

LABORATORIO 2

df -h --> muestra información de un sistemas de fitxeros

lsblk -e7 --> muestra información de la partición

1. CONFIGURACIÓN DEL ENTORNO

2. GESTIÓN BÁSICA

1) En el disco recién creado, crear 4 particiones de 1GB cada una y formatearlas: una de ellas será ext3, otra btrfs, otra xfs y última ext4.

```
nagore@instance-1:~$ sudo cfdisk /dev/sdc
```

```
nagore@instance-1:~$ sudo mkfs.ext3 /dev/sdc1
```

```
nagore@instance-1:~$ sudo mkfs.btrfs /dev/sdc2
```

```
nagore@instance-1:~$ sudo mkfs.xfs /dev/sdc3
```

```
nagore@instance-1:~$ sudo mkfs.ext4 /dev/sdc4
```

2) Configurar un montaje automático de las particiones (al arranque de la máquina virtual) en /disco[X] donde X es 1, 2, 3, 4. Se recomienda seguir estos pasos. Verificar que los montajes se mantienen al reiniciar la máquina. En caso de que la MV no arranque o no permita acceso vía SSH por problemas de configuración, se puede iniciar una terminal mínima llamada “consola serie” que provee acceso root para resolver incidencias.

https://cloud.google.com/compute/docs/disks/format-mount-disk-linux?hl=es-419#configure_automatic_mounting_on_vm_restart

```
nagore@instance-1:~$ sudo cp /etc/fstab /etc/fstab.backup
```

```
nagore@instance-1:~$ sudo blkid /dev/sdc1
```

```
nagore@instance-1:~$ sudo blkid /dev/sdc2
```

```
nagore@instance-1:~$ sudo blkid /dev/sdc3
```

```
nagore@instance-1:~$ sudo blkid /dev/sdc4
```

```
nagore@instance-1:~$ sudo nano /etc/fstab
```

```
LABEL=cloudimg-rootfs      /          ext4  discard,errors=remount-ro  0 1
```

```
    LABEL=UEFI  /boot/efi  vfat  umask=0077  0 1
```

```
    UUID=064c9f21-40f5-4205-9319-cb148eaf867b /mnt/disks/disk1 ext3
discard,defaults,nofail 0 2
```

```
    UUID=e8ca9151-b90e-4d39-876d-02f97d0f1dab /mnt/disks/disk2 btrfs
discard,defaults,nofail 0 2
```

```
UUID=f54b86a3-4b80-4acc-bafe-cd0af07bc3e4 /mnt/disks/disk3 xfs  
iscard,defaults,nofail 0 2
```

```
UUID=a33c2589-baac-4fe8-bbaf-453600fa59f6 /mnt/disks/disk4 ext4  
iscard,defaults,nofail 0 2
```

3) ¿Cuál de los sistemas de ficheros creados ocupa más espacio?

```
nagore@instance-1:~$ df -h
```

4) ¿Es posible acceder a una partición ext3 que ha sido montada como ext4? ¿Y al revés? ¿Por qué?

Sí, es posible acceder a una partición que originalmente fue formateada como ext3 pero que posteriormente se ha montado como ext4, y viceversa. Las particiones ext3 y ext4 son sistemas de archivos compatibles entre sí, lo que significa que puedes montar una partición ext3 como ext4 y viceversa sin problemas.

5) Desmontar y borrar las 3 últimas particiones. Crear una única partición ext4 de 8 GB.

```
nagore@instance-1:~$ sudo umount /mnt/disks/disk2
```

```
nagore@instance-1:~$ sudo umount /mnt/disks/disk3
```

```
nagore@instance-1:~$ sudo umount /mnt/disks/disk4
```

```
nagore@instance-1:~$ sudo cfdisk /dev/sdb
```

```
nagore@instance-1:~$ lsblk -e7
```

6) Copiar el contenido del directorio /var en la nueva partición ext4 que acabas de crear. Después, redimensionar la partición para que sea lo más pequeña posible.

```
nagore@instance-1:~$ sudo mkfs.ext4 /dev/sdb2
```

```
nagore@instance-1:~$ sudo mount -t ext4 /dev/sdb2 /mnt/disks/disk2
```

```
nagore@instance-1:~$ sudo cp -r /var/* /mnt/disks/disk2
```

```
nagore@instance-1:~$ sudo umount /mnt/disks/disk2
```

#reducir el sistema de archivos

```
nagore@instance-1:~$ sudo e2fsck -f /dev/sdb2
```

```
nagore@instance-1:~$ sudo resize2fs -M /dev/sdb2 #se reduce a 757184
```

