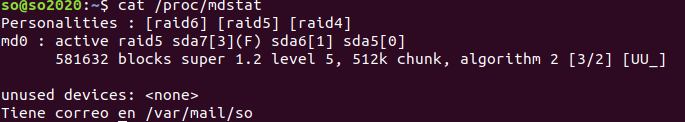
*c) Crear un directorio con dos archivos*

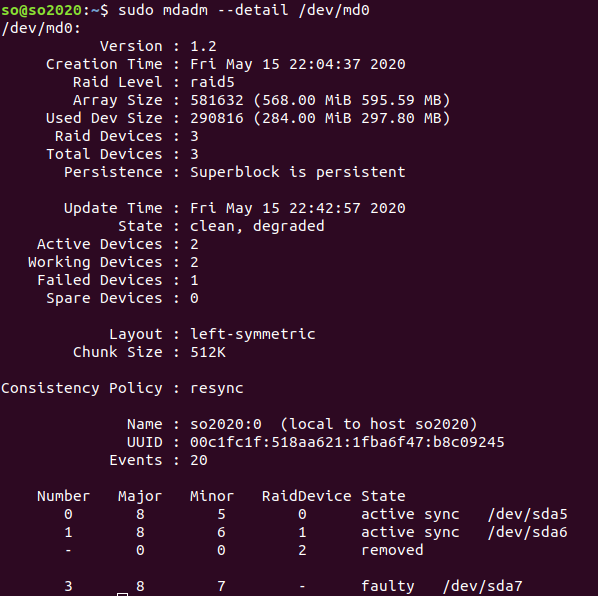


*d) Quitar una de las particiones del RAID. Para esto ponemos uno de los componentes en falla:*

*mdadm --fail /dev/md0 /dev/sda7*

*e) Observar el estado del RAID y contestar*

**

**

*1) ¿Cuál es el estado del RAID?*

El estado es Clean, degraded, antes era Clean.

*¿Cuántos dispositivos activos existen?*

Ahora existen 2 dispositivos activos, antes eran 3.

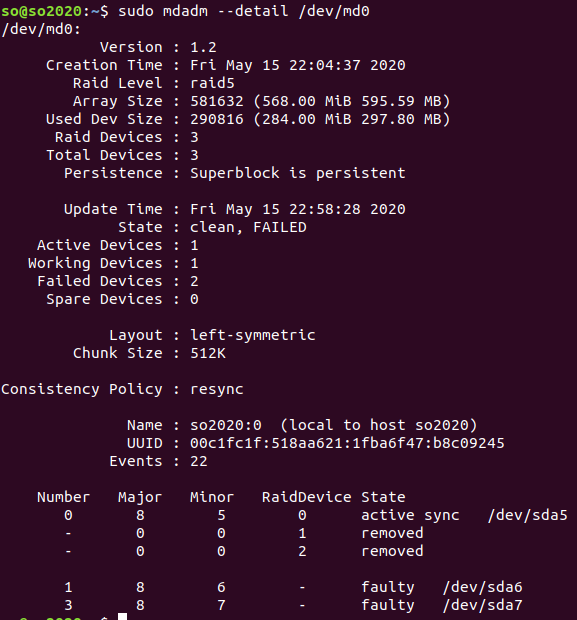
*2) El tamaño total y disponible del RAID, ¿se modificó?*

Los tamaños no se modificaron

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

sudo mdadm --fail /dev/md0 /dev/sda6

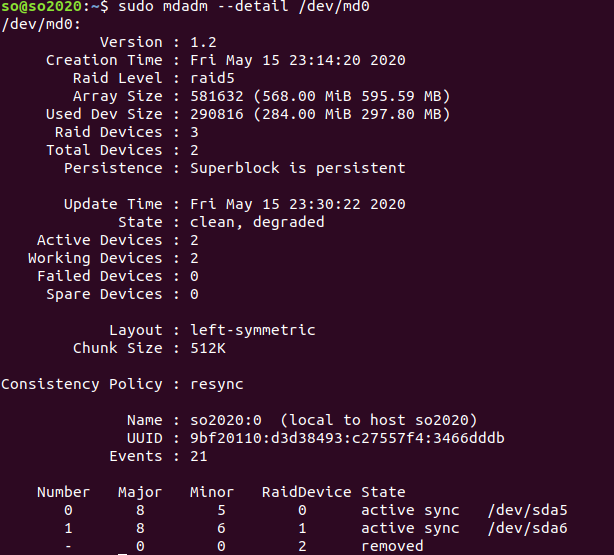
El estado cambia a Clean, FAILED, el raid falla, ya que solo hay una partición activa.



*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 acceder y si están creados los archivos.



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

Si hubiésemos tenido una partición hot-spare hubiera reemplazado a la que se puso en fail.

