

Si necesitamos conectar una unidad USB a una máquina virtual de Proxmox y no disponemos de un pendrive físico, debemos saber que es posible emularlo mediante algunos trucos de terminal. El truco es crear un archivo .img, que será nuestro pendrive virtual, respetando los siguientes pasos:

1. Creamos un archivo de imagen que será nuestro pendrive de 4GB:
`qemu-img create -f raw ~/pendrive.img 4G`
2. Comprobamos que la imagen se haya creado correctamente en formato raw:
`qemu-img info ~/pendrive.img`
3. Miramos cual es el primer dispositivo de loopback libre:
`sudo losetup -f`
4. Asociamos el archivo de imagen al primer dispositivo de loopback libre (en este caso el 1):
`sudo losetup /dev/loop1 ~/pendrive.img`
5. Formateamos el dispositivo virtual con NTFS:
`sudo mkfs.ntfs /dev/loop1`
6. Para montar la unidad virtual y empezar a meterle archivos podemos hacer:

```
mkdir ~/PenVirtual/  
sudo mount /dev/loop1 ~/PenVirtual/
```

7. Después de meterle algunos archivos, lo desmontamos con:
`sudo umount ~/PenVirtual/`
8. Finalmente liberamos el dispositivo de loopback, con:
`sudo losetup -d /dev/loop1`

Entonces, una vez seguidos los pasos de arriba y liberado el dispositivo de loopback, podemos agregar la imagen/pendrive a una máquina virtual de Proxmox ya encendida usando la consola de **qm monitor**, en Proxmox, de la siguiente manera:

```
qm monitor NroIDDeLaMV
```

Por ejemplo:

```
qm monitor 100
```

Se abrirá entonces la consola donde, para insertarle la imagen en caliente, ejecutaremos:

```
drive_add 0 id=PenEnImagen,if=none,format=raw,file=/home/nico/pendrive.img  
device_add usb-storage,drive=PenEnImagen
```

Trabajaremos con el «pendrive virtual» normalmente y, cuando ya no queramos utilizarlo, procederemos a desmontarlo y expulsarlo de forma segura de la MV, mediante su entorno gráfico (o mediante el método que prefiramos). Luego, con el mismo **qm monitor**, sacaremos el pendrive virtual de la MV de la siguiente manera:

```
drive_del PenEnImagen
```

NOTA: Si queremos que la imagen del pseudo-pendrive se agregue en modo lectura porque, por ejemplo, queremos aplicar alguna técnica forense informática sobre ella, podemos agregarla a la máquina virtual de la siguiente manera:

```
drive_add 0 id=PenEnImagen,if=none,format=raw,file=/home/nico/pendrive.img,read-only=on  
device_add usb-storage,drive=PenEnImagen
```

EXPLICACIÓN DE LOS PARÁMETROS

- **0:** Hace referencia al bus controlador donde se añadirá el drive. Normalmente 0 es el primer bus del sistema.

- **id=stick**: El ID del dispositivo, que podemos utilizar luego para referenciar o eliminar el disco.
- **if=none**: Para indicar que el disco se añadirá sin una interfaz concreta que luego asociaremos al dispositivo USB.
- **format=raw**: Indica que el formato del archivo de imagen es RAW.
- **file=/home/nico/pendrive.img**: Especifica la ruta al archivo de imagen que estás usando como disco virtual.

EXTRA 1: El método de agregar la imagen que hemos utilizado arriba sería el equivalente a meter el siguiente texto en el archivo .conf de configuración de la máquina virtual:

```
args: -drive id=PenEnImagen,if=none,format=raw,file=/home/nico/pendrive.img -device usb-storage,drive=PenEnEimagen
```

...con la diferencia de que, de la forma que lo hemos hecho, sería el equivalente a conectar el pendrive en caliente a un ordenador.

Metiendo esa línea en el **.conf**, también podemos tener la ventaja de indicarle a la máquina virtual que inicie siempre desde el pendrive virtual (útil para UNRAID o similares). Lo podemos hacer agregando el texto «,bootindex=1» (sin las comillas) al final de la línea, quedando esta así:

```
args: -drive id=PenEnImagen,if=none,format=raw,file=/home/nico/pendrive.img -device usb-storage,drive=PenEnEimagen,bootindex=1
```

EXTRA 2: Hay otra forma de agregar un archivo .img a una máquina virtual de PVE **importándolo como disco**.

