# SAN

## REST API reference

NetApp
February 07, 2026

This PDF was generated from https://docs.netapp.com/us-en/ontap-restapi-9171/san_endpoints.html on February 07, 2026. Always check docs.netapp.com for the latest.

# Table of Contents


SAN . . . 1
ONTAP REST API SAN endpoints . . . 1
Overview . . . 1
Fibre Channel . . . 1
iSCSI . . . 1
Initiator Groups . . . 2
Portsets . . . 2
LUN Maps . . . 3
LUNs . . . 3
VMware Virtual Volume (vVol) Bindings . . . 3
Retrieve FC port information . . . 3
Retrieve FC port information . . . 3
Retrieve FC logins . . . 12
Retrieve an FC login . . . 21
Manage FC WWPN aliases . . . 28
Manage FC WWPN aliases . . . 28
Retrieve FC WWPN aliases . . . 33
Create an FC WWPN alias . . . 38
Delete an FC WWPN alias . . . 43
Retrieve an FC WWPN alias . . . 45
Manage FC services for SVMs . . . 50
Manage FC services for SVMs . . . 50
Retrieve FC protocol services . . . 55
Create an FC protocol service . . . 71
Delete an FC protocol service . . . 81
Retrieve an FC protocol service . . . 83
Update an FC protocol service . . . 95
Retrieve historical performance metrics for the FC protocol service of an SVM . . . 103
Retrieve historical performance metrics for the FC protocol service of an SVM for a specific time . . . 112
Parameters . . . 112
Response . . . 113
Error . . . 114
Definitions . . . 115
Manage SAN igroups . . . 118
Manage SAN igroups . . . 118
Retrieve initiator groups . . . 136
Create an initiator group . . . 167
Retrieve nested initiator groups of an initiator group . . . 195
Add nested initiator groups to an initiator group . . . 200
Remove a nested initiator group from an initiator group . . . 205
Retrieve a nested initiator group of an initiator group . . . 208
Retrieve initiators of an initiator group . . . 213
Add initiators to an initiator group . . . 225

Delete an initiator from an initiator group . . . 240
Retrieve an initiator . . . 243
Update an initiator . . . 254
Delete an initiator group . . . 265
Retrieve an initiator group . . . 267
Update an initiator group . . . 288
Manage SAN initiators . . . 307
Manage SAN initiators . . . 307
Retrieve initiators . . . 307
Retrieve an initiator . . . 313
Manage iSCSI credentials . . . 318
Manage iSCSI credentials . . . 318
Retrieve iSCSI credentials . . . 324
Create iSCSI credentials . . . 333
Delete iSCSI credentials . . . 342
Retrieve specific iSCSI credentials . . . 344
Update iSCSI credentials . . . 350
Manage iSCSI services . . . 357
Manage iSCSI services . . . 357
Retrieve iSCSI services . . . 363
Create an iSCSI service . . . 379
Delete an iSCSI service . . . 390
Retrieve an iSCSI service . . . 392
Update an iSCSI service . . . 404
Retrieve historical performance metrics for the iSCSI protocol of an SVM . . . 412
Retrieve historical performance metrics for the iSCSI protocol service of an SVM for a specific time . . . 421
View iSCSI sessions . . . 427
View iSCSI sessions . . . 427
Retrieve iSCSI sessions . . . 431
Retrieve an iSCSI session . . . 441
Manage SAN LUN maps . . . 450
Manage SAN LUN maps . . . 450
Retrieve LUN maps . . . 454
Create a LUN map . . . 466
Delete a LUN map . . . 477
Retrieve a LUN map . . . 479
Add, remove, or discover LUN map reporting nodes . . . 488
Add, remove, or discover LUN map reporting nodes . . . 488
Retrieve LUN map reporting nodes . . . 489
Add a reporting node and its HA partner to a LUN map . . . 495
Remove a reporting node and its HA partner from a LUN map . . . 500
Retrieve a LUN map reporting node . . . 502
Manage SAN portsets . . . 507
Manage SAN portsets . . . 507
Retrieve portsets . . . 514

Create a portset . . . 524
Retrieve portset network interfaces . . . 533
Add portset network interfaces . . . 539
Delete portset network interfaces . . . 546
Retrieve specific portset network interfaces . . . 548
Delete a portset . . . 554
Retrieve a portset . . . 556
Manage SAN vVol bindings . . . 563
Manage SAN vVol bindings . . . 563
Retrieve vVol bindings . . . 567
Create a vVol binding . . . 575
Delete a vVol binding . . . 584
Retrieve a vVol binding . . . 586
Manage LUNs . . . 592
Manage LUNs . . . 592
Retrieve LUNs . . . 604
Create a LUN . . . 670
Delete a LUN . . . 737
Retrieve LUN properties or data . . . 740
Update an existing LUN . . . 792
Manage LUN attributes . . . 843
Manage LUN attributes . . . 844
Retrieve LUN attributes . . . 848
Add an attribute to a LUN . . . 853
Delete a LUN attribute . . . 858
Retrieve a LUN attribute . . . 860
Update a LUN attribute value . . . 863
Retrieve historical performance metrics for a LUN . . . 867
Parameters . . . 867
Response . . . 870
Error . . . 872
Definitions . . . 872
Retrieve historical performance metrics for a LUN for a specific time . . . 876
Parameters . . . 876
Response . . . 877
Error . . . 878
Definitions . . . 879

# SAN

# ONTAP REST API SAN endpoints

## Overview

The storage area network (SAN) endpoints and objects enable you to configure, provision, and manage SAN-related objects.

---

## Fibre Channel

### Logins

Fibre Channel logins represent connections, formed by Fibre Channel initiators, that have successfully logged in to ONTAP. This represents the Fibre Channel login on which higher-level protocols, such as Fibre Channel Protocol (FCP) and Non-Volatile Memory Express over Fibre Channel (NVMe over FC), rely.

The Fibre Channel logins REST API provides information about active Fibre Channel logins.

### WWPN Aliases

A WWPN (world wide port name) is a unique 64-bit identifier for a Fibre Channel initiator. It is displayed as a 16-character hexadecimal value. SAN administrators may find it easier to identify Fibre Channel initiators using an alias, especially in larger SANs.

The WWPN alias REST API allows you to create, delete and discover aliases for WWPNs.

### Services

A Fibre Channel Protocol (FCP) service defines the properties of the Fibre Channel Protocol target for an SVM. There can be at most one FCP service for a given SVM. An SVM's FCP service must be created before FCP initiators can login to the SVM.

The Fibre Channel Protocol (FCP) service REST API allows you to create, update, delete, and discover Fibre Channel Services for SVMs. Fibre Channel interfaces are the logical endpoints for Fibre Channel network connections to an SVM.

---

## iSCSI

### Credentials

An iSCSI credentials object defines the authentication credentials to be used between an iSCSI initiator and ONTAP. It identifies an authentication type, user names, and the passwords that must be used to authenticate a specific initiator.

The iSCSI credentials REST API allows you to create, update, delete, and discover iSCSI credential objects.

### Services

An iSCSI service defines the properties of the iSCSI target for an SVM. There can be at most one iSCSI service for an SVM. An SVM's iSCSI service must be created before iSCSI initiators can login to the SVM.

The iSCSI service REST API allows you to create, update, delete, and discover iSCSI services for SVMs.

### Sessions

An iSCSI session consists of one or more TCP connections that link an iSCSI initiator with an iSCSI target. TCP connections can be added and removed from an iSCSI session by the iSCSI initiator. Across all TCP connections within an iSCSI session, an initiator sees one and the same target. After the connection is established, iSCSI control, data, and status messages are communicated over the session.

The iSCSI sessions REST API provides information about iSCSI initiators that have successfully logged in to ONTAP.

### Learn More

- *IP Interfaces* found in the *networking* section. IP interfaces are the logical endpoints for iSCSI network connections to an SVM.

---

## Initiator Groups

An initiator group (igroup) is a collection of Fibre Channel WWPNs (world wide port names), iSCSI IQNs (qualified names), iSCSI EUIs (extended unique identifiers), or any combination of these, that identify host initiators.

Initiator groups are used to control which hosts can access specific LUNs. To grant access to a LUN from one or more hosts, a network administrator creates an initiator group containing the hosts' initiator names, and then creates a LUN map that associates the initiator group with the LUN.

The initiator group REST API allows you to create, update, delete, and discover initiator groups. It also enables you to add and remove initiators that can access the target and associated LUNs.

## Portsets

A portset is a collection of Fibre Channel Protocol and/or iSCSI network interfaces from the portset's SVM.

Portsets are used to limit the network interfaces through which an initiator can connect to mapped LUNs. When a portset is bound to an initiator group (igroup), and the initiator group is mapped to a LUN, the initiators of the initiator group can only reach the LUN through the network interfaces in the portset.

Portsets are not recommended for new configurations. With modern releases of ONTAP, it is recommended to use multiple SVMs and initiator groups with no bound portset to load balance applications over multiple ports on a node. Selective LUN mapping will automatically limit the number of visible paths to a LUN from the client host to those required for efficient access and high availability. The REST portset API is primarily intended for legacy use.

The portset REST API allows you to create, delete, and discover portsets, and to add and remove network interfaces from portsets.

A portset can be bound to one or more initiator groups. An initiator group (igroup) can be bound to at most one

portset.

## LUN Maps

A LUN map is an association between a LUN and an initiator group. When a LUN is mapped to an initiator group, the group's initiators are granted access to the LUN. The relationship between an initiator group and a LUN is many initiator groups to many LUNs.

The LUN map REST API allows you to create, delete, and discover LUN maps and manage the reporting nodes of a LUN map.

## LUNs

A LUN is the logical representation of storage in a storage area network (SAN).

The LUN REST API allows you to create, update, delete, and discover LUNs.

## VMware Virtual Volume (vVol) Bindings

A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.

Class `protocol_endpoint` and `vvol` LUNs support many-to-many vVol bindings. A LUN of one class can be bound to zero or more LUNs of the opposite class.

The vVol binding between any two specific LUNs is reference counted. When a REST POST is executed for a vVol binding that already exists, the vVol binding reference count is incremented. When a REST DELETE is executed, the vVol binding reference count is decremented. Only when the vVol binding count reaches zero, or the query parameter `delete_all_references` is supplied, is the vVol binding destroyed.

The vVol binding REST API allows you to create, delete, and discover vVol bindings.

# Retrieve FC port information

## Retrieve FC port information

### Overview

Fibre Channel (FC) ports are the physical ports of FC adapters on ONTAP cluster nodes that can be connected to FC networks to provide FC network connectivity. An FC port defines the location of an FC interface within the ONTAP cluster.

The Fibre Channel port REST API allows you to discover FC ports, obtain status information for FC ports, and configure FC port properties. POST and DELETE requests are not supported. You must physically add and remove FC adapters to ONTAP nodes to create and remove ports from the ONTAP cluster.

### Performance monitoring

Performance of an FC port can be monitored by observing the `metric.*` and `statistics.*` properties. These properties show the performance of an FC port in terms of IOPS, latency, and throughput. The

`metric.*` properties denote an average, whereas `statistics.*` properties denote a real-time monotonically increasing value aggregated across all nodes.

## Examples

#### Retrieving all FC ports

```
# The API:
GET /api/network/fc/ports

# The call:
curl -X GET "https://<mgmt-ip>/api/network/fc/ports" -H "Accept:
application/hal+json"

# The response:
{
"records": [
  {
    "node": {
      "name": "node1",
      "uuid": "3c768e01-1abc-4b3b-b7c0-629ceb62a497",
      "_links": {
        "self": {
          "href": "/api/cluster/nodes/3c768e01-1abc-4b3b-b7c0-
629ceb62a497"
        }
      }
    },
    "uuid": "931b20f8-b047-11e8-9af3-005056bb838e",
    "name": "0a",
    "_links": {
      "self": {
        "href": "/api/network/fc/ports/931b20f8-b047-11e8-9af3-
005056bb838e"
      }
    }
  },
  {
    "node": {
      "name": "node1",
      "uuid": "3c768e01-1abc-4b3b-b7c0-629ceb62a497",
      "_links": {
        "self": {
          "href": "/api/cluster/nodes/3c768e01-1abc-4b3b-b7c0-
629ceb62a497"
        }
      }
```

```
    },
    "uuid": "931b23f7-b047-11e8-9af3-005056bb838e",
    "name": "0b",
    "_links": {
      "self": {
        "href": "/api/network/fc/ports/931b23f7-b047-11e8-9af3-
005056bb838e"
      }
    }
  },
  {
    "node": {
      "name": "node1",
      "uuid": "3c768e01-1abc-4b3b-b7c0-629ceb62a497",
      "_links": {
        "self": {
          "href": "/api/cluster/nodes/3c768e01-1abc-4b3b-b7c0-
629ceb62a497"
        }
      }
    },
    "uuid": "931b25ba-b047-11e8-9af3-005056bb838e",
    "name": "0c",
    "_links": {
      "self": {
        "href": "/api/network/fc/ports/931b25ba-b047-11e8-9af3-
005056bb838e"
      }
    }
  },
  {
    "node": {
      "name": "node1",
      "uuid": "3c768e01-1abc-4b3b-b7c0-629ceb62a497",
      "_links": {
        "self": {
          "href": "/api/cluster/nodes/3c768e01-1abc-4b3b-b7c0-
629ceb62a497"
        }
      }
    },
    "uuid": "931b2748-b047-11e8-9af3-005056bb838e",
    "name": "0d",
    "_links": {
      "self": {
        "href": "/api/network/fc/ports/931b2748-b047-11e8-9af3-
```

```
005056bb838e"
      }
    }
  },
  {
    "node": {
      "name": "node1",
      "uuid": "3c768e01-1abc-4b3b-b7c0-629ceb62a497",
      "_links": {
        "self": {
          "href": "/api/cluster/nodes/3c768e01-1abc-4b3b-b7c0-
629ceb62a497"
        }
      }
    },
    "uuid": "931b28c2-b047-11e8-9af3-005056bb838e",
    "name": "0e",
    "_links": {
      "self": {
        "href": "/api/network/fc/ports/931b28c2-b047-11e8-9af3-
005056bb838e"
      }
    }
  },
  {
    "node": {
      "name": "node1",
      "uuid": "3c768e01-1abc-4b3b-b7c0-629ceb62a497",
      "_links": {
        "self": {
          "href": "/api/cluster/nodes/3c768e01-1abc-4b3b-b7c0-
629ceb62a497"
        }
      }
    },
    "uuid": "931b2a7b-b047-11e8-9af3-005056bb838e",
    "name": "0f",
    "_links": {
      "self": {
        "href": "/api/network/fc/ports/931b2a7b-b047-11e8-9af3-
005056bb838e"
      }
    }
  },
  {
    "node": {
```

```
      "name": "node1",
      "uuid": "3c768e01-1abc-4b3b-b7c0-629ceb62a497",
      "_links": {
        "self": {
          "href": "/api/cluster/nodes/3c768e01-1abc-4b3b-b7c0-
629ceb62a497"
        }
      }
    },
    "uuid": "931b2e2b-b047-11e8-9af3-005056bb838e",
    "name": "1b",
    "_links": {
      "self": {
        "href": "/api/network/fc/ports/931b2e2b-b047-11e8-9af3-
005056bb838e"
      }
    }
  }
],
"num_records": 8,
"_links": {
  "self": {
    "href": "/api/network/fc/ports"
  }
}
}
```

#### Retrieving all FC ports with state *online*

The `state` query parameter is used to perform the query.

```
# The API:
GET /api/network/fc/ports

# The call:
curl -X GET "https://<mgmt-ip>/api/network/fc/ports?state=online" -H
"Accept: application/hal+json"

# The response:
{
"records": [
  {
    "node": {
      "name": "node1",
```

```
      "uuid": "3c768e01-1abc-4b3b-b7c0-629ceb62a497",
      "_links": {
        "self": {
          "href": "/api/cluster/nodes/3c768e01-1abc-4b3b-b7c0-
629ceb62a497"
        }
      }
    },
    "uuid": "931b20f8-b047-11e8-9af3-005056bb838e",
    "name": "0a",
    "state": "online",
    "_links": {
      "self": {
        "href": "/api/network/fc/ports/931b20f8-b047-11e8-9af3-
005056bb838e"
      }
    }
  },
  {
    "node": {
      "name": "node1",
      "uuid": "3c768e01-1abc-4b3b-b7c0-629ceb62a497",
      "_links": {
        "self": {
          "href": "/api/cluster/nodes/3c768e01-1abc-4b3b-b7c0-
629ceb62a497"
        }
      }
    },
    "uuid": "931b23f7-b047-11e8-9af3-005056bb838e",
    "name": "0b",
    "state": "online",
    "_links": {
      "self": {
        "href": "/api/network/fc/ports/931b23f7-b047-11e8-9af3-
005056bb838e"
      }
    }
  },
  {
    "node": {
      "name": "node1",
      "uuid": "3c768e01-1abc-4b3b-b7c0-629ceb62a497",
      "_links": {
        "self": {
          "href": "/api/cluster/nodes/3c768e01-1abc-4b3b-b7c0-
```

```
629ceb62a497"
        }
      }
    },
    "uuid": "931b25ba-b047-11e8-9af3-005056bb838e",
    "name": "0c",
    "state": "online",
    "_links": {
      "self": {
        "href": "/api/network/fc/ports/931b25ba-b047-11e8-9af3-
005056bb838e"
      }
    }
  }
],
"num_records": 3,
"_links": {
  "self": {
    "href": "/api/network/fc/ports?state=online"
  }
}
}
```

#### Retrieving an FC port

```
# The API:
GET /api/network/fc/ports/{uuid}

# The call:
curl -X GET "https://<mgmt-ip>/api/network/fc/ports/931b20f8-b047-11e8-
9af3-005056bb838e" -H "Accept: application/hal+json"

# The response:
{
"node": {
  "name": "node1",
  "uuid": "5a534a72-b047-11e8-9af3-005056bb838e",
  "_links": {
    "self": {
      "href": "/api/cluster/nodes/5a534a72-b047-11e8-9af3-005056bb838e"
    }
  }
},
"uuid": "931b20f8-b047-11e8-9af3-005056bb838e",
```

```
"name": "0a",
"description": "Fibre Channel Target Adapter 0a (ACME Fibre Channel
Adapter, rev. 1.0.0, 8G)",
"enabled": true,
"fabric": {
  "connected": true,
  "connected_speed": 8,
  "name": "55:0e:b1:a0:20:40:80:00",
  "port_address": "52100",
  "switch_port": "ssan-g620-03:1"
},
"physical_protocol": "fibre_channel",
"speed": {
  "maximum": "8",
  "configured": "auto"
},
"state": "online",
"supported_protocols": [
  "fcp"
],
"transceiver": {
  "form_factor": "SFP",
  "manufacturer": "ACME",
  "capabilities": [
    4,
    8
  ],
  "part_number": "1000"
},
"wwnn": "50:0a:09:80:bb:83:8e:00",
"wwpn": "50:0a:09:82:bb:83:8e:00",
"metric": {
  "timestamp": "2019-04-09T05:50:15Z",
  "duration": "PT15S",
  "status": "ok",
  "latency": {
    "other": 0,
    "total": 0,
    "read": 0,
    "write": 0
  },
  "iops": {
    "read": 0,
    "write": 0,
    "other": 0,
    "total": 0
```

```
  },
  "throughput": {
    "read": 0,
    "write": 0,
    "total": 0
  }
},
"statistics": {
  "timestamp": "2019-04-09T05:50:42Z",
  "status": "ok",
  "latency_raw": {
    "other": 38298,
    "total": 38298,
    "read": 0,
    "write": 0
  },
  "iops_raw": {
    "read": 0,
    "write": 0,
    "other": 3,
    "total": 3
  },
  "throughput_raw": {
    "read": 0,
    "write": 0,
    "total": 0
  }
},
"_links": {
  "self": {
    "href": "/api/network/fc/ports/931b20f8-b047-11e8-9af3-005056bb838e"
  }
}
}
```


#### Disabling an FC port

If an active FC interface exists on an FC port, the port cannot be disabled.

```
# The API:
PATCH /api/network/fc/ports/{uuid}

# The call:
curl -X PATCH "http://<mgmt-ip>/api/network/fc/ports/931b20f8-b047-11e8-
9af3-005056bb838e" -H "Accept: application/hal+json" -d '{ "enabled":
false }'
```

## Retrieve FC logins

GET `/network/fc/logins`

**Introduced In:** 9.6

Retrieves FC logins.

### Related ONTAP commands

- `vserver fcp initiator show`

### Learn more

- SAN: DOC /network/fc/logins
- NVMe: DOC /network/fc/logins

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.name | string | query | False | Filter by svm.name |
| svm.uuid | string | query | False | Filter by svm.uuid |
| interface.uuid | string | query | False | Filter by interface.uuid |
| interface.wwpn | string | query | False | Filter by interface.wwpn |
| interface.name | string | query | False | Filter by interface.name |
| protocol | string | query | False | Filter by protocol |
| initiator.wwpn | string | query | False | Filter by initiator.wwpn |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| initiator.comment | string | query | False | Filter by initiator.comment<br>• Introduced in: 9.9 |
| initiator.port_address | string | query | False | Filter by initiator.port_address |
| initiator.aliases | string | query | False | Filter by initiator.aliases |
| initiator.wwnn | string | query | False | Filter by initiator.wwnn |
| igroups.uuid | string | query | False | Filter by igroups.uuid |
| igroups.name | string | query | False | Filter by igroups.name<br>• maxLength: 96<br>• minLength: 1 |
| 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.<br>• Default value: 1 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[fc_login] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "next": {
          "href": "/api/resourcelink"
        },
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "igroups": [
        {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "igroup1",
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        }
      ],
      "initiator": {
        "aliases": [
          "alias1"
        ],
        "comment": "This is an FC initiator for host 5",
        "port_address": "5060A",
        "wwnn": "2f:a0:00:a0:98:0b:56:13",
        "wwpn": "2f:a0:00:a0:98:0b:56:13"
      },
      "interface": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
```

```
          }
        },
        "name": "fc_lif1",
        "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
        "wwpn": "20:00:00:50:56:b4:13:a8"
      },
      "protocol": "string",
      "svm": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      }
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

igroups

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

initiator

Information about the logged in FC initiator.

| Name | Type | Description |
|---|---|---|
| aliases | array[string] | The logged in initiator world wide port name (WWPN) aliases. |
| comment | string | A comment available for use by the administrator. This is modifiable from the initiator REST endpoint directly. See PATCH /protocols/san/igroups/{igroup.uuid}/initiators/{name} . |

| Name | Type | Description |
|---|---|---|
| port_address | string | The port address of the initiator's FC port.<br><br>Each port in an FC switched fabric has its own unique port address for routing purposes. The port address is assigned by a switch in the fabric when that port logs in to the fabric. This property refers to the address given by a switch to the initiator port.<br><br>This is useful for obtaining statistics and diagnostic information from FC switches.<br><br>This is a hexadecimal encoded numeric value. |
| wwnn | string | The logged in initiator world wide node name (WWNN). |
| wwpn | string | The logged in initiator WWPN. |

interface

An FC interface.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the FC interface. |
| uuid | string | The unique identifier of the FC interface. |
| wwpn | string | The WWPN of the FC interface. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

fc_login

A Fibre Channel (FC) login represents a connection formed by an FC initiator that has successfully logged in to ONTAP. This represents the FC login on which higher-level protocols such as Fibre Channel Protocol and NVMe over Fibre Channel (NVMe/FC) rely.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| igroups | array[igroups] | The initiator groups in which the initiator is a member. |
| initiator | initiator | Information about the logged in FC initiator. |
| interface | interface | An FC interface. |
| protocol | string | The data protocol used to perform the login. |
| svm | svm | SVM, applies only to SVM-scoped objects. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_error

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

| Name | Type | Description |
|---|---|---|
| message | string | Error message |
| target | string | The target parameter that caused the error. |

## Retrieve an FC login

GET `/network/fc/logins/{interface.uuid}/{initiator.wwpn}`

**Introduced In:** 9.6

Retrieves an FC login.

### Related ONTAP commands

- `vserver fcp initiator show`

### Learn more

- SAN: DOC /network/fc/logins
- NVMe: DOC /network/fc/logins

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| interface.uuid | string | path | True | The unique identifier of the FC interface through which the initiator logged in. |
| initiator.wwpn | string | path | True | The world wide port name (WWPN) of the initiator. |
| fields | array[string] | query | False | Specify the fields to return. |

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
| --- | --- | --- |
| igroups | array[igroups] | The initiator groups in which the initiator is a member. |
| initiator | initiator | Information about the logged in FC initiator. |
| interface | interface | An FC interface. |
| protocol | string | The data protocol used to perform the login. |
| svm | svm | SVM, applies only to SVM-scoped objects. |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "igroups": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "igroup1",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "initiator": {
    "aliases": [
      "alias1"
    ],
    "comment": "This is an FC initiator for host 5",
    "port_address": "5060A",
    "wwnn": "2f:a0:00:a0:98:0b:56:13",
    "wwpn": "2f:a0:00:a0:98:0b:56:13"
  },
  "interface": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "fc_lif1",
    "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
    "wwpn": "20:00:00:50:56:b4:13:a8"
  },
  "protocol": "string",
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
```

```
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  }
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 4 | The Fibre Channel login specified does not exist. |
| 5373982 | An invalid WWN was specified. The length is incorrect. |
| 5373983 | An invalid WWN was specified. The format is incorrect. |
| 5374881 | The Fibre Channel interface specified does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

igroups

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

initiator

Information about the logged in FC initiator.

| Name | Type | Description |
|---|---|---|
| aliases | array[string] | The logged in initiator world wide port name (WWPN) aliases. |
| comment | string | A comment available for use by the administrator. This is modifiable from the initiator REST endpoint directly. See PATCH /protocols/san/igroups/{igroup.uuid}/initiators/{name} . |

| Name | Type | Description |
|---|---|---|
| port_address | string | The port address of the initiator's FC port.<br><br>Each port in an FC switched fabric has its own unique port address for routing purposes. The port address is assigned by a switch in the fabric when that port logs in to the fabric. This property refers to the address given by a switch to the initiator port.<br><br>This is useful for obtaining statistics and diagnostic information from FC switches.<br><br>This is a hexadecimal encoded numeric value. |
| wwnn | string | The logged in initiator world wide node name (WWNN). |
| wwpn | string | The logged in initiator WWPN. |

interface

An FC interface.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the FC interface. |
| uuid | string | The unique identifier of the FC interface. |
| wwpn | string | The WWPN of the FC interface. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

# Manage FC WWPN aliases

## Manage FC WWPN aliases

### Overview

A worldwide port name (WWPN) is a unique 64-bit identifier for a Fibre Channel (FC) initiator. It is displayed as a 16-character hexadecimal value. SAN administrators might find it easier to identify FC initiators using an alias, especially in larger SANs.

The WWPN alias REST API allows you to create, delete, and discover aliases for WWPNs.

Multiple aliases can be created for a WWPN, but you cannot use the same alias for multiple WWPNs.

An alias can consist of up to 32 characters. Valid characters are:

- A through Z

- a through z
- numbers 0 through 9
- hyphen ("-")
- underscore ("_")
- left and right braces ("{", "}")
- period (".")

### Examples

#### Creating a WWPN alias

```
# The API:
POST /api/network/fc/wwpn-aliases

# The call:
curl -X POST "https://<mgmt-ip>/api/network/fc/wwpn-aliases" -H "Accept:
application/json" -d '{ "svm": { "name": "svm1" }, "wwpn":
"50:0a:09:82:b4:30:25:05", "alias": "alias3" }'
```

#### Retrieving all properties of all WWPN aliases

The `fields` query parameter is used to request that all properties be returned.

```
# The API:
GET /api/network/fc/wwpn-aliases

# The call:
curl -X GET "https://<mgmt-ip>/api/network/fc/wwpn-aliases?fields=*" -H
"Accept: application/hal+json"

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "68589d3d-7efa-11e8-9eed-005056b43025",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/68589d3d-7efa-11e8-9eed-005056b43025"
        }
      }
    },
    "alias": "alias1",
```

```
    "wwpn": "20:00:00:50:56:b4:30:25",
    "_links": {
      "self": {
        "href": "/api/network/fc/wwpn-aliases/68589d3d-7efa-11e8-9eed-
005056b43025/alias1"
      }
    }
  },
  {
    "svm": {
      "uuid": "68589d3d-7efa-11e8-9eed-005056b43025",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/68589d3d-7efa-11e8-9eed-005056b43025"
        }
      }
    },
    "alias": "alias2",
    "wwpn": "50:0a:09:82:b4:30:25:00",
    "_links": {
      "self": {
        "href": "/api/network/fc/wwpn-aliases/68589d3d-7efa-11e8-9eed-
005056b43025/alias2"
      }
    }
  }
],
"num_records": 2,
"_links": {
  "self": {
    "href": "/api/network/fc/wwpn-aliases"
  }
}
}
```


#### Retrieving all WWPN aliases named "alias1"

The `alias` query parameter is used to specify a query for the value "alias1".

```
# The API:
GET /api/network/fc/wwpn-aliases

# The call:
curl -X GET "https://<mgmt-ip>/api/network/fc/wwpn-aliases?alias=alias1"
-H "Accept: application/hal+json"

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "68589d3d-7efa-11e8-9eed-005056b43025",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/68589d3d-7efa-11e8-9eed-005056b43025"
        }
      }
    },
    "alias": "alias1",
    "wwpn": "20:00:00:50:56:b4:30:25",
    "_links": {
      "self": {
        "href": "/api/network/fc/wwpn-aliases/68589d3d-7efa-11e8-9eed-
005056b43025/alias1"
      }
    }
  }
],
"num_records": 1,
"_links": {
  "self": {
    "href": "/api/network/fc/wwpn-aliases?alias=alias1"
  }
}
}
```

#### Retrieving a specific WWPN alias

The alias to be returned is identified by the UUID of its SVM and the alias name.

```
# The API:
GET /api/network/fc/wwpn-aliases/{svm.uuid}/{alias}

# The call:
curl -X GET "https://<mgmt-ip>/api/network/fc/wwpn-aliases/68589d3d-7efa-
11e8-9eed-005056b43025/alias2" -H "Accept: application/hal+json"

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "68589d3d-7efa-11e8-9eed-005056b43025",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/68589d3d-7efa-11e8-9eed-005056b43025"
        }
      }
    },
    "alias": "alias2",
    "wwpn": "50:0a:09:82:b4:30:25:00",
    "_links": {
      "self": {
        "href": "/api/network/fc/wwpn-aliases/68589d3d-7efa-11e8-9eed-
005056b43025/alias1"
      }
    }
  }
],
"num_records": 1,
"_links": {
  "self": {
    "href": "/api/network/fc/wwpn-aliases?alias=alias1"
  }
}
}
```

#### Deleting a WWPN alias

The alias to delete is identified by the UUID of its SVM and the alias name.

```
# The API:
DELETE /api/network/fc/wwpn-aliases/{svm.uuid}/{alias}

# The call:
curl -X DELETE "https://<mgmt-ip>/api/network/fc/wwpn-aliases/68589d3d-
7efa-11e8-9eed-005056b43025/alias2" -H "Accept: application/hal+json"
```

## Retrieve FC WWPN aliases

GET `/network/fc/wwpn-aliases`

**Introduced In:** 9.6

Retrieves FC WWPN aliases.

### Related ONTAP commands

- `vserver fcp wwpn-alias show`

### Learn more

- DOC /network/fc/wwpn-aliases

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.name | string | query | False | Filter by svm.name |
| svm.uuid | string | query | False | Filter by svm.uuid |
| alias | string | query | False | Filter by alias |
| wwpn | string | query | False | Filter by wwpn |
| 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.<br>• Default value: 1 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[wwpn_alias] | |

### Example response


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "alias": "host1",
      "svm": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "wwpn": "2f:a0:00:a0:98:0b:56:13"
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

wwpn_alias

A Fibre Channel (FC) world wide port name (WWPN) alias. A WWPN is a unique 64-bit identifier for an FC initiator. It is displayed as a 16-character hexadecimal value. SAN administrators may find it easier to identify FC initiators using an alias, especially in larger SANs.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| alias | string | The FC WWPN alias. Required in POST. |
| svm | svm | SVM, applies only to SVM-scoped objects. |

| Name | Type | Description |
|---|---|---|
| wwpn | string | The FC initiator WWPN. Required in POST. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 FC WWPN alias

POST `/network/fc/wwpn-aliases`

**Introduced In:** 9.6

Creates an FC WWPN alias.

### Required properties

- `svm.uuid` or `svm.name` - Existing SVM in which to create the FC alias.
- `alias` - Name of the FC alias.
- `wwpn` - FC WWPN for which to create the alias.

### Related ONTAP commands

- `vserver fcp wwpn-alias set`

### Learn more

- DOC /network/fc/wwpn-aliases

### Parameters

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

### Request Body

| Name | Type | Description |
|---|---|---|
| alias | string | The FC WWPN alias. Required in POST. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| wwpn | string | The FC initiator WWPN. Required in POST. |

### Example request

```
{
  "alias": "host1",
  "svm": {
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "wwpn": "2f:a0:00:a0:98:0b:56:13"
}
```

### Response

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| num_records | integer | The number of records in the response. |
| records | array[wwpn_alias] | |

### Example response


```
{
  "num_records": 1,
  "records": [
    {
      "alias": "host1",
      "svm": {
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "wwpn": "2f:a0:00:a0:98:0b:56:13"
    }
  ]
}
```


#### Headers

| Name | Description | Type |
|---|---|---|
| Location | Useful for tracking the resource location | string |

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | The specified SVM does not exist or cannot be accessed by this user. |
| 2621706 | The specified `svm.uuid` and `svm.name` do not refer to the same SVM. |
| 2621707 | No SVM was specified. Either `svm.name` or `svm.uuid` must be supplied. |
| 5373982 | An invalid WWN was specified. The length is incorrect. |
| 5373983 | An invalid WWN was specified. The format is incorrect. |
| 5374869 | The alias already exists. |

Also see the table of common errors in the Response body overview section of this documentation.

## Definitions

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

wwpn_alias

A Fibre Channel (FC) world wide port name (WWPN) alias. A WWPN is a unique 64-bit identifier for an FC initiator. It is displayed as a 16-character hexadecimal value. SAN administrators may find it easier to identify FC initiators using an alias, especially in larger SANs.

| Name | Type | Description |
|---|---|---|
| alias | string | The FC WWPN alias. Required in POST. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| wwpn | string | The FC initiator WWPN. Required in POST. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 FC WWPN alias

DELETE `/network/fc/wwpn-aliases/{svm.uuid}/{alias}`

**Introduced In:** 9.6

Deletes an FC WWPN alias.

### Related ONTAP commands

- `vserver fcp wwpn-alias remove`

### Learn more

- DOC /network/fc/wwpn-aliases

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of the SVM. |
| alias | string | path | True | The name of FC WWPN alias. |

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
| --- | --- |
| 2621462 | The specified SVM does not exist or cannot be accessed by this user. |
| 5374046 | The alias could not be found. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
| --- | --- | --- |
| error | returned_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 |

returned_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 FC WWPN alias

GET `/network/fc/wwpn-aliases/{svm.uuid}/{alias}`

**Introduced In:** 9.6

Retrieves an FC WWPN alias.

### Related ONTAP commands

- `vserver fcp wwpn-alias show`

### Learn more

- DOC /network/fc/wwpn-aliases

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of the SVM in which the alias is found. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| alias | string | path | True | The name of FC WWPN alias. |
| fields | array[string] | query | False | Specify the fields to return. |

**Response**

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| alias | string | The FC WWPN alias. Required in POST. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| wwpn | string | The FC initiator WWPN. Required in POST. |

### Example response


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "alias": "host1",
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "wwpn": "2f:a0:00:a0:98:0b:56:13"
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | The specified SVM does not exist or cannot be accessed by this user. |
| 5374046 | The alias could not be found. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

# Manage FC services for SVMs

## Manage FC services for SVMs

### Overview

A Fibre Channel Protocol (FC Protocol) service defines the properties of the FC Protocol target for an SVM. There can be at most one FC Protocol service for an SVM. An SVM FC Protocol service must be created before FC Protocol initiators can log in to the SVM.

The FC Protocol service REST API allows you to create, update, delete, and discover FC services for SVMs.

### Performance monitoring

Performance of the SVM can be monitored by the `metric.*` and `statistics.*` properties. These show the performance of the SVM in terms of IOPS, latency, and throughput. The `metric.*` properties denote an average whereas `statistics.*` properties denote a real-time monotonically increasing value aggregated across all nodes.

### Examples

#### Creating an FC Protocol service for an SVM

The simplest way to create an FC Protocol service is to specify only the SVM, either by name or UUID. By default, the new FC Protocol service is enabled.

In this example, the `return_records` query parameter is used to retrieve the new FC Protocol service object in the REST response.

```
# The API:
POST /api/protocols/san/fcp/services

# The call:
curl -X POST 'https://<mgmt-
ip>/api/protocols/san/fcp/services?return_records=true' -H 'Accept:
application/hal+json' -d '{ "svm": { "name": "svm1" } }'

# The response:
{
"num_records": 1,
"records": [
  {
    "svm": {
      "uuid": "5c659d90-c01a-11e8-88ed-005056bbb24b",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/5c659d90-c01a-11e8-88ed-005056bbb24b"
        }
      }
    },
    "enabled": true,
    "target": {
      "name": "20:00:00:50:56:bb:b2:4b"
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/fcp/services/5c659d90-c01a-11e8-88ed-
005056bbb24b"
      }
    }
  }
]
}
```

#### Retrieving FC Protocol services for all SVMs in the cluster

```
# The API:
GET /api/protocols/san/fcp/services

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/fcp/services' -H 'Accept:
```

```
application/hal+json'

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "5c659d90-c01a-11e8-88ed-005056bbb24b",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/5c659d90-c01a-11e8-88ed-005056bbb24b"
        }
      }
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/fcp/services/5c659d90-c01a-11e8-88ed-
005056bbb24b"
      }
    }
  },
  {
    "svm": {
      "uuid": "6011f874-c01a-11e8-88ed-005056bbb24b",
      "name": "svm2",
      "_links": {
        "self": {
          "href": "/api/svm/svms/6011f874-c01a-11e8-88ed-005056bbb24b"
        }
      }
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/fcp/services/6011f874-c01a-11e8-88ed-
005056bbb24b"
      }
    }
  }
],
"num_records": 2,
"_links": {
  "self": {
    "href": "/api/protocols/san/fcp/services"
  }
}
```

```
}
```

#### Retrieving details for a specific FC Protocol service

The FC Protocol service is identified by the UUID of its SVM.

```
# The API:
GET /api/protocols/san/fcp/services/{svm.uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/fcp/services/5c659d90-
c01a-11e8-88ed-005056bbb24b' -H 'Accept: application/hal+json'

# The response:
{
"svm": {
  "uuid": "5c659d90-c01a-11e8-88ed-005056bbb24b",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/5c659d90-c01a-11e8-88ed-005056bbb24b"
    }
  }
},
"enabled": true,
"target": {
  "name": "20:00:00:50:56:bb:b2:4b"
},
"_links": {
  "self": {
    "href": "/api/protocols/san/fcp/services/5c659d90-c01a-11e8-88ed-
005056bbb24b"
  }
}
}
```

#### Disabling an FC Protocol service

Disabling an FC Protocol service shuts down all active FC Protocol logins for the SVM and prevents new FC Protocol logins.

The FC Protocol service to update is identified by the UUID of its SVM.

```
# The API:
PATCH /api/protocols/san/fcp/services/{svm.uuid}

# The call:
curl -X PATCH 'https://<mgmt-ip>/api/protocols/san/fcp/services/5c659d90-
c01a-11e8-88ed-005056bbb24b' -H 'Accept: application/hal+json' -d '{
"enabled": "false" }'
```

You can retrieve the FC Protocol service to confirm the change.

In this example, the `fields` query parameter is used to limit the response to the `enabled` property and FC Protocol service identifiers.

```
# The API:
GET /api/protocols/san/fcp/services/{svm.uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/fcp/services/5c659d90-
c01a-11e8-88ed-005056bbb24b?fields=enabled' -H 'Accept:
application/hal+json'

# The response:
{
"svm": {
  "uuid": "5c659d90-c01a-11e8-88ed-005056bbb24b",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/5c659d90-c01a-11e8-88ed-005056bbb24b"
    }
  }
},
"enabled": false,
"_links": {
  "self": {
    "href": "/api/protocols/san/fcp/services/5c659d90-c01a-11e8-88ed-
005056bbb24b"
  }
}
}
```

#### Deleting an FC Protocol service

The FC Protocol service must be disabled before it can be deleted.

The FC Protocol service to delete is identified by the UUID of its SVM.

```
# The API:
DELETE /api/protocols/san/fcp/services/{svm.uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/fcp/services/5c659d90-
c01a-11e8-88ed-005056bbb24b' -H 'Accept: application/hal+json'
```

## Retrieve FC protocol services

GET `/protocols/san/fcp/services`

**Introduced In:** 9.6

Retrieves FC Protocol services.

#### 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.

- `statistics.*`
- `metric.*`

#### Related ONTAP commands

- `vserver fcp show`

#### Learn more

- DOC /protocols/san/fcp/services

#### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.name | string | query | False | Filter by svm.name |
| svm.uuid | string | query | False | Filter by svm.uuid |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| statistics.latency_raw.other | integer | query | False | Filter by statistics.latency_raw.other<br>• Introduced in: 9.7 |
| statistics.latency_raw.total | integer | query | False | Filter by statistics.latency_raw.total<br>• Introduced in: 9.7 |
| statistics.latency_raw.read | integer | query | False | Filter by statistics.latency_raw.read<br>• Introduced in: 9.7 |
| statistics.latency_raw.write | integer | query | False | Filter by statistics.latency_raw.write<br>• Introduced in: 9.7 |
| statistics.throughput_raw.read | integer | query | False | Filter by statistics.throughput_raw.read<br>• Introduced in: 9.7 |
| statistics.throughput_raw.total | integer | query | False | Filter by statistics.throughput_raw.total<br>• Introduced in: 9.7 |
| statistics.throughput_raw.write | integer | query | False | Filter by statistics.throughput_raw.write<br>• Introduced in: 9.7 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| statistics.iops_raw.other | integer | query | False | Filter by statistics.iops_raw.other<br>• Introduced in: 9.7 |
| statistics.iops_raw.total | integer | query | False | Filter by statistics.iops_raw.total<br>• Introduced in: 9.7 |
| statistics.iops_raw.read | integer | query | False | Filter by statistics.iops_raw.read<br>• Introduced in: 9.7 |
| statistics.iops_raw.write | integer | query | False | Filter by statistics.iops_raw.write<br>• Introduced in: 9.7 |
| statistics.status | string | query | False | Filter by statistics.status<br>• Introduced in: 9.7 |
| statistics.timestamp | string | query | False | Filter by statistics.timestamp<br>• Introduced in: 9.7 |
| metric.iops.other | integer | query | False | Filter by metric.iops.other<br>• Introduced in: 9.7 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| metric.iops.total | integer | query | False | Filter by metric.iops.total<br>• Introduced in: 9.7 |
| metric.iops.read | integer | query | False | Filter by metric.iops.read<br>• Introduced in: 9.7 |
| metric.iops.write | integer | query | False | Filter by metric.iops.write<br>• Introduced in: 9.7 |
| metric.status | string | query | False | Filter by metric.status<br>• Introduced in: 9.7 |
| metric.latency.other | integer | query | False | Filter by metric.latency.other<br>• Introduced in: 9.7 |
| metric.latency.total | integer | query | False | Filter by metric.latency.total<br>• Introduced in: 9.7 |
| metric.latency.read | integer | query | False | Filter by metric.latency.read<br>• Introduced in: 9.7 |
| metric.latency.write | integer | query | False | Filter by metric.latency.write<br>• Introduced in: 9.7 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| metric.duration | string | query | False | Filter by metric.duration<br>• Introduced in: 9.7 |
| metric.timestamp | string | query | False | Filter by metric.timestamp<br>• Introduced in: 9.7 |
| metric.throughput.read | integer | query | False | Filter by metric.throughput.read<br>• Introduced in: 9.7 |
| metric.throughput.total | integer | query | False | Filter by metric.throughput.total<br>• Introduced in: 9.7 |
| metric.throughput.write | integer | query | False | Filter by metric.throughput.write<br>• Introduced in: 9.7 |
| enabled | boolean | query | False | Filter by enabled |
| target.name | string | query | False | Filter by target.name<br>• maxLength: 128<br>• minLength: 1 |
| fields | array[string] | query | False | Specify the fields to return. |
| max_records | integer | query | False | Limit the number of records returned. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| return_records | boolean | query | False | The default is true for GET calls. When set to false, only the number of records is returned.<br>• 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[fcp_service] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "metric": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "duration": "PT15S",
        "iops": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "latency": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "status": "ok",
        "throughput": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "timestamp": "2017-01-25 06:20:13 -0500"
      },
      "statistics": {
        "iops_raw": {
```

```
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "latency_raw": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "status": "ok",
        "throughput_raw": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "timestamp": "2017-01-25 06:20:13 -0500"
      },
      "svm": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "target": {
        "name": "20:00:00:50:56:bb:b2:4b"
      }
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

metric

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |

| Name | Type | Description |
|---|---|---|
| latency | latency | The round trip latency in microseconds observed at the storage object. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

iops_raw

The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency_raw

The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput_raw

Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

statistics

These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

| Name | Type | Description |
|---|---|---|
| iops_raw | iops_raw | The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time. |
| latency_raw | latency_raw | The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput_raw | throughput_raw | Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time. |

| Name | Type | Description |
|---|---|---|
| timestamp | string | The timestamp of the performance data. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

target

| Name | Type | Description |
|---|---|---|
| name | string | The target name of the FC Protocol service. This is generated for the SVM during POST.<br><br>The FC Protocol target name is a world wide node name (WWNN).<br><br>If required, the target name can be modified using the ONTAP command line.<br><br>• example: 20:00:00:50:56:bb:b2:4b<br>• maxLength: 128<br>• minLength: 1<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

fcp_service

A Fibre Channel (FC) Protocol service defines the properties of the FC Protocol target for an SVM. There can be at most one FC Protocol service for an SVM. An SVM's FC Protocol service must be created before FC Protocol initiators can login to the SVM.

A FC Protocol service is identified by the UUID of its SVM.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| enabled | boolean | The administrative state of the FC Protocol service. The FC Protocol service can be disabled to block all FC Protocol connectivity to the SVM.<br><br>This is optional in POST and PATCH. The default setting is *true* (enabled) in POST. |
| metric | metric | Performance numbers, such as IOPS latency and throughput, for SVM protocols. |
| statistics | statistics | These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target | target | |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

## Create an FC protocol service

POST `/protocols/san/fcp/services`

**Introduced In:** 9.6

Creates an FC Protocol service.

### Required properties

- `svm.uuid` or `svm.name` - Existing SVM in which to create the FC Protocol service.

### Related ONTAP commands

- `vserver fcp create`

### Learn more

- DOC /protocols/san/fcp/services

### Parameters

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

### Request Body

| Name | Type | Description |
|---|---|---|
| enabled | boolean | The administrative state of the FC Protocol service. The FC Protocol service can be disabled to block all FC Protocol connectivity to the SVM.<br><br>This is optional in POST and PATCH. The default setting is *true* (enabled) in POST. |

| Name | Type | Description |
|---|---|---|
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target | target | |

**Example request**

```
{
  "svm": {
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "target": {
    "name": "20:00:00:50:56:bb:b2:4b"
  }
}
```

**Response**

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| num_records | integer | The number of records in the response. |
| records | array[fcp_service] | |

### Example response


```
{
  "num_records": 1,
  "records": [
    {
      "svm": {
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "target": {
        "name": "20:00:00:50:56:bb:b2:4b"
      }
    }
  ]
}
```


#### Headers

| Name | Description | Type |
|---|---|---|
| Location | Useful for tracking the resource location | string |

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 1115127 | The cluster lacks a valid FC Protocol license. |
| 2621462 | The supplied SVM does not exist. |
| 2621507 | The Fibre Channel Protocol is not allowed for the specified SVM. |
| 2621706 | The specified `svm.uuid` and `svm.name` do not refer to the same SVM. |
| 2621707 | No SVM was specified. Either `svm.name` or `svm.uuid` must be supplied. |
| 5374082 | The Fibre Channel Protocol service already exists for the SVM. |

| Error Code | Description |
| --- | --- |
| 5374092 | The Fibre Channel Protocol is not supported on the cluster hardware configuration; there are no Fibre Channel adapters. |
| 5374893 | The SVM is stopped. The SVM must be running to create a Fibre Channel Protocol service. |
| 5376452 | Service POST and DELETE are not supported on ASA r2. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |

| Name | Type | Description |
|---|---|---|
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

metric

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

| Name | Type | Description |
|---|---|---|
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |

| Name | Type | Description |
| --- | --- | --- |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

iops_raw

The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

| Name | Type | Description |
| --- | --- | --- |
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency_raw

The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput_raw

Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

statistics

These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

| Name | Type | Description |
| --- | --- | --- |
| iops_raw | iops_raw | The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time. |
| latency_raw | latency_raw | The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput_raw | throughput_raw | Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time. |
| timestamp | string | The timestamp of the performance data. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

target

| Name | Type | Description |
|---|---|---|
| name | string | The target name of the FC Protocol service. This is generated for the SVM during POST.<br><br>The FC Protocol target name is a world wide node name (WWNN).<br><br>If required, the target name can be modified using the ONTAP command line.<br><br>• example: 20:00:00:50:56:bb:b2:4b<br>• maxLength: 128<br>• minLength: 1<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

fcp_service

A Fibre Channel (FC) Protocol service defines the properties of the FC Protocol target for an SVM. There can be at most one FC Protocol service for an SVM. An SVM's FC Protocol service must be created before FC Protocol initiators can login to the SVM.

A FC Protocol service is identified by the UUID of its SVM.

| Name | Type | Description |
|---|---|---|
| enabled | boolean | The administrative state of the FC Protocol service. The FC Protocol service can be disabled to block all FC Protocol connectivity to the SVM.<br><br>This is optional in POST and PATCH. The default setting is *true* (enabled) in POST. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target | target | |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 FC protocol service

DELETE `/protocols/san/fcp/services/{svm.uuid}`

**Introduced In:** 9.6

Deletes an FC Protocol service. An FC Protocol service must be disabled before it can be deleted.

### Related ONTAP commands

- `vserver fcp delete`

### Learn more

- DOC /protocols/san/fcp/services

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of the SVM for which to delete the FC Protocol service. |

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | An SVM with the specified UUID does not exist. |
| 5373960 | The Fibre Channel Protocol service cannot be removed while it is enabled. |
| 5374083 | There is no Fibre Channel Protocol service for the specified SVM. |
| 5376452 | Service POST and DELETE are not supported on ASA r2. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 |

returned_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 FC protocol service

GET `/protocols/san/fcp/services/{svm.uuid}`

**Introduced In:** 9.6

Retrieves an FC Protocol service.

### Related ONTAP commands

- `vserver fcp show`

### Learn more

- DOC /protocols/san/fcp/services

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of the SVM for which to retrieve the FC Protocol service. |
| fields | array[string] | query | False | Specify the fields to return. |

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| enabled | boolean | The administrative state of the FC Protocol service. The FC Protocol service can be disabled to block all FC Protocol connectivity to the SVM.<br><br>This is optional in POST and PATCH. The default setting is *true* (enabled) in POST. |
| metric | metric | Performance numbers, such as IOPS latency and throughput, for SVM protocols. |

| Name | Type | Description |
| --- | --- | --- |
| statistics | statistics | These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target | target | |

**Example response**


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "metric": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "duration": "PT15S",
    "iops": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "timestamp": "2017-01-25 06:20:13 -0500"
  },
  "statistics": {
    "iops_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput_raw": {
```

```
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "timestamp": "2017-01-25 06:20:13 -0500"
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "target": {
    "name": "20:00:00:50:56:bb:b2:4b"
  }
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | An SVM with the specified UUID does not exist. |
| 5374083 | There is no Fibre Channel Protocol service for the specified SVM. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

metric

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |

| Name | Type | Description |
| --- | --- | --- |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

iops_raw

The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

| Name | Type | Description |
| --- | --- | --- |
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency_raw

The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput_raw

Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

statistics

These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

| Name | Type | Description |
| --- | --- | --- |
| iops_raw | iops_raw | The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time. |
| latency_raw | latency_raw | The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput_raw | throughput_raw | Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time. |
| timestamp | string | The timestamp of the performance data. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

target

| Name | Type | Description |
|---|---|---|
| name | string | The target name of the FC Protocol service. This is generated for the SVM during POST.<br><br>The FC Protocol target name is a world wide node name (WWNN).<br><br>If required, the target name can be modified using the ONTAP command line.<br><br>• example: 20:00:00:50:56:bb:b2:4b<br>• maxLength: 128<br>• minLength: 1<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 FC protocol service

PATCH `/protocols/san/fcp/services/{svm.uuid}`

**Introduced In:** 9.6

Updates an FC Protocol service.

### Related ONTAP commands

- `vserver fcp modify`
- `vserver fcp start`
- `vserver fcp stop`

### Learn more

- DOC /protocols/san/fcp/services

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of the SVM whose FC Protocol service is to be updated. |

### Request Body

| Name | Type | Description |
|---|---|---|
| enabled | boolean | The administrative state of the FC Protocol service. The FC Protocol service can be disabled to block all FC Protocol connectivity to the SVM.<br><br>This is optional in POST and PATCH. The default setting is *true* (enabled) in POST. |
| target | target | |

**Example request**

```
{
  "target": {
    "name": "20:00:00:50:56:bb:b2:4b"
  }
}
```

**Response**

```
Status: 200, Ok
```

**Error**

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | An SVM with the specified UUID does not exist. |
| 5374083 | There is no Fibre Channel Protocol service for the specified SVM. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |

| Name | Type | Description |
|---|---|---|
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

metric

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

| Name | Type | Description |
|---|---|---|
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |

| Name | Type | Description |
| --- | --- | --- |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

iops_raw

The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

| Name | Type | Description |
| --- | --- | --- |
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency_raw

The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput_raw

Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

statistics

These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

| Name | Type | Description |
| --- | --- | --- |
| iops_raw | iops_raw | The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time. |
| latency_raw | latency_raw | The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput_raw | throughput_raw | Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time. |
| timestamp | string | The timestamp of the performance data. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

target

| Name | Type | Description |
|---|---|---|
| name | string | The target name of the FC Protocol service. This is generated for the SVM during POST.<br><br>The FC Protocol target name is a world wide node name (WWNN).<br><br>If required, the target name can be modified using the ONTAP command line.<br><br>• example: 20:00:00:50:56:bb:b2:4b<br>• maxLength: 128<br>• minLength: 1<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

fcp_service

A Fibre Channel (FC) Protocol service defines the properties of the FC Protocol target for an SVM. There can be at most one FC Protocol service for an SVM. An SVM's FC Protocol service must be created before FC Protocol initiators can login to the SVM.

A FC Protocol service is identified by the UUID of its SVM.

| Name | Type | Description |
|---|---|---|
| enabled | boolean | The administrative state of the FC Protocol service. The FC Protocol service can be disabled to block all FC Protocol connectivity to the SVM.<br><br>This is optional in POST and PATCH. The default setting is *true* (enabled) in POST. |
| target | target | |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 historical performance metrics for the FC protocol service of an SVM

GET `/protocols/san/fcp/services/{svm.uuid}/metrics`

**Introduced In:** 9.7

Retrieves historical performance metrics for the FC Protocol service of an SVM.

**Parameters**

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| throughput.read | integer | query | False | Filter by throughput.read |
| throughput.total | integer | query | False | Filter by throughput.total |
| throughput.write | integer | query | False | Filter by throughput.write |
| status | string | query | False | Filter by status |
| duration | string | query | False | Filter by duration |
| timestamp | string | query | False | Filter by timestamp |
| latency.other | integer | query | False | Filter by latency.other |
| latency.total | integer | query | False | Filter by latency.total |
| latency.read | integer | query | False | Filter by latency.read |
| latency.write | integer | query | False | Filter by latency.write |
| iops.other | integer | query | False | Filter by iops.other |
| iops.total | integer | query | False | Filter by iops.total |
| iops.read | integer | query | False | Filter by iops.read |
| iops.write | integer | query | False | Filter by iops.write |
| svm.uuid | string | path | True | The unique identifier of the SVM. |

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| interval | string | query | False | The time range for the data. Examples can be 1h, 1d, 1m, 1w, 1y. The period for each time range is as follows:<br>• 1h: Metrics over the most recent hour sampled over 15 seconds.<br>• 1d: Metrics over the most recent day sampled over 5 minutes.<br>• 1w: Metrics over the most recent week sampled over 30 minutes.<br>• 1m: Metrics over the most recent month sampled over 2 hours.<br>• 1y: Metrics over the most recent year sampled over a day.<br>• Default value: 1<br>• enum: ["1h", "1d", "1w", "1m", "1y"] |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| 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.<br>• Default value: 15<br>• Max value: 120<br>• Min value: 0 |
| fields | array[string] | query | False | Specify the fields to return. |
| max_records | integer | query | False | Limit the number of records returned. |
| order_by | array[string] | query | False | Order results by specified fields and optional [asc |
| desc] direction. Default direction is 'asc' for ascending. | return_records | boolean | query | False |

**Response**

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| num_records | integer | Number of records |
| records | array[records] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "duration": "PT15S",
      "iops": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "latency": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "status": "ok",
      "svm": {
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "throughput": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "timestamp": "2017-01-25 06:20:13 -0500"
    }
  ]
}
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 8585947 | No metrics are available for the requested object. |
| 8586225 | An unexpected error occurred retrieving metrics for the requested object. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

svm

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the SVM. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

records

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
| --- | --- | --- |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| svm | svm | |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 historical performance metrics for the FC protocol service of an SVM for a specific time

GET `/protocols/san/fcp/services/{svm.uuid}/metrics/{timestamp}`

**Introduced In:** 9.14

Retrieves historical performance metrics for the FC Protocol service of an SVM for a specific time.

## Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of the SVM. |
| timestamp | string | path | True | The timestamp of the performance data.<br>• format: date-time |
| fields | array[string] | query | False | Specify the fields to return. |

## Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| svm | svm | |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |

| Name | Type | Description |
|---|---|---|
| timestamp | string | The timestamp of the performance data. |

**Example response**


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "duration": "PT15S",
  "iops": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "latency": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "svm": {
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "throughput": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "timestamp": "2017-01-25 06:20:13 -0500"
}
```


## Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
| --- | --- |
| 8585947 | No metrics are available for the requested object. |
| 8586225 | An unexpected error occurred retrieving metrics for the requested object. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
| --- | --- | --- |
| error | returned_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 | |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

svm

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the SVM. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

# Manage SAN igroups

## Manage SAN igroups

### Overview

An initiator group (igroup) is a collection of Fibre Channel (FC) world wide port names (WWPNs), and/or iSCSI Qualified Names (IQNs), and/or iSCSI EUIs (Extended Unique Identifiers) that identify host endpoints.

Initiator groups are used to control which hosts can access specific LUNs. To grant access to a LUN from one or more hosts, create an initiator group containing the host initiator names, then create a LUN map that associates the initiator group with the LUN.

An initiator group may contain either initiators or other initiator groups, but not both simultaneously. When a parent initiator group is mapped, it inherits all of the initiators of any initiator groups nested below it. If any nested initiator group is modified to contain different initiators, the parent initiator groups inherit the change. A parent can have many nested initiator groups and an initiator group can be nested under multiple parents. Initiators can only be added or removed from the initiator group that directly contains them. The maximum supported depth of nesting is three layers.

Best practice when using nested initiator groups is to match host hierarchies. A single initiator group should correspond to a single host. If a LUN needs to be mapped to multiple hosts, the initiator groups representing those hosts should be aggregated into a parent initiator group and the LUN should be mapped to that initiator group. For multi-ported hosts, initiators have a comment property where the port corresponding to the initiator can be documented.

The initiator group REST API allows you to create, update, delete, and discover initiator groups, and to add and remove initiators that can access the target and associated LUNs.

An initiator can appear in multiple initiator groups. An initiator group can be mapped to multiple LUNs. A specific initiator can be mapped to a specific LUN only once. With the introduction of nestable initiator groups, best practice is to use the hierarchy such that an initiator is only a direct member of a single initiator group, and that initiator group can then be referenced by other initiator groups. This avoid needing to update multiple initiator groups when initiators change.

All initiators or nested initiator groups in an initiator group must be from the same operating system. The initiator group’s operating system is specified when the initiator group is created.

When an initiator group is created, the `protocol` property is used to restrict member initiators to Fibre Channel (*fcp*), iSCSI (*iscsi*), or both (*mixed*). Initiator groups within a nested hierarchy may not have conflicting protocols.

Zero or more initiators or nested initiator groups can be supplied when the initiator group is created. After creation, initiators can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/initiators` endpoint. Initiator groups containing other initiator groups report the aggregated list of initiators from all nested initiator groups, but modifications of the initiator list must be performed on the initiator group that directly contains the initiators. See DELETE

/protocols/san/igroups/{igroup.uuid}/initiators/{name} for more details.

An FC WWPN consists of 16 hexadecimal digits grouped as 8 pairs separated by colons. The format for an iSCSI IQN is *iqn.yyyy-mm.reverse_domain_name:any*. The iSCSI EUI format consists of the *eui.* prefix followed by 16 hexadecimal characters.

## Examples

#### Creating an initiator group with no initiators

The example initiator group used here is for Linux iSCSI initiators only. Note that the `return_records` query parameter is used to obtain the newly created initiator group in the response.

```
# The API:
POST /api/protocols/san/igroups

# The call:
curl -X POST 'https://<mgmt-
ip>/api/protocols/san/igroups?return_records=true' -H 'Accept:
application/hal+json' -d '{ "svm": { "name": "svm1" }, "name": "igroup1",
"os_type": "linux", "protocol": "iscsi" }'

# The response:
{
"num_records": 1,
"records": [
  {
    "svm": {
      "uuid": "02b0dfff-aa28-11e8-a653-005056bb7072",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/02b0dfff-aa28-11e8-a653-005056bb7072"
        }
      }
    },
    "uuid": "8f249e7d-ab9f-11e8-b8a3-005056bb7072",
    "name": "igroup1",
    "protocol": "iscsi",
    "os_type": "linux",
    "_links": {
      "self": {
        "href": "/api/protocols/san/igroups/8f249e7d-ab9f-11e8-b8a3-
005056bb7072"
      }
    }
  }
]
}
```

#### Creating an initiator group with initiators

The example initiator group used here is for Windows. FC Protocol and iSCSI initiators are allowed. Note that the `return_records` query parameter is used to obtain the newly created initiator group in the response.

```
# The API:
```

```
POST /api/protocols/san/igroups

# The call:
curl -X POST 'https://<mgmt-
ip>/api/protocols/san/igroups?return_records=true' -H 'Accept:
application/hal+json' -d '{ "svm": { "name": "svm1" }, "name": "igroup2",
"os_type": "windows", "protocol": "mixed", "initiators": [ { "name":
"20:01:00:50:56:bb:70:72" }, { "name": "iqn.1991-05.com.ms:host1" } ] }'

# The response:
{
"num_records": 1,
"records": [
  {
    "svm": {
      "uuid": "02b0dfff-aa28-11e8-a653-005056bb7072",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/02b0dfff-aa28-11e8-a653-005056bb7072"
        }
      }
    },
    "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7072",
    "name": "igroup2",
    "protocol": "mixed",
    "os_type": "windows",
    "initiators": [
      {
        "name": "20:01:00:50:56:bb:70:72",
        "_links": {
          "self": {
            "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7072/initiators/20:01:00:50:56:bb:70:72"
          }
        }
      },
      {
        "name": "iqn.1991-05.com.ms:host1",
        "_links": {
          "self": {
            "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7072/initiators/iqn.1991-05.com.ms:host1"
          }
        }
      }
```

```
    ],
    "_links": {
      "self": {
        "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7072"
      }
    }
  }
]
}
```

#### Creating an initiator group with nested initiator groups

The example initiator group used here is for Windows. FC Protocol and iSCSI initiators are allowed. Note that the `return_records` query parameter is used to obtain the newly created initiator group in the response. The new initiator group is create so as to contain the initiator group created in the previous example. The initiators list reports all initiators nested below this initiator group, and note that the href link for the initiators refers to the initiator group that directly owns the initiator, not this initiator group.

```
# The API:
POST /api/protocols/san/igroups

# The call:
curl -X POST 'https://<mgmt-
ip>/api/protocols/san/igroups?return_records=true' -H 'Accept:
application/hal+json' -d '{ "svm": { "name": "svm1" }, "name": "igroup3",
"os_type": "windows", "protocol": "mixed", "igroups": [ { "name":
"igroup2" } ] }'

# The response:
{
"num_records": 1,
"records": [
  {
    "svm": {
      "uuid": "02b0dfff-aa28-11e8-a653-005056bb7072",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/02b0dfff-aa28-11e8-a653-005056bb7072"
        }
      }
    },
    "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7073",
    "name": "igroup3",
```

```
    "protocol": "mixed",
    "os_type": "windows",
    "igroups": [
      {
        "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7072",
        "name": "igroup2",
        "_links": { "self": { "href":
"/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-005056bb7072" } }
      }
    ],
    "initiators": [
      {
        "name": "20:01:00:50:56:bb:70:72",
        "igroup": {
          "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7072",
          "name": "igroup2",
          "_links": { "self": { "href":
"/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-005056bb7072" } }
        },
        "_links": {
          "self": {
            "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7072/initiators/20:01:00:50:56:bb:70:72"
          }
        }
      },
      {
        "name": "iqn.1991-05.com.ms:host1",
        "igroup": {
          "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7072",
          "name": "igroup2",
          "_links": { "self": { "href":
"/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-005056bb7072" } }
        },
        "_links": {
          "self": {
            "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7072/initiators/iqn.1991-05.com.ms:host1"
          }
        }
      }
    ],
    "_links": {
      "self": {
        "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7073"
```

```
      }
    }
  }
]
}
```

#### Retrieving all initiator groups

```
# The API:
GET /api/protocols/san/igroups

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/igroups' -H 'Accept:
application/hal+json'

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "02b0dfff-aa28-11e8-a653-005056bb7072",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/02b0dfff-aa28-11e8-a653-005056bb7072"
        }
      }
    },
    "uuid": "8f249e7d-ab9f-11e8-b8a3-005056bb7072",
    "name": "igroup1",
    "_links": {
      "self": {
        "href": "/api/protocols/san/igroups/8f249e7d-ab9f-11e8-b8a3-
005056bb7072"
      }
    }
  },
  {
    "svm": {
      "uuid": "02b0dfff-aa28-11e8-a653-005056bb7072",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/02b0dfff-aa28-11e8-a653-005056bb7072"
```

```
        }
      }
    },
    "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7072",
    "name": "igroup2",
    "_links": {
      "self": {
        "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7072"
      }
    }
  },
  {
    "svm": {
      "uuid": "02b0dfff-aa28-11e8-a653-005056bb7072",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/02b0dfff-aa28-11e8-a653-005056bb7072"
        }
      }
    },
    "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7073",
    "name": "igroup3",
    "_links": {
      "self": {
        "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7073"
      }
    }
  }
],
"num_records": 3,
"_links": {
  "self": {
    "href": "/api/protocols/san/igroups"
  }
}
}
```

#### Retrieving all properties of all initiator groups

The `fields` query parameter is used to request all initiator group properties. Note that the nested and parent initiator groups are considered expensive properties and will only be returned if explicitly requested.

