# INSTALACIÓN CENTOS (1)

Dada la situación en la que el tamaño de disco por defecto no es suficiente vamos a insertar un nuevo disco duro. Montaremos un nuevo volumen lógico para /var y le insertaremos el disco duro.

Una vez está la máquina recién instalada, en el menú Configuración, añadimos un nuevo disco duro (hasta tener 2). Con `lsblk` comprobamos que hay dos: sda, sdb.

Dar privilegios con `su`.

Para ver el tamaño de los dispositivos: `df -h`

**Hacer RAID**

1º: Crear Physical Volume (PV):

`pvs` (sólo está /dev/sda2)
`pvcreate /dev/sdb`
`pvs` (ahora está /dev/sdb)

2º: Extender el grupo de volúmenes (cl) con un nuevo disco:

```
vgextend cl /dev/sdb
```

A nivel de Logical Volume (LV) hay que crear el nuevo volumen lógico:

```
lvcreate -L 4G -n newvar cl
```

(-L → cómo de grande es)
(-n → nombre)
(cl → para indicar el grupo de volúmenes al que pertenece)

Con `lvdisplay` comprobamos que se ha creado `newvar`.
Tenemos un LV pero no hemos asignado ningún sistema de archivos.

3º: Crear sistema de archivos

```
mkfs -t ext4 /dev/mapper/cl-newvar
```

Otra opción válida sería: `mkfs.xfs /dev...`

Para hacer accesible la información de almacenamiento al sistema de archivos creamos un directorio donde vamos a montar el Logical Volume.

4º: Crear directorio

```
mkdir /media/newvar
```

5º: Montar el sistema de archivos en el directorio creado anteriormente

```
mount /dev/mapper/cl-newvar /media/newvar
```

Comprobamos con

```
mount | grep newvar
```

Hasta este momento el usuario no se ha visto afectado. Antes de copiar la información hay que cambiar el nivel de ejecución (run level) para que todos los usuarios logeados se queden fuera:

6º: Cambiar nivel de ejecución

```
systemctl isolate runlevel1.target
```

7º: Copiar información de /var en el LV

```
cp -a /var/. /media/newvar
```

(El punto (.) permite incluir los archivos ocultos)
Comprobamos: `ls -Z /var`

8º: Desmontar
`umount /media/newvar`

9º: Asignar `/var` al LV `newvar`
`vi /etc/fstab`

Insertar al final:
```
/dev/mapper/cl-newvar      /var      ext4      defaults      0 0
```

Finalizar: Esc + :  wq

10º: Montar fichero fstab por si ha habido algún fallo
`mount -a`

Comprobamos con `lslbk` que todo se ha ejecutado correctamente:

```
[root@localhost ~]# lsblk
NAME          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda             8:0    0    8G  0 disk
├─sda1          8:1    0    1G  0 part /boot
└─sda2          8:2    0    7G  0 part
  ├─cl-root   253:0    0  6,2G  0 lvm  /
  └─cl-swap   253:1    0  820M  0 lvm  [SWAP]
sdb             8:16   0    2G  0 disk
└─cl-newvar   253:2    0    1G  0 lvm  /var
sr0            11:0    1 1024M  0 rom
```

**Cifar Volumen Lógico (VL)**
LUKS (*Linux Unified Key Setup*)

1º: Cifrar /var
En /var se sobreescribe la información, por ello hay que hacer una copia de seguridad por si había algo: `mkdir /varRAID` `cp -a /var/. /varRAID`

2º: Instalar `cryptsetup`
`yum install cryptsetupq`
*Puede que se haya caído internet. Tal y como se hizo al principio de la instalación, habrá que ejecutar `ifup enp0s3`

3º: Desmontar el Volumen Lógico (LV) `/var`
`umount /dev/mapper/cl-newvar`

> Si se produce algún problema por el cual no se permita desmontar podemos saber mediante `lsof` quién usa un recurso. Si no está instalado: `yum install lsof`.
> Ejecutar `lsof /var` para obtener el PID del recurso que lo está utilizando. Seguidamente mataremos el proceso con `kill -9 PID`.

Habría que copiar transitoriamente de `/var` a `/varRAID` porque el propio `yum` utiliza `/var`, pero en este caso no lo vamos a hacer.

4º: Formatear. Cifrar el LV
acción objeto
`cryptsetup luksFormat /dev/mapper/cl-newvar`
Introducir YES y la contraseña junto a su verificación.

5º: Activar el LV cifrado
`cryptsetup luksOpen /dev/mapper/cl-newvar cl-newvar_crypt`
Introducir contraseña.
Se puede comprobar con `blkid`

6º: Crear Sistema de Archivos (S.A.) y montarlo
`mkdir /media/newvar_crypt`
`mkfs -t ext4 /dev/mapper/cl-newvar_crypt`
`mount /dev/mapper/cl-newvar_crypt /media/newvar_crypt`

7º: Copiar
`cp -a /varRAID/. /media/newvar_crypt/`

8º: Actualizar `fstab` y `crypttab`
Este paso es delicado porque hacemos referencia al UUID, que se obtiene con `blkid`, que posteriormente se copia con `grep (blkid | grep crypto)`
Comprobamos que está creado `crypttab`: `less /etc/crypttab` (q quitar)

`blkid | grep crypto >> /etc/crypttab`

El archivo `/etc/crypttab` para cada elemento identificado por UUID se encarga de activarlo por el archivo especificado y una vez activado le da una denominación. Con ese nombre, `fstab` coge el volumen ya activado y monta ahí `/var`.

Editar `/etc/crypttab`
`vi /etc/crypttab`

pmraid1-newvar_crypt UUID=.....* none

*sólo números y letras con guiones

Editar `/etc/fstab`

```
vi /etc/fstab
```

Añadir **`_crypt`** al nombre, de modo que quede:

```
/dev/mapper/cl-newvar_crypt      /var      ext4      defaults      0 0
```

Comprobamos con `lsblk` que todo se ha ejecutado correctamente:

```
[root@localhost ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda                   8:0    0    8G  0 disk
├─sda1                8:1    0    1G  0 part  /boot
└─sda2                8:2    0    7G  0 part
  ├─cl-root         253:0    0  6,2G  0 lvm   /
  └─cl-swap         253:1    0  820M  0 lvm   [SWAP]
sdb                   8:16   0    2G  0 disk
└─cl-newvar         253:2    0    1G  0 lvm
  └─cl-newvar_crypt 253:3    0 1022M  0 crypt /media/newvar_crypt
sr0                  11:0    1 1024M  0 rom
```

Reiniciar

```
reboot
```