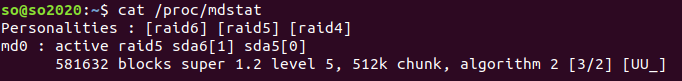
*h) Por último, remover la partición permanentemente del RAID (Obs.: esto es muy importante para que el próximo booteo mdadm no intente usar a esta partición como parte*

*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.*

**

**

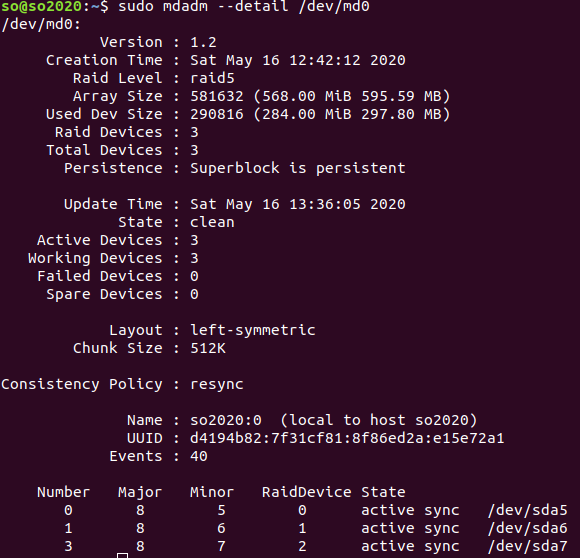
*11. Para evitar la pérdida de datos es fundamental volver al RAID a un estado estable (sacarlo del estado degradado). Para esto se agregará nuevamente la partición /dev/sda7 que se quitó en el paso anterior*

*a) Ejecutar el comando mdadm –add /dev/md0 /dev/sda7*

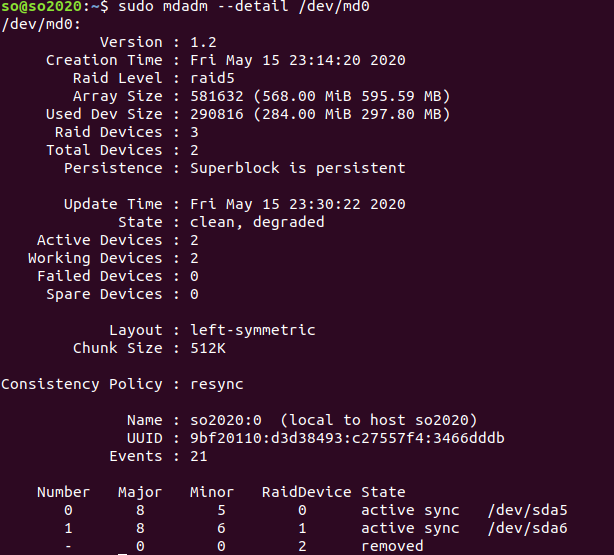
sudo mdadm /dev/md0 --add /dev/sde



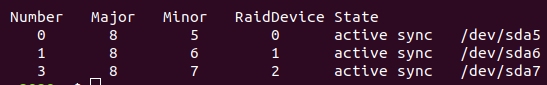
*b) Ejecutar el comando mdadm –detail /dev/md0*



*c) ¿Qué hace el RAID con la nueva partición recientemente agregada?*

**

Una vez que la vuelve a agregar desaparece el state removed y se vuelven a agregar los valores para esa partición



*¿Qué significa el estado “Rebuild Status”?*

El rebuild status informa que se está reconstruyendo los datos en la partición,

a partir de la información de las otras particiones.

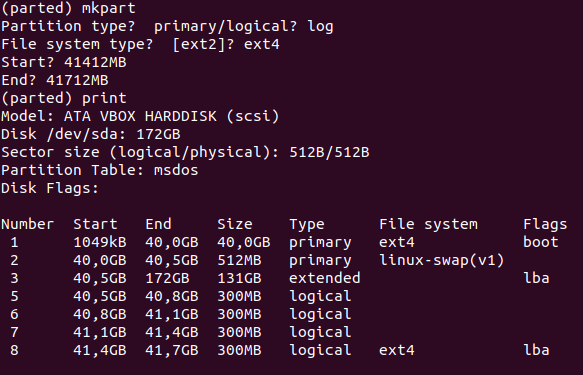
*d) ¿Es posible ingresar al recurso /mnt/rd5? ¿Se encuentran disponibles los datos creados en el punto anterior?*

Si, es posible seguir ingresar al recurso /mnt/rd5 y se encuentran disponibles los datos.



*12. 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”.*

*13. Usando el comando parted generar una nueva partición, /dev/sda8 (con igual tamaño a las anteriores).*