```
# The API:
GET /api/protocols/san/igroups

# The call:
curl -X GET 'https://<mgmt-
ip>/api/protocols/san/igroups?fields=*,igroups,parent_igroups' -H 'Accept:
application/hal+json'

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "02b0dfff-aa28-11e8-a653-005056bb7072",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/02b0dfff-aa28-11e8-a653-005056bb7072"
        }
      }
    },
    "uuid": "8f249e7d-ab9f-11e8-b8a3-005056bb7072",
    "name": "igroup1",
    "protocol": "iscsi",
    "os_type": "linux",
    "_links": {
      "self": {
        "href": "/api/protocols/san/igroups/8f249e7d-ab9f-11e8-b8a3-
005056bb7072"
      }
    }
  },
  {
    "svm": {
      "uuid": "02b0dfff-aa28-11e8-a653-005056bb7072",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/02b0dfff-aa28-11e8-a653-005056bb7072"
        }
      }
    },
    "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7072",
    "name": "igroup2",
    "parent_igroups": [
      {
```

```
        "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7073",
        "name": "igroup3",
        "_links": {
          "self": {
            "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7073"
          }
        }
      }
    ],
    "protocol": "mixed",
    "os_type": "windows",
    "initiators": [
      {
        "name": "20:01:00:50:56:bb:70:72",
        "_links": {
          "self": {
            "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7072/initiators/20:01:00:50:56:bb:70:72"
          }
        }
      },
      {
        "name": "iqn.1991-05.com.ms:host1",
        "_links": {
          "self": {
            "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7072/initiators/iqn.1991-05.com.ms:host1"
          }
        }
      }
    ],
    "_links": {
      "self": {
        "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7072"
      }
    }
  },
  {
    "svm": {
      "uuid": "02b0dfff-aa28-11e8-a653-005056bb7072",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/02b0dfff-aa28-11e8-a653-005056bb7072"
```

```
        }
      }
    },
    "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7073",
    "name": "igroup3",
    "protocol": "mixed",
    "os_type": "windows",
    "igroups": [
      {
        "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7072",
        "name": "igroup2",
        "_links": { "self": { "href":
"/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-005056bb7072" } }
      }
    ],
    "initiators": [
      {
        "name": "20:01:00:50:56:bb:70:72",
        "igroup": {
          "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7072",
          "name": "igroup2",
          "_links": { "self": { "href":
"/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-005056bb7072" } }
        },
        "_links": {
          "self": {
            "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7072/initiators/20:01:00:50:56:bb:70:72"
          }
        }
      },
      {
        "name": "iqn.1991-05.com.ms:host1",
        "igroup": {
          "uuid": "abf9c39d-ab9f-11e8-b8a3-005056bb7072",
          "name": "igroup2",
          "_links": { "self": { "href":
"/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-005056bb7072" } }
        },
        "_links": {
          "self": {
            "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7072/initiators/iqn.1991-05.com.ms:host1"
          }
        }
      }
```

```
    ],
    "_links": {
      "self": {
        "href": "/api/protocols/san/igroups/abf9c39d-ab9f-11e8-b8a3-
005056bb7073"
      }
    }
  }
],
"num_records": 3,
"_links": {
  "self": {
    "href": "/api/protocols/san/igroups?fields=*"
  }
}
}
```

#### Retrieving all initiator groups for Linux

The `os_type` query parameter is used to perform the query.

```
# The API:
GET /api/protocols/san/igroups

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

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "02b0dfff-aa28-11e8-a653-005056bb7072",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/02b0dfff-aa28-11e8-a653-005056bb7072"
        }
      }
    },
    "uuid": "8f249e7d-ab9f-11e8-b8a3-005056bb7072",
    "name": "igroup1",
    "os_type": "linux",
    "_links": {
      "self": {
        "href": "/api/protocols/san/igroups/8f249e7d-ab9f-11e8-b8a3-
005056bb7072"
      }
    }
  }
],
"num_records": 1,
"_links": {
  "self": {
    "href": "/api/protocols/san/igroups?os_type=linux"
  }
}
}
```

#### Retrieving a specific initiator group

```
# The API:
GET /api/protocols/san/igroups/{uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/igroups/8f249e7d-ab9f-
11e8-b8a3-005056bb7072' -H 'Accept: application/hal+json'

# The response:
{
"svm": {
  "uuid": "02b0dfff-aa28-11e8-a653-005056bb7072",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/02b0dfff-aa28-11e8-a653-005056bb7072"
    }
  }
},
"uuid": "8f249e7d-ab9f-11e8-b8a3-005056bb7072",
"name": "igroup1",
"protocol": "iscsi",
"os_type": "linux",
"_links": {
  "self": {
    "href": "/api/protocols/san/igroups/8f249e7d-ab9f-11e8-b8a3-
005056bb7072"
  }
}
}
```

#### Retrieving LUNs mapped to a specific initiator group

The `fields` parameter is used to specify the desired properties.

```
# The API:
GET /api/protocols/san/igroups

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/igroups/8f249e7d-ab9f-
11e8-b8a3-005056bb7072?fields=lun_maps' -H 'Accept: application/hal+json'

# The response:
{
```

```
"svm": {
  "uuid": "02b0dfff-aa28-11e8-a653-005056bb7072",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/02b0dfff-aa28-11e8-a653-005056bb7072"
    }
  }
},
"uuid": "8f249e7d-ab9f-11e8-b8a3-005056bb7072",
"name": "igroup1",
"lun_maps": [
  {
    "logical_unit_number": 0,
    "lun": {
      "name": "/vol/vol1/lun1",
      "uuid": "4b33ba57-c4e0-4dbb-bc47-214800d18a71",
      "node": {
        "name": "node1",
        "uuid": "f17182af-223f-4d51-8197-2cb2146d5c4c",
        "_links": {
          "self": {
            "href": "/api/cluster/nodes/f17182af-223f-4d51-8197-
2cb2146d5c4c"
          }
        }
      },
      "_links": {
        "self": {
          "href": "/api/storage/luns/4b33ba57-c4e0-4dbb-bc47-214800d18a71"
        }
      }
    }
  }
],
"_links": {
  "self": {
    "href": "/api/protocols/san/igroups/8f249e7d-ab9f-11e8-b8a3-
005056bb7072"
  }
}
}
```

#### Renaming an initiator group

Note that renaming an initiator group must be done in a PATCH request separate from any other modifications.

```
# The API:
PATCH /api/protocols/san/igroups/{uuid}

# The call:
curl -X PATCH 'https://<mgmt-ip>/api/protocols/san/igroups/8f249e7d-ab9f-
11e8-b8a3-005056bb7072' -H 'Accept: application/hal+json' -d '{ "name":
"igroup1_newName" }'
```

#### Changing the operating system type of an initiator group

```
# The API:
PATCH /api/protocols/san/igroups/{uuid}

# The call:
curl -X PATCH 'https://<mgmt-ip>/api/protocols/san/igroups/8f249e7d-ab9f-
11e8-b8a3-005056bb7072' -H 'Accept: application/hal+json' -d '{ "os_type":
"aix" }'
```

#### Adding an initiator to an initiator group

```
# The API:
POST /api/protocols/san/igroups/{igroup.uuid}/initiators

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/san/igroups/8f249e7d-ab9f-
11e8-b8a3-005056bb7072/initiators' -H 'Accept: application/hal+json' -d '{
"name": "iqn.1991-05.com.ms:host2" }'
```

#### Adding multiple initiators to an initiator group

Note the use of the `records` property to add multiple initiators to the initiator group in a single API call.

```
# The API:
POST /api/protocols/san/igroups/{igroup.uuid}/initiators

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/san/igroups/8f249e7d-ab9f-
11e8-b8a3-005056bb7072/initiators' -H 'Accept: application/hal+json' -d '{
"records": [ { "name": "iqn.1991-05.com.ms:host3" }, { "name": "iqn.1991-
05.com.ms:host4" } ] }'
```

#### Removing an initiator from an initiator group

```
# The API:
DELETE /api/protocols/san/igroups/{igroup.uuid}/initiators/iqn.1991-
05.com.ms:host3

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/igroups/8f249e7d-ab9f-
11e8-b8a3-005056bb7072/initiators/iqn.1991-05.com.ms:host3' -H 'Accept:
application/hal+json'
```

#### Removing an initiator from a mapped initiator group

Normally, removing an initiator from an initiator group that is mapped to a LUN is not allowed. The removal can be forced using the `allow_delete_while_mapped` query parameter.

```
# The API:
DELETE /api/protocols/san/igroups/{igroup.uuid}/initiators/iqn.1991-
05.com.ms:host4

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/igroups/8f249e7d-ab9f-
11e8-b8a3-005056bb7072/initiators/iqn.1991-
05.com.ms:host4?allow_delete_while_mapped=true' -H 'Accept:
application/hal+json'
```

#### Adding a nested initiator group to an initiator group

```
# The API:
POST /api/protocols/san/igroups/{igroup.uuid}/igroups

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/san/igroups/8f249e7d-ab9f-
11e8-b8a3-005056bb7072/igroups' -H 'Accept: application/hal+json' -d '{
"name": "host2_igroup" }'
```

#### Adding multiple nested initiator groups to an initiator group

Note the use of the `records` property to add multiple nested initiator groups to the initiator group in a single API call.

```
# The API:
POST /api/protocols/san/igroups/{igroup.uuid}/igroups

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/san/igroups/8f249e7d-ab9f-
11e8-b8a3-005056bb7072/igroups' -H 'Accept: application/hal+json' -d '{
"records": [ { "name": "host3_igroup" }, { "uuid": "c439efc8-0a70-11eb-
adc1-0242ac120002" } ] }'
```

#### Removing a nested initiator group from an initiator group

```
# The API:
DELETE /api/protocols/san/igroups/{igroup.uuid}/igroups/{uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/igroups/8f249e7d-ab9f-
11e8-b8a3-005056bb7072/igroups/c439efc8-0a70-11eb-adc1-0242ac120002' -H
'Accept: application/hal+json'
```

#### Removing a nested initiator group from a mapped initiator group

Normally, removing a nested initiator group from an initiator group that is mapped to a LUN is not allowed. The removal can be forced using the `allow_delete_while_mapped` query parameter.

```
# The API:
DELETE /api/protocols/san/igroups/{igroup.uuid}/igroups/{uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/igroups/8f249e7d-ab9f-
11e8-b8a3-005056bb7072/igroups/c439efc8-0a70-11eb-adc1-
0242ac120002?allow_delete_while_mapped=true' -H 'Accept:
application/hal+json'
```

#### Deleting an initiator group

```
# The API:
DELETE /api/protocols/san/igroups/{uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/igroups/abf9c39d-ab9f-
11e8-b8a3-005056bb7072' -H 'Accept: application/hal+json'
```

#### Deleting a mapped initiator group

Normally, deleting an initiator group that is mapped to a LUN is not allowed. The deletion can be forced using the `allow_delete_while_mapped` query parameter.

```
# The API:
DELETE /api/protocols/san/igroups/{uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/igroups/abf9c39d-ab9f-
11e8-b8a3-005056bb7072?allow_delete_while_mapped=true' -H 'Accept:
application/hal+json'
```

## Retrieve initiator groups

GET `/protocols/san/igroups`

**Introduced In:** 9.6

Retrieves initiator groups.

### 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.

- `connectivity_tracking.*`
- `igroups.*`
- `lun_maps.*`
- `parent_igroups.*`
- `target.*`

### Related ONTAP commands

- `lun igroup show`
- `lun mapping show`

### Learn more

- DOC /protocols/san/igroups

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| connectivity_tracking.connection_state | string | query | False | Filter by connectivity_tracking.connection_state<br>• Introduced in: 9.11 |
| connectivity_tracking.required_nodes.uuid | string | query | False | Filter by connectivity_tracking.required_nodes.uuid<br>• Introduced in: 9.11 |
| connectivity_tracking.required_nodes.name | string | query | False | Filter by connectivity_tracking.required_nodes.name<br>• Introduced in: 9.11 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| connectivity_tracking.alerts.summary.arguments.code | string | query | False | Filter by connectivity_tracking.alerts.summary.arguments.code<br>• Introduced in: 9.11 |
| connectivity_tracking.alerts.summary.arguments.message | string | query | False | Filter by connectivity_tracking.alerts.summary.arguments.message<br>• Introduced in: 9.11 |
| connectivity_tracking.alerts.summary.message | string | query | False | Filter by connectivity_tracking.alerts.summary.message<br>• Introduced in: 9.11 |
| connectivity_tracking.alerts.summary.code | string | query | False | Filter by connectivity_tracking.alerts.summary.code<br>• Introduced in: 9.11 |
| igroups.name | string | query | False | Filter by igroups.name<br>• Introduced in: 9.9<br>• maxLength: 96<br>• minLength: 1 |
| igroups.comment | string | query | False | Filter by igroups.comment<br>• Introduced in: 9.9<br>• maxLength: 254<br>• minLength: 0 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| igroups.uuid | string | query | False | Filter by igroups.uuid<br>• Introduced in: 9.9 |
| replication.peer_svm.uuid | string | query | False | Filter by replication.peer_svm.uuid<br>• Introduced in: 9.15 |
| replication.peer_svm.name | string | query | False | Filter by replication.peer_svm.name<br>• Introduced in: 9.15 |
| replication.state | string | query | False | Filter by replication.state<br>• Introduced in: 9.15 |
| replication.error.summary.arguments.code | string | query | False | Filter by replication.error.summary.arguments.code<br>• Introduced in: 9.15 |
| replication.error.summary.arguments.message | string | query | False | Filter by replication.error.summary.arguments.message<br>• Introduced in: 9.15 |
| replication.error.summary.message | string | query | False | Filter by replication.error.summary.message<br>• Introduced in: 9.15 |

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| replication.error.summary.code | string | query | False | Filter by replication.error.summary.code<br>• Introduced in: 9.15 |
| replication.error.igroup.uuid | string | query | False | Filter by replication.error.igroup.uuid<br>• Introduced in: 9.15 |
| replication.error.igroup.name | string | query | False | Filter by replication.error.igroup.name<br>• Introduced in: 9.15<br>• maxLength: 96<br>• minLength: 1 |
| replication.error.igroup.local_svm | boolean | query | False | Filter by replication.error.igroup.local_svm<br>• Introduced in: 9.15 |
| supports_igroups | boolean | query | False | Filter by supports_igroups<br>• Introduced in: 9.9 |
| parent_igroups.name | string | query | False | Filter by parent_igroups.name<br>• Introduced in: 9.9<br>• maxLength: 96<br>• minLength: 1 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| parent_igroups.uuid | string | query | False | Filter by parent_igroups.uuid<br>• Introduced in: 9.9 |
| parent_igroups.comment | string | query | False | Filter by parent_igroups.comment<br>• Introduced in: 9.9<br>• maxLength: 254<br>• minLength: 0 |
| os_type | string | query | False | Filter by os_type |
| name | string | query | False | Filter by name<br>• maxLength: 96<br>• minLength: 1 |
| target.firmware_revision | string | query | False | Filter by target.firmware_revision<br>• Introduced in: 9.11 |
| target.vendor_id | string | query | False | Filter by target.vendor_id<br>• Introduced in: 9.11 |
| target.product_id | string | query | False | Filter by target.product_id<br>• Introduced in: 9.11 |
| lun_maps.logical_unit_number | integer | query | False | Filter by lun_maps.logical_unit_number |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun_maps.lun.node.uuid | string | query | False | Filter by lun_maps.lun.node.uuid |
| lun_maps.lun.node.name | string | query | False | Filter by lun_maps.lun.node.name |
| lun_maps.lun.name | string | query | False | Filter by lun_maps.lun.name |
| lun_maps.lun.uuid | string | query | False | Filter by lun_maps.lun.uuid |
| comment | string | query | False | Filter by comment<br>• Introduced in: 9.9<br>• maxLength: 254<br>• minLength: 0 |
| protocol | string | query | False | Filter by protocol |
| delete_on_unmap | boolean | query | False | Filter by delete_on_unmap |
| svm.name | string | query | False | Filter by svm.name |
| svm.uuid | string | query | False | Filter by svm.uuid |
| portset.name | string | query | False | Filter by portset.name<br>• Introduced in: 9.9<br>• maxLength: 96<br>• minLength: 1 |
| portset.uuid | string | query | False | Filter by portset.uuid<br>• Introduced in: 9.9 |
| uuid | string | query | False | Filter by uuid |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| initiators.proximity.peer_svms.uuid | string | query | False | Filter by initiators.proximity.peer_svms.uuid<br>• Introduced in: 9.15 |
| initiators.proximity.peer_svms.name | string | query | False | Filter by initiators.proximity.peer_svms.name<br>• Introduced in: 9.15 |
| initiators.proximity.local_svm | boolean | query | False | Filter by initiators.proximity.local_svm<br>• Introduced in: 9.15 |
| initiators.igroup.uuid | string | query | False | Filter by initiators.igroup.uuid |
| initiators.igroup.name | string | query | False | Filter by initiators.igroup.name<br>• Introduced in: 9.9<br>• maxLength: 96<br>• minLength: 1 |
| initiators.connectivity_tracking.connection_state | string | query | False | Filter by initiators.connectivity_tracking.connection_state<br>• Introduced in: 9.11 |
| initiators.name | string | query | False | Filter by initiators.name<br>• maxLength: 96<br>• minLength: 1 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| initiators.comment | string | query | False | Filter by initiators.comment<br>• Introduced in: 9.9<br>• maxLength: 254<br>• minLength: 0 |
| 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.<br>• 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[igroup] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "comment": "string",
      "connectivity_tracking": {
        "alerts": [
          {
            "summary": {
              "arguments": [
                {
                  "code": "string",
                  "message": "string"
                }
              ],
              "code": "4",
              "message": "entry doesn't exist"
            }
          }
        ],
        "connection_state": "string",
        "required_nodes": [
          {
            "_links": {
              "self": {
                "href": "/api/resourcelink"
              }
            },
            "name": "node1",
            "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
          }
```

```
      ]
    },
    "igroups": [
      {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "comment": "string",
        "igroups": [
          null
        ],
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      }
    ],
    "initiators": [
      {
        "_links": {
          "connectivity_tracking": {
            "href": "/api/resourcelink"
          },
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "comment": "string",
        "connectivity_tracking": {
          "connection_state": "string"
        },
        "igroup": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "igroup1",
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        },
        "name": "iqn.1998-01.com.corp.iscsi:name1",
        "proximity": {
          "peer_svms": [
            {
              "_links": {
                "self": {
```

```
                "href": "/api/resourcelink"
              }
            },
            "name": "peer1",
            "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
          }
        ]
      }
    }
  ],
  "lun_maps": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "logical_unit_number": 0,
      "lun": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "lun1",
        "node": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "node1",
          "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
        },
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      }
    }
  ],
  "name": "igroup1",
  "os_type": "string",
  "parent_igroups": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
```

```
      },
      "comment": "string",
      "name": "igroup1",
      "parent_igroups": [
        null
      ],
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "portset": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "portset1",
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "protocol": "string",
  "replication": {
    "error": {
      "igroup": {
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "summary": {
        "arguments": [
          {
            "code": "string",
            "message": "string"
          }
        ],
        "code": "4",
        "message": "entry doesn't exist"
      }
    },
    "peer_svm": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "peer1",
      "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
    },
    "state": "string"
```

```
      },
      "svm": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "target": {
        "firmware_revision": "9111",
        "product_id": "LUN C-Mode",
        "vendor_id": "NETAPP"
      },
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

summary

A user friendly message describing the connection state of the initiator group.

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

alerts

| Name | Type | Description |
|---|---|---|
| summary | summary | A user friendly message describing the connection state of the initiator group. |

required_nodes

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | |
| uuid | string | |

connectivity_tracking

An overview of the connections to ONTAP by the initiators in this group.

| Name | Type | Description |
|---|---|---|
| alerts | array[alerts] | |
| connection_state | string | Connection state. |
| required_nodes | array[required_nodes] | Nodes to which the initiators in this group should be connected to ensure reliable service. This is the collection of any node hosting a LUN mapped to this igroup as well as the HA partners of those nodes. |

igroup_child

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| comment | string | A comment available for use by the administrator. |
| igroups | array[igroup_child] | Further nested initiator groups. |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

connectivity_tracking

A link to the initiator with connectivity information relevant to its membership of this initiator group.

| Name | Type | Description |
|---|---|---|
| href | string | |

self

A link to the initiator where mutations can be made. If the initiator is inherited from a nested initiator group,

the link refers to the initiator in the nested initiator group. In this case, mutations of the initiator will be applied to all initiator groups referencing the same nested initiator group.

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

| Name | Type | Description |
|---|---|---|
| connectivity_tracking | connectivity_tracking | A link to the initiator with connectivity information relevant to its membership of this initiator group. |
| self | self | A link to the initiator where mutations can be made. If the initiator is inherited from a nested initiator group, the link refers to the initiator in the nested initiator group. In this case, mutations of the initiator will be applied to all initiator groups referencing the same nested initiator group. |

connectivity_tracking

Overview of the initiator's connections to ONTAP.

| Name | Type | Description |
|---|---|---|
| connection_state | string | Connection state. |

igroup

The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

peer_svms

A reference to an SVM peer relationship.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

proximity

Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.

These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`.

| Name | Type | Description |
|---|---|---|
| local_svm | boolean | A boolean that indicates if the initiator is proximal to the SVM of the containing initiator group. This is required for any POST or PATCH that includes the `proximity` sub-object. |
| peer_svms | array[peer_svms] | An array of remote peer SVMs to which the initiator is proximal. |

initiators

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| connectivity_tracking | connectivity_tracking | Overview of the initiator's connections to ONTAP.<br>• readOnly: 1<br>• Introduced in: 9.11 |
| igroup | igroup | The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group. |
| name | string | The FC WWPN, iSCSI IQN, or iSCSI EUI that identifies the host initiator. Valid in POST only and not allowed when the `records` property is used.<br>An FC WWPN consists of 16 hexadecimal digits grouped as 8 pairs separated by colons. The format for an iSCSI IQN is *iqn.yyyy-mm.reverse_domain_name:any*. The iSCSI EUI format consists of the *eui.* prefix followed by 16 hexadecimal characters. |

| Name | Type | Description |
| --- | --- | --- |
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |

node

| Name | Type | Description |
| --- | --- | --- |
| _links | _links | |
| name | string | |
| uuid | string | |

lun

The LUN to which the initiator group is mapped.

| Name | Type | Description |
| --- | --- | --- |
| _links | _links | |
| name | string | The name of the LUN. |
| node | node | |
| uuid | string | The unique identifier of the LUN. |

lun_maps

A LUN map with which the initiator group is associated.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| logical_unit_number | integer | The logical unit number assigned to the LUN for initiators in the initiator group. |
| lun | lun | The LUN to which the initiator group is mapped. |

igroup_parent

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| comment | string | A comment available for use by the administrator. |
| name | string | The name of the initiator group. |
| parent_igroups | array[igroup_parent] | The initiator groups that contain this initiator group as a member. |
| uuid | string | The unique identifier of the initiator group. |

portset

The portset to which the initiator group is bound. Binding the initiator group to a portset restricts the initiators of the group to accessing mapped LUNs only through network interfaces in the portset.

In a nested initiator group hierarchy, only a portset bound to the initiator group at the same level at which it is mapped, applies; portsets bound to parent or child initiator groups are ignored.

Optional in POST and PATCH. To unbind a portset from the initiator group, PATCH the `portset` object to `null`, or PATCH `portset.name` to an empty string ("").

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the portset. |
| uuid | string | The unique identifier of the portset. |

igroup

| Name | Type | Description |
|---|---|---|
| local_svm | boolean | Indicates whether the reported igroup is on the local SVM or the peer SVM. When deleting a replicated igroup, the local copy is deleted first and then the peer copy is deleted. If the error is encountered between these two operations and only the peer igroup remains, the peer igroup is reported and the problem might need to be corrected on the peer cluster. |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

summary

A user friendly message describing the error.

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

error

Information about asynchronous errors encountered while replicating this igroup. Igroups within a peering relationship are replicated in the same stream, so the error reported here might be related to this igroup or a prior replicated igroup that is now blocking the replication of this igroup. Both the error information and the igroup encountering the error are reported. If the error is configuration related, it can be corrected on the referenced igroup. The replication is retried using exponential backoff up to a maximum of one retry every 5 minutes. Every operation on the same stream triggers an immediate retry and restarts the exponential backoff starting with a 1 second delay. If the error is system related, the retries should correct the error when the system enters a healthy state.

| Name | Type | Description |
|---|---|---|
| igroup | igroup | |
| summary | summary | A user friendly message describing the error. |

peer_svm

The peered SVM to which the initiator group should be replicated. Optional in POST and PATCH. To clear any previously set replication peer, PATCH the `replication.peer_svm` object to `null`.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

replication

Properties related to initiator group replication.

| Name | Type | Description |
|---|---|---|
| error | error | Information about asynchronous errors encountered while replicating this igroup. Igroups within a peering relationship are replicated in the same stream, so the error reported here might be related to this igroup or a prior replicated igroup that is now blocking the replication of this igroup. Both the error information and the igroup encountering the error are reported. If the error is configuration related, it can be corrected on the referenced igroup. The replication is retried using exponential backoff up to a maximum of one retry every 5 minutes. Every operation on the same stream triggers an immediate retry and restarts the exponential backoff starting with a 1 second delay. If the error is system related, the retries should correct the error when the system enters a healthy state. |

| Name | Type | Description |
|---|---|---|
| peer_svm | peer_svm | The peered SVM to which the initiator group should be replicated. Optional in POST and PATCH. To clear any previously set replication peer, PATCH the `replication.peer_svm` object to `null`. |
| state | string | The state of the replication queue associated with this igroup. If this igroup is not in the replication queue, the state is reported as *ok*. If this igroup is in the replication queue, but no errors have been encountered, the state is reported as *replicating*. If this igroup is in the replication queue and the queue is blocked by an error, the state is reported as *error*. When in the *error* state, additional context is provided by the `replication.error` property. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

target

Properties of the SCSI target to which the initiator group provides access.

| Name | Type | Description |
|---|---|---|
| firmware_revision | string | The firmware revision of the SCSI target specific to the OS type of the initiator group. |

| Name | Type | Description |
|---|---|---|
| product_id | string | The product ID of the SCSI target. |
| vendor_id | string | The vendor ID of the SCSI target. |

igroup

An initiator group (igroup) is a collection of Fibre Channel (FC) world wide port names (WWPNs), and/or iSCSI Qualified Names (IQNs), and/or iSCSI EUIs (Extended Unique Identifiers) that identify host endpoints.

Initiator groups are used to control which hosts can access specific LUNs. To grant access to a LUN from one or more hosts, create an initiator group containing the host initiator names, then create a LUN map that associates the initiator group with the LUN.

An initiator group may contain either initiators or other initiator groups, but not both simultaneously. When a parent initiator group is mapped, it inherits all of the initiators of any initiator groups nested below it. If any nested initiator group is modified to contain different initiators, the parent initiator groups inherit the change. A parent can have many nested initiator groups and an initiator group can be nested under multiple parents. Initiators can only be added or removed from the initiator group that directly contains them. The maximum supported depth of nesting is three layers.

Best practice when using nested initiator groups is to match host hierarchies. A single initiator group should correspond to a single host. If a LUN needs to be mapped to multiple hosts, the initiator groups representing those hosts should be aggregated into a parent initiator group and the LUN should be mapped to that initiator group. For multi-ported hosts, initiators have a comment property where the port corresponding to the initiator can be documented.

An initiator can appear in multiple initiator groups. An initiator group can be mapped to multiple LUNs. A specific initiator can be mapped to a specific LUN only once. With the introduction of nestable initiator groups, best practice is to use the hierarchy such that an initiator is only a direct member of a single initiator group, and that initiator group can then be referenced by other initiator groups.

All initiators or nested initiator groups in an initiator group must be from the same operating system. The initiator group's operating system is specified when the initiator group is created.

When an initiator group is created, the `protocol` property is used to restrict member initiators to Fibre Channel (*fcp*), iSCSI (*iscsi*), or both (*mixed*). Initiator groups within a nested hierarchy may not have conflicting protocols.

Zero or more initiators or nested initiator groups can be supplied when the initiator group is created. After creation, initiators can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/initiators` endpoint. Initiator groups containing other initiator groups report the aggregated list of initiators from all nested initiator groups, but modifications of the initiator list must be performed on the initiator group that directly contains the initiators. See DELETE /protocols/san/igroups/{igroup.uuid}/initiators/{name} for more details.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
| --- | --- | --- |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | An overview of the connections to ONTAP by the initiators in this group.<br>• readOnly: 1<br>• Introduced in: 9.11 |
| delete_on_unmap | boolean | An option that causes the initiator group to be deleted when the last LUN map associated with it is deleted. Optional in POST and PATCH. This property defaults to *false* when the initiator group is created. |

| Name | Type | Description |
| --- | --- | --- |
| igroups | array[igroup_child] | The existing initiator groups that are members of the group. Optional in POST.<br><br>This property is mutually exclusive with the *initiators* property during POST.<br><br>This array contains only the direct children of the initiator group. If the member initiator groups have further nested initiator groups, those are reported in the `igroups` property of the child initiator group.<br><br>Zero or more nested initiator groups can be supplied when the initiator group is created. The initiator group will act as if it contains the aggregation of all initiators in any nested initiator groups.<br><br>After creation, nested initiator groups can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/igroups` endpoint. See DELETE /protocols/san/igroups/{igroup.uuid}/igroups/{uuid} for more details. |

| Name | Type | Description |
| --- | --- | --- |
| initiators | array[initiators] | The initiators that are members of the group or any group nested below this group. Optional in POST.<br><br>This property is mutually exclusive with the *igroups* property during POST.<br><br>During GET, this array contains initiators that are members of this group or any nested initiator groups below this group. When initiators of nested groups are returned, they include links to the initiator group that directly contains the initiator.<br><br>Zero or more initiators can be supplied when the initiator group is created. After creation, initiators can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/initiators` endpoint. See DELETE /protocols/san/igroups/{igroup.uuid}/initiators/{name} for more details. |
| lun_maps | array[lun_maps] | All LUN maps with which the initiator is associated.<br><br>If the requested igroup is part of a remote, non-local, MetroCluster SVM, the LUN maps are not retrieved.<br><br>There is an added computational cost to retrieving property values for `lun_maps`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| name | string | The name of the initiator group. Required in POST; optional in PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| os_type | string | The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. Required in POST; optional in PATCH. |
| parent_igroups | array[igroup_parent] | The initiator groups that contain this initiator group as a member. |
| portset | portset | The portset to which the initiator group is bound. Binding the initiator group to a portset restricts the initiators of the group to accessing mapped LUNs only through network interfaces in the portset.<br><br>In a nested initiator group hierarchy, only a portset bound to the initiator group at the same level at which it is mapped, applies; portsets bound to parent or child initiator groups are ignored.<br><br>Optional in POST and PATCH. To unbind a portset from the initiator group, PATCH the `portset` object to `null`, or PATCH `portset.name` to an empty string (""). |
| protocol | string | The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. Optional in POST; if not supplied, this defaults to *mixed*.<br><br>The protocol of an initiator group cannot be changed after creation of the group. |
| replication | replication | Properties related to initiator group replication. |

| Name | Type | Description |
|---|---|---|
| supports_igroups | boolean | An initiator group may contain either initiators or other initiator groups, but not both simultaneously. This property is *true* when initiator groups can be added to this initiator group. The `initiators.name` property cannot be used to determine this via a query because it reports initiators inherited from nested igroups. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target | target | Properties of the SCSI target to which the initiator group provides access. |
| uuid | string | The unique identifier of the initiator group. |

returned_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 initiator group

POST `/protocols/san/igroups`

**Introduced In:** 9.6

Creates an initiator group.

### Required properties

- `svm.uuid` or `svm.name` - Existing SVM in which to create the initiator group.

- `name` - Name of the initiator group.
- `os_type` - Operating system of the initiator group's initiators.

### Recommended optional properties

- `initiators.name` - Name(s) of initiator group's initiators. This property can be used to create the initiator group and populate it with initiators in a single request.

### Default property values

If not specified in POST, the following default property values are assigned.

- `protocol` - *mixed* - Data protocol of the initiator group's initiators.

### Related ONTAP commands

- `lun igroup create`

### Learn more

- DOC /protocols/san/igroups

### Parameters

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

### Request Body

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | An overview of the connections to ONTAP by the initiators in this group.<br>• readOnly: 1<br>• Introduced in: 9.11 |

| Name | Type | Description |
| --- | --- | --- |
| delete_on_unmap | boolean | An option that causes the initiator group to be deleted when the last LUN map associated with it is deleted. Optional in POST and PATCH. This property defaults to *false* when the initiator group is created. |
| igroups | array[igroup_child] | The existing initiator groups that are members of the group. Optional in POST.<br><br>This property is mutually exclusive with the *initiators* property during POST.<br><br>This array contains only the direct children of the initiator group. If the member initiator groups have further nested initiator groups, those are reported in the `igroups` property of the child initiator group.<br><br>Zero or more nested initiator groups can be supplied when the initiator group is created. The initiator group will act as if it contains the aggregation of all initiators in any nested initiator groups.<br><br>After creation, nested initiator groups can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/igroups` endpoint. See DELETE /protocols/san/igroups/{igroup.uuid}/igroups/{uuid} for more details. |

| Name | Type | Description |
| --- | --- | --- |
| initiators | array[initiators] | The initiators that are members of the group or any group nested below this group. Optional in POST.<br><br>This property is mutually exclusive with the *igroups* property during POST.<br><br>During GET, this array contains initiators that are members of this group or any nested initiator groups below this group. When initiators of nested groups are returned, they include links to the initiator group that directly contains the initiator.<br><br>Zero or more initiators can be supplied when the initiator group is created. After creation, initiators can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/initiators` endpoint. See DELETE /protocols/san/igroups/{igroup.uuid}/initiators/{name} for more details. |
| lun_maps | array[lun_maps] | All LUN maps with which the initiator is associated.<br><br>If the requested igroup is part of a remote, non-local, MetroCluster SVM, the LUN maps are not retrieved.<br><br>There is an added computational cost to retrieving property values for `lun_maps`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| name | string | The name of the initiator group. Required in POST; optional in PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| os_type | string | The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. Required in POST; optional in PATCH. |
| parent_igroups | array[igroup_parent] | The initiator groups that contain this initiator group as a member. |
| portset | portset | The portset to which the initiator group is bound. Binding the initiator group to a portset restricts the initiators of the group to accessing mapped LUNs only through network interfaces in the portset.<br><br>In a nested initiator group hierarchy, only a portset bound to the initiator group at the same level at which it is mapped, applies; portsets bound to parent or child initiator groups are ignored.<br><br>Optional in POST and PATCH. To unbind a portset from the initiator group, PATCH the `portset` object to `null`, or PATCH `portset.name` to an empty string (""). |
| protocol | string | The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. Optional in POST; if not supplied, this defaults to *mixed*.<br><br>The protocol of an initiator group cannot be changed after creation of the group. |
| replication | replication | Properties related to initiator group replication. |

| Name | Type | Description |
| --- | --- | --- |
| supports_igroups | boolean | An initiator group may contain either initiators or other initiator groups, but not both simultaneously. This property is *true* when initiator groups can be added to this initiator group. The `initiators.name` property cannot be used to determine this via a query because it reports initiators inherited from nested igroups. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target | target | Properties of the SCSI target to which the initiator group provides access. |
| uuid | string | The unique identifier of the initiator group. |

**Example request**


```
{
  "comment": "string",
  "connectivity_tracking": {
    "alerts": [
      {}
    ],
    "connection_state": "string",
    "required_nodes": [
      {
        "name": "node1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      }
    ]
  },
  "igroups": [
    {
      "comment": "string",
      "igroups": [
        null
      ],
      "name": "igroup1",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "initiators": [
    {
      "comment": "string",
      "connectivity_tracking": {
        "connection_state": "string"
      },
      "igroup": {
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "name": "iqn.1998-01.com.corp.iscsi:name1",
      "proximity": {
        "peer_svms": [
          {
            "name": "peer1",
            "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
          }
        ]
      }
    }
```

```
  ],
  "lun_maps": [
    {
      "logical_unit_number": 0,
      "lun": {
        "name": "lun1",
        "node": {
          "name": "node1",
          "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
        },
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      }
    }
  ],
  "name": "igroup1",
  "os_type": "string",
  "parent_igroups": [
    {
      "comment": "string",
      "name": "igroup1",
      "parent_igroups": [
        null
      ],
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "portset": {
    "name": "portset1",
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "protocol": "string",
  "replication": {
    "error": {
      "igroup": {
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      }
    },
    "peer_svm": {
      "name": "peer1",
      "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
    },
    "state": "string"
  },
  "svm": {
    "name": "svm1",
```

```
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "target": {
    "firmware_revision": "9111",
    "product_id": "LUN C-Mode",
    "vendor_id": "NETAPP"
  },
  "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
}
```


### Response

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| num_records | integer | The number of records in the response. |
| records | array[igroup] | |

**Example response**


```
{
  "num_records": 1,
  "records": [
    {
      "comment": "string",
      "connectivity_tracking": {
        "alerts": [
          {}
        ],
        "connection_state": "string",
        "required_nodes": [
          {
            "name": "node1",
            "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
          }
        ]
      },
      "igroups": [
        {
          "comment": "string",
          "igroups": [
            null
          ],
          "name": "igroup1",
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        }
      ],
      "initiators": [
        {
          "comment": "string",
          "connectivity_tracking": {
            "connection_state": "string"
          },
          "igroup": {
            "name": "igroup1",
            "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
          },
          "name": "iqn.1998-01.com.corp.iscsi:name1",
          "proximity": {
            "peer_svms": [
              {
                "name": "peer1",
                "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
              }
```

```
        ]
      }
    }
  ],
  "lun_maps": [
    {
      "logical_unit_number": 0,
      "lun": {
        "name": "lun1",
        "node": {
          "name": "node1",
          "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
        },
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      }
    }
  ],
  "name": "igroup1",
  "os_type": "string",
  "parent_igroups": [
    {
      "comment": "string",
      "name": "igroup1",
      "parent_igroups": [
        null
      ],
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "portset": {
    "name": "portset1",
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "protocol": "string",
  "replication": {
    "error": {
      "igroup": {
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      }
    },
    "peer_svm": {
      "name": "peer1",
      "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
    },
    "state": "string"
```

```
      },
      "svm": {
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "target": {
        "firmware_revision": "9111",
        "product_id": "LUN C-Mode",
        "vendor_id": "NETAPP"
      },
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ]
}
```


#### Headers

| Name | Description | Type |
|---|---|---|
| Location | Useful for tracking the resource location | string |

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | The supplied SVM does not exist. |
| 2621706 | The specified `svm.uuid` and `svm.name` do not refer to the same SVM. |
| 2621707 | No SVM was specified. Either `svm.name` or `svm.uuid` must be supplied. |
| 5373958 | An invalid initiator group name was supplied. |
| 5373966 | An initiator group cannot be created in an SVM that is configured for NVMe. |
| 5373969 | A supplied initiator name looks like an iSCSI IQN initiator, but the portions after the prefix are missing. |
| 5373971 | A supplied initiator name looks like an iSCSI IQN initiator, but the date portion is invalid. |
| 5373972 | A supplied initiator name looks like an iSCSI IQN initiator, but the naming authority portion is invalid. |

| Error Code | Description |
| --- | --- |
| 5373977 | A supplied initiator name looks like an iSCSI EUI initiator, but the length is invalid. |
| 5373978 | A supplied initiator name looks like an iSCSI EUI initiator, but the format is invalid. |
| 5373982 | An invalid WWN was specified. The length is incorrect. |
| 5373983 | An invalid WWN was specified. The format is incorrect. |
| 5373992 | A supplied initiator name was too long to be valid. |
| 5373993 | A supplied initiator name did not match any valid format. |
| 5374023 | An initiator group with the same name already exists. |
| 5374027 | An attempt was made to bind a portset with no member network interfaces to the initiator group. |
| 5374028 | An attempt was made to bind a portset with an incompatible protocol to the initiator group. |
| 5374038 | An invalid Fibre Channel WWPN was supplied. |
| 5374039 | An invalid iSCSI initiator name was supplied. |
| 5374040 | Initiators and child initiator groups were both supplied, but only one option is allowed. |
| 5374732 | An initiator is already in another initiator group with a conflicting operating system type. |
| 5374735 | An attempt was made to add a child igroup that would exceed the maximum allowable depth. |
| 5374737 | A supplied child initiator group already exists in another initiator group’s hierarchy. |
| 5374739 | A supplied child initiator group has an operating system type that differs from the parent initiator group. |
| 5374740 | A supplied child initiator group has an protocol that differs from the parent initiator group. |
| 5374741 | A supplied child initiator group is already owned by a different child in the initiator group’s hierarchy. |
| 5374742 | A supplied child initiator group contains an initiator that is already owned by another initiator group in the hierarchy. |
| 5374745 | Initiator group cannot be added as a child to itself. |
| 5374746 | The cluster is currently running in a mixed version and nested initiator groups cannot be created until the effective cluster version reaches 9.9.1. |

| Error Code | Description |
| --- | --- |
| 5374747 | The cluster is currently running in a mixed version and initiator group comments cannot be created until the effective cluster version reaches 9.9.1. |
| 5374758 | An error was reported by the peer cluster while creating a replicated initiator group. The specific error will be included as a nested error. |
| 5374878 | The supplied child initiator group does not exist. |
| 5374911 | The supplied portset does not exist. |
| 5374917 | Duplicated initiators have conflicting property values. |
| 5375055 | The `local_svm` property of an initiator proximity was not specified. |
| 5375056 | An SVM peering relationship that does not have the initiator group's SVM as the local SVM was specified. |
| 5375258 | The igroup is already replicated to a different peer SVM. |
| 5375261 | Setting initiator proximity is not supported for the SVM type. |
| 5376057 | Setting initiator proximity is not supported for the ONTAP version. |
| 5376059 | Setting initiator proximity to a peer that is either the destination of an SVM DR relationship or in a Metrocluster configuration is not supported. |
| 5376253 | Initiator group replication requires an effective cluster version of 9.15.1. |
| 5376255 | Initiator group replication requires the peer cluster to have an effective cluster version of 9.15.1. |
| 5376488 | An NVMe over Fabrics subsystem already exists with the requested name. |
| 6620376 | SVM peering information is unavailable. |
| 6620384 | The supplied SVMs are not peered. |
| 26345672 | The specified SVM peering relationship was not found. |
| 26345673 | An SVM peering relationship between the initiator group's SVM and specified peer SVM was not found. |
| 26345675 | An SVM peering relationship UUID and name were specified and they do not refer to the same SVM peering relationship. |
| 26345680 | Supplied SVM peer is on the local cluster. The operation requires a peer on a remote cluster. |

Also see the table of common errors in the Response body overview section of this documentation.

## Definitions

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

summary

A user friendly message describing the connection state of the initiator group.

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

alerts

required_nodes

| Name | Type | Description |
|---|---|---|
| name | string | |
| uuid | string | |

connectivity_tracking

An overview of the connections to ONTAP by the initiators in this group.

| Name | Type | Description |
|---|---|---|
| alerts | array[alerts] | |
| connection_state | string | Connection state. |

| Name | Type | Description |
|---|---|---|
| required_nodes | array[required_nodes] | Nodes to which the initiators in this group should be connected to ensure reliable service. This is the collection of any node hosting a LUN mapped to this igroup as well as the HA partners of those nodes. |

igroup_child

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. |
| igroups | array[igroup_child] | Further nested initiator groups. |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

connectivity_tracking

A link to the initiator with connectivity information relevant to its membership of this initiator group.

| Name | Type | Description |
|---|---|---|
| href | string | |

self

A link to the initiator where mutations can be made. If the initiator is inherited from a nested initiator group, the link refers to the initiator in the nested initiator group. In this case, mutations of the initiator will be applied to all initiator groups referencing the same nested initiator group.

| Name | Type | Description |
|---|---|---|
| href | string | |

connectivity_tracking

Overview of the initiator's connections to ONTAP.

| Name | Type | Description |
|---|---|---|
| connection_state | string | Connection state. |

igroup

The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

peer_svms

A reference to an SVM peer relationship.

| Name | Type | Description |
|---|---|---|
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

proximity

Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.

These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`.

| Name | Type | Description |
|---|---|---|
| local_svm | boolean | A boolean that indicates if the initiator is proximal to the SVM of the containing initiator group. This is required for any POST or PATCH that includes the `proximity` sub-object. |
| peer_svms | array[peer_svms] | An array of remote peer SVMs to which the initiator is proximal. |

initiators

| Name | Type | Description |
| --- | --- | --- |
| _links | _links | |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | Overview of the initiator's connections to ONTAP.<br>• readOnly: 1<br>• Introduced in: 9.11 |
| igroup | igroup | The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group. |
| name | string | The FC WWPN, iSCSI IQN, or iSCSI EUI that identifies the host initiator. Valid in POST only and not allowed when the `records` property is used.<br>An FC WWPN consists of 16 hexadecimal digits grouped as 8 pairs separated by colons. The format for an iSCSI IQN is *iqn.yyyy-mm.reverse_domain_name:any*. The iSCSI EUI format consists of the *eui.* prefix followed by 16 hexadecimal characters. |

| Name | Type | Description |
|---|---|---|
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |

node

| Name | Type | Description |
|---|---|---|
| name | string | |
| uuid | string | |

lun

The LUN to which the initiator group is mapped.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the LUN. |
| node | node | |
| uuid | string | The unique identifier of the LUN. |

lun_maps

A LUN map with which the initiator group is associated.

| Name | Type | Description |
|---|---|---|
| logical_unit_number | integer | The logical unit number assigned to the LUN for initiators in the initiator group. |
| lun | lun | The LUN to which the initiator group is mapped. |

igroup_parent

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. |
| name | string | The name of the initiator group. |
| parent_igroups | array[igroup_parent] | The initiator groups that contain this initiator group as a member. |
| uuid | string | The unique identifier of the initiator group. |

portset

The portset to which the initiator group is bound. Binding the initiator group to a portset restricts the initiators of the group to accessing mapped LUNs only through network interfaces in the portset.

In a nested initiator group hierarchy, only a portset bound to the initiator group at the same level at which it is mapped, applies; portsets bound to parent or child initiator groups are ignored.

Optional in POST and PATCH. To unbind a portset from the initiator group, PATCH the `portset` object to `null`, or PATCH `portset.name` to an empty string ("").

| Name | Type | Description |
|---|---|---|
| name | string | The name of the portset. |
| uuid | string | The unique identifier of the portset. |

igroup

| Name | Type | Description |
|---|---|---|
| local_svm | boolean | Indicates whether the reported igroup is on the local SVM or the peer SVM. When deleting a replicated igroup, the local copy is deleted first and then the peer copy is deleted. If the error is encountered between these two operations and only the peer igroup remains, the peer igroup is reported and the problem might need to be corrected on the peer cluster. |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

summary

A user friendly message describing the error.

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

error

Information about asynchronous errors encountered while replicating this igroup. Igroups within a peering relationship are replicated in the same stream, so the error reported here might be related to this igroup or a prior replicated igroup that is now blocking the replication of this igroup. Both the error information and the igroup encountering the error are reported. If the error is configuration related, it can be corrected on the referenced igroup. The replication is retried using exponential backoff up to a maximum of one retry every 5 minutes. Every operation on the same stream triggers an immediate retry and restarts the exponential backoff starting with a 1 second delay. If the error is system related, the retries should correct the error when the system enters a healthy state.

| Name | Type | Description |
|---|---|---|
| igroup | igroup | |

peer_svm

The peered SVM to which the initiator group should be replicated. Optional in POST and PATCH. To clear any previously set replication peer, PATCH the `replication.peer_svm` object to `null`.

| Name | Type | Description |
| --- | --- | --- |
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

replication

Properties related to initiator group replication.

| Name | Type | Description |
| --- | --- | --- |
| error | error | Information about asynchronous errors encountered while replicating this igroup. Igroups within a peering relationship are replicated in the same stream, so the error reported here might be related to this igroup or a prior replicated igroup that is now blocking the replication of this igroup. Both the error information and the igroup encountering the error are reported. If the error is configuration related, it can be corrected on the referenced igroup. The replication is retried using exponential backoff up to a maximum of one retry every 5 minutes. Every operation on the same stream triggers an immediate retry and restarts the exponential backoff starting with a 1 second delay. If the error is system related, the retries should correct the error when the system enters a healthy state. |
| peer_svm | peer_svm | The peered SVM to which the initiator group should be replicated. Optional in POST and PATCH. To clear any previously set replication peer, PATCH the `replication.peer_svm` object to `null`. |

| Name | Type | Description |
|---|---|---|
| state | string | The state of the replication queue associated with this igroup. If this igroup is not in the replication queue, the state is reported as *ok*. If this igroup is in the replication queue, but no errors have been encountered, the state is reported as *replicating*. If this igroup is in the replication queue and the queue is blocked by an error, the state is reported as *error*. When in the *error* state, additional context is provided by the `replication.error` property. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

target

Properties of the SCSI target to which the initiator group provides access.

| Name | Type | Description |
|---|---|---|
| firmware_revision | string | The firmware revision of the SCSI target specific to the OS type of the initiator group. |
| product_id | string | The product ID of the SCSI target. |
| vendor_id | string | The vendor ID of the SCSI target. |

igroup

An initiator group (igroup) is a collection of Fibre Channel (FC) world wide port names (WWPNs), and/or iSCSI Qualified Names (IQNs), and/or iSCSI EUIs (Extended Unique Identifiers) that identify host

endpoints.

Initiator groups are used to control which hosts can access specific LUNs. To grant access to a LUN from one or more hosts, create an initiator group containing the host initiator names, then create a LUN map that associates the initiator group with the LUN.

An initiator group may contain either initiators or other initiator groups, but not both simultaneously. When a parent initiator group is mapped, it inherits all of the initiators of any initiator groups nested below it. If any nested initiator group is modified to contain different initiators, the parent initiator groups inherit the change. A parent can have many nested initiator groups and an initiator group can be nested under multiple parents. Initiators can only be added or removed from the initiator group that directly contains them. The maximum supported depth of nesting is three layers.

Best practice when using nested initiator groups is to match host hierarchies. A single initiator group should correspond to a single host. If a LUN needs to be mapped to multiple hosts, the initiator groups representing those hosts should be aggregated into a parent initiator group and the LUN should be mapped to that initiator group. For multi-ported hosts, initiators have a comment property where the port corresponding to the initiator can be documented.

An initiator can appear in multiple initiator groups. An initiator group can be mapped to multiple LUNs. A specific initiator can be mapped to a specific LUN only once. With the introduction of nestable initiator groups, best practice is to use the hierarchy such that an initiator is only a direct member of a single initiator group, and that initiator group can then be referenced by other initiator groups.

All initiators or nested initiator groups in an initiator group must be from the same operating system. The initiator group's operating system is specified when the initiator group is created.

When an initiator group is created, the `protocol` property is used to restrict member initiators to Fibre Channel (*fcp*), iSCSI (*iscsi*), or both (*mixed*). Initiator groups within a nested hierarchy may not have conflicting protocols.

Zero or more initiators or nested initiator groups can be supplied when the initiator group is created. After creation, initiators can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/initiators` endpoint. Initiator groups containing other initiator groups report the aggregated list of initiators from all nested initiator groups, but modifications of the initiator list must be performed on the initiator group that directly contains the initiators. See DELETE /protocols/san/igroups/{igroup.uuid}/initiators/{name} for more details.

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | An overview of the connections to ONTAP by the initiators in this group.<br>• readOnly: 1<br>• Introduced in: 9.11 |

| Name | Type | Description |
| --- | --- | --- |
| delete_on_unmap | boolean | An option that causes the initiator group to be deleted when the last LUN map associated with it is deleted. Optional in POST and PATCH. This property defaults to *false* when the initiator group is created. |
| igroups | array[igroup_child] | The existing initiator groups that are members of the group. Optional in POST.<br><br>This property is mutually exclusive with the *initiators* property during POST.<br><br>This array contains only the direct children of the initiator group. If the member initiator groups have further nested initiator groups, those are reported in the `igroups` property of the child initiator group.<br><br>Zero or more nested initiator groups can be supplied when the initiator group is created. The initiator group will act as if it contains the aggregation of all initiators in any nested initiator groups.<br><br>After creation, nested initiator groups can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/igroups` endpoint. See DELETE /protocols/san/igroups/{igroup.uuid}/igroups/{uuid} for more details. |

| Name | Type | Description |
| --- | --- | --- |
| initiators | array[initiators] | The initiators that are members of the group or any group nested below this group. Optional in POST.<br><br>This property is mutually exclusive with the *igroups* property during POST.<br><br>During GET, this array contains initiators that are members of this group or any nested initiator groups below this group. When initiators of nested groups are returned, they include links to the initiator group that directly contains the initiator.<br><br>Zero or more initiators can be supplied when the initiator group is created. After creation, initiators can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/initiators` endpoint. See DELETE /protocols/san/igroups/{igroup.uuid}/initiators/{name} for more details. |
| lun_maps | array[lun_maps] | All LUN maps with which the initiator is associated.<br><br>If the requested igroup is part of a remote, non-local, MetroCluster SVM, the LUN maps are not retrieved.<br><br>There is an added computational cost to retrieving property values for `lun_maps`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| name | string | The name of the initiator group. Required in POST; optional in PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| os_type | string | The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. Required in POST; optional in PATCH. |
| parent_igroups | array[igroup_parent] | The initiator groups that contain this initiator group as a member. |
| portset | portset | The portset to which the initiator group is bound. Binding the initiator group to a portset restricts the initiators of the group to accessing mapped LUNs only through network interfaces in the portset.<br><br>In a nested initiator group hierarchy, only a portset bound to the initiator group at the same level at which it is mapped, applies; portsets bound to parent or child initiator groups are ignored.<br><br>Optional in POST and PATCH. To unbind a portset from the initiator group, PATCH the `portset` object to `null`, or PATCH `portset.name` to an empty string (""). |
| protocol | string | The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. Optional in POST; if not supplied, this defaults to *mixed*.<br><br>The protocol of an initiator group cannot be changed after creation of the group. |
| replication | replication | Properties related to initiator group replication. |

| Name | Type | Description |
|---|---|---|
| supports_igroups | boolean | An initiator group may contain either initiators or other initiator groups, but not both simultaneously. This property is *true* when initiator groups can be added to this initiator group. The `initiators.name` property cannot be used to determine this via a query because it reports initiators inherited from nested igroups. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target | target | Properties of the SCSI target to which the initiator group provides access. |
| uuid | string | The unique identifier of the initiator group. |

returned_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 nested initiator groups of an initiator group

GET `/protocols/san/igroups/{igroup.uuid}/igroups`

**Introduced In:** 9.9

Retrieves nested initiator groups of an initiator group.

This API only reports the nested initiator groups that are direct children of the initiator group. Further nested initiator groups are reported by their direct parent initiator group.

### Related ONTAP commands

- `lun igroup show`

### Learn more

- DOC /protocols/san/igroups

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| igroup.uuid | string | path | True | The unique identifier of the parent initiator group. |
| 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.<br>• 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[igroup_nested] | |

### Example response


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "igroup": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "name": "igroup1",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ]
}
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374852 | The initiator group specified in the URI does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

igroup

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| uuid | string | The unique identifier of the parent initiator group. |

records

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

igroup_nested

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| igroup | igroup | |
| name | string | The name of the initiator group. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the initiator group. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 nested initiator groups to an initiator group

POST `/protocols/san/igroups/{igroup.uuid}/igroups`

**Introduced In:** 9.9

Adds one or more nested initiator groups to an initiator group. A single nested initiator group can be added by directly specifying the name or UUID. Multiple nested initiator groups can be added by specifying the names or UUIDs in the records array. Nested initiator groups cannot be added to an initiator group that already directly contains initiators.

### Required properties

- `name` and/or `uuid` or `records` - Nested initiator groups to add to the initiator group.

### Related ONTAP commands

- `lun igroup add`

### Learn more

- DOC /protocols/san/igroups

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| igroup.uuid | string | path | True | The unique identifier of the parent initiator group. |
| return_records | boolean | query | False | The default is false. If set to true, the records are returned.<br>• Default value: |

### Request Body

| Name | Type | Description |
|---|---|---|
| igroup | igroup | |
| name | string | The name of the initiator group. |
| records | array[records] | An array of initiator groups specified to add multiple nested initiator groups to an initiator group in a single API call. Not allowed when the `name` property is used. |
| uuid | string | The unique identifier of the initiator group. |

### Example request

```
{
  "igroup": {
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "name": "igroup1",
  "records": [
    {
      "name": "igroup1",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
}
```

### Response

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| num_records | integer | The number of records in the response. |
| records | array[igroup_nested] | |

### Example response


```
{
  "num_records": 1,
  "records": [
    {
      "igroup": {
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "name": "igroup1",
      "records": [
        {
          "name": "igroup1",
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        }
      ],
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ]
}
```


#### Headers

| Name | Description | Type |
|---|---|---|
| Location | Useful for tracking the resource location | string |

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
| --- | --- |
| 5374735 | An attempt was made to add a child igroup that would exceed the maximum allowable depth. |
| 5374736 | A supplied child initiator group already exists in the parent initiator group's hierarchy. |
| 5374737 | A supplied child initiator group already exists in another initiator group's hierarchy. |
| 5374739 | A supplied child initiator group has an operating system type that differs from the parent initiator group. |
| 5374740 | A supplied child initiator group has an protocol that differs from the parent initiator group. |
| 5374741 | A supplied child initiator group is already owned by a different child in the initiator group's hierarchy. |
| 5374742 | A supplied child initiator group contains an initiator that is already owned by another initiator group in the hierarchy. |
| 5374764 | An unreplicated initiator group cannot be added as a child of a replicated initiator group. |
| 5374852 | The initiator group specified in the URI does not exist. |
| 5374878 | The supplied child initiator group does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

igroup

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the parent initiator group. |

records

| Name | Type | Description |
|---|---|---|
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

igroup_nested

| Name | Type | Description |
|---|---|---|
| igroup | igroup | |
| name | string | The name of the initiator group. |
| records | array[records] | An array of initiator groups specified to add multiple nested initiator groups to an initiator group in a single API call. Not allowed when the `name` property is used. |
| uuid | string | The unique identifier of the initiator group. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |

| Name | Type | Description |
|---|---|---|
| message | string | Message argument |

returned_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. |

## Remove a nested initiator group from an initiator group

DELETE `/protocols/san/igroups/{igroup.uuid}/igroups/{uuid}`

**Introduced In:** 9.9

Removes a nested initiator group from an initiator group. This API does not delete the nested initiator group itself. It removes the relationship between a parent and child initiator group.

This API only supports removal of initiator groups owned directly by the initiator group. Further nested initiator groups must be removed from the direct parent initiator group.

### Related ONTAP commands

- `lun igroup remove`

### Learn more

- DOC /protocols/san/igroups

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| igroup.uuid | string | path | True | The unique identifier of the parent initiator group. |
| uuid | string | path | True | The unique identifier of the nested initiator group. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| allow_delete_while_mapped | boolean | query | False | Allows the deletion of a nested initiator group from of a mapped initiator group.<br><br>Deleting a nested initiator group from a mapped initiator group means that the LUNs, to which the initiator group is mapped, are no longer available to the initiators nested below the initiator group being removed. This might cause a disruption in the availability of data.<br><br>**This parameter should be used with caution.**<br><br>• Default value: |

**Response**

```
Status: 200, Ok
```

**Error**

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 1254213 | The initiator group is mapped to one or more LUNs and `allow_delete_while_mapped` has not been specified. |
| 5374738 | The child initiator group is not owned by the parent initiator group. |
| 5374743 | LUN maps exist for a parent initiator group. |

| Error Code | Description |
| --- | --- |
| 5374852 | The initiator group specified in the URI does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
| --- | --- | --- |
| error | returned_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 |

returned_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 a nested initiator group of an initiator group

GET `/protocols/san/igroups/{igroup.uuid}/igroups/{uuid}`

**Introduced In:** 9.9

Retrieves a nested initiator group of an initiator group.

This API only reports the nested initiator groups that are direct children of the initiator group. Further nested initiator groups are reported by their direct parent initiator group.

### Related ONTAP commands

- `lun igroup show`

### Learn more

- DOC /protocols/san/igroups

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| igroup.uuid | string | path | True | The unique identifier of the parent initiator group. |
| uuid | string | path | True | The unique identifier of the nested initiator group. |
| fields | array[string] | query | False | Specify the fields to return. |

**Response**

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| igroup | igroup | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

### Example response


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "igroup": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "name": "igroup1",
  "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 4 | The nested initiator group is not a member of the initiator group. |
| 5374852 | The parent initiator group specified in the URI does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

igroup

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| uuid | string | The unique identifier of the parent initiator group. |

records

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_error

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

| Name | Type | Description |
|---|---|---|
| message | string | Error message |
| target | string | The target parameter that caused the error. |

## Retrieve initiators of an initiator group

GET `/protocols/san/igroups/{igroup.uuid}/initiators`

**Introduced In:** 9.6

Retrieves initiators of an initiator group.

This API only reports initiators owned directly by the initiator group. Initiators of nested initiator groups are not included in this collection.

### 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.

- `connectivity_tracking.*`

### Related ONTAP commands

- `lun igroup show`

### Learn more

- DOC /protocols/san/igroups

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| igroup.uuid | string | path | True | The unique identifier of the initiator group. |
| fields | array[string] | query | False | Specify the fields to return. |
| max_records | integer | query | False | Limit the number of records returned. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| return_records | boolean | query | False | The default is true for GET calls. When set to false, only the number of records is returned.<br>• 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[igroup_initiator] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "comment": "string",
      "connectivity_tracking": {
        "alerts": [
          {
            "summary": {
              "arguments": [
                {
                  "code": "string",
                  "message": "string"
                }
              ],
              "code": "4",
              "message": "entry doesn't exist"
            }
          }
        ],
        "connection_state": "string",
        "connections": [
          {
            "logins": [
              {
                "interface": {
                  "fc": {
                    "_links": {
                      "self": {
                        "href": "/api/resourcelink"
                      }
```

```
                    },
                    "name": "fc_lif1",
                    "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
                    "wwpn": "20:00:00:50:56:b4:13:a8"
                  },
                  "ip": {
                    "_links": {
                      "self": {
                        "href": "/api/resourcelink"
                      }
                    },
                    "ip": {
                      "address": "10.10.10.7"
                    },
                    "name": "lif1",
                    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
                  }
                },
                "last_seen_time": "2021-03-14 00:19:00 -0500"
              }
            ],
            "node": {
              "_links": {
                "self": {
                  "href": "/api/resourcelink"
                }
              },
              "name": "node1",
              "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
            }
          }
        ]
      },
      "igroup": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "name": "iqn.1998-01.com.corp.iscsi:name1",
      "proximity": {
        "peer_svms": [
          {
```

```
            "_links": {
              "self": {
                "href": "/api/resourcelink"
              }
            },
            "name": "peer1",
            "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
          }
        ]
      }
    }
  ]
}
```


**Error**

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374852 | The initiator group specified in the URI does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

summary

A user friendly message describing the connection state.

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

alerts

| Name | Type | Description |
|---|---|---|
| summary | summary | A user friendly message describing the connection state. |

fc

An FC interface.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the FC interface. |
| uuid | string | The unique identifier of the FC interface. |
| wwpn | string | The WWPN of the FC interface. |

ip

IP information

| Name | Type | Description |
|---|---|---|
| address | string | IPv4 or IPv6 address |

ip

A network interface. Either UUID or name may be supplied on input.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| ip | ip | IP information |
| name | string | The name of the interface. If only the name is provided, the SVM scope must be provided by the object this object is embedded in. |
| uuid | string | The UUID that uniquely identifies the interface. |

interface

| Name | Type | Description |
|---|---|---|
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |

logins

| Name | Type | Description |
|---|---|---|
| connected | boolean | True if the initiator is currently logged in to this connection's interface. |
| interface | interface | |
| last_seen_time | string | The last time this initiator logged in. Logins not seen for 48 hours are cleared and not reported. |

node

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | |
| uuid | string | |

connections

| Name | Type | Description |
|---|---|---|
| logins | array[logins] | |
| node | node | |

connectivity_tracking

Overview of the initiator's connections to ONTAP.

| Name | Type | Description |
|---|---|---|
| alerts | array[alerts] | |
| connection_state | string | Connection state. |
| connections | array[connections] | |

igroup

The initiator group in which the initiator is found.

Note that this does not mean that the initiator cannot also be found in other initiator groups.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the initiator group. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the initiator group. |

peer_svms

A reference to an SVM peer relationship.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

proximity

Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.

These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`.

| Name | Type | Description |
|---|---|---|
| local_svm | boolean | A boolean that indicates if the initiator is proximal to the SVM of the containing initiator group. This is required for any POST or PATCH that includes the `proximity` sub-object. |
| peer_svms | array[peer_svms] | An array of remote peer SVMs to which the initiator is proximal. |

records

| Name | Type | Description |
| --- | --- | --- |
| _links | _links | |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| name | string | The FC WWPN, iSCSI IQN, or iSCSI EUI that identifies the host initiator. Valid in POST only and not allowed when the `records` property is used.<br><br>An FC WWPN consists of 16 hexadecimal digits grouped as 8 pairs separated by colons. The format for an iSCSI IQN is *iqn.yyyy-mm.reverse_domain_name:any*. The iSCSI EUI format consists of the *eui.* prefix followed by 16 hexadecimal characters. |
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |

igroup_initiator

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | Overview of the initiator's connections to ONTAP.<br>• readOnly: 1<br>• Introduced in: 9.11 |
| igroup | igroup | The initiator group in which the initiator is found.<br><br>Note that this does not mean that the initiator cannot also be found in other initiator groups. |
| name | string | The FC WWPN, iSCSI IQN, or iSCSI EUI that identifies the host initiator. Valid in POST only and not allowed when the `records` property is used.<br><br>An FC WWPN consists of 16 hexadecimal digits grouped as 8 pairs separated by colons. The format for an iSCSI IQN is *iqn.yyyy-mm.reverse_domain_name:any*. The iSCSI EUI format consists of the *eui.* prefix followed by 16 hexadecimal characters. |

| Name | Type | Description |
|---|---|---|
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |

returned_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 initiators to an initiator group

POST `/protocols/san/igroups/{igroup.uuid}/initiators`

**Introduced In:** 9.6

Adds one or more initiators to an initiator group.

This API does not support adding initiators to an initiator group that already contains nested initiator groups.

### Required properties

- `name` or `records.name` - Initiator name(s) to add to the initiator group.

