Manage NVMe subsystems

ONTAP 9.12.1 REST API reference

NetApp
May 08, 2024

Table of Contents

Manage NVMe subsystems

Protocols NVMe subsystems endpoint overview

Overview

An NVMe subsystem maintains configuration state and namespace access control for a set of NVMe-connected hosts.

The NVMe subsystem REST API allows you to create, update, delete, and discover NVMe subsystems as well as add and remove NVMe hosts that can access the subsystem and associated namespaces.

Examples

Creating an NVMe subsystem

```
# The API:
POST /api/protocols/nvme/subsystems

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/nvme/subsystems' -H 'Accept:
application/json' -d '{ "svm": { "name": "svm1" }, "name": "subsystem1",
"os_type": "linux" }'
```

Creating an NVMe subsystem with multiple NVMe subsystem hosts

```
# The API:
POST /api/protocols/nvme/subsystems

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/nvme/subsystems' -H 'Accept:
application/json' -d '{ "svm": { "name": "svm1" }, "name": "subsystem2",
"os_type": "vmware", "hosts": [ { "nqn": "nqn.1992-01.example.com:host1"
}, { "nqn": "nqn.1992-01.example.com:host2" } ] }'
```

Retrieving all NVMe subsystems

```
# The API:
GET /api/protocols/nvme/subsystems

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/nvme/subsystems' -H 'Accept:
application/json'

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "a009a9e7-4081-b576-7575-ada21efcaf16",
      "name": "svm1",
    },
    "uuid": "acde901a-a379-4a91-9ea6-1b728ed6696f",
    "name": "subsystem1",
  },
  {
    "svm": {
      "uuid": "a009a9e7-4081-b576-7575-ada21efcaf16",
      "name": "svm1",
    },
    "uuid": "bcde901a-a379-4a91-9ea6-1b728ed6696f",
    "name": "subsystem2",
  }
],
"num_records": 2,
}
```

Retrieving all NVMe subsystems with OS type *linux*

Note that the `os_type` query parameter is used to perform the query.

```
# The API:
GET /api/protocols/nvme/subsystems

# The call:
curl -X GET 'https://<mgmt-
ip>/api/protocols/nvme/subsystems?os_type=linux' -H 'Accept:
application/json'

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "a009a9e7-4081-b576-7575-ada21efcaf16",
      "name": "svm1",
    },
    "uuid": "acde901a-a379-4a91-9ea6-1b728ed6696f",
    "name": "subsystem1",
    "os_type": "linux",
  }
],
"num_records": 1,
}
```

Retrieving a specific NVMe subsystem

```
# The API:
GET /api/protocols/nvme/subsystems/{uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/nvme/subsystems/acde901a-
a379-4a91-9ea6-1b728ed6696f' -H 'Accept: application/json'

# The response:
{
"svm": {
  "uuid": "a009a9e7-4081-b576-7575-ada21efcaf16",
  "name": "svm1",
},
"uuid": "acde901a-a379-4a91-9ea6-1b728ed6696f",
"name": "subsystem1",
"os_type": "linux",
"target_nqn": "nqn.1992-
08.com.netapp:sn.d04594ef915b4c73b642169e72e4c0b1:subsystem.subsystem1",
"serial_number": "wtJNKNKD-uPLAAAAAAAD",
"io_queue": {
  "default": {
    "count": 4,
    "depth": 32
  }
}
}
```

Retrieving the NVMe namespaces mapped to a specific NVMe subsystem

Note that the `fields` query parameter is used to specify the desired properties.

```
# The API:
GET /api/protocols/nvme/subsystems/{uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/nvme/subsystems/acde901a-
a379-4a91-9ea6-1b728ed6696f?fields=subsystem_maps' -H 'Accept:
application/json'

# The response:
{
"svm": {
  "uuid": "a009a9e7-4081-b576-7575-ada21efcaf16",
  "name": "svm1",
},
"uuid": "acde901a-a379-4a91-9ea6-1b728ed6696f",
"name": "subsystem1",
"subsystem_maps": [
  {
    "anagrpid": "00000001h",
    "namespace": {
      "uuid": "eeaaca23-128d-4a7d-be4a-dc9106705799",
      "name": "/vol/vol1/namespace1"
    },
    "nsid": "00000001h"
  },
  {
    "anagrpid": "00000002h",
    "namespace": {
      "uuid": "feaaca23-83a0-4a7d-beda-dc9106705799",
      "name": "/vol/vol1/namespace2"
    },
    "nsid": "00000002h"
  }
]
}
```

Adding a comment about an NVMe subsystem

```
# The API:
PATCH /api/protocols/nvme/subsystems/{uuid}

# The call:
curl -X PATCH 'https://<mgmt-ip>/api/protocols/nvme/subsystems/acde901a-
a379-4a91-9ea6-1b728ed6696f' -H 'Accept: application/json' -d '{
"comment": "A brief comment about the subsystem" }'
```

Deleting an NVMe subsystem

```
# The API:
DELETE /api/protocols/nvme/subsystems/{uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/nvme/subsystems/acde901a-
a379-4a91-9ea6-1b728ed6696f' -H 'Accept: application/json'
```

Deleting an NVMe subsystem with mapped NVMe namespaces

Normally, deleting an NVMe subsystem that has mapped NVMe namespaces is not allowed. The deletion can be forced using the `allow_delete_while_mapped` query parameter.

```
# The API:
DELETE /api/protocols/nvme/subsystems/{uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/nvme/subsystems/acde901a-
a379-4a91-9ea6-1b728ed6696f?allow_delete_while_mapped=true' -H 'Accept:
application/json'
```

Delete an NVMe subsystem with NVMe subsystem hosts

Normally, deleting an NVMe subsystem with NVMe subsystem hosts is disallowed. The deletion can be forced using the `allow_delete_with_hosts` query parameter.

```
# The API:
DELETE /api/protocols/nvme/subsystems/{uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/nvme/subsystems/acde901a-
a379-4a91-9ea6-1b728ed6696f?allow_delete_with_hosts=true' -H 'Accept:
application/json'
```

An NVMe Subsystem Host

An NVMe subsystem host is a network host provisioned to an NVMe subsystem to access namespaces mapped to that subsystem.

