nvme-cli NVMe drive management

CONCEPTS

Use the character device (/dev/nvme0) for controller commands and the block device (/dev/nvme0n1) for namespace ones. The digits are not linked across the two.

Subsystem nvme-subsys0
- **Controller** /dev/nvme0
  - **Namespace** /dev/nvme0n1

nvme-cli 3.x groups commands (nvme id ctrl, nvme log smart, nvme ns create). The old dashed names still work as deprecated aliases that warn on stderr.

format, sanitize and ns delete destroy data and cannot be undone. Check the device path against nvme list first, twice.

DISCOVER

`nvme list`
List NVMe devices with size and model

`nvme list -o json`
List devices as JSON

`nvme list-subsys`
Show subsystems, controllers and paths

`nvme id ctrl /dev/nvme0`
Show controller model, serial, firmware

`nvme id ns /dev/nvme0n1`
Show namespace size and LBA formats

`nvme --version`
Show the nvme-cli version

HEALTH AND LOGS

`nvme log smart /dev/nvme0`
Show SMART health, wear, temperature

`nvme log smart /dev/nvme0 -o json`
SMART log as JSON for scripts

HEALTH AND LOGS (cont.)

`nvme log error /dev/nvme0`
Show the error log

Key SMART fields

Field	Meaning
`critical_warning`	Non-zero means act now
`percentage_used`	Wear estimate, may pass 100
`available_spare`	Spare blocks left (%)
`media_errors`	Data integrity errors
`unsafe_shutdowns`	Power loss without flush
`data_units_written`	Units of 512,000 bytes

SELF-TEST

`nvme device-self-test /dev/nvme0 -s 1`
Start a short self-test

`nvme device-self-test /dev/nvme0 -s 2`
Start an extended self-test

`nvme log self-test /dev/nvme0`
Show self-test results

FORMAT AND ERASE

`nvme format /dev/nvme0n1 --lbaf=<n>`
ERASES: reformat with LBA format n

`nvme sanitize /dev/nvme0 -a start-block-erase`
ERASES: block erase the whole drive

FIRMWARE

`nvme fw download /dev/nvme0 --fw=<file>`
Upload a firmware image

`nvme fw commit /dev/nvme0 -s <slot> -a 1`
Store image in slot, activate at reset

OPTIONS AND HELP

`nvme <command> -o json`
Machine-readable output