### Related ONTAP commands

- `lun igroup add`

### Learn more

- DOC /protocols/san/igroups

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| igroup.uuid | string | path | True | The unique identifier of the initiator group. |
| return_records | boolean | query | False | The default is false. If set to true, the records are returned.<br>• Default value: |

### Request Body

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | Overview of the initiator's connections to ONTAP.<br>• readOnly: 1<br>• Introduced in: 9.11 |
| igroup | igroup | The initiator group in which the initiator is found.<br>Note that this does not mean that the initiator cannot also be found in other initiator groups. |

| Name | Type | Description |
| --- | --- | --- |
| name | string | The FC WWPN, iSCSI IQN, or iSCSI EUI that identifies the host initiator. Valid in POST only and not allowed when the `records` property is used.<br><br>An FC WWPN consists of 16 hexadecimal digits grouped as 8 pairs separated by colons. The format for an iSCSI IQN is *iqn.yyyy-mm.reverse_domain_name:any*. The iSCSI EUI format consists of the *eui.* prefix followed by 16 hexadecimal characters. |
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |
| records | array[records] | An array of initiators specified to add multiple initiators to an initiator group in a single API call. Not allowed when the `name` property is used. |

**Example request**


```
{
  "comment": "string",
  "connectivity_tracking": {
    "alerts": [
      {}
    ],
    "connection_state": "string",
    "connections": [
      {
        "logins": [
          {
            "interface": {
              "fc": {
                "name": "fc_lif1",
                "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
                "wwpn": "20:00:00:50:56:b4:13:a8"
              },
              "ip": {
                "name": "lif1",
                "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
              }
            },
            "last_seen_time": "2021-03-14 00:19:00 -0500"
          }
        ],
        "node": {
          "name": "node1",
          "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
        }
      }
    ]
  },
  "igroup": {
    "name": "igroup1",
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "name": "iqn.1998-01.com.corp.iscsi:name1",
  "proximity": {
    "peer_svms": [
      {
        "name": "peer1",
        "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
      }
    ]
```

```
  },
  "records": [
    {
      "comment": "string",
      "name": "iqn.1998-01.com.corp.iscsi:name1",
      "proximity": {
        "peer_svms": [
          {
            "name": "peer1",
            "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
          }
        ]
      }
    }
  ]
}
```

**Response**

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| num_records | integer | The number of records in the response. |
| records | array[igroup_initiator] | |

**Example response**


```
{
  "num_records": 1,
  "records": [
    {
      "comment": "string",
      "connectivity_tracking": {
        "alerts": [
          {}
        ],
        "connection_state": "string",
        "connections": [
          {
            "logins": [
              {
                "interface": {
                  "fc": {
                    "name": "fc_lif1",
                    "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
                    "wwpn": "20:00:00:50:56:b4:13:a8"
                  },
                  "ip": {
                    "name": "lif1",
                    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
                  }
                },
                "last_seen_time": "2021-03-14 00:19:00 -0500"
              }
            ],
            "node": {
              "name": "node1",
              "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
            }
          }
        ]
      },
      "igroup": {
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "name": "iqn.1998-01.com.corp.iscsi:name1",
      "proximity": {
        "peer_svms": [
          {
            "name": "peer1",
```

```
            "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
          }
        ]
      },
      "records": [
        {
          "comment": "string",
          "name": "iqn.1998-01.com.corp.iscsi:name1",
          "proximity": {
            "peer_svms": [
              {
                "name": "peer1",
                "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
              }
            ]
          }
        }
      ]
    }
  ]
}
```


#### Headers

| Name | Description | Type |
|---|---|---|
| Location | Useful for tracking the resource location | string |

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 1254193 | Adding an initiator would cause the initiator to be mapped to the same LUN more than once. |
| 1254324 | Adding an initiator would cause the initiator to have the same logical unit identifier for multiple LUN maps. |
| 5373969 | A supplied initiator name looks like an iSCSI IQN initiator, but the portions after the prefix are missing. |
| 5373971 | A supplied initiator name looks like an iSCSI IQN initiator, but the date portion is invalid. |

| Error Code | Description |
| --- | --- |
| 5373972 | A supplied initiator name looks like an iSCSI IQN initiator, but the naming authority portion is invalid. |
| 5373977 | A supplied initiator name looks like an iSCSI EUI initiator, but the length is invalid. |
| 5373978 | A supplied initiator name looks like an iSCSI EUI initiator, but the format is invalid. |
| 5373982 | An invalid WWN was specified. The length is incorrect. |
| 5373983 | An invalid WWN was specified. The format is incorrect. |
| 5373992 | A supplied initiator name was too long to be valid. |
| 5373993 | A supplied initiator name did not match any valid format. |
| 5374033 | Initiators must be supplied. |
| 5374035 | A supplied initiator is already in the initiator group. |
| 5374038 | An invalid Fibre Channel WWPN was supplied. |
| 5374039 | An invalid iSCSI initiator name was supplied. |
| 5374734 | An initiator is already in another initiator group with a conflicting operating system type. |
| 5374761 | An error was reported by the peer cluster while adding an initiator to a replicated initiator group. The specific error will be included as a nested error. |
| 5374852 | The initiator group specified in the URI does not exist. |
| 5374917 | Duplicated initiators have conflicting property values. |
| 5375055 | The `local_svm` property of an initiator proximity was not specified. |
| 5375056 | An SVM peering relationship that does not have the initiator group's SVM as the local SVM was specified. |
| 5375261 | Setting initiator proximity is not supported for the SVM type. |
| 5376057 | Setting initiator proximity is not supported for the ONTAP version. |
| 5376059 | Setting initiator proximity to a peer that is either the destination of an SVM DR relationship or in a Metrocluster configuration is not supported. |
| 26345672 | The specified SVM peering relationship was not found. |
| 26345673 | An SVM peering relationship between the initiator group's SVM and specified peer SVM was not found. |

| Error Code | Description |
| --- | --- |
| 26345675 | An SVM peering relationship UUID and name were specified and they do not refer to the same SVM peering relationship. |
| 26345680 | Supplied SVM peer is on the local cluster. The operation requires a peer on a remote cluster. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

summary

A user friendly message describing the connection state.

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

alerts

fc

An FC interface.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the FC interface. |
| uuid | string | The unique identifier of the FC interface. |
| wwpn | string | The WWPN of the FC interface. |

ip

IP information

ip

A network interface. Either UUID or name may be supplied on input.

| Name | Type | Description |
|---|---|---|
| ip | ip | IP information |
| name | string | The name of the interface. If only the name is provided, the SVM scope must be provided by the object this object is embedded in. |
| uuid | string | The UUID that uniquely identifies the interface. |

interface

| Name | Type | Description |
|---|---|---|
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |

logins

| Name | Type | Description |
|---|---|---|
| connected | boolean | True if the initiator is currently logged in to this connection's interface. |
| interface | interface | |
| last_seen_time | string | The last time this initiator logged in. Logins not seen for 48 hours are cleared and not reported. |

node

| Name | Type | Description |
|---|---|---|
| name | string | |
| uuid | string | |

connections

| Name | Type | Description |
|---|---|---|
| logins | array[logins] | |

| Name | Type | Description |
|---|---|---|
| node | node | |

connectivity_tracking

Overview of the initiator's connections to ONTAP.

| Name | Type | Description |
|---|---|---|
| alerts | array[alerts] | |
| connection_state | string | Connection state. |
| connections | array[connections] | |

igroup

The initiator group in which the initiator is found.

Note that this does not mean that the initiator cannot also be found in other initiator groups.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

peer_svms

A reference to an SVM peer relationship.

| Name | Type | Description |
|---|---|---|
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

proximity

Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.

These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST

and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`.

| Name | Type | Description |
|---|---|---|
| local_svm | boolean | A boolean that indicates if the initiator is proximal to the SVM of the containing initiator group. This is required for any POST or PATCH that includes the `proximity` sub-object. |
| peer_svms | array[peer_svms] | An array of remote peer SVMs to which the initiator is proximal. |

records

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| name | string | The FC WWPN, iSCSI IQN, or iSCSI EUI that identifies the host initiator. Valid in POST only and not allowed when the `records` property is used.<br><br>An FC WWPN consists of 16 hexadecimal digits grouped as 8 pairs separated by colons. The format for an iSCSI IQN is *iqn.yyyy-mm.reverse_domain_name:any*. The iSCSI EUI format consists of the *eui.* prefix followed by 16 hexadecimal characters. |

| Name | Type | Description |
| --- | --- | --- |
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |

igroup_initiator

| Name | Type | Description |
| --- | --- | --- |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | Overview of the initiator's connections to ONTAP.<br><br>• readOnly: 1<br>• Introduced in: 9.11 |
| igroup | igroup | The initiator group in which the initiator is found.<br><br>Note that this does not mean that the initiator cannot also be found in other initiator groups. |

| Name | Type | Description |
| --- | --- | --- |
| name | string | The FC WWPN, iSCSI IQN, or iSCSI EUI that identifies the host initiator. Valid in POST only and not allowed when the `records` property is used.<br><br>An FC WWPN consists of 16 hexadecimal digits grouped as 8 pairs separated by colons. The format for an iSCSI IQN is *iqn.yyyy-mm.reverse_domain_name:any*. The iSCSI EUI format consists of the *eui.* prefix followed by 16 hexadecimal characters. |
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |
| records | array[records] | An array of initiators specified to add multiple initiators to an initiator group in a single API call. Not allowed when the `name` property is used. |

returned_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 initiator from an initiator group

DELETE `/protocols/san/igroups/{igroup.uuid}/initiators/{name}`

**Introduced In:** 9.6

Deletes an initiator from an initiator group.

This API only supports removal of initiators owned directly by the initiator group. Initiators of nested initiator groups must be removed on the initiator group that directly owns the initiator.

### Related ONTAP commands

- `lun igroup remove`

### Learn more

- DOC /protocols/san/igroups

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| igroup.uuid | string | path | True | The unique identifier of the initiator group. |
| name | string | path | True | The initiator name. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| allow_delete_while_mapped | boolean | query | False | Allows the deletion of an initiator from of a mapped initiator group.<br><br>Deleting an initiator from a mapped initiator group makes the LUNs to which the initiator group is mapped no longer available to the initiator. This might cause a disruption in the availability of data.<br><br>**This parameter should be used with caution.**<br><br>• Default value: |

**Response**

```
Status: 200, Ok
```

**Error**

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 1254213 | The initiator group is mapped to one or more LUNs and `allow_delete_while_mapped` has not been specified. |
| 5374034 | An initiator is not a member of the initiator group. |
| 5374041 | The initiator is not owned by the supplied initiator group. |
| 5374762 | An error was reported by the peer cluster while removing an initiator from a replicated initiator group. The specific error will be included as a nested error. |
| 5374852 | The initiator group specified in the URI does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
| --- | --- | --- |
| error | returned_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 |

returned_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 initiator

GET `/protocols/san/igroups/{igroup.uuid}/initiators/{name}`

**Introduced In:** 9.6

Retrieves an initiator of an initiator group.

This API only reports initiators owned directly by the initiator group. Initiators of nested initiator groups are not part of this collection.

### 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.

- `connectivity_tracking.*`

### Related ONTAP commands

- `lun igroup show`

### Learn more

- DOC /protocols/san/igroups

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| igroup.uuid | string | path | True | The unique identifier of the initiator group. |
| name | string | path | True | The initiator name. |
| fields | array[string] | query | False | Specify the fields to return. |

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | Overview of the initiator's connections to ONTAP.<br>• readOnly: 1<br>• Introduced in: 9.11 |
| igroup | igroup | The initiator group in which the initiator is found.<br>Note that this does not mean that the initiator cannot also be found in other initiator groups. |

| Name | Type | Description |
| --- | --- | --- |
| name | string | The FC WWPN, iSCSI IQN, or iSCSI EUI that identifies the host initiator. Valid in POST only and not allowed when the `records` property is used.<br><br>An FC WWPN consists of 16 hexadecimal digits grouped as 8 pairs separated by colons. The format for an iSCSI IQN is *iqn.yyyy-mm.reverse_domain_name:any*. The iSCSI EUI format consists of the *eui.* prefix followed by 16 hexadecimal characters. |
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |

**Example response**


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "comment": "string",
  "connectivity_tracking": {
    "alerts": [
      {
        "summary": {
          "arguments": [
            {
              "code": "string",
              "message": "string"
            }
          ],
          "code": "4",
          "message": "entry doesn't exist"
        }
      }
    ],
    "connection_state": "string",
    "connections": [
      {
        "logins": [
          {
            "interface": {
              "fc": {
                "_links": {
                  "self": {
                    "href": "/api/resourcelink"
                  }
                },
                "name": "fc_lif1",
                "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
                "wwpn": "20:00:00:50:56:b4:13:a8"
              },
              "ip": {
                "_links": {
                  "self": {
                    "href": "/api/resourcelink"
                  }
                },
```

```
                "ip": {
                  "address": "10.10.10.7"
                },
                "name": "lif1",
                "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
              }
            },
            "last_seen_time": "2021-03-14 00:19:00 -0500"
          }
        ],
        "node": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "node1",
          "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
        }
      }
    ]
  },
  "igroup": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "igroup1",
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "name": "iqn.1998-01.com.corp.iscsi:name1",
  "proximity": {
    "peer_svms": [
      {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "peer1",
        "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
      }
    ]
  }
}
```

**Error**

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 4 | The initiator is not a member of the initiator group. |
| 5374852 | The initiator group specified in the URI does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

summary

A user friendly message describing the connection state.

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

alerts

| Name | Type | Description |
|---|---|---|
| summary | summary | A user friendly message describing the connection state. |

fc

An FC interface.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the FC interface. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the FC interface. |
| wwpn | string | The WWPN of the FC interface. |

ip

IP information

| Name | Type | Description |
|---|---|---|
| address | string | IPv4 or IPv6 address |

ip

A network interface. Either UUID or name may be supplied on input.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| ip | ip | IP information |
| name | string | The name of the interface. If only the name is provided, the SVM scope must be provided by the object this object is embedded in. |
| uuid | string | The UUID that uniquely identifies the interface. |

interface

| Name | Type | Description |
|---|---|---|
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |

logins

| Name | Type | Description |
|---|---|---|
| connected | boolean | True if the initiator is currently logged in to this connection's interface. |
| interface | interface | |
| last_seen_time | string | The last time this initiator logged in. Logins not seen for 48 hours are cleared and not reported. |

node

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | |
| uuid | string | |

connections

| Name | Type | Description |
|---|---|---|
| logins | array[logins] | |
| node | node | |

connectivity_tracking

Overview of the initiator's connections to ONTAP.

| Name | Type | Description |
|---|---|---|
| alerts | array[alerts] | |
| connection_state | string | Connection state. |
| connections | array[connections] | |

igroup

The initiator group in which the initiator is found.

Note that this does not mean that the initiator cannot also be found in other initiator groups.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the initiator group. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the initiator group. |

peer_svms

A reference to an SVM peer relationship.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

proximity

Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.

These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`.

| Name | Type | Description |
|---|---|---|
| local_svm | boolean | A boolean that indicates if the initiator is proximal to the SVM of the containing initiator group. This is required for any POST or PATCH that includes the `proximity` sub-object. |
| peer_svms | array[peer_svms] | An array of remote peer SVMs to which the initiator is proximal. |

records

| Name | Type | Description |
| --- | --- | --- |
| _links | _links | |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| name | string | The FC WWPN, iSCSI IQN, or iSCSI EUI that identifies the host initiator. Valid in POST only and not allowed when the `records` property is used.<br><br>An FC WWPN consists of 16 hexadecimal digits grouped as 8 pairs separated by colons. The format for an iSCSI IQN is *iqn.yyyy-mm.reverse_domain_name:any*. The iSCSI EUI format consists of the *eui.* prefix followed by 16 hexadecimal characters. |
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |

returned_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 initiator

PATCH `/protocols/san/igroups/{igroup.uuid}/initiators/{name}`

**Introduced In:** 9.9

Updates an initiator of an initiator group.

This API only supports modification of initiators owned directly by the initiator group. Initiators of nested initiator groups must be modified on the initiator group that directly owns the initiator.

### Related ONTAP commands

- `lun igroup initiator modify`

### Learn more

- DOC /protocols/san/igroups

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| igroup.uuid | string | path | True | The unique identifier of the initiator group. |
| name | string | path | True | The initiator name. |

### Request Body

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| connectivity_tracking | connectivity_tracking | Overview of the initiator's connections to ONTAP.<br>• readOnly: 1<br>• Introduced in: 9.11 |
| igroup | igroup | The initiator group in which the initiator is found.<br><br>Note that this does not mean that the initiator cannot also be found in other initiator groups. |
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |
| records | array[records] | An array of initiators specified to add multiple initiators to an initiator group in a single API call. Not allowed when the `name` property is used. |

**Example request**


```
{
  "comment": "string",
  "connectivity_tracking": {
    "alerts": [
      {}
    ],
    "connection_state": "string",
    "connections": [
      {
        "logins": [
          {
            "interface": {
              "fc": {
                "name": "fc_lif1",
                "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
                "wwpn": "20:00:00:50:56:b4:13:a8"
              },
              "ip": {
                "name": "lif1",
                "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
              }
            },
            "last_seen_time": "2021-03-14 00:19:00 -0500"
          }
        ],
        "node": {
          "name": "node1",
          "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
        }
      }
    ]
  },
  "igroup": {
    "name": "igroup1",
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "proximity": {
    "peer_svms": [
      {
        "name": "peer1",
        "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
      }
    ]
  },
```

```
  "records": [
    {
      "comment": "string",
      "proximity": {
        "peer_svms": [
          {
            "name": "peer1",
            "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
          }
        ]
      }
    }
  ]
}
```

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374034 | An initiator is not a member of the initiator group. |
| 5374744 | The cluster is currently running in a mixed version and the initiators cannot be modified until the effective cluster version reaches 9.9.1. |
| 5374852 | The initiator group does not exist. |
| 5374918 | A subset of the provided list of initiators were modified before a failure occurred. |
| 5375055 | The `local_svm` property of an initiator proximity was not specified. |
| 5375056 | An SVM peering relationship that does not have the initiator group's SVM as the local SVM was specified. |
| 5375261 | Setting initiator proximity is not supported for the SVM type. |
| 5376057 | Setting initiator proximity is not supported for the ONTAP version. |

| Error Code | Description |
| --- | --- |
| 5376059 | Setting initiator proximity to a peer that is either the destination of an SVM DR relationship or in a Metrocluster configuration is not supported. |
| 26345672 | The specified SVM peering relationship was not found. |
| 26345673 | An SVM peering relationship between the initiator group's SVM and specified peer SVM was not found. |
| 26345675 | An SVM peering relationship UUID and name were specified and they do not refer to the same SVM peering relationship. |
| 26345680 | Supplied SVM peer is on the local cluster. The operation requires a peer on a remote cluster. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

summary

A user friendly message describing the connection state.

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

alerts

fc

An FC interface.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the FC interface. |
| uuid | string | The unique identifier of the FC interface. |
| wwpn | string | The WWPN of the FC interface. |

ip

IP information

ip

A network interface. Either UUID or name may be supplied on input.

| Name | Type | Description |
|---|---|---|
| ip | ip | IP information |
| name | string | The name of the interface. If only the name is provided, the SVM scope must be provided by the object this object is embedded in. |
| uuid | string | The UUID that uniquely identifies the interface. |

interface

| Name | Type | Description |
|---|---|---|
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |

logins

| Name | Type | Description |
|---|---|---|
| connected | boolean | True if the initiator is currently logged in to this connection's interface. |
| interface | interface | |
| last_seen_time | string | The last time this initiator logged in. Logins not seen for 48 hours are cleared and not reported. |

node

| Name | Type | Description |
|---|---|---|
| name | string | |
| uuid | string | |

connections

| Name | Type | Description |
|---|---|---|
| logins | array[logins] | |

| Name | Type | Description |
|---|---|---|
| node | node | |

connectivity_tracking

Overview of the initiator's connections to ONTAP.

| Name | Type | Description |
|---|---|---|
| alerts | array[alerts] | |
| connection_state | string | Connection state. |
| connections | array[connections] | |

igroup

The initiator group in which the initiator is found.

Note that this does not mean that the initiator cannot also be found in other initiator groups.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

peer_svms

A reference to an SVM peer relationship.

| Name | Type | Description |
|---|---|---|
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

proximity

Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.

These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST

and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`.

| Name | Type | Description |
|---|---|---|
| local_svm | boolean | A boolean that indicates if the initiator is proximal to the SVM of the containing initiator group. This is required for any POST or PATCH that includes the `proximity` sub-object. |
| peer_svms | array[peer_svms] | An array of remote peer SVMs to which the initiator is proximal. |

records

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |

igroup_initiator

| Name | Type | Description |
| --- | --- | --- |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | Overview of the initiator's connections to ONTAP.<br><br>• readOnly: 1<br>• Introduced in: 9.11 |
| igroup | igroup | The initiator group in which the initiator is found.<br><br>Note that this does not mean that the initiator cannot also be found in other initiator groups. |

| Name | Type | Description |
| --- | --- | --- |
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |
| records | array[records] | An array of initiators specified to add multiple initiators to an initiator group in a single API call. Not allowed when the `name` property is used. |

returned_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 initiator group

DELETE `/protocols/san/igroups/{uuid}`

**Introduced In:** 9.6

Deletes an initiator group.

### Related ONTAP commands

- `lun igroup delete`

### Learn more

- DOC /protocols/san/igroups

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| uuid | string | path | True | The unique identifier of the initiator group. |
| allow_delete_while_mapped | boolean | query | False | Allows the deletion of a mapped initiator group.<br><br>Deleting a mapped initiator group means that the LUNs, to which the initiator group is mapped, are no longer available to the initiators. This might cause a disruption in the availability of data.<br><br>**This parameter should be used with caution.**<br><br>• Default value: |

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 1254213 | The initiator group is mapped to one or more LUNs and `allow_delete_while_mapped` has not been specified. |
| 5374760 | An error was reported by the peer cluster while deleting a replicated initiator group. The specific error will be included as a nested error. |
| 5374852 | The initiator group does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 |

returned_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 initiator group

GET `/protocols/san/igroups/{uuid}`

**Introduced In:** 9.6

Retrieves an initiator group.

### 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.

- `connectivity_tracking.*`
- `igroups.*`
- `lun_maps.*`
- `parent_igroups.*`

### Related ONTAP commands

- `lun igroup show`
- `lun mapping show`

### Learn more

- DOC /protocols/san/igroups

### Parameters

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

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | An overview of the connections to ONTAP by the initiators in this group.<br>• readOnly: 1<br>• Introduced in: 9.11 |
| delete_on_unmap | boolean | An option that causes the initiator group to be deleted when the last LUN map associated with it is deleted. Optional in POST and PATCH. This property defaults to *false* when the initiator group is created. |

| Name | Type | Description |
| --- | --- | --- |
| igroups | array[igroup_child] | The existing initiator groups that are members of the group. Optional in POST.<br><br>This property is mutually exclusive with the *initiators* property during POST.<br><br>This array contains only the direct children of the initiator group. If the member initiator groups have further nested initiator groups, those are reported in the `igroups` property of the child initiator group.<br><br>Zero or more nested initiator groups can be supplied when the initiator group is created. The initiator group will act as if it contains the aggregation of all initiators in any nested initiator groups.<br><br>After creation, nested initiator groups can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/igroups` endpoint. See DELETE /protocols/san/igroups/{igroup.uuid}/igroups/{uuid} for more details. |

| Name | Type | Description |
| --- | --- | --- |
| initiators | array[initiators] | The initiators that are members of the group or any group nested below this group. Optional in POST.<br><br>This property is mutually exclusive with the *igroups* property during POST.<br><br>During GET, this array contains initiators that are members of this group or any nested initiator groups below this group. When initiators of nested groups are returned, they include links to the initiator group that directly contains the initiator.<br><br>Zero or more initiators can be supplied when the initiator group is created. After creation, initiators can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/initiators` endpoint. See DELETE /protocols/san/igroups/{igroup.uuid}/initiators/{name} for more details. |
| lun_maps | array[lun_maps] | All LUN maps with which the initiator is associated.<br><br>If the requested igroup is part of a remote, non-local, MetroCluster SVM, the LUN maps are not retrieved.<br><br>There is an added computational cost to retrieving property values for `lun_maps`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| name | string | The name of the initiator group. Required in POST; optional in PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| os_type | string | The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. Required in POST; optional in PATCH. |
| parent_igroups | array[igroup_parent] | The initiator groups that contain this initiator group as a member. |
| portset | portset | The portset to which the initiator group is bound. Binding the initiator group to a portset restricts the initiators of the group to accessing mapped LUNs only through network interfaces in the portset.<br><br>In a nested initiator group hierarchy, only a portset bound to the initiator group at the same level at which it is mapped, applies; portsets bound to parent or child initiator groups are ignored.<br><br>Optional in POST and PATCH. To unbind a portset from the initiator group, PATCH the `portset` object to `null`, or PATCH `portset.name` to an empty string (""). |
| protocol | string | The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. Optional in POST; if not supplied, this defaults to *mixed*.<br><br>The protocol of an initiator group cannot be changed after creation of the group. |
| replication | replication | Properties related to initiator group replication. |

| Name | Type | Description |
| --- | --- | --- |
| supports_igroups | boolean | An initiator group may contain either initiators or other initiator groups, but not both simultaneously. This property is *true* when initiator groups can be added to this initiator group. The `initiators.name` property cannot be used to determine this via a query because it reports initiators inherited from nested igroups. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target | target | Properties of the SCSI target to which the initiator group provides access. |
| uuid | string | The unique identifier of the initiator group. |

**Example response**


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "comment": "string",
  "connectivity_tracking": {
    "alerts": [
      {
        "summary": {
          "arguments": [
            {
              "code": "string",
              "message": "string"
            }
          ],
          "code": "4",
          "message": "entry doesn't exist"
        }
      }
    ],
    "connection_state": "string",
    "required_nodes": [
      {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "node1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      }
    ]
  },
  "igroups": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "comment": "string",
      "igroups": [
```

```
        null
      ],
      "name": "igroup1",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "initiators": [
    {
      "_links": {
        "connectivity_tracking": {
          "href": "/api/resourcelink"
        },
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "comment": "string",
      "connectivity_tracking": {
        "connection_state": "string"
      },
      "igroup": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "name": "iqn.1998-01.com.corp.iscsi:name1",
      "proximity": {
        "peer_svms": [
          {
            "_links": {
              "self": {
                "href": "/api/resourcelink"
              }
            },
            "name": "peer1",
            "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
          }
        ]
      }
    }
  ],
  "lun_maps": [
```

```
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "logical_unit_number": 0,
      "lun": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "lun1",
        "node": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "node1",
          "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
        },
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      }
    }
  ],
  "name": "igroup1",
  "os_type": "string",
  "parent_igroups": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "comment": "string",
      "name": "igroup1",
      "parent_igroups": [
        null
      ],
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "portset": {
    "_links": {
```

```
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "portset1",
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "protocol": "string",
  "replication": {
    "error": {
      "igroup": {
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "summary": {
        "arguments": [
          {
            "code": "string",
            "message": "string"
          }
        ],
        "code": "4",
        "message": "entry doesn't exist"
      }
    },
    "peer_svm": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "peer1",
      "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
    },
    "state": "string"
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "target": {
```

```
    "firmware_revision": "9111",
    "product_id": "LUN C-Mode",
    "vendor_id": "NETAPP"
  },
  "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374852 | The initiator group does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

summary

A user friendly message describing the connection state of the initiator group.

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

alerts

| Name | Type | Description |
|---|---|---|
| summary | summary | A user friendly message describing the connection state of the initiator group. |

required_nodes

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | |
| uuid | string | |

connectivity_tracking

An overview of the connections to ONTAP by the initiators in this group.

| Name | Type | Description |
|---|---|---|
| alerts | array[alerts] | |
| connection_state | string | Connection state. |
| required_nodes | array[required_nodes] | Nodes to which the initiators in this group should be connected to ensure reliable service. This is the collection of any node hosting a LUN mapped to this igroup as well as the HA partners of those nodes. |

igroup_child

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| comment | string | A comment available for use by the administrator. |
| igroups | array[igroup_child] | Further nested initiator groups. |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

connectivity_tracking

A link to the initiator with connectivity information relevant to its membership of this initiator group.

| Name | Type | Description |
|---|---|---|
| href | string | |

self

A link to the initiator where mutations can be made. If the initiator is inherited from a nested initiator group, the link refers to the initiator in the nested initiator group. In this case, mutations of the initiator will be applied to all initiator groups referencing the same nested initiator group.

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

| Name | Type | Description |
|---|---|---|
| connectivity_tracking | connectivity_tracking | A link to the initiator with connectivity information relevant to its membership of this initiator group. |
| self | self | A link to the initiator where mutations can be made. If the initiator is inherited from a nested initiator group, the link refers to the initiator in the nested initiator group. In this case, mutations of the initiator will be applied to all initiator groups referencing the same nested initiator group. |

connectivity_tracking

Overview of the initiator's connections to ONTAP.

| Name | Type | Description |
|---|---|---|
| connection_state | string | Connection state. |

igroup

The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

peer_svms

A reference to an SVM peer relationship.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
|---|---|---|
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

proximity

Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.

These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`.

| Name | Type | Description |
|---|---|---|
| local_svm | boolean | A boolean that indicates if the initiator is proximal to the SVM of the containing initiator group. This is required for any POST or PATCH that includes the `proximity` sub-object. |
| peer_svms | array[peer_svms] | An array of remote peer SVMs to which the initiator is proximal. |

initiators

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| connectivity_tracking | connectivity_tracking | Overview of the initiator's connections to ONTAP.<br>• readOnly: 1<br>• Introduced in: 9.11 |
| igroup | igroup | The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group. |
| name | string | The FC WWPN, iSCSI IQN, or iSCSI EUI that identifies the host initiator. Valid in POST only and not allowed when the `records` property is used.<br>An FC WWPN consists of 16 hexadecimal digits grouped as 8 pairs separated by colons. The format for an iSCSI IQN is *iqn.yyyy-mm.reverse_domain_name:any*. The iSCSI EUI format consists of the *eui.* prefix followed by 16 hexadecimal characters. |

| Name | Type | Description |
|---|---|---|
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |

node

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | |
| uuid | string | |

lun

The LUN to which the initiator group is mapped.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the LUN. |
| node | node | |
| uuid | string | The unique identifier of the LUN. |

lun_maps

A LUN map with which the initiator group is associated.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| logical_unit_number | integer | The logical unit number assigned to the LUN for initiators in the initiator group. |
| lun | lun | The LUN to which the initiator group is mapped. |

igroup_parent

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| comment | string | A comment available for use by the administrator. |
| name | string | The name of the initiator group. |
| parent_igroups | array[igroup_parent] | The initiator groups that contain this initiator group as a member. |
| uuid | string | The unique identifier of the initiator group. |

portset

The portset to which the initiator group is bound. Binding the initiator group to a portset restricts the initiators of the group to accessing mapped LUNs only through network interfaces in the portset.

In a nested initiator group hierarchy, only a portset bound to the initiator group at the same level at which it is mapped, applies; portsets bound to parent or child initiator groups are ignored.

Optional in POST and PATCH. To unbind a portset from the initiator group, PATCH the `portset` object to `null`, or PATCH `portset.name` to an empty string ("").

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the portset. |
| uuid | string | The unique identifier of the portset. |

igroup

| Name | Type | Description |
|---|---|---|
| local_svm | boolean | Indicates whether the reported igroup is on the local SVM or the peer SVM. When deleting a replicated igroup, the local copy is deleted first and then the peer copy is deleted. If the error is encountered between these two operations and only the peer igroup remains, the peer igroup is reported and the problem might need to be corrected on the peer cluster. |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

summary

A user friendly message describing the error.

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

error

Information about asynchronous errors encountered while replicating this igroup. Igroups within a peering relationship are replicated in the same stream, so the error reported here might be related to this igroup or a prior replicated igroup that is now blocking the replication of this igroup. Both the error information and the igroup encountering the error are reported. If the error is configuration related, it can be corrected on the referenced igroup. The replication is retried using exponential backoff up to a maximum of one retry every 5 minutes. Every operation on the same stream triggers an immediate retry and restarts the exponential backoff starting with a 1 second delay. If the error is system related, the retries should correct the error when the system enters a healthy state.

| Name | Type | Description |
|---|---|---|
| igroup | igroup | |
| summary | summary | A user friendly message describing the error. |

peer_svm

The peered SVM to which the initiator group should be replicated. Optional in POST and PATCH. To clear any previously set replication peer, PATCH the `replication.peer_svm` object to `null`.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

replication

Properties related to initiator group replication.

| Name | Type | Description |
|---|---|---|
| error | error | Information about asynchronous errors encountered while replicating this igroup. Igroups within a peering relationship are replicated in the same stream, so the error reported here might be related to this igroup or a prior replicated igroup that is now blocking the replication of this igroup. Both the error information and the igroup encountering the error are reported. If the error is configuration related, it can be corrected on the referenced igroup. The replication is retried using exponential backoff up to a maximum of one retry every 5 minutes. Every operation on the same stream triggers an immediate retry and restarts the exponential backoff starting with a 1 second delay. If the error is system related, the retries should correct the error when the system enters a healthy state. |

| Name | Type | Description |
|---|---|---|
| peer_svm | peer_svm | The peered SVM to which the initiator group should be replicated. Optional in POST and PATCH. To clear any previously set replication peer, PATCH the `replication.peer_svm` object to `null`. |
| state | string | The state of the replication queue associated with this igroup. If this igroup is not in the replication queue, the state is reported as *ok*. If this igroup is in the replication queue, but no errors have been encountered, the state is reported as *replicating*. If this igroup is in the replication queue and the queue is blocked by an error, the state is reported as *error*. When in the *error* state, additional context is provided by the `replication.error` property. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

target

Properties of the SCSI target to which the initiator group provides access.

| Name | Type | Description |
|---|---|---|
| firmware_revision | string | The firmware revision of the SCSI target specific to the OS type of the initiator group. |

| Name | Type | Description |
|---|---|---|
| product_id | string | The product ID of the SCSI target. |
| vendor_id | string | The vendor ID of the SCSI target. |

returned_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 initiator group

PATCH `/protocols/san/igroups/{uuid}`

**Introduced In:** 9.6

Updates an initiator group.

### Related ONTAP commands

- `lun igroup modify`
- `lun igroup rename`
- `lun igroup bind`
- `lun igroup unbind`

### Learn more

- DOC /protocols/san/igroups

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| uuid | string | path | True | The unique identifier of the initiator group. |

### Request Body

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | An overview of the connections to ONTAP by the initiators in this group.<br>• readOnly: 1<br>• Introduced in: 9.11 |
| delete_on_unmap | boolean | An option that causes the initiator group to be deleted when the last LUN map associated with it is deleted. Optional in POST and PATCH. This property defaults to *false* when the initiator group is created. |
| lun_maps | array[lun_maps] | All LUN maps with which the initiator is associated.<br><br>If the requested igroup is part of a remote, non-local, MetroCluster SVM, the LUN maps are not retrieved.<br><br>There is an added computational cost to retrieving property values for `lun_maps`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| name | string | The name of the initiator group. Required in POST; optional in PATCH. |
| os_type | string | The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. Required in POST; optional in PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| parent_igroups | array[igroup_parent] | The initiator groups that contain this initiator group as a member. |
| portset | portset | The portset to which the initiator group is bound. Binding the initiator group to a portset restricts the initiators of the group to accessing mapped LUNs only through network interfaces in the portset.<br><br>In a nested initiator group hierarchy, only a portset bound to the initiator group at the same level at which it is mapped, applies; portsets bound to parent or child initiator groups are ignored.<br><br>Optional in POST and PATCH. To unbind a portset from the initiator group, PATCH the `portset` object to `null`, or PATCH `portset.name` to an empty string (""). |
| replication | replication | Properties related to initiator group replication. |
| supports_igroups | boolean | An initiator group may contain either initiators or other initiator groups, but not both simultaneously. This property is *true* when initiator groups can be added to this initiator group. The `initiators.name` property cannot be used to determine this via a query because it reports initiators inherited from nested igroups. |
| target | target | Properties of the SCSI target to which the initiator group provides access. |
| uuid | string | The unique identifier of the initiator group. |

**Example request**


```
{
  "comment": "string",
  "connectivity_tracking": {
    "alerts": [
      {}
    ],
    "connection_state": "string",
    "required_nodes": [
      {
        "name": "node1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      }
    ]
  },
  "lun_maps": [
    {
      "logical_unit_number": 0,
      "lun": {
        "name": "lun1",
        "node": {
          "name": "node1",
          "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
        },
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      }
    }
  ],
  "name": "igroup1",
  "os_type": "string",
  "parent_igroups": [
    {
      "comment": "string",
      "name": "igroup1",
      "parent_igroups": [
        null
      ],
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "portset": {
    "name": "portset1",
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "replication": {
```

```
    "error": {
      "igroup": {
        "name": "igroup1",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      }
    },
    "peer_svm": {
      "name": "peer1",
      "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
    },
    "state": "string"
  },
  "target": {
    "firmware_revision": "9111",
    "product_id": "LUN C-Mode",
    "vendor_id": "NETAPP"
  },
  "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
}
```


**Response**

```
Status: 200, Ok
```

**Error**

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 1254264 | An attempt was made to bind a portset to an initiator group that is already bound to a portset. |
| 5373958 | An invalid initiator group name was supplied for a rename operation. |
| 5374023 | A rename operation failed because an initiator group with the same name already exists. |
| 5374027 | An attempt was made to bind a portset with no member network interfaces to the initiator group. |
| 5374028 | An attempt was made to bind a portset with an incompatible protocol to the initiator group. |

| Error Code | Description |
| --- | --- |
| 5374733 | An initiator is already in another initiator group with a conflicting operating system type. |
| 5374745 | An attempt was made to add an initiator group as a child to itself. |
| 5374746 | The cluster is currently running in a mixed version and nested initiator groups cannot be created until the effective cluster version reaches 9.9.1. |
| 5374747 | The cluster is currently running in a mixed version and initiator group comments cannot be created until the effective cluster version reaches 9.9.1. |
| 5374749 | An initiator group's replication peer SVM cannot be changed without first being cleared. |
| 5374759 | An error was reported by the peer cluster while modifying a replicated initiator group. The specific error will be included as a nested error. |
| 5374763 | An error was reported by the peer cluster while renaming a replicated initiator group. The specific error will be included as a nested error. |
| 5374765 | An initiator group cannot be replicated if it has unreplicated child initiator groups. |
| 5374766 | A replicated initiator group cannot be changed to unreplicated if it is the child of a replicated initiator group. |
| 5374770 | An unreplicated initiator group cannot be changed to replicated due to a conflict in its LUN maps. |
| 5374852 | The initiator group does not exist. |
| 5374868 | The initiator group was partially modified before an error was encountered while renaming the initiator group. |
| 5375258 | The igroup is already replicated to a different peer SVM. |
| 5376253 | Initiator group replication requires an effective cluster version of 9.15.1. |
| 5376255 | Initiator group replication requires the peer cluster to have an effective cluster version of 9.15.1. |
| 5376488 | An NVMe over Fabrics subsystem already exists with the requested name. |
| 6620376 | SVM peering information is unavailable. |
| 6620384 | The supplied SVMs are not peered. |

Also see the table of common errors in the Response body overview section of this documentation.

## Definitions

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

summary

A user friendly message describing the connection state of the initiator group.

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

alerts

required_nodes

| Name | Type | Description |
|---|---|---|
| name | string | |
| uuid | string | |

connectivity_tracking

An overview of the connections to ONTAP by the initiators in this group.

| Name | Type | Description |
|---|---|---|
| alerts | array[alerts] | |
| connection_state | string | Connection state. |

| Name | Type | Description |
|---|---|---|
| required_nodes | array[required_nodes] | Nodes to which the initiators in this group should be connected to ensure reliable service. This is the collection of any node hosting a LUN mapped to this igroup as well as the HA partners of those nodes. |

igroup_child

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. |
| igroups | array[igroup_child] | Further nested initiator groups. |

connectivity_tracking

A link to the initiator with connectivity information relevant to its membership of this initiator group.

| Name | Type | Description |
|---|---|---|
| href | string | |

self

A link to the initiator where mutations can be made. If the initiator is inherited from a nested initiator group, the link refers to the initiator in the nested initiator group. In this case, mutations of the initiator will be applied to all initiator groups referencing the same nested initiator group.

| Name | Type | Description |
|---|---|---|
| href | string | |

connectivity_tracking

Overview of the initiator's connections to ONTAP.

| Name | Type | Description |
|---|---|---|
| connection_state | string | Connection state. |

igroup

The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

peer_svms

A reference to an SVM peer relationship.

| Name | Type | Description |
|---|---|---|
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

proximity

Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.

These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`.

| Name | Type | Description |
|---|---|---|
| local_svm | boolean | A boolean that indicates if the initiator is proximal to the SVM of the containing initiator group. This is required for any POST or PATCH that includes the `proximity` sub-object. |
| peer_svms | array[peer_svms] | An array of remote peer SVMs to which the initiator is proximal. |

initiators

| Name | Type | Description |
| --- | --- | --- |
| _links | _links | |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | Overview of the initiator's connections to ONTAP.<br>• readOnly: 1<br>• Introduced in: 9.11 |
| igroup | igroup | The initiator group that directly owns the initiator, which is where modification of the initiator is supported. This property will only be populated when the initiator is a member of a nested initiator group. |
| proximity | proximity | Properties that define to what SVMs the initiator is proximal. This information is used to properly report active optimized and active non-optimized network paths via ALUA. If no configuration has been specified for an initiator, the sub-object will not be present in GET.<br><br>These properties can be set via initiator group POST and PATCH and apply to all instances of the initiator in all initiator groups in the SVM and its peers. The `proximity` sub-object for an initiator is set in POST and PATCH in its entirety and replaces any previously set proximity for the initiator within the SVM for the initiator within the SVM. The `local_svm` property must always be set to `true` or `false` when setting the `proximity` property. To clear any previously set proximity, POST or PATCH the `proximity` object to `null`. |

node

| Name | Type | Description |
|---|---|---|
| name | string | |
| uuid | string | |

lun

The LUN to which the initiator group is mapped.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the LUN. |
| node | node | |
| uuid | string | The unique identifier of the LUN. |

lun_maps

A LUN map with which the initiator group is associated.

| Name | Type | Description |
|---|---|---|
| logical_unit_number | integer | The logical unit number assigned to the LUN for initiators in the initiator group. |
| lun | lun | The LUN to which the initiator group is mapped. |

igroup_parent

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. |
| name | string | The name of the initiator group. |
| parent_igroups | array[igroup_parent] | The initiator groups that contain this initiator group as a member. |
| uuid | string | The unique identifier of the initiator group. |

portset

The portset to which the initiator group is bound. Binding the initiator group to a portset restricts the initiators of the group to accessing mapped LUNs only through network interfaces in the portset.

In a nested initiator group hierarchy, only a portset bound to the initiator group at the same level at which it is mapped, applies; portsets bound to parent or child initiator groups are ignored.

Optional in POST and PATCH. To unbind a portset from the initiator group, PATCH the `portset` object to `null`, or PATCH `portset.name` to an empty string ("").

| Name | Type | Description |
|---|---|---|
| name | string | The name of the portset. |
| uuid | string | The unique identifier of the portset. |

igroup

| Name | Type | Description |
|---|---|---|
| local_svm | boolean | Indicates whether the reported igroup is on the local SVM or the peer SVM. When deleting a replicated igroup, the local copy is deleted first and then the peer copy is deleted. If the error is encountered between these two operations and only the peer igroup remains, the peer igroup is reported and the problem might need to be corrected on the peer cluster. |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

summary

A user friendly message describing the error.

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

error

Information about asynchronous errors encountered while replicating this igroup. Igroups within a peering

relationship are replicated in the same stream, so the error reported here might be related to this igroup or a prior replicated igroup that is now blocking the replication of this igroup. Both the error information and the igroup encountering the error are reported. If the error is configuration related, it can be corrected on the referenced igroup. The replication is retried using exponential backoff up to a maximum of one retry every 5 minutes. Every operation on the same stream triggers an immediate retry and restarts the exponential backoff starting with a 1 second delay. If the error is system related, the retries should correct the error when the system enters a healthy state.

| Name | Type | Description |
|---|---|---|
| igroup | igroup | |

peer_svm

The peered SVM to which the initiator group should be replicated. Optional in POST and PATCH. To clear any previously set replication peer, PATCH the `replication.peer_svm` object to `null`.

| Name | Type | Description |
|---|---|---|
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

replication

Properties related to initiator group replication.

| Name | Type | Description |
| --- | --- | --- |
| error | error | Information about asynchronous errors encountered while replicating this igroup. Igroups within a peering relationship are replicated in the same stream, so the error reported here might be related to this igroup or a prior replicated igroup that is now blocking the replication of this igroup. Both the error information and the igroup encountering the error are reported. If the error is configuration related, it can be corrected on the referenced igroup. The replication is retried using exponential backoff up to a maximum of one retry every 5 minutes. Every operation on the same stream triggers an immediate retry and restarts the exponential backoff starting with a 1 second delay. If the error is system related, the retries should correct the error when the system enters a healthy state. |
| peer_svm | peer_svm | The peered SVM to which the initiator group should be replicated. Optional in POST and PATCH. To clear any previously set replication peer, PATCH the `replication.peer_svm` object to `null`. |
| state | string | The state of the replication queue associated with this igroup. If this igroup is not in the replication queue, the state is reported as *ok*. If this igroup is in the replication queue, but no errors have been encountered, the state is reported as *replicating*. If this igroup is in the replication queue and the queue is blocked by an error, the state is reported as *error*. When in the *error* state, additional context is provided by the `replication.error` property. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

target

Properties of the SCSI target to which the initiator group provides access.

| Name | Type | Description |
|---|---|---|
| firmware_revision | string | The firmware revision of the SCSI target specific to the OS type of the initiator group. |
| product_id | string | The product ID of the SCSI target. |
| vendor_id | string | The vendor ID of the SCSI target. |

igroup

An initiator group (igroup) is a collection of Fibre Channel (FC) world wide port names (WWPNs), and/or iSCSI Qualified Names (IQNs), and/or iSCSI EUIs (Extended Unique Identifiers) that identify host endpoints.

Initiator groups are used to control which hosts can access specific LUNs. To grant access to a LUN from one or more hosts, create an initiator group containing the host initiator names, then create a LUN map that associates the initiator group with the LUN.

An initiator group may contain either initiators or other initiator groups, but not both simultaneously. When a parent initiator group is mapped, it inherits all of the initiators of any initiator groups nested below it. If any nested initiator group is modified to contain different initiators, the parent initiator groups inherit the change. A parent can have many nested initiator groups and an initiator group can be nested under multiple parents. Initiators can only be added or removed from the initiator group that directly contains them. The maximum supported depth of nesting is three layers.

Best practice when using nested initiator groups is to match host hierarchies. A single initiator group should correspond to a single host. If a LUN needs to be mapped to multiple hosts, the initiator groups representing those hosts should be aggregated into a parent initiator group and the LUN should be mapped to that initiator group. For multi-ported hosts, initiators have a comment property where the port corresponding to the initiator can be documented.

An initiator can appear in multiple initiator groups. An initiator group can be mapped to multiple LUNs. A specific initiator can be mapped to a specific LUN only once. With the introduction of nestable initiator groups, best practice is to use the hierarchy such that an initiator is only a direct member of a single initiator group, and that initiator group can then be referenced by other initiator groups.

All initiators or nested initiator groups in an initiator group must be from the same operating system. The initiator group's operating system is specified when the initiator group is created.

When an initiator group is created, the `protocol` property is used to restrict member initiators to Fibre Channel (*fcp*), iSCSI (*iscsi*), or both (*mixed*). Initiator groups within a nested hierarchy may not have conflicting protocols.

Zero or more initiators or nested initiator groups can be supplied when the initiator group is created. After creation, initiators can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/initiators` endpoint. Initiator groups containing other initiator groups report the aggregated list of initiators from all nested initiator groups, but modifications of the initiator list must be performed on the initiator group that directly contains the initiators. See DELETE /protocols/san/igroups/{igroup.uuid}/initiators/{name} for more details.

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| connectivity_tracking | connectivity_tracking | An overview of the connections to ONTAP by the initiators in this group.<br>• readOnly: 1<br>• Introduced in: 9.11 |
| delete_on_unmap | boolean | An option that causes the initiator group to be deleted when the last LUN map associated with it is deleted. Optional in POST and PATCH. This property defaults to *false* when the initiator group is created. |

| Name | Type | Description |
| --- | --- | --- |
| lun_maps | array[lun_maps] | All LUN maps with which the initiator is associated.<br><br>If the requested igroup is part of a remote, non-local, MetroCluster SVM, the LUN maps are not retrieved.<br><br>There is an added computational cost to retrieving property values for `lun_maps`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| name | string | The name of the initiator group. Required in POST; optional in PATCH. |
| os_type | string | The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. Required in POST; optional in PATCH. |
| parent_igroups | array[igroup_parent] | The initiator groups that contain this initiator group as a member. |

| Name | Type | Description |
| --- | --- | --- |
| portset | portset | The portset to which the initiator group is bound. Binding the initiator group to a portset restricts the initiators of the group to accessing mapped LUNs only through network interfaces in the portset.<br><br>In a nested initiator group hierarchy, only a portset bound to the initiator group at the same level at which it is mapped, applies; portsets bound to parent or child initiator groups are ignored.<br><br>Optional in POST and PATCH. To unbind a portset from the initiator group, PATCH the `portset` object to `null`, or PATCH `portset.name` to an empty string (""). |
| replication | replication | Properties related to initiator group replication. |
| supports_igroups | boolean | An initiator group may contain either initiators or other initiator groups, but not both simultaneously. This property is *true* when initiator groups can be added to this initiator group. The `initiators.name` property cannot be used to determine this via a query because it reports initiators inherited from nested igroups. |
| target | target | Properties of the SCSI target to which the initiator group provides access. |
| uuid | string | The unique identifier of the initiator group. |

returned_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. |

# Manage SAN initiators

## Manage SAN initiators

### Overview

An initiator is a Fibre Channel (FC) world wide port name (WWPN), an iSCSI Qualified Name (IQN), or an iSCSI EUI (Extended Unique Identifier) that identifies a host endpoint. Initiators are collected into initiator groups (igroups) used to control which hosts can access specific LUNs. Initiators are also discovered as they log in to SAN network LIFs.

ONTAP supports configuration for an initiator. Configured properties apply to all uses of the initiator within an SVM. Although the same initiator may interact with multiple SVMs of a cluster, ONTAP treats initiator configuration as an SVM-scoped activity. For example, a comment may be set for an initiator in a specific SVM. The comment value applies to all uses of the initiator in the SVM including use in multiple initiator groups. But a different comment value may be specified for the same initiator in a different SVM.

The initiator REST API provides read-only access to properties of initiators.

An FC WWPN consists of 16 hexadecimal digits grouped as 8 pairs separated by colons. The format for an iSCSI IQN is *iqn.yyyy-mm.reverse_domain_name:any*. The iSCSI EUI format consists of the *eui.* prefix followed by 16 hexadecimal characters.

## Retrieve initiators

GET `/protocols/san/initiators`

**Introduced In:** 9.14

Retrieves initiators.

### Related ONTAP commands

- `lun igroup initiator show`

### Learn more

- DOC /protocols/san/initiators

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| comment | string | query | False | Filter by comment<br>• maxLength: 254<br>• minLength: 0 |
| protocol | string | query | False | Filter by protocol |
| name | string | query | False | Filter by name |
| svm.name | string | query | False | Filter by svm.name |
| svm.uuid | string | query | False | Filter by svm.uuid |
| 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.<br>• 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.<br>• Default value: 15<br>• Max value: 120<br>• Min value: 0 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| 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[initiator] | |

### Example response


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "comment": "My initiator comment.",
      "name": "iqn.2018-02.com.netapp.iscsi:name1",
      "protocol": "iscsi",
      "svm": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      }
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

svm

The SVM for which the initiator properties are configured.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

initiator

An initiator is a Fibre Channel (FC) world wide port name (WWPN), an iSCSI Qualified Name (IQN), or an iSCSI EUI (Extended Unique Identifier) that identifies a host endpoint. Initiators are collected into initiator groups (igroups) used to control which hosts can access specific LUNs. Initiators are also discovered as they log in to SAN network LIFs.

ONTAP supports configuration for an initiator. Configured properties apply to all uses of the initiator within an SVM. Although the same initiator may interact with multiple SVMs of a cluster, ONTAP treats initiator configuration as an SVM-scoped activity. For example, a comment may be set for an initiator in a specific SVM. The comment value applies to all uses of the initiator in the SVM including use in multiple initiator groups. But a different comment value may be specified for the same initiator in a different SVM.

| Name | Type | Description |
|---|---|---|
| comment | string | A user-specified comment. |
| name | string | The name of the initiator. |
| protocol | string | The protocol of the initiator. |
| svm | svm | The SVM for which the initiator properties are configured. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 initiator

GET `/protocols/san/initiators/{svm.uuid}/{name}`

**Introduced In:** 9.14

Retrieves an initiator.

### Related ONTAP commands

- `lun igroup initiator show`

### Learn more

- DOC /protocols/san/initiators

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of the SVM for which the initiator properties are configured. |
| name | string | path | True | The name of the initiator. |
| fields | array[string] | query | False | Specify the fields to return. |

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| comment | string | A user-specified comment. |
| name | string | The name of the initiator. |
| protocol | string | The protocol of the initiator. |
| svm | svm | The SVM for which the initiator properties are configured. |

### Example response


```
{
  "comment": "My initiator comment.",
  "name": "iqn.2018-02.com.netapp.iscsi:name1",
  "protocol": "iscsi",
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  }
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

svm

The SVM for which the initiator properties are configured.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

# Manage iSCSI credentials

## Manage iSCSI credentials

### Overview

An iSCSI credentials object defines authentication credentials to be used between an initiator and ONTAP. It identifies an authentication type, user names, and passwords that must be used to authenticate a specific initiator.

The iSCSI credentials REST API allows you to create, update, delete, and discover iSCSI credentials.

### How iSCSI authentication works

An iSCSI credentials object defines the authentication credentials to be used between an initiator and ONTAP. While establishing an iSCSI connection, the initiator sends a login request to ONTAP to begin an iSCSI session. ONTAP then either permits or denies the login request, or determines that a login is not required.

For an initiator, you can specify an authentication type, user names and passwords, and a whitelist of optional network addresses from which the initiator is allowed to connect.

### iSCSI authentication methods

- Challenge-Handshake Authentication Protocol (CHAP) - The initiator logs in using a CHAP user name and password. There are two types of CHAP user names and passwords:
  - Inbound - ONTAP authenticates the initiator. Inbound settings are required if you are using CHAP authentication.
  - Outbound - These are optional credentials to enable the initiator to authenticate ONTAP. You can use credentials only if inbound credentials are also being used.
- deny - The initiator is denied access to ONTAP.
- none - ONTAP does not require authentication for the initiator. The CHAP inbound/outbound password can be any valid string or an even number of valid hexadecimal digits preceded by '0X' or '0x'.

### Initiator address list

The initiator address list is a way to specify valid IP addresses from which the initiator is allowed to connect. If the list is specified and the source address of an iSCSI connection is not in the list, the connection is rejected. Initiator addresses can be specified in either IPv4 or IPv6 format and in one of two forms:

- Range

```
{
  "start": "192.168.0.0",
  "end": "192.168.0.255"
}
```

- Mask

```
{
  "address": "192.168.0.0",
  "netmask": "24"
}
```

### Initiator "default"

The default iSCSI authentication definition is created when the iSCSI service is created. An iSCSI credentials object with *default* as the initiator name identifies the default authentication for an SVM. The default credentials are used for any initiator that does not have specific iSCSI credentials. The default iSCSI authentication method is *none*, but can be changed to *deny* or *CHAP*. The default credentials object does not support an initiator address list.

## Examples

#### Creating iSCSI credentials requiring no authentication

```
# The API:
POST /api/protocols/san/iscsi/credentials

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/san/iscsi/credentials' -H
'Accept: application/hal+json' -d '{ "svm": { "name": "svm1" },
"initiator": "iqn.1992-08.com.netapp:initiator1", "authentication_type":
"none" }'
```

#### Creating iSCSI credentials using CHAP inbound authentication

```
# The API:
POST /api/protocols/san/iscsi/credentials

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/san/iscsi/credentials' -H
'Accept: application/hal+json' -d '{ "svm": { "name": "svm1" },
"initiator": "iqn.1992-08.com.netapp:initiator2", "authentication_type":
"CHAP", "chap": { "inbound": { "user": "user1", "password": "password1" }
} }'
```

#### Retrieving all properties of all iSCSI credentials

The `fields` query parameter is used to request all iSCSI credentials properties.

Passwords are not included in the GET output.

```
# The API:
GET /api/protocols/san/iscsi/credentials

# The call:
curl -X GET 'https://<mgmt-
ip>/api/protocols/san/iscsi/credentials?fields=*' -H 'Accept:
application/hal+json'

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "19d04b8e-94d7-11e8-8370-005056b48fd2",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/19d04b8e-94d7-11e8-8370-005056b48fd2"
        }
      }
    },
    "initiator": "default",
    "authentication_type": "none",
    "_links": {
      "self": {
        "href": "/api/protocols/san/iscsi/credentials/19d04b8e-94d7-11e8-
8370-005056b48fd2/default"
      }
    }
  },
  {
    "svm": {
      "uuid": "19d04b8e-94d7-11e8-8370-005056b48fd2",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/19d04b8e-94d7-11e8-8370-005056b48fd2"
        }
      }
    },
    "initiator": "iqn.1992-08.com.netapp:initiator1",
    "authentication_type": "none",
    "_links": {
      "self": {
```

```
        "href": "/api/protocols/san/iscsi/credentials/19d04b8e-94d7-11e8-
8370-005056b48fd2/iqn.1992-08.com.netapp:initiator1"
      }
    }
  },
  {
    "svm": {
      "uuid": "19d04b8e-94d7-11e8-8370-005056b48fd2",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/19d04b8e-94d7-11e8-8370-005056b48fd2"
        }
      }
    },
    "initiator": "iqn.1992-08.com.netapp:initiator2",
    "authentication_type": "chap",
    "chap": {
      "inbound": {
        "user": "user1"
      }
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/iscsi/credentials/19d04b8e-94d7-11e8-
8370-005056b48fd2/iqn.1992-08.com.netapp:initiator2"
      }
    }
  },
  {
    "svm": {
      "uuid": "25f617cf-94d7-11e8-8370-005056b48fd2",
      "name": "svm2",
      "_links": {
        "self": {
          "href": "/api/svm/svms/25f617cf-94d7-11e8-8370-005056b48fd2"
        }
      }
    },
    "initiator": "default",
    "authentication_type": "none",
    "_links": {
      "self": {
        "href": "/api/protocols/san/iscsi/credentials/25f617cf-94d7-11e8-
8370-005056b48fd2/default"
      }
```

```
    }
  },
  {
    "svm": {
      "uuid": "25f617cf-94d7-11e8-8370-005056b48fd2",
      "name": "svm2",
      "_links": {
        "self": {
          "href": "/api/svm/svms/25f617cf-94d7-11e8-8370-005056b48fd2"
        }
      }
    },
    "initiator": "iqn.1992-08.com.netapp:initiator2",
    "authentication_type": "none",
    "_links": {
      "self": {
        "href": "/api/protocols/san/iscsi/credentials/25f617cf-94d7-11e8-
8370-005056b48fd2/iqn.1992-08.com.netapp:initiator2"
      }
    }
  },
  {
    "svm": {
      "uuid": "25f617cf-94d7-11e8-8370-005056b48fd2",
      "name": "svm2",
      "_links": {
        "self": {
          "href": "/api/svm/svms/25f617cf-94d7-11e8-8370-005056b48fd2"
        }
      }
    },
    "initiator": "iqn.1992-08.com.netapp:initiator3",
    "authentication_type": "deny",
    "_links": {
      "self": {
        "href": "/api/protocols/san/iscsi/credentials/25f617cf-94d7-11e8-
8370-005056b48fd2/iqn.1992-08.com.netapp:initiator3"
      }
    }
  }
],
"num_records": 6,
"_links": {
  "self": {
    "href": "/api/protocols/san/iscsi/credentials?fields=*"
  }
```

```
}
}
```

#### Retrieving specific iSCSI credentials

```
# The API:
GET /api/protocols/san/iscsi/credentials/{svm.uuid}/{initiator}

# The call:
curl -X GET 'https://<mgmt-
ip>/api/protocols/san/iscsi/credentials/25f617cf-94d7-11e8-8370-
005056b48fd2/iqn.1992-08.com.netapp:initiator2' -H 'Accept:
application/hal+json'

# The response:
{
"svm": {
  "uuid": "25f617cf-94d7-11e8-8370-005056b48fd2",
  "name": "svm2",
  "_links": {
    "self": {
      "href": "/api/svm/svms/25f617cf-94d7-11e8-8370-005056b48fd2"
    }
  }
},
"initiator": "iqn.1992-08.com.netapp:initiator2",
"authentication_type": "chap",
"chap": {
  "inbound": {
    "user": "user1"
  }
},
"_links": {
  "self": {
    "href": "/api/protocols/san/iscsi/credentials/25f617cf-94d7-11e8-8370-
005056b48fd2/iqn.1992-08.com.netapp:initiator2"
  }
}
}
```

#### Updating the authentication type of iSCSI credentials

```
# The API:
PATCH /api/protocols/san/iscsi/credentials/{svm.uuid}/{initiator}

# The call:
curl -X PATCH 'https://<mgmt-
ip>/api/protocols/san/iscsi/credentials/25f617cf-94d7-11e8-8370-
005056b48fd2/iqn.1992-08.com.netapp:initiator2' -H 'Accept:
application/hal+json' -d '{ "authentication_type": "chap", "chap": {
"inbound": { "user": "user1", "password": "password1" } } }'
```

#### Updating the initiator address list of iSCSI credentials

```
# The API:
PATCH /api/protocols/san/iscsi/credentials/{svm.uuid}/{initiator}

# The call:
curl -X PATCH 'https://<mgmt-
ip>/api/protocols/san/iscsi/credentials/25f617cf-94d7-11e8-8370-
005056b48fd2/iqn.1992-08.com.netapp:initiator2' -H 'Accept:
application/hal+json' -d '{ "initiator_address": { "ranges": [ { "start":
"192.168.0.0", "end": "192.168.255.255" } ] } }'
```

#### Deleting iSCSI credentials

```
# The API:
DELETE /api/protocols/san/iscsi/credentials/{svm.uuid}/{initiator}

# The call:
curl -X DELETE 'https://<mgmt-
ip>/api/protocols/san/iscsi/credentials/25f617cf-94d7-11e8-8370-
005056b48fd2/iqn.1992-08.com.netapp:initiator2' -H 'Accept:
application/hal+json'
```

## Retrieve iSCSI credentials

GET `/protocols/san/iscsi/credentials`

**Introduced In:** 9.6

Retrieves iSCSI credentials.

### Related ONTAP commands

- `vserver iscsi security show`

### Learn more

- DOC /protocols/san/iscsi/credentials

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| authentication_type | string | query | False | Filter by authentication_type |
| initiator | string | query | False | Filter by initiator |
| chap.outbound.user | string | query | False | Filter by chap.outbound.user<br>• maxLength: 128<br>• minLength: 1 |
| chap.inbound.user | string | query | False | Filter by chap.inbound.user<br>• maxLength: 128<br>• minLength: 1 |
| svm.name | string | query | False | Filter by svm.name |
| svm.uuid | string | query | False | Filter by svm.uuid |
| initiator_address.ranges.end | string | query | False | Filter by initiator_address.ranges.end |
| initiator_address.ranges.family | string | query | False | Filter by initiator_address.ranges.family |
| initiator_address.ranges.start | string | query | False | Filter by initiator_address.ranges.start |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| initiator_address.masks.family | string | query | False | Filter by initiator_address.masks.family |
| initiator_address.masks.netmask | string | query | False | Filter by initiator_address.masks.netmask |
| initiator_address.masks.address | string | query | False | Filter by initiator_address.masks.address |
| 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.<br>• 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[iscsi_credentials] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "authentication_type": "string",
      "chap": {
        "inbound": {
          "user": "string"
        },
        "outbound": {
          "user": "string"
        }
      },
      "initiator": "iqn.1998-01.com.corp.iscsi:name1",
      "initiator_address": {
        "masks": [
          {
            "address": "10.10.10.7",
            "family": "string",
            "netmask": "24"
          }
        ],
        "ranges": [
          {
            "end": "10.10.10.7",
            "family": "string",
            "start": "10.10.10.7"
          }
        ]
      },
      "svm": {
```

```
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      }
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

inbound

Inbound CHAP credentials.

| Name | Type | Description |
|---|---|---|
| user | string | The inbound CHAP user name. Optional in POST and PATCH. |

outbound

Output CHAP credentials.

To clear previously set outbound CHAP credentials, set property `chap.outbound.user` to an empty string in PATCH.

| Name | Type | Description |
|---|---|---|
| user | string | The outbound CHAP user name. Optional in POST and PATCH.<br><br>To clear previously set outbound CHAP credentials, set this property to an empty string in PATCH. |

chap

Challenge-Handshake Authentication Protocol (CHAP) credentials.

| Name | Type | Description |
|---|---|---|
| inbound | inbound | Inbound CHAP credentials. |
| outbound | outbound | Output CHAP credentials.<br><br>To clear previously set outbound CHAP credentials, set property `chap.outbound.user` to an empty string in PATCH. |

ip_info

IP information

| Name | Type | Description |
|---|---|---|
| address | string | IPv4 or IPv6 address |
| family | string | IPv4 or IPv6 |
| netmask | string | Input as netmask length (16) or IPv4 mask (255.255.0.0). For IPv6, the default value is 64 with a valid range of 1 to 127. Output is always the netmask length. |

ip_address_range

IP address range

| Name | Type | Description |
|---|---|---|
| end | string | IPv4 or IPv6 address |
| family | string | IPv4 or IPv6 |
| start | string | IPv4 or IPv6 address |

initiator_address

Initiator address ranges.

| Name | Type | Description |
|---|---|---|
| masks | array[ip_info] | |
| ranges | array[ip_address_range] | |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

iscsi_credentials

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| authentication_type | string | The iSCSI authentication type. Required in POST; optional in PATCH. |
| chap | chap | Challenge-Handshake Authentication Protocol (CHAP) credentials. |
| initiator | string | The iSCSI initiator to which the credentials apply. Required in POST. |
| initiator_address | initiator_address | Initiator address ranges. |
| svm | svm | SVM, applies only to SVM-scoped objects. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 iSCSI credentials

POST `/protocols/san/iscsi/credentials`

**Introduced In:** 9.6

Creates iSCSI credentials.

### Required properties

- `svm.uuid` or `svm.name` - Existing SVM in which to create the iSCSI credentials.
- `initiator` - Initiator for which the iSCSI credentials are to be created.
- `authentication_type` - Type of authentication to use for the credentials.

### Recommended optional properties

- `chap.inbound.user` - In-bound CHAP authentication user name.
- `chap.inbound.password` - In-bound CHAP authentication password.
- `chap.outbound.user` - Out-bound CHAP authentication user name.
- `chap.outbound.password` - Out-bound CHAP authentication password.