Examples

Adding an NVMe subsystem host to an NVMe subsystem

```
# The API:
POST /protocols/nvme/subsystems/{subsystem.uuid}/hosts

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/nvme/subsystems/acde901a-
a379-4a91-9ea6-1b728ed6696f/hosts' -H 'Accept: application/json' -d '{
"nqn": "nqn.1992-01.com.example:subsys1.host1" }'
```

Adding multiple NVMe subsystem hosts to an NVMe subsystem

```
# The API:
POST /protocols/nvme/subsystems/{subsystem.uuid}/hosts

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/nvme/subsystems/acde901a-
a379-4a91-9ea6-1b728ed6696f/hosts' -H 'Accept: application/json' -d '{
"records": [ { "nqn": "nqn.1992-01.com.example:subsys1.host2" }, { "nqn":
"nqn.1992-01.com.example:subsys1.host3" } ] }'
```

Retrieving all NVMe subsystem hosts for an NVMe subsystem

```
# The API:
GET /protocols/nvme/subsystems/{subsystem.uuid}/hosts

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/nvme/subsystems/acde901a-
a379-4a91-9ea6-1b728ed6696f/hosts' -H 'Accept: application/json'

# The response:
{
"records": [
  {
    "nqn": "nqn.1992-01.com.example:subsys1.host1",
  },
  {
    "nqn": "nqn.1992-01.com.example:subsys1.host2",
  },
  {
    "nqn": "nqn.1992-01.com.example:subsys1.host3",
  }
],
"num_records": 3,
}
```

Retrieving a specific NVMe subsystem host for an NVMe subsystem

```
# The API:
GET /protocols/nvme/subsystems/{subsystem.uuid}/hosts/{nqn}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/nvme/subsystems/acde901a-
a379-4a91-9ea6-1b728ed6696f/hosts/nqn.1992-01.com.example:subsys1.host1'
-H 'Accept: application/json'

# The response:
{
"subsystem": {
  "uuid": "acde901a-a379-4a91-9ea6-1b728ed6696f",
},
"nqn": "nqn.1992-01.com.example:subsys1.host1",
"dh_hmac_chap": {
  "node": "none"
},
"io_queue": {
  "count": 4,
  "depth": 32
},
}
```

Deleting an NVMe subsystem host from an NVMe subsystem

```
# The API:
DELETE /protocols/nvme/subsystems/{subsystem.uuid}/hosts/{nqn}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/nvme/subsystems/acde901a-
a379-4a91-9ea6-1b728ed6696f/hosts/nqn.1992-01.com.example:subsys1.host1'
-H 'Accept: application/json'
```

Retrieve NVMe subsystems

GET `/protocols/nvme/subsystems`

Introduced In: 9.6

Retrieves NVMe subsystems.

Related ONTAP commands

- `vserver nvme subsystem host show`
- `vserver nvme subsystem map show`
- `vserver nvme subsystem show`

Learn more

- DOC /protocols/nvme/subsystems

Parameters

Name	Type	In	Required	Description
comment	string	query	False	Filter by comment • maxLength: 255 • minLength: 0
serial_number	string	query	False	Filter by serial_number • maxLength: 20 • minLength: 20
target_nqn	string	query	False	Filter by target_nqn • maxLength: 223 • minLength: 1
svm.uuid	string	query	False	Filter by svm.uuid
svm.name	string	query	False	Filter by svm.name
io_queue.default.depth	integer	query	False	Filter by io_queue.default.depth • Max value: 128 • Min value: 16
io_queue.default.count	integer	query	False	Filter by io_queue.default.count • Max value: 15 • Min value: 1

Name	Type	In	Required	Description
name	string	query	False	Filter by name • maxLength: 96 • minLength: 1
hosts.nqn	string	query	False	Filter by hosts.nqn
hosts.dh_hmac_chap.mode	string	query	False	Filter by hosts.dh_hmac_chap.mode • Introduced in: 9.12
hosts.dh_hmac_chap.group_size	string	query	False	Filter by hosts.dh_hmac_chap.group_size • Introduced in: 9.12
hosts.dh_hmac_chap.hash_function	string	query	False	Filter by hosts.dh_hmac_chap.hash_function • Introduced in: 9.12
os_type	string	query	False	Filter by os_type
delete_on_unmap	boolean	query	False	Filter by delete_on_unmap • Introduced in: 9.7
subsystem_maps.nsid	string	query	False	Filter by subsystem_maps.nsid
subsystem_maps.anagrpid	string	query	False	Filter by subsystem_maps.anagrpid
subsystem_maps.namespace.uuid	string	query	False	Filter by subsystem_maps.namespace.uuid

Name	Type	In	Required	Description
subsystem_maps.namespace.name	string	query	False	Filter by subsystem_maps.namespace.name
uuid	string	query	False	Filter by uuid
vendor_uuids	string	query	False	Filter by vendor_uuids • Introduced in: 9.9
fields	array[string]	query	False	Specify the fields to return.
max_records	integer	query	False	Limit the number of records returned.
return_records	boolean	query	False	The default is true for GET calls. When set to false, only the number of records is returned. • Default value: 1
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When iterating over a collection, the default is 15 seconds. ONTAP returns earlier if either max records or the end of the collection is reached. • Max value: 120 • Min value: 0 • Default value: 1
order_by	array[string]	query	False	Order results by specified fields and optional [asc

Response

```
Status: 200, Ok
```

Name	Type	Description
_links	_links	
num_records	integer	The number of records in the response.
records	array[nvme_subsystem]	

Example response


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "comment": "string",
    "hosts": {
      "dh_hmac_chap": {
        "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
        "group_size": "none",
        "hash_function": "sha_256",
        "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
        "mode": "bidirectional"
      },
      "nqn": "nqn.1992-01.example.com:string"
    },
    "io_queue": {
      "default": {
        "count": 4,
        "depth": 16
      }
    },
    "name": "subsystem1",
    "os_type": "aix",
    "serial_number": "wCVsgFMiuMhVAAAAAAAB",
    "subsystem_maps": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
```

```
      "anagrpid": "00103050h",
      "namespace": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "/vol/vol1/namespace1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "nsid": "00000001h"
    },
    "svm": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "svm1",
      "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
    },
    "target_nqn": "nqn.1992-01.example.com:string",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
    "vendor_uuids": {
    }
  }
}
```


Error