**

*14. Agregar la nueva partición al RAID:*

*mdadm --add /dev/md0 /dev/sda8*

**

*a) ¿Cómo se agregó la nueva partición?*

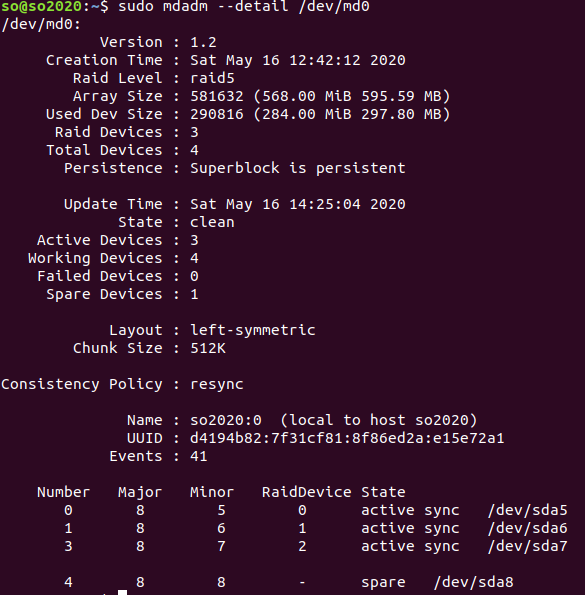
La nueva partición se agrega como “spare” se puede ver en el detalle el state

de sda8.

*¿Por qué?*

Se agrega como spare, debido que en raid 5 se necesitan mínimo 3 particiones, la 4 va a ser spare.

El hot-spare es una unidad que actúa como una unidad de reserva en el RAID. Es una unidad completamente funcional que no contiene datos y no se utiliza durante el funcionamiento normal. Si falla una unidad del volumen, el controlador reconstruye los datos de la unidad fallida a la unidad de hot-spare.

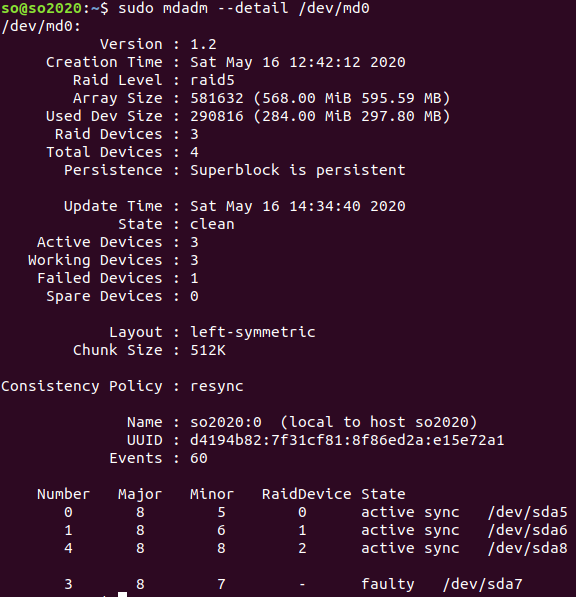
**

*15. 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?*

La vuelve como activa y en sincronización con las otras 2.

La partición sda8 que estaba como spare ahora pasa a reemplazar la sda7 que fue marcada como fail.

*16. 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 cómo se comporta*

**

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

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

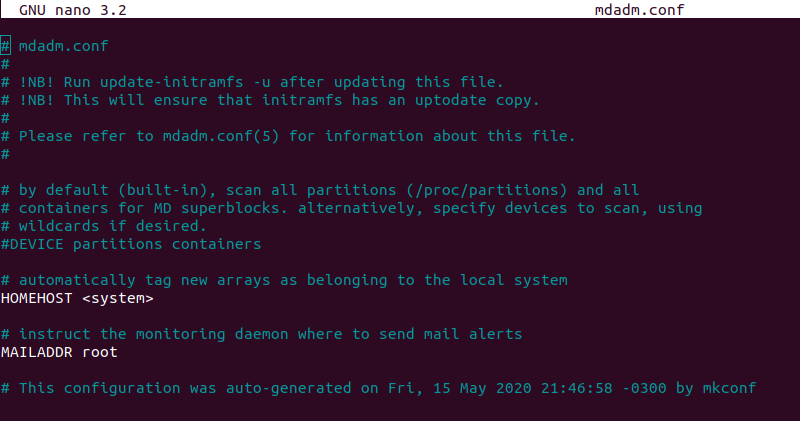
*d) Remover el RAID*

*mdadm --remove /dev/md0*

**

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

No existía la linea ARRAY

**

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



**LVM (Logical Volume Management)**

*A continuación se creará un LVM utilizando las particiones /dev/sda5, /dev/sda6 y /dev/sda7 (respetar el tamaño y nombre de los volúmenes y directorios). En principio solo se utilizarán las particiones 5 y 6, luego se agregará la partición 7 para incrementar el tamaño de los volúmenes.*

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

Logical Volume Manager es un método de localización del espacio disco duro en volúmenes lógicos que pueden ser fácilmente redimensionados en vez de particiones. Con LVM, el disco duro o grupo de discos duros está localizado para uno o más volúmenes físicos. Un volumen físico no abarca más de una unidad.

LVM permite una capa de abstracción entre el sistema operativo y las particiones/discos que utiliza. En la administración de discos tradicionales de un sistema operativo se busca que los discos esten disponibles en (/dev/sda, /dev/sdb, etc) y luego mira que las particiones están disponibles en los discos (/dev/sda1, /dev/sda2, etc .). Con LVM, los discos y particiones se puede abstraer de contener varios discos y particiones en un solo dispositivo. Los sistemas operativos no notará la diferencia, porque LVM sólo le mostrará al sistema operativo el grupo de volúmenes(discos) y los volúmenes lógicos (particiones) que han establecido.