### Related ONTAP commands

- `vserver iscsi security create`

### Learn more

- DOC /protocols/san/iscsi/credentials

### Parameters

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

**Request Body**

| Name | Type | Description |
|---|---|---|
| authentication_type | string | The iSCSI authentication type. Required in POST; optional in PATCH. |
| chap | chap | Challenge-Handshake Authentication Protocol (CHAP) credentials. |
| initiator | string | The iSCSI initiator to which the credentials apply. Required in POST. |
| initiator_address | initiator_address | Initiator address ranges. |
| svm | svm | SVM, applies only to SVM-scoped objects. |

### Example request


```
{
  "authentication_type": "string",
  "chap": {
    "inbound": {
      "password": "string",
      "user": "string"
    },
    "outbound": {
      "password": "string",
      "user": "string"
    }
  },
  "initiator": "iqn.1998-01.com.corp.iscsi:name1",
  "initiator_address": {
    "masks": [
      {
        "address": "10.10.10.7",
        "netmask": "24"
      }
    ],
    "ranges": [
      {
        "end": "10.10.10.7",
        "start": "10.10.10.7"
      }
    ]
  },
  "svm": {
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  }
}
```


### Response

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| num_records | integer | The number of records in the response. |

| Name | Type | Description |
|---|---|---|
| records | array[iscsi_credentials] | |

**Example response**


```
{
  "num_records": 1,
  "records": [
    {
      "authentication_type": "string",
      "chap": {
        "inbound": {
          "password": "string",
          "user": "string"
        },
        "outbound": {
          "password": "string",
          "user": "string"
        }
      },
      "initiator": "iqn.1998-01.com.corp.iscsi:name1",
      "initiator_address": {
        "masks": [
          {
            "address": "10.10.10.7",
            "netmask": "24"
          }
        ],
        "ranges": [
          {
            "end": "10.10.10.7",
            "start": "10.10.10.7"
          }
        ]
      },
      "svm": {
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      }
    }
  ]
}
```

#### Headers

| Name | Description | Type |
|---|---|---|
| Location | Useful for tracking the resource location | string |

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | An SVM with the specified UUID does not exist. |
| 2621706 | Both the SVM UUID and SVM name were supplied, but they do not refer to the same SVM. |
| 2621707 | No SVM was specified. Either `svm.name` or `svm.uuid` must be supplied. |
| 5373969 | A non-empty qualifier is required after the prefix. An example of a valid IQN is *iqn.1995-08.com.example:string*. |
| 5373970 | The IQN prefix is invalid. The correct IQN prefix is *iqn*. An example of a valid IQN is *iqn.1995-08.com.example:string*. |
| 5373971 | The date field is invalid. A valid date field is *yyyy-mm*. An example of a valid IQN is *iqn.1995-08.com.example:string*. |
| 5373972 | The naming authority and string fields can contain only the characters *a-z*, *0-9*, ., *:*, and -. |
| 5373977 | The EUI-64 identifier field must be exactly 16 hexadecimal digits. |
| 5373978 | The EUI formatted initiator name supplied is invalid. A valid EUI format is *eui.XXXXXXXXXXXXXXXX*, where X is a hexadecimal digit. |
| 5373997 | The initiator name supplied in invalid. The valid initiator name formats are *iqn.1995-08.com.example:string* or *eui.0123456789abcdef*. |
| 5374078 | The iSCSI service does not exist. |
| 5374142 | An iSCSI security credential already exists for the specified initiator. |
| 5374145 | The iSCSI security password must contain an even number of valid hex digits. |

| Error Code | Description |
| --- | --- |
| 5374147 | The CHAP inbound and outbound passwords must be different. |
| 5374149 | The inbound user and password properties are required for CHAP authentication. |
| 5374150 | Outbound CHAP authentication requires an outbound password. |
| 5374855 | The value for property `initiator_address.ranges.start` is greater than the value for property `initiator_address.ranges.end`. |
| 5374856 | The value for property `initiator_address.ranges.start` does not belong to the same IP address family as the value for property `initiator_address.ranges.end`. |
| 5374900 | Setting the CHAP authentication properties are not supported with authentication types *none* or *deny*. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

inbound

Inbound CHAP credentials.

| Name | Type | Description |
|---|---|---|
| password | string | The inbound CHAP password. Write-only; optional in POST and PATCH. |
| user | string | The inbound CHAP user name. Optional in POST and PATCH. |

outbound

Output CHAP credentials.

To clear previously set outbound CHAP credentials, set property `chap.outbound.user` to an empty string in PATCH.

| Name | Type | Description |
|---|---|---|
| password | string | The outbound CHAP password. Write-only; optional in POST and PATCH. |
| user | string | The outbound CHAP user name. Optional in POST and PATCH.<br><br>To clear previously set outbound CHAP credentials, set this property to an empty string in PATCH. |

chap

Challenge-Handshake Authentication Protocol (CHAP) credentials.

| Name | Type | Description |
|---|---|---|
| inbound | inbound | Inbound CHAP credentials. |

| Name | Type | Description |
| --- | --- | --- |
| outbound | outbound | Output CHAP credentials.<br><br>To clear previously set outbound CHAP credentials, set property `chap.outbound.user` to an empty string in PATCH. |

ip_info

IP information

| Name | Type | Description |
| --- | --- | --- |
| address | string | IPv4 or IPv6 address |
| netmask | string | Input as netmask length (16) or IPv4 mask (255.255.0.0). For IPv6, the default value is 64 with a valid range of 1 to 127. Output is always the netmask length. |

ip_address_range

IP address range

| Name | Type | Description |
| --- | --- | --- |
| end | string | IPv4 or IPv6 address |
| start | string | IPv4 or IPv6 address |

initiator_address

Initiator address ranges.

| Name | Type | Description |
| --- | --- | --- |
| masks | array[ip_info] | |
| ranges | array[ip_address_range] | |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

iscsi_credentials

| Name | Type | Description |
|---|---|---|
| authentication_type | string | The iSCSI authentication type. Required in POST; optional in PATCH. |
| chap | chap | Challenge-Handshake Authentication Protocol (CHAP) credentials. |
| initiator | string | The iSCSI initiator to which the credentials apply. Required in POST. |
| initiator_address | initiator_address | Initiator address ranges. |
| svm | svm | SVM, applies only to SVM-scoped objects. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

## Delete iSCSI credentials

DELETE `/protocols/san/iscsi/credentials/{svm.uuid}/{initiator}`

**Introduced In:** 9.6

Deletes specified iSCSI credentials.

### Related ONTAP commands

- `vserver iscsi security delete`

### Learn more

- DOC /protocols/san/iscsi/credentials

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of an SVM. |
| initiator | string | path | True | The iSCSI initiator of the credentials object. |

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | An SVM with the specified UUID does not exist. |

| Error Code | Description |
| --- | --- |
| 2621706 | Both the SVM UUID and SVM name were supplied, but they do not refer to the same SVM. |
| 2621707 | No SVM was specified. Either `svm.name` or `svm.uuid` must be supplied. |
| 5374148 | The default security credential cannot be deleted for an SVM. |
| 5374895 | The iSCSI security credential does not exist on the specified SVM. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
| --- | --- | --- |
| error | returned_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 |

returned_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 specific iSCSI credentials

GET `/protocols/san/iscsi/credentials/{svm.uuid}/{initiator}`

**Introduced In:** 9.6

Retrieves specified iSCSI credentials.

### Related ONTAP commands

- `vserver iscsi security show`

### Learn more

- DOC /protocols/san/iscsi/credentials

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of an SVM. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| initiator | string | path | True | The iSCSI initiator of the credentials object. |
| fields | array[string] | query | False | Specify the fields to return. |

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| authentication_type | string | The iSCSI authentication type. Required in POST; optional in PATCH. |
| chap | chap | Challenge-Handshake Authentication Protocol (CHAP) credentials. |
| initiator | string | The iSCSI initiator to which the credentials apply. Required in POST. |
| initiator_address | initiator_address | Initiator address ranges. |
| svm | svm | SVM, applies only to SVM-scoped objects. |

**Example response**


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "authentication_type": "string",
  "chap": {
    "inbound": {
      "user": "string"
    },
    "outbound": {
      "user": "string"
    }
  },
  "initiator": "iqn.1998-01.com.corp.iscsi:name1",
  "initiator_address": {
    "masks": [
      {
        "address": "10.10.10.7",
        "family": "string",
        "netmask": "24"
      }
    ],
    "ranges": [
      {
        "end": "10.10.10.7",
        "family": "string",
        "start": "10.10.10.7"
      }
    ]
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  }
}
```

### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

inbound

Inbound CHAP credentials.

| Name | Type | Description |
|---|---|---|
| user | string | The inbound CHAP user name. Optional in POST and PATCH. |

outbound

Output CHAP credentials.

To clear previously set outbound CHAP credentials, set property `chap.outbound.user` to an empty string in PATCH.

| Name | Type | Description |
|---|---|---|
| user | string | The outbound CHAP user name. Optional in POST and PATCH.<br><br>To clear previously set outbound CHAP credentials, set this property to an empty string in PATCH. |

chap

Challenge-Handshake Authentication Protocol (CHAP) credentials.

| Name | Type | Description |
|---|---|---|
| inbound | inbound | Inbound CHAP credentials. |

| Name | Type | Description |
| --- | --- | --- |
| outbound | outbound | Output CHAP credentials.<br><br>To clear previously set outbound CHAP credentials, set property `chap.outbound.user` to an empty string in PATCH. |

ip_info

IP information

| Name | Type | Description |
| --- | --- | --- |
| address | string | IPv4 or IPv6 address |
| family | string | IPv4 or IPv6 |
| netmask | string | Input as netmask length (16) or IPv4 mask (255.255.0.0). For IPv6, the default value is 64 with a valid range of 1 to 127. Output is always the netmask length. |

ip_address_range

IP address range

| Name | Type | Description |
| --- | --- | --- |
| end | string | IPv4 or IPv6 address |
| family | string | IPv4 or IPv6 |
| start | string | IPv4 or IPv6 address |

initiator_address

Initiator address ranges.

| Name | Type | Description |
| --- | --- | --- |
| masks | array[ip_info] | |
| ranges | array[ip_address_range] | |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 iSCSI credentials

PATCH `/protocols/san/iscsi/credentials/{svm.uuid}/{initiator}`

**Introduced In:** 9.6

Updates specified iSCSI credentials.

### Related ONTAP commands

- `vserver iscsi security add-initiator-address-ranges`
- `vserver iscsi security default`
- `vserver iscsi security modify`
- `vserver iscsi security remove-initiator-address-ranges`

### Learn more

- DOC /protocols/san/iscsi/credentials

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of an SVM. |
| initiator | string | path | True | The iSCSI initiator of the credentials object. |
| add_initiator_addresses | boolean | query | False | If *true*, the initiator addresses in the body merge into the existing addresses in the iSCSI security object rather than replace the existing addresses.<br>• Default value: |
| remove_initiator_addresses | boolean | query | False | If *true*, the initiator addresses in the body are removed from the existing addresses in the iSCSI security object rather than replace the existing addresses.<br>• Default value: |

### Request Body

| Name | Type | Description |
|---|---|---|
| authentication_type | string | The iSCSI authentication type. Required in POST; optional in PATCH. |
| chap | chap | Challenge-Handshake Authentication Protocol (CHAP) credentials. |
| initiator_address | initiator_address | Initiator address ranges. |

### Example request


```
{
  "authentication_type": "string",
  "chap": {
    "inbound": {
      "password": "string",
      "user": "string"
    },
    "outbound": {
      "password": "string",
      "user": "string"
    }
  },
  "initiator_address": {
    "masks": [
      {
        "address": "10.10.10.7",
        "netmask": "24"
      }
    ],
    "ranges": [
      {
        "end": "10.10.10.7",
        "start": "10.10.10.7"
      }
    ]
  }
}
```


### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | An SVM with the specified UUID does not exist. |

| Error Code | Description |
| --- | --- |
| 2621706 | Both the SVM UUID and SVM name were supplied, but they do not refer to the same SVM. |
| 2621707 | No SVM was specified. Either `svm.name` or `svm.uuid` must be supplied. |
| 5374145 | The iSCSI security password must contain an even number of valid hex digits. |
| 5374147 | The CHAP inbound and outbound passwords must be different. |
| 5374149 | The inbound user and password properties are required for CHAP authentication. |
| 5374150 | Outbound CHAP authentication requires an outbound password. |
| 5374155 | The functionality is not supported for the default security credential. |
| 5374855 | The value for property `initiator_address.ranges.start` is greater than the value for property `initiator_address.ranges.end`. |
| 5374856 | The value for property `initiator_address.ranges.start` does not belong to the same IP address family as the value for property `initiator_address.ranges.end`. |
| 5374895 | The iSCSI security credential does not exist on the specified SVM. |
| 5374900 | Setting the CHAP authentication properties are not supported with authentication types *none* or *deny*. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

inbound

Inbound CHAP credentials.

| Name | Type | Description |
|---|---|---|
| password | string | The inbound CHAP password. Write-only; optional in POST and PATCH. |
| user | string | The inbound CHAP user name. Optional in POST and PATCH. |

outbound

Output CHAP credentials.

To clear previously set outbound CHAP credentials, set property `chap.outbound.user` to an empty string in PATCH.

| Name | Type | Description |
|---|---|---|
| password | string | The outbound CHAP password. Write-only; optional in POST and PATCH. |
| user | string | The outbound CHAP user name. Optional in POST and PATCH.<br><br>To clear previously set outbound CHAP credentials, set this property to an empty string in PATCH. |

chap

Challenge-Handshake Authentication Protocol (CHAP) credentials.

| Name | Type | Description |
|---|---|---|
| inbound | inbound | Inbound CHAP credentials. |

| Name | Type | Description |
|---|---|---|
| outbound | outbound | Output CHAP credentials.<br><br>To clear previously set outbound CHAP credentials, set property `chap.outbound.user` to an empty string in PATCH. |

ip_info

IP information

| Name | Type | Description |
|---|---|---|
| address | string | IPv4 or IPv6 address |
| netmask | string | Input as netmask length (16) or IPv4 mask (255.255.0.0). For IPv6, the default value is 64 with a valid range of 1 to 127. Output is always the netmask length. |

ip_address_range

IP address range

| Name | Type | Description |
|---|---|---|
| end | string | IPv4 or IPv6 address |
| start | string | IPv4 or IPv6 address |

initiator_address

Initiator address ranges.

| Name | Type | Description |
|---|---|---|
| masks | array[ip_info] | |
| ranges | array[ip_address_range] | |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

iscsi_credentials

| Name | Type | Description |
|---|---|---|
| authentication_type | string | The iSCSI authentication type. Required in POST; optional in PATCH. |
| chap | chap | Challenge-Handshake Authentication Protocol (CHAP) credentials. |
| initiator_address | initiator_address | Initiator address ranges. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

# Manage iSCSI services

## Manage iSCSI services

### Overview

An iSCSI service defines the properties of the iSCSI target for an SVM. There can be at most one iSCSI service for an SVM. An SVM's iSCSI service must be created before iSCSI initiators can log in to the SVM.

The iSCSI service REST API allows you to create, update, delete, and discover iSCSI services for SVMs.

### Performance monitoring

Performance of the SVM can be monitored by the `metric.*` and `statistics.*` properties. These show the performance of the SVM in terms of IOPS, latency and throughput. The `metric.*` properties denote an average whereas `statistics.*` properties denote a real-time monotonically increasing value aggregated across all nodes.

### Examples

#### Creating an iSCSI service for an SVM

The simplest way to create an iSCSI service is to specify only the SVM, either by name or UUID. By default, the new iSCSI service is enabled and uses the SVM name as its target alias.

In this example, the `return_records` query parameter is used to retrieve the new iSCSI service object in the REST response.

```
# The API:
POST /api/protocols/san/iscsi/services

# The call:
curl -X POST 'https://<mgmt-
ip>/api/protocols/san/iscsi/services?return_records=true' -H 'Accept:
application/hal+json' -d '{ "svm": { "name": "svm1" } }'

# The response:
{
"num_records": 1,
"records": [
  {
    "svm": {
      "uuid": "19d04b8e-94d7-11e8-8370-005056b48fd2",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/19d04b8e-94d7-11e8-8370-005056b48fd2"
        }
      }
    },
    "enabled": true,
    "target": {
      "name": "iqn.1992-
08.com.netapp:sn.19d04b8e94d711e88370005056b48fd2:vs.4",
      "alias": "svm1"
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/iscsi/services/19d04b8e-94d7-11e8-
8370-005056b48fd2"
      }
    }
  }
]
}
```

#### Retrieving the iSCSI services for all SVMs in the cluster

```
# The API:
GET /api/protocols/san/iscsi/services
```

```
# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/iscsi/services' -H
'Accept: application/hal+json'

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "19d04b8e-94d7-11e8-8370-005056b48fd2",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/19d04b8e-94d7-11e8-8370-005056b48fd2"
        }
      }
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/iscsi/services/19d04b8e-94d7-11e8-
8370-005056b48fd2"
      }
    }
  },
  {
    "svm": {
      "uuid": "25f617cf-94d7-11e8-8370-005056b48fd2",
      "name": "svm2",
      "_links": {
        "self": {
          "href": "/api/svm/svms/25f617cf-94d7-11e8-8370-005056b48fd2"
        }
      }
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/iscsi/services/25f617cf-94d7-11e8-
8370-005056b48fd2"
      }
    }
  }
],
"num_records": 2,
"_links": {
  "self": {
    "href": "/api/protocols/san/iscsi/services"
```

```
  }
}
}
```

#### Retrieving details for a specific iSCSI service

The iSCSI service is identified by the UUID of its SVM.

```
# The API:
GET /api/protocols/san/iscsi/services/{svm.uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/iscsi/services/19d04b8e-
94d7-11e8-8370-005056b48fd2' -H 'Accept: application/hal+json'

# The response:
{
"svm": {
  "uuid": "19d04b8e-94d7-11e8-8370-005056b48fd2",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/19d04b8e-94d7-11e8-8370-005056b48fd2"
    }
  }
},
"enabled": true,
"target": {
  "name": "iqn.1992-
08.com.netapp:sn.19d04b8e94d711e88370005056b48fd2:vs.4",
  "alias": "svm1"
},
"_links": {
  "self": {
    "href": "/api/protocols/san/iscsi/services/19d04b8e-94d7-11e8-8370-
005056b48fd2"
  }
}
}
```

#### Disabling an iSCSI service

Disabling an iSCSI service shuts down all active iSCSI sessions for the SVM and prevents the creation of new iSCSI sessions.

The iSCSI service to update is identified by the UUID of its SVM.

```
# The API:
PATCH /api/protocols/san/iscsi/services/{svm.uuid}

# The call:
curl -X PATCH 'https://<mgmt-
ip>/api/protocols/san/iscsi/services/19d04b8e-94d7-11e8-8370-005056b48fd2'
-H 'Accept: application/hal+json' -d '{ "enabled": "false" }'
```

You can retrieve the iSCSI service to confirm the change.

In this example, the `fields` query parameter is used to limit the response to the `enabled` property and iSCSI service identifiers.

```
# The API:
GET /api/protocols/san/iscsi/services/{svm.uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/iscsi/services/19d04b8e-
94d7-11e8-8370-005056b48fd2?fields=enabled' -H 'Accept:
application/hal+json'

# The response:
{
"svm": {
  "uuid": "19d04b8e-94d7-11e8-8370-005056b48fd2",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/19d04b8e-94d7-11e8-8370-005056b48fd2"
    }
  }
},
"enabled": false,
"_links": {
  "self": {
    "href": "/api/protocols/san/iscsi/services/19d04b8e-94d7-11e8-8370-
005056b48fd2"
  }
}
}
```

#### Deleting an iSCSI service

The iSCSI service must be disabled before it can be deleted.

The iSCSI service to be deleted is identified by the UUID of its SVM.

```
# The API:
DELETE /api/protocols/san/iscsi/services/{svm.uuid}

# The call:
curl -X DELETE 'https://<mgmt-
ip>/api/protocols/san/iscsi/services/19d04b8e-94d7-11e8-8370-005056b48fd2'
-H 'Accept: application/hal+json'
```

## Retrieve iSCSI services

GET `/protocols/san/iscsi/services`

**Introduced In:** 9.6

Retrieves iSCSI services.

### 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.

- `statistics.*`
- `metric.*`

### Related ONTAP commands

- `vserver iscsi show`

### Learn more

- DOC /protocols/san/iscsi/services

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| statistics.latency_raw.other | integer | query | False | Filter by statistics.latency_raw.other<br>• Introduced in: 9.7 |
| statistics.latency_raw.total | integer | query | False | Filter by statistics.latency_raw.total<br>• Introduced in: 9.7 |
| statistics.latency_raw.read | integer | query | False | Filter by statistics.latency_raw.read<br>• Introduced in: 9.7 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| statistics.latency_raw.write | integer | query | False | Filter by statistics.latency_raw.write<br>• Introduced in: 9.7 |
| statistics.throughput_raw.read | integer | query | False | Filter by statistics.throughput_raw.read<br>• Introduced in: 9.7 |
| statistics.throughput_raw.total | integer | query | False | Filter by statistics.throughput_raw.total<br>• Introduced in: 9.7 |
| statistics.throughput_raw.write | integer | query | False | Filter by statistics.throughput_raw.write<br>• Introduced in: 9.7 |
| statistics.iops_raw.other | integer | query | False | Filter by statistics.iops_raw.other<br>• Introduced in: 9.7 |
| statistics.iops_raw.total | integer | query | False | Filter by statistics.iops_raw.total<br>• Introduced in: 9.7 |
| statistics.iops_raw.read | integer | query | False | Filter by statistics.iops_raw.read<br>• Introduced in: 9.7 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| statistics.iops_raw.write | integer | query | False | Filter by statistics.iops_raw.write<br>• Introduced in: 9.7 |
| statistics.status | string | query | False | Filter by statistics.status<br>• Introduced in: 9.7 |
| statistics.timestamp | string | query | False | Filter by statistics.timestamp<br>• Introduced in: 9.7 |
| svm.name | string | query | False | Filter by svm.name |
| svm.uuid | string | query | False | Filter by svm.uuid |
| target.alias | string | query | False | Filter by target.alias<br>• maxLength: 128<br>• minLength: 1 |
| target.name | string | query | False | Filter by target.name<br>• maxLength: 128<br>• minLength: 1 |
| enabled | boolean | query | False | Filter by enabled |
| metric.iops.other | integer | query | False | Filter by metric.iops.other<br>• Introduced in: 9.7 |
| metric.iops.total | integer | query | False | Filter by metric.iops.total<br>• Introduced in: 9.7 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| metric.iops.read | integer | query | False | Filter by metric.iops.read<br>• Introduced in: 9.7 |
| metric.iops.write | integer | query | False | Filter by metric.iops.write<br>• Introduced in: 9.7 |
| metric.status | string | query | False | Filter by metric.status<br>• Introduced in: 9.7 |
| metric.latency.other | integer | query | False | Filter by metric.latency.other<br>• Introduced in: 9.7 |
| metric.latency.total | integer | query | False | Filter by metric.latency.total<br>• Introduced in: 9.7 |
| metric.latency.read | integer | query | False | Filter by metric.latency.read<br>• Introduced in: 9.7 |
| metric.latency.write | integer | query | False | Filter by metric.latency.write<br>• Introduced in: 9.7 |
| metric.duration | string | query | False | Filter by metric.duration<br>• Introduced in: 9.7 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| metric.timestamp | string | query | False | Filter by metric.timestamp<br>• Introduced in: 9.7 |
| metric.throughput.read | integer | query | False | Filter by metric.throughput.read<br>• Introduced in: 9.7 |
| metric.throughput.total | integer | query | False | Filter by metric.throughput.total<br>• Introduced in: 9.7 |
| metric.throughput.write | integer | query | False | Filter by metric.throughput.write<br>• Introduced in: 9.7 |
| 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.<br>• Default value: 1 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[iscsi_service] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "metric": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "duration": "PT15S",
        "iops": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "latency": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "status": "ok",
        "throughput": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "timestamp": "2017-01-25 06:20:13 -0500"
      },
      "statistics": {
        "iops_raw": {
```

```
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "latency_raw": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "status": "ok",
        "throughput_raw": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "timestamp": "2017-01-25 06:20:13 -0500"
      },
      "svm": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "target": {
        "alias": "svm1",
        "name": "iqn.1992-
08.com.netapp:sn.574caf71890911e8a6b7005056b4ea79:vs.2"
      }
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

metric

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |

| Name | Type | Description |
| --- | --- | --- |
| latency | latency | The round trip latency in microseconds observed at the storage object. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

iops_raw

The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

| Name | Type | Description |
| --- | --- | --- |
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency_raw

The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput_raw

Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

statistics

These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

| Name | Type | Description |
|---|---|---|
| iops_raw | iops_raw | The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time. |
| latency_raw | latency_raw | The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput_raw | throughput_raw | Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time. |

| Name | Type | Description |
|---|---|---|
| timestamp | string | The timestamp of the performance data. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

target

| Name | Type | Description |
|---|---|---|
| alias | string | The iSCSI target alias of the iSCSI service.<br><br>The target alias can contain one (1) to 128 characters and feature any printable character except space (" "). A PATCH request with an empty alias ("") clears the alias.<br><br>Optional in POST and PATCH. In POST, this defaults to the name of the SVM. |

| Name | Type | Description |
|---|---|---|
| name | string | The iSCSI target name of the iSCSI service. This is generated for the SVM during POST.<br><br>If required, the target name can be modified using the ONTAP command line.<br><br>• example: iqn.1992-08.com.netapp:sn.574caf71890911e8a6b7005056b4ea79:vs.2<br>• maxLength: 128<br>• minLength: 1<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

iscsi_service

An iSCSI service defines the properties of the iSCSI target for an SVM. There can be at most one iSCSI service for an SVM. An SVM's iSCSI service must be created before iSCSI initiators can log in to the SVM.

An iSCSI service is identified by the UUID of its SVM.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| enabled | boolean | The administrative state of the iSCSI service. The iSCSI service can be disabled to block all iSCSI connectivity to the SVM.<br><br>Optional in POST and PATCH. The default setting is *true* (enabled) in POST. |
| metric | metric | Performance numbers, such as IOPS latency and throughput, for SVM protocols. |

| Name | Type | Description |
|---|---|---|
| statistics | statistics | These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target | target | |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 iSCSI service

POST `/protocols/san/iscsi/services`

**Introduced In:** 9.6

Creates an iSCSI service.

### Required properties

- `svm.uuid` or `svm.name` - Existing SVM in which to create the iSCSI service.

### Related ONTAP commands

- `vserver iscsi create`

### Learn more

- DOC /protocols/san/iscsi/services

### Parameters

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

### Request Body

| Name | Type | Description |
|---|---|---|
| enabled | boolean | The administrative state of the iSCSI service. The iSCSI service can be disabled to block all iSCSI connectivity to the SVM.<br><br>Optional in POST and PATCH. The default setting is *true* (enabled) in POST. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target | target | |

### Example request

```
{
  "svm": {
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "target": {
    "alias": "svm1",
    "name": "iqn.1992-
08.com.netapp:sn.574caf71890911e8a6b7005056b4ea79:vs.2"
  }
}
```

### Response

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| num_records | integer | The number of records in the response. |
| records | array[iscsi_service] | |

### Example response


```
{
  "num_records": 1,
  "records": [
    {
      "svm": {
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "target": {
        "alias": "svm1",
        "name": "iqn.1992-
08.com.netapp:sn.574caf71890911e8a6b7005056b4ea79:vs.2"
      }
    }
  ]
}
```


#### Headers

| Name | Description | Type |
|---|---|---|
| Location | Useful for tracking the resource location | string |

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 1115127 | The cluster lacks a valid iSCSI license. |
| 2621462 | The supplied SVM does not exist. |
| 2621507 | The iSCSI protocol is not allowed for the specified SVM. |
| 2621706 | The specified `svm.uuid` and `svm.name` do not refer to the same SVM. |
| 2621707 | No SVM was specified. Either `svm.name` or `svm.uuid` must be supplied. |
| 5373966 | An iSCSI service cannot be created in an SVM that is configured for NVMe. |

| Error Code | Description |
| --- | --- |
| 5374077 | An iSCSI service already exists for the specified SVM. |
| 5374893 | The SVM is stopped. The SVM must be running to create an iSCSI service. |
| 5376452 | Service POST and DELETE are not supported on ASA r2. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |

| Name | Type | Description |
|---|---|---|
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

metric

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

| Name | Type | Description |
|---|---|---|
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |

| Name | Type | Description |
| --- | --- | --- |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

iops_raw

The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

| Name | Type | Description |
| --- | --- | --- |
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency_raw

The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput_raw

Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

statistics

These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

| Name | Type | Description |
| --- | --- | --- |
| iops_raw | iops_raw | The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time. |
| latency_raw | latency_raw | The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput_raw | throughput_raw | Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time. |
| timestamp | string | The timestamp of the performance data. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

target

| Name | Type | Description |
|---|---|---|
| alias | string | The iSCSI target alias of the iSCSI service.<br><br>The target alias can contain one (1) to 128 characters and feature any printable character except space (" "). A PATCH request with an empty alias ("") clears the alias.<br><br>Optional in POST and PATCH. In POST, this defaults to the name of the SVM. |
| name | string | The iSCSI target name of the iSCSI service. This is generated for the SVM during POST.<br><br>If required, the target name can be modified using the ONTAP command line.<br><br>• example: iqn.1992-08.com.netapp:sn.574caf71890911e8a6b7005056b4ea79:vs.2<br>• maxLength: 128<br>• minLength: 1<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

iscsi_service

An iSCSI service defines the properties of the iSCSI target for an SVM. There can be at most one iSCSI service for an SVM. An SVM's iSCSI service must be created before iSCSI initiators can log in to the SVM.

An iSCSI service is identified by the UUID of its SVM.

| Name | Type | Description |
|---|---|---|
| enabled | boolean | The administrative state of the iSCSI service. The iSCSI service can be disabled to block all iSCSI connectivity to the SVM.<br><br>Optional in POST and PATCH. The default setting is *true* (enabled) in POST. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target | target | |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 iSCSI service

DELETE `/protocols/san/iscsi/services/{svm.uuid}`

**Introduced In:** 9.6

Deletes an iSCSI service. An iSCSI service must be disabled before it can be deleted.

### Related ONTAP commands

- `vserver iscsi delete`

### Learn more

- DOC /protocols/san/iscsi/services

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of the SVM for which to delete the iSCSI service. |

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | An SVM with the specified UUID does not exist. |
| 5373960 | The iSCSI service is enabled. The iSCSI service must be disabled before it can be deleted. |
| 5374078 | The SVM does not have an iSCSI service. |
| 5376452 | Service POST and DELETE are not supported on ASA r2. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 |

returned_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 iSCSI service

GET `/protocols/san/iscsi/services/{svm.uuid}`

**Introduced In:** 9.6

Retrieves an iSCSI service.

### Related ONTAP commands

- `vserver iscsi show`

### Learn more

- DOC /protocols/san/iscsi/services

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of the SVM for which to retrieve the iSCSI service. |
| fields | array[string] | query | False | Specify the fields to return. |

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| enabled | boolean | The administrative state of the iSCSI service. The iSCSI service can be disabled to block all iSCSI connectivity to the SVM.<br><br>Optional in POST and PATCH. The default setting is *true* (enabled) in POST. |
| metric | metric | Performance numbers, such as IOPS latency and throughput, for SVM protocols. |

| Name | Type | Description |
| --- | --- | --- |
| statistics | statistics | These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target | target | |

**Example response**


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "metric": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "duration": "PT15S",
    "iops": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "timestamp": "2017-01-25 06:20:13 -0500"
  },
  "statistics": {
    "iops_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput_raw": {
```

```
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "timestamp": "2017-01-25 06:20:13 -0500"
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "target": {
    "alias": "svm1",
    "name": "iqn.1992-
08.com.netapp:sn.574caf71890911e8a6b7005056b4ea79:vs.2"
  }
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | An SVM with the specified UUID does not exist. |
| 5374078 | The SVM does not have an iSCSI service. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

metric

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |

| Name | Type | Description |
| --- | --- | --- |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

iops_raw

The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

| Name | Type | Description |
| --- | --- | --- |
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency_raw

The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput_raw

Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

statistics

These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

| Name | Type | Description |
| --- | --- | --- |
| iops_raw | iops_raw | The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time. |
| latency_raw | latency_raw | The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput_raw | throughput_raw | Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time. |
| timestamp | string | The timestamp of the performance data. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

target

| Name | Type | Description |
|---|---|---|
| alias | string | The iSCSI target alias of the iSCSI service.<br><br>The target alias can contain one (1) to 128 characters and feature any printable character except space (" "). A PATCH request with an empty alias ("") clears the alias.<br><br>Optional in POST and PATCH. In POST, this defaults to the name of the SVM. |
| name | string | The iSCSI target name of the iSCSI service. This is generated for the SVM during POST.<br><br>If required, the target name can be modified using the ONTAP command line.<br><br>• example: iqn.1992-08.com.netapp:sn.574caf71890911e8a6b7005056b4ea79:vs.2<br>• maxLength: 128<br>• minLength: 1<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 iSCSI service

PATCH `/protocols/san/iscsi/services/{svm.uuid}`

**Introduced In:** 9.6

Updates an iSCSI service.

### Related ONTAP commands

- `vserver iscsi modify`
- `vserver iscsi start`
- `vserver iscsi stop`

### Learn more

- DOC /protocols/san/iscsi/services

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of the SVM for which to update the iSCSI service. |

### Request Body

| Name | Type | Description |
|---|---|---|
| enabled | boolean | The administrative state of the iSCSI service. The iSCSI service can be disabled to block all iSCSI connectivity to the SVM.<br><br>Optional in POST and PATCH. The default setting is *true* (enabled) in POST. |
| target | target | |

### Example request

```
{
  "target": {
    "alias": "svm1",
    "name": "iqn.1992-
08.com.netapp:sn.574caf71890911e8a6b7005056b4ea79:vs.2"
  }
}
```

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | An SVM with the specified UUID does not exist. |
| 5374078 | The SVM does not have an iSCSI service. |

Also see the table of common errors in the Response body overview section of this documentation.

### Definitions

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |

| Name | Type | Description |
|---|---|---|
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

metric

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

| Name | Type | Description |
|---|---|---|
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |

| Name | Type | Description |
| --- | --- | --- |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

iops_raw

The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

| Name | Type | Description |
| --- | --- | --- |
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency_raw

The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput_raw

Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

statistics

These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

| Name | Type | Description |
| --- | --- | --- |
| iops_raw | iops_raw | The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time. |
| latency_raw | latency_raw | The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput_raw | throughput_raw | Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time. |
| timestamp | string | The timestamp of the performance data. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

target

| Name | Type | Description |
|---|---|---|
| alias | string | The iSCSI target alias of the iSCSI service.<br><br>The target alias can contain one (1) to 128 characters and feature any printable character except space (" "). A PATCH request with an empty alias ("") clears the alias.<br><br>Optional in POST and PATCH. In POST, this defaults to the name of the SVM. |
| name | string | The iSCSI target name of the iSCSI service. This is generated for the SVM during POST.<br><br>If required, the target name can be modified using the ONTAP command line.<br><br>• example: iqn.1992-08.com.netapp:sn.574caf71890911e8a6b7005056b4ea79:vs.2<br>• maxLength: 128<br>• minLength: 1<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

iscsi_service

An iSCSI service defines the properties of the iSCSI target for an SVM. There can be at most one iSCSI service for an SVM. An SVM's iSCSI service must be created before iSCSI initiators can log in to the SVM.

An iSCSI service is identified by the UUID of its SVM.

| Name | Type | Description |
|---|---|---|
| enabled | boolean | The administrative state of the iSCSI service. The iSCSI service can be disabled to block all iSCSI connectivity to the SVM.<br><br>Optional in POST and PATCH. The default setting is *true* (enabled) in POST. |
| target | target | |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 historical performance metrics for the iSCSI protocol of an SVM

GET `/protocols/san/iscsi/services/{svm.uuid}/metrics`

**Introduced In:** 9.7

Retrieves historical performance metrics for the iSCSI protocol service of an SVM.

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| status | string | query | False | Filter by status |
| throughput.read | integer | query | False | Filter by throughput.read |
| throughput.total | integer | query | False | Filter by throughput.total |
| throughput.write | integer | query | False | Filter by throughput.write |
| iops.other | integer | query | False | Filter by iops.other |
| iops.total | integer | query | False | Filter by iops.total |
| iops.read | integer | query | False | Filter by iops.read |
| iops.write | integer | query | False | Filter by iops.write |
| latency.other | integer | query | False | Filter by latency.other |
| latency.total | integer | query | False | Filter by latency.total |
| latency.read | integer | query | False | Filter by latency.read |
| latency.write | integer | query | False | Filter by latency.write |
| timestamp | string | query | False | Filter by timestamp |
| duration | string | query | False | Filter by duration |
| svm.uuid | string | path | True | The unique identifier of the SVM. |

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| interval | string | query | False | The time range for the data. Examples can be 1h, 1d, 1m, 1w, 1y. The period for each time range is as follows:<br>• 1h: Metrics over the most recent hour sampled over 15 seconds.<br>• 1d: Metrics over the most recent day sampled over 5 minutes.<br>• 1w: Metrics over the most recent week sampled over 30 minutes.<br>• 1m: Metrics over the most recent month sampled over 2 hours.<br>• 1y: Metrics over the most recent year sampled over a day.<br>• Default value: 1<br>• enum: ["1h", "1d", "1w", "1m", "1y"] |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| 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.<br>• Default value: 15<br>• Max value: 120<br>• Min value: 0 |
| fields | array[string] | query | False | Specify the fields to return. |
| max_records | integer | query | False | Limit the number of records returned. |
| order_by | array[string] | query | False | Order results by specified fields and optional [asc |
| desc] direction. Default direction is 'asc' for ascending. | return_records | boolean | query | False |

**Response**

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| num_records | integer | Number of records |
| records | array[records] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "duration": "PT15S",
      "iops": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "latency": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "status": "ok",
      "svm": {
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "throughput": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "timestamp": "2017-01-25 06:20:13 -0500"
    }
  ]
}
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 8585947 | No metrics are available for the requested object. |
| 8586225 | An unexpected error occurred retrieving metrics for the requested object. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

svm

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the SVM. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

records

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
| --- | --- | --- |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| svm | svm | |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 historical performance metrics for the iSCSI protocol service of an SVM for a specific time

GET `/protocols/san/iscsi/services/{svm.uuid}/metrics/{timestamp}`

**Introduced In:** 9.14

Retrieves historical performance metrics for the iSCSI protocol service of an SVM for a specific time.

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of the SVM. |
| timestamp | string | path | True | The timestamp of the performance data.<br>• format: date-time |
| fields | array[string] | query | False | Specify the fields to return. |

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |
| status | string | Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| svm | svm | |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |

| Name | Type | Description |
|---|---|---|
| timestamp | string | The timestamp of the performance data. |

**Example response**


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "duration": "PT15S",
  "iops": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "latency": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "svm": {
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "throughput": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "timestamp": "2017-01-25 06:20:13 -0500"
}
```


**Error**

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 8585947 | No metrics are available for the requested object. |
| 8586225 | An unexpected error occurred retrieving metrics for the requested object. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

svm

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the SVM. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

# View iSCSI sessions

## View iSCSI sessions

### Overview

An iSCSI session is one or more TCP connections that link an iSCSI initiator with an iSCSI target. TCP connections can be added and removed from an iSCSI session by the iSCSI initiator. Across all TCP connections within an iSCSI session, an initiator sees one and the same target. After the connection is established, iSCSI control, data, and status messages are communicated over the session.

The iSCSI sessions REST API provides information about iSCSI initiators that have successfully logged in to ONTAP.

### Examples

#### Retrieving all iSCSI sessions

```
# The API:
GET /api/protocols/san/iscsi/sessions

# The call:
curl -X GET "https://<mgmt-ip>/api/protocols/san/iscsi/sessions" -H
"Accept: application/hal+json"

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "a009a9e7-4081-b576-7575-ada21efcaf16",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/a009a9e7-4081-b576-7575-ada21efcaf16"
        }
      }
    },
    "target_portal_group": "iscsi_lif1",
    "tsih": 10,
    "_links": {
```

```
      "self": {
        "href": "/api/protocols/san/iscsi/sessions/a009a9e7-4081-b576-
7575-ada21efcaf16/iscsi_lif1/10"
      }
    }
  },
  {
    "svm": {
      "uuid": "b009a9e7-4081-b576-7575-ada21efcaf16",
      "name": "svm2",
      "_links": {
        "self": {
          "href": "/api/svm/svms/b009a9e7-4081-b576-7575-ada21efcaf16"
        }
      }
    },
    "target_portal_group": "iscsi_lif2",
    "tsih": 11,
    "_links": {
      "self": {
        "href": "/api/protocols/san/iscsi/sessions/b009a9e7-4081-b576-
7575-ada21efcaf16/iscsi_lif2/11"
      }
    }
  }
],
"num_records": 2,
"_links": {
  "self": {
    "href": "/api/protocols/san/iscsi/sessions"
  }
}
}
```

#### Retrieving all of the iSCSI sessions under the target portal group *iscsi_lif1*

The `tpgroup` query parameter is used to perform the query.

```
# The API:
GET /api/protocols/san/iscsi/sessions

# The call:
curl -X GET "https://<mgmt-
ip>/api/protocols/san/iscsi/sessions?tpgroup=iscsi_lif1" -H "Accept:
application/hal+json"

# The response:
{
"records": [
  {
    "svm": {
      "uuid": "a009a9e7-4081-b576-7575-ada21efcaf16",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/a009a9e7-4081-b576-7575-ada21efcaf16"
        }
      }
    },
    "target_portal_group": "iscsi_lif1",
    "tsih": 10,
    "_links": {
      "self": {
        "href": "/api/protocols/san/iscsi/sessions/a009a9e7-4081-b576-
7575-ada21efcaf16/iscsi_lif1/10"
      }
    }
  }
],
"num_records": 1,
"_links": {
  "self": {
    "href": "/api/protocols/san/iscsi/sessions"
  }
}
}
```

#### Retrieving an iSCSI session

```
# The API:
GET
```

```
/api/protocols/san/iscsi/sessions/{svm.uuid}/{target_portal_group}/{tsih}

# The call:
curl -X GET "https://<mgmt-ip>/api/protocols/san/iscsi/sessions/a009a9e7-
4081-b576-7575-ada21efcaf16/iscsi_lif1/10" -H "Accept:
application/hal+json"

# The response:
{
"svm": {
  "uuid": "a009a9e7-4081-b576-7575-ada21efcaf16",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/a009a9e7-4081-b576-7575-ada21efcaf16"
    }
  }
},
"target_portal_group": "iscsi_lif1",
"tsih": 10,
"initiator": {
  "name": "iqn.1994-05.com.example:string",
  "comment": "Example information about this initiator"
},
"isid": "61:62:63:64:65:00",
"target_portal_group_tag": 1027,
"connections": [
  {
    "cid": 1,
    "authentication_type": "chap",
    "initiator_address": {
      "address": "10.224.123.85",
      "port": 43827
    },
    "interface": {
      "name": "iscsi_lif1",
      "uuid": "c15439b4-dbb4-11e8-90ac-005056bba882",
      "ip": {
        "address": "192.168.0.1",
        "port": 3260
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c15439b4-dbb4-11e8-90ac-
005056bba882"
        }
```

```
      }
    }
  }
],
"igroups": [
  {
    "uuid": "af7838cd-f993-4faf-90b7-5524787ae1e8",
    "name": "igroup1",
    "_links": {
      "self": {
        "href": "/api/protocols/san/igroups/af7838cd-f993-4faf-90b7-
5524787ae1e8"
      }
    }
  },
  {
    "uuid": "bf7838cd-f993-4faf-90b7-5524787ae1e8",
    "name": "igroup2",
    "_links": {
      "self": {
        "href": "/api/protocols/san/igroups/bf7838cd-f993-4faf-90b7-
5524787ae1e8"
      }
    }
  }
],
"_links": {
  "self": {
    "href": "/api/protocols/san/iscsi/sessions/a009a9e7-4081-b576-7575-
ada21efcaf16/iscsi_lif1/10"
  }
}
}
```


## Retrieve iSCSI sessions

GET `/protocols/san/iscsi/sessions`

**Introduced In:** 9.6

Retrieves iSCSI sessions.

### Related ONTAP commands

- `vserver iscsi connection show`
- `vserver iscsi session parameter show`

- `vserver iscsi session show`

### Learn more

- DOC /protocols/san/iscsi/sessions

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| tsih | integer | query | False | Filter by tsih |
| isid | string | query | False | Filter by isid |
| target_portal_group_tag | integer | query | False | Filter by target_portal_group_tag |
| connections.authentication_type | string | query | False | Filter by connections.authentication_type |
| connections.cid | integer | query | False | Filter by connections.cid |
| connections.interface.ip.port | integer | query | False | Filter by connections.interface.ip.port<br>• Max value: 65536<br>• Min value: 1 |
| connections.interface.ip.address | string | query | False | Filter by connections.interface.ip.address |
| connections.interface.uuid | string | query | False | Filter by connections.interface.uuid |
| connections.interface.name | string | query | False | Filter by connections.interface.name |
| connections.initiator_address.address | string | query | False | Filter by connections.initiator_address.address |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| connections.initiator_address.port | integer | query | False | Filter by connections.initiator_address.port |
| initiator.name | string | query | False | Filter by initiator.name |
| initiator.alias | string | query | False | Filter by initiator.alias |
| initiator.comment | string | query | False | Filter by initiator.comment<br>• Introduced in: 9.9 |
| target_portal_group | string | query | False | Filter by target_portal_group |
| igroups.uuid | string | query | False | Filter by igroups.uuid |
| igroups.name | string | query | False | Filter by igroups.name<br>• maxLength: 96<br>• minLength: 1 |
| svm.name | string | query | False | Filter by svm.name |
| svm.uuid | string | query | False | Filter by svm.uuid |
| 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.<br>• Default value: 1 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| 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.<br>• Max value: 120<br>• Min value: 0<br>• Default value: 15 |
| 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[iscsi_session] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "connections": [
        {
          "_links": {
            "next": {
              "href": "/api/resourcelink"
            },
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "authentication_type": "string",
          "cid": 0,
          "initiator_address": {
            "address": "10.10.10.7",
            "port": 55432
          },
          "interface": {
            "_links": {
              "self": {
                "href": "/api/resourcelink"
              }
            },
            "ip": {
              "address": "10.10.10.7",
              "port": 3260
            },
            "name": "lif1",
```

```
            "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
          }
        }
      ],
      "igroups": [
        {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "igroup1",
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        }
      ],
      "initiator": {
        "alias": "initiator_alias1",
        "comment": "This is an iSCSI initiator for host 5",
        "name": "iqn.1992-01.example.com:string"
      },
      "isid": "61:62:63:64:65:00",
      "svm": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "target_portal_group": "tpgroup1",
      "target_portal_group_tag": 0,
      "tsih": 0
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

initiator_address

The TCP socket information for the initiator end of the connection. This is useful for network packet debugging.

| Name | Type | Description |
|---|---|---|
| address | string | The TCP IPv4 or IPv6 address of the initiator end of the iSCSI connection. |
| port | integer | The TCP port number of the initiator end of the iSCSI connection. |

ip

The IP information. ONTAP only supports port 3260.

| Name | Type | Description |
|---|---|---|
| address | string | IPv4 or IPv6 address |
| port | integer | The TCP port number of the iSCSI access endpoint. |

interface

The network interface information for the target end of the connection.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| ip | ip | The IP information. ONTAP only supports port 3260.<br>• readOnly: 1<br>• Introduced in: 9.6 |
| name | string | The name of the interface. |
| uuid | string | The UUID that uniquely identifies the interface. |

iscsi_connection

An active iSCSI connection.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| authentication_type | string | The iSCSI authentication type used to establish the connection. |
| cid | integer | The identifier of the connection within the session. |
| initiator_address | initiator_address | The TCP socket information for the initiator end of the connection. This is useful for network packet debugging. |
| interface | interface | The network interface information for the target end of the connection. |

igroups

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

initiator

The initiator that created the session.

| Name | Type | Description |
|---|---|---|
| alias | string | The initiator alias. |
| comment | string | A comment available for use by the administrator. This is modifiable from the initiator REST endpoint directly. See PATCH /protocols/san/igroups/{igroup.uuid}/initiators/{name} . |
| name | string | The world wide unique name of the initiator. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

iscsi_session

An iSCSI session is one or more TCP connections that link an iSCSI initiator with an iSCSI target. TCP connections can be added and removed from an iSCSI session by the iSCSI initiator. Across all TCP connections within an iSCSI session, an initiator sees one and the same target. After the connection is established, iSCSI control, data, and status messages are communicated over the session.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| connections | array[iscsi_connection] | The iSCSI connections that make up the iSCSI session. |
| igroups | array[igroups] | The initiator groups in which the initiator is a member. |
| initiator | initiator | The initiator that created the session. |

| Name | Type | Description |
|---|---|---|
| isid | string | The initiator portion of the session identifier specified by the initiator during login. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target_portal_group | string | The target portal group to which the session belongs. |
| target_portal_group_tag | integer | The target portal group tag of the session. |
| tsih | integer | The target session identifier handle (TSIH) of the session. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 iSCSI session

GET `/protocols/san/iscsi/sessions/{svm.uuid}/{tpgroup}/{tsih}`

**Introduced In:** 9.6

Retrieves an iSCSI session.

### Related ONTAP commands

- `vserver iscsi connection show`
- `vserver iscsi session parameter show`
- `vserver iscsi session show`

### Learn more

- DOC /protocols/san/iscsi/sessions

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.uuid | string | path | True | The unique identifier of the SVM of the iSCSI session. |
| tpgroup | string | path | True | The target portal group of the iSCSI session. |
| tsih | integer | path | True | The target session identifying handle. |
| fields | array[string] | query | False | Specify the fields to return. |

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| connections | array[iscsi_connection] | The iSCSI connections that make up the iSCSI session. |
| igroups | array[igroups] | The initiator groups in which the initiator is a member. |
| initiator | initiator | The initiator that created the session. |

| Name | Type | Description |
| --- | --- | --- |
| isid | string | The initiator portion of the session identifier specified by the initiator during login. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| target_portal_group | string | The target portal group to which the session belongs. |
| target_portal_group_tag | integer | The target portal group tag of the session. |
| tsih | integer | The target session identifier handle (TSIH) of the session. |

**Example response**


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "connections": [
    {
      "_links": {
        "next": {
          "href": "/api/resourcelink"
        },
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "authentication_type": "string",
      "cid": 0,
      "initiator_address": {
        "address": "10.10.10.7",
        "port": 55432
      },
      "interface": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "ip": {
          "address": "10.10.10.7",
          "port": 3260
        },
        "name": "lif1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      }
    }
  ],
  "igroups": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
```

```
      "name": "igroup1",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "initiator": {
    "alias": "initiator_alias1",
    "comment": "This is an iSCSI initiator for host 5",
    "name": "iqn.1992-01.example.com:string"
  },
  "isid": "61:62:63:64:65:00",
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "target_portal_group": "tpgroup1",
  "target_portal_group_tag": 0,
  "tsih": 0
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 2621462 | An SVM with the specified UUID does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

_links

| Name | Type | Description |
|---|---|---|
| next | href | |
| self | href | |

initiator_address

The TCP socket information for the initiator end of the connection. This is useful for network packet debugging.

| Name | Type | Description |
|---|---|---|
| address | string | The TCP IPv4 or IPv6 address of the initiator end of the iSCSI connection. |
| port | integer | The TCP port number of the initiator end of the iSCSI connection. |

ip

The IP information. ONTAP only supports port 3260.

| Name | Type | Description |
|---|---|---|
| address | string | IPv4 or IPv6 address |
| port | integer | The TCP port number of the iSCSI access endpoint. |

interface

The network interface information for the target end of the connection.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| ip | ip | The IP information. ONTAP only supports port 3260.<br>• readOnly: 1<br>• Introduced in: 9.6 |
| name | string | The name of the interface. |
| uuid | string | The UUID that uniquely identifies the interface. |

iscsi_connection

An active iSCSI connection.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| authentication_type | string | The iSCSI authentication type used to establish the connection. |
| cid | integer | The identifier of the connection within the session. |
| initiator_address | initiator_address | The TCP socket information for the initiator end of the connection. This is useful for network packet debugging. |
| interface | interface | The network interface information for the target end of the connection. |

igroups

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

initiator

The initiator that created the session.

| Name | Type | Description |
|---|---|---|
| alias | string | The initiator alias. |
| comment | string | A comment available for use by the administrator. This is modifiable from the initiator REST endpoint directly. See PATCH /protocols/san/igroups/{igroup.uuid}/initiators/{name} . |
| name | string | The world wide unique name of the initiator. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

# Manage SAN LUN maps

## Manage SAN LUN maps

### Overview

A LUN map is an association between a LUN and an initiator group. When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between an initiator group and a LUN is many initiator groups to many LUNs.

A LUN map also configures the cluster nodes from which network paths to the LUN are advertised via the SAN protocols as part of the Selective LUN Map (SLM) functionality of ONTAP. These nodes are referred to as the reporting nodes of a LUN map. For further information, see DOC /protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes .

The LUN map REST API allows you to create, delete, and discover LUN maps, and manage the reporting nodes of a LUN map.

### Examples

#### Creating a LUN map

```
# The API:
POST /api/protocols/san/lun-maps

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/san/lun-maps' -H 'Accept:
application/hal+json' -d '{ "svm": { "name": "svm1" }, "igroup": { "name":
"igroup1" }, "lun": { "name": "/vol/vol1/lun1" } }'
```

#### Retrieving all of the LUN maps

```
# The API:
GET /api/protocols/san/lun-maps

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/lun-maps' -H 'Accept:
application/hal+json'

# The response:
```

```
{
"records": [
  {
    "svm": {
      "uuid": "03157e81-24c5-11e9-9ec1-005056bba643",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/03157e81-24c5-11e9-9ec1-005056bba643"
        }
      }
    },
    "lun": {
      "uuid": "a60d9862-9bee-49a6-8162-20d2421bb1a6",
      "name": "/vol/vol1/lun1",
      "_links": {
        "self": {
          "href": "/api/storage/luns/a60d9862-9bee-49a6-8162-20d2421bb1a6"
        }
      }
    },
    "igroup": {
      "uuid": "40d98b2c-24c5-11e9-9ec1-005056bba643",
      "name": "ig1",
      "_links": {
        "self": {
          "href": "/api/protocols/san/igroups/40d98b2c-24c5-11e9-9ec1-
005056bba643"
        }
      }
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/lun-maps/a60d9862-9bee-49a6-8162-
20d2421bb1a6/40d98b2c-24c5-11e9-9ec1-005056bba643"
      }
    }
  }
],
"num_records": 1,
"_links": {
  "self": {
    "href": "/api/protocols/san/lun-maps"
  }
}
}
```

**Retrieving a specific LUN map**

```
# The API:
GET /api/protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/lun-maps/a60d9862-9bee-
49a6-8162-20d2421bb1a6/40d98b2c-24c5-11e9-9ec1-005056bba643' -H 'Accept:
application/hal+json'

# The response:
{
"svm": {
  "uuid": "03157e81-24c5-11e9-9ec1-005056bba643",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/03157e81-24c5-11e9-9ec1-005056bba643"
    }
  }
},
"lun": {
  "uuid": "a60d9862-9bee-49a6-8162-20d2421bb1a6",
  "name": "/vol/vol1/lun1",
  "node": {
    "uuid": "7d8607ea-24c1-11e9-9ec1-005056bba643",
    "name": "node1",
    "_links": {
      "self": {
        "href": "/api/cluster/nodes/7d8607ea-24c1-11e9-9ec1-005056bba643"
      }
    }
  },
  "smbc": {
    "replicated": false
  },
  "_links": {
    "self": {
      "href": "/api/storage/luns/a60d9862-9bee-49a6-8162-20d2421bb1a6"
    }
  }
},
"igroup": {
  "uuid": "40d98b2c-24c5-11e9-9ec1-005056bba643",
  "name": "ig1",
  "os_type": "linux",
```

```
  "protocol": "mixed",
  "replicated": false,
  "_links": {
    "self": {
      "href": "/api/protocols/san/igroups/40d98b2c-24c5-11e9-9ec1-
005056bba643"
    }
  }
},
"logical_unit_number": 0,
"reporting_nodes": [
  {
    "uuid": "11a465f5-2ac0-11eb-a303-005056bb1e81",
    "name": "node1",
    "_links": {
      "node": {
        "href": "/cluster/nodes/11a465f5-2ac0-11eb-a303-005056bb1e81"
      },
      "self": {
        "href": "/api/protocols/san/lun-maps/a60d9862-9bee-49a6-8162-
20d2421bb1a6/40d98b2c-24c5-11e9-9ec1-005056bba643/reporting-
nodes/11a465f5-2ac0-11eb-a303-005056bb1e81"
      }
    }
  },
  {
    "uuid": "6c7cb50f-2abf-11eb-9840-005056bbd490",
    "name": "node2",
    "_links": {
      "node": {
        "href": "/cluster/nodes/6c7cb50f-2abf-11eb-9840-005056bbd490"
      },
      "self": {
        "href": "/api/protocols/san/lun-maps/a60d9862-9bee-49a6-8162-
20d2421bb1a6/40d98b2c-24c5-11e9-9ec1-005056bba643/reporting-
nodes/6c7cb50f-2abf-11eb-9840-005056bbd490"
      }
    }
  }
],
"_links": {
  "self": {
    "href": "/api/protocols/san/lun-maps/a60d9862-9bee-49a6-8162-
20d2421bb1a6/40d98b2c-24c5-11e9-9ec1-005056bba643"
  }
}
```

```
}
```

#### Deleting a LUN map

```
# The API:
DELETE /api/protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/lun-maps/a60d9862-
9bee-49a6-8162-20d2421bb1a6/40d98b2c-24c5-11e9-9ec1-005056bba643' -H
'Accept: application/hal+json'
```

## Retrieve LUN maps

GET `/protocols/san/lun-maps`

**Introduced In:** 9.6

Retrieves LUN maps.

### Related ONTAP commands

- `lun mapping show`

### Learn more

- DOC /protocols/san/lun-maps

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | query | False | Filter by lun.uuid |
| lun.name | string | query | False | Filter by lun.name |
| lun.node.name | string | query | False | Filter by lun.node.name |
| lun.node.uuid | string | query | False | Filter by lun.node.uuid |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.smbc.replicated | boolean | query | False | Filter by lun.smbc.replicated<br>• Introduced in: 9.15 |
| svm.name | string | query | False | Filter by svm.name |
| svm.uuid | string | query | False | Filter by svm.uuid |
| igroup.uuid | string | query | False | Filter by igroup.uuid |
| igroup.name | string | query | False | Filter by igroup.name |
| igroup.protocol | string | query | False | Filter by igroup.protocol |
| igroup.os_type | string | query | False | Filter by igroup.os_type |
| igroup.replicated | boolean | query | False | Filter by igroup.replicated<br>• Introduced in: 9.15 |
| igroup.initiators | string | query | False | Filter by igroup.initiators |
| reporting_nodes.uuid | string | query | False | Filter by reporting_nodes.uuid<br>• Introduced in: 9.10 |
| reporting_nodes.name | string | query | False | Filter by reporting_nodes.name<br>• Introduced in: 9.10 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| logical_unit_number | integer | query | False | Filter by logical_unit_number<br>• Max value: 4095<br>• Min value: 0 |
| 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.<br>• 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.<br>• Max value: 120<br>• Min value: 0<br>• Default value: 15 |
| 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[lun_map] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "igroup": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "initiators": [
          "iqn.1998-01.com.corp.iscsi:name1"
        ],
        "name": "igroup1",
        "os_type": "string",
        "protocol": "string",
        "uuid": "1ad8544d-8cd1-91e0-9e1c-723478563412"
      },
      "logical_unit_number": 1,
      "lun": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "/vol/volume1/qtree1/lun1",
        "node": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
```

```
          },
          "name": "node1",
          "uuid": "1cf8aa42-8cd1-12e0-a11c-423468563412"
        },
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "reporting_nodes": [
        {
          "_links": {
            "node": {
              "href": "/api/resourcelink"
            },
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "node1",
          "uuid": "5ac8eb9c-4e32-dbaa-57ca-fb905976f54e"
        }
      ],
      "svm": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      }
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

igroup

The initiator group to which the LUN is mapped. Required in POST by supplying either the `igroup.uuid`, `igroup.name`, or both.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| initiators | array[string] | The initiators that are members of the initiator group. |
| name | string | The name of the initiator group. Valid in POST. |
| os_type | string | The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. |
| protocol | string | The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. |
| replicated | boolean | This property reports if the initiator group is replicated. |
| uuid | string | The unique identifier of the initiator group. Valid in POST. |

node

The LUN node.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the LUN's node. |
| uuid | string | The unique identifier of the LUN node. |

smbc

"Properties related to SM-BC replication."

| Name | Type | Description |
|---|---|---|
| replicated | boolean | This property reports if the LUN is replicated via SM-BC. |

lun

The LUN to which the initiator group is mapped. Required in POST by supplying either the `lun.uuid`, `lun.name`, or both.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the LUN. Valid in POST. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<namespace>" where the qtree name is optional. |
| node | node | The LUN node. |
| smbc | smbc | "Properties related to SM-BC replication." |
| uuid | string | The unique identifier of the LUN. Valid in POST. |

_links

| Name | Type | Description |
|---|---|---|
| node | href | |
| self | href | |

reporting_nodes

A cluster node from which network paths to the LUN are advertised by ONTAP via the SAN protocols.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the node.<br><br>Either `uuid` or `name` are required in POST. |
| uuid | string | The unique identifier of the node.<br><br>Either `uuid` or `name` are required in POST. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

lun_map

A LUN map is an association between a LUN and an initiator group. When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
| --- | --- | --- |
| igroup | igroup | The initiator group to which the LUN is mapped. Required in POST by supplying either the `igroup.uuid`, `igroup.name`, or both. |
| logical_unit_number | integer | The logical unit number assigned to the LUN when mapped to the specified initiator group. The number is used to identify the LUN to initiators in the initiator group when communicating through Fibre Channel Protocol or iSCSI. Optional in POST; if no value is provided, ONTAP assigns the lowest available value.<br>• example: 1<br>• Max value: 4095<br>• Min value: 0<br>• Introduced in: 9.6<br>• readCreate: 1<br>• x-nullable: true |
| lun | lun | The LUN to which the initiator group is mapped. Required in POST by supplying either the `lun.uuid`, `lun.name`, or both. |

| Name | Type | Description |
|---|---|---|
| reporting_nodes | array[reporting_nodes] | The cluster nodes from which network paths to the mapped LUNs are advertised via the SAN protocols as part of the Selective LUN Map (SLM) feature of ONTAP.<br><br>When a LUN map is created, the cluster node hosting the LUN and its high availability (HA) partner are set as the default reporting node. In POST, the property `additional_reporting_node` may be used to add an additional node and its HA partner.<br><br>For further information, see DOC /protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes .<br><br>• readOnly: 1<br>• Introduced in: 9.10 |
| svm | svm | SVM, applies only to SVM-scoped objects. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 a LUN map

POST `/protocols/san/lun-maps`

**Introduced In:** 9.6

Creates a LUN map.

### Required properties

- `svm.uuid` or `svm.name` - Existing SVM in which to create the LUN map.
- `igroup.uuid` or `igroup.name` - Existing initiator group to map to the specified LUN.
- `lun.uuid` or `lun.name` - Existing LUN to map to the specified initiator group.

### Default property values

If not specified in POST, the following default property values are assigned.

- `logical_unit_number` - If no value is provided, ONTAP assigns the lowest available value.

