



SISTEMAS OPERATIVOS

Práctica 6

Requisitos

Para poder realizar la siguiente práctica es necesario tener instalado un sistema operativo Linux con un espacio libre de 4GB como mínimo. Además, se deberán instalar los paquetes necesarios para poder generar el RAID y LVM.

Obs. 1: al final de la práctica se encuentran algunos links de referencia para las siguientes preguntas

Obs. 2: en el pdf de la explicación de la práctica, al final de la misma, se encuentran algunas imágenes de como realizar un particionado manual. No es necesario seguirlo. Lo importante es que al momento de la instalación quede en el disco un espacio libre de 6GB para poder crear las nuevas particiones que se usarán para realizar la práctica.

File Systems

1. ¿Qué es un file system?
2. Describa las principales diferencias y similitudes entre los file systems: FAT, NTFS, Ext(2,3,4), XFS y HFS+
3. ¿Qué es el particionado? ¿Qué es el UUID? ¿Para qué se lo utiliza? (Hint: ver el comando blkid)
4. ¿Es necesario tener un file system para acceder a una partición?
5. ¿Qué es el área de swap en Linux? ¿Existe un área similar en Windows?
6. ¿Qué función cumple el directorio lost+found en Linux?
7. En Linux, ¿dónde se almacena el nombre y los metadatos de los archivos?
8. Seleccione uno de sus file systems (una partición) y conteste usando el comando dumpe2fs:
 - ¿Qué información describe el comando dumpe2fs?
 - ¿Cuál es el tamaño de bloque del file system?
 - ¿Cuántos inodos en total contiene el file system? ¿Cuántos archivos como máximo se pueden crear con el estado actual del file system?

- ¿Cuántos grupos de bloques existen?
 - ¿Cómo haría para incrementar la cantidad de inodos de un file system?
9. ¿Qué es el file system procfs? ¿Y el sysfs?
10. Consultando el sysfs, ¿cuál es el tamaño del sector lógico de su disco? ¿Y el físico?
11. Usando el directorio /proc, contestar:
- ¿Cuál es la versión de SO que tiene instalado?
 - ¿Cuál es procesador de su máquina?
 - ¿Cuánta memoria RAM disponible tiene?
 - ¿Qué archivo debería consultar si se quiere ver el mismo resultado que el comando lsmem?
12. Usando el comando stat, contestar:
- ¿Cuándo fue la última vez que se modificó el archivo /etc/group?
 - ¿Cuál es la diferencia entre los datos Cambio (Change) y Modificación (Modify)?
 - ¿Cuál es el inodo que ocupa? ¿Cuántos bloques ocupa?
 - ¿Qué número de inodo ocupa el directorio raíz?
 - ¿Es posible conocer la fecha de creación de un file en Linux? ¿Por qué?
13. Los permisos por defecto de Linux al crear un archivo o directorio son 666 y 777 respectivamente. Cree un nuevo archivo y analice sus permisos. ¿Es así? ¿Por qué sucede esto?
14. ¿Qué es un link simbólico? ¿En qué se diferencia de un hard-link?
15. 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?
16. ¿Para qué sirven los permisos especiales en Linux? Analizar el Sticky-bit, SUID y SGID
17. ¿Cuáles son los permisos del archivo /etc/shadow? ¿Por qué puedo modificar mi password sino soy usuario root?
18. Crear un archivo en el directorio /tmp. Si abre otra consola y se loguea con un usuario distinto, ¿puede borrar ese archivo? ¿Por qué? (Hint.: ver permisos especiales en Linux)
19. ¿Qué es Advanced Format en los disco rígidos? ¿Qué es 512e?
20. ¿Por qué es recomendable alinear las particiones?
21. ¿Qué es el sistemas de archivo F2FS?