Debido a que los grupos de volúmenes y volúmenes lógicos no están físicamente atados a una unidad de disco duro, que hace que sea fácil de cambiar dinámicamente el tamaño y crear nuevos discos y particiones. Además, LVM le puede dar las características que el sistema de archivos no es capaz de hacer. Por ejemplo, Ext3 no tiene soporte para snapshots en vivo, pero si se está utilizando LVM estos tienen la capacidad de tomar un snapshot de los volúmenes lógicos sin necesidad de desmontar el disco.

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

Una característica muy interesante de LVM es que nos permite realizar snapshots de volúmenes lógicos. Estos snapshots pueden ser utilizados para revertir el estado inicial y conocido de un LV. Los snapshots se pueden utilizar a modo de backup.

En el LVM 1, el snapshot usa COW (copy-on-write). En el momento que se crea el snapshot este tiene hard-links a los inodos de la data real. Así que mientras los datos no cambien el snapshot solo tiene apuntadores pero no los datos en su interior, pero si los datos se modifican en el Logical Volume original, el snapshot automáticamente clona la información, ésta antes de ser modificada queda en el snapshot y los datos nuevos en el sistema de archivos activo.

LVM 2 los snapshot funcionan como en LVM 1, pero con la característica de poder escribir datos, si los datos se escriben en el snapshot que se encuentre en un punto de montaje del sistema, ese bloque se marca como usado en el snapshot y no se copia nada al volumen original.

*3. Instalar la herramienta lvm2: apt-get install lvm2*

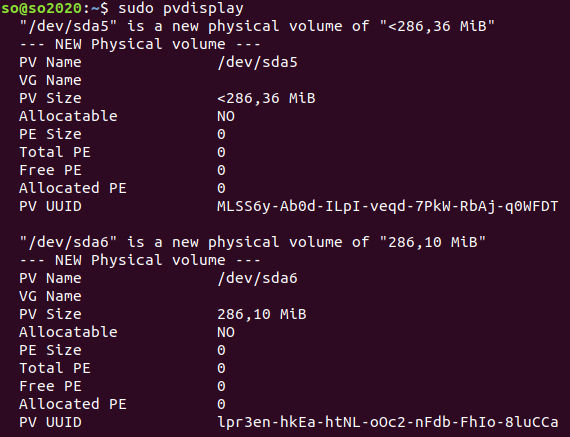
*4. Ejecutar el siguiente comando, ¿qué es lo que realiza?*

*pvcreate /dev/sda5 /dev/sda6*



La forma de entregar un dispositivo al LVM es marcarlo como un Physical Volume (PV), por lo tanto con pvcreate estamos haciendo esto mismo.

*5. Mediante el comando pvdisplay observar el estado del volumen físico recientemente creado*



*6. Crear un grupo de volúmenes (volume group, VG) llamado “so\_X”, donde X es el número de grupo asignado.*

*vgcreate so\\_52 /dev/sda5 /dev/sd6*

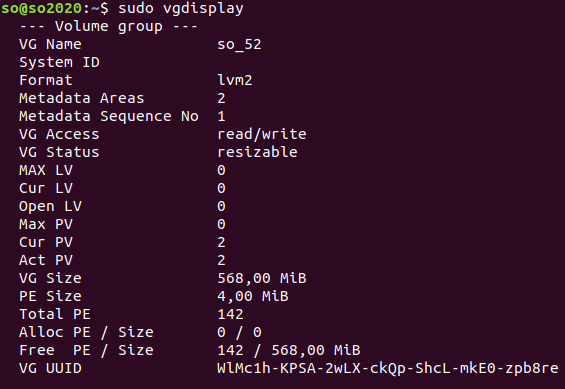
**

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

*¿Cuál es tamaño total del VG?* 568 MiB

*¿Qué significa PE?* Las Physical Extensions (PE) son las unidades de asignación mínima del LVM. Por defecto son de 4MB (aunque se puede cambiar en la creación del VG). Es decir, cada vez que creemos un LV lo haremos en múltiplos de (por defecto) 4MB.

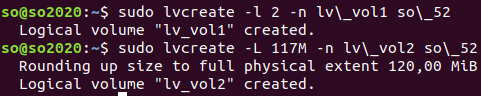
*¿Qué define?* ​El tamaño de bloque de los extends, que son bloques contiguos del disco que se usan para prevenir fragmentación.

**

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

*lvcreate -l 2 -n lv\\_vol1 so\\_52*

*lvcreate -L 117M -n lv\\_vol2 so\\_52*

**

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

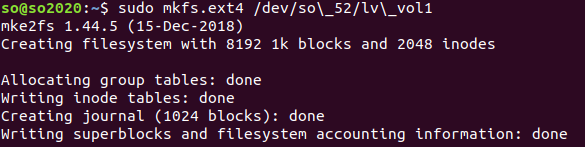
Para crear un LV, con el comando lvcreate, debemos indicarle el VG al que pertenece (como argumento del comando), el tamaño (en PEs con la opción -l o en megas/gigas/teras con -L) y opcionalmente, pero muy recomendable, el nombre que queremos darle (-n).