### Related ONTAP commands

- `lun mapping create`

### Learn more

- DOC /protocols/san/lun-maps

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| additional_reporting_node.uuid | string | query | False | The unique identifier of an ONTAP cluster node to add to the default reporting nodes for the LUN map. The HA partner for the node is also added.<br>• Introduced in: 9.10 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| additional_reporting_node.name | string | query | False | The name of an ONTAP cluster node to add to the default reporting nodes for the LUN map. The HA partner for the node is also added.<br>• Introduced in: 9.10 |
| return_records | boolean | query | False | The default is false. If set to true, the records are returned.<br>• Default value: |

**Request Body**

| Name | Type | Description |
|---|---|---|
| igroup | igroup | The initiator group to which the LUN is mapped. Required in POST by supplying either the `igroup.uuid`, `igroup.name`, or both. |
| logical_unit_number | integer | The logical unit number assigned to the LUN when mapped to the specified initiator group. The number is used to identify the LUN to initiators in the initiator group when communicating through Fibre Channel Protocol or iSCSI. Optional in POST; if no value is provided, ONTAP assigns the lowest available value.<br>• example: 1<br>• Max value: 4095<br>• Min value: 0<br>• Introduced in: 9.6<br>• readCreate: 1<br>• x-nullable: true |

| Name | Type | Description |
| --- | --- | --- |
| lun | lun | The LUN to which the initiator group is mapped. Required in POST by supplying either the `lun.uuid`, `lun.name`, or both. |
| reporting_nodes | array[reporting_nodes] | The cluster nodes from which network paths to the mapped LUNs are advertised via the SAN protocols as part of the Selective LUN Map (SLM) feature of ONTAP.<br><br>When a LUN map is created, the cluster node hosting the LUN and its high availability (HA) partner are set as the default reporting node. In POST, the property `additional_reporting_node` may be used to add an additional node and its HA partner.<br><br>For further information, see DOC /protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes .<br><br>• readOnly: 1<br>• Introduced in: 9.10 |
| svm | svm | SVM, applies only to SVM-scoped objects. |

### Example request


```
{
  "igroup": {
    "initiators": [
      "iqn.1998-01.com.corp.iscsi:name1"
    ],
    "name": "igroup1",
    "os_type": "string",
    "protocol": "string",
    "uuid": "1ad8544d-8cd1-91e0-9e1c-723478563412"
  },
  "logical_unit_number": 1,
  "lun": {
    "name": "/vol/volume1/qtree1/lun1",
    "node": {
      "name": "node1",
      "uuid": "1cf8aa42-8cd1-12e0-a11c-423468563412"
    },
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "reporting_nodes": [
    {
      "name": "node1",
      "uuid": "5ac8eb9c-4e32-dbaa-57ca-fb905976f54e"
    }
  ],
  "svm": {
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  }
}
```


### Response

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| num_records | integer | The number of records in the response. |
| records | array[lun_map] | |

### Example response


```
{
  "num_records": 1,
  "records": [
    {
      "igroup": {
        "initiators": [
          "iqn.1998-01.com.corp.iscsi:name1"
        ],
        "name": "igroup1",
        "os_type": "string",
        "protocol": "string",
        "uuid": "1ad8544d-8cd1-91e0-9e1c-723478563412"
      },
      "logical_unit_number": 1,
      "lun": {
        "name": "/vol/volume1/qtree1/lun1",
        "node": {
          "name": "node1",
          "uuid": "1cf8aa42-8cd1-12e0-a11c-423468563412"
        },
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "reporting_nodes": [
        {
          "name": "node1",
          "uuid": "5ac8eb9c-4e32-dbaa-57ca-fb905976f54e"
        }
      ],
      "svm": {
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      }
    }
  ]
}
```


### Headers

| Name | Description | Type |
|---|---|---|
| Location | Useful for tracking the resource location | string |

**Error**

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 1254201 | The specified LUN does not exist. |
| 1254207 | The LUN is already mapped to the same initiator group. |
| 2621462 | The specified SVM does not exist or is not accessible to the caller. |
| 2621706 | Both the SVM UUID and SVM name were supplied, but don’t refer to the same SVM. |
| 2621707 | No SVM was specified. Either `svm.name` or `svm.uuid` must be supplied. |
| 5374053 | The LUN is the destination of an ongoing restore operation and is inaccessible for I/O and management. Wait for the restore to complete and try the command again. |
| 5374127 | The specified LUN name is invalid. |
| 5374238 | The operation is not allowed on a LUN in a snapshot. |
| 5374316 | A LUN move operation is in progress on the source LUN. |
| 5374329 | A LUN of class `vvol` cannot be mapped. |
| 5374573 | A node has no interface configured with the iSCSI or Fibre Channel protocols for the specified SVM. |
| 5374574 | Multiple nodes have no interface configured with the iSCSI or Fibre Channel protocols for the specified SVM. |
| 5374581 | A node has no interface configured with the iSCSI protocol for the specified SVM. |
| 5374582 | Multiple nodes have no interface configured with the iSCSI protocol for the specified SVM. |
| 5374583 | A node has no interface configured with the Fibre Channel protocol for the specified SVM. |
| 5374584 | Multiple nodes have no interface configured with the Fibre Channel protocol for the specified SVM. |
| 5374652 | An error was reported by the peer cluster while creating a replicated LUN map. The specific error will be included as a nested error. |

| Error Code | Description |
| --- | --- |
| 5374901 | Either `lun.uuid` or `lun.name` must be provided to create a LUN map. |
| 5374902 | Either `igroup.uuid` or `igroup.name` must be provided to create a LUN map. |
| 5374920 | The cluster node specified in additional_reporting_node does not exist. |
| 5374921 | The cluster node name and UUID specified in additional_reporting_node do not refer to the same cluster node. |
| 5376356 | The LUN is part of a SnapMirror active sync relationship that has a different SVM peering relationship than the igroup. |
| 5376461 | The specified LUN name is invalid. |
| 5376462 | The specified LUN name is too long. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

igroup

The initiator group to which the LUN is mapped. Required in POST by supplying either the `igroup.uuid`, `igroup.name`, or both.

| Name | Type | Description |
|---|---|---|
| initiators | array[string] | The initiators that are members of the initiator group. |
| name | string | The name of the initiator group. Valid in POST. |
| os_type | string | The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. |
| protocol | string | The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. |
| replicated | boolean | This property reports if the initiator group is replicated. |
| uuid | string | The unique identifier of the initiator group. Valid in POST. |

node

The LUN node.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the LUN’s node. |
| uuid | string | The unique identifier of the LUN node. |

smbc

"Properties related to SM-BC replication."

| Name | Type | Description |
|---|---|---|
| replicated | boolean | This property reports if the LUN is replicated via SM-BC. |

lun

The LUN to which the initiator group is mapped. Required in POST by supplying either the `lun.uuid`, `lun.name`, or both.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the LUN. Valid in POST. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<namespace>" where the qtree name is optional. |
| node | node | The LUN node. |
| smbc | smbc | "Properties related to SM-BC replication." |
| uuid | string | The unique identifier of the LUN. Valid in POST. |

reporting_nodes

A cluster node from which network paths to the LUN are advertised by ONTAP via the SAN protocols.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the node.<br><br>Either `uuid` or `name` are required in POST. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the node.<br><br>Either `uuid` or `name` are required in POST. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

lun_map

A LUN map is an association between a LUN and an initiator group. When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.

| Name | Type | Description |
|---|---|---|
| igroup | igroup | The initiator group to which the LUN is mapped. Required in POST by supplying either the `igroup.uuid`, `igroup.name`, or both. |

| Name | Type | Description |
| --- | --- | --- |
| logical_unit_number | integer | The logical unit number assigned to the LUN when mapped to the specified initiator group. The number is used to identify the LUN to initiators in the initiator group when communicating through Fibre Channel Protocol or iSCSI. Optional in POST; if no value is provided, ONTAP assigns the lowest available value.<br>• example: 1<br>• Max value: 4095<br>• Min value: 0<br>• Introduced in: 9.6<br>• readCreate: 1<br>• x-nullable: true |
| lun | lun | The LUN to which the initiator group is mapped. Required in POST by supplying either the `lun.uuid`, `lun.name`, or both. |
| reporting_nodes | array[reporting_nodes] | The cluster nodes from which network paths to the mapped LUNs are advertised via the SAN protocols as part of the Selective LUN Map (SLM) feature of ONTAP.<br><br>When a LUN map is created, the cluster node hosting the LUN and its high availability (HA) partner are set as the default reporting node. In POST, the property `additional_reporting_node` may be used to add an additional node and its HA partner.<br><br>For further information, see DOC /protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes .<br>• readOnly: 1<br>• Introduced in: 9.10 |

| Name | Type | Description |
|---|---|---|
| svm | svm | SVM, applies only to SVM-scoped objects. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 a LUN map

DELETE `/protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}`

**Introduced In:** 9.6

Deletes a LUN map.

### Related ONTAP commands

- `lun mapping delete`

### Learn more

- DOC /protocols/san/lun-maps

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | |
| igroup.uuid | string | path | True | |

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374875 | The specified LUN does not exist or is not accessible to the caller. |
| 5374878 | The specified initiator group does not exist, is not accessible to the caller, or is not in the same SVM as the specified LUN. |
| 5374922 | The specified LUN map does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 |

returned_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 a LUN map

GET `/protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}`

**Introduced In:** 9.6

Retrieves a LUN map.

**Related ONTAP commands**

- `lun mapping show`

**Learn more**

- DOC /protocols/san/lun-maps

**Parameters**

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | The unique identifier of the LUN. |
| igroup.uuid | string | path | True | The unique identifier of the igroup. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| fields | array[string] | query | False | Specify the fields to return. |

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| igroup | igroup | The initiator group to which the LUN is mapped. Required in POST by supplying either the `igroup.uuid`, `igroup.name`, or both. |
| logical_unit_number | integer | The logical unit number assigned to the LUN when mapped to the specified initiator group. The number is used to identify the LUN to initiators in the initiator group when communicating through Fibre Channel Protocol or iSCSI. Optional in POST; if no value is provided, ONTAP assigns the lowest available value.<br>• example: 1<br>• Max value: 4095<br>• Min value: 0<br>• Introduced in: 9.6<br>• readCreate: 1<br>• x-nullable: true |
| lun | lun | The LUN to which the initiator group is mapped. Required in POST by supplying either the `lun.uuid`, `lun.name`, or both. |

| Name | Type | Description |
| --- | --- | --- |
| reporting_nodes | array[reporting_nodes] | The cluster nodes from which network paths to the mapped LUNs are advertised via the SAN protocols as part of the Selective LUN Map (SLM) feature of ONTAP.<br><br>When a LUN map is created, the cluster node hosting the LUN and its high availability (HA) partner are set as the default reporting node. In POST, the property `additional_reporting_node` may be used to add an additional node and its HA partner.<br><br>For further information, see DOC /protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes .<br><br>• readOnly: 1<br>• Introduced in: 9.10 |
| svm | svm | SVM, applies only to SVM-scoped objects. |

**Example response**


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "igroup": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "initiators": [
      "iqn.1998-01.com.corp.iscsi:name1"
    ],
    "name": "igroup1",
    "os_type": "string",
    "protocol": "string",
    "uuid": "1ad8544d-8cd1-91e0-9e1c-723478563412"
  },
  "logical_unit_number": 1,
  "lun": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "/vol/volume1/qtree1/lun1",
    "node": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "node1",
      "uuid": "1cf8aa42-8cd1-12e0-a11c-423468563412"
    },
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "reporting_nodes": [
    {
      "_links": {
        "node": {
          "href": "/api/resourcelink"
```

```
        },
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "node1",
      "uuid": "5ac8eb9c-4e32-dbaa-57ca-fb905976f54e"
    }
  ],
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  }
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374852 | The initiator group does not exist or is not accessible to the caller. |
| 5374875 | The LUN does not exist or is not accessible to the caller. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

igroup

The initiator group to which the LUN is mapped. Required in POST by supplying either the `igroup.uuid`, `igroup.name`, or both.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| initiators | array[string] | The initiators that are members of the initiator group. |
| name | string | The name of the initiator group. Valid in POST. |
| os_type | string | The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. |
| protocol | string | The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. |
| replicated | boolean | This property reports if the initiator group is replicated. |
| uuid | string | The unique identifier of the initiator group. Valid in POST. |

node

The LUN node.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
|---|---|---|
| name | string | The name of the LUN's node. |
| uuid | string | The unique identifier of the LUN node. |

smbc

"Properties related to SM-BC replication."

| Name | Type | Description |
|---|---|---|
| replicated | boolean | This property reports if the LUN is replicated via SM-BC. |

lun

The LUN to which the initiator group is mapped. Required in POST by supplying either the `lun.uuid`, `lun.name`, or both.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the LUN. Valid in POST. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<namespace>" where the qtree name is optional. |
| node | node | The LUN node. |
| smbc | smbc | "Properties related to SM-BC replication." |
| uuid | string | The unique identifier of the LUN. Valid in POST. |

_links

| Name | Type | Description |
|---|---|---|
| node | href | |
| self | href | |

reporting_nodes

A cluster node from which network paths to the LUN are advertised by ONTAP via the SAN protocols.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the node.<br><br>Either `uuid` or `name` are required in POST. |
| uuid | string | The unique identifier of the node.<br><br>Either `uuid` or `name` are required in POST. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_error

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

| Name | Type | Description |
|---|---|---|
| message | string | Error message |
| target | string | The target parameter that caused the error. |

# Add, remove, or discover LUN map reporting nodes

## Add, remove, or discover LUN map reporting nodes

### Overview

A LUN map's reporting nodes are the cluster nodes from which network paths to a mapped LUN are advertised via the SAN protocols as part of the Selective LUN Map (SLM) feature of ONTAP. SLM reduces the number of paths from the host to a mapped LUN and enables management of a single initiator group (igroup) per host.

If there are no reporting nodes in a LUN map, network paths to all cluster nodes having the appropriate network interfaces (LIFs) in the SVM are advertised. This is not a typical configuration and is reserved for limited specific use cases. Note that having no reporting nodes in a LUN map differs subtly from having all reporting nodes in the LUN map. If a LUN map has an empty reporting nodes list and a new node is added to the cluster, a path to the new node will also be advertised. If a LUN map has all cluster nodes in its reporting nodes list and a new node is added to the cluster, a path to the new node is not advertised unless the LUN map's reporting nodes are explicitly updated to include the new node.

If portsets are used to further restrict access for initiators to specific LIFs, the mapped LUN will be accessible only via the LIFs in the portset that are on the reporting nodes of the LUN map.

When a LUN map is created, the cluster node hosting the mapped LUN and its high availability (HA) partner are set as the initial reporting nodes.

Before moving a mapped LUN or a volume containing mapped LUNs to another HA pair within the same cluster, the destination node should be added to the LUN map's reporting nodes. This ensures that active, optimized LUN paths are maintained. After moving a mapped LUN or a volume containing mapped LUNs to another HA pair within the same cluster, the cluster node that previously hosted the mapped LUN should be removed from the LUN map's reporting node. Further details for this workflow may be found in the ONTAP SAN Administration documentation - see `Modifying the SLM reporting-nodes list`.

### Examples

#### Adding a node to a LUN map

This example adds a cluster node, and its high availability (HA) partner cluster node, to a LUN map's reporting nodes.

```
# The API:
POST /api/protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/san/lun-maps/c8ad35ce-ef03-
4577-88f0-9ad150ac6bbf/b10a8165-8346-11eb-ab8e-005056bbb402/reporting-
nodes' -H 'Accept: application/hal+json' -d '{ "name": "node2" }'
```

#### Removing a node from a LUN map

This example removes a cluster node, and its high availability (HA) partner cluster node, from a LUN map's reporting nodes.

```
# The API:
DELETE /api/protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-
nodes/{uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/lun-maps/c8ad35ce-
ef03-4577-88f0-9ad150ac6bbf/b10a8165-8346-11eb-ab8e-
005056bbb402/reporting-nodes/6d2cd7d5-493a-daf8-9ae1-219e4ad6f77d' -H
'Accept: application/hal+json'
```

## Retrieve LUN map reporting nodes

GET `/protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes`

**Introduced In:** 9.10

Retrieves LUN map reporting nodes.

### Related ONTAP commands

- `lun mapping show`

### Learn more

- DOC /protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | The unique identifier of the LUN. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| igroup.uuid | string | path | True | The unique identifier of the igroup. |
| name | string | query | False | Filter by name |
| uuid | string | query | False | Filter by uuid |
| 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.<br>• 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[lun_map_reporting_node] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "node": {
          "href": "/api/resourcelink"
        },
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "igroup": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "lun": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "name": "node1",
      "uuid": "5ac8eb9c-4e32-dbaa-57ca-fb905976f54e"
    }
  ]
}
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374875 | The specified LUN does not exist or is not accessible to the caller. |
| 5374878 | The specified initiator group does not exist, is not accessible to the caller, or is not in the same SVM as the specified LUN. |
| 5374922 | The specified LUN map does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 |
|---|---|---|
| node | href | |
| self | href | |

_links

| Name | Type | Description |
|---|---|---|
| self | href | |

igroup

The initiator group of the LUN map of the reporting node.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| uuid | string | The unique identifier of the initiator group. |

lun

The LUN of the LUN map of the reporting node.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| uuid | string | The unique identifier of the LUN. |

lun_map_reporting_node

A cluster node from which network paths to the LUN are advertised by ONTAP via the SAN protocols.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| igroup | igroup | The initiator group of the LUN map of the reporting node. |
| lun | lun | The LUN of the LUN map of the reporting node. |
| name | string | The name of the node.<br><br>Either `uuid` or `name` are required in POST. |
| uuid | string | The unique identifier of the node.<br><br>Either `uuid` or `name` are required in POST. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 a reporting node and its HA partner to a LUN map

POST `/protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes`

**Introduced In:** 9.10

Adds a reporting node and its HA partner to a LUN map.

### Required properties

- `uuid` or `name` - A cluster node to add.

### Related ONTAP commands

- `lun mapping add-reporting-nodes`

### Learn more

- DOC /protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | The unique identifier of the LUN. |
| igroup.uuid | string | path | True | The unique identifier of the igroup. |
| return_records | boolean | query | False | The default is false. If set to true, the records are returned.<br>• Default value: |

### Request Body

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| igroup | igroup | The initiator group of the LUN map of the reporting node. |
| lun | lun | The LUN of the LUN map of the reporting node. |
| name | string | The name of the node.<br><br>Either `uuid` or `name` are required in POST. |
| uuid | string | The unique identifier of the node.<br><br>Either `uuid` or `name` are required in POST. |

### Example request

```
{
  "igroup": {
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "lun": {
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "name": "node1",
  "uuid": "5ac8eb9c-4e32-dbaa-57ca-fb905976f54e"
}
```

### Response

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| num_records | integer | The number of records in the response. |
| records | array[lun_map_reporting_node] | |

### Example response

```
{
  "num_records": 1,
  "records": [
    {
      "igroup": {
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "lun": {
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "name": "node1",
      "uuid": "5ac8eb9c-4e32-dbaa-57ca-fb905976f54e"
    }
  ]
}
```

#### Headers

| Name | Description | Type |
|---|---|---|
| Location | Useful for tracking the resource location | string |

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374875 | The specified LUN does not exist or is not accessible to the caller. |
| 5374878 | The specified initiator group does not exist, is not accessible to the caller, or is not in the same SVM as the specified LUN. |
| 5374920 | The specified cluster node does not exist. |
| 5374921 | The specified cluster node name and UUID do not refer to the same cluster node. |
| 5374922 | The specified LUN map does not exist. |
| 5374923 | A cluster node `uuid` or `name` must be specified to add a reporting node. |

Also see the table of common errors in the Response body overview section of this documentation.

### Definitions

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

igroup

The initiator group of the LUN map of the reporting node.

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the initiator group. |

lun

The LUN of the LUN map of the reporting node.

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the LUN. |

lun_map_reporting_node

A cluster node from which network paths to the LUN are advertised by ONTAP via the SAN protocols.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| igroup | igroup | The initiator group of the LUN map of the reporting node. |
| lun | lun | The LUN of the LUN map of the reporting node. |
| name | string | The name of the node.<br><br>Either `uuid` or `name` are required in POST. |
| uuid | string | The unique identifier of the node.<br><br>Either `uuid` or `name` are required in POST. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

## Remove a reporting node and its HA partner from a LUN map

DELETE `/protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes/{uuid}`

**Introduced In:** 9.10

Removes a reporting node and its HA partner from a LUN map.

### Related ONTAP commands

- `lun mapping remove-reporting-nodes`

### Learn more

- DOC /protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | |
| igroup.uuid | string | path | True | |
| uuid | string | path | True | |

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374608 | An attempt was made to delete the last reporting node for a LUN. The REST API requires that a LUN map must maintain at least one reporting node. |
| 5374875 | The specified LUN does not exist or is not accessible to the caller. |
| 5374878 | The specified initiator group does not exist, is not accessible to the caller, or is not in the same SVM as the specified LUN. |
| 5374922 | The specified LUN map does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 |

returned_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 a LUN map reporting node

GET `/protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes/{uuid}`

**Introduced In:** 9.10

Retrieves a LUN map reporting node.

#### Related ONTAP commands

- `lun mapping show`

#### Learn more

- DOC /protocols/san/lun-maps/{lun.uuid}/{igroup.uuid}/reporting-nodes

#### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | The unique identifier of the LUN. |

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

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| igroup | igroup | The initiator group of the LUN map of the reporting node. |
| lun | lun | The LUN of the LUN map of the reporting node. |
| name | string | The name of the node.<br><br>Either `uuid` or `name` are required in POST. |
| uuid | string | The unique identifier of the node.<br><br>Either `uuid` or `name` are required in POST. |

### Example response


```
{
  "_links": {
    "node": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "igroup": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "lun": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "name": "node1",
  "uuid": "5ac8eb9c-4e32-dbaa-57ca-fb905976f54e"
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374875 | The specified LUN does not exist or is not accessible to the caller. |
| 5374878 | The specified initiator group does not exist, is not accessible to the caller, or is not in the same SVM as the specified LUN. |

| Error Code | Description |
| --- | --- |
| 5374922 | The specified LUN map does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
| --- | --- | --- |
| error | returned_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 |
|---|---|---|
| node | href | |
| self | href | |

_links

| Name | Type | Description |
|---|---|---|
| self | href | |

igroup

The initiator group of the LUN map of the reporting node.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| uuid | string | The unique identifier of the initiator group. |

lun

The LUN of the LUN map of the reporting node.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| uuid | string | The unique identifier of the LUN. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

# Manage SAN portsets

## Manage SAN portsets

### Overview

A portset is a collection of Fibre Channel Protocol and/or iSCSI network interfaces from the portset's SVM.

Portsets are used to limit the network interfaces through which an initiator can connect to mapped LUNs. When a portset is bound to an initiator group (igroup), and the initiator group is mapped to a LUN, the initiators of the initiator group can only reach the LUN through the network interfaces in the portset.

Portsets are not recommended for new configurations. With modern releases of ONTAP, it is recommended to use multiple SVMs and initiator groups with no bound portset to load balance applications over multiple ports on a node. Selective LUN mapping will automatically limit the number of visible paths to a LUN from the client host to those required for efficient access and high availability. The REST portset API is primarily intended for legacy use.

The portset REST API allows you to create, delete, and discover portsets, and to add and remove network interfaces from portsets.

A portset can be bound to one or more initiator groups. An initiator group (igroup) can be bound to at most one portset.

When a portset is created, the `protocol` property is used to restrict member network interfaces to Fibre Channel Protocol (*fcp*), iSCSI (*iscsi*), or both (*mixed*).

Zero or more network interfaces can be supplied when the portset is created. After creation, network interfaces can be added to or removed from the portset using the `/protocols/san/portsets/{portset.uuid}/interfaces` endpoint. See DELETE /protocols/san/portsets/{portset.uuid}/interfaces/{name} for more details.

### Examples

#### Creating a portset with no network interfaces

The example portset uses the default `mixed` protocol. Note that the `return_records` query parameter is used to obtain the newly created portset in the response.

```
# The API:
POST /api/protocols/san/portsets

# The call:
curl -X POST 'https://<mgmt-
ip>/api/protocols/san/portsets?return_records=true' -H 'Accept:
application/hal+json' -d '{ "svm" : { "name": "svm1" }, "name": "portset1"
}'

# The response:
{
"num_records": 1,
"records": [
  {
    "uuid": "5d7b1dfa-1ed7-11eb-8b0f-005056bb3521",
    "svm": {
      "uuid": "31484775-1e23-11eb-b2a8-005056bb3521",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/31484775-1e23-11eb-b2a8-005056bb3521"
        }
      }
    },
    "name": "portset1",
    "protocol": "mixed",
    "_links": {
      "self": {
        "href": "/api/protocols/san/portsets/5d7b1dfa-1ed7-11eb-8b0f-
005056bb3521"
      }
    }
  }
]
}
```

#### Creating an iSCSI portset with two network interfaces

Note that the `return_records` query parameter is used to obtain the newly created portset in the response.

```
# The API:
POST /api/protocols/san/portsets
```

```
# The call:
curl -X POST 'https://<mgmt-
ip>/api/protocols/san/portsets?return_records=true' -H 'Accept:
application/hal+json' -d '{ "svm" : { "name": "svm1" }, "name":
"portset2", "protocol": "iscsi", "interfaces": [ { "ip": { "name": "lif1"
} }, { "ip": { "name": "lif2" } } ] }'

# The response:
{
"num_records": 1,
"records": [
  {
    "uuid": "816c0d49-1ed7-11eb-8b0f-005056bb3521",
    "svm": {
      "uuid": "31484775-1e23-11eb-b2a8-005056bb3521",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/31484775-1e23-11eb-b2a8-005056bb3521"
        }
      }
    },
    "name": "portset2",
    "protocol": "iscsi",
    "interfaces": [
      {
        "uuid": "f37bfb01-1e2a-11eb-b2a8-005056bb3521",
        "ip": {
          "uuid": "f37bfb01-1e2a-11eb-b2a8-005056bb3521",
          "name": "lif1",
          "ip": {
            "address": "192.168.1.100"
          },
          "_links": {
            "self": {
              "href": "/api/network/ip/interfaces/f37bfb01-1e2a-11eb-b2a8-
005056bb3521"
            }
          }
        },
        "_links": {
          "self": {
            "href": "/api/protocols/san/portsets/816c0d49-1ed7-11eb-8b0f-
005056bb3521/interfaces/f37bfb01-1e2a-11eb-b2a8-005056bb3521"
          }
        }
```

```
      },
      {
        "uuid": "f92178e7-1e2a-11eb-b2a8-005056bb3521",
        "ip": {
          "uuid": "f92178e7-1e2a-11eb-b2a8-005056bb3521",
          "name": "lif2",
          "ip": {
            "address": "192.168.1.101"
          },
          "_links": {
            "self": {
              "href": "/api/network/ip/interfaces/f92178e7-1e2a-11eb-b2a8-
005056bb3521"
            }
          }
        },
        "_links": {
          "self": {
            "href": "/api/protocols/san/portsets/816c0d49-1ed7-11eb-8b0f-
005056bb3521/interfaces/f92178e7-1e2a-11eb-b2a8-005056bb3521"
          }
        }
      }
    ],
    "_links": {
      "self": {
        "href": "/api/protocols/san/portsets/816c0d49-1ed7-11eb-8b0f-
005056bb3521"
      }
    }
  }
]
}
```

#### Retrieving a summary of all portsets

```
# The API:
GET /api/protocols/san/portsets

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/portsets' -H 'Accept:
application/hal+json'

# The response:
```

```
{
"records": [
  {
    "uuid": "5d7b1dfa-1ed7-11eb-8b0f-005056bb3521",
    "name": "portset1",
    "_links": {
      "self": {
        "href": "/api/protocols/san/portsets/5d7b1dfa-1ed7-11eb-8b0f-
005056bb3521"
      }
    }
  },
  {
    "uuid": "816c0d49-1ed7-11eb-8b0f-005056bb3521",
    "name": "portset2",
    "_links": {
      "self": {
        "href": "/api/protocols/san/portsets/816c0d49-1ed7-11eb-8b0f-
005056bb3521"
      }
    }
  },
  {
    "uuid": "b716b4d2-1ed7-11eb-8b0f-005056bb3521",
    "name": "portset3",
    "_links": {
      "self": {
        "href": "/api/protocols/san/portsets/b716b4d2-1ed7-11eb-8b0f-
005056bb3521"
      }
    }
  }
],
"num_records": 3,
"_links": {
  "self": {
    "href": "/api/protocols/san/portsets"
  }
}
}
```

#### Retrieving details for a specific portset

```
# The API:
```

```
GET /api/protocols/san/portsets/{uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/portsets/b716b4d2-1ed7-
11eb-8b0f-005056bb3521' -H 'Accept: application/hal+json'

# The response:
{
"uuid": "b716b4d2-1ed7-11eb-8b0f-005056bb3521",
"svm": {
  "uuid": "31484775-1e23-11eb-b2a8-005056bb3521",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/31484775-1e23-11eb-b2a8-005056bb3521"
    }
  }
},
"name": "portset3",
"protocol": "fcp",
"interfaces": [
  {
    "uuid": "164eb052-1e2b-11eb-b2a8-005056bb3521",
    "fc": {
      "uuid": "164eb052-1e2b-11eb-b2a8-005056bb3521",
      "name": "lif5",
      "wwpn": "20:01:00:50:56:bb:35:21",
      "_links": {
        "self": {
          "href": "/api/network/fc/interfaces/164eb052-1e2b-11eb-b2a8-
005056bb3521"
        }
      }
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/portsets/b716b4d2-1ed7-11eb-8b0f-
005056bb3521/interfaces/164eb052-1e2b-11eb-b2a8-005056bb3521"
      }
    }
  },
  {
    "uuid": "197ba2b7-1e2b-11eb-b2a8-005056bb3521",
    "fc": {
      "uuid": "197ba2b7-1e2b-11eb-b2a8-005056bb3521",
      "name": "lif6",
```

```
      "wwpn": "20:02:00:50:56:bb:35:21",
      "_links": {
        "self": {
          "href": "/api/network/fc/interfaces/197ba2b7-1e2b-11eb-b2a8-
005056bb3521"
        }
      }
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/portsets/b716b4d2-1ed7-11eb-8b0f-
005056bb3521/interfaces/197ba2b7-1e2b-11eb-b2a8-005056bb3521"
      }
    }
  }
],
"_links": {
  "self": {
    "href": "/api/protocols/san/portsets/b716b4d2-1ed7-11eb-8b0f-
005056bb3521"
  }
}
}
```

#### Deleting a portset

```
# The API:
DELETE /api/protocols/san/portsets/{uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/portsets/b716b4d2-
1ed7-11eb-8b0f-005056bb3521' -H 'Accept: application/hal+json'
```

#### Adding a network interface to a portset

```
# The API:
POST /api/protocols/san/portsets/{uuid}/interfaces

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/san/portsets/5d7b1dfa-1ed7-
11eb-8b0f-005056bb3521/interfaces' -H 'Accept: application/hal+json' -d '{
"fc": { "name": "lif4" } }'
}
```

#### Adding multiple network interfaces to a portset in a single call

```
# The API:
POST /api/protocols/san/portsets/{uuid}/interfaces

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/san/portsets/5d7b1dfa-1ed7-
11eb-8b0f-005056bb3521/interfaces' -H 'Accept: application/hal+json' -d '{
"records": [ { "ip": { "name": "lif1" } },  { "ip": { "name": "lif2" } },
{ "fc": { "name": "lif5" } } ] }'
```

#### Removing a network interface from a portset

```
# The API:
DELETE /api/protocols/san/portsets/{uuid}/interfaces/f92178e7-1e2a-11eb-
b2a8-005056bb3521

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/portsets/5d7b1dfa-
1ed7-11eb-8b0f-005056bb3521/interfaces/f92178e7-1e2a-11eb-b2a8-
005056bb3521' -H 'Accept: application/hal+json'
```

## Retrieve portsets

GET `/protocols/san/portsets`

**Introduced In:** 9.9

Retrieves portsets.

### Related ONTAP commands

- `lun portset show`

### Learn more

- DOC /protocols/san/portsets

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.name | string | query | False | Filter by svm.name |
| svm.uuid | string | query | False | Filter by svm.uuid |
| protocol | string | query | False | Filter by protocol |
| name | string | query | False | Filter by name<br>• maxLength: 96<br>• minLength: 1 |
| interfaces.uuid | string | query | False | Filter by interfaces.uuid |
| interfaces.fc.uuid | string | query | False | Filter by interfaces.fc.uuid |
| interfaces.fc.wwpn | string | query | False | Filter by interfaces.fc.wwpn |
| interfaces.fc.name | string | query | False | Filter by interfaces.fc.name |
| interfaces.ip.uuid | string | query | False | Filter by interfaces.ip.uuid |
| interfaces.ip.name | string | query | False | Filter by interfaces.ip.name |
| interfaces.ip.ip.address | string | query | False | Filter by interfaces.ip.ip.address |
| uuid | string | query | False | Filter by uuid |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| igroups.uuid | string | query | False | Filter by igroups.uuid |
| igroups.name | string | query | False | Filter by igroups.name<br>• maxLength: 96<br>• minLength: 1 |
| 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.<br>• 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[portset] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "igroups": [
        {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "igroup1",
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        }
      ],
      "interfaces": [
        {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "fc": {
            "_links": {
              "self": {
                "href": "/api/resourcelink"
              }
            },
            "name": "fc_lif1",
            "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
            "wwpn": "20:00:00:50:56:b4:13:a8"
```

```
          },
          "ip": {
            "_links": {
              "self": {
                "href": "/api/resourcelink"
              }
            },
            "ip": {
              "address": "10.10.10.7"
            },
            "name": "lif1",
            "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
          },
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        }
      ],
      "name": "portset1",
      "protocol": "string",
      "svm": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

igroups

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

fc

An FC interface.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the FC interface. |
| uuid | string | The unique identifier of the FC interface. |
| wwpn | string | The WWPN of the FC interface. |

ip

IP information

| Name | Type | Description |
|---|---|---|
| address | string | IPv4 or IPv6 address |

ip

A network interface. Either UUID or name may be supplied on input.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| ip | ip | IP information |
| name | string | The name of the interface. If only the name is provided, the SVM scope must be provided by the object this object is embedded in. |
| uuid | string | The UUID that uniquely identifies the interface. |

interfaces

A container for either a Fibre Channel network interface or an IP network interface. On POST `fc` and `ip` are mutually exclusive.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |
| uuid | string | The unique identifier of the network interface. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

portset

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| igroups | array[igroups] | An array initiator groups to which the portset is bound. |
| interfaces | array[interfaces] | An array of network interfaces that are members of the portset. These are the only network interfaces through which the initiators of a bound igroup can access mapped LUNs.<br><br>Zero or more network interfaces can be supplied when the portset is created. After creation, network interfaces can be added or removed from the portset using the `/protocols/san/portsets/{portset.uuid}/interfaces` endpoint. See DELETE /protocols/san/portsets/{portset.uuid}/interfaces/{uuid} for more details. |
| name | string | The name of the portset. Required in POST.<br><br>The name of a portset cannot be changed after creation. |
| protocol | string | The protocols supported by the portset. This restricts the type of network interfaces that can be added to the portset. Optional in POST; if not supplied, this defaults to *mixed*.<br><br>The protocol of a portset cannot be changed after creation. |

| Name | Type | Description |
|---|---|---|
| svm | svm | SVM, applies only to SVM-scoped objects. |
| uuid | string | The unique identifier for a portset. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 a portset

POST `/protocols/san/portsets`

**Introduced In:** 9.9

Creates a portset.

### Required properties

- `svm.uuid` or `svm.name` - Existing SVM in which to create the portset.
- `name` - Name of the portset.

### Recommended optional properties

- `protocol` - The network protocol of the interfaces in the portset.
- `interfaces` - Network interfaces to include in the portset. This property can be used to create the portset and populate it with network interfaces in a single request.

### Default property values

If not specified in POST, the following default property values are assigned.

- `protocol` - *mixed* - Data protocol of the portset's network interfaces.

### Related ONTAP commands

- `lun portset create`

### Learn more

- DOC /protocols/san/portsets

### Parameters

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

### Request Body

| Name | Type | Description |
|---|---|---|
| igroups | array[igroups] | An array initiator groups to which the portset is bound. |
| interfaces | array[interfaces] | An array of network interfaces that are members of the portset. These are the only network interfaces through which the initiators of a bound igroup can access mapped LUNs.<br><br>Zero or more network interfaces can be supplied when the portset is created. After creation, network interfaces can be added or removed from the portset using the `/protocols/san/portsets/{portset.uuid}/interfaces` endpoint. See DELETE /protocols/san/portsets/{portset.uuid}/interfaces/{uuid} for more details. |

| Name | Type | Description |
| --- | --- | --- |
| name | string | The name of the portset. Required in POST.<br><br>The name of a portset cannot be changed after creation. |
| protocol | string | The protocols supported by the portset. This restricts the type of network interfaces that can be added to the portset. Optional in POST; if not supplied, this defaults to *mixed*.<br><br>The protocol of a portset cannot be changed after creation. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| uuid | string | The unique identifier for a portset. |

### Example request


```
{
  "igroups": [
    {
      "name": "igroup1",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "interfaces": [
    {
      "fc": {
        "name": "fc_lif1",
        "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
        "wwpn": "20:00:00:50:56:b4:13:a8"
      },
      "ip": {
        "name": "lif1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "name": "portset1",
  "protocol": "string",
  "svm": {
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
}
```


### Response

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| num_records | integer | The number of records in the response. |
| records | array[portset] | |

### Example response


```
{
  "num_records": 1,
  "records": [
    {
      "igroups": [
        {
          "name": "igroup1",
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        }
      ],
      "interfaces": [
        {
          "fc": {
            "name": "fc_lif1",
            "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
            "wwpn": "20:00:00:50:56:b4:13:a8"
          },
          "ip": {
            "name": "lif1",
            "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
          },
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        }
      ],
      "name": "portset1",
      "protocol": "string",
      "svm": {
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "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 |
|---|---|
| 1254259 | A portset with the same name already exists in the SVM. |
| 2621462 | The specified SVM does not exist. |
| 2621706 | The specified `svm.uuid` and `svm.name` do not refer to the same SVM. |
| 2621707 | No SVM was specified. Either `svm.name` or `svm.uuid` must be supplied. |
| 5373958 | The specified portset name contains invalid characters. |
| 5373982 | An invalid WWN was specified. The length is incorrect. |
| 5373983 | An invalid WWN was specified. The format is incorrect. |
| 5374905 | An invalid interfaces array element was specified. |
| 5374906 | A specified network interface was not found. |
| 5374907 | The specified network interface UUID and name don’t identify the same network interface. |
| 5374914 | An attempt was made to add a network interface of an incompatible protocol to a portset. |

Also see the table of common errors in the Response body overview section of this documentation.

### Definitions

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

igroups

| Name | Type | Description |
|---|---|---|
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

fc

An FC interface.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the FC interface. |
| uuid | string | The unique identifier of the FC interface. |
| wwpn | string | The WWPN of the FC interface. |

ip

IP information

ip

A network interface. Either UUID or name may be supplied on input.

| Name | Type | Description |
|---|---|---|
| ip | ip | IP information |
| name | string | The name of the interface. If only the name is provided, the SVM scope must be provided by the object this object is embedded in. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The UUID that uniquely identifies the interface. |

interfaces

A container for either a Fibre Channel network interface or an IP network interface. On POST `fc` and `ip` are mutually exclusive.

| Name | Type | Description |
|---|---|---|
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |
| uuid | string | The unique identifier of the network interface. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

portset

| Name | Type | Description |
|---|---|---|
| igroups | array[igroups] | An array initiator groups to which the portset is bound. |

| Name | Type | Description |
|---|---|---|
| interfaces | array[interfaces] | An array of network interfaces that are members of the portset. These are the only network interfaces through which the initiators of a bound igroup can access mapped LUNs.<br><br>Zero or more network interfaces can be supplied when the portset is created. After creation, network interfaces can be added or removed from the portset using the `/protocols/san/portsets/{portset.uuid}/interfaces` endpoint. See DELETE /protocols/san/portsets/{portset.uuid}/interfaces/{uuid} for more details. |
| name | string | The name of the portset. Required in POST.<br><br>The name of a portset cannot be changed after creation. |
| protocol | string | The protocols supported by the portset. This restricts the type of network interfaces that can be added to the portset. Optional in POST; if not supplied, this defaults to *mixed*.<br><br>The protocol of a portset cannot be changed after creation. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| uuid | string | The unique identifier for a portset. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 portset network interfaces

GET `/protocols/san/portsets/{portset.uuid}/interfaces`

**Introduced In:** 9.9

Retrieves interfaces of a portset.

### Related ONTAP commands

- `lun portset show`

### Learn more

- DOC /protocols/san/portsets

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| portset.uuid | string | path | True | The unique identifier of the portset. |
| 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.<br>• Default value: 1 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[portset_interface] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "fc": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "fc_lif1",
        "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
        "wwpn": "20:00:00:50:56:b4:13:a8"
      },
      "ip": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "ip": {
          "address": "10.10.10.7"
        },
        "name": "lif1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "portset": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
```

```
        },
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ]
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374908 | The portset specified in the URI does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

fc

An FC interface.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the FC interface. |
| uuid | string | The unique identifier of the FC interface. |
| wwpn | string | The WWPN of the FC interface. |

ip

IP information

| Name | Type | Description |
|---|---|---|
| address | string | IPv4 or IPv6 address |

ip

A network interface. Either UUID or name may be supplied on input.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
|---|---|---|
| ip | ip | IP information |
| name | string | The name of the interface. If only the name is provided, the SVM scope must be provided by the object this object is embedded in. |
| uuid | string | The UUID that uniquely identifies the interface. |

portset

The portset in which the network interface is found.

Note that this does not mean that the network interface cannot also be found in other portsets.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| uuid | string | The unique identifier of the portset. |

records

A container for either a Fibre Channel network interface or an IP network interface. On POST `fc` and `ip` are mutually exclusive.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |
| uuid | string | The unique identifier of the network interface. |

portset_interface

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| fc | fc | An FC interface. |

| Name | Type | Description |
|---|---|---|
| ip | ip | A network interface. Either UUID or name may be supplied on input. |
| portset | portset | The portset in which the network interface is found.<br><br>Note that this does not mean that the network interface cannot also be found in other portsets. |
| uuid | string | The unique identifier of the network interface. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 portset network interfaces

POST `/protocols/san/portsets/{portset.uuid}/interfaces`

**Introduced In:** 9.9

Adds one or more interfaces to a portset.

### Required properties

- `fc`, `ip` or `records` - Network interface(s) to add to the portset.

### Related ONTAP commands

- `lun portset add`

### Learn more

- DOC /protocols/san/portsets

### Parameters

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

### Request Body

| Name | Type | Description |
|---|---|---|
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |
| portset | portset | The portset in which the network interface is found.<br><br>Note that this does not mean that the network interface cannot also be found in other portsets. |
| records | array[records] | An array of network interfaces specified to add multiple interfaces to a portset in a single API call. Valid in POST only and not allowed when the `fc` or `ip` property is used. |
| uuid | string | The unique identifier of the network interface. |

### Example request

```
{
  "fc": {
    "name": "fc_lif1",
    "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
    "wwpn": "20:00:00:50:56:b4:13:a8"
  },
  "ip": {
    "name": "lif1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "portset": {
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "records": [
    {
      "fc": {
        "name": "fc_lif1",
        "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
        "wwpn": "20:00:00:50:56:b4:13:a8"
      },
      "ip": {
        "name": "lif1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
}
```

### Response

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| num_records | integer | The number of records in the response. |
| records | array[portset_interface] | |

### Example response


```
{
  "num_records": 1,
  "records": [
    {
      "fc": {
        "name": "fc_lif1",
        "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
        "wwpn": "20:00:00:50:56:b4:13:a8"
      },
      "ip": {
        "name": "lif1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "portset": {
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "records": [
        {
          "fc": {
            "name": "fc_lif1",
            "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
            "wwpn": "20:00:00:50:56:b4:13:a8"
          },
          "ip": {
            "name": "lif1",
            "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
          },
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        }
      ],
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ]
}
```


### Headers

| Name | Description | Type |
|---|---|---|
| Location | Useful for tracking the resource location | string |

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
| --- | --- |
| 5374906 | A specified network interface was not found. |
| 5374907 | The specified network interface UUID and name don’t identify the same network interface. |
| 5374909 | An invalid combination of network interface properties was specified. |
| 5374910 | An incomplete set of network interface properties was specified. |
| 5374914 | An attempt was made to add a network interface of an incompatible protocol to a portset. |
| 5374915 | An attempt was made to add a duplicate network interface to a portset. |

Also see the table of common errors in the Response body overview section of this documentation.

### Definitions

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

fc

An FC interface.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the FC interface. |
| uuid | string | The unique identifier of the FC interface. |
| wwpn | string | The WWPN of the FC interface. |

ip

IP information

ip

A network interface. Either UUID or name may be supplied on input.

| Name | Type | Description |
|---|---|---|
| ip | ip | IP information |
| name | string | The name of the interface. If only the name is provided, the SVM scope must be provided by the object this object is embedded in. |
| uuid | string | The UUID that uniquely identifies the interface. |

portset

The portset in which the network interface is found.

Note that this does not mean that the network interface cannot also be found in other portsets.

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the portset. |

records

A container for either a Fibre Channel network interface or an IP network interface. On POST `fc` and `ip` are mutually exclusive.

| Name | Type | Description |
|---|---|---|
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |
| uuid | string | The unique identifier of the network interface. |

portset_interface

| Name | Type | Description |
|---|---|---|
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |
| portset | portset | The portset in which the network interface is found.<br><br>Note that this does not mean that the network interface cannot also be found in other portsets. |
| records | array[records] | An array of network interfaces specified to add multiple interfaces to a portset in a single API call. Valid in POST only and not allowed when the `fc` or `ip` property is used. |
| uuid | string | The unique identifier of the network interface. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 portset network interfaces

DELETE `/protocols/san/portsets/{portset.uuid}/interfaces/{uuid}`

**Introduced In:** 9.9

Deletes a network interface from a portset.

### Related ONTAP commands

- `lun portset remove`

### Learn more

- DOC /protocols/san/portsets

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| portset.uuid | string | path | True | The unique identifier of the portset. |
| uuid | string | path | True | The unique identifier of the network interface in the portset. |

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374906 | A specified network interface was not found. |
| 5374908 | The portset specified in the URI does not exist. |
| 5374916 | The specified network interface is not in the portset. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 |

returned_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 specific portset network interfaces

GET `/protocols/san/portsets/{portset.uuid}/interfaces/{uuid}`

**Introduced In:** 9.9

Retrieves a network interface of a portset.

### Related ONTAP commands

- `lun portset show`

### Learn more

- DOC /protocols/san/portsets

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| portset.uuid | string | path | True | The unique identifier of the portset. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| uuid | string | path | True | The unique identifier of the network interface in the portset. |
| fields | array[string] | query | False | Specify the fields to return. |

**Response**

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |
| portset | portset | The portset in which the network interface is found.<br><br>Note that this does not mean that the network interface cannot also be found in other portsets. |
| uuid | string | The unique identifier of the network interface. |

### Example response


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "fc": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "fc_lif1",
    "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
    "wwpn": "20:00:00:50:56:b4:13:a8"
  },
  "ip": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "ip": {
      "address": "10.10.10.7"
    },
    "name": "lif1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "portset": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 4 | The interface is not a member of the portset. |
| 5374908 | The portset specified in the URI does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

fc

An FC interface.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the FC interface. |
| uuid | string | The unique identifier of the FC interface. |
| wwpn | string | The WWPN of the FC interface. |

ip

IP information

| Name | Type | Description |
|---|---|---|
| address | string | IPv4 or IPv6 address |

ip

A network interface. Either UUID or name may be supplied on input.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| ip | ip | IP information |
| name | string | The name of the interface. If only the name is provided, the SVM scope must be provided by the object this object is embedded in. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The UUID that uniquely identifies the interface. |

portset

The portset in which the network interface is found.

Note that this does not mean that the network interface cannot also be found in other portsets.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| uuid | string | The unique identifier of the portset. |

records

A container for either a Fibre Channel network interface or an IP network interface. On POST `fc` and `ip` are mutually exclusive.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |
| uuid | string | The unique identifier of the network interface. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_error

| Name | Type | Description |
|---|---|---|
| arguments | array[error_arguments] | Message arguments |

| Name | Type | Description |
|---|---|---|
| code | string | Error code |
| message | string | Error message |
| target | string | The target parameter that caused the error. |

## Delete a portset

DELETE `/protocols/san/portsets/{uuid}`

**Introduced In:** 9.9

Deletes a portset.

### Related ONTAP commands

- `lun portset delete`

### Learn more

- DOC /protocols/san/portsets

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| uuid | string | path | True | The unique identifier of the portset. |
| allow_delete_while_bound | boolean | query | False | Allows deletion of a portset that is bound to an igroup.<br><br>Deleting a portset can expand the set of LIFs through which a LUN is available.<br><br>• Default value: |

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374063 | An attempt was made to delete a portset that is bound to an initiator group without using query parameter `allow_delete_while_bound`. |
| 5374908 | The portset does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 |

returned_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 a portset

GET `/protocols/san/portsets/{uuid}`

**Introduced In:** 9.9

Retrieves a portset.

**Related ONTAP commands**

- `lun portset show`

**Learn more**

- DOC /protocols/san/portsets

**Parameters**

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

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| igroups | array[igroups] | An array initiator groups to which the portset is bound. |
| interfaces | array[interfaces] | An array of network interfaces that are members of the portset. These are the only network interfaces through which the initiators of a bound igroup can access mapped LUNs.<br><br>Zero or more network interfaces can be supplied when the portset is created. After creation, network interfaces can be added or removed from the portset using the `/protocols/san/portsets/{portset.uuid}/interfaces` endpoint. See DELETE /protocols/san/portsets/{portset.uuid}/interfaces/{uuid} for more details. |
| name | string | The name of the portset. Required in POST.<br><br>The name of a portset cannot be changed after creation. |
| protocol | string | The protocols supported by the portset. This restricts the type of network interfaces that can be added to the portset. Optional in POST; if not supplied, this defaults to *mixed*.<br><br>The protocol of a portset cannot be changed after creation. |
| svm | svm | SVM, applies only to SVM-scoped objects. |
| uuid | string | The unique identifier for a portset. |

**Example response**


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "igroups": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "igroup1",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "interfaces": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "fc": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "fc_lif1",
        "uuid": "3a09ab42-4da1-32cf-9d35-3385a6101a0b",
        "wwpn": "20:00:00:50:56:b4:13:a8"
      },
      "ip": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "ip": {
          "address": "10.10.10.7"
        },
        "name": "lif1",
```

```
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    }
  ],
  "name": "portset1",
  "protocol": "string",
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 4 | The portset does not exist. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

igroups

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

fc

An FC interface.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the FC interface. |
| uuid | string | The unique identifier of the FC interface. |
| wwpn | string | The WWPN of the FC interface. |

ip

IP information

| Name | Type | Description |
|---|---|---|
| address | string | IPv4 or IPv6 address |

ip

A network interface. Either UUID or name may be supplied on input.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| ip | ip | IP information |
| name | string | The name of the interface. If only the name is provided, the SVM scope must be provided by the object this object is embedded in. |
| uuid | string | The UUID that uniquely identifies the interface. |

interfaces

A container for either a Fibre Channel network interface or an IP network interface. On POST `fc` and `ip` are mutually exclusive.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| fc | fc | An FC interface. |
| ip | ip | A network interface. Either UUID or name may be supplied on input. |
| uuid | string | The unique identifier of the network interface. |

svm

SVM, applies only to SVM-scoped objects.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

# Manage SAN vVol bindings

## Manage SAN vVol bindings

### Overview

A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.

Class `protocol_endpoint` and `vvol` LUNs support many-to-many vVol bindings. A LUN of one class can be bound to zero or more LUNs of the opposite class.

The vVol binding between any two specific LUNs is reference counted. When a REST POST is executed for a vVol binding that already exists, the vVol binding reference count is incremented. When a REST DELETE is executed, the vVol binding reference count is decremented. Only when the vVol binding count reaches zero, or the query parameter `delete_all_references` is supplied, is the vVol binding destroyed.

The vVol binding REST API allows you to create, delete, and discover vVol bindings.

### Examples

#### Creating a vVol binding

```
# The API:
POST /api/protocols/san/vvol-bindings

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/san/vvol-bindings' -H
'Accept: application/hal+json' -d '{ "svm": { "name": "svm1" },
"protocol_endpoint": { "name": "/vol/vol1/pe1" }, "vvol" : { "name":
"/vol/vol1/vvol1" } }'
```

#### Retrieving all vVol bindings

```
# The API:
GET /api/protocols/san/vvol-bindings

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/vvol-bindings' -H
'Accept: application/hal+json'

# The response:
{
"records": [
  {
    "protocol_endpoint": {
      "uuid": "2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4",
      "name": "/vol/vol1/pe1",
      "_links": {
        "self": {
          "href": "/api/storage/luns/2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4"
        }
      }
    },
    "vvol": {
      "uuid": "28c02623-42fa-4f5f-a984-a02044bfc005",
      "name": "/vol/vol1/vvol1",
      "_links": {
        "self": {
          "href": "/api/storage/luns/28c02623-42fa-4f5f-a984-a02044bfc005"
        }
      }
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/vvol-bindings/2aab57f3-dc5d-491e-80d2-
```

```
15c7ed5dd5c4/28c02623-42fa-4f5f-a984-a02044bfc005"
      }
    }
  },
  {
    "protocol_endpoint": {
      "uuid": "2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4",
      "name": "/vol/vol1/pe1",
      "_links": {
        "self": {
          "href": "/api/storage/luns/2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4"
        }
      }
    },
    "vvol": {
      "uuid": "a8d4ba93-918f-40ad-a1e4-4d7b244bdcdf",
      "name": "/vol/vol1/vvol2",
      "_links": {
        "self": {
          "href": "/api/storage/luns/a8d4ba93-918f-40ad-a1e4-4d7b244bdcdf"
        }
      }
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/vvol-bindings/2aab57f3-dc5d-491e-80d2-
15c7ed5dd5c4/a8d4ba93-918f-40ad-a1e4-4d7b244bdcdf"
      }
    }
  }
],
"num_records": 2,
"_links": {
  "self": {
    "href": "/api/protocols/san/vvol-bindings"
  }
}
}
```

#### Retrieving a specific vVol binding

```
# The API:
GET /api/protocols/san/vvol-bindings/{protocol_endpoint.uuid}/{vvol.uuid}
```

```
# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/vvol-bindings/2aab57f3-
dc5d-491e-80d2-15c7ed5dd5c4/28c02623-42fa-4f5f-a984-a02044bfc005' -H
'Accept: application/hal+json'

# The response:
{
"protocol_endpoint": {
  "uuid": "2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4",
  "name": "/vol/vol1/pe1",
  "_links": {
    "self": {
      "href": "/api/storage/luns/2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4"
    }
  }
},
"vvol": {
  "uuid": "28c02623-42fa-4f5f-a984-a02044bfc005",
  "name": "/vol/vol1/vvol1",
  "_links": {
    "self": {
      "href": "/api/storage/luns/28c02623-42fa-4f5f-a984-a02044bfc005"
    }
  }
},
"svm": {
  "uuid": "bf295ccc-a6bb-11eb-93e8-005056bb470f",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/bf295ccc-a6bb-11eb-93e8-005056bb470f"
    }
  }
},
"id": 2411392,
"is_optimal": true,
"count": 1,
"_links": {
  "self": {
    "href": "/api/protocols/san/vvol-bindings/2aab57f3-dc5d-491e-80d2-
15c7ed5dd5c4/28c02623-42fa-4f5f-a984-a02044bfc005"
  }
}
}
```

#### Deleting a vVol binding

```
# The API:
DELETE /api/protocols/san/vvol-
bindings/{protocol_endpoint.uuid}/{vvol.uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/vvol-
bindings/2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4/28c02623-42fa-4f5f-a984-
a02044bfc005' -H 'Accept: application/hal+json'
```

## Retrieve vVol bindings

GET `/protocols/san/vvol-bindings`

**Introduced In:** 9.10

Retrieves vVol bindings.

#### Related ONTAP commands

- `lun bind show`
- DOC /protocols/san/vvol-bindings

#### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.name | string | query | False | Filter by svm.name |
| svm.uuid | string | query | False | Filter by svm.uuid |
| id | integer | query | False | Filter by id |
| secondary_id | string | query | False | Filter by secondary_id<br>• Introduced in: 9.13 |
| count | integer | query | False | Filter by count |
| is_optimal | boolean | query | False | Filter by is_optimal |
| protocol_endpoint.uuid | string | query | False | Filter by protocol_endpoint.uuid |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| protocol_endpoint.name | string | query | False | Filter by protocol_endpoint.name |
| vvol.uuid | string | query | False | Filter by vvol.uuid |
| vvol.name | string | query | False | Filter by vvol.name |
| 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.<br>• 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[vvol_binding] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "count": 1,
      "id": 1,
      "is_optimal": 1,
      "protocol_endpoint": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "/vol/volume1/lun1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "secondary_id": "0000D20000010000h",
      "svm": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "vvol": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
```

```
        },
        "name": "/vol/volume1/lun1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      }
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

protocol_endpoint

The class `protocol_endpoint` LUN in the vVol binding. Required in POST.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of a LUN. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<namespace>" where the qtree name is optional. |
| uuid | string | The unique identifier of the LUN. |

svm

The SVM in which the vVol binding and its LUNs are located. Required in POST.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

vvol

The class `vvol` LUN in the vVol binding. Required in POST.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of a LUN. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<namespace>" where the qtree name is optional. |
| uuid | string | The unique identifier of the LUN. |

vvol_binding

A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.

Class `protocol_endpoint` and `vvol` LUNs support many-to-many vVol bindings. A LUN of one class can be bound to zero or more LUNs of the opposite class.

The vVol binding between any two specific LUNs is reference counted. When a REST POST is executed for a vVol binding that already exists, the vVol binding reference count is incremented. When a REST DELETE is executed, the vVol binding reference count is decremented. Only when the vVol binding count reaches zero, or the query parameter `delete_all_references` is supplied, is the vVol binding destroyed.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
| --- | --- | --- |
| count | integer | The vVol binding between any two specific LUNs is reference counted. When a REST POST is executed for a vVol binding that already exists, the vVol binding reference count is incremented. When a REST DELETE is executed, the vVol binding reference count is decremented. Only when the vVol binding count reaches zero, or the query parameter `delete_all_references` is supplied, is the vVol binding destroyed. |
| id | integer | The ONTAP internal identifier assigned to the vVol binding. The bind identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>This property was included in early releases of the REST API for vVols and is maintained for backward compatibility. See the `secondary_id` property, which replaces `id`.<br><br>• example: 1<br>• readOnly: 1<br>• x-ntap-deprecated: 9.13.1<br>• Introduced in: 9.10<br>• x-nullable: true |
| is_optimal | boolean | Indicates if the class `protocol_endpoint` LUN and the class `vvol` LUN are on the same cluster node. |
| protocol_endpoint | protocol_endpoint | The class `protocol_endpoint` LUN in the vVol binding. Required in POST. |

| Name | Type | Description |
|---|---|---|
| secondary_id | string | The identifier assigned to the vVol binding, known as the secondary LUN ID. The identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>The format for a secondary LUN ID is 16 hexadecimal digits (zero-filled) followed by a lower case "h". |
| svm | svm | The SVM in which the vVol binding and its LUNs are located. Required in POST. |
| vvol | vvol | The class `vvol` LUN in the vVol binding. Required in POST. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 a vVol binding

POST `/protocols/san/vvol-bindings`

**Introduced In:** 9.10

Creates a vVol binding. The binding between any two specific LUNs is reference counted. When a binding is created that already exists, the binding count is incremented.

### Required properties

- `svm.uuid` or `svm.name` - Existing SVM in which to create the vVol binding.
- `protocol_endpoint.uuid` or `protocol_endpoint.name` - Existing class `protocol_endpoint` LUN to bind to the specified class `vvol` LUN.
- `vvol.uuid` or `vvol.name` - Existing class `vvol` LUN to bind to the specified class `protocol_endpoint` LUN.

### Related ONTAP commands

- `lun bind create`

### Learn more

- DOC /protocols/san/vvol-bindings

### Parameters

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

### Request Body

| Name | Type | Description |
|---|---|---|
| count | integer | The vVol binding between any two specific LUNs is reference counted. When a REST POST is executed for a vVol binding that already exists, the vVol binding reference count is incremented. When a REST DELETE is executed, the vVol binding reference count is decremented. Only when the vVol binding count reaches zero, or the query parameter `delete_all_references` is supplied, is the vVol binding destroyed. |

| Name | Type | Description |
| --- | --- | --- |
| id | integer | The ONTAP internal identifier assigned to the vVol binding. The bind identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>This property was included in early releases of the REST API for vVols and is maintained for backward compatibility. See the `secondary_id` property, which replaces `id`.<br><br>• example: 1<br>• readOnly: 1<br>• x-ntap-deprecated: 9.13.1<br>• Introduced in: 9.10<br>• x-nullable: true |
| is_optimal | boolean | Indicates if the class `protocol_endpoint` LUN and the class `vvol` LUN are on the same cluster node. |
| protocol_endpoint | protocol_endpoint | The class `protocol_endpoint` LUN in the vVol binding. Required in POST. |
| secondary_id | string | The identifier assigned to the vVol binding, known as the secondary LUN ID. The identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>The format for a secondary LUN ID is 16 hexadecimal digits (zero-filled) followed by a lower case "h". |
| svm | svm | The SVM in which the vVol binding and its LUNs are located. Required in POST. |
| vvol | vvol | The class `vvol` LUN in the vVol binding. Required in POST. |

### Example request


```
{
  "count": 1,
  "id": 1,
  "is_optimal": 1,
  "protocol_endpoint": {
    "name": "/vol/volume1/lun1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "secondary_id": "0000D20000010000h",
  "svm": {
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "vvol": {
    "name": "/vol/volume1/lun1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  }
}
```


### Response

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| num_records | integer | The number of records in the response. |
| records | array[vvol_binding] | |

### Example response


```
{
  "num_records": 1,
  "records": [
    {
      "count": 1,
      "id": 1,
      "is_optimal": 1,
      "protocol_endpoint": {
        "name": "/vol/volume1/lun1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "secondary_id": "0000D20000010000h",
      "svm": {
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "vvol": {
        "name": "/vol/volume1/lun1",
        "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 |
|---|---|
| 2621462 | The specified SVM does not exist or is not accessible to the caller. |
| 2621706 | Both the SVM UUID and SVM name were supplied, but don't refer to the same SVM. |

| Error Code | Description |
| --- | --- |
| 2621707 | No SVM was specified. Either `svm.name` or `svm.uuid` must be supplied. |
| 5374238 | A LUN in a snapshot was specified. |
| 5374323 | The LUN specified as the protocol endpoint LUN is not of class `protocol_endpoint`. |
| 5374325 | The LUN specified as the vVol LUN is not of class `vvol`. |
| 5374874 | The UUID and name supplied for the protocol endpoint of Vvol LUN do not refer to the same LUN. Use to the `target` property of the error object to differentiate between the protocol endpoint LUN and the vVol LUN. |
| 5374875 | The protocol endpoint or vVol LUN was not found or is not accessible to the caller. Use to the `target` property of the error object to differentiate between the protocol endpoint LUN and the vVol LUN. |
| 5374876 | The protocol endpoint or vVol LUN was not found in the SVM. Use to the `target` property of the error object to differentiate between the protocol endpoint LUN and the vVol LUN. |
| 5374924 | No protocol endpoint LUN was supplied. |
| 5374925 | No vVol LUN was supplied. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

protocol_endpoint

The class `protocol_endpoint` LUN in the vVol binding. Required in POST.

| Name | Type | Description |
|---|---|---|
| name | string | The name of a LUN. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<namespace>" where the qtree name is optional. |
| uuid | string | The unique identifier of the LUN. |

svm

The SVM in which the vVol binding and its LUNs are located. Required in POST.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

vvol

The class `vvol` LUN in the vVol binding. Required in POST.

| Name | Type | Description |
|---|---|---|
| name | string | The name of a LUN. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<namespace>" where the qtree name is optional. |
| uuid | string | The unique identifier of the LUN. |

vvol_binding

A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.

Class `protocol_endpoint` and `vvol` LUNs support many-to-many vVol bindings. A LUN of one class can be bound to zero or more LUNs of the opposite class.

The vVol binding between any two specific LUNs is reference counted. When a REST POST is executed for a vVol binding that already exists, the vVol binding reference count is incremented. When a REST DELETE is executed, the vVol binding reference count is decremented. Only when the vVol binding count reaches zero, or the query parameter `delete_all_references` is supplied, is the vVol binding destroyed.

| Name | Type | Description |
|---|---|---|
| count | integer | The vVol binding between any two specific LUNs is reference counted. When a REST POST is executed for a vVol binding that already exists, the vVol binding reference count is incremented. When a REST DELETE is executed, the vVol binding reference count is decremented. Only when the vVol binding count reaches zero, or the query parameter `delete_all_references` is supplied, is the vVol binding destroyed. |

| Name | Type | Description |
| --- | --- | --- |
| id | integer | The ONTAP internal identifier assigned to the vVol binding. The bind identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>This property was included in early releases of the REST API for vVols and is maintained for backward compatibility. See the `secondary_id` property, which replaces `id`.<br><br>• example: 1<br>• readOnly: 1<br>• x-ntap-deprecated: 9.13.1<br>• Introduced in: 9.10<br>• x-nullable: true |
| is_optimal | boolean | Indicates if the class `protocol_endpoint` LUN and the class `vvol` LUN are on the same cluster node. |
| protocol_endpoint | protocol_endpoint | The class `protocol_endpoint` LUN in the vVol binding. Required in POST. |
| secondary_id | string | The identifier assigned to the vVol binding, known as the secondary LUN ID. The identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>The format for a secondary LUN ID is 16 hexadecimal digits (zero-filled) followed by a lower case "h". |
| svm | svm | The SVM in which the vVol binding and its LUNs are located. Required in POST. |
| vvol | vvol | The class `vvol` LUN in the vVol binding. Required in POST. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 a vVol binding

DELETE `/protocols/san/vvol-bindings/{protocol_endpoint.uuid}/{vvol.uuid}`

**Introduced In:** 9.10

Deletes a vVol binding. The binding between any two specific LUNs is reference counted. When a binding is deleted, the binding count is decremented, but the LUNs remain bound if the resultant reference count is greater than zero. When the binding count reaches zero, the binding is destroyed.

### Related ONTAP commands

- `lun bind destroy`

### Learn more

- DOC /protocols/san/vvol-bindings

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| protocol_endpoint.uuid | string | path | True | The unique identifier of the class `protocol_endpoint` LUN. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| vvol.uuid | string | path | True | The unique identifier of the class `vvol` LUN. |
| delete_all_references | boolean | query | False | Forces deletion of the binding regardless of the reference count value.<br>• Default value: |

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374875 | The vVol binding was not found because the protocol endpoint or vVol LUN was not found. Use to the `target` property of the error object to differentiate between the protocol endpoint LUN and the vVol LUN. |
| 5374926 | The vVol binding was not found. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 |

returned_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 a vVol binding

GET `/protocols/san/vvol-bindings/{protocol_endpoint.uuid}/{vvol.uuid}`

**Introduced In:** 9.10

Retrieves a vVol binding.

### Related ONTAP commands

- `lun bind show`

### Learn more

- DOC /protocols/san/vvol-bindings

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| protocol_endpoint.uuid | string | path | True | The unique identifier of the class `protocol_endpoint` LUN. |
| vvol.uuid | string | path | True | The unique identifier of the class `vvol` LUN. |
| fields | array[string] | query | False | Specify the fields to return. |

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| count | integer | The vVol binding between any two specific LUNs is reference counted. When a REST POST is executed for a vVol binding that already exists, the vVol binding reference count is incremented. When a REST DELETE is executed, the vVol binding reference count is decremented. Only when the vVol binding count reaches zero, or the query parameter `delete_all_references` is supplied, is the vVol binding destroyed. |

| Name | Type | Description |
| --- | --- | --- |
| id | integer | The ONTAP internal identifier assigned to the vVol binding. The bind identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>This property was included in early releases of the REST API for vVols and is maintained for backward compatibility. See the `secondary_id` property, which replaces `id`.<br><br>• example: 1<br>• readOnly: 1<br>• x-ntap-deprecated: 9.13.1<br>• Introduced in: 9.10<br>• x-nullable: true |
| is_optimal | boolean | Indicates if the class `protocol_endpoint` LUN and the class `vvol` LUN are on the same cluster node. |
| protocol_endpoint | protocol_endpoint | The class `protocol_endpoint` LUN in the vVol binding. Required in POST. |
| secondary_id | string | The identifier assigned to the vVol binding, known as the secondary LUN ID. The identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>The format for a secondary LUN ID is 16 hexadecimal digits (zero-filled) followed by a lower case "h". |
| svm | svm | The SVM in which the vVol binding and its LUNs are located. Required in POST. |
| vvol | vvol | The class `vvol` LUN in the vVol binding. Required in POST. |

### Example response


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "count": 1,
  "id": 1,
  "is_optimal": 1,
  "protocol_endpoint": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "/vol/volume1/lun1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "secondary_id": "0000D20000010000h",
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "vvol": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "/vol/volume1/lun1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  }
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 4 | The vVol binding was not found. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

protocol_endpoint

The class `protocol_endpoint` LUN in the vVol binding. Required in POST.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of a LUN. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<namespace>" where the qtree name is optional. |
| uuid | string | The unique identifier of the LUN. |

svm

The SVM in which the vVol binding and its LUNs are located. Required in POST.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

vvol

The class `vvol` LUN in the vVol binding. Required in POST.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of a LUN. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<namespace>" where the qtree name is optional. |
| uuid | string | The unique identifier of the LUN. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

# Manage LUNs

## Manage LUNs

### Overview

A LUN is the logical representation of storage in a storage area network (SAN).

The LUN REST API allows you to create, update, delete, and discover LUNs.

A LUN must be mapped to an initiator group to grant access to the initiator group's initiators (client hosts). Initiators can then access the LUN and perform I/O over a Fibre Channel (FC) fabric using the FC Protocol or a TCP/IP network using iSCSI.

See the LUN object model to learn more about each of the properties supported by the LUN REST API.

A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.

LUN names are paths of the form "/vol/<volume>[/<qtree>]/<lun>" where the qtree name is optional.

A LUN can be created to a specified size using thin or thick provisioning. A LUN can then be renamed, resized, cloned, moved to a different volume and copied. LUNs support the assignment of a QoS policy for performance management or a QoS policy can be assigned to a volume containing one or more LUNs.

### Performance monitoring

Performance of a LUN can be monitored by observing the `metric.*` and `statistics.*` properties. These properties show the space utilization and performance of a LUN in terms of IOPS, latency and throughput. The `metric.*` properties denote an average whereas `statistics.*` properties denote a real-time monotonically increasing value aggregated across all nodes.

### Examples

#### Creating a LUN

This example creates a 300 gigabyte, thin-provisioned LUN in SVM *svm1*, volume *vol1*, configured for use by *linux* initiators. The `return_records` query parameter is used to retrieve properties of the newly created LUN in the POST response.