```
Status: Default, Error
```

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
next	href	
self	href	

_links

Name	Type	Description
self	href	

dh_hmac_chap

A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.

Name	Type	Description
controller_secret_key	string	The controller secret for NVMe in-band authentication. The value of this property is used by the NVMe host to authenticate the NVMe controller while establishing a connection. If unset, the controller is not authenticated. When supplied, the property `host_secret_key` must also be supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a controller secret has been set for the host, but the controller secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.

Name	Type	Description
group_size	string	The Diffie-Hellman group size for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `2048_bit`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.
hash_function	string	The hash function for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `sha_256`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.
host_secret_key	string	The host secret for NVMe in-band authentication. The value of this property is used by the NVMe controller to authenticate the NVMe host while establishing a connection. If unset, no authentication is performed by the host or controller. This property must be supplied if any other NVMe in-band authentication properties are supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a host secret has been set for the host, but the host secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.

Name	Type	Description
mode	string	The expected NVMe in-band authentication mode for the host. This property is an indication of which secrets are configured for the host. When set to: • none: The host has neither the host nor controller secret configured, and no authentication is performed. • unidirectional: The host has a host secret configured. The controller will authenticate the host. • bidirectional: The host has both a host and controller secret configured. The controller will authenticate the host and the host will authenticate the controller.

hosts

Name	Type	Description
dh_hmac_chap	dh_hmac_chap	A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.
nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target.

default

The default I/O queue parameters inherited by NVMe hosts in the NVMe subsystem.

Name	Type	Description
count	integer	The number of host I/O queue pairs.
depth	integer	The host I/O queue depth.

io_queue

The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.

Name	Type	Description
default	default	The default I/O queue parameters inherited by NVMe hosts in the NVMe subsystem.

namespace

An NVMe namespace mapped to the NVMe subsystem.

Name	Type	Description
_links	_links	
name	string	The name of the NVMe namespace.
uuid	string	The unique identifier of the NVMe namespace.

subsystem_maps

An NVMe namespace mapped to the NVMe subsystem.

Name	Type	Description
_links	_links	
anagrpid	string	The Asymmetric Namespace Access Group ID (ANAGRPID) of the NVMe namespace. The format for an ANAGRPIP is 8 hexadecimal digits (zero-filled) followed by a lower case "h".
namespace	namespace	An NVMe namespace mapped to the NVMe subsystem.
nsid	string	The NVMe namespace identifier. This is an identifier used by an NVMe controller to provide access to the NVMe namespace. The format for an NVMe namespace identifier is 8 hexadecimal digits (zero-filled) followed by a lower case "h".

svm

Name	Type	Description
_links	_links	
name	string	The name of the SVM.
uuid	string	The unique identifier of the SVM.

nvme_subsystem

An NVMe subsystem maintains configuration state and namespace access control for a set of NVMe-connected hosts.

Name	Type	Description
_links	_links	
comment	string	A configurable comment for the NVMe subsystem. Optional in POST and PATCH.
delete_on_unmap	boolean	An option that causes the subsystem to be deleted when the last subsystem map associated with it is deleted. Optional in POST and PATCH. This property defaults to *false* when the subsystem is created.
hosts	array[hosts]	The NVMe hosts configured for access to the NVMe subsystem. Optional in POST.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.
name	string	The name of the NVMe subsystem. Once created, an NVMe subsystem cannot be renamed. Required in POST.
os_type	string	The host operating system of the NVMe subsystem's hosts. Required in POST.
serial_number	string	The serial number of the NVMe subsystem.

Name	Type	Description
subsystem_maps	array[subsystem_maps]	The NVMe namespaces mapped to the NVMe subsystem. There is an added computational cost to retrieving property values for `subsystem_maps`. They are not populated for either a collection GET or an instance GET unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.
svm	svm	
target_nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target.
uuid	string	The unique identifier of the NVMe subsystem.
vendor_uuids	array[string]	Vendor-specific identifiers (UUIDs) optionally assigned to an NVMe subsystem when the subsystem is created. The identifiers are used to enable vendor-specific NVMe protocol features. The identifiers are provided by a host application vendor and shared with NetApp prior to a joint product release. Creating an NVMe subsystem with an unknown or non-specific identifier will have no effect on the NVMe subsystem. Refer to the ONTAP SAN Administration Guide for a list of the supported vendor-specific identifiers. After a subsystem is created, the vendor-specific identifiers cannot be changed or removed. Optional in POST. • Introduced in: 9.9 • readCreate: 1

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Create an NVMe subsystem

POST `/protocols/nvme/subsystems`

Introduced In: 9.6

Creates an NVMe subsystem.

Required properties

- `svm.uuid` or `svm.name` - Existing SVM in which to create the NVMe subsystem.
- `name` - Name for NVMe subsystem. Once created, an NVMe subsytem cannot be renamed.
- `os_type` - Operating system of the NVMe subsystem's hosts.

Related ONTAP commands

- `vserver nvme subsystem create`

Learn more

- DOC /protocols/nvme/subsystems

Parameters

Name	Type	In	Required	Description
return_records	boolean	query	False	The default is false. If set to true, the records are returned. • Default value:

Request Body

Name	Type	Description
_links	_links	
comment	string	A configurable comment for the NVMe subsystem. Optional in POST and PATCH.
delete_on_unmap	boolean	An option that causes the subsystem to be deleted when the last subsystem map associated with it is deleted. Optional in POST and PATCH. This property defaults to *false* when the subsystem is created.
hosts	array[hosts]	The NVMe hosts configured for access to the NVMe subsystem. Optional in POST.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.
name	string	The name of the NVMe subsystem. Once created, an NVMe subsystem cannot be renamed. Required in POST.
os_type	string	The host operating system of the NVMe subsystem's hosts. Required in POST.
serial_number	string	The serial number of the NVMe subsystem.

Name	Type	Description
subsystem_maps	array[subsystem_maps]	The NVMe namespaces mapped to the NVMe subsystem. There is an added computational cost to retrieving property values for `subsystem_maps`. They are not populated for either a collection GET or an instance GET unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.
svm	svm	
target_nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target.
uuid	string	The unique identifier of the NVMe subsystem.
vendor_uuids	array[string]	Vendor-specific identifiers (UUIDs) optionally assigned to an NVMe subsystem when the subsystem is created. The identifiers are used to enable vendor-specific NVMe protocol features. The identifiers are provided by a host application vendor and shared with NetApp prior to a joint product release. Creating an NVMe subsystem with an unknown or non-specific identifier will have no effect on the NVMe subsystem. Refer to the ONTAP SAN Administration Guide for a list of the supported vendor-specific identifiers. After a subsystem is created, the vendor-specific identifiers cannot be changed or removed. Optional in POST. • Introduced in: 9.9 • readCreate: 1

Example request


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "comment": "string",
  "hosts": {
    "dh_hmac_chap": {
      "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "group_size": "none",
      "hash_function": "sha_256",
      "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "mode": "bidirectional"
    },
    "nqn": "nqn.1992-01.example.com:string"
  },
  "io_queue": {
    "default": {
      "count": 4,
      "depth": 16
    }
  },
  "name": "subsystem1",
  "os_type": "aix",
  "serial_number": "wCVsgFMiuMhVAAAAAAAB",
  "subsystem_maps": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "anagrpid": "00103050h",
    "namespace": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "/vol/vol1/namespace1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
```

```
      "nsid": "00000001h"
    },
    "svm": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "svm1",
      "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
    },
    "target_nqn": "nqn.1992-01.example.com:string",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
    "vendor_uuids": {
    }
  }
```


Response

```
Status: 201, Created
```

Name	Type	Description
_links	_links	
num_records	integer	The number of records in the response.
records	array[nvme_subsystem]	

Example response


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "comment": "string",
    "hosts": {
      "dh_hmac_chap": {
        "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
        "group_size": "none",
        "hash_function": "sha_256",
        "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
        "mode": "bidirectional"
      },
      "nqn": "nqn.1992-01.example.com:string"
    },
    "io_queue": {
      "default": {
        "count": 4,
        "depth": 16
      }
    },
    "name": "subsystem1",
    "os_type": "aix",
    "serial_number": "wCVsgFMiuMhVAAAAAAAB",
    "subsystem_maps": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
```