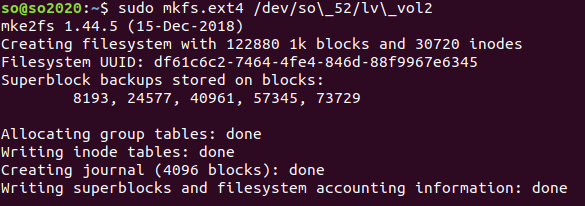
La diferencia es que en el primer comando se le pasa el valor en PEs con -l y en la segunda opción con un valor en MB con -L para asignar un volúmen al LV.

*10. ¿Con qué tamaño se generó el LV lv\_vol2? ¿Por qué?*

Se creó con el tamaño 117MB porque fue especificado en el parámetro -L

*11. Formatear los dos LV generados en el paso anterior con un file system de tipo ext4: mkfs.ext4 /dev/so\\_52/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)*

**

**

*sudo mount /dev/so\_52/lv\_vol1 /mnt/vol1/*

*sudo mount /dev/so\_52/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)*

**

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

**

No es posible, porque no queda espacio en el dispositivo.

Cuando nuestros LVs han consumido todo el espacio del VG no podremos seguir creciendo éstos cuando lo necesitemos. Para solventar esta situación podemos ampliar el LV con el comando lvextend.

*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*

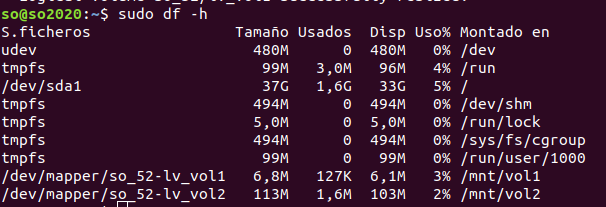
*lvextend -L +20M /dev/so\\_52/lv\\_vol1*



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

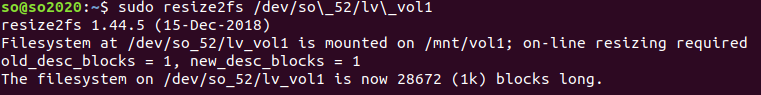
Porque lo que hicimos fue aumentar el tamaño lógico, pero el sistema de archivos aún no sabe del cambio.

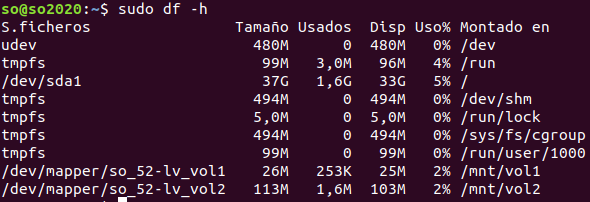
Una vez ampliado el LV debemos ampliar el sistema de ficheros. Si nos fijamos, aunque hemos ampliado el LV, el espacio disponible en el sistema de ficheros siguen siendo el mismo.



*c) Incrementar el tamaño del file system:*

*resize2fs /dev/so\\_52/lv\\_vol1*

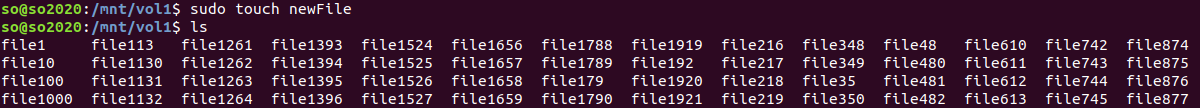
**

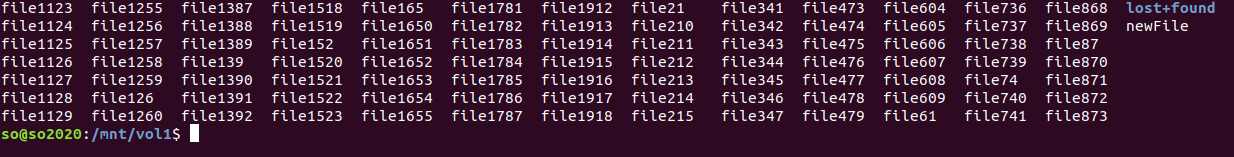
**

*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 se pudo crear un archivo nuevo porque ampliamos el espacio del LV vol1.**

**

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

No, no hay suficiente espacio, redimensionando la partición sda7 ya que la sda5 y sda6 aunque llegan a los 500M, al haber utilizado 148 M no alcanza.

Cuando los LVs han consumido todo el espacio del VG no se puede seguir aumentando el creciendo de los LVs o agregar otro cuando lo necesitemos. Para solventar esta situación podemos ampliar el VG añadiendo un PV nuevo. El comando vgextend es muy similar al de creación (vgcreate), simplemente hay que indicar el nombre del VG a ampliar y el (o los) PV(s) nuevo(s) a añadir.