#reducir la particion a 757184

```
nagore@instance-1:~$ sudo cfdisk /dev/sdb
```

#volver a montar

```
nagore@instance-1:~$ sudo mkfs.ext4 /dev/sdb2
```

```
nagore@instance-1:~$ sudo mount -t ext4 /dev/sdb2 /mnt/disks/disk2
```

#comprobar los archivos siguen igual

```
nagore@instance-1:~$ df -h
```

7) Eliminar la configuración de montaje automático realizada en el 2º paso.

```
nagore@instance-1:~$ sudo cp /etc/fstab.backup /etc/fstab
```

```
nagore@instance-1:~$ nano /etc/fstab
```

3. COMPARATIVA DE RENDIMIENTO

1) Borrar las particiones creadas en el disco en la sección anterior. En este ejercicio nos referiremos a este disco como balanceado.

```
nagore@instance-1:~$ sudo cfdisk /dev/sdb
```

2) Añadir un nuevo disco de 10 GB de tipo “SSD” a la máquina virtual. En este ejercicio nos referiremos a este disco como SSD.

3) En ambos discos, crear 1 partición de 4 GB y formatearla como Ext4. Montar la partición del disco balanceado en el directorio /discoBalanceado y la partición del disco SSD en el directorio /discoSSD. No es necesario configurar los montajes para que se realicen en el arranque del sistema.

```
nagore@instance-1:~$ sudo mkfs.ext4 /dev/sdb1
```

```
nagore@instance-1:~$ sudo mkfs.ext4 /dev/sdc1
```

```
nagore@instance-1:~$ mkdir discoBalanceado
```

```
nagore@instance-1:~$ sudo mount -t ext4 /dev/sdb1 /home/nagore/discoBalanceado
```

```
nagore@instance-1:~$ mkdir discoSSD
```

```
nagore@instance-1:~$ sudo mount -t ext4 /dev/sdc1 /home/nagore/discoSSD
```

4) Revisar la 1ª sección de este artículo para aprender a realizar una prueba de rendimiento con fio.

https://docs.gitlab.com/ee/administration/operations/filesystem_benchmarking.html

5) Instalar fio en el sistema siguiendo el comando que se indica en el artículo.

```
nagore@instance-1:~$ sudo apt install fio
```

6) Utilizar fio para realizar la siguiente prueba:

- a. Creación de un fichero de contenido aleatorio de 1 GB.
- b. Mezcla de 75% y 25 % entre operaciones de lectura y escritura.
- c. Resto de parámetro por defecto (cómo indicados en el ejemplo del artículo).

Esta prueba se debe realizar con ambos discos. El nombre del fichero que se utilice es indiferente, pero debe estar dentro de cada carpeta /disco correspondiente en cada prueba. Los valores más representativos de las pruebas son las métricas de velocidad de lectura (READ) y escritura (WRITE), que se encuentran entre las últimas líneas que genera fio. Estos valores se indican en kB/s o MB/s, ¿qué diferencia hay entre ambos discos?

```
nagore@instance-1:~$ sudo fio --randrepeat=1 --ioengine=libaio --direct=1 --gtod_reduce=1 --name=test_balanceado --bs=4k --iodepth=64 --readwrite=randrw --rwmixread=75 --size=1G --filename=/discoBalanceado/testfile
```

```
nagore@instance-1:~$ sudo fio --randrepeat=1 --ioengine=libaio --direct=1 --gtod_reduce=1 --name=test_balanceado --bs=4k --iodepth=64 --readwrite=randrw --rwmixread=75 --size=1G --filename=/discoSSD/testfile
```

7) Repetir la prueba anterior, esta vez utilizando una mezcla de 50% de operaciones de lectura y 50% de operaciones de escritura. ¿Varían los resultados respecto a la 1ª prueba?

```
nagore@instance-1:~$ sudo fio --randrepeat=1 --ioengine=libaio --direct=1 --gtod_reduce=1 --name=test_balanceado --bs=4k --iodepth=64 --readwrite=randrw --rwmixread=50 --size=1G --filename=/discoBalanceado/testfile
```

```
nagore@instance-1:~$ sudo fio --randrepeat=1 --ioengine=libaio --direct=1 --gtod_reduce=1 --name=test_balanceado --bs=4k --iodepth=64 --readwrite=randrw --rwmixread=50 --size=1G --filename=/discoSSD/testfile
```

8) Revisar el precio de los discos de tipo balanceado y SSD en Google Cloud. ¿La diferencia de precio es proporcional con la diferencia de rendimiento?

