

## Capítulo 5: LVM (Logical Volume Management)

LVM es un conjunto de herramientas que permiten la administración de unidades de almacenamiento denominadas volúmenes lógicos. Quizás su característica más distintiva sea que provee flexibilidad para administrar el almacenamiento del sistema de manera que pueda expandirse fácilmente cuando la necesidad así lo requiera.

Una ventaja adicional de LVM es que permite tomar instantáneas de los volúmenes lógicos, las cuales pueden funcionar como copias de respaldo o ser utilizadas para migraciones, pruebas, restauraciones, etc., sin afectar el funcionamiento del sistema en su totalidad.

### Particionado tradicional vs. LVM

Las distribuciones más utilizadas realizan hoy en día la instalación sobre volúmenes lógicos por defecto. ¿Cuáles son las razones por las cuales el tradicional particionado versus LVM (Logical Volume Management) ha perdido terreno?

Para empezar, recordemos que al menos en MBR, hay una limitación en el número de particiones primarias por disco. En el caso de que necesitemos redimensionar una de ellas, se requiere mucho cuidado. Si bien podemos achicar una partición para darle espacio a otra que nos haya quedado pequeña, este procedimiento causa los siguientes inconvenientes:

- **Provoca downtime.** Esto significa que mientras estemos redimensionando las particiones nadie más podrá utilizar el equipo. Si se trata de nuestro equipo personal, los únicos afectados seríamos nosotros. Pero si estamos hablando de un servidor al que necesiten acceder otros usuarios, esta alternativa puede no ser viable.
- **Requiere copias de respaldo.** Si no se dispone de espacio extra en alguna partición, no quedará más remedio que agregar otro disco al equipo. Si queremos utilizar el nuevo dispositivo para /home (por ejemplo), esto implicará mover todos los archivos de los usuarios de la ubicación anterior a la nueva. ¿Por qué esto es un inconveniente? Dependiendo del tamaño del backup, la restauración agregará tiempo al downtime del que hablamos en el punto anterior.

Por otro lado, el redimensionado de los volúmenes lógicos bajo LVM se puede realizar sin interrumpir el funcionamiento normal del equipo. Además, el procedimiento es mucho más rápido que en el particionado tradicional. Los cambios quedan disponibles a los usuarios inmediatamente sin necesidad de reiniciar ningún servicio.

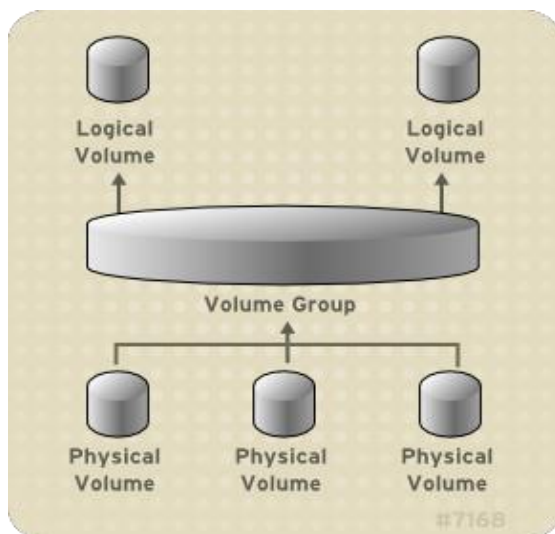


## Estructura de LVM

Un volumen lógico es -a grandes rasgos- un dispositivo de almacenamiento que puede haber sido creado sobre la unión de 1) discos enteros o 2) particiones de los mismos utilizando el administrador de volúmenes lógicos. A diferencia del particionado tradicional, LVM nos provee la flexibilidad de redimensionar volúmenes lógicos fácilmente. Veamos un poco más en detalle de qué se trata el proceso.

La estructura de LVM consiste de los siguientes componentes:

- Dos (o más) discos rígidos o particiones sobre los que se crearán los volúmenes físicos (**PVs**, de **Physical Volume**). En este capítulo utilizaremos 3 discos de 500 MB representados por `/dev/sdb`, `/dev/sdc`, y `/dev/sdd` para crear tres PVs. Es importante aclarar que podemos crear cada uno de ellos sobre el disco entero directamente o particionarlo primero. Si elegimos la segunda opción, debemos indicar que la partición sea del tipo 8e (Linux LVM). En esta oportunidad nosotros crearemos el PV sobre el disco entero sin particionar.
- Un grupo de volúmenes (**VGs**, de **Volume Group**) se crea a partir de uno o más volúmenes físicos. Por simplicidad, podemos considerar a un grupo de volúmenes como una unidad de almacenamiento (similar a un disco entero en el particionado tradicional).
- En cada VG podemos crear varios volúmenes lógicos, los cuales pueden considerarse como análogos de las particiones tradicionales.