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

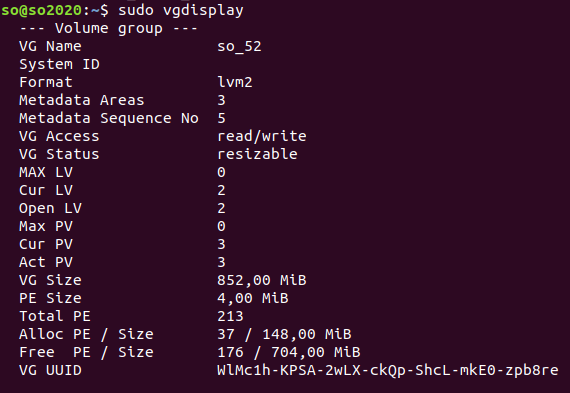
*pvcreate /dev/sda7*



*vgextend so\\_52 /dev/sda7*

**

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

**

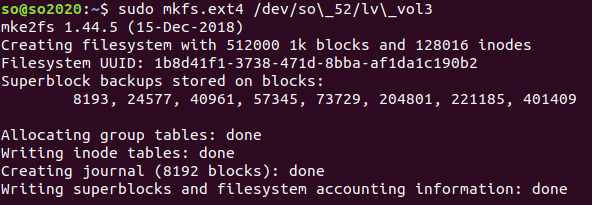
*21. Generar el nuevo LV de 500M (llamarlo lv\_vol3)*

*lvcreate -L 500M -n lv\\_vol3 so\\_52*

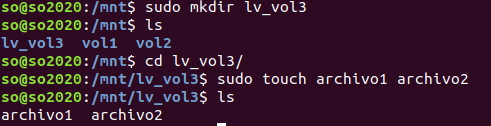
**

*22. Montar este nuevo LV en el directorio /mnt/lv\_vol3 y crear un directorio con dos archivos adentro (los nombre pueden ser cualquiera)*



**



**

**

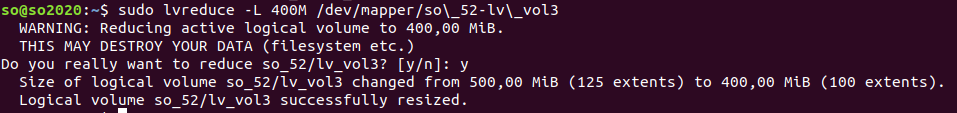
*En el siguiente paso se intentará reducir el tamaño del LV generado en el punto 21) de 500M a 400M*

*23. Desmontar el LV.*

sudo umount /dev/so\\_52/lv\_vol3

*24. Reducir el LV lv\_vol3 (decir Sí/Yes al aviso que aparece al ejecutar el siguiente comando)*

*lvreduce -L 400M /dev/mapper/so\\_52-lv\\_vol3*

**

*25. Montar el LV reducido en el paso anterior. ¿Es posible ver el directorio y los archivos generados en el paso anterior?*

sudo mount /dev/so\_52/lv\_vol3 /mnt/lv\_vol3



No es posible ver los archivos



*26. ¿Cuál es el error en el procedimiento anterior?*

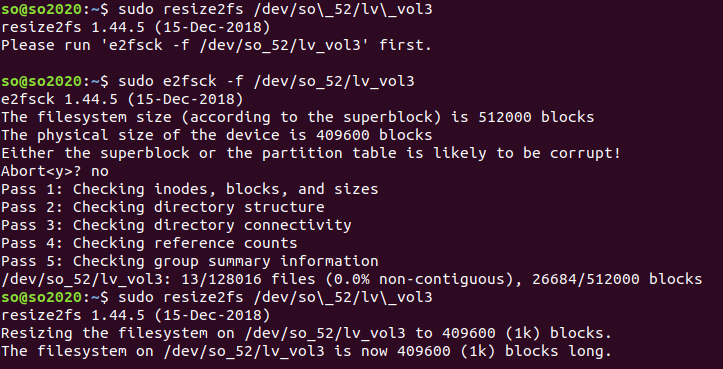
El error es que faltó redimensionar el file system antes de modificar el tamaño del LV.

*27. Investigar y ejecutar los pasos necesarios, y de forma ordenada, para poder achicar el LV a 400M.*

Desmontar sudo umount /dev/so\_52/lv\_vol3 se debe desmontar primero porque no se puede modificar el filesystem de manera online.