```
# The API:
POST /api/storage/luns

# The call:
curl -X POST 'https://<mgmt-ip>/api/storage/luns?return_records=true' -H
'Accept: application/hal+json' -d '{ "svm": { "name": "svm1" }, "os_type":
"linux", "space": { "size": "300G" }, "name" : "/vol/vol1/lun1" }'

# The response:
{
"num_records": 1,
"records": [
  {
    "uuid": "5a24ae5b-28af-47fb-b129-5adf6cfba0a6",
    "svm": {
      "uuid": "6bf967fd-2a1c-11e9-b682-005056bbc17d",
      "name": "svm1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/6bf967fd-2a1c-11e9-b682-005056bbc17d"
        }
```

```
      }
    },
    "name": "/vol/vol1/lun1",
    "location": {
      "logical_unit": "lun1",
      "volume": {
        "uuid": "71cd0dba-2a1c-11e9-b682-005056bbc17d",
        "name": "vol1",
        "_links": {
          "self": {
            "href": "/api/storage/volumes/71cd0dba-2a1c-11e9-b682-
005056bbc17d"
          }
        }
      }
    },
    "class": "regular",
    "enabled": true,
    "os_type": "linux",
    "serial_number": "wf0Iq+N4uck3",
    "space": {
      "guarantee": {
        "requested": false,
        "reserved": false
      },
      "scsi_thin_provisioning_support_enabled": true,
      "size": 322163441664,
      "used": 0
    },
    "status": {
      "container_state": "online",
      "read_only": false,
      "state": "online"
    },
    "_links": {
      "self": {
        "href": "/api/storage/luns/5a24ae5b-28af-47fb-b129-5adf6cfba0a6"
      }
    }
  }
]
}
```


#### Updating a LUN

This example sets the `comment` property of a LUN.

```
# The API:
PATCH /api/storage/luns/{uuid}

# The call:
curl -X PATCH 'https://<mgmt-ip>/api/storage/luns/5a24ae5b-28af-47fb-b129-
5adf6cfba0a6' -H 'Accept: application/hal+json' -d '{ "comment": "Data for
the finance department." }'
```

#### Retrieving LUNs

This example retrieves summary information for all online LUNs in SVM *svm1*. The `svm.name` and `status.state` query parameters are used to find the desired LUNs.

```
# The API:
GET /api/storage/luns

# The call:
curl -X GET 'https://<mgmt-
ip>/api/storage/luns?svm.name=svm1&status.state=online' -H 'Accept:
application/hal+json'

# The response:
{
"records": [
  {
    "uuid": "5a24ae5b-28af-47fb-b129-5adf6cfba0a6",
    "svm": {
      "name": "svm1"
    },
    "name": "/vol/vol1/lun1",
    "status": {
      "state": "online"
    },
    "_links": {
      "self": {
        "href": "/api/storage/luns/5a24ae5b-28af-47fb-b129-5adf6cfba0a6"
      }
    }
  },
  {
    "uuid": "c903a978-9bac-4ce9-8237-4a3ba8b13f08",
    "svm": {
      "name": "svm1"
    },
    "name": "/vol/vol1/lun2",
```

```
    "status": {
      "state": "online"
    },
    "_links": {
      "self": {
        "href": "/api/storage/luns/c903a978-9bac-4ce9-8237-4a3ba8b13f08"
      }
    }
  },
  {
    "uuid": "7faf0a9e-0a47-4876-8318-3638d5da16bf",
    "svm": {
      "name": "svm1"
    },
    "name": "/vol/vol2/lun3",
    "status": {
      "state": "online"
    },
    "_links": {
      "self": {
        "href": "/api/storage/luns/7faf0a9e-0a47-4876-8318-3638d5da16bf"
      }
    }
  }
],
"num_records": 3,
"_links": {
  "self": {
    "href": "/api/storage/luns?svm.name=svm1&status.state=online"
  }
}
}
```

**Retrieving details for a specific LUN**

In this example, the `fields` query parameter is used to request all fields, including advanced fields, that would not otherwise be returned by default for the LUN.

```
# The API:
GET /api/storage/luns/{uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/storage/luns/5a24ae5b-28af-47fb-b129-
5adf6cfba0a6?fields=**' -H 'Accept: application/hal+json'

# The response:
```

```
{
"uuid": "5a24ae5b-28af-47fb-b129-5adf6cfba0a6",
"svm": {
  "uuid": "6bf967fd-2a1c-11e9-b682-005056bbc17d",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/6bf967fd-2a1c-11e9-b682-005056bbc17d"
    }
  }
},
"name": "/vol/vol1/lun1",
"location": {
  "logical_unit": "lun1",
  "volume": {
    "uuid": "71cd0dba-2a1c-11e9-b682-005056bbc17d",
    "name": "vol1",
    "_links": {
      "self": {
        "href": "/api/storage/volumes/71cd0dba-2a1c-11e9-b682-
005056bbc17d"
      }
    }
  }
},
"auto_delete": false,
"class": "vvol",
"comment": "Data for the finance department.",
"enabled": true,
"lun_maps": [
  {
    "logical_unit_number": 0,
    "igroup": {
      "uuid": "2b9d57e1-2a66-11e9-b682-005056bbc17d",
      "name": "ig1",
      "_links": {
        "self": {
          "href": "/api/protocols/san/igroups/2b9d57e1-2a66-11e9-b682-
005056bbc17d"
        }
      }
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/lun-maps/5a24ae5b-28af-47fb-b129-
5adf6cfba0a6/2b9d57e1-2a66-11e9-b682-005056bbc17d"
```

```
      }
    }
  }
],
"os_type": "linux",
"serial_number": "wf0Iq+N4uck3",
"space": {
  "guarantee": {
    "requested": false,
    "reserved": false
  },
  "scsi_thin_provisioning_support_enabled": true,
  "size": 322163441664,
  "used": 0
},
"vvol": {
  "is_bound": true,
  "bindings": [
    {
      "id": 4304512,
      "partner": {
        "uuid": "353c7262-be4b-4176-acf3-f1021faa8b64",
        "name": "/vol/vol1/pelun1",
        "_links": {
          "self": {
            "href": "/api/storage/luns/353c7262-be4b-4176-acf3-
f1021faa8b64"
          }
        }
      },
      "_links": {
        "self": {
          "href": "/api/protocols/san/vvol-bindings/353c7262-be4b-4176-
acf3-f1021faa8b64/5a24ae5b-28af-47fb-b129-5adf6cfba0a6"
        }
      }
    }
  ]
},
 "metric": {
  "timestamp": "2019-04-09T05:50:15Z",
  "duration": "PT15S",
  "status": "ok",
  "latency": {
    "other": 0,
    "total": 0,
```

```
    "read": 0,
    "write": 0
  },
  "iops": {
    "read": 0,
    "write": 0,
    "other": 0,
    "total": 0
  },
  "throughput": {
    "read": 0,
    "write": 0,
    "other": 0,
    "total": 0
  }
},
"statistics": {
  "timestamp": "2019-04-09T05:50:42Z",
  "status": "ok",
  "latency_raw": {
    "other": 38298,
    "total": 38298,
    "read": 0,
    "write": 0
  },
  "iops_raw": {
    "read": 0,
    "write": 0,
    "other": 3,
    "total": 3
  },
  "throughput_raw": {
    "read": 0,
    "write": 0,
    "other": 0,
    "total": 0
  }
},
"status": {
  "container_state": "online",
  "mapped": true,
  "read_only": false,
  "state": "online"
},
"consistency_group": {
  "name": "vol1",
```

```
  "uuid": "6d657aaf-b57a-5396-82ea-c01329e46c79",
  "_links": {
    "self": {
      "href": "/api/application/consistency-groups/6d657aaf-b57a-5396-
82ea-c01329e46c79"
    }
  }
},
"_links": {
  "self": {
    "href": "/api/storage/luns/5a24ae5b-28af-47fb-b129-
5adf6cfba0a6?fields=**"
  }
}
}
```

**Deleting a LUN**

```
# The API:
DELETE /api/storage/luns/{uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/storage/luns/c903a978-9bac-4ce9-
8237-4a3ba8b13f08' -H 'Accept: application/hal+json'
```

---

### LUN data

The LUN REST API also supports reading data from and writing data to a LUN via the REST API as multipart/form-data.

Reading data is performed using a GET request on the LUN endpoint. The request header must include `Accept: multipart/form-data`. When this header entry is provided, query parameters `data.offset` and `data.size` are required and used to specify the portion of the LUN's data to read; no other query parameters are allowed. Reads are limited to one megabyte (1MB) per request. Data is returned as `multipart/form-data` content with exactly one form entry containing the data. The form entry has content type `application/octet-stream`.

Writing data is performed using a PATCH request on the LUN endpoint. The request header must include `Content-Type: multipart/form-data`. When this header entry is provided, query parameter `data.offset` is required and used to specify the location within the LUN at which to write the data; no other query parameters are allowed. The request body must be `multipart/form-data` content with exactly one form entry containing the data to write. The content type entry of the form data is ignored and always treated as `application/octet-stream`. Writes are limited to one megabyte (1MB) per request.

#### Reading data from a LUN

```
# The API:
GET /api/storage/luns/{uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/storage/luns/c903a978-9bac-4ce9-8237-
4a3ba8b13f08?data.offset=0&data.size=9' -H 'Accept: multipart/form-data'

# In the response header:
Content-Type: multipart/form-data; boundary="c6e9cf51ab354af0"

# The response body:
--c6e9cf51ab354af0
Content-Disposition: form-data;
Content-Type: application/octet-stream
data here
--c6e9cf51ab354af0--
```

#### Writing data to a LUN

This example writes the contents of a file to a LUN starting at offset 1024.

```
# The API:
PATCH /api/storage/luns/{uuid}

# The call:
curl -X PATCH 'https://<mgmt-ip>/api/storage/luns/c903a978-9bac-4ce9-8237-
4a3ba8b13f08?data.offset=1024' -F "data=@file;type=application/octet-
stream"
```

---

### Cloning LUNs

A clone of a LUN is an independent "copy" of the LUN that shares unchanged data blocks with the original. As blocks of the source and clone are modified, unique blocks are written for each. LUN clones can be created quickly and consume very little space initially. They can be created for the purpose of back-up, or to replicate data for multiple consumers.

Space reservations can be set for the LUN clone independent of the source LUN by setting the `space.guarantee.requested` property in a POST or PATCH request.

A LUN clone can be set to auto-delete by setting the `auto_delete` property. If the LUN's volume is configured for automatic deletion, LUNs that have auto-delete enabled are deleted when a volume is nearly full to reclaim a target amount of free space in the volume.

The value of property `space.scsi_thin_provisioning_support_enabled` is not propagated to the

destination when a LUN is cloned as a new LUN; it is reset to false. The value of this property is maintained from the destination LUN when a LUN is overwritten as a clone.

#### Creating a new LUN clone

You create a new LUN clone as you create any LUN - a POST request to /storage/luns. Set `clone.source.uuid` or `clone.source.name` to identify the source LUN from which the clone is created. The LUN clone and its source must reside in the same volume.

The source LUN can reside in a snapshot, in which case the `clone.source.name` field must be used to identify it. Add `/.snapshot/<snapshot_name>` to the path after the volume name to identify the snapshot. For example `/vol/vol1/.snapshot/snap1/lun1`.

By default, new LUN clones do not inherit the QoS policy of the source LUN; a QoS policy should be set for the clone by setting the `qos_policy` property.

```
# The API:
POST /api/storage/luns

# The call:
curl -X POST 'https://<mgmt-ip>/api/storage/luns' -H 'Accept:
application/hal+json' -d '{ "svm": { "name": "svm1" }, "name":
"/vol/vol1/lun2clone1", "clone": { "source": { "name": "/vol/vol1/lun2" }
}, "qos_policy": { "name": "qos1" } }'
```

#### Over-writing an existing LUN's data as a clone of another

You can overwrite an existing LUN as a clone of another, using a PATCH request to /storage/luns/{uuid}. Set the `clone.source.uuid` or `clone.source.name` property to identify the source LUN from which the clone data is taken. The LUN clone and its source must reside in the same volume.

When used in a PATCH request, the patched LUN's data is overwritten as a clone of the source. The following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: `class`, `auto_delete`, `lun_maps`, `vvol`, `serial_number`, `status.state`, and `uuid`.

Persistent reservations for the updated LUN are also preserved.

```
# The API:
PATCH /api/storage/luns/{uuid}

# The call:
curl -X PATCH 'https://<mgmt-ip>/api/storage/luns/5a24ae5b-28af-47fb-b129-
5adf6cfba0a6' -H 'Accept: application/hal+json' -d '{ "clone": { "source":
{ "name": "/vol/vol1/lun2" } } }'
```

### Converting an NVMe namespace into a LUN

An existing NVMe namespace can be converted in-place to a LUN with no modification to the data blocks. In other words, there is no additional copy created for the data blocks. There are certain requirements for converting an NVMe namespace to a LUN. For instance, the namespace should not be mapped to an NVMe subsystem. Additionally, the namespace should not have a block size other than 512 bytes.

The conversion process updates the metadata to the NVMe namespace, making it a LUN. The conversion is both time and space efficient. After conversion, the new LUN behaves as a regular LUN and may be mapped to an initiator group.

#### Convert an NVMe namespace into a LUN

You convert an NVMe namespace into a LUN by calling a POST to /storage/luns. Set `convert.namespace.uuid` or `convert.namespace.name` to identify the source NVMe namespace which is to be converted in-place into a LUN.

```
# The API:
POST /api/storage/luns

# The call:
curl -X POST 'https://<mgmt-ip>/api/storage/luns' -H 'Accept:
application/hal+json' -d '{ "svm": { "name": "svm1" }, "convert": {
"namespace": { "name": "/vol/vol1/namespace1" } } }'
```

### Moving LUNs between volumes

You move a LUN between volumes by using a PATCH request to /storage/luns/{uuid}. Set the volume portion of the fully qualified LUN path `name` property, `path.volume.uuid`, or `path.volume.name` property to a different volume than the LUN's current volume. Moving a LUN between volumes is an asynchronous activity. A successful request returns a response of 200 synchronously, which indicates that the movement has been successfully queued. The LUN object can then be further polled with a GET request to `/storage/luns/{uuid}` to monitor the status of the movement.

The `movement` sub-object of the LUN object is populated while a LUN movement is in progress and for two minutes following completion of a movement.

#### Starting a LUN movement

```
# The API:
PATCH /api/storage/luns/{uuid}

# The call:
curl -X PATCH 'https://<mgmt-ip>/api/storage/luns/7faf0a9e-0a47-4876-8318-
3638d5da16bf' -H 'Accept: application/hal+json' -d '{ "name":
"/vol/vol1/lun3" }'
```

#### Checking on the status of the LUN movement

```
# The API:
GET /api/storage/luns/{uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/storage/luns/7faf0a9e-0a47-4876-8318-
3638d5da16bf?fields=movement' -H 'Accept: application/hal+json'

# The response:
{
"uuid": "7faf0a9e-0a47-4876-8318-3638d5da16bf",
"name": "/vol/vol1/lun3",
"movement": {
  "paths": {
    "destination": "/vol/vol1/lun3",
    "source": "/vol/vol2/lun3"
  },
  "progress": {
    "elapsed": 1,
    "percent_complete": 0,
    "state": "preparing",
    "volume_snapshot_blocked": false
  }
},
"_links": {
  "self": {
    "href": "/api/storage/luns/7faf0a9e-0a47-4876-8318-3638d5da16bf"
  }
}
}
```

## Retrieve LUNs

GET `/storage/luns`

**Introduced In:** 9.6

Retrieves LUNs.

#### 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.

- attributes.*
- auto_delete
- copy.*
- lun_maps.*
- movement.*
- space.physical_used
- space.physical_used_by_snapshots
- space.efficiency_ratio
- statistics.*
- vvol.bindings.*
- metric.*

### Related ONTAP commands

- lun bind show
- lun copy show
- lun mapping show
- lun move show
- lun show
- volume file clone show-autodelete

### Learn more

- DOC /storage/luns

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| svm.name | string | query | False | Filter by svm.name |
| svm.uuid | string | query | False | Filter by svm.uuid |
| class | string | query | False | Filter by class |
| space.guarantee.reserved | boolean | query | False | Filter by space.guarantee.reserved |
| space.guarantee.requested | boolean | query | False | Filter by space.guarantee.requested |

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| space.size | integer | query | False | Filter by space.size<br>• Max value: 140737488355328<br>• Min value: 4096 |
| space.used | integer | query | False | Filter by space.used |
| space.efficiency_ratio | number | query | False | Filter by space.efficiency_ratio<br>• Introduced in: 9.16 |
| space.physical_used_by_snapshots | integer | query | False | Filter by space.physical_used_by_snapshots<br>• Introduced in: 9.16 |
| space.physical_used | integer | query | False | Filter by space.physical_used<br>• Introduced in: 9.16 |
| space.scsi_thin_provisioning_support_enabled | boolean | query | False | Filter by space.scsi_thin_provisioning_support_enabled<br>• Introduced in: 9.10 |
| metric.status | string | query | False | Filter by metric.status<br>• Introduced in: 9.7 |
| metric.latency.other | integer | query | False | Filter by metric.latency.other<br>• Introduced in: 9.7 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| metric.latency.total | integer | query | False | Filter by metric.latency.total<br>• Introduced in: 9.7 |
| metric.latency.read | integer | query | False | Filter by metric.latency.read<br>• Introduced in: 9.7 |
| metric.latency.write | integer | query | False | Filter by metric.latency.write<br>• Introduced in: 9.7 |
| metric.timestamp | string | query | False | Filter by metric.timestamp<br>• Introduced in: 9.7 |
| metric.throughput.other | integer | query | False | Filter by metric.throughput.other<br>• Introduced in: 9.7 |
| metric.throughput.total | integer | query | False | Filter by metric.throughput.total<br>• Introduced in: 9.7 |
| metric.throughput.read | integer | query | False | Filter by metric.throughput.read<br>• Introduced in: 9.7 |

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| metric.throughput.write | integer | query | False | Filter by metric.throughput.write<br>• Introduced in: 9.7 |
| metric.duration | string | query | False | Filter by metric.duration<br>• Introduced in: 9.7 |
| metric.iops.other | integer | query | False | Filter by metric.iops.other<br>• Introduced in: 9.7 |
| metric.iops.total | integer | query | False | Filter by metric.iops.total<br>• Introduced in: 9.7 |
| metric.iops.read | integer | query | False | Filter by metric.iops.read<br>• Introduced in: 9.7 |
| metric.iops.write | integer | query | False | Filter by metric.iops.write<br>• Introduced in: 9.7 |
| comment | string | query | False | Filter by comment<br>• maxLength: 254<br>• minLength: 0 |

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| attributes.value | string | query | False | Filter by attributes.value<br>• Introduced in: 9.10<br>• maxLength: 4091<br>• minLength: 1 |
| attributes.name | string | query | False | Filter by attributes.name<br>• Introduced in: 9.10<br>• maxLength: 4091<br>• minLength: 1 |
| movement.paths.source | string | query | False | Filter by movement.paths.source |
| movement.paths.destination | string | query | False | Filter by movement.paths.destination |
| movement.progress.elapsed | integer | query | False | Filter by movement.progress.elapsed |
| movement.progress.state | string | query | False | Filter by movement.progress.state |
| movement.progress.percent_complete | integer | query | False | Filter by movement.progress.percent_complete<br>• Max value: 100<br>• Min value: 0 |
| movement.progress.failure.arguments.code | string | query | False | Filter by movement.progress.failure.arguments.code |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| movement.progress.failure.arguments.message | string | query | False | Filter by movement.progress.failure.arguments.message |
| movement.progress.failure.message | string | query | False | Filter by movement.progress.failure.message |
| movement.progress.failure.code | string | query | False | Filter by movement.progress.failure.code |
| movement.progress.volume_snapshot_blocked | boolean | query | False | Filter by movement.progress.volume_snapshot_blocked |
| movement.max_throughput | integer | query | False | Filter by movement.max_throughput |
| lun_maps.logical_unit_number | integer | query | False | Filter by lun_maps.logical_unit_number |
| lun_maps.igroup.protocol | string | query | False | Filter by lun_maps.igroup.protocol<br>• Introduced in: 9.16 |
| lun_maps.igroup.comment | string | query | False | Filter by lun_maps.igroup.comment<br>• Introduced in: 9.16<br>• maxLength: 254<br>• minLength: 0 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun_maps.igroup.igroups.uuid | string | query | False | Filter by lun_maps.igroup.igroups.uuid<br>• Introduced in: 9.16 |
| lun_maps.igroup.igroups.name | string | query | False | Filter by lun_maps.igroup.igroups.name<br>• Introduced in: 9.16<br>• maxLength: 96<br>• minLength: 1 |
| lun_maps.igroup.initiators.name | string | query | False | Filter by lun_maps.igroup.initiators.name<br>• Introduced in: 9.16 |
| lun_maps.igroup.initiators.comment | string | query | False | Filter by lun_maps.igroup.initiators.comment<br>• Introduced in: 9.16<br>• maxLength: 254<br>• minLength: 0 |
| lun_maps.igroup.os_type | string | query | False | Filter by lun_maps.igroup.os_type<br>• Introduced in: 9.16 |
| lun_maps.igroup.name | string | query | False | Filter by lun_maps.igroup.name |
| lun_maps.igroup.uuid | string | query | False | Filter by lun_maps.igroup.uuid |

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| status.container_state | string | query | False | Filter by status.container_state |
| status.read_only | boolean | query | False | Filter by status.read_only |
| status.state | string | query | False | Filter by status.state |
| status.mapped | boolean | query | False | Filter by status.mapped |
| qos_policy.uuid | string | query | False | Filter by qos_policy.uuid |
| qos_policy.name | string | query | False | Filter by qos_policy.name |
| serial_number_hex | string | query | False | Filter by serial_number_hex<br>• Introduced in: 9.17 |
| vvol.is_bound | boolean | query | False | Filter by vvol.is_bound<br>• Introduced in: 9.10 |
| vvol.bindings.secondary_id | string | query | False | Filter by vvol.bindings.secondary_id<br>• Introduced in: 9.13 |
| vvol.bindings.partner.name | string | query | False | Filter by vvol.bindings.partner.name<br>• Introduced in: 9.10 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| vvol.bindings.partner.uuid | string | query | False | Filter by vvol.bindings.partner.uuid<br>• Introduced in: 9.10 |
| vvol.bindings.id | integer | query | False | Filter by vvol.bindings.id<br>• Introduced in: 9.10 |
| serial_number | string | query | False | Filter by serial_number<br>• maxLength: 12<br>• minLength: 12 |
| uuid | string | query | False | Filter by uuid |
| consistency_group.uuid | string | query | False | Filter by consistency_group.uuid<br>• Introduced in: 9.10 |
| consistency_group.name | string | query | False | Filter by consistency_group.name<br>• Introduced in: 9.10 |
| location.logical_unit | string | query | False | Filter by location.logical_unit |
| location.qtree.name | string | query | False | Filter by location.qtree.name |
| location.qtree.id | integer | query | False | Filter by location.qtree.id<br>• Max value: 4994<br>• Min value: 0 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| location.volume.uuid | string | query | False | Filter by location.volume.uuid |
| location.volume.name | string | query | False | Filter by location.volume.name |
| location.node.uuid | string | query | False | Filter by location.node.uuid<br>• Introduced in: 9.10 |
| location.node.name | string | query | False | Filter by location.node.name<br>• Introduced in: 9.10 |
| statistics.throughput_raw.other | integer | query | False | Filter by statistics.throughput_raw.other<br>• Introduced in: 9.7 |
| statistics.throughput_raw.total | integer | query | False | Filter by statistics.throughput_raw.total<br>• Introduced in: 9.7 |
| statistics.throughput_raw.read | integer | query | False | Filter by statistics.throughput_raw.read<br>• Introduced in: 9.7 |
| statistics.throughput_raw.write | integer | query | False | Filter by statistics.throughput_raw.write<br>• Introduced in: 9.7 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| statistics.iops_raw.other | integer | query | False | Filter by statistics.iops_raw.other<br>• Introduced in: 9.7 |
| statistics.iops_raw.total | integer | query | False | Filter by statistics.iops_raw.total<br>• Introduced in: 9.7 |
| statistics.iops_raw.read | integer | query | False | Filter by statistics.iops_raw.read<br>• Introduced in: 9.7 |
| statistics.iops_raw.write | integer | query | False | Filter by statistics.iops_raw.write<br>• Introduced in: 9.7 |
| statistics.latency_raw.other | integer | query | False | Filter by statistics.latency_raw.other<br>• Introduced in: 9.7 |
| statistics.latency_raw.total | integer | query | False | Filter by statistics.latency_raw.total<br>• Introduced in: 9.7 |
| statistics.latency_raw.read | integer | query | False | Filter by statistics.latency_raw.read<br>• Introduced in: 9.7 |

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| statistics.latency_raw.write | integer | query | False | Filter by statistics.latency_raw.write<br>• Introduced in: 9.7 |
| statistics.timestamp | string | query | False | Filter by statistics.timestamp<br>• Introduced in: 9.7 |
| statistics.status | string | query | False | Filter by statistics.status<br>• Introduced in: 9.7 |
| enabled | boolean | query | False | Filter by enabled |
| copy.destinations.uuid | string | query | False | Filter by copy.destinations.uuid<br>• Introduced in: 9.10 |
| copy.destinations.name | string | query | False | Filter by copy.destinations.name<br>• Introduced in: 9.10 |
| copy.destinations.progress.volume_snapshot_blocked | boolean | query | False | Filter by copy.destinations.progress.volume_snapshot_blocked<br>• Introduced in: 9.10 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| copy.destinations.progress.failure.arguments.code | string | query | False | Filter by copy.destinations.progress.failure.arguments.code<br>• Introduced in: 9.10 |
| copy.destinations.progress.failure.arguments.message | string | query | False | Filter by copy.destinations.progress.failure.arguments.message<br>• Introduced in: 9.10 |
| copy.destinations.progress.failure.message | string | query | False | Filter by copy.destinations.progress.failure.message<br>• Introduced in: 9.10 |
| copy.destinations.progress.failure.code | string | query | False | Filter by copy.destinations.progress.failure.code<br>• Introduced in: 9.10 |
| copy.destinations.progress.state | string | query | False | Filter by copy.destinations.progress.state<br>• Introduced in: 9.10 |
| copy.destinations.progress.percent_complete | integer | query | False | Filter by copy.destinations.progress.percent_complete<br>• Introduced in: 9.10<br>• Max value: 100<br>• Min value: 0 |

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| copy.destinations.progress.elapsed | integer | query | False | Filter by copy.destinations.progress.elapsed<br>• Introduced in: 9.10 |
| copy.destinations.peer.uuid | string | query | False | Filter by copy.destinations.peer.uuid<br>• Introduced in: 9.14 |
| copy.destinations.peer.name | string | query | False | Filter by copy.destinations.peer.name<br>• Introduced in: 9.14 |
| copy.destinations.max_throughput | integer | query | False | Filter by copy.destinations.max_throughput<br>• Introduced in: 9.10 |
| copy.source.max_throughput | integer | query | False | Filter by copy.source.max_throughput<br>• Introduced in: 9.10 |
| copy.source.peer.uuid | string | query | False | Filter by copy.source.peer.uuid<br>• Introduced in: 9.14 |
| copy.source.peer.name | string | query | False | Filter by copy.source.peer.name<br>• Introduced in: 9.14 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| copy.source.progress.volume_snapshot_blocked | boolean | query | False | Filter by copy.source.progress.volume_snapshot_blocked<br>• Introduced in: 9.10 |
| copy.source.progress.state | string | query | False | Filter by copy.source.progress.state<br>• Introduced in: 9.10 |
| copy.source.progress.failure.arguments.code | string | query | False | Filter by copy.source.progress.failure.arguments.code<br>• Introduced in: 9.10 |
| copy.source.progress.failure.arguments.message | string | query | False | Filter by copy.source.progress.failure.arguments.message<br>• Introduced in: 9.10 |
| copy.source.progress.failure.message | string | query | False | Filter by copy.source.progress.failure.message<br>• Introduced in: 9.10 |
| copy.source.progress.failure.code | string | query | False | Filter by copy.source.progress.failure.code<br>• Introduced in: 9.10 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| copy.source.progress.percent_complete | integer | query | False | Filter by copy.source.progress.percent_complete<br>• Introduced in: 9.10<br>• Max value: 100<br>• Min value: 0 |
| copy.source.progress.elapsed | integer | query | False | Filter by copy.source.progress.elapsed<br>• Introduced in: 9.10 |
| copy.source.name | string | query | False | Filter by copy.source.name<br>• Introduced in: 9.10 |
| copy.source.uuid | string | query | False | Filter by copy.source.uuid<br>• Introduced in: 9.10 |
| auto_delete | boolean | query | False | Filter by auto_delete |
| os_type | string | query | False | Filter by os_type |
| name | string | query | False | Filter by name |
| create_time | string | query | False | Filter by create_time<br>• Introduced in: 9.7 |
| fields | array[string] | query | False | Specify the fields to return. |
| max_records | integer | query | False | Limit the number of records returned. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| return_records | boolean | query | False | The default is true for GET calls. When set to false, only the number of records is returned.<br>• 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.<br>• Max value: 120<br>• Min value: 0<br>• Default value: 15 |
| 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[lun] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "attributes": [
        {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "name1",
          "value": "value1"
        }
      ],
      "class": "string",
      "comment": "string",
      "consistency_group": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "cg1",
        "uuid": "4abc2317-4332-9d37-93a0-20bd29c22df0"
      },
      "copy": {
        "destinations": [
          {
            "_links": {
              "self": {
```

```
            "href": "/api/resourcelink"
          }
        },
        "max_throughput": 0,
        "name": "/vol/vol1/lun1",
        "peer": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "peer1",
          "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
        },
        "progress": {
          "elapsed": 0,
          "failure": {
            "arguments": [
              {
                "code": "string",
                "message": "string"
              }
            ],
            "code": "4",
            "message": "entry doesn't exist"
          },
          "percent_complete": 0,
          "state": "string"
        },
        "uuid": "1bc327d5-4654-5284-a116-f182282240b4"
      }
    ],
    "source": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "/vol/vol2/lun1",
      "peer": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "peer1",
```

```
        "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
      },
      "progress": {
        "elapsed": 0,
        "failure": {
          "arguments": [
            {
              "code": "string",
              "message": "string"
            }
          ],
          "code": "4",
          "message": "entry doesn't exist"
        },
        "percent_complete": 0,
        "state": "string"
      },
      "uuid": "03c05019-40d9-3945-c767-dca4c3be5e90"
    }
  },
  "create_time": "2018-06-04 15:00:00 -0400",
  "location": {
    "logical_unit": "lun1",
    "node": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "node1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "qtree": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "id": 1,
      "name": "qt1"
    },
    "volume": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
```

```
        },
        "name": "volume1",
        "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
      }
    },
    "lun_maps": [
      {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "igroup": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "comment": "string",
          "igroups": [
            {
              "_links": {
                "self": {
                  "href": "/api/resourcelink"
                }
              },
              "name": "igroup1",
              "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
            }
          ],
          "initiators": [
            {
              "comment": "my comment",
              "name": "iqn.1998-01.com.corp.iscsi:name1"
            }
          ],
          "name": "igroup1",
          "os_type": "string",
          "protocol": "string",
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        }
      }
    ],
    "metric": {
      "_links": {
        "self": {
```

```
            "href": "/api/resourcelink"
          }
        },
        "duration": "PT15S",
        "iops": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "latency": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "status": "ok",
        "throughput": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "timestamp": "2017-01-25 06:20:13 -0500"
      },
      "movement": {
        "paths": {
          "destination": "/vol/vol1/lun1",
          "source": "/vol/vol2/lun2"
        },
        "progress": {
          "elapsed": 0,
          "failure": {
            "arguments": [
              {
                "code": "string",
                "message": "string"
              }
            ],
            "code": "4",
            "message": "entry doesn't exist"
          },
          "percent_complete": 0,
          "state": "string"
        }
      },
      "name": "/vol/volume1/qtree1/lun1",
      "os_type": "string",
      "qos_policy": {
```

```
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "qos1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "serial_number": "string",
    "serial_number_hex": "string",
    "space": {
      "efficiency_ratio": 2.5,
      "physical_used": 1073741824,
      "physical_used_by_snapshots": 1073741824,
      "size": 1073741824,
      "used": 0
    },
    "statistics": {
      "iops_raw": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "latency_raw": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "status": "ok",
      "throughput_raw": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "timestamp": "2017-01-25 06:20:13 -0500"
    },
    "status": {
      "container_state": "string",
      "state": "online"
    },
    "svm": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
```

```
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
      "vvol": {
        "bindings": [
          {
            "_links": {
              "self": {
                "href": "/api/resourcelink"
              }
            },
            "id": 1,
            "partner": {
              "_links": {
                "self": {
                  "href": "/api/resourcelink"
                }
              },
              "name": "/vol/vol1/lun1",
              "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
            },
            "secondary_id": "0000D20000010000h"
          }
        ]
      }
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

attributes

A name/value pair optionally stored with the LUN. Attributes are available to callers to persist small amounts of application-specific metadata. They are in no way interpreted by ONTAP.

Attribute names and values must be at least one byte and no more than 4091 bytes in length. The sum of the name and value lengths must be no more than 4092 bytes.

Optional in POST.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The attribute name. |
| value | string | The attribute value. |

source

The source LUN for a LUN clone operation. This can be specified using property `clone.source.uuid` or `clone.source.name`. If both properties are supplied, they must refer to the same LUN.

Valid in POST to create a new LUN as a clone of the source.

Valid in PATCH to overwrite an existing LUN's data as a clone of another.

clone

This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: `auto_delete`,

`qos_policy`, `space.guarantee.requested` and `space.scsi_thin_provisioning_support_enabled`.

When used in a PATCH, the patched LUN's data is over-written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: `class`, `auto_delete`, `lun_maps`, `serial_number`, `status.state`, and `uuid`.

Persistent reservations for the patched LUN are also preserved.

consistency_group

The LUN's consistency group. This property is populated for LUNs that are members of a consistency group. If the LUN is a member of a child consistency group, the parent consistency group is reported. A LUN's consistency group is the consistency group of its containing volume.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the consistency group. |
| uuid | string | The unique identifier of the consistency group. |

namespace

The source namespace for convert operation. This can be specified using property `convert.namespace.uuid` or `convert.namespace.name`. If both properties are supplied, they must refer to the same NVMe namespace.

Valid in POST. A convert request from NVMe namespace to LUN cannot be combined with setting any other LUN properties. All other properties of the converted LUN come from the source NVMe namespace.

convert

This sub-object is used in POST to convert a valid in-place NVMe namespace to a LUN. Setting a property in this sub-object indicates that a conversion from the specified NVMe namespace to LUN is desired.

peer

The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.

This is only populated by GET when the LUN copy is inter-SVM.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

error

Error information provided if the asynchronous LUN copy operation fails.

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

progress

Properties related to the progress of an active or recently completed LUN copy.

| Name | Type | Description |
|---|---|---|
| elapsed | integer | The amount of time that has elapsed since the start of the LUN copy, in seconds. |
| failure | error | Error information provided if the asynchronous LUN copy operation fails. |
| percent_complete | integer | The percentage completed of the LUN copy. |
| state | string | The state of the LUN copy. |

| Name | Type | Description |
|---|---|---|
| volume_snapshot_blocked | boolean | This property reports if volume snapshots are blocked by the LUN copy. This property can be polled to identify when volume snapshots can be resumed after beginning a LUN copy. |

destinations

A LUN copy operation in which the containing LUN is the source of the copy.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| max_throughput | integer | The maximum data throughput, in bytes per second, that should be utilized in support of the LUN copy. See property `copy.source.max_throughput` for further details. |
| name | string | The fully qualified path of the LUN copy destination composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN. |
| peer | peer | The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.<br><br>This is only populated by GET when the LUN copy is inter-SVM. |
| progress | progress | Properties related to the progress of an active or recently completed LUN copy. |
| uuid | string | The unique identifier of the LUN copy destination. |

peer

The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.

Set this in POST to specify the source SVM for an inter-SVM LUN copy. Only populated by GET when the LUN copy is inter-SVM.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

progress

Properties related to the progress of an active or recently completed LUN copy.

| Name | Type | Description |
|---|---|---|
| elapsed | integer | The amount of time that has elapsed since the start of the LUN copy, in seconds. |
| failure | error | Error information provided if the asynchronous LUN copy operation fails. |
| percent_complete | integer | The percentage completed of the LUN copy. |
| state | string | The state of the LUN copy.<br><br>Valid in PATCH when an LUN copy is active. Set to *paused* to pause a LUN copy. Set to *replicating* to resume a paused LUN copy. |
| volume_snapshot_blocked | boolean | This property reports if volume snapshots are blocked by the LUN copy. This property can be polled to identify when volume snapshots can be resumed after beginning a LUN copy. |

source

The source LUN of a LUN copy operation in which the containing LUN is the destination of the copy.

Valid in POST except when creating a LUN clone. A LUN copy request cannot be combined with setting any other LUN properties except the destination location. All other properties of the destination LUN come from the source LUN.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| max_throughput | integer | The maximum data throughput, in bytes per second, that should be utilized in support of the LUN copy. This property can be used to throttle a transfer and limit its impact on the performance of the source and destination nodes. The specified value will be rounded up to the nearest megabyte.<br><br>If this property is not specified in a POST that begins a LUN copy, throttling is not applied to the data transfer.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>Valid only in a POST that begins a LUN copy or a PATCH when a LUN copy is already in process.<br><br>• format: int64<br>• Introduced in: 9.10<br>• x-nullable: true |
| name | string | The fully qualified path of the LUN copy source composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN.<br><br>Set this property in POST to specify the source for a LUN copy operation. |

| Name | Type | Description |
| --- | --- | --- |
| peer | peer | The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.<br><br>Set this in POST to specify the source SVM for an inter-SVM LUN copy. Only populated by GET when the LUN copy is inter-SVM. |
| progress | progress | Properties related to the progress of an active or recently completed LUN copy. |
| uuid | string | The unique identifier of the LUN copy source.<br><br>Set this property in POST to specify the source for a LUN copy operation. |

copy

This sub-object applies to LUN copy operations. A LUN can be copied with a POST request that supplies `copy.source` properties.

Copying a LUN is an asynchronous activity begun by a POST request that specifies the source of the copy in the `copy.source` properties. The data for the LUN is then asynchronously copied from the source to the destination. The time required to complete the copy depends on the size of the LUN and the load on the cluster. The `copy` sub-object is populated while a LUN copy is in progress and for two (2) minutes following completion of a copy.

While LUNs are being copied, the status of the LUN copy operations can be obtained using a GET of the source or destination LUN that requests the `copy` properties. If the LUN is the source LUN for one or more copy operations, the `copy.destinations` array is populated in GET. If the containing LUN is the destination LUN for a copy operation, the `copy.source` sub-object is populated in GET. The LUN copy operation can be further modified using a PATCH on the properties on the `copy.source` sub-object of the copy destination LUN.

There is an added computational cost to retrieving property values for `copy`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

| Name | Type | Description |
|---|---|---|
| destinations | array[destinations] | An array of destination LUNs of LUN copy operations in which the containing LUN is the source of the copy. |
| source | source | The source LUN of a LUN copy operation in which the containing LUN is the destination of the copy.<br><br>Valid in POST except when creating a LUN clone. A LUN copy request cannot be combined with setting any other LUN properties except the destination location. All other properties of the destination LUN come from the source LUN. |

node

The cluster node that hosts the LUN.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | |
| uuid | string | |

qtree

The qtree in which the LUN is optionally located. Valid in POST and PATCH.

If properties `name` and `location.qtree.name` and/or `location.qtree.id` are specified in the same request, they must refer to the same qtree.

A PATCH that modifies the qtree of the LUN is considered a rename operation.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| id | integer | The identifier for the qtree, unique within the qtree's volume. |
| name | string | The name of the qtree. |

volume

The volume in which the LUN is located. Valid in POST and PATCH.

If properties `name` and `location.volume.name` and/or `location.volume.uuid` are specified in the same request, they must refer to the same volume.

A PATCH that modifies the volume of the LUN begins an asynchronous LUN movement operation.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the volume. This field cannot be specified in a PATCH method. |
| uuid | string | Unique identifier for the volume. This corresponds to the instance-uuid that is exposed in the CLI and ONTAPI. It does not change due to a volume move.<br>• example: 028baa66-41bd-11e9-81d5-00a0986138f7<br>• Introduced in: 9.6<br>• x-nullable: true |

location

The location of the LUN within the ONTAP cluster. LUNs support rename and move between volumes. Valid in POST and PATCH.

| Name | Type | Description |
|---|---|---|
| logical_unit | string | The base name component of the LUN. Valid in POST and PATCH.<br><br>If properties `name` and `location.logical_unit` are specified in the same request, they must refer to the base name.<br><br>A PATCH that modifies the base name of the LUN is considered a rename operation. |
| node | node | The cluster node that hosts the LUN. |

| Name | Type | Description |
| --- | --- | --- |
| qtree | qtree | The qtree in which the LUN is optionally located. Valid in POST and PATCH.<br><br>If properties `name` and `location.qtree.name` and/or `location.qtree.id` are specified in the same request, they must refer to the same qtree.<br><br>A PATCH that modifies the qtree of the LUN is considered a rename operation. |
| volume | volume | The volume in which the LUN is located. Valid in POST and PATCH.<br><br>If properties `name` and `location.volume.name` and/or `location.volume.uuid` are specified in the same request, they must refer to the same volume.<br><br>A PATCH that modifies the volume of the LUN begins an asynchronous LUN movement operation. |

self_link

| Name | Type | Description |
| --- | --- | --- |
| self | href | |

igroups

| Name | Type | Description |
| --- | --- | --- |
| _links | self_link | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

initiators

The initiators that are members of the initiator group.

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. |
| name | string | Name of initiator that is a member of the initiator group. |

igroup

The initiator group to which the LUN is mapped.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| igroups | array[igroups] | The existing initiator groups that are members of the group. Optional in POST.<br><br>This property is mutually exclusive with the *initiators* property during POST.<br><br>This array contains only the direct children of the initiator group. If the member initiator groups have further nested initiator groups, those are reported in the `igroups` property of the child initiator group.<br><br>Zero or more nested initiator groups can be supplied when the initiator group is created. The initiator group will act as if it contains the aggregation of all initiators in any nested initiator groups.<br><br>After creation, nested initiator groups can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/igroups` endpoint. See DELETE /protocols/san/igroups/{igroup.uuid}/igroups/{uuid} for more details. |
| initiators | array[initiators] | The initiators that are members of the group. |
| name | string | The name of the initiator group. |
| os_type | string | The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. |

| Name | Type | Description |
|---|---|---|
| protocol | string | The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. Optional in POST; if not supplied, this defaults to *mixed*.<br><br>The protocol of an initiator group cannot be changed after creation of the group. |
| uuid | string | The unique identifier of the initiator group. |

lun_maps

A LUN map is an association between a LUN and an initiator group.

When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| igroup | igroup | The initiator group to which the LUN is mapped. |
| logical_unit_number | integer | The logical unit number assigned to the LUN when mapped to the specified initiator group. The number is used to identify the LUN to initiators in the initiator group when communicating through the Fibre Channel Protocol or iSCSI. Optional in POST; if no value is provided, ONTAP assigns the lowest available value. This property is not supported when the *provisioning_options.count* property is 2 or more.<br><br>• Introduced in: 9.6<br>• readCreate: 1<br>• x-nullable: true |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

metric

Performance numbers, such as IOPS latency and throughput.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |

| Name | Type | Description |
| --- | --- | --- |
| status | string | Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

paths

The fully qualified LUN path names involved in the LUN movement.

| Name | Type | Description |
| --- | --- | --- |
| destination | string | The fully qualified path of the LUN movement destination composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN. |

| Name | Type | Description |
| --- | --- | --- |
| source | string | The fully qualified path of the LUN movement source composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN. |

error

Error information provided if the asynchronous LUN movement operation fails.

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

progress

Properties related to the progress of an active or recently completed LUN movement.

| Name | Type | Description |
| --- | --- | --- |
| elapsed | integer | The amount of time that has elapsed since the start of the LUN movement, in seconds. |
| failure | error | Error information provided if the asynchronous LUN movement operation fails. |
| percent_complete | integer | The percentage completed of the LUN movement. |
| state | string | The state of the LUN movement.<br><br>Valid in PATCH when an LUN movement is active. Set to *paused* to pause a LUN movement. Set to *replicating* to resume a paused LUN movement. |

| Name | Type | Description |
| --- | --- | --- |
| volume_snapshot_blocked | boolean | This property reports if volume snapshots are blocked by the LUN movement. This property can be polled to identify when volume snapshots can be resumed after beginning a LUN movement. |

movement

This sub-object applies to LUN movement between volumes. A LUN can be moved to a new volume with a PATCH request that changes either the volume portion of property `name`, `location.volume.uuid`, or `location.volume.name`. If the volume is changed using more than one of these properties, the supplied properties used must refer to the same volume.

Moving a LUN between volumes is an asynchronous activity begun by a PATCH request. The data for the LUN is then asynchronously copied from the source volume to the destination volume. The time required to complete the move depends on the size of the LUN and the load on the cluster. The `movement` sub-object is populated while a LUN movement is in progress and for two (2) minutes following completion of a movement.

While the LUN is being moved, the status of the LUN movement operation can be obtained using a GET for the LUN that requests the `movement` properties. The LUN movement operation can be further modified using a PATCH on the properties on the `movement` sub-object.

There is an added computational cost to retrieving property values for `movement`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

| Name | Type | Description |
| --- | --- | --- |
| max_throughput | integer | The maximum data throughput, in bytes per second, that should be utilized in support of the LUN movement. This property can be used to throttle a transfer and limit its impact on the performance of the source and destination nodes. The specified value will be rounded up to the nearest megabyte.<br><br>If this property is not specified in a POST that begins a LUN movement, throttling is not applied to the data transfer.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>This property is valid only in a POST that begins a LUN movement or a PATCH when a LUN movement is already in process.<br><br>• format: int64<br>• Introduced in: 9.6<br>• x-nullable: true |
| paths | paths | The fully qualified LUN path names involved in the LUN movement. |
| progress | progress | Properties related to the progress of an active or recently completed LUN movement. |

qos_policy

The QoS policy for the volume provisioned to host the LUN. This property is only supported when the request provisions a new volume. This property is mutually exclusive with the LUN granular `qos_policy`. If no `qos_policy` is provided at LUN or volume granularity, a volume granular policy is set based on the `storage_service.name`, which defaults to the most performant service available.

| Name | Type | Description |
| --- | --- | --- |
| _links | self_link | |

| Name | Type | Description |
|---|---|---|
| name | string | The QoS policy group name. This is mutually exclusive with UUID and other QoS attributes during POST and PATCH. |
| uuid | string | The QoS policy group UUID. This is mutually exclusive with name and other QoS attributes during POST and PATCH. |

snapshot_policy_reference

This is a reference to the snapshot policy.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | |
| uuid | string | |

storage_service

Determines the placement of the LUN based on the value specified. This property is only supported for regular and vvol LUNs. Valid in POST.

object_stores

tiering

The tiering placement and policy definitions for the volume provisioned to host the LUN. This property is only supported when the request provisions a new volume.

| Name | Type | Description |
|---|---|---|
| object_stores | array[object_stores] | Object stores to use. Used for placement. |

| Name | Type | Description |
| --- | --- | --- |
| policy | string | Policy that determines whether the user data blocks of a volume in a FabricPool will be tiered to the cloud store when they become cold.<br><br>FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of a volume block increases if it is accessed frequently and decreases when it is not. Valid in POST or PATCH.<br><br>all ‐ Allows tiering of both snapshots and active file system user data to the cloud store as soon as possible by ignoring the temperature on the volume blocks.<br><br>auto ‐ Allows tiering of both snapshot and active file system user data to the cloud store<br><br>none ‐ Volume blocks are not be tiered to the cloud store.<br><br>snapshot_only ‐ Allows tiering of only the volume snapshots not associated with the active file system.<br><br>The default tiering policy is "snapshot-only" for a FlexVol volume and "none" for a FlexGroup volume. The default minimum cooling period for the "snapshot-only" tiering policy is 2 days and for the "auto" tiering policy it is 31 days. |

provisioning_options

Options that are applied to the operation.

| Name | Type | Description |
|---|---|---|
| storage_service | storage_service | Determines the placement of the LUN based on the value specified. This property is only supported for regular and vvol LUNs. Valid in POST. |

qos_policy

The QoS policy for the LUN. Both traditional and adaptive QoS policies are supported. If both property `qos_policy.uuid` and `qos_policy.name` are specified in the same request, they must refer to the same QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set the property `qos_policy.name` to an empty string ("") in a PATCH request. Valid in POST and PATCH.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set this property to an empty string ("") in a PATCH request. Valid in POST and PATCH. |
| uuid | string | The unique identifier of the QoS policy. Valid in POST and PATCH. |

guarantee

Properties that request and report the space guarantee for the LUN.

| Name | Type | Description |
|---|---|---|
| requested | boolean | The requested space reservation policy for the LUN. If *true*, a space reservation is requested for the LUN; if *false*, the LUN is thin provisioned. Guaranteeing a space reservation request for a LUN requires that the volume in which the LUN resides is also space reserved and that the fractional reserve for the volume is 100%. Valid in POST and PATCH. This property is caller settable as described above. |

| Name | Type | Description |
|---|---|---|
| reserved | boolean | Reports if the LUN is space guaranteed.<br><br>If *true*, a space guarantee is requested and the containing volume and aggregate support the request. If *false*, a space guarantee is not requested or a space guarantee is requested and either the containing volume or aggregate do not support the request. |

space

The storage space related properties of the LUN.

| Name | Type | Description |
|---|---|---|
| efficiency_ratio | number | The storage efficiency ratio of the LUN without snapshots. (Logical Used / Used) This property is not available on the LUN object in the REST API and is not reported for GET requests. See the containing volume object for this information. |
| guarantee | guarantee | Properties that request and report the space guarantee for the LUN. |
| physical_used | integer | The number of bytes consumed on the disk by the LUN, excluding snapshots. This property is not available on the LUN object in the REST API and is not reported for GET requests. See the containing volume object for this information. |
| physical_used_by_snapshots | integer | The number of bytes consumed on the disk by the LUN's snapshots. This property is not available on the LUN object in the REST API and is not reported for GET requests. See the containing volume object for this information. |

| Name | Type | Description |
| --- | --- | --- |
| scsi_thin_provisioning_support_enabled | boolean | To leverage the benefits of SCSI thin provisioning, it must be supported by your host. SCSI thin provisioning uses the Logical Block Provisioning feature as defined in the SCSI SBC-3 standard. Only hosts that support this standard can use SCSI thin provisioning in ONTAP.<br><br>When you disable SCSI thin provisioning support in ONTAP, you turn off the following SCSI thin provisioning features:<br><br>• Unmapping and reporting space usage for space reclamation<br>• Reporting resource exhaustion errors<br><br>The value of this property is not propagated to the destination when a LUN is cloned as a new LUN or copied; it is reset to false. The value of this property is maintained from the destination LUN when a LUN is overwritten as a clone.<br><br>Valid in POST and PATCH.<br><br>• Default value: 1<br>• Introduced in: 9.10<br>• x-nullable: true |

| Name | Type | Description |
| --- | --- | --- |
| size | integer | The total provisioned size of the LUN. The LUN size can be increased but not decreased using the REST interface.<br><br>The maximum and minimum sizes listed here are the absolute maximum and absolute minimum sizes, in bytes. The actual minimum and maximum sizes vary depending on the ONTAP version, ONTAP platform and the available space in the containing volume and aggregate.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>• example: 1073741824<br>• format: int64<br>• Max value: 140737488355328<br>• Min value: 4096<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
|---|---|---|
| used | integer | The amount of space consumed by the main data stream of the LUN.<br><br>This value is the total space consumed in the volume by the LUN, including filesystem overhead, but excluding prefix and suffix streams. Due to internal filesystem overhead and the many ways SAN filesystems and applications utilize blocks within a LUN, this value does not necessarily reflect actual consumption/availability from the perspective of the filesystem or application. Without specific knowledge of how the LUN blocks are utilized outside of ONTAP, this property should not be used as an indicator for an out-of-space condition.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>• format: int64<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

iops_raw

The number of I/O operations observed at the storage object. This can be used along with delta time to calculate the rate of I/O operations per unit of time.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency_raw

The raw latency in microseconds observed at the storage object. This can be divided by the raw IOPS value to calculate the average latency per I/O operation.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput_raw

Throughput bytes observed at the storage object. This can be used along with delta time to calculate the rate of throughput bytes per unit of time.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |

| Name | Type | Description |
|---|---|---|
| write | integer | Performance metric for write I/O operations. |

statistics

These are raw performance numbers, such as IOPS latency and throughput. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

| Name | Type | Description |
|---|---|---|
| iops_raw | iops_raw | The number of I/O operations observed at the storage object. This can be used along with delta time to calculate the rate of I/O operations per unit of time. |
| latency_raw | latency_raw | The raw latency in microseconds observed at the storage object. This can be divided by the raw IOPS value to calculate the average latency per I/O operation. |
| status | string | Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |

| Name | Type | Description |
|---|---|---|
| throughput_raw | throughput_raw | Throughput bytes observed at the storage object. This can be used along with delta time to calculate the rate of throughput bytes per unit of time. |
| timestamp | string | The timestamp of the performance data. |

status

Status information about the LUN.

| Name | Type | Description |
|---|---|---|
| container_state | string | The state of the volume and aggregate that contain the LUN. LUNs are only available when their containers are available. |
| mapped | boolean | Reports if the LUN is mapped to one or more initiator groups.<br><br>There is an added computational cost to retrieving this property's value. It is not populated for a GET request unless it is explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| read_only | boolean | Reports if the LUN allows only read access. |
| state | string | The state of the LUN. Normal states for a LUN are *online* and *offline*. Other states indicate errors. |

svm

The SVM in which the LUN is located.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

partner

The LUN partner that this LUN is bound to. If this LUN is a `vvol` class LUN, the partner is a `protocol_endpoint` class LUN.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the partner LUN. |
| uuid | string | The unique identifier of the partner LUN. |

bindings

A vVol binding with which the LUN is associated.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
| --- | --- | --- |
| id | integer | The ONTAP internal identifier assigned to the vVol binding. The bind identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>This property was included in early releases of the REST API for vVols and is maintained for backward compatibility. See the `secondary_id` property, which replaces `id`.<br><br>• example: 1<br>• readOnly: 1<br>• x-ntap-deprecated: 9.13.1<br>• Introduced in: 9.10<br>• x-nullable: true |
| partner | partner | The LUN partner that this LUN is bound to. If this LUN is a `vvol` class LUN, the partner is a `protocol_endpoint` class LUN. |
| secondary_id | string | The identifier assigned to the vVol binding, known as the secondary LUN ID. The identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>The format for a secondary LUN ID is 16 hexadecimal digits (zero-filled) followed by a lower case "h". |

vvol

A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.

See DELETE /protocols/san/vvol-bindings to learn more about deleting vVol bindings.

There is an added computational cost to retrieving property values for `vvol`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

| Name | Type | Description |
|---|---|---|
| bindings | array[bindings] | Bindings between the LUN, which must be of class `protocol_endpoint` or `vvol`, and LUNs of the opposite class.<br><br>A class `vvol` LUN must be bound to a class `protocol_endpoint` LUN in order to be accessed. Class `protocol_endpoint` and `vvol` LUNs allow many-to-many bindings. A LUN of one class is allowed to be bound to zero or more LUNs of the opposite class. The binding between any two specific LUNs is reference counted. When a binding is created that already exists, the binding count is incremented. When a binding is deleted, the binding count is decremented, but the LUNs remain bound if the resultant reference count is greater than zero. When the binding count reaches zero, the binding is destroyed.<br><br>The bindings array contains LUNs of the opposite class of the containing LUN object.<br><br>There is an added computational cost to retrieving property values for `vvol.bindings`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

| Name | Type | Description |
| --- | --- | --- |
| is_bound | boolean | Reports if the LUN is part of a VMware virtual volume (vVol) bind relationship. This is `true` if the LUN is of class `protocol_endpoint` or `vvol` and has one or more bindings to a LUN of the opposite class. This is false if the LUN is of class `regular` or unbound. |

lun

A LUN is the logical representation of storage in a storage area network (SAN).

A LUN must be mapped to an initiator group to grant access to the initiator group's initiators (client hosts). Initiators can then access the LUN and perform I/O over a Fibre Channel (FC) fabric using the FC Protocol or a TCP/IP network using iSCSI.

See the LUN object model to learn more about each of the properties supported by the LUN REST API.

A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.

LUN names are paths of the form "/vol/<volume>[/<qtree>]/<lun>" where the qtree name is optional.

A LUN can be created to a specified size using thin or thick provisioning. A LUN can then be renamed, resized, cloned, moved to a different volume and copied. LUNs support the assignment of a QoS policy for performance management or a QoS policy can be assigned to a volume containing one or more LUNs.

| Name | Type | Description |
| --- | --- | --- |
| _links | _links | |

| Name | Type | Description |
| --- | --- | --- |
| attributes | array[attributes] | An array of name/value pairs optionally stored with the LUN. Attributes are available to callers to persist small amounts of application-specific metadata. They are in no way interpreted by ONTAP.<br><br>Attribute names and values must be at least one byte and no more than 4091 bytes in length. The sum of the name and value lengths must be no more than 4092 bytes.<br><br>Valid in POST except when creating a LUN clone. A cloned can already have attributes from its source. You can add, modify, and delete the attributes of a LUN clone in separate requests after creation of the LUN.<br><br>Attributes can be added/modified/removed for an existing LUN using the /api/storage/luns/{lun.uuid}/attributes endpoint. For further information, see DOC /storage/luns/{lun.uuid}/attributes .<br><br>There is an added computational cost to retrieving property values for `attributes`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.<br><br>• Introduced in: 9.10<br>• readCreate: 1 |

| Name | Type | Description |
| --- | --- | --- |
| auto_delete | boolean | This property marks the LUN for auto deletion when the volume containing the LUN runs out of space. This is most commonly set on LUN clones.<br><br>When set to *true*, the LUN becomes eligible for automatic deletion when the volume runs out of space. Auto deletion only occurs when the volume containing the LUN is also configured for auto deletion and free space in the volume decreases below a particular threshold.<br><br>This property is optional in POST and PATCH. The default value for a new LUN is *false*.<br><br>There is an added computational cost to retrieving this property's value. It is not populated for a GET request unless it is explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| class | string | The class of LUN.<br><br>Optional in POST. |
| comment | string | A configurable comment available for use by the administrator. Valid in POST and PATCH. |
| consistency_group | consistency_group | The LUN's consistency group. This property is populated for LUNs that are members of a consistency group. If the LUN is a member of a child consistency group, the parent consistency group is reported. A LUN's consistency group is the consistency group of its containing volume. |

| Name | Type | Description |
| --- | --- | --- |
| copy | copy | This sub-object applies to LUN copy operations. A LUN can be copied with a POST request that supplies `copy.source` properties.<br><br>Copying a LUN is an asynchronous activity begun by a POST request that specifies the source of the copy in the `copy.source` properties. The data for the LUN is then asynchronously copied from the source to the destination. The time required to complete the copy depends on the size of the LUN and the load on the cluster. The `copy` sub-object is populated while a LUN copy is in progress and for two (2) minutes following completion of a copy.<br><br>While LUNs are being copied, the status of the LUN copy operations can be obtained using a GET of the source or destination LUN that requests the `copy` properties. If the LUN is the source LUN for one or more copy operations, the `copy.destinations` array is populated in GET. If the containing LUN is the destination LUN for a copy operation, the `copy.source` sub-object is populated in GET. The LUN copy operation can be further modified using a PATCH on the properties on the `copy.source` sub-object of the copy destination LUN.<br><br>There is an added computational cost to retrieving property values for `copy`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| create_time | string | The time the LUN was created. |

| Name | Type | Description |
| --- | --- | --- |
| enabled | boolean | The enabled state of the LUN. LUNs can be disabled to prevent access to the LUN. Certain error conditions also cause the LUN to become disabled. If the LUN is disabled, check the `status.state` property to determine if the LUN is administratively disabled (*offline*) or has become disabled as a result of an error. A LUN in an error condition can be brought online by setting the `enabled` property to *true* or brought administratively offline by setting the `enabled` property to *false*. Upon creation, a LUN is enabled by default. Valid in PATCH. |
| location | location | The location of the LUN within the ONTAP cluster. LUNs support rename and move between volumes. Valid in POST and PATCH.<br><br>• Introduced in: 9.6 |
| lun_maps | array[lun_maps] | The LUN maps with which the LUN is associated.<br><br>There is an added computational cost to retrieving property values for `lun_maps`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. These properties are supported for GET only. |
| metric | metric | Performance numbers, such as IOPS latency and throughput. |

| Name | Type | Description |
| --- | --- | --- |
| movement | movement | This sub-object applies to LUN movement between volumes. A LUN can be moved to a new volume with a PATCH request that changes either the volume portion of property `name`, `location.volume.uuid`, or `location.volume.name`. If the volume is changed using more than one of these properties, the supplied properties used must refer to the same volume.<br><br>Moving a LUN between volumes is an asynchronous activity begun by a PATCH request. The data for the LUN is then asynchronously copied from the source volume to the destination volume. The time required to complete the move depends on the size of the LUN and the load on the cluster. The `movement` sub-object is populated while a LUN movement is in progress and for two (2) minutes following completion of a movement.<br><br>While the LUN is being moved, the status of the LUN movement operation can be obtained using a GET for the LUN that requests the `movement` properties. The LUN movement operation can be further modified using a PATCH on the properties on the `movement` sub-object.<br><br>There is an added computational cost to retrieving property values for `movement`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

| Name | Type | Description |
| --- | --- | --- |
| name | string | The name of the LUN. Valid in POST and PATCH.<br><br>A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<lun>" where the qtree name is optional.<br><br>A PATCH that modifies the qtree and/or base name portion of the LUN path is considered a rename operation.<br><br>A PATCH that modifies the volume portion of the LUN path begins an asynchronous LUN movement operation. |
| os_type | string | The operating system type of the LUN.<br><br>Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone. |
| qos_policy | qos_policy | The QoS policy for the LUN. Both traditional and adaptive QoS policies are supported. If both property `qos_policy.uuid` and `qos_policy.name` are specified in the same request, they must refer to the same QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set the property `qos_policy.name` to an empty string ("") in a PATCH request. Valid in POST and PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| serial_number | string | The LUN serial number. The serial number is generated by ONTAP when the LUN is created.<br>• maxLength: 12<br>• minLength: 12<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |
| serial_number_hex | string | The LUN serial number encoded in hexadecimal format. The serial number is generated by ONTAP when the LUN is created.<br>• readOnly: 1<br>• Introduced in: 9.17<br>• x-nullable: true |
| space | space | The storage space related properties of the LUN. |
| statistics | statistics | These are raw performance numbers, such as IOPS latency and throughput. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster. |
| status | status | Status information about the LUN. |
| svm | svm | The SVM in which the LUN is located. |
| uuid | string | The unique identifier of the LUN. The UUID is generated by ONTAP when the LUN is created.<br>• example: 1cd8a442-86d1-11e0-ae1c-123478563412<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
|---|---|---|
| vvol | vvol | A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.<br><br>See DELETE /protocols/san/vvol-bindings to learn more about deleting vVol bindings.<br><br>There is an added computational cost to retrieving property values for `vvol`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

returned_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 a LUN

POST `/storage/luns`

**Introduced In:** 9.6

Creates a LUN.

### Required properties

- `svm.uuid` or `svm.name` - Existing SVM in which to create the LUN.
- `name`, `location.volume.name` or `location.volume.uuid` - Existing volume in which to create the LUN.
- `name` or `location.logical_unit` - Base name of the LUN.
- `os_type` - Operating system from which the LUN will be accessed. Required when creating a non-clone LUN and disallowed when creating a clone of an existing LUN. A clone's `os_type` is taken from the source LUN.
- `space.size` - Size of the LUN. Required when creating a non-clone LUN and disallowed when creating a clone of an existing LUN. A clone's size is taken from the source LUN.

### Recommended optional properties

- `qos_policy.name` or `qos_policy.uuid` - Existing traditional or adaptive QoS policy to be applied to the LUN. All LUNs should be managed by a QoS policy at the volume or LUN level.

### Default property values

If not specified in POST, the follow default property values are assigned.

- `auto_delete` - *false*

### Related ONTAP commands

- `lun create`
- `lun convert-from-namespace`
- `lun copy start`
- `volume file clone autodelete`
- `volume file clone create`

POST is asynchronous when creating a new LUN. It is synchronous when converting a namespace to a LUN via the `convert` property.

### Learn more

- DOC /storage/luns

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| return_timeout | integer | query | False | The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.<br>• Default value: 0<br>• Max value: 120<br>• Min value: 0 |
| return_records | boolean | query | False | The default is false. If set to true, the records are returned.<br>• Default value: |

**Request Body**

| Name | Type | Description |
| --- | --- | --- |
| attributes | array[attributes] | An array of name/value pairs optionally stored with the LUN. Attributes are available to callers to persist small amounts of application-specific metadata. They are in no way interpreted by ONTAP.<br><br>Attribute names and values must be at least one byte and no more than 4091 bytes in length. The sum of the name and value lengths must be no more than 4092 bytes.<br><br>Valid in POST except when creating a LUN clone. A cloned can already have attributes from its source. You can add, modify, and delete the attributes of a LUN clone in separate requests after creation of the LUN.<br><br>Attributes can be added/modified/removed for an existing LUN using the /api/storage/luns/{lun.uuid}/attributes endpoint. For further information, see DOC /storage/luns/{lun.uuid}/attributes .<br><br>There is an added computational cost to retrieving property values for `attributes`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.<br><br>• Introduced in: 9.10<br>• readCreate: 1 |

| Name | Type | Description |
| --- | --- | --- |
| auto_delete | boolean | This property marks the LUN for auto deletion when the volume containing the LUN runs out of space. This is most commonly set on LUN clones.<br><br>When set to *true*, the LUN becomes eligible for automatic deletion when the volume runs out of space. Auto deletion only occurs when the volume containing the LUN is also configured for auto deletion and free space in the volume decreases below a particular threshold.<br><br>This property is optional in POST and PATCH. The default value for a new LUN is *false*.<br><br>There is an added computational cost to retrieving this property's value. It is not populated for a GET request unless it is explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| class | string | The class of LUN.<br><br>Optional in POST. |

| Name | Type | Description |
| --- | --- | --- |
| clone | clone | This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: `auto_delete`, `qos_policy`, `space.guarantee.requested` and `space.scsi_thin_provisioning_support_enabled`.<br><br>When used in a PATCH, the patched LUN's data is over-written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: `class`, `auto_delete`, `lun_maps`, `serial_number`, `status.state`, and `uuid`.<br><br>Persistent reservations for the patched LUN are also preserved. |
| comment | string | A configurable comment available for use by the administrator. Valid in POST and PATCH. |
| consistency_group | consistency_group | The LUN's consistency group. This property is populated for LUNs that are members of a consistency group. If the LUN is a member of a child consistency group, the parent consistency group is reported. A LUN's consistency group is the consistency group of its containing volume. |
| convert | convert | This sub-object is used in POST to convert a valid in-place NVMe namespace to a LUN. Setting a property in this sub-object indicates that a conversion from the specified NVMe namespace to LUN is desired. |

| Name | Type | Description |
| --- | --- | --- |
| copy | copy | This sub-object applies to LUN copy operations. A LUN can be copied with a POST request that supplies `copy.source` properties.<br><br>Copying a LUN is an asynchronous activity begun by a POST request that specifies the source of the copy in the `copy.source` properties. The data for the LUN is then asynchronously copied from the source to the destination. The time required to complete the copy depends on the size of the LUN and the load on the cluster. The `copy` sub-object is populated while a LUN copy is in progress and for two (2) minutes following completion of a copy.<br><br>While LUNs are being copied, the status of the LUN copy operations can be obtained using a GET of the source or destination LUN that requests the `copy` properties. If the LUN is the source LUN for one or more copy operations, the `copy.destinations` array is populated in GET. If the containing LUN is the destination LUN for a copy operation, the `copy.source` sub-object is populated in GET. The LUN copy operation can be further modified using a PATCH on the properties on the `copy.source` sub-object of the copy destination LUN.<br><br>There is an added computational cost to retrieving property values for `copy`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| create_time | string | The time the LUN was created. |

| Name | Type | Description |
| --- | --- | --- |
| location | location | The location of the LUN within the ONTAP cluster. LUNs support rename and move between volumes. Valid in POST and PATCH.<br><br>• Introduced in: 9.6 |
| lun_maps | array[lun_maps] | The LUN maps with which the LUN is associated.<br><br>There is an added computational cost to retrieving property values for `lun_maps`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. These properties are supported for GET only. |

| Name | Type | Description |
| --- | --- | --- |
| movement | movement | This sub-object applies to LUN movement between volumes. A LUN can be moved to a new volume with a PATCH request that changes either the volume portion of property `name`, `location.volume.uuid`, or `location.volume.name`. If the volume is changed using more than one of these properties, the supplied properties used must refer to the same volume.<br><br>Moving a LUN between volumes is an asynchronous activity begun by a PATCH request. The data for the LUN is then asynchronously copied from the source volume to the destination volume. The time required to complete the move depends on the size of the LUN and the load on the cluster. The `movement` sub-object is populated while a LUN movement is in progress and for two (2) minutes following completion of a movement.<br><br>While the LUN is being moved, the status of the LUN movement operation can be obtained using a GET for the LUN that requests the `movement` properties. The LUN movement operation can be further modified using a PATCH on the properties on the `movement` sub-object.<br><br>There is an added computational cost to retrieving property values for `movement`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

| Name | Type | Description |
| --- | --- | --- |
| name | string | The name of the LUN. Valid in POST and PATCH.<br><br>A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<lun>" where the qtree name is optional.<br><br>A PATCH that modifies the qtree and/or base name portion of the LUN path is considered a rename operation.<br><br>A PATCH that modifies the volume portion of the LUN path begins an asynchronous LUN movement operation. |
| os_type | string | The operating system type of the LUN.<br><br>Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone. |
| provisioning_options | provisioning_options | Options that are applied to the operation. |
| qos_policy | qos_policy | The QoS policy for the LUN. Both traditional and adaptive QoS policies are supported. If both property `qos_policy.uuid` and `qos_policy.name` are specified in the same request, they must refer to the same QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set the property `qos_policy.name` to an empty string ("") in a PATCH request. Valid in POST and PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| serial_number | string | The LUN serial number. The serial number is generated by ONTAP when the LUN is created.<br>• maxLength: 12<br>• minLength: 12<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |
| serial_number_hex | string | The LUN serial number encoded in hexadecimal format. The serial number is generated by ONTAP when the LUN is created.<br>• readOnly: 1<br>• Introduced in: 9.17<br>• x-nullable: true |
| space | space | The storage space related properties of the LUN. |
| status | status | Status information about the LUN. |
| svm | svm | The SVM in which the LUN is located. |
| uuid | string | The unique identifier of the LUN. The UUID is generated by ONTAP when the LUN is created.<br>• example: 1cd8a442-86d1-11e0-ae1c-123478563412<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
| --- | --- | --- |
| vvol | vvol | A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.<br><br>See DELETE /protocols/san/vvol-bindings to learn more about deleting vVol bindings.<br><br>There is an added computational cost to retrieving property values for `vvol`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

**Example request**


