



PRÁCTICA 3. RECURSOS DE ALMACENAMIENTO VIRTUAL

Virtualización y Procesamiento Distribuido

Este documento aborda la configuración y administración de diferentes métodos de almacenamiento para entornos virtualizados basados en KVM y Libvirt. Se exponen procedimientos para crear, asignar y gestionar volúmenes locales y remotos, incluyendo particiones físicas y contenedores NFS, con el fin de optimizar los recursos y garantizar la persistencia de los datos en máquinas virtuales.

Francisco Javier López-Dufour Morales

30-03-2025

Tabla de contenido

Introducción	2
Desarrollo.....	3
Fase 2. Contenedores locales	5
Tarea 1. Creación de un nuevo volumen en el contenedor por defecto	5
Fase 2. Contenedores locales	11
Tarea 2. Creación de una nueva partición en el host anfitrión	11
Tarea 3. Creación de un contenedor en una partición lógica	16
Fase 3. Contenedores en Red	24
Tarea 4. Creación de un contenedor NFS de imágenes ISO	24
Tarea 5. Creación de un contenedor NFS para volúmenes de máquinas virtuales.....	26
Pruebas y Validación	32
Verificación de volúmenes locales	32
Verificación de contenedores en red.....	33
Verificación de montajes automáticos	34
Conclusiones.....	36
Bibliografía	37

Introducción

En el ámbito de la virtualización, gestionar el almacenamiento de manera segura y eficiente constituye uno de los pilares fundamentales para el correcto funcionamiento de los entornos virtuales. La asignación de **volúmenes**, la configuración de **particiones** y la conexión a **recursos de red** (como los servicios NFS) exigen una planificación cuidadosa para no comprometer la integridad de los datos ni el rendimiento de las aplicaciones.

Este documento se centra en mostrar los pasos esenciales para crear y administrar diferentes clases de contenedores de almacenamiento. Desde la preparación de volúmenes locales hasta la integración de particiones reales del host, pasando por el montaje de recursos en red, cada sección describe de forma detallada las herramientas y procedimientos de configuración, evidenciando la importancia de la verificación y persistencia tras cada cambio realizado.

Desarrollo

Como paso previo a la realización de las tareas de esta práctica, es necesario crear una nueva máquina virtual llamada mvp3, que será el resultado de la clonación de la máquina virtual creada en la práctica 1 (mvp1). Esta clonación debe realizarse de manera que la interfaz de red de mvp3 posea una dirección MAC diferente a la que posee la interfaz de red de mvp1.

Para realizar la clonación se utiliza el comando `virt-clone`:

```
root@lq-d25:~# virt-clone --original mvp1 --name mvp3 --file
/var/lib/libvirt/images/mvp3.qcow2 --mac=00:16:3e:37:a0:03
Allocating 'mvp3.qcow2' | 1.8
GB 00:05 ...

El clon 'mvp3' ha sido creado exitosamente.
```

Explicación del comando:

- `virt-clone`: Herramienta que permite clonar máquinas virtuales existentes
- `--original mvp1`: Especifica la máquina virtual de origen
- `--name mvp3`: Define el nombre de la nueva máquina virtual
- `--file`: Especifica la ruta al archivo de imagen del nuevo disco
- `--mac`: Establece una dirección MAC diferente para la interfaz de red

Importancia de la MAC: Especificar una MAC diferente es crucial para evitar conflictos de direcciones en la red, ya que la máquina clonada operará en la misma red que la original. Libvirt y las herramientas de clonación suelen generar MACs únicas automáticamente si no se especifica una, pero hacerlo manualmente asegura el control.

Para verificar el estado de las particiones en el sistema anfitrión, se utiliza el comando `lsblk`:

```
root@lq-d25:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0    0 953,9G  0 disk
├─sda1       8:1    0   50M  0 part
├─sda2       8:2    0 477,2G  0 part
├─sda3       8:3    0    4G  0 part
├─sda4       8:4    0    1K  0 part
├─sda5       8:5    0  97,7G  0 part
├─sda6       8:6    0  31,1G  0 part [SWAP]
├─sda7       8:7    0 146,5G  0 part /
├─sda8       8:8    0 146,5G  0 part
├─sda9       8:9    0    1G  0 part
└─sda10      8:10   0    2G  0 part
sdb          8:16    1    0B  0 disk
zram0       252:0    0    8G  0 disk [SWAP]
```

Este comando nos muestra la estructura actual del disco, lo que será útil para las tareas posteriores cuando necesitemos crear nuevas particiones.

Fase 2. Contenedores locales

Tarea 1. Creación de un nuevo volumen en el contenedor por defecto

En esta tarea se crea un nuevo volumen virtual con las características especificadas (nombre Vol1_p3, tipo raw, tamaño 1GB) en el contenedor por defecto.

1. Creación del volumen:

```
root@lq-d25:~# virsh vol-create-as default Vol1_p3.img 1G --format raw
Se ha creado el volumen Vol1_p3
```

Explicación del comando:

- `virsh vol-create-as`: Crea un volumen en un pool de almacenamiento
- `default`: Especifica el pool de almacenamiento por defecto
- `Vol1_p3.img`: Nombre del volumen a crear
- `1G`: Tamaño del volumen (1 Gigabyte)
- `--format raw`: Especifica el formato raw (disco sin formato)

Formatos de Volumen: El formato **raw** es una imagen de disco simple, bit a bit, sin metadatos adicionales. Ofrece buen rendimiento, pero ocupa todo el espacio asignado desde el principio. El formato **qcow2** (QEMU Copy On Write 2) es más avanzado, permite características como snapshots, thin provisioning (el espacio se ocupa sólo cuando se escribe) y compresión, a costa de una ligera sobrecarga de rendimiento.

2. Verificación de los dispositivos actualmente asociados a la máquina virtual:

```
root@lq-d25:~# virsh domblklist mvp3 --details
Tipo    Dispositivo  Destino  Fuente
-----
-
file    disk        vda      /var/lib/libvirt/images/mvp3.qcow2
file    cdrom        sda      -
```

3. Desmontamos el cdrom que ya no es necesario:

```
root@lq-d25:~# virsh detach-disk mvp3 sda --config
El disco ha sido desmontado exitosamente
```

4. Asociación del volumen a la máquina virtual como dispositivo SATA:

```
root@lq-d25:~# virsh attach-disk mvp3 /var/lib/libvirt/images/Vol1_p3.img
sda --config --type disk --driver qemu --subdriver raw
```

El disco ha sido asociado exitosamente

Explicación del comando:

- `virsh attach-disk`: Asocia un disco a una máquina virtual
- `mvp3`: Nombre de la máquina virtual
- `/var/lib/libvirt/images/Vol1_p3.img`: Ruta al archivo de volumen
- `sda`: Nombre del dispositivo en la máquina virtual
- `--config`: Hace que la configuración sea **persistente** tras reiniciar
- `--type disk`: Especifica que se trata de un disco
- `--driver qemu`: Usar el controlador QEMU para acceder al disco
- `--subdriver raw`: Especifica el formato raw del disco

Controladores y Subcontroladores: El parámetro `--driver qemu` indica que QEMU se encargará de manejar el acceso al disco. El `--subdriver raw` le dice a QEMU que el formato subyacente del archivo es raw. Esto es importante para que QEMU interprete correctamente la imagen del disco. Para volúmenes `qcow2`, se usaría `--subdriver qcow2`.