Correr el siguiente comando: sudo e2fsck -f /dev/so\_52/lv\_vol3

Redimensionar el file system: sudo resize2fs /dev/so\\_52/lv\\_vol3

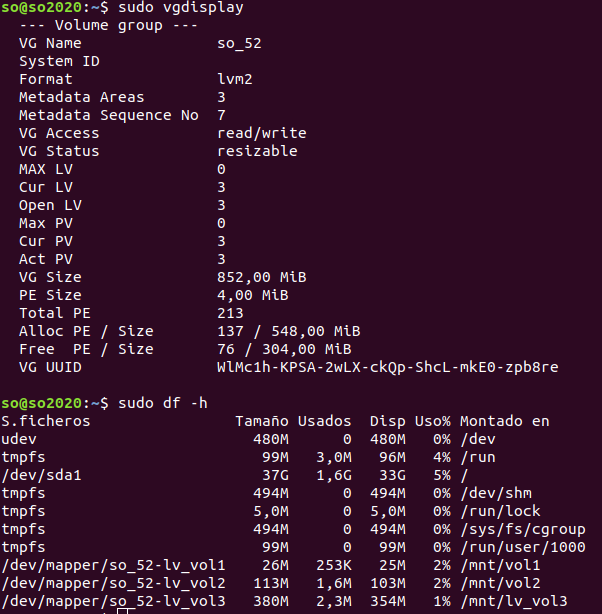


Modificar el LV: lvreduce -L 400M /dev/mapper/so\\_52-lv\\_vol3



Montar: sudo mount /dev/so\_52/lv\_vol3 /mnt/lv\_vol3

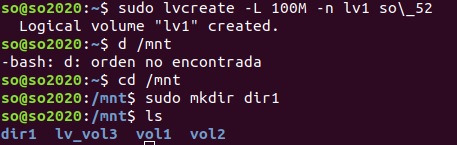
Para checkear si fue modificado el tamaño: sudo df -h o sudo vgdisplay

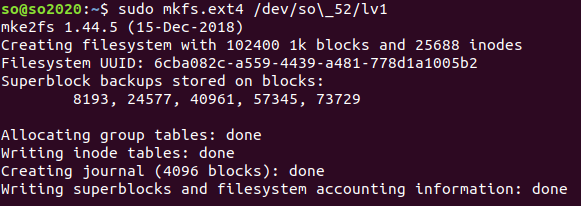
**

*A continuación se mostrará el funcionamiento de los snapshot en LVM*

*28. Generar un LV de 100M, nombrarlo lv1, (o usar uno de los generados en pasos anteriores).*

lvcreate -L 100M -n lv1 so\\_52



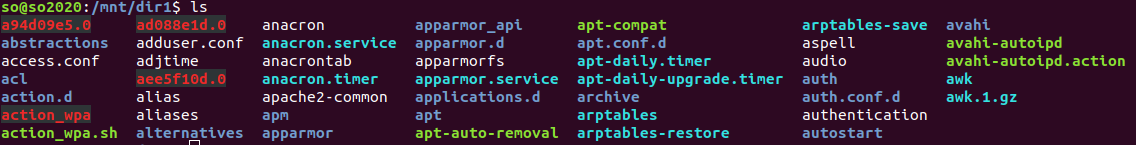


*Montarlo en el directorio /dir1*

sudo mount /dev/so\_52/lv1 /mnt/dir1

*29. Copiar desde /etc todo los archivos y directorios que comiencen con la letra a.*

sudo find /etc/ -name 'a\*' -exec cp -r "{}" /mnt/dir1/ \;

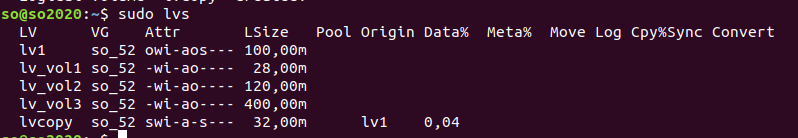


*30. Mediante el siguiente comando generar un snapshot del LV anterior*

*lvcreate -L 30M -s /dev/so\\_52/lv1 -n lvcopy (-s indica que este LV será un snapshot)*