```
      "anagrpid": "00103050h",
      "namespace": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "/vol/vol1/namespace1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "nsid": "00000001h"
    },
    "svm": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "svm1",
      "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
    },
    "target_nqn": "nqn.1992-01.example.com:string",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
    "vendor_uuids": {
    }
  }
}
```


Headers

Name	Description	Type
Location	Useful for tracking the resource location	string

Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
2621462	The supplied SVM does not exist.

Error Code	Description
2621706	The specified `svm.uuid` and `svm.name` do not refer to the same SVM.
2621707	The svm.uuid or svm.name must be provided.
72089635	Setting vendor-specific UUIDs on NVMe subsystems is not supported until the effective cluster version is 9.9 or later.
72089709	The NVMe subsystem name contains an invalid character.
72089711	An invalid vendor-specific UUID was specified.
72089712	A duplicate vendor-specific UUID was specific.
72089713	Too many vendor UUIDs were supplied.
72089771	The NQN is invalid. A non-empty qualifier is required after the prefix. An example of a valid NQN is *nqn.1992-01.com.example:string*.
72089772	The NQN is invalid. Add the prefix *'nqn'*. An example of a valid NQN is *nqn.1992-01.com.example:string*.
72089773	The NQN is invalid. The date field must be formatted *yyyy-mm*. An example of a valid NQN is *nqn.1992-01.com.example:string*.
72090003	A host to be added to an NVMe subsystem is missing the "nqn" property.
72090025	The NVMe subsystem already exists for the SVM.
72090029	The NVMe service does not exist.
72090030	A partial success occured while adding multiple NVMe subsystem hosts to an NVMe subsystem.
72090036	An NVMe subsystem host NQN was duplicated in the input.
72090042	The `dh_hmac_chap.host_secret_key` property is required when setting any other NVMe in-band authentication properties for a host.

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
self	href	

dh_hmac_chap

A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.

Name	Type	Description
controller_secret_key	string	The controller secret for NVMe in-band authentication. The value of this property is used by the NVMe host to authenticate the NVMe controller while establishing a connection. If unset, the controller is not authenticated. When supplied, the property `host_secret_key` must also be supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a controller secret has been set for the host, but the controller secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.
group_size	string	The Diffie-Hellman group size for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `2048_bit`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.

Name	Type	Description
hash_function	string	The hash function for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `sha_256`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.
host_secret_key	string	The host secret for NVMe in-band authentication. The value of this property is used by the NVMe controller to authenticate the NVMe host while establishing a connection. If unset, no authentication is performed by the host or controller. This property must be supplied if any other NVMe in-band authentication properties are supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a host secret has been set for the host, but the host secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.

Name	Type	Description
mode	string	The expected NVMe in-band authentication mode for the host. This property is an indication of which secrets are configured for the host. When set to: • none: The host has neither the host nor controller secret configured, and no authentication is performed. • unidirectional: The host has a host secret configured. The controller will authenticate the host. • bidirectional: The host has both a host and controller secret configured. The controller will authenticate the host and the host will authenticate the controller.

hosts

Name	Type	Description
dh_hmac_chap	dh_hmac_chap	A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.
nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target.

default

The default I/O queue parameters inherited by NVMe hosts in the NVMe subsystem.

Name	Type	Description
count	integer	The number of host I/O queue pairs.
depth	integer	The host I/O queue depth.

io_queue

The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.

Name	Type	Description
default	default	The default I/O queue parameters inherited by NVMe hosts in the NVMe subsystem.

namespace

An NVMe namespace mapped to the NVMe subsystem.

Name	Type	Description
_links	_links	
name	string	The name of the NVMe namespace.
uuid	string	The unique identifier of the NVMe namespace.

subsystem_maps

An NVMe namespace mapped to the NVMe subsystem.

Name	Type	Description
_links	_links	
anagrpid	string	The Asymmetric Namespace Access Group ID (ANAGRPID) of the NVMe namespace. The format for an ANAGRPIP is 8 hexadecimal digits (zero-filled) followed by a lower case "h".
namespace	namespace	An NVMe namespace mapped to the NVMe subsystem.
nsid	string	The NVMe namespace identifier. This is an identifier used by an NVMe controller to provide access to the NVMe namespace. The format for an NVMe namespace identifier is 8 hexadecimal digits (zero-filled) followed by a lower case "h".

svm

Name	Type	Description
_links	_links	
name	string	The name of the SVM.
uuid	string	The unique identifier of the SVM.

nvme_subsystem

An NVMe subsystem maintains configuration state and namespace access control for a set of NVMe-connected hosts.

Name	Type	Description
_links	_links	
comment	string	A configurable comment for the NVMe subsystem. Optional in POST and PATCH.
delete_on_unmap	boolean	An option that causes the subsystem to be deleted when the last subsystem map associated with it is deleted. Optional in POST and PATCH. This property defaults to *false* when the subsystem is created.
hosts	array[hosts]	The NVMe hosts configured for access to the NVMe subsystem. Optional in POST.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.
name	string	The name of the NVMe subsystem. Once created, an NVMe subsystem cannot be renamed. Required in POST.
os_type	string	The host operating system of the NVMe subsystem's hosts. Required in POST.
serial_number	string	The serial number of the NVMe subsystem.

Name	Type	Description
subsystem_maps	array[subsystem_maps]	The NVMe namespaces mapped to the NVMe subsystem. There is an added computational cost to retrieving property values for `subsystem_maps`. They are not populated for either a collection GET or an instance GET unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.
svm	svm	
target_nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target.
uuid	string	The unique identifier of the NVMe subsystem.
vendor_uuids	array[string]	Vendor-specific identifiers (UUIDs) optionally assigned to an NVMe subsystem when the subsystem is created. The identifiers are used to enable vendor-specific NVMe protocol features. The identifiers are provided by a host application vendor and shared with NetApp prior to a joint product release. Creating an NVMe subsystem with an unknown or non-specific identifier will have no effect on the NVMe subsystem. Refer to the ONTAP SAN Administration Guide for a list of the supported vendor-specific identifiers. After a subsystem is created, the vendor-specific identifiers cannot be changed or removed. Optional in POST. • Introduced in: 9.9 • readCreate: 1

_links

Name	Type	Description
next	href	
self	href	

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Retrieve NVMe subsystem hosts

GET `/protocols/nvme/subsystems/{subsystem.uuid}/hosts`

Introduced In: 9.6

Retrieves the NVMe subsystem hosts of an NVMe subsystem.

Expensive properties

There is an added computational cost to retrieving values for these properties. They are not included by default in GET results and must be explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

- `subsystem_maps.*`

Related ONTAP commands

- `vserver nvme subsystem map show`
- `vserver nvme subsystem show`

Learn more

- DOC /protocols/nvme/subsystems

Parameters

Name	Type	In	Required	Description
subsystem.uuid	string	path	True	The unique identifier of the NVMe subsystem.
fields	array[string]	query	False	Specify the fields to return.
max_records	integer	query	False	Limit the number of records returned.
return_records	boolean	query	False	The default is true for GET calls. When set to false, only the number of records is returned. • Default value: 1
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When iterating over a collection, the default is 15 seconds. ONTAP returns earlier if either max records or the end of the collection is reached. • Default value: 1 • Max value: 120 • Min value: 0
order_by	array[string]	query	False	Order results by specified fields and optional [asc

Response

```
Status: 200, Ok
```

Name	Type	Description
_links	_links	
num_records	integer	The number of records in the response.
records	array[nvme_subsystem_host]	

Example response


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "dh_hmac_chap": {
      "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "group_size": "none",
      "hash_function": "sha_256",
      "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "mode": "bidirectional"
    },
    "io_queue": {
      "count": 4,
      "depth": 32
    },
    "nqn": "nqn.1992-01.example.com:string",
    "records": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "dh_hmac_chap": {
        "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
        "group_size": "none",
        "hash_function": "sha_256",
        "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
        "mode": "bidirectional"
```

```
      },
      "io_queue": {
        "count": 4,
        "depth": 32
      },
      "nqn": "nqn.1992-01.example.com:string",
      "subsystem": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "subsystem1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      }
    },
    "subsystem": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "subsystem1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  }
}
```


Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
72090001	The NVMe subsystem does not exist.

