lsblk Block device inspection

START HERE

`lsblk`
List disks and partitions

`lsblk -f`
Add filesystem, label, UUID

`lsblk -o NAME,SIZE,TYPE,MOUNTPOINTS`
Pick your columns

`lsblk -p`
Show full /dev paths

`sudo blkid`
Show UUIDs and labels

`findmnt`
Tree of mounts

`findmnt <path>`
Which device holds this?

`ls -l /dev/disk/by-uuid`
Stable device names

Disk sda, nvme0n1 → `partition table` → **Partition** sda1, nvme0n1p1 → `mkfs` → **Filesystem** UUID, LABEL → `mount` → **Mount point** /, /mnt/data

CONCEPTS

lsblk, findmnt and blkid only read. The tools that change disks (wipefs without -n, mkfs, fdisk) are separate: check the device name first.

Kernel names like sda change between boots. Refer to disks by UUID, LABEL or /dev/disk/by-* in fstab and scripts.

lsblk reads sysfs and udev, so it works without root. blkid without root shows only what its cache already knows; use sudo for a fresh probe.

TYPE tells what a row is: disk, part, lvm, crypt, raid1, loop, rom. Children are shown beneath the device they sit on.

LIST DEVICES

`lsblk`
Tree of block devices

`lsblk -f`
Filesystem type, label, UUID, usage

`lsblk -a`
Include empty and hidden devices

`lsblk -d`
Disks only, no partitions

`lsblk -l`
Flat list instead of tree

`lsblk -p`
Print full /dev/... paths

`lsblk <disk>`
Show only this disk and its children

`lsblk -s <part>`
Show parents instead of children

`lsblk -e7`
Hide loop devices (major 7)

`lsblk -i`
ASCII tree, safe for pasting

`lsblk -t`
Topology: sector sizes, scheduler

`lsblk -m`
Owner, group and mode

`lsblk -D`
Discard (TRIM) support

`lsblk -S`
SCSI devices: HCTL, vendor, transport

`lsblk -N`
NVMe devices

CHOOSE COLUMNS

`lsblk -o NAME,SIZE,TYPE,MOUNTPOINTS`
Pick output columns

`lsblk -o +UUID,LABEL`
Add columns to the default set

`lsblk -O`
Print every column

`lsblk -H`
List available columns

`lsblk -x SIZE`
Sort by a column

`lsblk -b`
Sizes in bytes

`lsblk -n`
No header line

`lsblk -o NAME,SERIAL,MODEL,TRAN`
Identify a physical disk

`lsblk -o NAME,ROTA,HOTPLUG,RM`
SSD vs HDD, removable flags

`lsblk -o NAME,PARTUUID,PARTLABEL`
Partition table identifiers

OUTPUT FOR SCRIPTS

`lsblk -J`
JSON output

`lsblk -P`
KEY="value" pairs, one line per device

`lsblk -r`
Raw output, no tree characters

`lsblk -dno NAME,SIZE`
Bare disk list, no header

`lsblk -Q 'TYPE=="part"'`
Filter rows by expression

`lsblk -o NAME -n -r <disk>`
Names of a disk and its partitions

`lsblk -y`
Column names usable as shell variables

BLKID

`sudo blkid`
UUID, LABEL and TYPE of every device

`sudo blkid <dev>`
Probe one device

`sudo blkid -s UUID -o value <dev>`
Print just the UUID

`sudo blkid -o export <dev>`
KEY=value lines, sourceable

`sudo blkid -o list`
Table of devices and mount state

BLKID (cont.)

`blkid -U <uuid>`
Device name for a UUID

`blkid -L <label>`
Device name for a label

`sudo blkid -t TYPE=ext4`
Find devices by token

`sudo blkid -p <dev>`
Low-level probe, bypasses cache

FINDMNT

`findmnt`
Mounted filesystems as a tree

`findmnt <path>`
Mount that serves this mount point or device

`findmnt -T <path>`
Mount that contains this path

`findmnt -no SOURCE <path>`
Just the device behind a path

`findmnt -S UUID=<uuid>`
Find a mount by device tag

`findmnt -t ext4,xfs`
Limit to filesystem types

`findmnt --real`
Hide pseudo filesystems (proc, sysfs)

`findmnt -l`
Flat list instead of tree

`findmnt -o TARGET,SOURCE,FSTYPE,OPTIONS`
Choose columns

`findmnt -D`
Output like df

`findmnt -J`
JSON output

`findmnt -s`
Read /etc/fstab instead of live mounts

`findmnt --verify`
Check fstab for mistakes

`findmnt -R <path>`
Include submounts

`findmnt -p`
Watch for mount changes

STABLE NAMES

`ls -l /dev/disk/by-uuid`
Filesystem UUID to device

`ls -l /dev/disk/by-label`
Filesystem label to device

`ls -l /dev/disk/by-id`
Model and serial based names

`ls -l /dev/disk/by-path`
Names from the bus location

`ls -l /dev/disk/by-partuuid`
Partition table UUID to partition

STABLE NAMES (cont.)

`readlink -f /dev/disk/by-uuid/<uuid>`
Resolve a link to sdX

by-id and by-path links exist only if udev made them. Disks without a serial may be missing from by-id.

WIPE, RESCAN, UDEV

`sudo wipefs <dev>`
List signatures, erase nothing

`sudo wipefs -n -a <dev>`
Dry run of a full wipe

`sudo wipefs -a <dev>`
Erase all signatures (destructive)

`sudo wipefs -b -a <dev>`
Wipe and back up the signatures

`sudo partprobe <disk>`
Make the kernel re-read the table

`udevadm info <dev>`
Show udev data and symlinks

`udevadm info -q property <dev>`
Only the udev properties

`udevadm info -a <dev>`
Attributes up the device chain

`sudo udevadm settle`
Wait for the udev queue to drain

wipefs -a destroys the signatures that tools use to recognise the disk. Always run it with -n first and read the list.

RECIPES

Find the USB stick safely

```
lsblk -o NAME,SIZE,MODEL,TRAN,
HOTPLUG,MOUNTPOINTS
# plug it in, run again,
compare
lsblk -p -d -o
NAME,SIZE,MODEL /dev/sdb
```

fstab entry from a UUID

```
uuid=$(sudo blkid -s UUID -o
value /dev/sdb1)
echo "UUID=$uuid /mnt/data ext4
defaults 0 2"
findmnt --verify
```

What is using this path?

```
findmnt -T /var/lib/docker
lsblk -s $(findmnt -no SOURCE -
T /var/lib/docker)
df -h /var/lib/docker
```

Disk names and sizes as JSON

```
lsblk -J -b -d -o
NAME,SIZE,TYPE |
  jq -r '.blockdevices[] |
"\(.name) \(.size)"'
```