```
{
  "attributes": [
    {
      "name": "name1",
      "value": "value1"
    }
  ],
  "class": "string",
  "clone": {
    "source": {
      "name": "/vol/volume1/lun1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "comment": "string",
  "consistency_group": {
    "name": "cg1",
    "uuid": "4abc2317-4332-9d37-93a0-20bd29c22df0"
  },
  "convert": {
    "namespace": {
      "name": "/vol/volume1/namespace1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "copy": {
    "destinations": [
      {
        "max_throughput": 0,
        "name": "/vol/vol1/lun1",
        "peer": {
          "name": "peer1",
          "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
        },
        "progress": {
          "elapsed": 0,
          "percent_complete": 0,
          "state": "string"
        },
        "uuid": "1bc327d5-4654-5284-a116-f182282240b4"
      }
    ],
    "source": {
      "name": "/vol/vol2/lun1",
```

```
      "peer": {
        "name": "peer1",
        "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
      },
      "uuid": "03c05019-40d9-3945-c767-dca4c3be5e90"
    }
  },
  "create_time": "2018-06-04 15:00:00 -0400",
  "enabled": true,
  "location": {
    "logical_unit": "lun1",
    "node": {
      "name": "node1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "qtree": {
      "id": 1,
      "name": "qt1"
    },
    "volume": {
      "name": "volume1",
      "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
    }
  },
  "lun_maps": [
    {
      "igroup": {
        "comment": "string",
        "igroups": [
          {
            "name": "igroup1",
            "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
          }
        ],
        "initiators": [
          {
            "comment": "my comment",
            "name": "iqn.1998-01.com.corp.iscsi:name1"
          }
        ],
        "name": "igroup1",
        "os_type": "string",
        "protocol": "string",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      }
    }
```

```
  ],
  "movement": {
    "paths": {
      "destination": "/vol/vol1/lun1",
      "source": "/vol/vol2/lun2"
    }
  },
  "name": "/vol/volume1/qtree1/lun1",
  "os_type": "string",
  "provisioning_options": {
    "qos_policy": {
      "name": "performance",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "snapshot_policy": {
      "name": "default",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "storage_service": {
      "name": "string"
    },
    "tiering": {
      "control": "string",
      "object_stores": [
        {
          "name": "string"
        }
      ],
      "policy": "string"
    }
  },
  "qos_policy": {
    "name": "qos1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "serial_number": "string",
  "serial_number_hex": "string",
  "space": {
    "efficiency_ratio": 2.5,
    "physical_used": 1073741824,
    "physical_used_by_snapshots": 1073741824,
    "size": 1073741824,
    "used": 0
  },
  "status": {
    "container_state": "string",
```

```
    "state": "online"
  },
  "svm": {
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
  "vvol": {
    "bindings": [
      {
        "id": 1,
        "partner": {
          "name": "/vol/vol1/lun1",
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        },
        "secondary_id": "0000D20000010000h"
      }
    ]
  }
}
```


### Response

```
Status: 202, Accepted
```

| Name | Type | Description |
|---|---|---|
| job | job_link | |

### Example response


```
{
  "job": {
    "uuid": "string"
  }
}
```


#### Headers

| Name | Description | Type |
|---|---|---|
| Location | Useful for tracking the resource location | string |

### Response

```
Status: 201, Created
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 917927 | The specified volume was not found. |
| 918236 | The specified `location.volume.uuid` and `location.volume.name` do not refer to the same volume. |
| 1260121 | Cloning a LUN to a volume different than the source volume is not supported. |
| 1260136 | The specified destination for a clone operation already exists as a LUN, namespace, or file. |
| 2621462 | The specified SVM does not exist. |
| 2621706 | The specified `svm.uuid` and `svm.name` do not refer to the same SVM. |
| 2621707 | No SVM was specified. Either `svm.name` or `svm.uuid` must be supplied. |
| 5242927 | The specified qtree was not found. |
| 5242950 | The specified `location.qtree.id` and `location.qtree.name` do not refer to the same qtree. |
| 5374121 | A LUN name can only contain characters A-Z, a-z, 0-9, "-", ".", "_", "{" and "}". |
| 5374123 | A negative size was provided for the LUN. |
| 5374124 | The specified size is too small for the LUN. |
| 5374125 | The specified size is too large for the LUN. |
| 5374127 | The specified LUN name is invalid. |
| 5374129 | LUNs cannot be created on a load sharing mirror volume. |
| 5374130 | An invalid size value was provided. |
| 5374237 | LUNs cannot be created on an SVM root volume. |
| 5374238 | LUNs cannot be created in snapshots. |

| Error Code | Description |
| --- | --- |
| 5374241 | A size value with invalid units was provided. |
| 5374242 | A LUN or NVMe namespace already exists at the specified path. |
| 5374352 | An invalid name was provided for the LUN. |
| 5374614 | The specified storage availability zone does not exist. |
| 5374707 | Creating a LUN in the specific volume is not allowed because the volume is reserved for an application. |
| 5374858 | The volume specified by `name` is not the same as that specified by `location.volume`. |
| 5374859 | No volume was specified for the LUN. |
| 5374860 | The qtree specified by `name` is not the same as that specified by `location.qtree`. |
| 5374861 | The LUN base name specified by `name` is not the same as that specified by `location.logical_unit`. |
| 5374862 | No LUN path base name was provided for the LUN. |
| 5374863 | An error occurred after successfully creating the LUN. Some properties were not set. |
| 5374874 | The specified `clone.source.uuid` and `clone.source.name` do not refer to the same LUN. |
| 5374875 | The specified `clone.source` was not found. |
| 5374876 | The specified `clone.source` was not found. |
| 5374883 | The property cannot be specified when creating a LUN clone. The `target` property of the error object identifies the property. |
| 5374884 | A property that is required when creating a new LUN that is not a LUN clone or LUN copy was not supplied. The `target` property of the error object identifies the property. |
| 5374886 | An error occurred after successfully creating the LUN preventing the retrieval of its properties. |
| 5374899 | The `clone.source.uuid` property is not supported when specifying a source LUN from a snapshot. |
| 5374928 | An incomplete attribute name/value pair was supplied. |
| 5374929 | The combined sizes of an attribute name and value are too large. |
| 5374932 | A name for an attribute was duplicated. |

| Error Code | Description |
| --- | --- |
| 5374942 | The property cannot be specified at the same time when creating a LUN as a clone. The `target` property of the error object identifies the other property given with clone. |
| 5374943 | The property cannot be specified at the same time when creating a LUN as a copy. The `target` property of the error object identifies the other property given with copy. |
| 5374944 | The property cannot be specified when converting an NVMe namespace into a LUN. The `target` property of the error object identifies the property. |
| 5375054 | The source LUN is required when requesting an SVM copy operation. |
| 5375059 | An unsuitable QoS policy was specified. |
| 5375061 | The specified `location.storage_availability_zone.uuid` and `location.storage_availability_zone.name` do not refer to the same storage availability zone. |
| 5376461 | The specified LUN name is invalid. |
| 5376462 | The specified LUN name is too long. |
| 5376463 | The snapshot portion of the specified LUN name is too long. |
| 5376469 | The property cannot be set during the LUN create operation on this platform. |
| 5376515 | The specified storage availability zone is not configured for provisioning on the specified SVM. |
| 5440509 | No suitable storage can be found for the specified requirements. |
| 5440688 | A LUN cannot be created with the same name as an existing LUN. |
| 5702832 | A LUN or namespace with the same name is already being created. LUN and namespace names must be unique within an SVM. |
| 7018877 | Maximum combined total (50) of file and LUN copy and move operations reached. When one or more of the operations has completed, try the command again. |
| 13565952 | The LUN clone request failed. |
| 26345672 | An SVM peer relationship was not found with the provided 'copy.source.peer.uuid'. |

| Error Code | Description |
| --- | --- |
| 26345673 | An SVM peer relationship between the SVM provided in 'copy.source.peer.name' and the SVM provided in either 'svm.name' or 'svm.uuid' was not found. |
| 26345674 | The SVM peer relationship provided in 'copy.source.peer.uuid' does not match a peer relationship between the local SVM provided in either 'svm.uuid' or 'svm.name' and the peer SVM provided in 'copy.source.peer.name'. |
| 26345675 | The destination SVM provided via 'svm.name' or 'svm.uuid' does not match the local SVM for the provided SVM peer relationship UUID in 'copy.source.peer.uuid'. |
| 72089755 | NVMe namespace with a block size of 4096 bytes cannot be converted to a LUN. |
| 72089756 | Namespace is currently mapped to subsystem. |
| 72089757 | NVMe namespace in a snapshot cannot be converted to a LUN. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

attributes

A name/value pair optionally stored with the LUN. Attributes are available to callers to persist small amounts of application-specific metadata. They are in no way interpreted by ONTAP.

Attribute names and values must be at least one byte and no more than 4091 bytes in length. The sum of the name and value lengths must be no more than 4092 bytes.

Optional in POST.

| Name | Type | Description |
|---|---|---|
| name | string | The attribute name. |
| value | string | The attribute value. |

source

The source LUN for a LUN clone operation. This can be specified using property `clone.source.uuid` or `clone.source.name`. If both properties are supplied, they must refer to the same LUN.

Valid in POST to create a new LUN as a clone of the source.

Valid in PATCH to overwrite an existing LUN's data as a clone of another.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the clone source LUN. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<namespace>" where the qtree name is optional.<br><br>Valid in POST and PATCH. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the clone source LUN. Valid in POST and PATCH. |

clone

This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: `auto_delete`, `qos_policy`, `space.guarantee.requested` and `space.scsi_thin_provisioning_support_enabled`.

When used in a PATCH, the patched LUN's data is over-written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: `class`, `auto_delete`, `lun_maps`, `serial_number`, `status.state`, and `uuid`.

Persistent reservations for the patched LUN are also preserved.

| Name | Type | Description |
|---|---|---|
| source | source | The source LUN for a LUN clone operation. This can be specified using property `clone.source.uuid` or `clone.source.name`. If both properties are supplied, they must refer to the same LUN.<br><br>Valid in POST to create a new LUN as a clone of the source.<br><br>Valid in PATCH to overwrite an existing LUN's data as a clone of another. |

consistency_group

The LUN's consistency group. This property is populated for LUNs that are members of a consistency group. If the LUN is a member of a child consistency group, the parent consistency group is reported. A LUN's consistency group is the consistency group of its containing volume.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the consistency group. |
| uuid | string | The unique identifier of the consistency group. |

namespace

The source namespace for convert operation. This can be specified using property `convert.namespace.uuid` or `convert.namespace.name`. If both properties are supplied, they must refer to the same NVMe namespace.

Valid in POST. A convert request from NVMe namespace to LUN cannot be combined with setting any other LUN properties. All other properties of the converted LUN come from the source NVMe namespace.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the source NVMe namespace. Valid in POST. An NVMe namespace is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>NVMe namespace names are paths of the form "/vol/<volume>[/<qtree>]/<NVMe namespace>" where the qtree name is optional. |
| uuid | string | The unique identifier of the source NVMe namespace. Valid in POST. |

convert

This sub-object is used in POST to convert a valid in-place NVMe namespace to a LUN. Setting a property in this sub-object indicates that a conversion from the specified NVMe namespace to LUN is desired.

| Name | Type | Description |
|---|---|---|
| namespace | namespace | The source namespace for convert operation. This can be specified using property `convert.namespace.uuid` or `convert.namespace.name`. If both properties are supplied, they must refer to the same NVMe namespace.<br><br>Valid in POST. A convert request from NVMe namespace to LUN cannot be combined with setting any other LUN properties. All other properties of the converted LUN come from the source NVMe namespace. |

peer

The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.

This is only populated by GET when the LUN copy is inter-SVM.

| Name | Type | Description |
|---|---|---|
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

error

Error information provided if the asynchronous LUN copy operation fails.

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

progress

Properties related to the progress of an active or recently completed LUN copy.

| Name | Type | Description |
|---|---|---|
| elapsed | integer | The amount of time that has elapsed since the start of the LUN copy, in seconds. |

| Name | Type | Description |
|---|---|---|
| percent_complete | integer | The percentage completed of the LUN copy. |
| state | string | The state of the LUN copy. |
| volume_snapshot_blocked | boolean | This property reports if volume snapshots are blocked by the LUN copy. This property can be polled to identify when volume snapshots can be resumed after beginning a LUN copy. |

destinations

A LUN copy operation in which the containing LUN is the source of the copy.

| Name | Type | Description |
|---|---|---|
| max_throughput | integer | The maximum data throughput, in bytes per second, that should be utilized in support of the LUN copy. See property `copy.source.max_throughput` for further details. |
| name | string | The fully qualified path of the LUN copy destination composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN. |
| peer | peer | The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.<br><br>This is only populated by GET when the LUN copy is inter-SVM. |
| progress | progress | Properties related to the progress of an active or recently completed LUN copy. |
| uuid | string | The unique identifier of the LUN copy destination. |

peer

The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.

Set this in POST to specify the source SVM for an inter-SVM LUN copy. Only populated by GET when the LUN copy is inter-SVM.

| Name | Type | Description |
|---|---|---|
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

progress

Properties related to the progress of an active or recently completed LUN copy.

| Name | Type | Description |
|---|---|---|
| elapsed | integer | The amount of time that has elapsed since the start of the LUN copy, in seconds. |
| percent_complete | integer | The percentage completed of the LUN copy. |
| volume_snapshot_blocked | boolean | This property reports if volume snapshots are blocked by the LUN copy. This property can be polled to identify when volume snapshots can be resumed after beginning a LUN copy. |

source

The source LUN of a LUN copy operation in which the containing LUN is the destination of the copy.

Valid in POST except when creating a LUN clone. A LUN copy request cannot be combined with setting any other LUN properties except the destination location. All other properties of the destination LUN come from the source LUN.

| Name | Type | Description |
| --- | --- | --- |
| max_throughput | integer | The maximum data throughput, in bytes per second, that should be utilized in support of the LUN copy. This property can be used to throttle a transfer and limit its impact on the performance of the source and destination nodes. The specified value will be rounded up to the nearest megabyte.<br><br>If this property is not specified in a POST that begins a LUN copy, throttling is not applied to the data transfer.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>Valid only in a POST that begins a LUN copy or a PATCH when a LUN copy is already in process.<br><br>• format: int64<br>• Introduced in: 9.10<br>• x-nullable: true |
| name | string | The fully qualified path of the LUN copy source composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN.<br><br>Set this property in POST to specify the source for a LUN copy operation. |

| Name | Type | Description |
|---|---|---|
| peer | peer | The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.<br><br>Set this in POST to specify the source SVM for an inter-SVM LUN copy. Only populated by GET when the LUN copy is inter-SVM. |
| uuid | string | The unique identifier of the LUN copy source.<br><br>Set this property in POST to specify the source for a LUN copy operation. |

copy

This sub-object applies to LUN copy operations. A LUN can be copied with a POST request that supplies `copy.source` properties.

Copying a LUN is an asynchronous activity begun by a POST request that specifies the source of the copy in the `copy.source` properties. The data for the LUN is then asynchronously copied from the source to the destination. The time required to complete the copy depends on the size of the LUN and the load on the cluster. The `copy` sub-object is populated while a LUN copy is in progress and for two (2) minutes following completion of a copy.

While LUNs are being copied, the status of the LUN copy operations can be obtained using a GET of the source or destination LUN that requests the `copy` properties. If the LUN is the source LUN for one or more copy operations, the `copy.destinations` array is populated in GET. If the containing LUN is the destination LUN for a copy operation, the `copy.source` sub-object is populated in GET. The LUN copy operation can be further modified using a PATCH on the properties on the `copy.source` sub-object of the copy destination LUN.

There is an added computational cost to retrieving property values for `copy`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

| Name | Type | Description |
|---|---|---|
| destinations | array[destinations] | An array of destination LUNs of LUN copy operations in which the containing LUN is the source of the copy. |

| Name | Type | Description |
|---|---|---|
| source | source | The source LUN of a LUN copy operation in which the containing LUN is the destination of the copy.<br><br>Valid in POST except when creating a LUN clone. A LUN copy request cannot be combined with setting any other LUN properties except the destination location. All other properties of the destination LUN come from the source LUN. |

node

The cluster node that hosts the LUN.

| Name | Type | Description |
|---|---|---|
| name | string | |
| uuid | string | |

qtree

The qtree in which the LUN is optionally located. Valid in POST and PATCH.

If properties `name` and `location.qtree.name` and/or `location.qtree.id` are specified in the same request, they must refer to the same qtree.

A PATCH that modifies the qtree of the LUN is considered a rename operation.

| Name | Type | Description |
|---|---|---|
| id | integer | The identifier for the qtree, unique within the qtree's volume. |
| name | string | The name of the qtree. |

volume

The volume in which the LUN is located. Valid in POST and PATCH.

If properties `name` and `location.volume.name` and/or `location.volume.uuid` are specified in the same request, they must refer to the same volume.

A PATCH that modifies the volume of the LUN begins an asynchronous LUN movement operation.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the volume. This field cannot be specified in a PATCH method. |
| uuid | string | Unique identifier for the volume. This corresponds to the instance-uuid that is exposed in the CLI and ONTAPI. It does not change due to a volume move.<br>• example: 028baa66-41bd-11e9-81d5-00a0986138f7<br>• Introduced in: 9.6<br>• x-nullable: true |

location

The location of the LUN within the ONTAP cluster. LUNs support rename and move between volumes. Valid in POST and PATCH.

| Name | Type | Description |
|---|---|---|
| logical_unit | string | The base name component of the LUN. Valid in POST and PATCH.<br><br>If properties `name` and `location.logical_unit` are specified in the same request, they must refer to the base name.<br><br>A PATCH that modifies the base name of the LUN is considered a rename operation. |
| node | node | The cluster node that hosts the LUN. |

| Name | Type | Description |
| --- | --- | --- |
| qtree | qtree | The qtree in which the LUN is optionally located. Valid in POST and PATCH.<br><br>If properties `name` and `location.qtree.name` and/or `location.qtree.id` are specified in the same request, they must refer to the same qtree.<br><br>A PATCH that modifies the qtree of the LUN is considered a rename operation. |
| volume | volume | The volume in which the LUN is located. Valid in POST and PATCH.<br><br>If properties `name` and `location.volume.name` and/or `location.volume.uuid` are specified in the same request, they must refer to the same volume.<br><br>A PATCH that modifies the volume of the LUN begins an asynchronous LUN movement operation. |

self_link

igroups

| Name | Type | Description |
| --- | --- | --- |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

initiators

The initiators that are members of the initiator group.

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. |
| name | string | Name of initiator that is a member of the initiator group. |

igroup

The initiator group to which the LUN is mapped.

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |
| igroups | array[igroups] | The existing initiator groups that are members of the group. Optional in POST.<br><br>This property is mutually exclusive with the *initiators* property during POST.<br><br>This array contains only the direct children of the initiator group. If the member initiator groups have further nested initiator groups, those are reported in the `igroups` property of the child initiator group.<br><br>Zero or more nested initiator groups can be supplied when the initiator group is created. The initiator group will act as if it contains the aggregation of all initiators in any nested initiator groups.<br><br>After creation, nested initiator groups can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/igroups` endpoint. See DELETE /protocols/san/igroups/{igroup.uuid}/igroups/{uuid} for more details. |

| Name | Type | Description |
|---|---|---|
| initiators | array[initiators] | The initiators that are members of the group. |
| name | string | The name of the initiator group. |
| os_type | string | The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. |
| protocol | string | The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. Optional in POST; if not supplied, this defaults to *mixed*.<br><br>The protocol of an initiator group cannot be changed after creation of the group. |
| uuid | string | The unique identifier of the initiator group. |

lun_maps

A LUN map is an association between a LUN and an initiator group.

When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.

| Name | Type | Description |
|---|---|---|
| igroup | igroup | The initiator group to which the LUN is mapped. |

| Name | Type | Description |
|---|---|---|
| logical_unit_number | integer | The logical unit number assigned to the LUN when mapped to the specified initiator group. The number is used to identify the LUN to initiators in the initiator group when communicating through the Fibre Channel Protocol or iSCSI. Optional in POST; if no value is provided, ONTAP assigns the lowest available value. This property is not supported when the *provisioning_options.count* property is 2 or more.<br>• Introduced in: 9.6<br>• readCreate: 1<br>• x-nullable: true |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

metric

Performance numbers, such as IOPS latency and throughput.

| Name | Type | Description |
| --- | --- | --- |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |
| status | string | Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

paths

The fully qualified LUN path names involved in the LUN movement.

| Name | Type | Description |
|---|---|---|
| destination | string | The fully qualified path of the LUN movement destination composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN. |
| source | string | The fully qualified path of the LUN movement source composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN. |

error

Error information provided if the asynchronous LUN movement operation fails.

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

progress

Properties related to the progress of an active or recently completed LUN movement.

| Name | Type | Description |
|---|---|---|
| elapsed | integer | The amount of time that has elapsed since the start of the LUN movement, in seconds. |
| percent_complete | integer | The percentage completed of the LUN movement. |
| volume_snapshot_blocked | boolean | This property reports if volume snapshots are blocked by the LUN movement. This property can be polled to identify when volume snapshots can be resumed after beginning a LUN movement. |

movement

This sub-object applies to LUN movement between volumes. A LUN can be moved to a new volume with a PATCH request that changes either the volume portion of property `name`, `location.volume.uuid`, or `location.volume.name`. If the volume is changed using more than one of these properties, the supplied properties used must refer to the same volume.

Moving a LUN between volumes is an asynchronous activity begun by a PATCH request. The data for the LUN is then asynchronously copied from the source volume to the destination volume. The time required to complete the move depends on the size of the LUN and the load on the cluster. The `movement` sub-object is populated while a LUN movement is in progress and for two (2) minutes following completion of a movement.

While the LUN is being moved, the status of the LUN movement operation can be obtained using a GET for the LUN that requests the `movement` properties. The LUN movement operation can be further modified using a PATCH on the properties on the `movement` sub-object.

There is an added computational cost to retrieving property values for `movement`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

| Name | Type | Description |
|---|---|---|
| max_throughput | integer | The maximum data throughput, in bytes per second, that should be utilized in support of the LUN movement. This property can be used to throttle a transfer and limit its impact on the performance of the source and destination nodes. The specified value will be rounded up to the nearest megabyte.<br><br>If this property is not specified in a POST that begins a LUN movement, throttling is not applied to the data transfer.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>This property is valid only in a POST that begins a LUN movement or a PATCH when a LUN movement is already in process.<br><br>• format: int64<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
|---|---|---|
| paths | paths | The fully qualified LUN path names involved in the LUN movement. |

qos_policy

The QoS policy for the volume provisioned to host the LUN. This property is only supported when the request provisions a new volume. This property is mutually exclusive with the LUN granular `qos_policy`. If no `qos_policy` is provided at LUN or volume granularity, a volume granular policy is set based on the `storage_service.name`, which defaults to the most performant service available.

| Name | Type | Description |
|---|---|---|
| name | string | The QoS policy group name. This is mutually exclusive with UUID and other QoS attributes during POST and PATCH. |
| uuid | string | The QoS policy group UUID. This is mutually exclusive with name and other QoS attributes during POST and PATCH. |

snapshot_policy_reference

This is a reference to the snapshot policy.

| Name | Type | Description |
|---|---|---|
| name | string | |
| uuid | string | |

storage_service

Determines the placement of the LUN based on the value specified. This property is only supported for regular and vvol LUNs. Valid in POST.

| Name | Type | Description |
|---|---|---|
| name | string | Storage service name. If not specified, the default value is the most performant for the platform. |

object_stores

| Name | Type | Description |
| --- | --- | --- |
| name | string | The name of the object store to use. Used for placement. |

tiering

The tiering placement and policy definitions for the volume provisioned to host the LUN. This property is only supported when the request provisions a new volume.

| Name | Type | Description |
| --- | --- | --- |
| control | string | Storage tiering placement rules for the object. |
| object_stores | array[object_stores] | Object stores to use. Used for placement. |

| Name | Type | Description |
| --- | --- | --- |
| policy | string | Policy that determines whether the user data blocks of a volume in a FabricPool will be tiered to the cloud store when they become cold.<br><br>FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of a volume block increases if it is accessed frequently and decreases when it is not. Valid in POST or PATCH.<br><br>all ‐ Allows tiering of both snapshots and active file system user data to the cloud store as soon as possible by ignoring the temperature on the volume blocks.<br><br>auto ‐ Allows tiering of both snapshot and active file system user data to the cloud store<br><br>none ‐ Volume blocks are not be tiered to the cloud store.<br><br>snapshot_only ‐ Allows tiering of only the volume snapshots not associated with the active file system.<br><br>The default tiering policy is "snapshot-only" for a FlexVol volume and "none" for a FlexGroup volume. The default minimum cooling period for the "snapshot-only" tiering policy is 2 days and for the "auto" tiering policy it is 31 days. |

provisioning_options

Options that are applied to the operation.

| Name | Type | Description |
| --- | --- | --- |
| auto | boolean | If the volume specified in the request does not exist, automatically provision one of appropriate size. If the volume does exist, resize it to accommodate the new LUN.<br><br>This property is only supported on Unified ONTAP.<br><br>The following behavior is different from a traditional POST request:<br><br>• The operation is asynchronous.<br>• The `qos_policy` property is applied to the provisioned volume instead of the LUN. A default QoS policy is applied to the volume if one is not provided.<br>• The `provisioning_options.count` property is supported, provisioning *count* LUNs on the volume using the specified properties.<br>• The `lun_maps` property is supported. If the specified initiator group does not exist, it is created. The LUN is mapped to this initiator group. If an initiator group is provisioned in this way, it is deleted after it is no longer mapped to any LUNs.<br>• The `clone`, `copy`, and `convert` properties are not supported.<br>• When performing `records` based operations, specifying this property in the query applies to the entire operation. Specifying it for an individual record within the request applies to only that record.<br>• Many other `provisioning_options` properties are supported to control the placement of the LUN and the properties of the volume containing the LUN. |

| Name | Type | Description |
| --- | --- | --- |
| count | integer | The number of LUNs to provision with these properties. Only POST requests based on `space.size` are supported. When provided, the name is considered a prefix, and a suffix of the form _<*N*> is generated where N is the next available numeric index, starting with 1. |
| qos_policy | qos_policy | The QoS policy for the volume provisioned to host the LUN. This property is only supported when the request provisions a new volume. This property is mutually exclusive with the LUN granular `qos_policy`. If no `qos_policy` is provided at LUN or volume granularity, a volume granular policy is set based on the `storage_service.name`, which defaults to the most performant service available. |
| snapshot_policy | snapshot_policy_reference | This is a reference to the snapshot policy. |
| storage_service | storage_service | Determines the placement of the LUN based on the value specified. This property is only supported for regular and vvol LUNs. Valid in POST. |
| tiering | tiering | The tiering placement and policy definitions for the volume provisioned to host the LUN. This property is only supported when the request provisions a new volume. |

| Name | Type | Description |
| --- | --- | --- |
| use_mirrored_aggregates | boolean | Specifies whether mirrored aggregates are selected when provisioning the volume to host the LUN. Only mirrored aggregates are used if this parameter is set to *true* and only unmirrored aggregates are used if this parameter is set to *false*. The default value is *true* for a MetroCluster configuration and is *false* for a non-MetroCluster configuration. |

qos_policy

The QoS policy for the LUN. Both traditional and adaptive QoS policies are supported. If both property `qos_policy.uuid` and `qos_policy.name` are specified in the same request, they must refer to the same QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set the property `qos_policy.name` to an empty string ("") in a PATCH request. Valid in POST and PATCH.

| Name | Type | Description |
| --- | --- | --- |
| name | string | The name of the QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set this property to an empty string ("") in a PATCH request. Valid in POST and PATCH. |
| uuid | string | The unique identifier of the QoS policy. Valid in POST and PATCH. |

guarantee

Properties that request and report the space guarantee for the LUN.

| Name | Type | Description |
| --- | --- | --- |
| requested | boolean | The requested space reservation policy for the LUN. If *true*, a space reservation is requested for the LUN; if *false*, the LUN is thin provisioned. Guaranteeing a space reservation request for a LUN requires that the volume in which the LUN resides is also space reserved and that the fractional reserve for the volume is 100%. Valid in POST and PATCH. This property is caller settable as described above. |
| reserved | boolean | Reports if the LUN is space guaranteed.<br><br>If *true*, a space guarantee is requested and the containing volume and aggregate support the request. If *false*, a space guarantee is not requested or a space guarantee is requested and either the containing volume or aggregate do not support the request. |

space

The storage space related properties of the LUN.

| Name | Type | Description |
| --- | --- | --- |
| efficiency_ratio | number | The storage efficiency ratio of the LUN without snapshots. (Logical Used / Used) This property is not available on the LUN object in the REST API and is not reported for GET requests. See the containing volume object for this information. |
| guarantee | guarantee | Properties that request and report the space guarantee for the LUN. |

| Name | Type | Description |
| --- | --- | --- |
| physical_used | integer | The number of bytes consumed on the disk by the LUN, excluding snapshots. This property is not available on the LUN object in the REST API and is not reported for GET requests. See the containing volume object for this information. |
| physical_used_by_snapshots | integer | The number of bytes consumed on the disk by the LUN's snapshots. This property is not available on the LUN object in the REST API and is not reported for GET requests. See the containing volume object for this information. |

| Name | Type | Description |
| --- | --- | --- |
| scsi_thin_provisioning_support_enabled | boolean | To leverage the benefits of SCSI thin provisioning, it must be supported by your host. SCSI thin provisioning uses the Logical Block Provisioning feature as defined in the SCSI SBC-3 standard. Only hosts that support this standard can use SCSI thin provisioning in ONTAP.<br><br>When you disable SCSI thin provisioning support in ONTAP, you turn off the following SCSI thin provisioning features:<br><br>• Unmapping and reporting space usage for space reclamation<br>• Reporting resource exhaustion errors<br><br>The value of this property is not propagated to the destination when a LUN is cloned as a new LUN or copied; it is reset to false. The value of this property is maintained from the destination LUN when a LUN is overwritten as a clone.<br><br>Valid in POST and PATCH.<br><br>• Default value: 1<br>• Introduced in: 9.10<br>• x-nullable: true |

| Name | Type | Description |
| --- | --- | --- |
| size | integer | The total provisioned size of the LUN. The LUN size can be increased but not decreased using the REST interface.<br><br>The maximum and minimum sizes listed here are the absolute maximum and absolute minimum sizes, in bytes. The actual minimum and maximum sizes vary depending on the ONTAP version, ONTAP platform and the available space in the containing volume and aggregate.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>• example: 1073741824<br>• format: int64<br>• Max value: 140737488355328<br>• Min value: 4096<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
|---|---|---|
| used | integer | The amount of space consumed by the main data stream of the LUN.<br><br>This value is the total space consumed in the volume by the LUN, including filesystem overhead, but excluding prefix and suffix streams. Due to internal filesystem overhead and the many ways SAN filesystems and applications utilize blocks within a LUN, this value does not necessarily reflect actual consumption/availability from the perspective of the filesystem or application. Without specific knowledge of how the LUN blocks are utilized outside of ONTAP, this property should not be used as an indicator for an out-of-space condition.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>• format: int64<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

iops_raw

The number of I/O operations observed at the storage object. This can be used along with delta time to calculate the rate of I/O operations per unit of time.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency_raw

The raw latency in microseconds observed at the storage object. This can be divided by the raw IOPS value to calculate the average latency per I/O operation.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput_raw

Throughput bytes observed at the storage object. This can be used along with delta time to calculate the rate of throughput bytes per unit of time.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |

| Name | Type | Description |
|---|---|---|
| write | integer | Performance metric for write I/O operations. |

statistics

These are raw performance numbers, such as IOPS latency and throughput. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

| Name | Type | Description |
|---|---|---|
| iops_raw | iops_raw | The number of I/O operations observed at the storage object. This can be used along with delta time to calculate the rate of I/O operations per unit of time. |
| latency_raw | latency_raw | The raw latency in microseconds observed at the storage object. This can be divided by the raw IOPS value to calculate the average latency per I/O operation. |
| status | string | Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |

| Name | Type | Description |
|---|---|---|
| throughput_raw | throughput_raw | Throughput bytes observed at the storage object. This can be used along with delta time to calculate the rate of throughput bytes per unit of time. |
| timestamp | string | The timestamp of the performance data. |

status

Status information about the LUN.

| Name | Type | Description |
|---|---|---|
| container_state | string | The state of the volume and aggregate that contain the LUN. LUNs are only available when their containers are available. |
| mapped | boolean | Reports if the LUN is mapped to one or more initiator groups.<br><br>There is an added computational cost to retrieving this property's value. It is not populated for a GET request unless it is explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| read_only | boolean | Reports if the LUN allows only read access. |
| state | string | The state of the LUN. Normal states for a LUN are *online* and *offline*. Other states indicate errors. |

svm

The SVM in which the LUN is located.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

partner

The LUN partner that this LUN is bound to. If this LUN is a `vvol` class LUN, the partner is a `protocol_endpoint` class LUN.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the partner LUN. |
| uuid | string | The unique identifier of the partner LUN. |

bindings

A vVol binding with which the LUN is associated.

| Name | Type | Description |
|---|---|---|
| id | integer | The ONTAP internal identifier assigned to the vVol binding. The bind identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>This property was included in early releases of the REST API for vVols and is maintained for backward compatibility. See the `secondary_id` property, which replaces `id`.<br><br>• example: 1<br>• readOnly: 1<br>• x-ntap-deprecated: 9.13.1<br>• Introduced in: 9.10<br>• x-nullable: true |
| partner | partner | The LUN partner that this LUN is bound to. If this LUN is a `vvol` class LUN, the partner is a `protocol_endpoint` class LUN. |

| Name | Type | Description |
| --- | --- | --- |
| secondary_id | string | The identifier assigned to the vVol binding, known as the secondary LUN ID. The identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>The format for a secondary LUN ID is 16 hexadecimal digits (zero-filled) followed by a lower case "h". |

vvol

A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.

See DELETE /protocols/san/vvol-bindings to learn more about deleting vVol bindings.

There is an added computational cost to retrieving property values for `vvol`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

| Name | Type | Description |
| --- | --- | --- |
| bindings | array[bindings] | Bindings between the LUN, which must be of class `protocol_endpoint` or `vvol`, and LUNs of the opposite class.<br><br>A class `vvol` LUN must be bound to a class `protocol_endpoint` LUN in order to be accessed. Class `protocol_endpoint` and `vvol` LUNs allow many-to-many bindings. A LUN of one class is allowed to be bound to zero or more LUNs of the opposite class. The binding between any two specific LUNs is reference counted. When a binding is created that already exists, the binding count is incremented. When a binding is deleted, the binding count is decremented, but the LUNs remain bound if the resultant reference count is greater than zero. When the binding count reaches zero, the binding is destroyed.<br><br>The bindings array contains LUNs of the opposite class of the containing LUN object.<br><br>There is an added computational cost to retrieving property values for `vvol.bindings`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| is_bound | boolean | Reports if the LUN is part of a VMware virtual volume (vVol) bind relationship. This is `true` if the LUN is of class `protocol_endpoint` or `vvol` and has one or more bindings to a LUN of the opposite class. This is false if the LUN is of class `regular` or unbound. |

lun

A LUN is the logical representation of storage in a storage area network (SAN).

A LUN must be mapped to an initiator group to grant access to the initiator group's initiators (client hosts). Initiators can then access the LUN and perform I/O over a Fibre Channel (FC) fabric using the FC Protocol or a TCP/IP network using iSCSI.

See the LUN object model to learn more about each of the properties supported by the LUN REST API.

A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.

LUN names are paths of the form "/vol/<volume>[/<qtree>]/<lun>" where the qtree name is optional.

A LUN can be created to a specified size using thin or thick provisioning. A LUN can then be renamed, resized, cloned, moved to a different volume and copied. LUNs support the assignment of a QoS policy for performance management or a QoS policy can be assigned to a volume containing one or more LUNs.

| Name | Type | Description |
| --- | --- | --- |
| attributes | array[attributes] | An array of name/value pairs optionally stored with the LUN. Attributes are available to callers to persist small amounts of application-specific metadata. They are in no way interpreted by ONTAP.<br><br>Attribute names and values must be at least one byte and no more than 4091 bytes in length. The sum of the name and value lengths must be no more than 4092 bytes.<br><br>Valid in POST except when creating a LUN clone. A cloned can already have attributes from its source. You can add, modify, and delete the attributes of a LUN clone in separate requests after creation of the LUN.<br><br>Attributes can be added/modified/removed for an existing LUN using the /api/storage/luns/{lun.uuid}/attributes endpoint. For further information, see DOC /storage/luns/{lun.uuid}/attributes .<br><br>There is an added computational cost to retrieving property values for `attributes`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.<br><br>• Introduced in: 9.10<br>• readCreate: 1 |

| Name | Type | Description |
| --- | --- | --- |
| auto_delete | boolean | This property marks the LUN for auto deletion when the volume containing the LUN runs out of space. This is most commonly set on LUN clones.<br><br>When set to *true*, the LUN becomes eligible for automatic deletion when the volume runs out of space. Auto deletion only occurs when the volume containing the LUN is also configured for auto deletion and free space in the volume decreases below a particular threshold.<br><br>This property is optional in POST and PATCH. The default value for a new LUN is *false*.<br><br>There is an added computational cost to retrieving this property's value. It is not populated for a GET request unless it is explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| class | string | The class of LUN.<br><br>Optional in POST. |

| Name | Type | Description |
| --- | --- | --- |
| clone | clone | This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: `auto_delete`, `qos_policy`, `space.guarantee.requested` and `space.scsi_thin_provisioning_support_enabled`.<br><br>When used in a PATCH, the patched LUN's data is over-written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: `class`, `auto_delete`, `lun_maps`, `serial_number`, `status.state`, and `uuid`.<br><br>Persistent reservations for the patched LUN are also preserved. |
| comment | string | A configurable comment available for use by the administrator. Valid in POST and PATCH. |
| consistency_group | consistency_group | The LUN's consistency group. This property is populated for LUNs that are members of a consistency group. If the LUN is a member of a child consistency group, the parent consistency group is reported. A LUN's consistency group is the consistency group of its containing volume. |

| Name | Type | Description |
| --- | --- | --- |
| convert | convert | This sub-object is used in POST to convert a valid in-place NVMe namespace to a LUN. Setting a property in this sub-object indicates that a conversion from the specified NVMe namespace to LUN is desired. |

| Name | Type | Description |
| --- | --- | --- |
| copy | copy | This sub-object applies to LUN copy operations. A LUN can be copied with a POST request that supplies `copy.source` properties.<br><br>Copying a LUN is an asynchronous activity begun by a POST request that specifies the source of the copy in the `copy.source` properties. The data for the LUN is then asynchronously copied from the source to the destination. The time required to complete the copy depends on the size of the LUN and the load on the cluster. The `copy` sub-object is populated while a LUN copy is in progress and for two (2) minutes following completion of a copy.<br><br>While LUNs are being copied, the status of the LUN copy operations can be obtained using a GET of the source or destination LUN that requests the `copy` properties. If the LUN is the source LUN for one or more copy operations, the `copy.destinations` array is populated in GET. If the containing LUN is the destination LUN for a copy operation, the `copy.source` sub-object is populated in GET. The LUN copy operation can be further modified using a PATCH on the properties on the `copy.source` sub-object of the copy destination LUN.<br><br>There is an added computational cost to retrieving property values for `copy`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| create_time | string | The time the LUN was created. |

| Name | Type | Description |
| --- | --- | --- |
| location | location | The location of the LUN within the ONTAP cluster. LUNs support rename and move between volumes. Valid in POST and PATCH.<br><br>• Introduced in: 9.6 |
| lun_maps | array[lun_maps] | The LUN maps with which the LUN is associated.<br><br>There is an added computational cost to retrieving property values for `lun_maps`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. These properties are supported for GET only. |

| Name | Type | Description |
| --- | --- | --- |
| movement | movement | This sub-object applies to LUN movement between volumes. A LUN can be moved to a new volume with a PATCH request that changes either the volume portion of property `name`, `location.volume.uuid`, or `location.volume.name`. If the volume is changed using more than one of these properties, the supplied properties used must refer to the same volume.<br><br>Moving a LUN between volumes is an asynchronous activity begun by a PATCH request. The data for the LUN is then asynchronously copied from the source volume to the destination volume. The time required to complete the move depends on the size of the LUN and the load on the cluster. The `movement` sub-object is populated while a LUN movement is in progress and for two (2) minutes following completion of a movement.<br><br>While the LUN is being moved, the status of the LUN movement operation can be obtained using a GET for the LUN that requests the `movement` properties. The LUN movement operation can be further modified using a PATCH on the properties on the `movement` sub-object.<br><br>There is an added computational cost to retrieving property values for `movement`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

| Name | Type | Description |
| --- | --- | --- |
| name | string | The name of the LUN. Valid in POST and PATCH.<br><br>A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<lun>" where the qtree name is optional.<br><br>A PATCH that modifies the qtree and/or base name portion of the LUN path is considered a rename operation.<br><br>A PATCH that modifies the volume portion of the LUN path begins an asynchronous LUN movement operation. |
| os_type | string | The operating system type of the LUN.<br><br>Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone. |
| provisioning_options | provisioning_options | Options that are applied to the operation. |
| qos_policy | qos_policy | The QoS policy for the LUN. Both traditional and adaptive QoS policies are supported. If both property `qos_policy.uuid` and `qos_policy.name` are specified in the same request, they must refer to the same QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set the property `qos_policy.name` to an empty string ("") in a PATCH request. Valid in POST and PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| serial_number | string | The LUN serial number. The serial number is generated by ONTAP when the LUN is created.<br>• maxLength: 12<br>• minLength: 12<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |
| serial_number_hex | string | The LUN serial number encoded in hexadecimal format. The serial number is generated by ONTAP when the LUN is created.<br>• readOnly: 1<br>• Introduced in: 9.17<br>• x-nullable: true |
| space | space | The storage space related properties of the LUN. |
| status | status | Status information about the LUN. |
| svm | svm | The SVM in which the LUN is located. |
| uuid | string | The unique identifier of the LUN. The UUID is generated by ONTAP when the LUN is created.<br>• example: 1cd8a442-86d1-11e0-ae1c-123478563412<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
| --- | --- | --- |
| vvol | vvol | A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.<br><br>See DELETE /protocols/san/vvol-bindings to learn more about deleting vVol bindings.<br><br>There is an added computational cost to retrieving property values for `vvol`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

job_link

| Name | Type | Description |
| --- | --- | --- |
| uuid | string | The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation. |

returned_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. |

## Delete a LUN

DELETE `/storage/luns/{uuid}`

**Introduced In:** 9.6

Deletes a LUN.

### Related ONTAP commands

- `lun copy cancel`
- `lun delete`

### Learn more

- DOC /storage/luns

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| uuid | string | path | True | The unique identifier of the LUN to retrieve. |
| allow_delete_while_mapped | boolean | query | False | Allows deletion of a mapped LUN.<br><br>A mapped LUN might be in use. Deleting a mapped LUN also deletes the LUN map and makes the data no longer available. This might cause a disruption in the availability of data.<br><br>**This parameter should be used with caution.**<br><br>• Default value: |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| return_timeout | integer | query | False | The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.<br>• Default value: 0<br>• Max value: 120<br>• Min value: 0 |

**Response**

```
Status: 200, Ok
```

**Response**

```
Status: 202, Accepted
```

**Error**

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 1254197 | The LUN is mapped and cannot be deleted without specifying the `allow_delete_while_mapped` query parameter. |
| 5374705 | Deleting the LUN is not allowed because it is part of an application. |
| 5374865 | The LUN's aggregate is offline. The aggregate must be online to modify or remove the LUN. |
| 5374866 | The LUN's volume is offline. The volume must be online to modify or remove the LUN. |
| 5374875 | The specified LUN was not found. |
| 5374876 | The specified LUN was not found. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 |

returned_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 LUN properties or data

GET `/storage/luns/{uuid}`

**Introduced In:** 9.6

Retrieves a LUN's properties or a LUN's data.

LUN data read requests are distinguished by the header entry `Accept: multipart/form-data`. When this header entry is provided, query parameters `data.offset` and `data.size` are required and used to specify the portion of the LUN's data to read; no other query parameters are allowed. Reads are limited to one megabyte (1MB) per request. Data is returned as `multipart/form-data` content with exactly one form entry containing the data. The form entry has content type `application/octet-stream`.

### 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.

- `attributes.*`
- `auto_delete`
- `copy.*`
- `lun_maps.*`

- `movement.*`
- `space.physical_used`
- `space.physical_used_by_snapshots`
- `space.efficiency_ratio`
- `statistics.*`
- `vvol.bindings.*`
- `metric.*`

### Related ONTAP commands

- `lun bind show`
- `lun copy show`
- `lun mapping show`
- `lun move show`
- `lun show`
- `volume file clone show-autodelete`

### Learn more

- DOC /storage/luns

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| uuid | string | path | True | The unique identifier of the LUN to retrieve. |

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| data.offset | integer | query | False | The offset, in bytes, at which to begin reading LUN data.<br><br>LUN data read requests are distinguished by the header entry `Accept: multipart/form-data`. When this header entry is provided, query parameters `data.offset` and `data.size` are required and used to specify the portion of the LUN's data to read; no other query parameters are allowed. Reads are limited to one megabyte (1MB) per request. Data is returned as `multipart/form-data` content with exactly one form entry containing the data. The form entry has content type `application/octet-stream`.<br><br>• format: int64<br>• Min value: 0<br>• Introduced in: 9.11 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| data.size | integer | query | False | The size, in bytes, of LUN data to read.<br><br>LUN data read requests are distinguished by the header entry `Accept: multipart/form-data`. When this header entry is provided, query parameters `data.offset` and `data.size` are required and used to specify the portion of the LUN's data to read; no other query parameters are allowed. Reads are limited to one megabyte (1MB) per request. Data is returned as `multipart/form-data` content with exactly one form entry containing the data. The form entry has content type `application/octet-stream`.<br><br>• format: int64<br>• Min value: 1<br>• Introduced in: 9.11<br>• Max value: 1048576 |
| fields | array[string] | query | False | Specify the fields to return. |

**Response**

```
Status: 200, Ok
```

| Name | Type | Description |
| --- | --- | --- |
| _links | _links | |
| attributes | array[attributes] | An array of name/value pairs optionally stored with the LUN. Attributes are available to callers to persist small amounts of application-specific metadata. They are in no way interpreted by ONTAP.<br><br>Attribute names and values must be at least one byte and no more than 4091 bytes in length. The sum of the name and value lengths must be no more than 4092 bytes.<br><br>Valid in POST except when creating a LUN clone. A cloned can already have attributes from its source. You can add, modify, and delete the attributes of a LUN clone in separate requests after creation of the LUN.<br><br>Attributes can be added/modified/removed for an existing LUN using the /api/storage/luns/{lun.uuid}/attributes endpoint. For further information, see DOC /storage/luns/{lun.uuid}/attributes .<br><br>There is an added computational cost to retrieving property values for `attributes`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.<br><br>• Introduced in: 9.10<br>• readCreate: 1 |

| Name | Type | Description |
| --- | --- | --- |
| auto_delete | boolean | This property marks the LUN for auto deletion when the volume containing the LUN runs out of space. This is most commonly set on LUN clones.<br><br>When set to *true*, the LUN becomes eligible for automatic deletion when the volume runs out of space. Auto deletion only occurs when the volume containing the LUN is also configured for auto deletion and free space in the volume decreases below a particular threshold.<br><br>This property is optional in POST and PATCH. The default value for a new LUN is *false*.<br><br>There is an added computational cost to retrieving this property's value. It is not populated for a GET request unless it is explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| class | string | The class of LUN.<br><br>Optional in POST. |
| comment | string | A configurable comment available for use by the administrator. Valid in POST and PATCH. |
| consistency_group | consistency_group | The LUN's consistency group. This property is populated for LUNs that are members of a consistency group. If the LUN is a member of a child consistency group, the parent consistency group is reported. A LUN's consistency group is the consistency group of its containing volume. |

| Name | Type | Description |
| --- | --- | --- |
| copy | copy | This sub-object applies to LUN copy operations. A LUN can be copied with a POST request that supplies `copy.source` properties.<br><br>Copying a LUN is an asynchronous activity begun by a POST request that specifies the source of the copy in the `copy.source` properties. The data for the LUN is then asynchronously copied from the source to the destination. The time required to complete the copy depends on the size of the LUN and the load on the cluster. The `copy` sub-object is populated while a LUN copy is in progress and for two (2) minutes following completion of a copy.<br><br>While LUNs are being copied, the status of the LUN copy operations can be obtained using a GET of the source or destination LUN that requests the `copy` properties. If the LUN is the source LUN for one or more copy operations, the `copy.destinations` array is populated in GET. If the containing LUN is the destination LUN for a copy operation, the `copy.source` sub-object is populated in GET. The LUN copy operation can be further modified using a PATCH on the properties on the `copy.source` sub-object of the copy destination LUN.<br><br>There is an added computational cost to retrieving property values for `copy`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| create_time | string | The time the LUN was created. |

| Name | Type | Description |
| --- | --- | --- |
| enabled | boolean | The enabled state of the LUN. LUNs can be disabled to prevent access to the LUN. Certain error conditions also cause the LUN to become disabled. If the LUN is disabled, check the `status.state` property to determine if the LUN is administratively disabled (*offline*) or has become disabled as a result of an error. A LUN in an error condition can be brought online by setting the `enabled` property to *true* or brought administratively offline by setting the `enabled` property to *false*. Upon creation, a LUN is enabled by default. Valid in PATCH. |
| location | location | The location of the LUN within the ONTAP cluster. LUNs support rename and move between volumes. Valid in POST and PATCH.<br><br>• Introduced in: 9.6 |
| lun_maps | array[lun_maps] | The LUN maps with which the LUN is associated.<br><br>There is an added computational cost to retrieving property values for `lun_maps`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. These properties are supported for GET only. |
| metric | metric | Performance numbers, such as IOPS latency and throughput. |

| Name | Type | Description |
| --- | --- | --- |
| movement | movement | This sub-object applies to LUN movement between volumes. A LUN can be moved to a new volume with a PATCH request that changes either the volume portion of property `name`, `location.volume.uuid`, or `location.volume.name`. If the volume is changed using more than one of these properties, the supplied properties used must refer to the same volume.<br><br>Moving a LUN between volumes is an asynchronous activity begun by a PATCH request. The data for the LUN is then asynchronously copied from the source volume to the destination volume. The time required to complete the move depends on the size of the LUN and the load on the cluster. The `movement` sub-object is populated while a LUN movement is in progress and for two (2) minutes following completion of a movement.<br><br>While the LUN is being moved, the status of the LUN movement operation can be obtained using a GET for the LUN that requests the `movement` properties. The LUN movement operation can be further modified using a PATCH on the properties on the `movement` sub-object.<br><br>There is an added computational cost to retrieving property values for `movement`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

| Name | Type | Description |
| --- | --- | --- |
| name | string | The name of the LUN. Valid in POST and PATCH.<br><br>A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<lun>" where the qtree name is optional.<br><br>A PATCH that modifies the qtree and/or base name portion of the LUN path is considered a rename operation.<br><br>A PATCH that modifies the volume portion of the LUN path begins an asynchronous LUN movement operation. |
| os_type | string | The operating system type of the LUN.<br><br>Required in POST when creating a LUN that is not a clone of another. Disallowed in POST when creating a LUN clone. |
| qos_policy | qos_policy | The QoS policy for the LUN. Both traditional and adaptive QoS policies are supported. If both property `qos_policy.uuid` and `qos_policy.name` are specified in the same request, they must refer to the same QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set the property `qos_policy.name` to an empty string ("") in a PATCH request. Valid in POST and PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| serial_number | string | The LUN serial number. The serial number is generated by ONTAP when the LUN is created.<br>• maxLength: 12<br>• minLength: 12<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |
| serial_number_hex | string | The LUN serial number encoded in hexadecimal format. The serial number is generated by ONTAP when the LUN is created.<br>• readOnly: 1<br>• Introduced in: 9.17<br>• x-nullable: true |
| space | space | The storage space related properties of the LUN. |
| statistics | statistics | These are raw performance numbers, such as IOPS latency and throughput. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster. |
| status | status | Status information about the LUN. |
| svm | svm | The SVM in which the LUN is located. |
| uuid | string | The unique identifier of the LUN. The UUID is generated by ONTAP when the LUN is created.<br>• example: 1cd8a442-86d1-11e0-ae1c-123478563412<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
| --- | --- | --- |
| vvol | vvol | A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.<br><br>See DELETE /protocols/san/vvol-bindings to learn more about deleting vVol bindings.<br><br>There is an added computational cost to retrieving property values for `vvol`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

**Example response**


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "attributes": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "name1",
      "value": "value1"
    }
  ],
  "class": "string",
  "comment": "string",
  "consistency_group": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "cg1",
    "uuid": "4abc2317-4332-9d37-93a0-20bd29c22df0"
  },
  "copy": {
    "destinations": [
      {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "max_throughput": 0,
        "name": "/vol/vol1/lun1",
        "peer": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
```

```
        "name": "peer1",
        "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
      },
      "progress": {
        "elapsed": 0,
        "failure": {
          "arguments": [
            {
              "code": "string",
              "message": "string"
            }
          ],
          "code": "4",
          "message": "entry doesn't exist"
        },
        "percent_complete": 0,
        "state": "string"
      },
      "uuid": "1bc327d5-4654-5284-a116-f182282240b4"
    }
  ],
  "source": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "/vol/vol2/lun1",
    "peer": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "peer1",
      "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
    },
    "progress": {
      "elapsed": 0,
      "failure": {
        "arguments": [
          {
            "code": "string",
            "message": "string"
          }
        ],
```

```
          "code": "4",
          "message": "entry doesn't exist"
        },
        "percent_complete": 0,
        "state": "string"
      },
      "uuid": "03c05019-40d9-3945-c767-dca4c3be5e90"
    }
  },
  "create_time": "2018-06-04 15:00:00 -0400",
  "location": {
    "logical_unit": "lun1",
    "node": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "node1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "qtree": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "id": 1,
      "name": "qt1"
    },
    "volume": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "volume1",
      "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
    }
  },
  "lun_maps": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
```

```
      },
      "igroup": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "comment": "string",
        "igroups": [
          {
            "_links": {
              "self": {
                "href": "/api/resourcelink"
              }
            },
            "name": "igroup1",
            "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
          }
        ],
        "initiators": [
          {
            "comment": "my comment",
            "name": "iqn.1998-01.com.corp.iscsi:name1"
          }
        ],
        "name": "igroup1",
        "os_type": "string",
        "protocol": "string",
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      }
    }
  ],
  "metric": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "duration": "PT15S",
    "iops": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency": {
      "read": 200,
```

```
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "timestamp": "2017-01-25 06:20:13 -0500"
  },
  "movement": {
    "paths": {
      "destination": "/vol/vol1/lun1",
      "source": "/vol/vol2/lun2"
    },
    "progress": {
      "elapsed": 0,
      "failure": {
        "arguments": [
          {
            "code": "string",
            "message": "string"
          }
        ],
        "code": "4",
        "message": "entry doesn't exist"
      },
      "percent_complete": 0,
      "state": "string"
    }
  },
  "name": "/vol/volume1/qtree1/lun1",
  "os_type": "string",
  "qos_policy": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "qos1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "serial_number": "string",
  "serial_number_hex": "string",
  "space": {
```

```
    "efficiency_ratio": 2.5,
    "physical_used": 1073741824,
    "physical_used_by_snapshots": 1073741824,
    "size": 1073741824,
    "used": 0
  },
  "statistics": {
    "iops_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "timestamp": "2017-01-25 06:20:13 -0500"
  },
  "status": {
    "container_state": "string",
    "state": "online"
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
  "vvol": {
    "bindings": [
      {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
```

```
        },
        "id": 1,
        "partner": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "/vol/vol1/lun1",
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        },
        "secondary_id": "0000D20000010000h"
      }
    ]
  }
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374875 | The specified LUN was not found. |
| 5374876 | The specified LUN was not found. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

attributes

A name/value pair optionally stored with the LUN. Attributes are available to callers to persist small amounts of application-specific metadata. They are in no way interpreted by ONTAP.

Attribute names and values must be at least one byte and no more than 4091 bytes in length. The sum of the name and value lengths must be no more than 4092 bytes.

Optional in POST.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The attribute name. |
| value | string | The attribute value. |

source

The source LUN for a LUN clone operation. This can be specified using property `clone.source.uuid` or `clone.source.name`. If both properties are supplied, they must refer to the same LUN.

Valid in POST to create a new LUN as a clone of the source.

Valid in PATCH to overwrite an existing LUN's data as a clone of another.

clone

This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: `auto_delete`, `qos_policy`, `space.guarantee.requested` and `space.scsi_thin_provisioning_support_enabled`.

When used in a PATCH, the patched LUN's data is over-written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: `class`, `auto_delete`, `lun_maps`, `serial_number`, `status.state`, and `uuid`.

Persistent reservations for the patched LUN are also preserved.

consistency_group

The LUN's consistency group. This property is populated for LUNs that are members of a consistency group. If the LUN is a member of a child consistency group, the parent consistency group is reported. A LUN's consistency group is the consistency group of its containing volume.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the consistency group. |
| uuid | string | The unique identifier of the consistency group. |

namespace

The source namespace for convert operation. This can be specified using property `convert.namespace.uuid` or `convert.namespace.name`. If both properties are supplied, they must refer to the same NVMe namespace.

Valid in POST. A convert request from NVMe namespace to LUN cannot be combined with setting any other LUN properties. All other properties of the converted LUN come from the source NVMe namespace.

convert

This sub-object is used in POST to convert a valid in-place NVMe namespace to a LUN. Setting a property in this sub-object indicates that a conversion from the specified NVMe namespace to LUN is desired.

peer

The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.

This is only populated by GET when the LUN copy is inter-SVM.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

error

Error information provided if the asynchronous LUN copy operation fails.

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

progress

Properties related to the progress of an active or recently completed LUN copy.

| Name | Type | Description |
|---|---|---|
| elapsed | integer | The amount of time that has elapsed since the start of the LUN copy, in seconds. |
| failure | error | Error information provided if the asynchronous LUN copy operation fails. |
| percent_complete | integer | The percentage completed of the LUN copy. |
| state | string | The state of the LUN copy. |
| volume_snapshot_blocked | boolean | This property reports if volume snapshots are blocked by the LUN copy. This property can be polled to identify when volume snapshots can be resumed after beginning a LUN copy. |

destinations

A LUN copy operation in which the containing LUN is the source of the copy.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| max_throughput | integer | The maximum data throughput, in bytes per second, that should be utilized in support of the LUN copy. See property `copy.source.max_throughput` for further details. |
| name | string | The fully qualified path of the LUN copy destination composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN. |
| peer | peer | The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.<br><br>This is only populated by GET when the LUN copy is inter-SVM. |
| progress | progress | Properties related to the progress of an active or recently completed LUN copy. |
| uuid | string | The unique identifier of the LUN copy destination. |

peer

The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.

Set this in POST to specify the source SVM for an inter-SVM LUN copy. Only populated by GET when the LUN copy is inter-SVM.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

progress

Properties related to the progress of an active or recently completed LUN copy.

| Name | Type | Description |
|---|---|---|
| elapsed | integer | The amount of time that has elapsed since the start of the LUN copy, in seconds. |
| failure | error | Error information provided if the asynchronous LUN copy operation fails. |
| percent_complete | integer | The percentage completed of the LUN copy. |
| state | string | The state of the LUN copy.<br><br>Valid in PATCH when an LUN copy is active. Set to *paused* to pause a LUN copy. Set to *replicating* to resume a paused LUN copy. |
| volume_snapshot_blocked | boolean | This property reports if volume snapshots are blocked by the LUN copy. This property can be polled to identify when volume snapshots can be resumed after beginning a LUN copy. |

source

The source LUN of a LUN copy operation in which the containing LUN is the destination of the copy.

Valid in POST except when creating a LUN clone. A LUN copy request cannot be combined with setting any other LUN properties except the destination location. All other properties of the destination LUN come from the source LUN.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
| --- | --- | --- |
| max_throughput | integer | The maximum data throughput, in bytes per second, that should be utilized in support of the LUN copy. This property can be used to throttle a transfer and limit its impact on the performance of the source and destination nodes. The specified value will be rounded up to the nearest megabyte.<br><br>If this property is not specified in a POST that begins a LUN copy, throttling is not applied to the data transfer.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>Valid only in a POST that begins a LUN copy or a PATCH when a LUN copy is already in process.<br><br>• format: int64<br>• Introduced in: 9.10<br>• x-nullable: true |
| name | string | The fully qualified path of the LUN copy source composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN.<br><br>Set this property in POST to specify the source for a LUN copy operation. |

| Name | Type | Description |
| --- | --- | --- |
| peer | peer | The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.<br><br>Set this in POST to specify the source SVM for an inter-SVM LUN copy. Only populated by GET when the LUN copy is inter-SVM. |
| progress | progress | Properties related to the progress of an active or recently completed LUN copy. |
| uuid | string | The unique identifier of the LUN copy source.<br><br>Set this property in POST to specify the source for a LUN copy operation. |

copy

This sub-object applies to LUN copy operations. A LUN can be copied with a POST request that supplies `copy.source` properties.

Copying a LUN is an asynchronous activity begun by a POST request that specifies the source of the copy in the `copy.source` properties. The data for the LUN is then asynchronously copied from the source to the destination. The time required to complete the copy depends on the size of the LUN and the load on the cluster. The `copy` sub-object is populated while a LUN copy is in progress and for two (2) minutes following completion of a copy.

While LUNs are being copied, the status of the LUN copy operations can be obtained using a GET of the source or destination LUN that requests the `copy` properties. If the LUN is the source LUN for one or more copy operations, the `copy.destinations` array is populated in GET. If the containing LUN is the destination LUN for a copy operation, the `copy.source` sub-object is populated in GET. The LUN copy operation can be further modified using a PATCH on the properties on the `copy.source` sub-object of the copy destination LUN.

There is an added computational cost to retrieving property values for `copy`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

| Name | Type | Description |
|---|---|---|
| destinations | array[destinations] | An array of destination LUNs of LUN copy operations in which the containing LUN is the source of the copy. |
| source | source | The source LUN of a LUN copy operation in which the containing LUN is the destination of the copy.<br><br>Valid in POST except when creating a LUN clone. A LUN copy request cannot be combined with setting any other LUN properties except the destination location. All other properties of the destination LUN come from the source LUN. |

node

The cluster node that hosts the LUN.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | |
| uuid | string | |

qtree

The qtree in which the LUN is optionally located. Valid in POST and PATCH.

If properties `name` and `location.qtree.name` and/or `location.qtree.id` are specified in the same request, they must refer to the same qtree.

A PATCH that modifies the qtree of the LUN is considered a rename operation.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| id | integer | The identifier for the qtree, unique within the qtree's volume. |
| name | string | The name of the qtree. |

volume

The volume in which the LUN is located. Valid in POST and PATCH.

If properties `name` and `location.volume.name` and/or `location.volume.uuid` are specified in the same request, they must refer to the same volume.