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
next	href	
self	href	

_links

Name	Type	Description
self	href	

dh_hmac_chap

A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.

Name	Type	Description
controller_secret_key	string	The controller secret for NVMe in-band authentication. The value of this property is used by the NVMe host to authenticate the NVMe controller while establishing a connection. If unset, the controller is not authenticated. When supplied, the property `host_secret_key` must also be supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a controller secret has been set for the host, but the controller secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.

Name	Type	Description
group_size	string	The Diffie-Hellman group size for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `2048_bit`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.
hash_function	string	The hash function for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `sha_256`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.
host_secret_key	string	The host secret for NVMe in-band authentication. The value of this property is used by the NVMe controller to authenticate the NVMe host while establishing a connection. If unset, no authentication is performed by the host or controller. This property must be supplied if any other NVMe in-band authentication properties are supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a host secret has been set for the host, but the host secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.

Name	Type	Description
mode	string	The expected NVMe in-band authentication mode for the host. This property is an indication of which secrets are configured for the host. When set to: • none: The host has neither the host nor controller secret configured, and no authentication is performed. • unidirectional: The host has a host secret configured. The controller will authenticate the host. • bidirectional: The host has both a host and controller secret configured. The controller will authenticate the host and the host will authenticate the controller.

io_queue

The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.

Name	Type	Description
count	integer	The number of I/O queue pairs. The default value is inherited from the owning NVMe subsystem.
depth	integer	The I/O queue depth. The default value is inherited from the owning NVMe subsystem.

subsystem

The NVMe subsystem to which the NVMe host has been provisioned.

Name	Type	Description
_links	_links	
name	string	The name of the NVMe subsystem.

Name	Type	Description
uuid	string	The unique identifier of the NVMe subsystem.

records

The NVMe host provisioned to access NVMe namespaces mapped to a subsystem.

Name	Type	Description
_links	_links	
dh_hmac_chap	dh_hmac_chap	A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.
nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target. Not allowed in POST when the `records` property is used.
subsystem	subsystem	The NVMe subsystem to which the NVMe host has been provisioned.

nvme_subsystem_host

The NVMe host provisioned to access NVMe namespaces mapped to a subsystem.

Name	Type	Description
_links	_links	
dh_hmac_chap	dh_hmac_chap	A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.

Name	Type	Description
nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target. Not allowed in POST when the `records` property is used.
records	array[records]	An array of NVMe hosts specified to add multiple NVMe hosts to an NVMe subsystem in a single API call. Valid in POST only.
subsystem	subsystem	The NVMe subsystem to which the NVMe host has been provisioned.

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Add NVMe subsystem hosts

POST `/protocols/nvme/subsystems/{subsystem.uuid}/hosts`

Introduced In: 9.6

Adds NVMe subsystem host(s) to an NVMe subsystem.

Required properties

- `nqn` or `records.nqn` - NVMe host(s) NQN(s) to add to the NVMe subsystem.

Related ONTAP commands

- `vserver nvme subsystem host add`

Learn more

- DOC /protocols/nvme/subsystems

Parameters

Name	Type	In	Required	Description
subsystem.uuid	string	path	True	The unique identifier of the NVMe subsystem.
return_records	boolean	query	False	The default is false. If set to true, the records are returned. • Default value:

Request Body

Name	Type	Description
_links	_links	
dh_hmac_chap	dh_hmac_chap	A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.
nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target. Not allowed in POST when the `records` property is used.

Name	Type	Description
records	array[records]	An array of NVMe hosts specified to add multiple NVMe hosts to an NVMe subsystem in a single API call. Valid in POST only.
subsystem	subsystem	The NVMe subsystem to which the NVMe host has been provisioned.

Example request


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "dh_hmac_chap": {
    "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
    "group_size": "none",
    "hash_function": "sha_256",
    "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
    "mode": "bidirectional"
  },
  "io_queue": {
    "count": 4,
    "depth": 32
  },
  "nqn": "nqn.1992-01.example.com:string",
  "records": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "dh_hmac_chap": {
      "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "group_size": "none",
      "hash_function": "sha_256",
      "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "mode": "bidirectional"
    },
    "io_queue": {
      "count": 4,
      "depth": 32
    },
    "nqn": "nqn.1992-01.example.com:string",
    "subsystem": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
```

```
          }
        },
        "name": "subsystem1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      }
    },
    "subsystem": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "subsystem1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  }
```


Response

```
Status: 201, Created
```

Name	Type	Description
_links	_links	
num_records	integer	The number of records in the response.
records	array[nvme_subsystem_host]	

Example response


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "dh_hmac_chap": {
      "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "group_size": "none",
      "hash_function": "sha_256",
      "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "mode": "bidirectional"
    },
    "io_queue": {
      "count": 4,
      "depth": 32
    },
    "nqn": "nqn.1992-01.example.com:string",
    "records": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "dh_hmac_chap": {
        "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
        "group_size": "none",
        "hash_function": "sha_256",
        "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
        "mode": "bidirectional"
```

```
      },
      "io_queue": {
        "count": 4,
        "depth": 32
      },
      "nqn": "nqn.1992-01.example.com:string",
      "subsystem": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "subsystem1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      }
    },
    "subsystem": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "subsystem1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  }
}
```


Headers

Name	Description	Type
Location	Useful for tracking the resource location	string

Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
262186	The "records" array and other host properties are mutually exclusive.

Error Code	Description
72089705	The NVMe subsystem host already exists for the NVMe subsystem.
72089771	The NQN is invalid. A non-empty qualifier is required after the prefix. An example of a valid NQN is *nqn.1992-01.com.example:string*.
72089772	The NQN is invalid. Add the prefix *'nqn'*. An example of a valid NQN is *nqn.1992-01.com.example:string*.
72089773	The NQN is invalid. The date field must be formatted *yyyy-mm*. An example of a valid NQN is *nqn.1992-01.com.example:string*.
72090001	The NVMe subsystem does not exist.
72090003	A host to be added to an NVMe subsystem is missing the "nqn" property.
72090041	An element in the "records" array contains an invalid property.
72090042	The `dh_hmac_chap.host_secret_key` property is required when setting any other NVMe in-band authentication properties for a host.

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
self	href	

dh_hmac_chap

A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.

Name	Type	Description
controller_secret_key	string	The controller secret for NVMe in-band authentication. The value of this property is used by the NVMe host to authenticate the NVMe controller while establishing a connection. If unset, the controller is not authenticated. When supplied, the property `host_secret_key` must also be supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a controller secret has been set for the host, but the controller secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.
group_size	string	The Diffie-Hellman group size for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `2048_bit`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.

Name	Type	Description
hash_function	string	The hash function for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `sha_256`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.
host_secret_key	string	The host secret for NVMe in-band authentication. The value of this property is used by the NVMe controller to authenticate the NVMe host while establishing a connection. If unset, no authentication is performed by the host or controller. This property must be supplied if any other NVMe in-band authentication properties are supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a host secret has been set for the host, but the host secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.