## Creación de volúmenes lógicos

Para crear volúmenes físicos sobre los discos mencionados anteriormente, haremos lo siguiente:



```
pvccreate /dev/sdb
pvccreate /dev/sdc
pvccreate /dev/sdd
```

Deberíamos ver el siguiente mensaje de confirmación luego de ejecutar cada comando:

```
root@debiancla:~# pvccreate /dev/sdb
Physical volume "/dev/sdb" successfully created.
root@debiancla:~# pvccreate /dev/sdc
Physical volume "/dev/sdc" successfully created.
root@debiancla:~# pvccreate /dev/sdd
Physical volume "/dev/sdd" successfully created.
root@debiancla:~# pvs /dev/sdb
PV          VG Fmt Attr PSize  PFree
/dev/sdb    lvm2 --- 500,00m 500,00m
root@debiancla:~# pvdisplay /dev/sdc
"/dev/sdc" is a new physical volume of "500,00 MiB"
--- NEW Physical volume ---
PV Name                /dev/sdc
VG Name
PV Size                 500,00 MiB
Allocatable             NO
PE Size                 0
Total PE                0
Free PE                 0
Allocated PE            0
PV UUID                 a1ux07-Mc9n-VwEL-h2cM-KKCj-SqQ7
```

---

*Para mostrar los atributos o ver información sobre un volumen físico podemos utilizar los comandos `pvdisplay` y `pvs`, respectivamente, seguidos de la ruta del PV (`pvdisplay /dev/sdb` o `pvs /dev/sdb`, por ejemplo). Si omitimos esta última veremos la lista de todos los volúmenes físicos que existen en nuestro sistema.*

---

Ahora es momento de crear el grupo de volúmenes (lo llamaremos **vgcurso**). Solamente utilizaremos **/dev/sdb** y **/dev/sdc** por ahora para mostrar más adelante lo fácil que es extender un VG agregando un dispositivo adicional (**/dev/sdd**). Luego utilizaremos `vgs` y `vgdisplay` para ver los detalles del grupo de volúmenes que acabamos de crear.

```
vgcreate vgcurso /dev/sdb /dev/sdc
```



```

root@debiancla:~# vgcreate vgcurso /dev/sdb /dev/sdc
Volume group "vgcurso" successfully created
root@debiancla:~# vgs vgcurso
  VG      #PV #LV #SN Attr   VSize   VFree
vgcurso   2   0   0 wz--n- 992,00m 992,00m
root@debiancla:~# vgdisplay vgcurso
--- Volume group ---
VG Name                vgcurso
System ID
Format                 lvm2
Metadata Areas         2
Metadata Sequence No   1
VG Access               read/write
VG Status               resizable
MAX LV                 0
Cur LV                 0
Open LV                 0
Max PV                  0
Cur PV                 2
Act PV                  2
VG Size                 992,00 MiB
PE Size                 1,00 MiB

```

Tengamos en cuenta que el espacio de almacenamiento disponible en **vgcurso** es ahora de  $2 \times 500 \text{ MB} = 1 \text{ GB}$ . Este hecho nos permite ver otra de las ventajas de LVM: podemos crear una unidad de almacenamiento de considerable capacidad utilizando discos o particiones de menor tamaño.

Para crear un primer LV de tamaño de 600 MB llamado **lvadmin** sobre **vgcurso**, debemos hacer lo siguiente (si omitimos `-n lvadmin`, el nombre del LV será asignado automáticamente por `lvcreate`):

```
lvcreate -n lvadmin -L 600M vgcurso
```

```

root@debiancla:~# lvcreate -n lvadmin -L 600M vgcurso
Logical volume "lvadmin" created.
root@debiancla:~# lvs vgcurso/lvadmin
  LV      VG      Attr      LSize   Pool Origin Data%  Meta%
lvadmin  vgcurso -wi-a----- 600,00m
root@debiancla:~# lvdisplay vgcurso/lvadmin
--- Logical volume ---
LV Path                /dev/vgcurso/lvadmin
LV Name                 lvadmin
VG Name                 vgcurso
LV UUID                 JqZgk7-8apK-pn4l-53D1-E0XT-XmTI-g5HT6
LV Write Access         read/write
LV Creation host, time  debiancla, 2018-10-09 11:05:12 -0300
LV Status                available
# open                   0
LV Size                  600,00 MiB
Current LE               150
Segments                 2

```



En este punto ya podemos crear un sistema de archivos sobre **lvadmin** y montarlo en algún directorio creado previamente para tal fin. Utilizaremos **/mnt/cursoadmin** y prestaremos atención a utilizar la ruta completa al dispositivo esta vez:

```
mkfs.ext4 /dev/vgcurso/lvadmin
mount /dev/vgcurso/lvadmin /mnt/cursoadmin
```

Antes de proseguir, podemos asegurarnos de que el dispositivo fue correctamente montado utilizando cualquiera de los siguientes comandos:

```
mount | grep lvadmin
df -h /mnt/cursoadmin
```

```
root@debiancla:~# mount | grep lvadmin
/dev/mapper/vgcurso-lvadmin on /mnt/cursoadmin type ext4 (rw,relatime,c
root@debiancla:~# df -h /mnt/cursoadmin
S.ficheros          Tamaño Usados  Disp Uso% Montado en
/dev/mapper/vgcurso-lvadmin  575M   912K   532M   1% /mnt/cursoadmin
root@debiancla:~#
```

Como vemos en la imagen de arriba, ya estamos en condiciones de comenzar a utilizar el volumen lógico. Sin embargo, en este capítulo nos enfocaremos en las operaciones posibles a través de LVM más que en utilizar el dispositivo de almacenamiento (tema que ya hemos aprendido previamente).