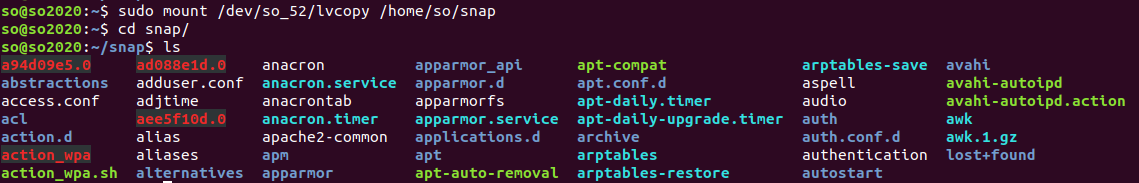
**

*31. Verificar la creación del snapshot con el comando lvs.*

**

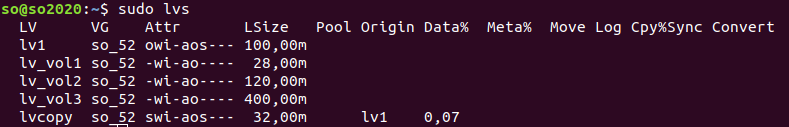
*Montarlo en el directorio /snap.*

*¿Qué contenido tiene en el snapshot?*

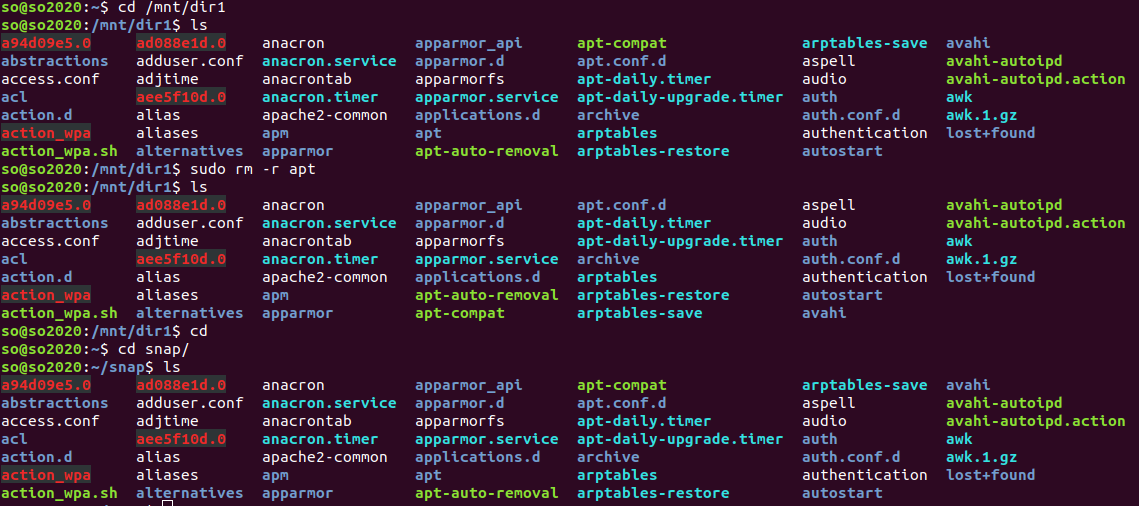


*32. ¿Cuánto espacio hay consumido en el snapshot creado? ¿Por qué sucede esto?*

El snapshot está consumiendo 0,07%, ya que sería lo que está alocado.



*33. 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?*

**

El snapshot sigue teniendo apt.

Al eliminar notamos que ese % crece. Esto se debe a que al momento de crear el snapshot solo se copian los metadatos en el snapshot. Una vez que se empiezan a modificar los bloques en el LV original es cuando se empiezan a mover los datos al snapshot, entonces se incrementa la cantidad de datos usados.

*34. 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.*

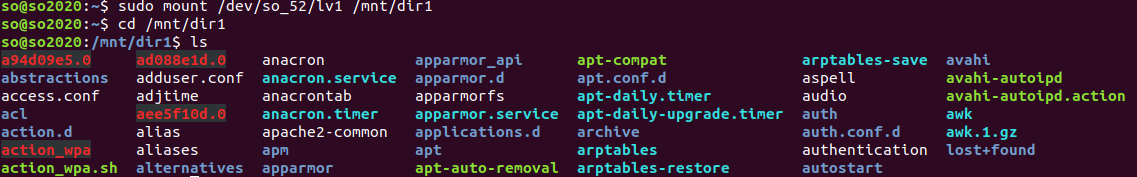
*Luego, realizar el “merge” de ambos LVs:*

*lvconvert --merge /dev/so\\_52/lvcopy*

**

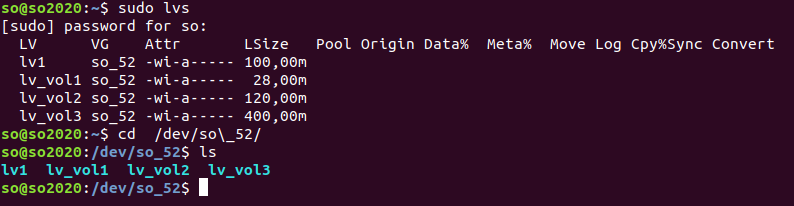
*35. Comprobar si el LV original contiene nuevamente los datos eliminados anteriormente (Deberá montarlo nuevamente.)*

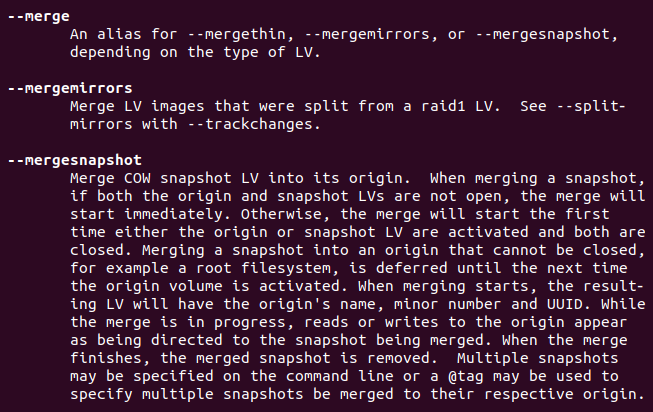
Contiene nuevamente la carpeta apt

**

*36. ¿Qué sucedió con el snapshot?*

Cuando el merge finalizá, se elimina el snapshot, esta información está detallada en la documentación de lvconvert.

**

**

*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\\_X/lv1*

*lvchange -ay /dev/so\\_X/lv1*