dd

- dd → comando para flujos de datos
  - **if**: archivo de origen
  - **of**: archivo de destino
  - **ibs**: número bytes leídos cada vez
  - **obs**: número bytes leídos cada vez
  - **bs**: número de bytes leídos y escritos cada vez
  - **cbs**: número de bytes a convertir cada vez
  - **skips**: número de bloques a saltar en **if** antes de empezar a copiar
  - **seek**: número de bloques a saltar en **of** antes de empezar a copiar
  - **count**: número de bloques a copiar de **if** en lugar del archivo completo

ejemplos

MBR

> **Advertencia**
> Restaurar el MBR con una tabla de particiones diferente hará que sus datos sean ilegibles y casi imposible de recuperar.

- copia del MBR

```
dd if=/dev/sda of=/boot/mbr-backup bs=512 count=1
```

- restauración del MBR

```
dd if=/root/mbr-backup of=/dev/sda bs=512 count=1
```

> **Advertencia**
> Este comando inutilizará el sistema, impidiento su arraque. Habrá que entrar en modo rescue y reinstalar el GRUB (si lo tienes)

-
```
dd if=/dev/zero of=/dev/sda bs=446 count=1
```

redimensionando

1. copiar las particiones de la tarjeta de origen

```
dd bs=1m status=progress if=/dev/sdXX of=./file1.img
```

2. redimensionar la partición de datos con:

```
sudo e2fsck ./file1.img
sudo resize2fs ./file1.img 14G
```

3. crear con `fdisk` las particiones en la tarjeta destino
4. copiar con dd
5. montar la partición de datos por si hubiese que cambiar algún dato en **/etc/fstab**

/via: https://superuser.com/questions/1525980/how-to-copy-a-file-system-to-a-smaller-device

reset dispositivo

- llena el disco de datos aleatorios.

```
dd if=/dev/urandom of=/dev/sda bs=4k
```

- llena el disco de 0.

```
dd if=/dev/zero of=/dev/sda bs=4k
```

- verifica que el disco esté a 0

```
dd if=/dev/sda | hexdump -C | grep [^00] a cero
```

- Borra GPT del disco. Como GPT escribe los datos al principio y al final del disco, después de borrar desde el principio, tenemos que encontrar el número de sectores (el segundo comando), y entonces borrar los 20 últimos sectores:

```
dd if=/dev/zero of=/dev/sda bs=512 count=2
fdisk -s /dev/sda # averiguar número de sectores
dd if=/dev/zero of=/dev/sda seek=(number_of_sectors - 20) bs=1k
```

ISO

- https://www.cyberciti.biz/tips/linux-creating-cd-rom-iso-image.html

```
blocks=$(isosize -d 2048 /dev/sr0)
sudo dd if=/dev/sr0 of=/tmp/output.iso bs=2048 count=$blocks
status=progress
sudo dd if=/path/to/my/iso.img of=/dev/sdd bs=4096 conv=noerror
```

- graba disco a imagen ISO

```
dd if=/dev/disk2s1 of=/Users/admin/imagen.iso
```

- graba una imagen ISO a disco (USB)

```
dd bs=4M if=image.iso of=/dev/sdxx
```

copias

- Crea una imagen gzip de la segunda partición del segundo disco

```
dd if=/dev/sdb2 ibs=4096 | gzip > partition.image.gz conv=noerror
```

progress

- muestra estadísticas de transferencia (solo GNU Coreutils 8.24+)

```
sudo dd status=progress if=... of=...
```

- usar utilidad PV para progreso (y si le pasas el «size», hace cálculo de restante)

```
sudo dd if=... | pv <-s size> | sudo dd of=...
```

- http://askubuntu.com/questions/215505/how-do-you-monitor-the-progress-of-dd

From:
https://matebcn.eu/ - **miguel angel torres egea**

Permanent link:
https://matebcn.eu/linux:bash:dd

Last update: **15/07/2025 19:44**