## Operaciones

Para agregar un PV a un VG, utilizaremos el comando **vgextend** seguido del nombre del VG y finalmente del PV:

```
vgextend vgcurso /dev/sdd
```

Al utilizar el comando **vgs** (o **vgdisplay**, para mayor detalle) seguido del nombre del grupo de volúmenes podemos ver el tamaño de este después de agregar el volumen físico:

```
root@debiancla:~# vgextend vgcurso /dev/sdd
Volume group "vgcurso" successfully extended
root@debiancla:~# vgs vgcurso
VG      #PV #LV #SN Attr   VSize  VFree
vgcurso  3  1  0 wz--n- 1,45g 888,00m
root@debiancla:~#
```

Para achicar un LV existente, debemos prestar especial cuidado ya que si reducimos el tamaño por debajo del espacio utilizado actualmente, ¡perderemos datos irreversiblemente! Habiendo tenido eso en cuenta, vamos a disminuir el tamaño de **lvadmin** a 450 MB:



```
lvresize --resizefs --size 450M vgcurso/lvadmin
```

La opción `--resizefs` redimensiona el volumen lógico y el sistema de archivos asociado en un solo paso. Como el dispositivo está montado actualmente se nos pedirá confirmación para desmontarlo a fin de proceder con la acción que indicamos:

```
root@debiancla:~# lvresize --resizefs --size 450M vgcurso/lvadmin
Rounding size to boundary between physical extents: 452,00 MiB.
Do you want to unmount "/mnt/cursoadmin"? [Y|n] y
fsck de util-linux 2.29.2
/dev/mapper/vgcurso-lvadmin: 11/38400 ficheros (0.0% no contiguos), 6740/153600 bloques
resize2fs 1.43.4 (31-Jan-2017)
Cambiando el tamaño del sistema de ficheros en /dev/mapper/vgcurso-lvadmin a 115712 (4k) bloq
El sistema de ficheros en /dev/mapper/vgcurso-lvadmin tiene ahora 115712 bloques (de 4k).

Size of logical volume vgcurso/lvadmin changed from 600,00 MiB (150 extents) to 452,00 MiB
Logical volume vgcurso/lvadmin successfully resized.
root@debiancla:~#
```

Si no hubiéramos creado previamente un sistema de archivos podríamos haber utilizado simplemente

```
lvreduce -L 450M vgcurso/lvadmin
```

para modificar el tamaño de **lvadmin**.

Veamos ahora cómo crear un segundo LV -al cual llamaremos **lvredes**- y asignarle todo el espacio disponible en **vgcurso**. Vamos a notar que el comando es casi idéntico al que utilizamos en el post anterior, pero el argumento `100%FREE` le indicará a `lvcreate` que utilice todo el espacio disponible para el nuevo volumen lógico.

```
lvcreate -n lvredes -l 100%FREE vgcurso
```

```
root@debiancla:~# lvcreate -n lvredes -l 100%FREE vgcurso
Logical volume "lvredes" created.
root@debiancla:~# lvs vgcurso/lvadmin vgcurso/lvredes
LV      VG      Attr      LSize   Pool Origin Data%  Meta
lvadmin vgcurso -wi-ao---- 452,00m
lvredes vgcurso -wi-a----- 1,01g
root@debiancla:~#
```

Cada una de las operaciones que utilizamos hasta el momento tienen su contrapartida (que detallamos a continuación) y la posibilidad de renombrar los componentes que creamos:

- `pvremove`, `vgremove`, y `lvremove` (seguidas de la ruta al PV, VG, y LV, respectivamente) se utilizan para quitar el dispositivo de la estructura de LVM.





Como es de esperarse, esto NO elimina el dispositivo subyacente - solamente remueve la etiqueta que lo identifica como volumen físico, grupo de volúmenes, o volumen lógico.

- `vgrename` y `lvrename` (seguidas del VG y de la combinación VG/LV actuales, respectivamente) nos permiten renombrar grupos de volúmenes y volúmenes lógicos. Por ejemplo, para renombrar **vgcurso** como **vgcarreralinux** y **lvadmin** como **lvadministrador** debemos hacer

```
vgrename vgcurso vgcarreralinux  
lvrename vgcarreralinux lvadmin lvadministrador
```

```
root@debiancla:~# vgrename vgcurso vgcarreralinux  
Volume group "vgcurso" successfully renamed to "vgcarreralinux"  
root@debiancla:~# lvrename vgcarreralinux lvadmin lvadministrador  
Renamed "lvadmin" to "lvadministrador" in volume group "vgcarreralinux"  
root@debiancla:~# █
```