https://cloud.google.com/compute/disks-image-pricing?hl=es-419&_ga=2.41740450.-1424953120.1666343954

El disco Balanceado tiene un precio de 0.110\$ per GB, y el SSD, 0.187\$ per GB

9) Desmontar y eliminar el disco de tipo SSD.

```
nagore@instance-1:~$ sudo umount /home/nagore/discoSSD
```

```
nagore@instance-1:~$ sudo cfdisk /dev/sdc
```

4. GESTIÓN AVANZADA

1) Borrar las particiones creadas en las secciones anteriores en el disco.

```
nagore@instance-1:~$ sudo umount /home/nagore/discoBalanceado
```

```
nagore@instance-1:~$ sudo cfdisk /dev/sdb
```

2) Añadir un nuevo disco de 10 GB de tipo balanceado a la máquina virtual.

3) Crear 2 particiones de 3 GB en cada disco. Crear un volumen lógico LVM con 3 de las 4 particiones. Crear una partición ext4 en el volumen usando el 100% de espacio.

```
nagore@instance-1:~$ sudo cfdisk /dev/sdb
```

```
nagore@instance-1:~$ sudo cfdisk /dev/sdc
```

```
nagore@instance-1:~$ sudo pvcreate -ff /dev/sdb1 /dev/sdb2 /dev/sdc1
```

Comprobamos que se hayan creado

```
nagore@instance-1:~$ sudo pvdisplay
```

```
nagore@instance-1:~$ sudo vgcreate grupovol /dev/sdb1 /dev/sdb2 /dev/sdc1
```

```
nagore@instance-1:~$ sudo pvdisplay
```

```
nagore@instance-1:~$ sudo lvcreate -l 100%FREE -n mivolumen grupovol
```

```
nagore@instance-1:~$ sudo pvdisplay
```

```
nagore@instance-1:~$ sudo mkfs.ext4 /dev/grupovol/mivolumen
```

4) Montar el sistema de ficheros en un directorio y comprobar su estado. Para ello, copiar alguno de los directorios del sistema operativo a él.

```
nagore@instance-1:~$ sudo mkdir /diskoVol
```

```
nagore@instance-1:~$ sudo mount /dev/grupovol/mivolumen /diskoVol
```

```
nagore@instance-1:~$ lsblk -e7
```

```
nagore@instance-1:~$ sudo cp -r /var/* /diskoVol
```

```
nagore@instance-1:~$ ls /diskoVol
```

5) Añadir la 4ª partición al volumen lógico y extender el tamaño del sistema de ficheros para que ocupe el total del volumen. Comprobar que los datos copiados en el paso anterior siguen estando.

```
nagore@instance-1:~$ sudo pvcreate -ff /dev/sdc2
nagore@instance-1:~$ sudo vgextend grupovol /dev/sdc2
nagore@instance-1:~$ sudo lvextend -l +100%FREE /dev/grupovol/mivolumen
nagore@instance-1:~$ df -h --> /dev/mapper/grupovol-mivolumen
nagore@instance-1:~$ sudo resize2fs -p /dev/mapper/grupovol-mivolumen
nagore@instance-1:~$ df -h #comprobar el tamaño del sistema de archivos
nagore@instance-1:~$ ls /diskoVol #comprobar que siguen estando los datos
```

6) Borrar el volumen lógico recién creado (utilizar los comandos LVM apropiados).

```
nagore@instance-1:~$ sudo umount /diskoVol
nagore@instance-1:~$ sudo rm -rf /diskoVol
nagore@instance-1:~$ sudo lvremove /dev/mapper/grupovol-mivolumen
nagore@instance-1:~$ sudo lvdisplay
```

7) Crear un sistema RAID 5 con 3 de las particiones. Crear un sistema de ficheros ext4 para el sistema RAID 5 y hacerlo accesible. Copiar el contenido de la carpeta /var a la carpeta del sistema RAID.

1. Crear el sistema RAID 5

```
nagore@instance-1:~$ sudo mdadm --create /dev/md0 --level=5 --raid-devices=3 /dev/sdb1
/dev/sdb2 /dev/sdc1
```

2. Crear un sistema de archivos ext4 en el sistema RAID 5

```
nagore@instance-1:~$ sudo mkfs.ext4 /dev/md0
```

3. Montar el sistema de archivos

```
nagore@instance-1:~$ sudo mkdir /mnt/raid
nagore@instance-1:~$ sudo mount /dev/md0 /mnt/raid
```

4. Copiar el contenido de /var al sistema RAID

```
nagore@instance-1:~$ sudo cp -r /var/* /mnt/raid
nagore@instance-1:~$ ls /mnt/raid
```

8) Simular un fallo en el tercer disco (parámetro -f). Recuperar la información perdida usando la partición que quedó libre.