5. Verificación de la asociación del volumen:

```
root@lq-d25:~# virsh domblklist mvp3 --details
Tipo   Dispositivo  Destino  Fuente
-----
-
file   disk        vda      /var/lib/libvirt/images/mvp3.qcow2
file   disk        sda      /var/lib/libvirt/images/Vol1_p3.img
```

6. Iniciar sesión en la máquina virtual mvp3:

```
root@lq-d25:~# ssh root@192.168.122.242
The authenticity of host '192.168.122.242 (192.168.122.242)' can't
be established.
ED25519 key fingerprint is
SHA256:gRFGvZlUIel5P1EJEdiEEgvXQ48k7iMy90z5SDPY2h4.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:1: 192.168.122.123
  ~/.ssh/known_hosts:4: mvp1.vpd.com
  ~/.ssh/known_hosts:8: 192.168.122.124
  ~/.ssh/known_hosts:9: 192.168.122.113
  ~/.ssh/known_hosts:10: 192.168.122.37
Are you sure you want to continue connecting
(yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.122.242' (ED25519) to the list
of known hosts.
Web console: https://mvp1.vpd.com:9090/ or
https://192.168.122.242:9090/

Last login: Fri Feb 28 20:06:07 2025
```

7. Verificar que el disco está disponible en la máquina virtual:

```
root@mvp1:~# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda                  8:0    0    1G  0 disk
sr0                  11:0    1 1024M  0 rom
zram0                251:0    0  1,9G  0 disk [SWAP]
vda                  252:0    0   10G  0 disk
├─vda1                252:1    0    1M  0 part
├─vda2                252:2    0    1G  0 part /boot
└─vda3                252:3    0    9G  0 part
   └─fedora-root      253:0    0    9G  0 lvm  /
```

8. Crear una partición de 512MB en el disco con fdisk:

```
root@mvp1:~# fdisk /dev/sda

Welcome to fdisk (util-linux 2.40.4).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS (MBR) disklabel with disk identifier 0xc4abc78f.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-2097151, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-2097151,
default 2097151): +512M

Created a new partition 1 of type 'Linux' and of size 512 MiB.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

Explicación del proceso de particionado:

- n: Crea una nueva partición
- p: Selecciona partición primaria
- 1: Asigna el número 1 a la partición
- Primera sector: valor por defecto (2048)
- Último sector: +512M para crear una partición de 512 MB
- w: Escribe los cambios en la tabla de particiones

9. Crear un sistema de archivos XFS en la nueva partición:

```
root@mvp1:~# mkfs.xfs -f /dev/sda1
meta-data=/dev/sda1            isize=512    agcount=4,
agsize=32768 blks
       =                       sectsz=512   attr=2, projid32bit=1
       =                       crc=1        finobt=1, sparse=1,
rmapbt=1
       =                       reflink=1    bigtime=1
inobtcount=1 nnext64=1
data     =                       bsize=4096   blocks=131072,
imaxpct=25
       =                       sunit=0      swidth=0 blks
naming   =version 2             bsize=4096   ascii-ci=0, ftype=1
log       =internal log         bsize=4096   blocks=16384,
version=2
       =                       sectsz=512   sunit=0 blks, lazy-
count=1
realtime =none                 extsz=4096   blocks=0, rtextents=0
Discarding blocks...Done.
```

Explicación del comando:

- **mkfs.xfs**: Crea un sistema de archivos XFS
- **-f**: Fuerza la creación incluso si ya existe un sistema de archivos
- **/dev/sda1**: Partición donde se creará el sistema de archivos

10. Reiniciar la máquina virtual para verificar la persistencia del disco:

```
root@mvp1:~# reboot
root@mvp1:~# Connection to 192.168.122.242 closed by remote host.
Connection to 192.168.122.242 closed.
root@lq-d25:~# ssh root@192.168.122.242
Web console: https://mvp1.vpd.com:9090/ or
https://192.168.122.242:9090/

Last login: Fri Feb 28 20:10:38 2025 from 192.168.122.1
```

11. Verificar que la partición sigue disponible después del reinicio:

```
root@mvp1:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0    0    1G  0 disk
└─sda1       8:1    0   512M  0 part
sr0         11:0    1 1024M  0 rom
zram0       251:0    0   1,9G  0 disk [SWAP]
vda         252:0    0    10G  0 disk
├─vda1      252:1    0     1M  0 part
└─vda2      252:2    0     1G  0 part /boot
```

```
└─vda3          252:3    0    9G  0 part
   └─fedora-root 253:0    0    9G  0 lvm  /
```

12. Montar el sistema de archivos:

```
root@mvp1:~# mkdir -p /mnt/nuevo_disco
root@mvp1:~# mount /dev/sda1 /mnt/nuevo_disco
```

13. Verificar el montaje:

```
root@mvp1:~# df -h /mnt/nuevo_disco
S.ficheros      Tamaño Usados  Disp Uso% Montado en
/dev/sda1       504M    24K   478M   1% /mnt/nuevo_disco
```

14. Verificar detalles de la partición:

```
root@mvp1:~# fdisk -l /dev/sda
Disk /dev/sda: 1 GiB, 1073741824 bytes, 2097152 sectors
Disk model: QEMU HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xc4abc78f

Device      Boot Start      End Sectors  Size Id Type
/dev/sda1   2048 1050623 1048576   512M 83 Linux
```

15. Crear el archivo test.txt requerido:

```
root@mvp1:~# touch /mnt/nuevo_disco/test.txt
```

16. Verificar que se ha creado correctamente:

```
root@mvp1:~# ls -l /mnt/nuevo_disco
total 16
drwx-----. 2 root root 16384 feb 28 20:15 lost+found
-rw-r--r--. 1 root root      0 feb 28 20:19 test.txt
```

17. Configurar el montaje automático en el arranque:

```
root@mvp1:~# echo "/dev/sda1 /mnt/nuevo_disco xfs defaults 0 0" >>
/etc/fstab
```

18. Reiniciar para verificar el montaje automático:

```
root@mvp1:~# reboot
root@mvp1:~# Connection to 192.168.122.242 closed by remote host.
Connection to 192.168.122.242 closed.
root@lq-d25:~# ssh root@192.168.122.242
Web console: https://mvp1.vpd.com:9090/ or
https://192.168.122.242:9090/

Last login: Fri Feb 28 20:24:06 2025 from 192.168.122.1
```

19. Verificar que el sistema de archivos se ha montado automáticamente:

```
root@mvp1:~# ls -l /mnt/nuevo_disco
total 0
-rw-r--r--. 1 root root 0 feb 28 20:28 test.txt
```

Fase 2. Contenedores locales

Tarea 2. Creación de una nueva partición en el host anfitrión

En esta tarea se debe crear una partición lógica nueva de 1GB en el host anfitrión y asociarla a la máquina mvp3.

Nota: Esta tarea requiere crear una partición en un disco que contiene particiones utilizadas por otros estudiantes. Si no se hace correctamente, puede dañar el sistema. No se debe realizar esta tarea hasta estar seguro de cómo se particionan unidades de disco.

A diferencia de la Tarea 1, donde trabajamos con un volumen virtual (archivo), en esta tarea vamos a trabajar con una partición física real del host anfitrión. Es fundamental identificar correctamente el disco y la partición extendida para no afectar a otros estudiantes.

1. Identificar las particiones existentes en el host anfitrión:

```
root@lq-d25:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0    0 953,9G  0 disk
├─sda1       8:1    0   50M  0 part
├─sda2       8:2    0 477,2G  0 part
├─sda3       8:3    0    4G  0 part
├─sda4       8:4    0    1K  0 part
├─sda5       8:5    0  97,7G  0 part
├─sda6       8:6    0  31,1G  0 part [SWAP]
├─sda7       8:7    0 146,5G  0 part /
├─sda8       8:8    0 146,5G  0 part
├─sda9       8:9    0    1G  0 part
└─sda10      8:10   0    2G  0 part
sdb          8:16   1    0B  0 disk
zram0       252:0    0    8G  0 disk [SWAP]
```

La salida del comando lsblk muestra que nos encontramos en un sistema con un único disco principal (/dev/sda) que ya contiene varias particiones. Según los requisitos de la práctica, debemos trabajar con el laboratorio LQ-D y crear la nueva partición lógica en la partición extendida del disco /dev/sda.

2. Crear una partición lógica de 1GB utilizando fdisk:

