Rocky Enterprise Linux 9.2 Manual Pages on command 'nvme-admin-passthru.1'

$ man nvme-admin-passthru.1

NVME-ADMIN-PASSTHR(1) NVMe Manual NVME-ADMIN-PASSTHR(1)

NAME

nvme-admin-passthru - Submit an arbitrary admin command, return results

SYNOPSIS

nvme-admin-passthru <device> [--opcode=<opcode> | -o <opcode>]

[--flags=<flags> | -f <flags>] [-rsvd=<rsvd> | -R <rsvd>]

[--namespace-id=<nsid>] [--cdw2=<cdw2>] [--cdw3=<cdw3>]

[--cdw10=<cdw10>] [--cdw11=<cdw11>] [--cdw12=<cdw12>]

[--cdw13=<cdw13>] [--cdw14=<cdw14>] [--cdw15=<cdw15>]

[--data-len=<data-len> | -l <data-len>]

[--metadata-len=<len> | -m <len>]

[--input-file=<file> | -i <file>]

[--read | -r] [--write | -w]

[--timeout=<to> | -t <to>]

[--show-command | -s]

[--dry-run | -d]

[--raw-binary | -b]

[--prefill=<prefill> | -p <prefill>]

[--latency | -T]

DESCRIPTION

Submits an arbitrary NVMe admin command and returns the applicable results. This may be the simply the commands result and status, or may also include a buffer if the command returns one. This command does no interpretation of the opcodes or options.

The <device> parameter is mandatory and may be either the NVMe character device (ex: /dev/nvme0), or a namespace block device (ex: /dev/nvme0n1).

On success, the returned structure (if applicable) may be returned in one of several ways depending on the option flags; the structure may printed by the program as a hex dump, or may be returned as a raw buffer printed to stdout for another program to parse.

OPTIONS

-o <opcode>, --opcode=<opcode>

The NVMe opcode to send to the device in the command

-f <flags>, --flags=<flags>

The NVMe command flags to send to the device in the command

-R <rsvd>, --rsvd=<rsvd>

The value for the reserved field in the command.

-n <nsid>, --namespace-id=<nsid>

The value for the ns-id in the command.

--cdw[2-3,10-15]=<cdw>

Specifies the command dword value for that specified entry in the command

-r, --read, -w, --write

Used for the data-direction for the command and required for commands sending/receiving data. Don?t use both read and write at the same time.

-i <file>, --input-file=<file>

If the command is a data-out (write) command, use this file to fill the buffer sent to the device. If no file is given, assumed to use STDIN.

-l <data-len>, --data-len=<data-len>

The data length for the buffer used for this command.

-m <data-len>, --metadata-len=<data-len>

The metadata length for the buffer used for this command.

-s, --show-cmd

Print out the command to be sent.

-d, --dry-run

Do not actually send the command. If want to use --dry-run option, --show-cmd option must be set. Otherwise --dry-run option will be ignored.

-b, --raw-binary

Print the raw returned buffer to stdout if the command returns a structure.

-p, --prefill

Prefill the buffer with a predetermined byte value. Defaults to 0. This may be useful if the data you are writing is shorter than the required buffer, and you need to pad it with a known value. It may also be useful if you need to confirm if a device is overwriting a buffer for a data-in command.

-T, --latency

Print out the latency the IOCTL took (in us).

EXAMPLES

? The following will run the admin command with opcode=6 and cdw10=1, which corresponds to an identify controller command. This example requires the data-len param be 4096, which is the size of the returned structure. The -r option is used because it is a data-in command

```
# nvme admin-passthru /dev/nvme0 --opcode=06 --data-len=4096 --cdw10=1 -r
```

? Or if you want to save that structure to a file:

```
# nvme admin-passthru /dev/nvme0 --opcode=06 --data-len=4096 --cdw10=1 -r -b > id_ns.raw
```

NVME

Part of the nvme-user suite