Name	Type	Description
mode	string	The expected NVMe in-band authentication mode for the host. This property is an indication of which secrets are configured for the host. When set to: • none: The host has neither the host nor controller secret configured, and no authentication is performed. • unidirectional: The host has a host secret configured. The controller will authenticate the host. • bidirectional: The host has both a host and controller secret configured. The controller will authenticate the host and the host will authenticate the controller.

io_queue

The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.

Name	Type	Description
count	integer	The number of I/O queue pairs. The default value is inherited from the owning NVMe subsystem.
depth	integer	The I/O queue depth. The default value is inherited from the owning NVMe subsystem.

subsystem

The NVMe subsystem to which the NVMe host has been provisioned.

Name	Type	Description
_links	_links	
name	string	The name of the NVMe subsystem.

Name	Type	Description
uuid	string	The unique identifier of the NVMe subsystem.

records

The NVMe host provisioned to access NVMe namespaces mapped to a subsystem.

Name	Type	Description
_links	_links	
dh_hmac_chap	dh_hmac_chap	A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.
nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target. Not allowed in POST when the `records` property is used.
subsystem	subsystem	The NVMe subsystem to which the NVMe host has been provisioned.

nvme_subsystem_host

The NVMe host provisioned to access NVMe namespaces mapped to a subsystem.

Name	Type	Description
_links	_links	
dh_hmac_chap	dh_hmac_chap	A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.

Name	Type	Description
nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target. Not allowed in POST when the `records` property is used.
records	array[records]	An array of NVMe hosts specified to add multiple NVMe hosts to an NVMe subsystem in a single API call. Valid in POST only.
subsystem	subsystem	The NVMe subsystem to which the NVMe host has been provisioned.

_links

Name	Type	Description
next	href	
self	href	

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Delete an NVMe subsystem host

DELETE `/protocols/nvme/subsystems/{subsystem.uuid}/hosts/{nqn}`

Introduced In: 9.6

Deletes an NVMe subsystem host from an NVMe subsystem.

Related ONTAP commands

- `vserver nvme subsystem host remove`

Learn more

- DOC /protocols/nvme/subsystems

Parameters

Name	Type	In	Required	Description
subsystem.uuid	string	path	True	The unique identifier of the NVMe subsystem.
nqn	string	path	True	The NVMe qualified name (NQN) used to identify the NVMe subsystem host.

Response

```
Status: 200, Ok
```

Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
72089771	The NQN is invalid. A non-empty qualifier is required after the prefix. An example of a valid NQN is *nqn.1992-01.com.example:string*.
72089772	The NQN is invalid. Add the prefix *'nqn'*. An example of a valid NQN is *nqn.1992-01.com.example:string*.

Error Code	Description
72089773	The NQN is invalid. The date field must be formatted *yyyy-mm*. An example of a valid NQN is *nqn.1992-01.com.example:string*.
72090001	The NVMe subsystem does not exist.

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Retrieve an NVMe subsystem host

GET `/protocols/nvme/subsystems/{subsystem.uuid}/hosts/{nqn}`

Introduced In: 9.6

Retrieves an NVMe subsystem host of an NVMe subsystem.

Related ONTAP commands

- `vserver nvme subsystem host show`

Learn more

- DOC /protocols/nvme/subsystems

Parameters

Name	Type	In	Required	Description
subsystem.uuid	string	path	True	The unique identifier of the NVMe subsystem.

Name	Type	In	Required	Description
nqn	string	path	True	The NVMe qualified name (NQN) used to identify the NVMe subsystem host.
fields	array[string]	query	False	Specify the fields to return.

Response

```
Status: 200, Ok
```

Name	Type	Description
_links	_links	
dh_hmac_chap	dh_hmac_chap	A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.
nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target. Not allowed in POST when the `records` property is used.
records	array[records]	An array of NVMe hosts specified to add multiple NVMe hosts to an NVMe subsystem in a single API call. Valid in POST only.
subsystem	subsystem	The NVMe subsystem to which the NVMe host has been provisioned.

Example response


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "dh_hmac_chap": {
    "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
    "group_size": "none",
    "hash_function": "sha_256",
    "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
    "mode": "bidirectional"
  },
  "io_queue": {
    "count": 4,
    "depth": 32
  },
  "nqn": "nqn.1992-01.example.com:string",
  "records": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "dh_hmac_chap": {
      "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "group_size": "none",
      "hash_function": "sha_256",
      "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "mode": "bidirectional"
    },
    "io_queue": {
      "count": 4,
      "depth": 32
    },
    "nqn": "nqn.1992-01.example.com:string",
    "subsystem": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
```

```
          }
        },
        "name": "subsystem1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      }
    },
    "subsystem": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "subsystem1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  }
```


Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
72090001	The NVMe subsystem does not exist.

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
self	href	

dh_hmac_chap

A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.

Name	Type	Description
controller_secret_key	string	The controller secret for NVMe in-band authentication. The value of this property is used by the NVMe host to authenticate the NVMe controller while establishing a connection. If unset, the controller is not authenticated. When supplied, the property `host_secret_key` must also be supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a controller secret has been set for the host, but the controller secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.
group_size	string	The Diffie-Hellman group size for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `2048_bit`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.

Name	Type	Description
hash_function	string	The hash function for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `sha_256`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.
host_secret_key	string	The host secret for NVMe in-band authentication. The value of this property is used by the NVMe controller to authenticate the NVMe host while establishing a connection. If unset, no authentication is performed by the host or controller. This property must be supplied if any other NVMe in-band authentication properties are supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a host secret has been set for the host, but the host secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.

Name	Type	Description
mode	string	The expected NVMe in-band authentication mode for the host. This property is an indication of which secrets are configured for the host. When set to: • none: The host has neither the host nor controller secret configured, and no authentication is performed. • unidirectional: The host has a host secret configured. The controller will authenticate the host. • bidirectional: The host has both a host and controller secret configured. The controller will authenticate the host and the host will authenticate the controller.

io_queue

The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.

Name	Type	Description
count	integer	The number of I/O queue pairs. The default value is inherited from the owning NVMe subsystem.
depth	integer	The I/O queue depth. The default value is inherited from the owning NVMe subsystem.

subsystem

The NVMe subsystem to which the NVMe host has been provisioned.

Name	Type	Description
_links	_links	
name	string	The name of the NVMe subsystem.

Name	Type	Description
uuid	string	The unique identifier of the NVMe subsystem.

records

The NVMe host provisioned to access NVMe namespaces mapped to a subsystem.

Name	Type	Description
_links	_links	
dh_hmac_chap	dh_hmac_chap	A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.
nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target. Not allowed in POST when the `records` property is used.
subsystem	subsystem	The NVMe subsystem to which the NVMe host has been provisioned.

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message

Name	Type	Description
target	string	The target parameter that caused the error.

Remove an NVMe subsystem

DELETE `/protocols/nvme/subsystems/{uuid}`

Introduced In: 9.6

Removes an NVMe subsystem.

Related ONTAP commands

- `vserver nvme subsystem delete`

Learn more

- DOC /protocols/nvme/subsystems

Parameters

Name	Type	In	Required	Description
uuid	string	path	True	The unique identifier of the NVMe subsystem.
allow_delete_while_mapped	boolean	query	False	Allows for the deletion of a mapped NVMe subsystem.
allow_delete_with_hosts	boolean	query	False	Allows for the deletion of an NVMe subsystem with NVMe hosts.

Response