Usando el espacio libre dejado en el disco de la MV, realizar los siguientes pasos y contestar:

1. Usando la herramienta “fdisk” o “parted” de Linux generar dos particiones de 1GB cada una.
2. Formatear la primera partición con ext3 y la segunda con ext4.
3. Montar las particiones en dos directorios “disc1” y “disc2”

4. Desarrollar un único script que genere en cada partición 1000 archivos y escriba en cada uno la palabra “Hola”. El último comando del script debe ser “df -h”
5. ¿Cuál es el espacio consumido por los archivos en cada partición? ¿Son iguales?
6. Esperar un tiempo (1 min.) y ejecutar el comando “df -h” nuevamente. ¿Cómo es la salida ahora? ¿Por qué?
7. Monte nuevamente la partición correspondiente con la opción adecuada para que no suceda lo del punto anterior
8. Generar un archivo en el directorio “disc1”. Observar los datos del inodo del archivo generado
9. Leer el contenido del archivo. ¿Se modificaron algunos de los datos del inodo del archivo?
10. ¿Cómo puede modificar el comportamiento anterior con el fin de evitar esas modificaciones? ¿Es posible hacerlo sin desmontar la partición?

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 descargarla con las herramientas apt/yum.

1. ¿Qué es un RAID? Explique las diferencias entre los distintos niveles de RAID
2. Utilizar el comando parted -l (debe ejecutarlo con sudo) para ver la tabla de particiones. Conteste:
 - a) ¿Cómo llama Linux al dispositivo físico? ¿Cuál es su tamaño total?
 - b) ¿Cuántas particiones existen? ¿Qué tipo de file-system contiene cada una?
 - c) ¿Qué significa la bandera arranque?
3. Usando el comando parted crear una nueva partición de tipo extendida (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 ext3.

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?
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:

```
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?
8. Ejecutar la siguiente consulta y contestar:

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

- a) ¿Cuál es el tamaño total del RAID?
- b) ¿Cuál es el tamaño total utilizable para almacenar datos?

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)

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

- a) Crear un file system de tipo ext3 en el RAID 5 recién generado

```
mkfs.ext3 /dev/md0
```

- b) Montar la partición con el file system generado en el directorio `/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? ¿Cuántos dispositivos activos existen?

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

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

```
mdadm -r /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?

- 2) ¿Qué hubiese sucedido si teníamos otra partición como “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

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`

b) Ejecutar el comando `mdadm --detail /dev/md0`

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

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

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

14. Agregar la nueva partición al RAID:

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

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

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?

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 `-r`). 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`

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

LVM (Logical Volumen 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?
2. ¿Cómo funcionan los “snapshots” en LVM?
3. Instalar la herramienta lvm2 con apt/yum
4. Indicar que las particiones 5 y 6 funcionarán como volúmenes físicos

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

5. Mediante el comando `pvdisk` observar el estado del volumen físico recientemente creado
6. Crear un grupo de volúmenes (volume group, VG) llamado “so_vg”

```
vgcreate so\_vg /dev/sda5 /dev/sd6
```

7. Utilizar el comando `vgdisplay` para ver el estado del VG
 - ¿Cuál es tamaño total del VG?
 - ¿Qué significa PE?
8. Crear dos volúmenes lógicos (logical volume, LV) de 8MB y 120MB respectivamente

```
lvcreate -l 2 -n lv\_vol1 so\_vg  
lvcreate -L 120M -n lv\_vol2 so\_vg
```

9. ¿Cuál es la diferencia entre los dos comandos utilizados en el punto anterior?
10. Formatear los dos LV generados en el paso anterior con un file system de tipo ext3:

```
mkfs.ext3 /dev/so_vg/lv_vol1
```

11. Crear dos directorios, vol1 y vol2, dentro de /mnt y montar ambos LG en estos directorios (montar el LV lv_vol1 en el directorio vol1 y lv_vol2 en el directorio vol2)
12. Ejecutar el comando *proof* (Puede tomar un rato su ejecución. Este comando estará disponible en la plataforma y deberán copiarlo a la VM)
13. Crear un nuevo archivo en /mnt/vol1. ¿Es posible? ¿Por qué?
14. Es posible solucionar esto incrementando el tamaño del file system. Para esto primero se debe incrementar el tamaño del LV correspondiente:

- a) Desmontar el LV lv_vol1
- b) Extender el LV lv_vol1 en 20M

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

- c) Incrementar el tamaño del file system. Antes de incrementar el tamaño es necesario ejecutar el comando e2fsck para comprobar el file system esté correcto:

```
e2fsck -f /dev/so_vg/lv_vol1  
resize2fs /dev/so_vg/lv_vol1
```