```
root@lq-d25:~# fdisk /dev/sda

Welcome to fdisk (util-linux 2.39.4).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.
```

This disk is currently in use - repartitioning is probably a bad idea.
It's recommended to umount all file systems, and swapoff all swap partitions on this disk.

Command (m for help): p

Disk /dev/sda: 953,87 GiB, 1024209543168 bytes, 2000409264 sectors
Disk model: SSD-1TB
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xd5de6562

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sda1	*	2048	104447	102400	50M	7	
HPFS/NTFS/exFAT							
/dev/sda2		104448	1000833023	1000728576	477,2G	7	
HPFS/NTFS/exFAT							
/dev/sda3		1000833024	1009221631	8388608	4G	83	Linux
/dev/sda4		1009221632	2000408575	991186944	472,6G	f	W95
Ext'd (LBA)							
/dev/sda5		1009225728	1214025727	204800000	97,7G	83	Linux
/dev/sda6		1214027776	1279252479	65224704	31,1G	82	Linux
swap / Solaris							
/dev/sda7		1279254528	1586454527	307200000	146,5G	83	Linux
/dev/sda8		1586456576	1893656575	307200000	146,5G	83	Linux
/dev/sda9		1893658624	1895755775	2097152	1G	83	Linux
/dev/sda10		1895757824	1899952127	4194304	2G	83	Linux

Command (m for help): n
All primary partitions are in use.
Adding logical partition 11
First sector (1899954176-2000408575, default 1899954176):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (1899954176-2000408575, default 2000408575): +1G

Created a new partition 11 of type 'Linux' and of size 1 GiB.

Command (m for help): w
The partition table has been altered.
Syncing disks.

Explicación del proceso de particionado:

- **p**: Muestra la tabla de particiones actual para verificar el espacio y las particiones existentes
- **n**: Crea una nueva partición
- Ya que todas las particiones primarias están en uso, fdisk crea automáticamente una partición lógica dentro de la partición extendida
- Se aceptan los valores predeterminados para el sector inicial

- Se especifica +1G para el último sector, creando una partición de 1GB
- w: Escribe los cambios en la tabla de particiones y sale

3. Verificar la creación de la nueva partición:

```
root@lq-d25:~# lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
sda          8:0    0 953,9G  0 disk
├─sda1       8:1    0   50M  0 part
├─sda2       8:2    0 477,2G  0 part
├─sda3       8:3    0    4G  0 part
├─sda4       8:4    0    1K  0 part
├─sda5       8:5    0  97,7G  0 part
├─sda6       8:6    0  31,1G  0 part [SWAP]
├─sda7       8:7    0 146,5G  0 part /
├─sda8       8:8    0 146,5G  0 part
├─sda9       8:9    0    1G  0 part
├─sda10      8:10   0    2G  0 part
├─sda11      8:11   0    1G  0 part
sdb          8:16    1    0B  0 disk
zram0       252:0    0    8G  0 disk [SWAP]
```

La nueva partición /dev/sda11 aparece en la lista con el tamaño correcto.

4. Asociar la partición creada a la máquina virtual mvp3 como disco sdb:

```
root@lq-d25:~# virsh attach-disk mvp3 /dev/sda11 sdb --config --
type disk --driver qemu --subdriver raw
El disco ha sido asociado exitosamente
```

Explicación del comando:

- virsh attach-disk: Asocia un disco a una máquina virtual
- mvp3: Nombre de la máquina virtual
- /dev/sda11: Ruta al dispositivo de partición física
- sdb: Nombre del dispositivo en la máquina virtual
- --config: Hace que la configuración sea persistente tras reiniciar
- --type disk: Especifica que se trata de un disco
- --driver qemu: Usar el controlador QEMU para acceder al disco
- --subdriver raw: Especifica el formato raw del disco

5. Iniciar sesión en la máquina virtual mvp3 para verificar que el nuevo disco está disponible:

```
root@lq-d25:~# virsh domifaddr mvp3
Nombre      dirección MAC      Protocol  Address
-----
vnet0       00:16:3e:37:a0:03   ipv4      192.168.122.242/24
```

```

root@lq-d25:~# ssh root@192.168.122.242
Web console: https://mvp1.vpd.com:9090/ or
https://192.168.122.242:9090/

Last login: Thu Mar  6 19:23:40 2025
root@mvp1:~#

```

6. Verificar que el nuevo disco está presente en la máquina virtual:

```

root@mvp1:~# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda                  8:0    0    1G  0 disk
└─sda1               8:1    0   512M  0 part
sdb                  8:16    0    1G  0 disk
zram0               251:0    0   1,9G  0 disk [SWAP]
vda                 252:0    0   10G  0 disk
├─vda1              252:1    0    1M  0 part
├─vda2              252:2    0    1G  0 part /boot
└─vda3              252:3    0    9G  0 part
    └─fedora-root    253:0    0    9G  0 lvm  /

```

Como se puede observar, el disco aparece como `/dev/sdb` en la máquina virtual, tal como se especificó en el comando `attach-disk`.

Nomenclatura de Discos en VMs: Linux nombra los discos secuencialmente (`sda`, `sdb`, `sdc`, etc.) según el orden en que los detecta el sistema operativo dentro de la VM. El nombre asignado (`sdb` en este caso) no tiene relación directa con el nombre de la partición física en el host (`sda11`), sino con el orden en que se añadió a la VM.

7. Crear un sistema de archivos XFS directamente en el disco (sin particionar):

```

root@mvp1:~# mkfs.xfs /dev/sdb
meta-data=/dev/sdb            isize=512    agcount=4,
agsize=65536 blks
                =               sectsz=512    attr=2, projid32bit=1
                =               crc=1        finobt=1, sparse=1,
rmapbt=1
                =               reflink=1    bigtime=1
inobtcount=1 nnext64=1
data        =                  bsize=4096    blocks=262144,
imaxpct=25
                =                  sunit=0     swidth=0 blks
naming      =version 2          bsize=4096    ascii-ci=0, ftype=1
log         =internal log      bsize=4096    blocks=16384,
version=2
                =                  sectsz=512    sunit=0 blks, lazy-
count=1
realtime    =none              extsz=4096    blocks=0, rtextents=0
Discarding blocks...Done.

```

Explicación del comando:

- `mkfs.xfs`: Crea un sistema de archivos XFS
- `/dev/sdb`: Dispositivo donde se creará el sistema de archivos (en el disco completo, sin particionar)

8. Montar el sistema de archivos y crear archivo de prueba:

```
root@mvp1:~# mkdir -p /mnt/disco_fisico
root@mvp1:~# mount /dev/sdb /mnt/disco_fisico
root@mvp1:~# touch /mnt/disco_fisico/test.txt
root@mvp1:~# ls -l /mnt/disco_fisico
total 0
-rw-r--r--. 1 root root 0 mar  6 19:51 test.txt
```

9. Configurar el montaje automático en el directorio /VDB:

```
root@mvp1:~# echo "/dev/sdb /mnt/disco_fisico xfs defaults 0 0" >>
/etc/fstab
```

10. Verificaciones adicionales:

Verificar el montaje:

```
root@mvp1:~# df -h | grep sdb
/dev/sdb                960M    51M   910M    6% /mnt/disco_fisico
```

Verificar que el disco no está particionado:

```
root@mvp1:~# fdisk -l /dev/sdb
Disk /dev/sdb: 1 GiB, 1073741824 bytes, 2097152 sectors
Disk model: QEMU HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
```

La salida de `fdisk` confirma que el disco no tiene una tabla de particiones, lo que es correcto según los requisitos de la tarea.

Diferencia entre los discos `sda` y `vda`:

- `/dev/sda`: Dispositivo emulado como disco SATA/SCSI.
- `/dev/vda`: Dispositivo paravirtualizado (`virtio`) que ofrece mejor rendimiento al reducir la emulación de hardware. Esta diferencia en la nomenclatura refleja el tipo de controlador de disco utilizado por el hipervisor para presentar el almacenamiento a la máquina virtual.

Tarea 3. Creación de un contenedor en una partición lógica

El objetivo de esta tarea es crear un contenedor de almacenamiento donde poder crear volúmenes (discos virtuales) para los sistemas invitados.

Para ello, se debe crear una partición lógica nueva de 2GB utilizando fdisk, siguiendo las mismas consideraciones sobre el laboratorio mencionadas en la tarea anterior.

1. Crear una partición lógica de 2GB:

```
root@lq-d25:~# lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
sda          8:0    0 953,9G  0 disk
├─sda1       8:1    0    50M  0 part
├─sda2       8:2    0 477,2G  0 part
├─sda3       8:3    0     4G  0 part
├─sda4       8:4    0     1K  0 part
├─sda5       8:5    0  97,7G  0 part
├─sda6       8:6    0  31,1G  0 part [SWAP]
├─sda7       8:7    0 146,5G  0 part /
├─sda8       8:8    0 146,5G  0 part
├─sda9       8:9    0     1G  0 part
├─sda10      8:10   0     2G  0 part
├─sda11      8:11   0     1G  0 part
sdb          8:16   1      0B  0 disk
zram0       252:0    0     8G  0 disk [SWAP]
```

```
root@lq-d25:~# fdisk /dev/sda

Welcome to fdisk (util-linux 2.39.4).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

This disk is currently in use - repartitioning is probably a bad
idea.
It's recommended to umount all file systems, and swapoff all swap
partitions on this disk.

Command (m for help): p

Disk /dev/sda: 953,87 GiB, 1024209543168 bytes, 2000409264 sectors
Disk model: SSD-1TB
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xd5de6562

Device      Boot      Start          End      Sectors  Size Id Type
```

```

/dev/sda1 *          2048      104447      102400      50M  7
HPFS/NTFS/exFAT
/dev/sda2          104448 1000833023 1000728576 477,2G  7
HPFS/NTFS/exFAT
/dev/sda3          1000833024 1009221631      8388608      4G 83 Linux
/dev/sda4          1009221632 2000408575 991186944 472,6G  f W95
Ext'd (LBA)
/dev/sda5          1009225728 1214025727 204800000 97,7G 83 Linux
/dev/sda6          1214027776 1279252479 65224704 31,1G 82 Linux
swap / Solaris
/dev/sda7          1279254528 1586454527 307200000 146,5G 83 Linux
/dev/sda8          1586456576 1893656575 307200000 146,5G 83 Linux
/dev/sda9          1893658624 1895755775 2097152 1G 83 Linux
/dev/sda10         1895757824 1899952127 4194304 2G 83 Linux
/dev/sda11         1899954176 1902051327 2097152 1G 83 Linux

Command (m for help): n
All primary partitions are in use.
Adding logical partition 12
First sector (1902053376-2000408575, default 1902053376):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (1902053376-
2000408575, default 2000408575): +2G

Created a new partition 12 of type 'Linux' and of size 2 GiB.

Command (m for help): w
The partition table has been altered.
Syncing disks.

```

Explicación del proceso de particionado:

- p: Muestra la tabla de particiones actual para verificar el espacio y las particiones existentes
- n: Crea una nueva partición
- Ya que todas las particiones primarias están en uso, fdisk crea automáticamente la partición lógica 12 dentro de la partición extendida
- Se aceptan los valores predeterminados para el sector inicial
- Se especifica +2G para el último sector, creando una partición de 2GB
- w: Escribe los cambios en la tabla de particiones y sale

2. Verificar la creación de la nueva partición:

```

root@lq-d25:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0    0 953,9G  0 disk
├─sda1       8:1    0   50M  0 part
├─sda2       8:2    0 477,2G  0 part
├─sda3       8:3    0    4G  0 part
├─sda4       8:4    0    1K  0 part
├─sda5       8:5    0  97,7G  0 part
├─sda6       8:6    0  31,1G  0 part [SWAP]
└─sda7       8:7    0 146,5G  0 part /

```

```

├─sda8      8:8      0 146,5G  0 part
├─sda9      8:9      0      1G  0 part
├─sda10     8:10     0      2G  0 part
├─sda11     8:11     0      1G  0 part
└─sda12     8:12     0      2G  0 part
sdb         8:16     1      0B  0 disk
zram0       252:0     0      8G  0 disk [SWAP]

```

3. Crear un sistema de archivos ext4 en la nueva partición:

```

root@lq-d25:~# mkfs.ext4 /dev/sda12
mke2fs 1.47.0 (5-Feb-2023)
Descartando los bloques del dispositivo: hecho
Se está creando un sistema de ficheros con 524288 bloques de 4k y
131072 nodos-i
UUID del sistema de ficheros: d49322db-bde4-4a05-8e49-dc3c37f7484d
Respaldos del superbloque guardados en los bloques:
    32768, 98304, 163840, 229376, 294912

Reservando las tablas de grupo: hecho
Escribiendo las tablas de nodos-i: hecho
Creando el fichero de transacciones (16384 bloques): hecho
Escribiendo superbloques y la información contable del sistema de
ficheros: 0/1hecho

```

4. Crear un directorio para montar la partición:

```

root@lq-d25:~# mkdir -p /var/lib/libvirt/Pool_Particion

```

5. Montar la partición en el directorio:

```

root@lq-d25:~# mount /dev/sda12 /var/lib/libvirt/Pool_Particion

```

6. Configurar el montaje automático en el anfitrión:

Obtener el UUID de la partición:

```

root@lq-d25:~# blkid /dev/sda12
/dev/sda12: UUID="d49322db-bde4-4a05-8e49-dc3c37f7484d"
BLOCK_SIZE="4096" TYPE="ext4" PARTUUID="d5de6562-0c"

```

Añadir entrada al fichero fstab:

```

root@lq-d25:~# echo "UUID=d49322db-bde4-4a05-8e49-dc3c37f7484d
/var/lib/libvirt/Pool_Particion ext4 defaults 0 0" >> /etc/fstab

```

7. Definir el storage pool en libvirt:

```
root@lq-d25:~# virsh pool-define-as Contenedor_Particion fs --
source-dev /dev/sda12 --target /var/lib/libvirt/Pool_Particion
El grupo Contenedor_Particion ha sido definido
```

Explicación del comando:

- `virsh pool-define-as`: Define un nuevo storage pool
- `Contenedor_Particion`: Nombre del pool
- `fs`: Tipo de pool (filesystem)
- `--source-dev /dev/sda12`: Dispositivo de origen para el pool
- `--target /var/lib/libvirt/Pool_Particion`: Ruta al directorio donde se almacenarán los volúmenes
- `/var/lib/libvirt/Pool_Particion`: Ruta al directorio donde se almacenarán los volúmenes

Tipos de Storage Pool: Libvirt soporta varios tipos de pools (`dir`, `fs`, `netfs`, `logical`, `disk`, etc.). El tipo `fs` se usa cuando la fuente es un dispositivo de bloque (como una partición `/dev/sda12`) que ya tiene un sistema de archivos y está montado en el host. El tipo `dir` se usa cuando la fuente es simplemente un directorio existente en el host. `netfs` se usa para NFS.

8. Construir el pool:

```
root@lq-d25:~# virsh pool-build Contenedor_Particion
El pool Contenedor_Particion ha sido compilado
```

9. Iniciar el pool:

```
root@lq-d25:~# virsh pool-start Contenedor_Particion
Se ha iniciado el grupo Contenedor_Particion
```

```
root@lq-d25:~# virsh pool-autostart Contenedor_Particion
El grupo Contenedor_Particion ha sido marcado como iniciable
automáticamente
```

10. Verificar que el pool se ha creado correctamente:

```
root@lq-d25:~# virsh pool-list --all
Nombre                Estado  Inicio automático
-----
Contenedor_Particion  activo  si
default               activo  si
```

11. Crear un volumen qcow2 en el storage pool:

```
root@lq-d25:~# virsh vol-create-as Contenedor_Particion Vol2_p3 1G
--format qcow2
Se ha creado el volumen Vol2_p3
```

Explicación del comando:

- **virsh vol-create-as:** Crea un volumen en un pool de almacenamiento
- **Contenedor_Particion:** Nombre del pool donde se creará el volumen
- **Vol2_p3:** Nombre del volumen a crear
- **1G:** Tamaño del volumen (1 Gigabyte)
- **--format qcow2:** Especifica el formato qcow2 para el disco virtual

12. Verificar la creación del volumen:

```
root@lq-d25:~# virsh vol-list Contenedor_Particion
Nombre          Ruta
-----
lost+found       /var/lib/libvirt/Pool_Particion/lost+found
Vol2_p3          /var/lib/libvirt/Pool_Particion/Vol2_p3
```

13. Asociar el volumen a la máquina virtual mvp3 como disco vdb:

```
root@lq-d25:~# virsh attach-disk mvp3
/var/lib/libvirt/Pool_Particion/Vol2_p3 vdb --driver qemu --
subdriver qcow2 --type disk --config
El disco ha sido asociado exitosamente
```

Explicación del comando:

- **virsh attach-disk:** Asocia un disco a una máquina virtual
- **mvp3:** Nombre de la máquina virtual
- **/var/lib/libvirt/Pool_Particion/Vol2_p3:** Ruta al archivo de volumen
- **vdb:** Nombre del dispositivo en la máquina virtual
- **--driver qemu:** Usar el controlador QEMU para acceder al disco
- **--subdriver qcow2:** Especifica el formato qcow2 del disco
- **--type disk:** Especifica que se trata de un disco
- **--config:** Hace que la configuración sea persistente tras reiniciar

14. Verificar que el disco se ha asociado correctamente:

```
root@lq-d25:~# virsh domblklist mvp3
Destino  Fuente
```

```

-----
vda      /var/lib/libvirt/images/mvp3.qcow2
vdb      /var/lib/libvirt/Pool_Particion/Vol2_p3
sda      /var/lib/libvirt/images/Vol1_p3.img
sdb      /dev/sda11

```

15. Iniciar sesión en la máquina virtual:

```

root@lq-d25:~# ssh root@192.168.122.242
Web console: https://mvp1.vpd.com:9090/ or
https://192.168.122.242:9090/

Last login: Thu Mar  6 20:04:54 2025 from 192.168.122.1

```

16. Verificar que el disco está disponible:

```

root@mvp1:~# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda                  8:0    0    1G  0 disk
└─sda1               8:1    0   512M  0 part
sdb                  8:16   0    1G  0 disk /mnt/disco_fisico
zram0               251:0    0   1,9G  0 disk [SWAP]
vda                 252:0    0   10G  0 disk
├─vda1              252:1    0    1M  0 part
├─vda2              252:2    0    1G  0 part /boot
└─vda3              252:3    0    9G  0 part
   └─fedora-root     253:0    0    9G  0 lvm  /
vdb                 252:16   0    1G  0 disk /VDB

```

17. Crear un sistema de archivos XFS en el disco completo (sin particionar):

```

root@mvp1:~# mkfs.xfs /dev/vdb
meta-data=/dev/vdb            isize=512    agcount=4,
agsize=65536 blks
        =                       sectsz=512    attr=2, projid32bit=1
        =                       crc=1         finobt=1, sparse=1,
rmapbt=1
        =                       reflink=1     bigtime=1
inobtcount=1 nrext64=1
data      =                       bsize=4096   blocks=262144,
imaxpct=25
        =                       sunit=0      swidth=0 blks
naming    =version 2           bsize=4096   ascii-ci=0, ftype=1
log       =internal log       bsize=4096   blocks=16384,
version=2
        =                       sectsz=512   sunit=0 blks, lazy-
count=1
realtime  =none                extsz=4096   blocks=0, rtextents=0
Discarding blocks...Done.

```

18. Crear directorio de montaje y montar el disco:

```
root@mvp1:~# mkdir -p /VDB
root@mvp1:~# mount /dev/vdb /VDB
```

20. Crear archivo de prueba:

```
root@mvp1:~# touch /VDB/test.txt
root@mvp1:~# ls -l /VDB
total 0
-rw-r--r--. 1 root root 0 mar  6 20:29 test.txt
```

21. Configurar el montaje automático en la máquina virtual:

Obtener el UUID del disco:

```
root@mvp1:~# blkid /dev/vdb
/dev/vdb: UUID="0051b7c0-22f9-4d30-9dc4-cc70f44ee818"
BLOCK_SIZE="512" TYPE="xfs"
```

Añadir entrada a fstab:

```
root@mvp1:~# echo "UUID=0051b7c0-22f9-4d30-9dc4-cc70f44ee818 /VDB
xfs defaults 0 0" >> /etc/fstab
```

19. Verificar la configuración:

```
root@mvp1:~# mount -a
mount: (hint) your fstab has been modified, but systemd still uses
the old version; use 'systemctl daemon-reload' to update.

root@mvp1:~# systemctl daemon-reload

root@mvp1:~# df -h | grep /VDB
/dev/vdb                960M    51M  910M    6% /VDB
```

20. Reiniciar la máquina virtual para comprobar que los montajes persisten:

```
root@mvp1:~# reboot
root@mvp1:~# Connection to 192.168.122.242 closed by remote host.
Connection to 192.168.122.242 closed.
root@lq-d25:~# ssh root@192.168.122.242
```

```
Web console: https://mvp1.vpd.com:9090/ or  
https://192.168.122.242:9090/
```

```
Last login: Thu Mar 7 20:35:42 2025 from 192.168.122.1
```

21. Verificar montajes después del reinicio:

```
root@mvp1:~# df -h | grep -E 'nuevo_disco|VDB|VDC'  
/dev/sda1          504M    24K  478M    1% /mnt/nuevo_disco  
/dev/vdb           960M    51M  910M    6% /VDB  
/dev/vdc           960M    51M  910M    6% /VDC
```

```
root@mvp1:~# ls -l /mnt/nuevo_disco/test.txt /VDB/test.txt  
/VDC/test.txt  
-rw-r--r--. 1 root root 0 mar 7 20:15 /mnt/nuevo_disco/test.txt  
-rw-r--r--. 1 root root 0 mar 7 20:18 /VDB/test.txt  
-rw-r--r--. 1 root root 29 mar 7 20:28 /VDC/test.txt
```


Fase 3. Contenedores en Red

Tarea 4. Creación de un contenedor NFS de imágenes ISO

En esta tarea se debe crear un nuevo contenedor de tipo NFS que permita acceder a las imágenes ISO de las distintas distribuciones Fedora exportadas por el servidor `disnas2.dis.ulpgc.es`. Los datos para crear este contenedor son:

- Nombre del contenedor: `CONT_ISOS_COMP`
- Ruta del directorio local asociado al contenedor:
`/var/lib/libvirt/images/ISOS`

1. Crear el directorio local para el montaje NFS:

```
root@lq-d25:~# mkdir -p /var/lib/libvirt/images/ISOS
```

2. Verificar que el servidor NFS está accesible:

```
root@lq-d25:~# ping disnas2.dis.ulpgc.es
PING disnas2.dis.ulpgc.es (10.22.146.216) 56(84) bytes of data.
64 bytes from disnas2.dis.ulpgc.es (10.22.146.216): icmp_seq=1
ttl=64 time=0.237 ms
64 bytes from disnas2.dis.ulpgc.es (10.22.146.216): icmp_seq=2
ttl=64 time=0.321 ms
64 bytes from disnas2.dis.ulpgc.es (10.22.146.216): icmp_seq=3
ttl=64 time=0.337 ms
^C
--- disnas2.dis.ulpgc.es ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2028ms
rtt min/avg/max/mdev = 0.237/0.298/0.337/0.043 ms
```

3. Comprobar los directorios exportados por el servidor:

```
root@lq-d25:~# showmount -e disnas2.dis.ulpgc.es
Export list for disnas2.dis.ulpgc.es:
/disas2-itsi      *
/imagenes        *
```

4. Definir el pool de almacenamiento NFS:

```
root@lq-d25:~# virsh pool-define-as CONT_ISOS_COMP netfs --source-
host disnas2.dis.ulpgc.es --source-path
/imagenes/fedora/41/isos/x86_64 --target
/var/lib/libvirt/images/ISOS
El grupo CONT_ISOS_COMP ha sido definido
```

Explicación del comando:

- `virsh pool-define-as`: Define un nuevo storage pool
- `CONT_ISOS_COMP`: Nombre del pool
- `netfs`: Tipo de pool (sistema de archivos en red)
- `--source-host`: Especifica el host remoto
- `--source-path`: Especifica la ruta exportada en el host remoto
- `--target`: Especifica el directorio de montaje local

5. Construir el pool:

```
root@lq-d25:~# virsh pool-build CONT_ISOS_COMP
El pool CONT_ISOS_COMP ha sido compilado
```

6. Iniciar el pool:

```
root@lq-d25:~# virsh pool-start CONT_ISOS_COMP
Se ha iniciado el grupo CONT_ISOS_COMP
```

7. Desactivar el inicio automático del pool (según requerimientos):

```
root@lq-d25:~# virsh pool-autostart --disable CONT_ISOS_COMP
La marca de inicio automático del grupo CONT_ISOS_COMP se ha
desactivado
```

8. Verificar que el pool se ha creado correctamente:

```
root@lq-d25:~# virsh pool-list --all
Nombre                Estado    Inicio automático
-----
CONT_ISOS_COMP        activo   no
Contenedor_Particion  activo   si
default               activo   si
```

9. Listar los volúmenes disponibles en el pool:

```
root@lq-d25:~# virsh vol-list CONT_ISOS_COMP
Nombre                Ruta
-----
Fedora-Server-netinst-x86_64-41-1.4.iso
/var/lib/libvirt/images/ISOS/Fedora-Server-netinst-x86_64-41-
1.4.iso
```

10. Verificar que el sistema de archivos NFS está montado correctamente:

```
root@lq-d25:~# mount | grep disnas2
disnas2.dis.ulpgc.es:/imagenes/fedora/41/isos/x86_64 on
/var/lib/libvirt/images/ISOS type nfs
(rw,relatime,vers=3,rsize=1048576,wsiz=1048576,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,mountaddr=10.22.146.216,mountvers=3,mountport=57049,mountproto=udp,local_lock=none,addr=10.22.146.216)
```

Tarea 5. Creación de un contenedor NFS para volúmenes de máquinas virtuales

En esta tarea se debe crear un contenedor de almacenamiento NFS que permita almacenar volúmenes virtuales de disco para máquinas virtuales KVM. Los datos para crear este contenedor son:

- Nombre del contenedor: CONT_VOL_COMP
- Host NFS: disnas2.dis.ulpgc.es
- Ruta del directorio en el servidor NFS: /disnas2-itsi
- Directorio local para el punto de montaje:
/var/lib/libvirt/images/COMPARTIDO

1. Crear el directorio local para el montaje NFS:

```
root@lq-d25:~# mkdir -p /var/lib/libvirt/images/COMPARTIDO
```

2. Verificar que el servidor NFS está accesible:

```
root@lq-d25:~# ping disnas2.dis.ulpgc.es
PING disnas2.dis.ulpgc.es (10.22.146.216) 56(84) bytes of data.
64 bytes from disnas2.dis.ulpgc.es (10.22.146.216): icmp_seq=1
ttl=64 time=0.296 ms
64 bytes from disnas2.dis.ulpgc.es (10.22.146.216): icmp_seq=2
ttl=64 time=0.343 ms
64 bytes from disnas2.dis.ulpgc.es (10.22.146.216): icmp_seq=3
ttl=64 time=0.325 ms
^C
--- disnas2.dis.ulpgc.es ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2048ms
rtt min/avg/max/mdev = 0.296/0.321/0.343/0.019 ms
```

3. Comprobar que el directorio /disnas2-itsi está exportado por el servidor NFS:

```
root@lq-d25:~# showmount -e disnas2.dis.ulpgc.es
```

```
Export list for disnas2.dis.ulpgc.es:
/disas2-itsi      *
/imagenes         *
```

4. Definir el pool de almacenamiento NFS:

```
root@lq-d25:~# virsh pool-define-as CONT_VOL_COMP netfs --source-
host disnas2.dis.ulpgc.es --source-path /disnas2-itsi --target
/var/lib/libvirt/images/COMPARTIDO
El grupo CONT_VOL_COMP ha sido definido
```

5. Construir el pool:

```
root@lq-d25:~# virsh pool-build CONT_VOL_COMP
El pool CONT_VOL_COMP ha sido compilado
```

6. Iniciar el pool:

```
root@lq-d25:~# virsh pool-start CONT_VOL_COMP
Se ha iniciado el grupo CONT_VOL_COMP
```

7. Desactivar el inicio automático del pool (según requerimientos):

```
root@lq-d25:~# virsh pool-autostart --disable CONT_VOL_COMP
La marca de inicio automático del grupo CONT_VOL_COMP se ha
desactivado
```

8. Verificar que el pool se ha creado correctamente:

```
root@lq-d25:~# virsh pool-list --all
Nombre                Estado    Inicio automático
-----
CONT_ISOS_COMP        activo    no
CONT_VOL_COMP         activo    no
Contenedor_Particion  activo    si
default               activo    si
```

9. Crear un volumen qcow2 en el pool:

```
root@lq-d25:~# virsh vol-create-as CONT_VOL_COMP
pc25_LQD_ANFITRION1_Vol3_p3 1G --format qcow2 --prealloc-metadata
Se ha creado el volumen pc25_LQD_ANFITRION1_Vol3_p3
```

Explicación del comando:

- **virsh vol-create-as:** Crea un volumen en un pool de almacenamiento
- **CONT_VOL_COMP:** Nombre del pool donde se creará el volumen
- **pc25_LQD_ANFITRION1_Vol3_p3:** Nombre del volumen a crear (incluye identificadores específicos)
- **1G:** Tamaño del volumen (1 Gigabyte)
- **--format qcow2:** Especifica el formato qcow2 para el disco virtual
- **--prealloc-metadata:** Preasigna los metadatos para mejorar el rendimiento

10. Verificar la creación del volumen:

```
root@lq-d25:~# virsh vol-list CONT_VOL_COMP | grep pc25
pc25_LQD_ANFITRION1_Vol3_p3
/var/lib/libvirt/images/COMPARTIDO/pc25_LQD_ANFITRION1_Vol3_p3
```

11. Asociar el volumen a la máquina virtual mvp3 como disco vdc:

```
root@lq-d25:~# virsh attach-disk mvp3
/var/lib/libvirt/images/COMPARTIDO/pc25_LQD_ANFITRION1_Vol3_p3 vdc
--driver qemu --subdriver qcow2 --targetbus virtio --persistent
El disco ha sido asociado exitosamente
```

Explicación del comando:

- **virsh attach-disk:** Asocia un disco a una máquina virtual
- **mvp3:** Nombre de la máquina virtual
- **/var/lib/libvirt/images/COMPARTIDO/pc25_LQD_ANFITRION1_Vol3_p3:** Ruta al archivo de volumen
- **vdc:** Nombre del dispositivo en la máquina virtual
- **--driver qemu:** Usar el controlador QEMU para acceder al disco
- **--subdriver qcow2:** Especifica el formato qcow2 del disco
- **--targetbus virtio:** Utiliza el bus paravirtualizado virtio para mejor rendimiento
- **--persistent:** Hace que la configuración sea persistente tras reiniciar (equivalente a --config)

12. Verificar que el disco se ha asociado correctamente:

```
root@lq-d25:~# virsh domblklist mvp3
Destino  Fuente
-----
vda      /var/lib/libvirt/images/mvp3.qcow2
vdb      /var/lib/libvirt/Pool_Particion/Vol2_p3
vdc      /var/lib/libvirt/images/COMPARTIDO/pc25_LQD_ANFITRION1_Vol3_p3
```

sda	/var/lib/libvirt/images/Vol1_p3.img
sdb	/dev/sda11

13. Iniciar sesión en la máquina virtual:

```
root@lq-d25:~# ssh root@192.168.122.242
Web console: https://mvp1.vpd.com:9090/ or
https://192.168.122.242:9090/

Last login: Thu Mar  6 20:35:42 2025 from 192.168.122.1
```

14. Verificar que el disco está disponible:

```
root@mvp1:~# lsblk
NAME                                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda                                  8:0    0    1G  0 disk
└─sda1                              8:1    0 512M  0 part
sdb                                  8:16   0    1G  0 disk /mnt/disco_fisico
zram0                               251:0   0 1,9G  0 disk [SWAP]
vda                                  252:0   0   10G  0 disk
├─vda1                              252:1   0    1M  0 part
├─vda2                              252:2   0    1G  0 part /boot
└─vda3                              252:3   0    9G  0 part
    └─fedora-root                    253:0   0    9G  0 lvm  /
vdb                                  252:16  0    1G  0 disk /VDB
vdc                                  252:32  0    1G  0 disk
```

15. Crear un sistema de archivos XFS en el disco completo (sin particionar):

```
root@mvp1:~# mkfs.xfs /dev/vdc
meta-data=/dev/vdc                isize=512    agcount=4,
agsize=65536 blks
        =                       sectsz=512    attr=2, projid32bit=1
        =                       crc=1        finobt=1, sparse=1,
rmapbt=1
        =                       reflink=1    bigtime=1
inobtcount=1 nnext64=1
data      =                       bsize=4096   blocks=262144,
imaxpct=25
        =                       sunit=0      swidth=0 blks
naming    =version 2             bsize=4096   ascii-ci=0, ftype=1
log       =internal log         bsize=4096   blocks=16384,
version=2
        =                       sectsz=512   sunit=0 blks, lazy-
count=1
realtime  =none                 extsz=4096   blocks=0, rtextents=0
Discarding blocks...Done.
```

16. Crear directorio de montaje y montar el disco:

```
root@mvp1:~# mkdir -p /VDC
root@mvp1:~# mount /dev/vdc /VDC
```

17. Crear archivo de prueba con contenido:

```
root@mvp1:~# echo "Este es un archivo de prueba" > /VDC/test.txt
root@mvp1:~# cat /VDC/test.txt
Este es un archivo de prueba
```

18. Configurar el montaje automático en la máquina virtual:

Obtener el UUID del disco:

```
root@mvp1:~# blkid /dev/vdc
/dev/vdc: UUID="877f6a37-3466-4e81-95f7-9cf4e64421a4"
BLOCK_SIZE="512" TYPE="xfs"
```

Añadir entrada a fstab:

```
root@mvp1:~# echo "UUID=877f6a37-3466-4e81-95f7-9cf4e64421a4 /VDC
xfs defaults 0 0" >> /etc/fstab
```

19. Verificar la configuración:

```
root@mvp1:~# mount -a
mount: (hint) your fstab has been modified, but systemd still uses
the old version; use 'systemctl daemon-reload' to update.

root@mvp1:~# systemctl daemon-reload

root@mvp1:~# df -h | grep /VDC
/dev/vdc                960M    51M   910M    6% /VDC
```

20. Reiniciar la máquina virtual para comprobar que los montajes persisten:

```
root@mvp1:~# reboot
root@mvp1:~# Connection to 192.168.122.242 closed by remote host.
Connection to 192.168.122.242 closed.
root@lq-d25:~# ssh root@192.168.122.242
Web console: https://mvp1.vpd.com:9090/ or
https://192.168.122.242:9090/

Last login: Thu Mar  7 20:35:42 2025 from 192.168.122.1
```

21. Verificar montajes después del reinicio:

```
root@mvp1:~# df -h | grep -E 'nuevo_disco|VDB|VDC'
```

/dev/sda1	504M	24K	478M	1%	/mnt/nuevo_disco
/dev/vdb	960M	51M	910M	6%	/VDB
/dev/vdc	960M	51M	910M	6%	/VDC

```
root@mvp1:~# ls -l /mnt/nuevo_disco/test.txt /VDB/test.txt /VDC/test.txt
```

-rw-r--r--.	1	root	root	0	mar	7	20:15	/mnt/nuevo_disco/test.txt
-rw-r--r--.	1	root	root	0	mar	7	20:18	/VDB/test.txt
-rw-r--r--.	1	root	root	29	mar	7	20:28	/VDC/test.txt

Pruebas y Validación

Verificación de volúmenes locales

1. Obtener la configuración XML del volumen Vol1_p3:

```
root@lq-d25:~# virsh vol-dumpxml Vol1_p3.img --pool default
<volume type='file'>
  <name>Vol1_p3.img</name>
  <key>/var/lib/libvirt/images/Vol1_p3.img</key>
  <capacity unit='bytes'>1073741824</capacity>
  <allocation unit='bytes'>1073745920</allocation>
  <physical unit='bytes'>1073741824</physical>
  <target>
    <path>/var/lib/libvirt/images/Vol1_p3.img</path>
    <format type='raw'></format>
    <permissions>
      <mode>0600</mode>
      <owner>107</owner>
      <group>107</group>
      <label>system_u:object_r:virt_image_t:s0</label>
    </permissions>
    <timestamps>
      <atime>1743186522.682454277</atime>
      <mtime>1742588756.589402740</mtime>
      <ctime>1742589707.294322444</ctime>
      <btime>0</btime>
    </timestamps>
  </target>
</volume>
```

2. Obtener la configuración XML del pool Contenedor_Particion:

```
root@lq-d25:~# virsh pool-dumpxml Contenedor_Particion
<pool type='fs'>
  <name>Contenedor_Particion</name>
  <uuid>27a6ba6a-0480-4509-8638-8e63da52e934</uuid>
  <capacity unit='bytes'>2040373248</capacity>
  <allocation unit='bytes'>3424256</allocation>
  <available unit='bytes'>2036948992</available>
  <source>
    <device path='/dev/sda14'></device>
    <format type='auto'></format>
  </source>
  <target>
    <path>/var/lib/libvirt/Pool_Particion</path>
    <permissions>
      <mode>0755</mode>
      <owner>0</owner>
      <group>0</group>
      <label>system_u:object_r:unlabeled_t:s0</label>
    </permissions>
  </target>
</pool>
```

```
</permissions>
</target>
</pool>
```

3. Verificar el estado del pool Contenedor_Particion:

```
root@lq-d25:~# virsh pool-info Contenedor_Particion
Nombre:          Contenedor_Particion
UUID:           27a6ba6a-0480-4509-8638-8e63da52e934
Estado:          ejecutando
Persistente:     si
Autoinicio:      si
Capacidad:       1,90 GiB
Ubicación:       3,27 MiB
Disponible:      1,90 GiB
```

Verificación de contenedores en red

1. Obtener la configuración XML del pool CONT_ISOS_COMP:

```
root@lq-d25:~# virsh pool-dumpxml CONT_ISOS_COMP
<pool type='netfs'>
  <name>CONT_ISOS_COMP</name>
  <uuid>ec08bbd7-a213-45a7-aa8c-3f67a2eec145</uuid>
  <capacity unit='bytes'>0</capacity>
  <allocation unit='bytes'>0</allocation>
  <available unit='bytes'>0</available>
  <source>
    <host name='disnas2.dis.ulpgc.es'>/>
    <dir path='/imagenes/fedora/41/isos/x86_64'>/>
    <format type='auto'>/>
  </source>
  <target>
    <path>/var/lib/libvirt/images/ISOS</path>
  </target>
</pool>
```

2. Obtener la configuración XML del pool CONT_VOL_COMP:

```
root@lq-d25:~# virsh pool-dumpxml CONT_VOL_COMP
<pool type='netfs'>
  <name>CONT_VOL_COMP</name>
  <uuid>d131e98a-c98b-4bd5-ae8b-453912a010c8</uuid>
  <capacity unit='bytes'>0</capacity>
  <allocation unit='bytes'>0</allocation>
  <available unit='bytes'>0</available>
  <source>
    <host name='disnas2.dis.ulpgc.es'>/>
    <dir path='/disnas2-itsi'>/>
  </source>
</pool>
```

```

    <format type='auto' />
</source>
<target>
  <path>/var/lib/libvirt/images/COMPARTIDO</path>
</target>
</pool>

```

3. Verificar el volumen creado en CONT_VOL_COMP:

```

root@lq-d25:~# virsh vol-dumpxml pc25_LQD_ANFITRION1_Vol3_p3 --pool
CONT_VOL_COMP
<volume type='file'>
  <name>pc25_LQD_ANFITRION1_Vol3_p3</name>

  <key>/var/lib/libvirt/images/COMPARTIDO/pc25_LQD_ANFITRION1_Vol3_p3
</key>
  <capacity unit='bytes'>1073741824</capacity>
  <allocation unit='bytes'>1074098176</allocation>
  <physical unit='bytes'>1074135040</physical>
  <target>
    <path>/var/lib/libvirt/images/COMPARTIDO/pc25_LQD_ANFITRION1_Vo
l3_p3</path>
    <format type='qcow2' />
    <permissions>
      <mode>0600</mode>
      <owner>0</owner>
      <group>0</group>
      <label>system_u:object_r:nfs_t:s0</label>
    </permissions>
    <timestamps>
      <atime>1743169357.576104934</atime>
      <mtime>1742588841.435690406</mtime>
      <ctime>1742589704.476700364</ctime>
      <btime>0</btime>
    </timestamps>
    <clusterSize unit='B'>65536</clusterSize>
  </target>
</volume>

```

Verificación de montajes automáticos

1. Verificar los montajes NFS activos:

```

root@lq-d25:~# mount | grep nfs
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw,relatime)
disnas2.dis.ulpgc.es:/disnas2-itsi on
/var/lib/libvirt/images/COMPARTIDO type nfs
(rw,nosuid,nodev,noexec,relatime,vers=3,rsz=1048576,wsz=1048576
,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,mountaddr=10
.22.146.216,mountvers=3,mountport=57049,mountproto=udp,local_lock=n
one,addr=10.22.146.216)

```

```
disnas2.dis.ulpgc.es:/imagenes/fedora/41/isos/x86_64 on
/var/lib/libvirt/images/ISOS type nfs
(rw,nosuid,nodev,noexec,relatime,vers=3,rsz=1048576,wsz=1048576
,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,mountaddr=10
.22.146.216,mountvers=3,mountport=57049,mountproto=udp,local_lock=n
one,addr=10.22.146.216)
```

2. Verificar el estado de todos los pools:

```
root@lq-d25:~# virsh pool-list --details
Nombre          Estado          Inicio
automático      Persistente     Capacidad       Alojamiento     Disponible
-----
CONT_ISOS_COMP  ejecutando     no              247,03 GiB      1,46 GiB        245,57 GiB
CONT_VOL_COMP   ejecutando     no              395,85 GiB      351,47 GiB      44,37 GiB
Contenedor_Particion ejecutando     si              1,90 GiB        3,27 MiB        1,90 GiB
default         ejecutando     si              143,13 GiB      26,27 GiB      116,85 GiB
ISO             ejecutando     si              143,13 GiB      26,27 GiB      116,85 GiB
```

3. Verificar todos los volúmenes asociados a la máquina virtual:

```
root@lq-d25:~# virsh domblklist mvp3 --details
Tipo    Dispositivo    Destino    Fuente
-----
file    disk          vda        /var/lib/libvirt/images/mvp3.qcow2
file    disk          vdb        /var/lib/libvirt/Pool_Particion/Vol
2_p3
file    disk          vdc        /var/lib/libvirt/images/COMPARTIDO/
pc25_LQD_ANFITRION1_Vol3_p3
file    disk          sda        /var/lib/libvirt/images/Vol1_p3.img
block   disk          sdb        /dev/sda13
```

Conclusiones

Esta práctica ha validado experimentalmente la gestión de diversos mecanismos de almacenamiento para entornos de virtualización KVM/QEMU gestionados con libvirt. Se ha demostrado la capacidad de configurar y operar con éxito:

1. **Volúmenes basados en archivos:** Creación de volúmenes (.img, .qcow2) dentro de pools de tipo directorio (default) y su asociación a máquinas virtuales mediante `virsh vol-create-as` y `virsh attach-disk`.
2. **Pools basados en particiones físicas:** Integración de particiones del sistema anfitrión (/dev/sdaX) como pools de almacenamiento de tipo `fs`, permitiendo el uso directo de dispositivos de bloque por libvirt.
3. **Pools basados en NFS:** Configuración de pools de tipo `netfs` para acceder a recursos compartidos en red (ISO y volúmenes de disco), demostrando la viabilidad del almacenamiento centralizado.

Se ha verificado la correcta implementación de la persistencia de las configuraciones, tanto a nivel de hipervisor (mediante la opción `--config o --persistent` en `virsh`) como a nivel del sistema operativo invitado (mediante la edición de `/etc/fstab` para el montaje automático de sistemas de archivos). La utilización de diferentes tipos de controladores (`raw`, `qcow2`) y buses (SATA/emulado, `virtio`) ha permitido constatar las distintas opciones y funcionalidad disponibles.

La finalización satisfactoria de las tareas evidencia la adquisición de competencias esenciales para el diseño e implementación de infraestructuras de almacenamiento virtualizadas robustas, flexibles y eficientes, adaptadas a distintos requisitos operativos.

Bibliografía

- [1] Red Hat Documentation, «Storage Management Guide,» [En línea]. Available: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/storage_administration_guide/index. [Último acceso: 30 03 2025].
- [2] Z. Y. P. D. N. L. E. J. R. S. Herrmann J, «Documentación oficial de KVM,» 2025. [En línea]. Available: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/virtualization_getting_started_guide/index.
- [3] J. K. P. W. S. Peter Boy, «Fedora Server Documentation,» 2025. [En línea]. Available: <https://docs.fedoraproject.org/en-US/fedora-server>.
- [4] Z. Y. P. D. N. L. E. J. R. S. Herrmann J, «Red Hat Enterprise Linux 7. Virtualization Deployment and Administration Guide. Installing, configuring, and managing virtual machines on a RHEL physical machine,» [En línea]. Available: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/. [Último acceso: 09 03 2025].