```
Status: 200, Ok
```

Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
72090001	The NVMe subsystem does not exist.
72090023	The NVMe subsystem contains one or more mapped namespaces. Use the `allow_delete_while_mapped` query parameter to delete an NVMe subsystem with mapped NVMe namespaces.
72090024	The NVMe subsystem contains one or more NVMe hosts. Use the `allow_delete_with_hosts` query parameter to delete an NVMe subsystem with NVMe hosts.

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Retrieve an NVMe subsystem

GET `/protocols/nvme/subsystems/{uuid}`

Introduced In: 9.6

Retrieves an NVMe subsystem.

Expensive properties

There is an added computational cost to retrieving values for these properties. They are not included by default in GET results and must be explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

- `subsystem_maps.*`

Related ONTAP commands

- `vserver nvme subsystem host show`
- `vserver nvme subsystem map show`
- `vserver nvme subsystem show`

Learn more

- DOC /protocols/nvme/subsystems

Parameters

Name	Type	In	Required	Description
uuid	string	path	True	The unique identifier of the NVMe subsystem.
fields	array[string]	query	False	Specify the fields to return.

Response

```
Status: 200, Ok
```

Name	Type	Description
_links	_links	
comment	string	A configurable comment for the NVMe subsystem. Optional in POST and PATCH.
delete_on_unmap	boolean	An option that causes the subsystem to be deleted when the last subsystem map associated with it is deleted. Optional in POST and PATCH. This property defaults to *false* when the subsystem is created.
hosts	array[hosts]	The NVMe hosts configured for access to the NVMe subsystem. Optional in POST.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.
name	string	The name of the NVMe subsystem. Once created, an NVMe subsystem cannot be renamed. Required in POST.

Name	Type	Description
os_type	string	The host operating system of the NVMe subsystem's hosts. Required in POST.
serial_number	string	The serial number of the NVMe subsystem.
subsystem_maps	array[subsystem_maps]	The NVMe namespaces mapped to the NVMe subsystem. There is an added computational cost to retrieving property values for `subsystem_maps`. They are not populated for either a collection GET or an instance GET unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.
svm	svm	
target_nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target.
uuid	string	The unique identifier of the NVMe subsystem.

Name	Type	Description
vendor_uuids	array[string]	Vendor-specific identifiers (UUIDs) optionally assigned to an NVMe subsystem when the subsystem is created. The identifiers are used to enable vendor-specific NVMe protocol features. The identifiers are provided by a host application vendor and shared with NetApp prior to a joint product release. Creating an NVMe subsystem with an unknown or non-specific identifier will have no effect on the NVMe subsystem. Refer to the ONTAP SAN Administration Guide for a list of the supported vendor-specific identifiers. After a subsystem is created, the vendor-specific identifiers cannot be changed or removed. Optional in POST. • Introduced in: 9.9 • readCreate: 1

Example response


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "comment": "string",
  "hosts": {
    "dh_hmac_chap": {
      "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "group_size": "none",
      "hash_function": "sha_256",
      "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "mode": "bidirectional"
    },
    "nqn": "nqn.1992-01.example.com:string"
  },
  "io_queue": {
    "default": {
      "count": 4,
      "depth": 16
    }
  },
  "name": "subsystem1",
  "os_type": "aix",
  "serial_number": "wCVsgFMiuMhVAAAAAAAB",
  "subsystem_maps": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "anagrpid": "00103050h",
    "namespace": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "/vol/vol1/namespace1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
```

```
    "nsid": "00000001h"
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "target_nqn": "nqn.1992-01.example.com:string",
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
  "vendor_uuids": {
  }
}
```


Error

```
Status: Default, Error
```

Name	Type	Description
error	error	

Example error


```
{
  "error": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```


Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
self	href	

dh_hmac_chap

A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.

Name	Type	Description
controller_secret_key	string	The controller secret for NVMe in-band authentication. The value of this property is used by the NVMe host to authenticate the NVMe controller while establishing a connection. If unset, the controller is not authenticated. When supplied, the property `host_secret_key` must also be supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a controller secret has been set for the host, but the controller secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.
group_size	string	The Diffie-Hellman group size for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `2048_bit`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.

Name	Type	Description
hash_function	string	The hash function for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `sha_256`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.
host_secret_key	string	The host secret for NVMe in-band authentication. The value of this property is used by the NVMe controller to authenticate the NVMe host while establishing a connection. If unset, no authentication is performed by the host or controller. This property must be supplied if any other NVMe in-band authentication properties are supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a host secret has been set for the host, but the host secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.

Name	Type	Description
mode	string	The expected NVMe in-band authentication mode for the host. This property is an indication of which secrets are configured for the host. When set to: • none: The host has neither the host nor controller secret configured, and no authentication is performed. • unidirectional: The host has a host secret configured. The controller will authenticate the host. • bidirectional: The host has both a host and controller secret configured. The controller will authenticate the host and the host will authenticate the controller.

hosts

Name	Type	Description
dh_hmac_chap	dh_hmac_chap	A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.
nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target.

default

The default I/O queue parameters inherited by NVMe hosts in the NVMe subsystem.

Name	Type	Description
count	integer	The number of host I/O queue pairs.
depth	integer	The host I/O queue depth.

io_queue

The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.

Name	Type	Description
default	default	The default I/O queue parameters inherited by NVMe hosts in the NVMe subsystem.

namespace

An NVMe namespace mapped to the NVMe subsystem.

Name	Type	Description
_links	_links	
name	string	The name of the NVMe namespace.
uuid	string	The unique identifier of the NVMe namespace.

subsystem_maps

An NVMe namespace mapped to the NVMe subsystem.

Name	Type	Description
_links	_links	
anagrpid	string	The Asymmetric Namespace Access Group ID (ANAGRPID) of the NVMe namespace. The format for an ANAGRPIP is 8 hexadecimal digits (zero-filled) followed by a lower case "h".
namespace	namespace	An NVMe namespace mapped to the NVMe subsystem.
nsid	string	The NVMe namespace identifier. This is an identifier used by an NVMe controller to provide access to the NVMe namespace. The format for an NVMe namespace identifier is 8 hexadecimal digits (zero-filled) followed by a lower case "h".

svm

Name	Type	Description
_links	_links	
name	string	The name of the SVM.
uuid	string	The unique identifier of the SVM.

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Update an NVMe subsystem

PATCH `/protocols/nvme/subsystems/{uuid}`

Introduced In: 9.6

Updates an NVMe subsystem.

Related ONTAP commands

- `vserver nvme subsystem modify`

Learn more

- DOC /protocols/nvme/subsystems

Parameters

Name	Type	In	Required	Description
uuid	string	path	True	The unique identifier of the NVMe subsystem.

Request Body

Name	Type	Description
_links	_links	
comment	string	A configurable comment for the NVMe subsystem. Optional in POST and PATCH.
delete_on_unmap	boolean	An option that causes the subsystem to be deleted when the last subsystem map associated with it is deleted. Optional in POST and PATCH. This property defaults to *false* when the subsystem is created.
hosts	array[hosts]	The NVMe hosts configured for access to the NVMe subsystem. Optional in POST.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.
name	string	The name of the NVMe subsystem. Once created, an NVMe subsystem cannot be renamed. Required in POST.
os_type	string	The host operating system of the NVMe subsystem's hosts. Required in POST.
serial_number	string	The serial number of the NVMe subsystem.

Name	Type	Description
subsystem_maps	array[subsystem_maps]	The NVMe namespaces mapped to the NVMe subsystem. There is an added computational cost to retrieving property values for `subsystem_maps`. They are not populated for either a collection GET or an instance GET unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.
svm	svm	
target_nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target.
uuid	string	The unique identifier of the NVMe subsystem.
vendor_uuids	array[string]	Vendor-specific identifiers (UUIDs) optionally assigned to an NVMe subsystem when the subsystem is created. The identifiers are used to enable vendor-specific NVMe protocol features. The identifiers are provided by a host application vendor and shared with NetApp prior to a joint product release. Creating an NVMe subsystem with an unknown or non-specific identifier will have no effect on the NVMe subsystem. Refer to the ONTAP SAN Administration Guide for a list of the supported vendor-specific identifiers. After a subsystem is created, the vendor-specific identifiers cannot be changed or removed. Optional in POST. • Introduced in: 9.9 • readCreate: 1

Example request


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "comment": "string",
  "hosts": {
    "dh_hmac_chap": {
      "controller_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "group_size": "none",
      "hash_function": "sha_256",
      "host_secret_key": "DHHC-
1:00:ia6zGodOr4SEG0Zzaw398rpY0wqipUWj4jWjUh4HWUz6aQ2n:",
      "mode": "bidirectional"
    },
    "nqn": "nqn.1992-01.example.com:string"
  },
  "io_queue": {
    "default": {
      "count": 4,
      "depth": 16
    }
  },
  "name": "subsystem1",
  "os_type": "aix",
  "serial_number": "wCVsgFMiuMhVAAAAAAAB",
  "subsystem_maps": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "anagrpid": "00103050h",
    "namespace": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "/vol/vol1/namespace1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
```

```
    "nsid": "00000001h"
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "target_nqn": "nqn.1992-01.example.com:string",
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
  "vendor_uuids": {
  }
}
```