15. Montar nuevamente el LV en /mnt/vol1
16. ¿Siguen estando los datos disponibles?
17. Intentar crear un nuevo archivo en /mnt/vol1. ¿Es posible? ¿Por qué?
18. Se desea crear un nuevo LV de 500M. ¿Hay suficiente espacio?
19. Para solucionar el punto anterior se debe extender el tamaño del VG, por lo tanto, se agregará la partición sda7 al VG:

```
pvccreate /dev/sda7  
vgextend so_vg /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)
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 18) de 500M a 400M
23. Desmontar el LV.
24. Reducir el LV lv_vol3 (decir Sí/Yes al aviso que aparece al ejecutar el siguiente comando)

```
lvreduce -L 400M /dev/mapper/so_vg-lv_vol3
```

25. Montar el LV reducido en el paso anterior. ¿Es posible ver el directorio y los archivos generados en el paso anterior?
26. ¿Cuál es el error en el procedimiento anterior? ¿Cuáles serían los pasos correctos?
27. Ejecute los pasos necesarios y de forma ordenada para poder achicar el LV a 400M (investigar)
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). Montarlo en el directorio /dir1
29. Copiar desde /etc todo los archivos y directorios que comiencen con las letra a,b,c,d,y e.
30. Mediante el siguiente comando generar un snapshot del LV anterior

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

31. Verificar la creación del snapshot con el comando lvs. Montarlo en el directorio /snap
32. ¿Cuánto espacio hay consumido en el snapshot creado? ¿Por qué sucede esto?
33. Para probar el snapshot, elimine una carpeta del LV original (por ej, la carpeta cups). ¿Se eliminó desde el LV original? ¿Y del snapshot?
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 debe desmontar el LV original (lv1). Luego, realizar el “merge” de ambos LVs:

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

35. Comprobar si el LV original contiene nuevamente los datos eliminados anteriormente (el snapshot es eliminado automáticamente)
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\_vg/lv1  
lvchange -ay /dev/so\_vg/lv1
```

BTRFS & ZFS

1. Tanto para BTRFS como para ZFS, responder:
 - ¿Cuál es el significado de las siglas?
 - ¿Quién los creó? ¿Cuál es su modo de licenciamiento?
 - ¿Cuáles son las características más importante de cada uno?
 - Investigar qué es la técnica *copy-on-write*

2. Generar en la MV dos particiones de un 3GB cada una. Crear 3 nuevo directorio llamado /disco5, /volumen1 y /volumen2.
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.
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`)
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`.
6. Generar un archivo de 2000MB en el directorio /disco5. Analizar nuevamente el espacio alocado/ocupado. Usar el comando `dd` para generar el archivo. (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?
7. Asignar la otra partición a /disco5. ¿Se modificaron los valores con respecto al punto anterior?
8. Generar un archivo de 3000MB y almacenarlo en el directorio /disco5. ¿Aumenta el espacio alocado? ¿Cuánto espacio se ha ocupado realmente?
9. Usando las dos particiones anteriores crear un RAID1 y montarlo /disco5. ¿Qué partición puede elegir para montar el file system?
10. ¿Es posible generar los dos archivos anteriores en ese filesystem? ¿Por qué?
11. Eliminar todo el contenido de /disco5 y generar dos subvolúmenes, llamados vol1 y vol2. ¿Puede ver los subvolúmenes creados? ¿Qué ID tiene cada uno? ¿Qué significa el ID 5?
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 subvolúmenes?
13. Generar un archivo de 300MB en el directorio vol1, ¿es posible ver el archivo en /volumen1? Si ejecuta el comando `df -h`, ¿qué espacio se ha consumido? ¿Por qué sucede esto?
14. Limitar el tamaño del subvolumen volumen2 a 300MB. Intentar copiar un archivo de 400MB y ver qué sucede. Elevar el tamaño de la cuota a 450MB, ¿es posible ahora?
15. 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?
16. Modificar el contenido el archivo original agregándole **para sistemas operativos**. ¿Se modifica la copia en el snapshot?
17. Si se desea volver al subvolumen original, ¿cómo lo haría? (sin hacer un copy de los archivos)

- <http://www.nongnu.org/ext2-doc/ext2.html>

- <http://e2fsprogs.sourceforge.net/ext2intro.html>
- <http://www.cyberciti.biz/tips/understanding-unixlinux-filesystem-superblock.html>
- <http://www.tldp.org/LDP/tlk/fs/filesystem.html>
- <http://www.geekride.com/hard-link-vs-soft-link/>