# *Red Hat Enterprise Linux Release 9.2 Manual Pages on 'nvme-connect.1' command*

***$ man nvme-connect.1***

```
NVME-CONNECT(1)                NVMe Manual              NVME-CONNECT(1)
```

NAME

nvme-connect - Connect to a Fabrics controller.

SYNOPSIS

```
nvme connect
        [--transport=<trtype>     | -t <trtype>]
        [--nqn=<subnqn>           | -n <subnqn>]
        [--traddr=<traddr>        | -a <traddr>]
        [--trsvcid=<trsvcid>      | -s <trsvcid>]
        [--host-traddr=<traddr>   | -w <traddr>]
        [--host-iface=<iface>     | -f <iface>]
        [--hostnqn=<hostnqn>      | -q <hostnqn>]
        [--hostid=<hostid>        | -I <hostid>]
        [--config-file=<cfg>      | -J <cfg> ]
        [--dhchap-secret=<secret> | -S <secret>]
        [--dhchap-ctrl-secret=<secret> | -C <secret>]
        [--nr-io-queues=<#>       | -i <#>]
        [--nr-write-queues=<#>    | -W <#>]
        [--nr-poll-queues=<#>     | -P <#>]
        [--queue-size=<#>         | -Q <#>]
        [--keep-alive-tmo=<#>     | -k <#>]
        [--reconnect-delay=<#>    | -c <#>]
        [--ctrl-loss-tmo=<#>      | -l <#>]
        [--duplicate-connect      | -D]
        [--disable-sqflow         | -d]
        [--hdr-digest             | -g]
        [--data-digest            | -G]
        [--dump-config            | -O]
        [--output-format=<fmt>    | -o <fmt>]
```

## DESCRIPTION

Create a transport connection to a remote system (specified by --traddr and --trsvcid) and create a NVMe over Fabrics controller for the NVMe subsystem specified by the --nqn option.

## OPTIONS

-t <trtype>, --transport=<trtype>

This field specifies the network fabric being used for a NVMe-over-Fabrics network. Current string values include:

```
?????????????????????????????????????
?Value ? Definition                 ?
?????????????????????????????????????
?rdma  ? The network fabric is an   ?
?      ? rdma network (RoCE, iWARP, ?
?      ? Infiniband, basic rdma,    ?
?      ? etc)                       ?
?????????????????????????????????????
?fc    ? WIP The network fabric is  ?
?      ? a Fibre Channel network.   ?
?????????????????????????????????????
?tcp   ? The network fabric is a    ?
?      ? TCP/IP network.            ?
?????????????????????????????????????
```

```
?loop  ? Connect to a NVMe over     ?
?      ? Fabrics target on the      ?
?      ? local host                 ?
????????????????????????????????????????
```

-n <subnqn>, --nqn <subnqn>

This field specifies the name for the NVMe subsystem to connect to.

-a <traddr>, --traddr=<traddr>

This field specifies the network address of the Controller. For transports using IP addressing (e.g. rdma) this should be an IP-based address (ex. IPv4).

-s <trsvcid>, --trsvcid=<trsvcid>

This field specifies the transport service id. For transports using IP addressing (e.g. rdma) this field is the port number. By default, the IP port number for the RDMA transport is 4420.

-w <traddr>, --host-traddr=<traddr>

This field specifies the network address used on the host to connect to the Controller. For TCP, this sets the source address on the socket.

-f <iface>, --host-iface=<iface>

This field specifies the network interface used on the host to connect to the Controller (e.g. IP eth1, enp2s0, enx78e7d1ea46da). This forces the connection to be made on a specific interface instead of letting the system decide.

-q <hostnqn>, --hostnqn=<hostnqn>

Overrides the default Host NQN that identifies the NVMe Host. If this option is not specified, the default is read from /etc/nvme/hostnqn first. If that does not exist, the autogenerated

NQN value from the NVMe Host kernel module is used next. The Host NQN uniquely identifies the NVMe Host.

-I <hostid>, --hostid=<hostid>

UUID(Universally Unique Identifier) to be discovered which should be formatted.

-J <cfg>, --config-file=<cfg>

Use the specified JSON configuration file instead of the default /etc/nvme/config.json file or none to not read in an existing configuration file. The JSON configuration file format is documented in https://github.com/linux-nvme/libnvme/doc/config-schema.json

-S <secret>, --dhchap-secret=<secret>

NVMe In-band authentication secret; needs to be in ASCII format as specified in NVMe 2.0 section 8.13.5.8 Secret representation. If this option is not specified, the default is read from /etc/nvme/hostkey. If that does not exist no in-band authentication is attempted.

-C <secret>, --dhchap-ctrl-secret=<secret>

NVMe In-band authentication controller secret for bi-directional authentication; needs to be in ASCII format as specified in NVMe 2.0 section 8.13.5.8 Secret representation. If not present bi-directional authentication is not attempted.

-i <#>, --nr-io-queues=<#>

Overrides the default number of I/O queues create by the driver.

-W <#>, --nr-write-queues=<#>

Adds additional queues that will be used for write I/O.

-P <#>, --nr-poll-queues=<#>

Adds additional queues that will be used for polling latency sensitive I/O.

-Q <#>, --queue-size=<#>

Overrides the default number of elements in the I/O queues created by the driver.

-k <#>, --keep-alive-tmo=<#>

Overrides the default keep alive timeout (in seconds).

-c <#>, --reconnect-delay=<#>

Overrides the default delay (in seconds) before reconnect is attempted after a connect loss.

-l <#>, --ctrl-loss-tmo=<#>

Overrides the default controller loss timeout period (in seconds).

-D, --duplicate-connect

Allows duplicated connections between same transport host and subsystem port.

-d, --disable-sqflow

Disables SQ flow control to omit head doorbell update for submission queues when sending nvme completions.

-g, --hdr-digest

Generates/verifies header digest (TCP).

-G, --data-digest

Generates/verifies data digest (TCP).

-O, --dump-config

Print out resulting JSON configuration file to stdout.

-o <format>, --output-format=<format>

Set the reporting format to normal or json. Only one output format can be used at a time. When this option is specified, the device associated with the connection will be printed. Nothing is printed otherwise.

## EXAMPLES

? Connect to a subsystem named nqn.2014-08.com.example:nvme:nvm-subsystem-sn-d78432 on the IP4 address 192.168.1.3. Port 4420 is used by default:

```
# nvme connect --transport=rdma --traddr=192.168.1.3 \
--nqn=nqn.2014-08.com.example:nvme:nvm-subsystem-sn-d78432
```

## SEE ALSO

nvme-discover(1) nvme-connect-all(1)

## AUTHORS

This was co-written by Jay Freyensee[1] and Christoph Hellwig[2]

## NVME

Part of the nvme-user suite

## NOTES

1. Jay Freyensee

   mailto:james.p.freyensee@intel.com

2. Christoph Hellwig

   mailto:hch@lst.de