Response

```
Status: 200, Ok
```

Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
72090001	The NVMe subsystem does not exist.

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": {
      "code": "string",
      "message": "string"
    },
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
self	href	

dh_hmac_chap

A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.

Name	Type	Description
controller_secret_key	string	The controller secret for NVMe in-band authentication. The value of this property is used by the NVMe host to authenticate the NVMe controller while establishing a connection. If unset, the controller is not authenticated. When supplied, the property `host_secret_key` must also be supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a controller secret has been set for the host, but the controller secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.
group_size	string	The Diffie-Hellman group size for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `2048_bit`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.

Name	Type	Description
hash_function	string	The hash function for NVMe in-band authentication. When property `host_secret_key` is provided, this property defaults to `sha_256`. When supplied, the property `host_secret_key` must also be supplied. Optional in POST.
host_secret_key	string	The host secret for NVMe in-band authentication. The value of this property is used by the NVMe controller to authenticate the NVMe host while establishing a connection. If unset, no authentication is performed by the host or controller. This property must be supplied if any other NVMe in-band authentication properties are supplied. Optional in POST. This property is write-only. The `mode` property can be used to identify if a host secret has been set for the host, but the host secret value cannot be read. To change the value, the host must be deleted from the subsystem and re-added.

Name	Type	Description
mode	string	The expected NVMe in-band authentication mode for the host. This property is an indication of which secrets are configured for the host. When set to: • none: The host has neither the host nor controller secret configured, and no authentication is performed. • unidirectional: The host has a host secret configured. The controller will authenticate the host. • bidirectional: The host has both a host and controller secret configured. The controller will authenticate the host and the host will authenticate the controller.

hosts

Name	Type	Description
dh_hmac_chap	dh_hmac_chap	A container for properties of NVMe in-band authentication with the DH-HMAC-CHAP protocol.
nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target.

default

The default I/O queue parameters inherited by NVMe hosts in the NVMe subsystem.

Name	Type	Description
count	integer	The number of host I/O queue pairs.
depth	integer	The host I/O queue depth.

io_queue

The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.

Name	Type	Description
default	default	The default I/O queue parameters inherited by NVMe hosts in the NVMe subsystem.

namespace

An NVMe namespace mapped to the NVMe subsystem.

Name	Type	Description
_links	_links	
name	string	The name of the NVMe namespace.
uuid	string	The unique identifier of the NVMe namespace.

subsystem_maps

An NVMe namespace mapped to the NVMe subsystem.

Name	Type	Description
_links	_links	
anagrpid	string	The Asymmetric Namespace Access Group ID (ANAGRPID) of the NVMe namespace. The format for an ANAGRPIP is 8 hexadecimal digits (zero-filled) followed by a lower case "h".
namespace	namespace	An NVMe namespace mapped to the NVMe subsystem.
nsid	string	The NVMe namespace identifier. This is an identifier used by an NVMe controller to provide access to the NVMe namespace. The format for an NVMe namespace identifier is 8 hexadecimal digits (zero-filled) followed by a lower case "h".

svm

Name	Type	Description
_links	_links	
name	string	The name of the SVM.
uuid	string	The unique identifier of the SVM.

nvme_subsystem

An NVMe subsystem maintains configuration state and namespace access control for a set of NVMe-connected hosts.

Name	Type	Description
_links	_links	
comment	string	A configurable comment for the NVMe subsystem. Optional in POST and PATCH.
delete_on_unmap	boolean	An option that causes the subsystem to be deleted when the last subsystem map associated with it is deleted. Optional in POST and PATCH. This property defaults to *false* when the subsystem is created.
hosts	array[hosts]	The NVMe hosts configured for access to the NVMe subsystem. Optional in POST.
io_queue	io_queue	The properties of the submission queue used to submit I/O commands for execution by the NVMe controller.
name	string	The name of the NVMe subsystem. Once created, an NVMe subsystem cannot be renamed. Required in POST.
os_type	string	The host operating system of the NVMe subsystem's hosts. Required in POST.
serial_number	string	The serial number of the NVMe subsystem.

Name	Type	Description
subsystem_maps	array[subsystem_maps]	The NVMe namespaces mapped to the NVMe subsystem. There is an added computational cost to retrieving property values for `subsystem_maps`. They are not populated for either a collection GET or an instance GET unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.
svm	svm	
target_nqn	string	The NVMe qualified name (NQN) used to identify the NVMe storage target.
uuid	string	The unique identifier of the NVMe subsystem.
vendor_uuids	array[string]	Vendor-specific identifiers (UUIDs) optionally assigned to an NVMe subsystem when the subsystem is created. The identifiers are used to enable vendor-specific NVMe protocol features. The identifiers are provided by a host application vendor and shared with NetApp prior to a joint product release. Creating an NVMe subsystem with an unknown or non-specific identifier will have no effect on the NVMe subsystem. Refer to the ONTAP SAN Administration Guide for a list of the supported vendor-specific identifiers. After a subsystem is created, the vendor-specific identifiers cannot be changed or removed. Optional in POST. • Introduced in: 9.9 • readCreate: 1

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.