0. Verificar el estado del sistema RAID

```
nagore@instance-1:~$ sudo mdadm --detail /dev/md0
```

1. Simular un fallo en el tercer disco

```
nagore@instance-1:~$ sudo mdadm --fail /dev/md0 /dev/sdc1
```

2. Eliminar el disco fallido

```
nagore@instance-1:~$ sudo mdadm --remove /dev/md0 /dev/sdc1
```

3. Verificar el estado del sistema RAID

```
nagore@instance-1:~$ sudo mdadm --detail /dev/md0
```

#o tambien podemos comprobarlo asi

```
nagore@instance-1:~$ cat /proc/mdstat
```

4. Restaurar la configuración RAID

```
nagore@instance-1:~$ sudo mdadm --add /dev/md0 /dev/sdc1
```

5. Verificar la recuperación

```
nagore@instance-1:~$ cat /proc/mdstat
```

9) Desmontar y eliminar el dispositivo RAID. Eliminar el disco creado en el paso 2.

1. Detener el dispositivo RAID

```
nagore@instance-1:~$ sudo mdadm --stop /dev/md0
```

2. Borrar la información de configuración RAID

```
nagore@instance-1:~$ sudo mdadm --zero-superblock /dev/sdb1 /dev/sdb2 /dev/sdc1
```

3. Eliminar el dispositivo RAID

```
nagore@instance-1:~$ sudo mdadm --remove /dev/md0
```

4. Borrar el punto de anclaje

```
nagore@instance-1:~$ sudo rm -rf /mnt/raid
```

5. COPIAS DE SEGURIDAD

1) Borrar las particiones creadas en la sección anterior en el disco.

```
nagore@instance-1:~$ sudo cfdisk /dev/sdb
```

2) Crear 1 partición de 4 GB en el disco y formatearla como ext4. Montarla en un directorio llamado /backups.

```
nagore@instance-1:~$ lsblk -e7
```

```
nagore@instance-1:~$ sudo mkfs.ext4 /dev/sdb1
```

```
nagore@instance-1:~$ sudo mkdir /backups
```

```
nagore@instance-1:~$ sudo mount /dev/sdb1 /backups
```

3) Instalar rsnapshot en el sistema y revisar este documento donde se detalla su configuración. <https://wiki.archlinux.org/title/Rsnapshot>

```
nagore@instance-1:~$ sudo apt install rsnapshot
```

4) Configurar rsnapshot de la siguiente forma:

A. Directorio para almacenar las copias de seguridad: /backups.

B. Niveles de copia e intervalos:

i. "horaria", 24

ii. "diaria", 7

iii. "semanal", 4

C. Directorios a guardar (todos se almacenan en el directorio /backups): /home, /etc y /var/log

```
nagore@instance-1:~$ sudo nano /etc/rsnapshot.conf
```

```
# Directorio para almacenar las copias de seguridad
```

```
snapshot_root /backups
```

```
# Niveles de copia e intervalos
```

```
retain hourly 24
```

```
retain daily 7
```

```
retain weekly 4
```

```
# Directorios a guardar (todos se almacenan en el directorio /backups)
```

```
backup /home/      localhost/
```

```
backup /etc/       localhost /
```

```
backup /var/log/   localhost /
```


5) Verificar que la configuración es correcta con el comando rsnapshot configtest.

```
nagore@instance-1:~$ sudo rsnapshot configtest
```

(dar permisos?)

6) Realizar una copia de tipo “horaria” y revisar que los contenidos se han copiado correctamente.

```
nagore@instance-1:~$ sudo rsnapshot hourly
```

```
nagore@instance-1:~$ ls /backups
```

7) Crear una carpeta y un fichero nuevo en el directorio /home de tu usuario (incluye algo de texto en el fichero). Después, realizar una nueva copia de tipo “horaria”.

```
nagore@instance-1:/home$ sudo mkdir miCarpeta
```

```
nagore@instance-1:/home$ cd miCarpeta/
```

```
nagore@instance-1:/home/miCarpeta$ sudo nano archivo.txt
```

```
nagore@instance-1:~$ sudo rsnapshot hourly
```

8) Verificar que la nueva copia se ha hecho correctamente y revisar los cambios entre ambas copias con el comando rsnapshot-diff.

```
nagore@instance-1:~$ sudo rsnapshot-diff -v /backups/hourly.0 /backups/hourly.1
```