A PATCH that modifies the volume of the LUN begins an asynchronous LUN movement operation.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the volume. This field cannot be specified in a PATCH method. |
| uuid | string | Unique identifier for the volume. This corresponds to the instance-uuid that is exposed in the CLI and ONTAPI. It does not change due to a volume move.<br>• example: 028baa66-41bd-11e9-81d5-00a0986138f7<br>• Introduced in: 9.6<br>• x-nullable: true |

location

The location of the LUN within the ONTAP cluster. LUNs support rename and move between volumes. Valid in POST and PATCH.

| Name | Type | Description |
|---|---|---|
| logical_unit | string | The base name component of the LUN. Valid in POST and PATCH.<br><br>If properties `name` and `location.logical_unit` are specified in the same request, they must refer to the base name.<br><br>A PATCH that modifies the base name of the LUN is considered a rename operation. |
| node | node | The cluster node that hosts the LUN. |

| Name | Type | Description |
| --- | --- | --- |
| qtree | qtree | The qtree in which the LUN is optionally located. Valid in POST and PATCH.<br><br>If properties `name` and `location.qtree.name` and/or `location.qtree.id` are specified in the same request, they must refer to the same qtree.<br><br>A PATCH that modifies the qtree of the LUN is considered a rename operation. |
| volume | volume | The volume in which the LUN is located. Valid in POST and PATCH.<br><br>If properties `name` and `location.volume.name` and/or `location.volume.uuid` are specified in the same request, they must refer to the same volume.<br><br>A PATCH that modifies the volume of the LUN begins an asynchronous LUN movement operation. |

self_link

| Name | Type | Description |
| --- | --- | --- |
| self | href | |

igroups

| Name | Type | Description |
| --- | --- | --- |
| _links | self_link | |
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

initiators

The initiators that are members of the initiator group.

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. |
| name | string | Name of initiator that is a member of the initiator group. |

igroup

The initiator group to which the LUN is mapped.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| comment | string | A comment available for use by the administrator. Valid in POST and PATCH. |

| Name | Type | Description |
| --- | --- | --- |
| igroups | array[igroups] | The existing initiator groups that are members of the group. Optional in POST.<br><br>This property is mutually exclusive with the *initiators* property during POST.<br><br>This array contains only the direct children of the initiator group. If the member initiator groups have further nested initiator groups, those are reported in the `igroups` property of the child initiator group.<br><br>Zero or more nested initiator groups can be supplied when the initiator group is created. The initiator group will act as if it contains the aggregation of all initiators in any nested initiator groups.<br><br>After creation, nested initiator groups can be added or removed from the initiator group using the `/protocols/san/igroups/{igroup.uuid}/igroups` endpoint. See DELETE /protocols/san/igroups/{igroup.uuid}/igroups/{uuid} for more details. |
| initiators | array[initiators] | The initiators that are members of the group. |
| name | string | The name of the initiator group. |
| os_type | string | The host operating system of the initiator group. All initiators in the group should be hosts of the same operating system. |

| Name | Type | Description |
|---|---|---|
| protocol | string | The protocols supported by the initiator group. This restricts the type of initiators that can be added to the initiator group. Optional in POST; if not supplied, this defaults to *mixed*.<br><br>The protocol of an initiator group cannot be changed after creation of the group. |
| uuid | string | The unique identifier of the initiator group. |

lun_maps

A LUN map is an association between a LUN and an initiator group.

When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| igroup | igroup | The initiator group to which the LUN is mapped. |
| logical_unit_number | integer | The logical unit number assigned to the LUN when mapped to the specified initiator group. The number is used to identify the LUN to initiators in the initiator group when communicating through the Fibre Channel Protocol or iSCSI. Optional in POST; if no value is provided, ONTAP assigns the lowest available value. This property is not supported when the *provisioning_options.count* property is 2 or more.<br><br>• Introduced in: 9.6<br>• readCreate: 1<br>• x-nullable: true |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |

| Name | Type | Description |
|---|---|---|
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

metric

Performance numbers, such as IOPS latency and throughput.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |

| Name | Type | Description |
| --- | --- | --- |
| status | string | Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

paths

The fully qualified LUN path names involved in the LUN movement.

| Name | Type | Description |
| --- | --- | --- |
| destination | string | The fully qualified path of the LUN movement destination composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN. |

| Name | Type | Description |
|---|---|---|
| source | string | The fully qualified path of the LUN movement source composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN. |

error

Error information provided if the asynchronous LUN movement operation fails.

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

progress

Properties related to the progress of an active or recently completed LUN movement.

| Name | Type | Description |
|---|---|---|
| elapsed | integer | The amount of time that has elapsed since the start of the LUN movement, in seconds. |
| failure | error | Error information provided if the asynchronous LUN movement operation fails. |
| percent_complete | integer | The percentage completed of the LUN movement. |
| state | string | The state of the LUN movement.<br><br>Valid in PATCH when an LUN movement is active. Set to *paused* to pause a LUN movement. Set to *replicating* to resume a paused LUN movement. |

| Name | Type | Description |
| --- | --- | --- |
| volume_snapshot_blocked | boolean | This property reports if volume snapshots are blocked by the LUN movement. This property can be polled to identify when volume snapshots can be resumed after beginning a LUN movement. |

movement

This sub-object applies to LUN movement between volumes. A LUN can be moved to a new volume with a PATCH request that changes either the volume portion of property `name`, `location.volume.uuid`, or `location.volume.name`. If the volume is changed using more than one of these properties, the supplied properties used must refer to the same volume.

Moving a LUN between volumes is an asynchronous activity begun by a PATCH request. The data for the LUN is then asynchronously copied from the source volume to the destination volume. The time required to complete the move depends on the size of the LUN and the load on the cluster. The `movement` sub-object is populated while a LUN movement is in progress and for two (2) minutes following completion of a movement.

While the LUN is being moved, the status of the LUN movement operation can be obtained using a GET for the LUN that requests the `movement` properties. The LUN movement operation can be further modified using a PATCH on the properties on the `movement` sub-object.

There is an added computational cost to retrieving property values for `movement`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

| Name | Type | Description |
| --- | --- | --- |
| max_throughput | integer | The maximum data throughput, in bytes per second, that should be utilized in support of the LUN movement. This property can be used to throttle a transfer and limit its impact on the performance of the source and destination nodes. The specified value will be rounded up to the nearest megabyte.<br><br>If this property is not specified in a POST that begins a LUN movement, throttling is not applied to the data transfer.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>This property is valid only in a POST that begins a LUN movement or a PATCH when a LUN movement is already in process.<br><br>• format: int64<br>• Introduced in: 9.6<br>• x-nullable: true |
| paths | paths | The fully qualified LUN path names involved in the LUN movement. |
| progress | progress | Properties related to the progress of an active or recently completed LUN movement. |

qos_policy

The QoS policy for the volume provisioned to host the LUN. This property is only supported when the request provisions a new volume. This property is mutually exclusive with the LUN granular `qos_policy`. If no `qos_policy` is provided at LUN or volume granularity, a volume granular policy is set based on the `storage_service.name`, which defaults to the most performant service available.

| Name | Type | Description |
| --- | --- | --- |
| _links | self_link | |

| Name | Type | Description |
|---|---|---|
| name | string | The QoS policy group name. This is mutually exclusive with UUID and other QoS attributes during POST and PATCH. |
| uuid | string | The QoS policy group UUID. This is mutually exclusive with name and other QoS attributes during POST and PATCH. |

snapshot_policy_reference

This is a reference to the snapshot policy.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | |
| uuid | string | |

storage_service

Determines the placement of the LUN based on the value specified. This property is only supported for regular and vvol LUNs. Valid in POST.

object_stores

tiering

The tiering placement and policy definitions for the volume provisioned to host the LUN. This property is only supported when the request provisions a new volume.

| Name | Type | Description |
|---|---|---|
| object_stores | array[object_stores] | Object stores to use. Used for placement. |

| Name | Type | Description |
| --- | --- | --- |
| policy | string | Policy that determines whether the user data blocks of a volume in a FabricPool will be tiered to the cloud store when they become cold.<br><br>FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of a volume block increases if it is accessed frequently and decreases when it is not. Valid in POST or PATCH.<br><br>all ‐ Allows tiering of both snapshots and active file system user data to the cloud store as soon as possible by ignoring the temperature on the volume blocks.<br><br>auto ‐ Allows tiering of both snapshot and active file system user data to the cloud store<br><br>none ‐ Volume blocks are not be tiered to the cloud store.<br><br>snapshot_only ‐ Allows tiering of only the volume snapshots not associated with the active file system.<br><br>The default tiering policy is "snapshot-only" for a FlexVol volume and "none" for a FlexGroup volume. The default minimum cooling period for the "snapshot-only" tiering policy is 2 days and for the "auto" tiering policy it is 31 days. |

provisioning_options

Options that are applied to the operation.

| Name | Type | Description |
|---|---|---|
| storage_service | storage_service | Determines the placement of the LUN based on the value specified. This property is only supported for regular and vvol LUNs. Valid in POST. |

qos_policy

The QoS policy for the LUN. Both traditional and adaptive QoS policies are supported. If both property `qos_policy.uuid` and `qos_policy.name` are specified in the same request, they must refer to the same QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set the property `qos_policy.name` to an empty string ("") in a PATCH request. Valid in POST and PATCH.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set this property to an empty string ("") in a PATCH request. Valid in POST and PATCH. |
| uuid | string | The unique identifier of the QoS policy. Valid in POST and PATCH. |

guarantee

Properties that request and report the space guarantee for the LUN.

| Name | Type | Description |
|---|---|---|
| requested | boolean | The requested space reservation policy for the LUN. If *true*, a space reservation is requested for the LUN; if *false*, the LUN is thin provisioned. Guaranteeing a space reservation request for a LUN requires that the volume in which the LUN resides is also space reserved and that the fractional reserve for the volume is 100%. Valid in POST and PATCH. This property is caller settable as described above. |

| Name | Type | Description |
| --- | --- | --- |
| reserved | boolean | Reports if the LUN is space guaranteed.<br><br>If *true*, a space guarantee is requested and the containing volume and aggregate support the request. If *false*, a space guarantee is not requested or a space guarantee is requested and either the containing volume or aggregate do not support the request. |

space

The storage space related properties of the LUN.

| Name | Type | Description |
| --- | --- | --- |
| efficiency_ratio | number | The storage efficiency ratio of the LUN without snapshots. (Logical Used / Used) This property is not available on the LUN object in the REST API and is not reported for GET requests. See the containing volume object for this information. |
| guarantee | guarantee | Properties that request and report the space guarantee for the LUN. |
| physical_used | integer | The number of bytes consumed on the disk by the LUN, excluding snapshots. This property is not available on the LUN object in the REST API and is not reported for GET requests. See the containing volume object for this information. |
| physical_used_by_snapshots | integer | The number of bytes consumed on the disk by the LUN's snapshots. This property is not available on the LUN object in the REST API and is not reported for GET requests. See the containing volume object for this information. |

| Name | Type | Description |
| --- | --- | --- |
| scsi_thin_provisioning_support_enabled | boolean | To leverage the benefits of SCSI thin provisioning, it must be supported by your host. SCSI thin provisioning uses the Logical Block Provisioning feature as defined in the SCSI SBC-3 standard. Only hosts that support this standard can use SCSI thin provisioning in ONTAP.<br><br>When you disable SCSI thin provisioning support in ONTAP, you turn off the following SCSI thin provisioning features:<br><br>• Unmapping and reporting space usage for space reclamation<br>• Reporting resource exhaustion errors<br><br>The value of this property is not propagated to the destination when a LUN is cloned as a new LUN or copied; it is reset to false. The value of this property is maintained from the destination LUN when a LUN is overwritten as a clone.<br><br>Valid in POST and PATCH.<br><br>• Default value: 1<br>• Introduced in: 9.10<br>• x-nullable: true |

| Name | Type | Description |
| --- | --- | --- |
| size | integer | The total provisioned size of the LUN. The LUN size can be increased but not decreased using the REST interface.<br><br>The maximum and minimum sizes listed here are the absolute maximum and absolute minimum sizes, in bytes. The actual minimum and maximum sizes vary depending on the ONTAP version, ONTAP platform and the available space in the containing volume and aggregate.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>• example: 1073741824<br>• format: int64<br>• Max value: 140737488355328<br>• Min value: 4096<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
|---|---|---|
| used | integer | The amount of space consumed by the main data stream of the LUN.<br><br>This value is the total space consumed in the volume by the LUN, including filesystem overhead, but excluding prefix and suffix streams. Due to internal filesystem overhead and the many ways SAN filesystems and applications utilize blocks within a LUN, this value does not necessarily reflect actual consumption/availability from the perspective of the filesystem or application. Without specific knowledge of how the LUN blocks are utilized outside of ONTAP, this property should not be used as an indicator for an out-of-space condition.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>• format: int64<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

iops_raw

The number of I/O operations observed at the storage object. This can be used along with delta time to calculate the rate of I/O operations per unit of time.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency_raw

The raw latency in microseconds observed at the storage object. This can be divided by the raw IOPS value to calculate the average latency per I/O operation.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput_raw

Throughput bytes observed at the storage object. This can be used along with delta time to calculate the rate of throughput bytes per unit of time.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |

| Name | Type | Description |
|---|---|---|
| write | integer | Performance metric for write I/O operations. |

statistics

These are raw performance numbers, such as IOPS latency and throughput. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

| Name | Type | Description |
|---|---|---|
| iops_raw | iops_raw | The number of I/O operations observed at the storage object. This can be used along with delta time to calculate the rate of I/O operations per unit of time. |
| latency_raw | latency_raw | The raw latency in microseconds observed at the storage object. This can be divided by the raw IOPS value to calculate the average latency per I/O operation. |
| status | string | Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |

| Name | Type | Description |
|---|---|---|
| throughput_raw | throughput_raw | Throughput bytes observed at the storage object. This can be used along with delta time to calculate the rate of throughput bytes per unit of time. |
| timestamp | string | The timestamp of the performance data. |

status

Status information about the LUN.

| Name | Type | Description |
|---|---|---|
| container_state | string | The state of the volume and aggregate that contain the LUN. LUNs are only available when their containers are available. |
| mapped | boolean | Reports if the LUN is mapped to one or more initiator groups.<br><br>There is an added computational cost to retrieving this property's value. It is not populated for a GET request unless it is explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| read_only | boolean | Reports if the LUN allows only read access. |
| state | string | The state of the LUN. Normal states for a LUN are *online* and *offline*. Other states indicate errors. |

svm

The SVM in which the LUN is located.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

partner

The LUN partner that this LUN is bound to. If this LUN is a `vvol` class LUN, the partner is a `protocol_endpoint` class LUN.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the partner LUN. |
| uuid | string | The unique identifier of the partner LUN. |

bindings

A vVol binding with which the LUN is associated.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
| --- | --- | --- |
| id | integer | The ONTAP internal identifier assigned to the vVol binding. The bind identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>This property was included in early releases of the REST API for vVols and is maintained for backward compatibility. See the `secondary_id` property, which replaces `id`.<br><br>• example: 1<br>• readOnly: 1<br>• x-ntap-deprecated: 9.13.1<br>• Introduced in: 9.10<br>• x-nullable: true |
| partner | partner | The LUN partner that this LUN is bound to. If this LUN is a `vvol` class LUN, the partner is a `protocol_endpoint` class LUN. |
| secondary_id | string | The identifier assigned to the vVol binding, known as the secondary LUN ID. The identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>The format for a secondary LUN ID is 16 hexadecimal digits (zero-filled) followed by a lower case "h". |

vvol

A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.

See DELETE /protocols/san/vvol-bindings to learn more about deleting vVol bindings.

There is an added computational cost to retrieving property values for `vvol`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

| Name | Type | Description |
|---|---|---|
| bindings | array[bindings] | Bindings between the LUN, which must be of class `protocol_endpoint` or `vvol`, and LUNs of the opposite class.<br><br>A class `vvol` LUN must be bound to a class `protocol_endpoint` LUN in order to be accessed. Class `protocol_endpoint` and `vvol` LUNs allow many-to-many bindings. A LUN of one class is allowed to be bound to zero or more LUNs of the opposite class. The binding between any two specific LUNs is reference counted. When a binding is created that already exists, the binding count is incremented. When a binding is deleted, the binding count is decremented, but the LUNs remain bound if the resultant reference count is greater than zero. When the binding count reaches zero, the binding is destroyed.<br><br>The bindings array contains LUNs of the opposite class of the containing LUN object.<br><br>There is an added computational cost to retrieving property values for `vvol.bindings`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

| Name | Type | Description |
|---|---|---|
| is_bound | boolean | Reports if the LUN is part of a VMware virtual volume (vVol) bind relationship. This is `true` if the LUN is of class `protocol_endpoint` or `vvol` and has one or more bindings to a LUN of the opposite class. This is false if the LUN is of class `regular` or unbound. |

returned_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 existing LUN

PATCH `/storage/luns/{uuid}`

**Introduced In:** 9.6

Updates an existing LUN in one of several ways:

- Updates the properties of a LUN.
- Writes data to a LUN. LUN data write requests are distinguished by the header entry `Content-Type: multipart/form-data`. When this header entry is provided, query parameter `data.offset` is required and used to specify the location within the LUN at which to write the data; no other query parameters are allowed. The request body must be `multipart/form-data` content with exactly one form entry containing the data to write. The content type entry of the form data is ignored and always treated as `application/octet-stream`. Writes are limited to one megabyte (1MB) per request.
- Overwrites the contents of a LUN as a clone of another.
- Begins the movement of a LUN between volumes. PATCH can also pause and resume the movement of a LUN between volumes that is already in active.

### Related ONTAP commands

- `lun copy modify`

- `lun copy pause`
- `lun copy resume`
- `lun modify`
- `lun move-in-volume`
- `lun move modify`
- `lun move pause`
- `lun move resume`
- `lun move start`
- `lun resize`
- `volume file clone autodelete`

### Learn more

- DOC /storage/luns

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| uuid | string | path | True | The unique identifier of the LUN to retrieve. |

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| data.offset | integer | query | False | The offset, in bytes, at which to begin writing LUN data.<br><br>LUN data write requests are distinguished by the header entry `Content-Type: multipart/form-data`. When this header entry is provided, query parameter `data.offset` is required and used to specify the location within the LUN at which to write the data; no other query parameters are allowed. The request body must be `multipart/form-data` content with exactly one form entry containing the data to write. The content type entry of the form data is ignored and always treated as `application/octet-stream`. Writes are limited to one megabyte (1MB) per request.<br><br>• format: int64<br>• Min value: 0<br>• Introduced in: 9.11 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| return_timeout | integer | query | False | The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.<br>• Min value: 0<br>• Default value: 0<br>• Max value: 120 |

**Request Body**

| Name | Type | Description |
| --- | --- | --- |
| auto_delete | boolean | This property marks the LUN for auto deletion when the volume containing the LUN runs out of space. This is most commonly set on LUN clones.<br><br>When set to *true*, the LUN becomes eligible for automatic deletion when the volume runs out of space. Auto deletion only occurs when the volume containing the LUN is also configured for auto deletion and free space in the volume decreases below a particular threshold.<br><br>This property is optional in POST and PATCH. The default value for a new LUN is *false*.<br><br>There is an added computational cost to retrieving this property's value. It is not populated for a GET request unless it is explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

| Name | Type | Description |
| --- | --- | --- |
| clone | clone | This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: `auto_delete`, `qos_policy`, `space.guarantee.requested` and `space.scsi_thin_provisioning_support_enabled`.<br><br>When used in a PATCH, the patched LUN's data is over-written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: `class`, `auto_delete`, `lun_maps`, `serial_number`, `status.state`, and `uuid`.<br><br>Persistent reservations for the patched LUN are also preserved. |
| comment | string | A configurable comment available for use by the administrator. Valid in POST and PATCH. |
| consistency_group | consistency_group | The LUN's consistency group. This property is populated for LUNs that are members of a consistency group. If the LUN is a member of a child consistency group, the parent consistency group is reported. A LUN's consistency group is the consistency group of its containing volume. |

| Name | Type | Description |
| --- | --- | --- |
| copy | copy | This sub-object applies to LUN copy operations. A LUN can be copied with a POST request that supplies `copy.source` properties.<br><br>Copying a LUN is an asynchronous activity begun by a POST request that specifies the source of the copy in the `copy.source` properties. The data for the LUN is then asynchronously copied from the source to the destination. The time required to complete the copy depends on the size of the LUN and the load on the cluster. The `copy` sub-object is populated while a LUN copy is in progress and for two (2) minutes following completion of a copy.<br><br>While LUNs are being copied, the status of the LUN copy operations can be obtained using a GET of the source or destination LUN that requests the `copy` properties. If the LUN is the source LUN for one or more copy operations, the `copy.destinations` array is populated in GET. If the containing LUN is the destination LUN for a copy operation, the `copy.source` sub-object is populated in GET. The LUN copy operation can be further modified using a PATCH on the properties on the `copy.source` sub-object of the copy destination LUN.<br><br>There is an added computational cost to retrieving property values for `copy`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| create_time | string | The time the LUN was created. |

| Name | Type | Description |
| --- | --- | --- |
| enabled | boolean | The enabled state of the LUN. LUNs can be disabled to prevent access to the LUN. Certain error conditions also cause the LUN to become disabled. If the LUN is disabled, check the `status.state` property to determine if the LUN is administratively disabled (*offline*) or has become disabled as a result of an error. A LUN in an error condition can be brought online by setting the `enabled` property to *true* or brought administratively offline by setting the `enabled` property to *false*. Upon creation, a LUN is enabled by default. Valid in PATCH. |
| location | location | The location of the LUN within the ONTAP cluster. LUNs support rename and move between volumes. Valid in POST and PATCH.<br>• Introduced in: 9.6 |

| Name | Type | Description |
| --- | --- | --- |
| movement | movement | This sub-object applies to LUN movement between volumes. A LUN can be moved to a new volume with a PATCH request that changes either the volume portion of property `name`, `location.volume.uuid`, or `location.volume.name`. If the volume is changed using more than one of these properties, the supplied properties used must refer to the same volume.<br><br>Moving a LUN between volumes is an asynchronous activity begun by a PATCH request. The data for the LUN is then asynchronously copied from the source volume to the destination volume. The time required to complete the move depends on the size of the LUN and the load on the cluster. The `movement` sub-object is populated while a LUN movement is in progress and for two (2) minutes following completion of a movement.<br><br>While the LUN is being moved, the status of the LUN movement operation can be obtained using a GET for the LUN that requests the `movement` properties. The LUN movement operation can be further modified using a PATCH on the properties on the `movement` sub-object.<br><br>There is an added computational cost to retrieving property values for `movement`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

| Name | Type | Description |
| --- | --- | --- |
| name | string | The name of the LUN. Valid in POST and PATCH.<br><br>A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<lun>" where the qtree name is optional.<br><br>A PATCH that modifies the qtree and/or base name portion of the LUN path is considered a rename operation.<br><br>A PATCH that modifies the volume portion of the LUN path begins an asynchronous LUN movement operation. |
| qos_policy | qos_policy | The QoS policy for the LUN. Both traditional and adaptive QoS policies are supported. If both property `qos_policy.uuid` and `qos_policy.name` are specified in the same request, they must refer to the same QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set the property `qos_policy.name` to an empty string ("") in a PATCH request. Valid in POST and PATCH. |
| serial_number | string | The LUN serial number. The serial number is generated by ONTAP when the LUN is created.<br><br>• maxLength: 12<br>• minLength: 12<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
| --- | --- | --- |
| serial_number_hex | string | The LUN serial number encoded in hexadecimal format. The serial number is generated by ONTAP when the LUN is created.<br>• readOnly: 1<br>• Introduced in: 9.17<br>• x-nullable: true |
| space | space | The storage space related properties of the LUN. |
| status | status | Status information about the LUN. |
| uuid | string | The unique identifier of the LUN. The UUID is generated by ONTAP when the LUN is created.<br>• example: 1cd8a442-86d1-11e0-ae1c-123478563412<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
| --- | --- | --- |
| vvol | vvol | A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.<br><br>See DELETE /protocols/san/vvol-bindings to learn more about deleting vVol bindings.<br><br>There is an added computational cost to retrieving property values for `vvol`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

**Example request**


```
{
  "clone": {
    "source": {
      "name": "/vol/volume1/lun1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  },
  "comment": "string",
  "consistency_group": {
    "name": "cg1",
    "uuid": "4abc2317-4332-9d37-93a0-20bd29c22df0"
  },
  "copy": {
    "destinations": [
      {
        "max_throughput": 0,
        "name": "/vol/vol1/lun1",
        "peer": {
          "name": "peer1",
          "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
        },
        "progress": {
          "elapsed": 0,
          "percent_complete": 0,
          "state": "string"
        },
        "uuid": "1bc327d5-4654-5284-a116-f182282240b4"
      }
    ],
    "source": {
      "peer": {
        "name": "peer1",
        "uuid": "4204cf77-4c82-9bdb-5644-b5a841c097a9"
      },
      "progress": {
        "elapsed": 0,
        "percent_complete": 0,
        "state": "string"
      }
    }
  },
  "create_time": "2018-06-04 15:00:00 -0400",
  "location": {
    "logical_unit": "lun1",
```

```
    "node": {
      "name": "node1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "qtree": {
      "id": 1,
      "name": "qt1"
    },
    "volume": {
      "name": "volume1",
      "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
    }
  },
  "movement": {
    "paths": {
      "destination": "/vol/vol1/lun1",
      "source": "/vol/vol2/lun2"
    },
    "progress": {
      "elapsed": 0,
      "percent_complete": 0,
      "state": "string"
    }
  },
  "name": "/vol/volume1/qtree1/lun1",
  "qos_policy": {
    "name": "qos1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "serial_number": "string",
  "serial_number_hex": "string",
  "space": {
    "efficiency_ratio": 2.5,
    "physical_used": 1073741824,
    "physical_used_by_snapshots": 1073741824,
    "size": 1073741824,
    "used": 0
  },
  "status": {
    "container_state": "string",
    "state": "online"
  },
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412",
  "vvol": {
    "bindings": [
      {
```

```
        "id": 1,
        "partner": {
          "name": "/vol/vol1/lun1",
          "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
        },
        "secondary_id": "0000D20000010000h"
      }
    ]
  }
}
```

### Response

```
Status: 200, Ok
```

### Response

```
Status: 202, Accepted
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 917927 | The specified volume was not found. |
| 918236 | The specified `location.volume.uuid` and `location.volume.name` do not refer to the same volume. |
| 1254196 | A LUN already exists at the specified path. |
| 1254226 | Cannot patch a LUN's size to its current size. |
| 5242927 | The specified qtree was not found. |
| 5242950 | The specified `location.qtree.id` and `location.qtree.name` do not refer to the same qtree. |
| 5374124 | The specified LUN size is too small. |
| 5374125 | The specified LUN size is too large. |
| 5374127 | The specified LUN name is invalid. |

| Error Code | Description |
| --- | --- |
| 5374130 | An invalid size value was provided. |
| 5374241 | A size value with invalid units was provided. |
| 5374480 | Modifying the LUN is not allowed because it is in a foreign LUN import relationship. |
| 5374858 | The volume specified by `name` is not the same as that specified by `location.volume`. |
| 5374860 | The qtree specified by `name` is not the same as that specified by `location.qtree`. |
| 5374861 | The LUN base name specified by `name` is not the same as that specified by `location.logical_unit`. |
| 5374864 | An error occurred after successfully overwriting data for the LUN as a clone. Some properties were not modified. |
| 5374865 | The LUN's aggregate is offline. The aggregate must be online to modify or remove the LUN. |
| 5374866 | The LUN's volume is offline. The volume must be online to modify or remove the LUN. |
| 5374874 | The specified `clone.source.uuid` and `clone.source.name` do not refer to the same LUN. |
| 5374875 | The specified LUN was not found. This can apply to `clone.source` or the target LUN. The `target` property of the error object identifies the property. |
| 5374876 | The specified LUN was not found. This can apply to `clone.source` or the target LUN. The `target` property of the error object identifies the property. |
| 5374885 | An error occurred after successfully modifying some of the properties of the LUN. Some properties were not modified. |
| 5374889 | An invalid value was specified for `movement.progress.state`. Active LUN movement operations can be PATCHed to only *paused* or *replicating*. |
| 5374892 | An attempt was made to reduce the size of a LUN. |
| 5374904 | The destination volume is not online. |
| 5375059 | An unsuitable QoS policy was specified. |
| 5376461 | The specified LUN name is invalid. |
| 5376462 | The specified LUN name is too long. |
| 5376463 | The snapshot portion of the specified LUN name is too long. |

| Error Code | Description |
| --- | --- |
| 5376466 | An attempt was made to rename a LUN to a snapshot name. |
| 5376467 | An attempt was made to rename a primary LUN to a secondary name. |
| 5376468 | An attempt was made to rename a LUN to a reserved name. |
| 5376470 | The property cannot be set during the LUN modify operation on this platform. |
| 7018877 | Maximum combined total (50) of file and LUN copy and move operations reached. When one or more of the operations has completed, try the command again. |
| 7018919 | A copy or move job exists with the same destination LUN. |
| 8454243 | The specified QoS policy does not exist while modifying a LUN. |
| 13565952 | The LUN clone request failed. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

attributes

A name/value pair optionally stored with the LUN. Attributes are available to callers to persist small amounts of application-specific metadata. They are in no way interpreted by ONTAP.

Attribute names and values must be at least one byte and no more than 4091 bytes in length. The sum of the name and value lengths must be no more than 4092 bytes.

Optional in POST.

source

The source LUN for a LUN clone operation. This can be specified using property `clone.source.uuid` or `clone.source.name`. If both properties are supplied, they must refer to the same LUN.

Valid in POST to create a new LUN as a clone of the source.

Valid in PATCH to overwrite an existing LUN's data as a clone of another.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the clone source LUN. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<namespace>" where the qtree name is optional.<br><br>Valid in POST and PATCH. |
| uuid | string | The unique identifier of the clone source LUN. Valid in POST and PATCH. |

clone

This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: `auto_delete`,

`qos_policy`, `space.guarantee.requested` and `space.scsi_thin_provisioning_support_enabled`.

When used in a PATCH, the patched LUN's data is over-written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: `class`, `auto_delete`, `lun_maps`, `serial_number`, `status.state`, and `uuid`.

Persistent reservations for the patched LUN are also preserved.

| Name | Type | Description |
|---|---|---|
| source | source | The source LUN for a LUN clone operation. This can be specified using property `clone.source.uuid` or `clone.source.name`. If both properties are supplied, they must refer to the same LUN.<br><br>Valid in POST to create a new LUN as a clone of the source.<br><br>Valid in PATCH to overwrite an existing LUN's data as a clone of another. |

consistency_group

The LUN's consistency group. This property is populated for LUNs that are members of a consistency group. If the LUN is a member of a child consistency group, the parent consistency group is reported. A LUN's consistency group is the consistency group of its containing volume.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the consistency group. |
| uuid | string | The unique identifier of the consistency group. |

namespace

The source namespace for convert operation. This can be specified using property `convert.namespace.uuid` or `convert.namespace.name`. If both properties are supplied, they must refer to the same NVMe namespace.

Valid in POST. A convert request from NVMe namespace to LUN cannot be combined with setting any other LUN properties. All other properties of the converted LUN come from the source NVMe namespace.

convert

This sub-object is used in POST to convert a valid in-place NVMe namespace to a LUN. Setting a property in this sub-object indicates that a conversion from the specified NVMe namespace to LUN is

desired.

peer

The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.

This is only populated by GET when the LUN copy is inter-SVM.

| Name | Type | Description |
|---|---|---|
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

error

Error information provided if the asynchronous LUN copy operation fails.

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

progress

Properties related to the progress of an active or recently completed LUN copy.

| Name | Type | Description |
|---|---|---|
| elapsed | integer | The amount of time that has elapsed since the start of the LUN copy, in seconds. |

| Name | Type | Description |
|---|---|---|
| percent_complete | integer | The percentage completed of the LUN copy. |
| state | string | The state of the LUN copy. |
| volume_snapshot_blocked | boolean | This property reports if volume snapshots are blocked by the LUN copy. This property can be polled to identify when volume snapshots can be resumed after beginning a LUN copy. |

destinations

A LUN copy operation in which the containing LUN is the source of the copy.

| Name | Type | Description |
|---|---|---|
| max_throughput | integer | The maximum data throughput, in bytes per second, that should be utilized in support of the LUN copy. See property `copy.source.max_throughput` for further details. |
| name | string | The fully qualified path of the LUN copy destination composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN. |
| peer | peer | The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.<br><br>This is only populated by GET when the LUN copy is inter-SVM. |
| progress | progress | Properties related to the progress of an active or recently completed LUN copy. |
| uuid | string | The unique identifier of the LUN copy destination. |

peer

The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.

Set this in POST to specify the source SVM for an inter-SVM LUN copy. Only populated by GET when the LUN copy is inter-SVM.

| Name | Type | Description |
|---|---|---|
| name | string | The local name of the peer SVM. This name is unique among all local and peer SVMs. |
| uuid | string | The unique identifier of the SVM peer relationship. This is the UUID of the relationship, not the UUID of the peer SVM itself. |

progress

Properties related to the progress of an active or recently completed LUN copy.

| Name | Type | Description |
|---|---|---|
| elapsed | integer | The amount of time that has elapsed since the start of the LUN copy, in seconds. |
| percent_complete | integer | The percentage completed of the LUN copy. |
| state | string | The state of the LUN copy.<br><br>Valid in PATCH when an LUN copy is active. Set to *paused* to pause a LUN copy. Set to *replicating* to resume a paused LUN copy. |
| volume_snapshot_blocked | boolean | This property reports if volume snapshots are blocked by the LUN copy. This property can be polled to identify when volume snapshots can be resumed after beginning a LUN copy. |

source

The source LUN of a LUN copy operation in which the containing LUN is the destination of the copy.

Valid in POST except when creating a LUN clone. A LUN copy request cannot be combined with setting

any other LUN properties except the destination location. All other properties of the destination LUN come from the source LUN.

| Name | Type | Description |
|---|---|---|
| max_throughput | integer | The maximum data throughput, in bytes per second, that should be utilized in support of the LUN copy. This property can be used to throttle a transfer and limit its impact on the performance of the source and destination nodes. The specified value will be rounded up to the nearest megabyte.<br><br>If this property is not specified in a POST that begins a LUN copy, throttling is not applied to the data transfer.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>Valid only in a POST that begins a LUN copy or a PATCH when a LUN copy is already in process.<br><br>• format: int64<br>• Introduced in: 9.10<br>• x-nullable: true |
| peer | peer | The SVM peer relationship object for an inter-SVM LUN copy operation. The peer SVM in the relationship is the source SVM and the local SVM is the destination SVM.<br><br>Set this in POST to specify the source SVM for an inter-SVM LUN copy. Only populated by GET when the LUN copy is inter-SVM. |
| progress | progress | Properties related to the progress of an active or recently completed LUN copy. |

copy

This sub-object applies to LUN copy operations. A LUN can be copied with a POST request that supplies `copy.source` properties.

Copying a LUN is an asynchronous activity begun by a POST request that specifies the source of the copy in the `copy.source` properties. The data for the LUN is then asynchronously copied from the source to the destination. The time required to complete the copy depends on the size of the LUN and the load on the cluster. The `copy` sub-object is populated while a LUN copy is in progress and for two (2) minutes following completion of a copy.

While LUNs are being copied, the status of the LUN copy operations can be obtained using a GET of the source or destination LUN that requests the `copy` properties. If the LUN is the source LUN for one or more copy operations, the `copy.destinations` array is populated in GET. If the containing LUN is the destination LUN for a copy operation, the `copy.source` sub-object is populated in GET. The LUN copy operation can be further modified using a PATCH on the properties on the `copy.source` sub-object of the copy destination LUN.

There is an added computational cost to retrieving property values for `copy`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

| Name | Type | Description |
|---|---|---|
| destinations | array[destinations] | An array of destination LUNs of LUN copy operations in which the containing LUN is the source of the copy. |
| source | source | The source LUN of a LUN copy operation in which the containing LUN is the destination of the copy.<br><br>Valid in POST except when creating a LUN clone. A LUN copy request cannot be combined with setting any other LUN properties except the destination location. All other properties of the destination LUN come from the source LUN. |

node

The cluster node that hosts the LUN.

| Name | Type | Description |
|---|---|---|
| name | string | |
| uuid | string | |

qtree

The qtree in which the LUN is optionally located. Valid in POST and PATCH.

If properties `name` and `location.qtree.name` and/or `location.qtree.id` are specified in the same request, they must refer to the same qtree.

A PATCH that modifies the qtree of the LUN is considered a rename operation.

| Name | Type | Description |
|---|---|---|
| id | integer | The identifier for the qtree, unique within the qtree's volume. |
| name | string | The name of the qtree. |

volume

The volume in which the LUN is located. Valid in POST and PATCH.

If properties `name` and `location.volume.name` and/or `location.volume.uuid` are specified in the same request, they must refer to the same volume.

A PATCH that modifies the volume of the LUN begins an asynchronous LUN movement operation.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the volume. This field cannot be specified in a PATCH method. |
| uuid | string | Unique identifier for the volume. This corresponds to the instance-uuid that is exposed in the CLI and ONTAPI. It does not change due to a volume move.<br>• example: 028baa66-41bd-11e9-81d5-00a0986138f7<br>• Introduced in: 9.6<br>• x-nullable: true |

location

The location of the LUN within the ONTAP cluster. LUNs support rename and move between volumes. Valid in POST and PATCH.

| Name | Type | Description |
| --- | --- | --- |
| logical_unit | string | The base name component of the LUN. Valid in POST and PATCH.<br><br>If properties `name` and `location.logical_unit` are specified in the same request, they must refer to the base name.<br><br>A PATCH that modifies the base name of the LUN is considered a rename operation. |
| node | node | The cluster node that hosts the LUN. |
| qtree | qtree | The qtree in which the LUN is optionally located. Valid in POST and PATCH.<br><br>If properties `name` and `location.qtree.name` and/or `location.qtree.id` are specified in the same request, they must refer to the same qtree.<br><br>A PATCH that modifies the qtree of the LUN is considered a rename operation. |
| volume | volume | The volume in which the LUN is located. Valid in POST and PATCH.<br><br>If properties `name` and `location.volume.name` and/or `location.volume.uuid` are specified in the same request, they must refer to the same volume.<br><br>A PATCH that modifies the volume of the LUN begins an asynchronous LUN movement operation. |

self_link

igroups

| Name | Type | Description |
|---|---|---|
| name | string | The name of the initiator group. |
| uuid | string | The unique identifier of the initiator group. |

initiators

The initiators that are members of the initiator group.

| Name | Type | Description |
|---|---|---|
| comment | string | A comment available for use by the administrator. |
| name | string | Name of initiator that is a member of the initiator group. |

igroup

The initiator group to which the LUN is mapped.

lun_maps

A LUN map is an association between a LUN and an initiator group.

When a LUN is mapped to an initiator group, the initiator group's initiators are granted access to the LUN. The relationship between a LUN and an initiator group is many LUNs to many initiator groups.

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

metric

Performance numbers, such as IOPS latency and throughput.

| Name | Type | Description |
| --- | --- | --- |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |
| status | string | Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |

paths

The fully qualified LUN path names involved in the LUN movement.

| Name | Type | Description |
|---|---|---|
| destination | string | The fully qualified path of the LUN movement destination composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN. |
| source | string | The fully qualified path of the LUN movement source composed of a "/vol" prefix, the volume name, the optional qtree name, and base name of the LUN. |

error

Error information provided if the asynchronous LUN movement operation fails.

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

progress

Properties related to the progress of an active or recently completed LUN movement.

| Name | Type | Description |
|---|---|---|
| elapsed | integer | The amount of time that has elapsed since the start of the LUN movement, in seconds. |
| percent_complete | integer | The percentage completed of the LUN movement. |
| state | string | The state of the LUN movement.<br><br>Valid in PATCH when an LUN movement is active. Set to *paused* to pause a LUN movement. Set to *replicating* to resume a paused LUN movement. |

| Name | Type | Description |
| --- | --- | --- |
| volume_snapshot_blocked | boolean | This property reports if volume snapshots are blocked by the LUN movement. This property can be polled to identify when volume snapshots can be resumed after beginning a LUN movement. |

movement

This sub-object applies to LUN movement between volumes. A LUN can be moved to a new volume with a PATCH request that changes either the volume portion of property `name`, `location.volume.uuid`, or `location.volume.name`. If the volume is changed using more than one of these properties, the supplied properties used must refer to the same volume.

Moving a LUN between volumes is an asynchronous activity begun by a PATCH request. The data for the LUN is then asynchronously copied from the source volume to the destination volume. The time required to complete the move depends on the size of the LUN and the load on the cluster. The `movement` sub-object is populated while a LUN movement is in progress and for two (2) minutes following completion of a movement.

While the LUN is being moved, the status of the LUN movement operation can be obtained using a GET for the LUN that requests the `movement` properties. The LUN movement operation can be further modified using a PATCH on the properties on the `movement` sub-object.

There is an added computational cost to retrieving property values for `movement`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

| Name | Type | Description |
| --- | --- | --- |
| max_throughput | integer | The maximum data throughput, in bytes per second, that should be utilized in support of the LUN movement. This property can be used to throttle a transfer and limit its impact on the performance of the source and destination nodes. The specified value will be rounded up to the nearest megabyte.<br><br>If this property is not specified in a POST that begins a LUN movement, throttling is not applied to the data transfer.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>This property is valid only in a POST that begins a LUN movement or a PATCH when a LUN movement is already in process.<br><br>• format: int64<br>• Introduced in: 9.6<br>• x-nullable: true |
| paths | paths | The fully qualified LUN path names involved in the LUN movement. |
| progress | progress | Properties related to the progress of an active or recently completed LUN movement. |

qos_policy

The QoS policy for the volume provisioned to host the LUN. This property is only supported when the request provisions a new volume. This property is mutually exclusive with the LUN granular `qos_policy`. If no `qos_policy` is provided at LUN or volume granularity, a volume granular policy is set based on the `storage_service.name`, which defaults to the most performant service available.

| Name | Type | Description |
|---|---|---|
| name | string | The QoS policy group name. This is mutually exclusive with UUID and other QoS attributes during POST and PATCH. |
| uuid | string | The QoS policy group UUID. This is mutually exclusive with name and other QoS attributes during POST and PATCH. |

snapshot_policy_reference

This is a reference to the snapshot policy.

| Name | Type | Description |
|---|---|---|
| name | string | |
| uuid | string | |

storage_service

Determines the placement of the LUN based on the value specified. This property is only supported for regular and vvol LUNs. Valid in POST.

object_stores

| Name | Type | Description |
|---|---|---|
| name | string | The name of the object store to use. Used for placement. |

tiering

The tiering placement and policy definitions for the volume provisioned to host the LUN. This property is only supported when the request provisions a new volume.

| Name | Type | Description |
|---|---|---|
| control | string | Storage tiering placement rules for the object. |
| object_stores | array[object_stores] | Object stores to use. Used for placement. |

provisioning_options

Options that are applied to the operation.

| Name | Type | Description |
|---|---|---|
| storage_service | storage_service | Determines the placement of the LUN based on the value specified. This property is only supported for regular and vvol LUNs. Valid in POST. |

qos_policy

The QoS policy for the LUN. Both traditional and adaptive QoS policies are supported. If both property `qos_policy.uuid` and `qos_policy.name` are specified in the same request, they must refer to the same QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set the property `qos_policy.name` to an empty string ("") in a PATCH request. Valid in POST and PATCH.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set this property to an empty string ("") in a PATCH request. Valid in POST and PATCH. |
| uuid | string | The unique identifier of the QoS policy. Valid in POST and PATCH. |

guarantee

Properties that request and report the space guarantee for the LUN.

| Name | Type | Description |
|---|---|---|
| requested | boolean | The requested space reservation policy for the LUN. If *true*, a space reservation is requested for the LUN; if *false*, the LUN is thin provisioned. Guaranteeing a space reservation request for a LUN requires that the volume in which the LUN resides is also space reserved and that the fractional reserve for the volume is 100%. Valid in POST and PATCH. This property is caller settable as described above. |

| Name | Type | Description |
| --- | --- | --- |
| reserved | boolean | Reports if the LUN is space guaranteed.<br><br>If *true*, a space guarantee is requested and the containing volume and aggregate support the request. If *false*, a space guarantee is not requested or a space guarantee is requested and either the containing volume or aggregate do not support the request. |

space

The storage space related properties of the LUN.

| Name | Type | Description |
| --- | --- | --- |
| efficiency_ratio | number | The storage efficiency ratio of the LUN without snapshots. (Logical Used / Used) This property is not available on the LUN object in the REST API and is not reported for GET requests. See the containing volume object for this information. |
| guarantee | guarantee | Properties that request and report the space guarantee for the LUN. |
| physical_used | integer | The number of bytes consumed on the disk by the LUN, excluding snapshots. This property is not available on the LUN object in the REST API and is not reported for GET requests. See the containing volume object for this information. |
| physical_used_by_snapshots | integer | The number of bytes consumed on the disk by the LUN's snapshots. This property is not available on the LUN object in the REST API and is not reported for GET requests. See the containing volume object for this information. |

| Name | Type | Description |
| --- | --- | --- |
| scsi_thin_provisioning_support_enabled | boolean | To leverage the benefits of SCSI thin provisioning, it must be supported by your host. SCSI thin provisioning uses the Logical Block Provisioning feature as defined in the SCSI SBC-3 standard. Only hosts that support this standard can use SCSI thin provisioning in ONTAP.<br><br>When you disable SCSI thin provisioning support in ONTAP, you turn off the following SCSI thin provisioning features:<br><br>• Unmapping and reporting space usage for space reclamation<br>• Reporting resource exhaustion errors<br><br>The value of this property is not propagated to the destination when a LUN is cloned as a new LUN or copied; it is reset to false. The value of this property is maintained from the destination LUN when a LUN is overwritten as a clone.<br><br>Valid in POST and PATCH.<br><br>• Default value: 1<br>• Introduced in: 9.10<br>• x-nullable: true |

| Name | Type | Description |
| --- | --- | --- |
| size | integer | The total provisioned size of the LUN. The LUN size can be increased but not decreased using the REST interface.<br><br>The maximum and minimum sizes listed here are the absolute maximum and absolute minimum sizes, in bytes. The actual minimum and maximum sizes vary depending on the ONTAP version, ONTAP platform and the available space in the containing volume and aggregate.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>• example: 1073741824<br>• format: int64<br>• Max value: 140737488355328<br>• Min value: 4096<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
|---|---|---|
| used | integer | The amount of space consumed by the main data stream of the LUN.<br><br>This value is the total space consumed in the volume by the LUN, including filesystem overhead, but excluding prefix and suffix streams. Due to internal filesystem overhead and the many ways SAN filesystems and applications utilize blocks within a LUN, this value does not necessarily reflect actual consumption/availability from the perspective of the filesystem or application. Without specific knowledge of how the LUN blocks are utilized outside of ONTAP, this property should not be used as an indicator for an out-of-space condition.<br><br>For more information, see *Size properties* in the *docs* section of the ONTAP REST API documentation.<br><br>• format: int64<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

iops_raw

The number of I/O operations observed at the storage object. This can be used along with delta time to calculate the rate of I/O operations per unit of time.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency_raw

The raw latency in microseconds observed at the storage object. This can be divided by the raw IOPS value to calculate the average latency per I/O operation.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput_raw

Throughput bytes observed at the storage object. This can be used along with delta time to calculate the rate of throughput bytes per unit of time.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |

| Name | Type | Description |
| --- | --- | --- |
| write | integer | Performance metric for write I/O operations. |

statistics

These are raw performance numbers, such as IOPS latency and throughput. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

| Name | Type | Description |
| --- | --- | --- |
| iops_raw | iops_raw | The number of I/O operations observed at the storage object. This can be used along with delta time to calculate the rate of I/O operations per unit of time. |
| latency_raw | latency_raw | The raw latency in microseconds observed at the storage object. This can be divided by the raw IOPS value to calculate the average latency per I/O operation. |
| status | string | Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |

| Name | Type | Description |
|---|---|---|
| throughput_raw | throughput_raw | Throughput bytes observed at the storage object. This can be used along with delta time to calculate the rate of throughput bytes per unit of time. |
| timestamp | string | The timestamp of the performance data. |

status

Status information about the LUN.

| Name | Type | Description |
|---|---|---|
| container_state | string | The state of the volume and aggregate that contain the LUN. LUNs are only available when their containers are available. |
| mapped | boolean | Reports if the LUN is mapped to one or more initiator groups.<br><br>There is an added computational cost to retrieving this property's value. It is not populated for a GET request unless it is explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| read_only | boolean | Reports if the LUN allows only read access. |
| state | string | The state of the LUN. Normal states for a LUN are *online* and *offline*. Other states indicate errors. |

svm

The SVM in which the LUN is located.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the SVM. This field cannot be specified in a PATCH method. |

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the SVM. This field cannot be specified in a PATCH method. |

partner

The LUN partner that this LUN is bound to. If this LUN is a `vvol` class LUN, the partner is a `protocol_endpoint` class LUN.

| Name | Type | Description |
|---|---|---|
| name | string | The name of the partner LUN. |
| uuid | string | The unique identifier of the partner LUN. |

bindings

A vVol binding with which the LUN is associated.

| Name | Type | Description |
|---|---|---|
| id | integer | The ONTAP internal identifier assigned to the vVol binding. The bind identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>This property was included in early releases of the REST API for vVols and is maintained for backward compatibility. See the `secondary_id` property, which replaces `id`.<br><br>• example: 1<br>• readOnly: 1<br>• x-ntap-deprecated: 9.13.1<br>• Introduced in: 9.10<br>• x-nullable: true |
| partner | partner | The LUN partner that this LUN is bound to. If this LUN is a `vvol` class LUN, the partner is a `protocol_endpoint` class LUN. |

| Name | Type | Description |
| --- | --- | --- |
| secondary_id | string | The identifier assigned to the vVol binding, known as the secondary LUN ID. The identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.<br><br>The format for a secondary LUN ID is 16 hexadecimal digits (zero-filled) followed by a lower case "h". |

vvol

A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.

See DELETE /protocols/san/vvol-bindings to learn more about deleting vVol bindings.

There is an added computational cost to retrieving property values for `vvol`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more.

| Name | Type | Description |
| --- | --- | --- |
| bindings | array[bindings] | Bindings between the LUN, which must be of class `protocol_endpoint` or `vvol`, and LUNs of the opposite class.<br><br>A class `vvol` LUN must be bound to a class `protocol_endpoint` LUN in order to be accessed. Class `protocol_endpoint` and `vvol` LUNs allow many-to-many bindings. A LUN of one class is allowed to be bound to zero or more LUNs of the opposite class. The binding between any two specific LUNs is reference counted. When a binding is created that already exists, the binding count is incremented. When a binding is deleted, the binding count is decremented, but the LUNs remain bound if the resultant reference count is greater than zero. When the binding count reaches zero, the binding is destroyed.<br><br>The bindings array contains LUNs of the opposite class of the containing LUN object.<br><br>There is an added computational cost to retrieving property values for `vvol.bindings`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| is_bound | boolean | Reports if the LUN is part of a VMware virtual volume (vVol) bind relationship. This is `true` if the LUN is of class `protocol_endpoint` or `vvol` and has one or more bindings to a LUN of the opposite class. This is false if the LUN is of class `regular` or unbound. |

lun

A LUN is the logical representation of storage in a storage area network (SAN).

A LUN must be mapped to an initiator group to grant access to the initiator group's initiators (client hosts). Initiators can then access the LUN and perform I/O over a Fibre Channel (FC) fabric using the FC Protocol or a TCP/IP network using iSCSI.

See the LUN object model to learn more about each of the properties supported by the LUN REST API.

A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.

LUN names are paths of the form "/vol/<volume>[/<qtree>]/<lun>" where the qtree name is optional.

A LUN can be created to a specified size using thin or thick provisioning. A LUN can then be renamed, resized, cloned, moved to a different volume and copied. LUNs support the assignment of a QoS policy for performance management or a QoS policy can be assigned to a volume containing one or more LUNs.

| Name | Type | Description |
|---|---|---|
| auto_delete | boolean | This property marks the LUN for auto deletion when the volume containing the LUN runs out of space. This is most commonly set on LUN clones.<br><br>When set to *true*, the LUN becomes eligible for automatic deletion when the volume runs out of space. Auto deletion only occurs when the volume containing the LUN is also configured for auto deletion and free space in the volume decreases below a particular threshold.<br><br>This property is optional in POST and PATCH. The default value for a new LUN is *false*.<br><br>There is an added computational cost to retrieving this property's value. It is not populated for a GET request unless it is explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

| Name | Type | Description |
| --- | --- | --- |
| clone | clone | This sub-object is used in POST to create a new LUN as a clone of an existing LUN, or PATCH to overwrite an existing LUN as a clone of another. Setting a property in this sub-object indicates that a LUN clone is desired. Consider the following other properties when cloning a LUN: `auto_delete`, `qos_policy`, `space.guarantee.requested` and `space.scsi_thin_provisioning_support_enabled`.<br><br>When used in a PATCH, the patched LUN's data is over-written as a clone of the source and the following properties are preserved from the patched LUN unless otherwise specified as part of the PATCH: `class`, `auto_delete`, `lun_maps`, `serial_number`, `status.state`, and `uuid`.<br><br>Persistent reservations for the patched LUN are also preserved. |
| comment | string | A configurable comment available for use by the administrator. Valid in POST and PATCH. |
| consistency_group | consistency_group | The LUN's consistency group. This property is populated for LUNs that are members of a consistency group. If the LUN is a member of a child consistency group, the parent consistency group is reported. A LUN's consistency group is the consistency group of its containing volume. |

| Name | Type | Description |
| --- | --- | --- |
| copy | copy | This sub-object applies to LUN copy operations. A LUN can be copied with a POST request that supplies `copy.source` properties.<br><br>Copying a LUN is an asynchronous activity begun by a POST request that specifies the source of the copy in the `copy.source` properties. The data for the LUN is then asynchronously copied from the source to the destination. The time required to complete the copy depends on the size of the LUN and the load on the cluster. The `copy` sub-object is populated while a LUN copy is in progress and for two (2) minutes following completion of a copy.<br><br>While LUNs are being copied, the status of the LUN copy operations can be obtained using a GET of the source or destination LUN that requests the `copy` properties. If the LUN is the source LUN for one or more copy operations, the `copy.destinations` array is populated in GET. If the containing LUN is the destination LUN for a copy operation, the `copy.source` sub-object is populated in GET. The LUN copy operation can be further modified using a PATCH on the properties on the `copy.source` sub-object of the copy destination LUN.<br><br>There is an added computational cost to retrieving property values for `copy`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |
| create_time | string | The time the LUN was created. |

| Name | Type | Description |
| --- | --- | --- |
| enabled | boolean | The enabled state of the LUN. LUNs can be disabled to prevent access to the LUN. Certain error conditions also cause the LUN to become disabled. If the LUN is disabled, check the `status.state` property to determine if the LUN is administratively disabled (*offline*) or has become disabled as a result of an error. A LUN in an error condition can be brought online by setting the `enabled` property to *true* or brought administratively offline by setting the `enabled` property to *false*. Upon creation, a LUN is enabled by default. Valid in PATCH. |
| location | location | The location of the LUN within the ONTAP cluster. LUNs support rename and move between volumes. Valid in POST and PATCH.<br>• Introduced in: 9.6 |

| Name | Type | Description |
| --- | --- | --- |
| movement | movement | This sub-object applies to LUN movement between volumes. A LUN can be moved to a new volume with a PATCH request that changes either the volume portion of property `name`, `location.volume.uuid`, or `location.volume.name`. If the volume is changed using more than one of these properties, the supplied properties used must refer to the same volume.<br><br>Moving a LUN between volumes is an asynchronous activity begun by a PATCH request. The data for the LUN is then asynchronously copied from the source volume to the destination volume. The time required to complete the move depends on the size of the LUN and the load on the cluster. The `movement` sub-object is populated while a LUN movement is in progress and for two (2) minutes following completion of a movement.<br><br>While the LUN is being moved, the status of the LUN movement operation can be obtained using a GET for the LUN that requests the `movement` properties. The LUN movement operation can be further modified using a PATCH on the properties on the `movement` sub-object.<br><br>There is an added computational cost to retrieving property values for `movement`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

| Name | Type | Description |
| --- | --- | --- |
| name | string | The name of the LUN. Valid in POST and PATCH.<br><br>A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.<br><br>LUN names are paths of the form "/vol/<volume>[/<qtree>]/<lun>" where the qtree name is optional.<br><br>A PATCH that modifies the qtree and/or base name portion of the LUN path is considered a rename operation.<br><br>A PATCH that modifies the volume portion of the LUN path begins an asynchronous LUN movement operation. |
| qos_policy | qos_policy | The QoS policy for the LUN. Both traditional and adaptive QoS policies are supported. If both property `qos_policy.uuid` and `qos_policy.name` are specified in the same request, they must refer to the same QoS policy. To remove the QoS policy from a LUN, leaving it with no QoS policy, set the property `qos_policy.name` to an empty string ("") in a PATCH request. Valid in POST and PATCH. |
| serial_number | string | The LUN serial number. The serial number is generated by ONTAP when the LUN is created.<br><br>• maxLength: 12<br>• minLength: 12<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
| --- | --- | --- |
| serial_number_hex | string | The LUN serial number encoded in hexadecimal format. The serial number is generated by ONTAP when the LUN is created.<br>• readOnly: 1<br>• Introduced in: 9.17<br>• x-nullable: true |
| space | space | The storage space related properties of the LUN. |
| status | status | Status information about the LUN. |
| uuid | string | The unique identifier of the LUN. The UUID is generated by ONTAP when the LUN is created.<br>• example: 1cd8a442-86d1-11e0-ae1c-123478563412<br>• readOnly: 1<br>• Introduced in: 9.6<br>• x-nullable: true |

| Name | Type | Description |
|---|---|---|
| vvol | vvol | A VMware virtual volume (vVol) binding is an association between a LUN of class `protocol_endpoint` and a LUN of class `vvol`. Class `protocol_endpoint` LUNs are mapped to igroups and granted access using the same configuration as class `regular` LUNs. When a class `vvol` LUN is bound to a mapped class `protocol_endpoint` LUN, VMware can access the class `vvol` LUN through the class `protocol_endpoint` LUN mapping.<br><br>See DELETE /protocols/san/vvol-bindings to learn more about deleting vVol bindings.<br><br>There is an added computational cost to retrieving property values for `vvol`. They are not populated for a GET request unless explicitly requested using the `fields` query parameter. See Requesting specific fields to learn more. |

returned_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. |

# Manage LUN attributes

# Manage LUN attributes

## Overview

LUN attributes are caller-defined name/value pairs optionally stored with a LUN. Attributes are available to persist small amounts of application-specific metadata. They are in no way interpreted by ONTAP.

Attribute names and values must be at least one byte and no more than 4091 bytes in length. The sum of the name and value lengths must be no more than 4092 bytes.

The LUN attributes REST API allows you to create, update, delete, and discover attributes for a LUN. The LUN REST API also allows you to set attributes when a LUN is first created.

## Examples

### Retrieving all attributes from a LUN

This example uses the LUN attribute REST API endpoint with the `fields` query parameter to request the names and values.

```
# The API:
GET /api/storage/luns/{lun.uuid}/attributes

# The call:
curl -X GET 'https://<mgmt-ip>/api/storage/luns/4bc204df-ecd8-4f35-8207-
d0ccb4db3a90/attributes?fields=*' -H 'Accept: application/hal+json'

# The response:
{
"records": [
  {
    "name": "name1",
    "value": "value1",
    "_links": {
      "self": {
        "href": "/api/storage/luns/4bc204df-ecd8-4f35-8207-
d0ccb4db3a90/attributes/name1"
      }
    }
  },
  {
    "name": "name2",
    "value": "value2",
    "_links": {
      "self": {
        "href": "/api/storage/luns/4bc204df-ecd8-4f35-8207-
d0ccb4db3a90/attributes/name2"
      }
    }
  }
],
"num_records": 2,
"_links": {
  "self": {
    "href": "/api/storage/luns/4bc204df-ecd8-4f35-8207-
d0ccb4db3a90/attributes"
  }
}
}
```

This example uses the LUN REST endpoint with the `fields` query parameter to request the attributes properties.

```
# The API:
GET /api/storage/luns/{uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/storage/luns/4bc204df-ecd8-4f35-8207-
d0ccb4db3a90?fields=attributes' -H 'Accept: application/hal+json'

# The response:
{
"uuid": "4bc204df-ecd8-4f35-8207-d0ccb4db3a90",
"name": "/vol/vol1/lun1",
"attributes": [
  {
    "name": "name1",
    "value": "name1",
    "_links": {
      "self": {
        "href": "/api/storage/luns/4bc204df-ecd8-4f35-8207-
d0ccb4db3a90/attributes/name1"
      }
    }
  },
  {
    "name": "name2",
    "value": "value2",
    "_links": {
      "self": {
        "href": "/api/storage/luns/4bc204df-ecd8-4f35-8207-
d0ccb4db3a90/attributes/name2"
      }
    }
  }
],
"_links": {
  "self": {
    "href": "/api/storage/luns/4bc204df-ecd8-4f35-8207-d0ccb4db3a90"
  }
}
}
```

#### Adding an attribute to a LUN

```
# The API:
POST /api/storage/luns/{lun.uuid}/attributes

# The call:
curl -X POST 'https://<mgmt-ip>/api/storage/luns/4bc204df-ecd8-4f35-8207-
d0ccb4db3a90/attributes?return_records=true' -H 'Accept:
application/hal+json' -d '{ "name": "name1", "value": "value1" }'

# The response:
{
"num_records": 1,
"records": [
  {
    "name": "name1",
    "value": "value1",
    "_links": {
      "self": {
        "href": "/api/storage/luns/4bc204df-ecd8-4f35-8207-
d0ccb4db3a90/attributes/name1"
      }
    }
  }
]
}
```

#### Modifying an attribute value for a LUN

```
# The API
PATCH /api/storage/luns/{lun.uuid}/attributes/{name}

# The call:
curl -X PATCH 'https://<mgmt-ip>/api/storage/luns/4bc204df-ecd8-4f35-8207-
d0ccb4db3a90/attributes/name1' -H 'Accept: application/hal+json' -d '{
"value": "newValue" }'
```

#### Deleting an attribute from a LUN

```
# The API:
DELETE /api/storage/luns/{lun.uuid}/attributes/{name}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/storage/luns/4bc204df-ecd8-4f35-
8207-d0ccb4db3a90/attributes/name1' -H 'Accept: application/hal+json'
```

## Retrieve LUN attributes

GET `/storage/luns/{lun.uuid}/attributes`

**Introduced In:** 9.10

Retrieves LUN attributes.

### Learn more

- DOC /storage/luns/{lun.uuid}/attributes

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | The unique identifier of the LUN. |
| name | string | query | False | Filter by name<br>• maxLength: 4091<br>• minLength: 1 |
| value | string | query | False | Filter by value<br>• maxLength: 4091<br>• minLength: 1 |
| fields | array[string] | query | False | Specify the fields to return. |
| max_records | integer | query | False | Limit the number of records returned. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| return_records | boolean | query | False | The default is true for GET calls. When set to false, only the number of records is returned.<br>• 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.<br>• Default value: 15<br>• Max value: 120<br>• 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[lun_attribute] | |

### Example response


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "lun": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
      },
      "name": "name1",
      "value": "value1"
    }
  ]
}
```


### Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

lun

The LUN for which the attribute is set.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| uuid | string | The unique identifier of the LUN. |

lun_attribute

A name/value pair optionally stored with the LUN. Attributes are available to callers to persist small amounts of application-specific metadata. They are in no way interpreted by ONTAP.

Attribute names and values must be at least one byte and no more than 4091 bytes in length. The sum of the name and value lengths must be no more than 4092 bytes.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| lun | lun | The LUN for which the attribute is set. |
| name | string | The attribute name. Required in POST. |
| value | string | The attribute value. Required in POST; valid in PATCH. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 an attribute to a LUN

POST `/storage/luns/{lun.uuid}/attributes`

**Introduced In:** 9.10

Adds an attribute to a LUN.

### Required properties

- `name` - The name of the attribute to add.
- `value` - The value of the attribute to add.

### Learn more

- DOC /storage/luns/{lun.uuid}/attributes

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | The unique identifier of the LUN. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| return_timeout | integer | query | False | The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.<br>• Default value: 0<br>• Max value: 120<br>• Min value: 0 |
| return_records | boolean | query | False | The default is false. If set to true, the records are returned.<br>• Default value: |

**Request Body**

| Name | Type | Description |
|---|---|---|
| lun | lun | The LUN for which the attribute is set. |
| name | string | The attribute name. Required in POST. |

| Name | Type | Description |
|---|---|---|
| value | string | The attribute value. Required in POST; valid in PATCH. |

#### Example request

```
{
  "lun": {
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "name": "name1",
  "value": "value1"
}
```

### Response

```
Status: 202, Accepted
```

| Name | Type | Description |
|---|---|---|
| job | job_link | |

#### Example response

```
{
  "job": {
    "uuid": "string"
  }
}
```

#### Headers

| Name | Description | Type |
|---|---|---|
| Location | Useful for tracking the resource location | string |

### Response

```
Status: 201, Created
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374875 | The specified LUN was not found. |
| 5374928 | An incomplete attribute name/value pair was supplied. |
| 5374929 | The combined sizes of an attribute name and value are too large. |
| 5374930 | The attribute already exists for the LUN. |
| 5375060 | The copy start operation failed. |

Also see the table of common errors in the Response body overview section of this documentation.

### Definitions

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

lun

The LUN for which the attribute is set.

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the LUN. |

lun_attribute

A name/value pair optionally stored with the LUN. Attributes are available to callers to persist small amounts of application-specific metadata. They are in no way interpreted by ONTAP.

Attribute names and values must be at least one byte and no more than 4091 bytes in length. The sum of the name and value lengths must be no more than 4092 bytes.

| Name | Type | Description |
|---|---|---|
| lun | lun | The LUN for which the attribute is set. |
| name | string | The attribute name. Required in POST. |
| value | string | The attribute value. Required in POST; valid in PATCH. |

job_link

| Name | Type | Description |
|---|---|---|
| uuid | string | The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |

| Name | Type | Description |
|---|---|---|
| message | string | Message argument |

returned_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 a LUN attribute

DELETE `/storage/luns/{lun.uuid}/attributes/{name}`

**Introduced In:** 9.10

Deletes a LUN attribute.

### Learn more

- DOC /storage/luns/{lun.uuid}/attributes

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | |
| name | string | path | True | |

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374875 | The specified LUN was not found. |
| 5374931 | The specified attribute was not found. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 |

returned_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 a LUN attribute

GET `/storage/luns/{lun.uuid}/attributes/{name}`

**Introduced In:** 9.10

Retrieves a LUN attribute.

**Learn more**

- DOC /storage/luns/{lun.uuid}/attributes

**Parameters**

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | The unique identifier of the LUN. |
| name | string | path | True | The name of the attribute. |
| fields | array[string] | query | False | Specify the fields to return. |

### Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| lun | lun | The LUN for which the attribute is set. |
| name | string | The attribute name. Required in POST. |
| value | string | The attribute value. Required in POST; valid in PATCH. |

### Example response


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "lun": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "name": "name1",
  "value": "value1"
}
```


### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 5374875 | The specified LUN was not found. |
| 5374931 | The specified attribute was not found. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

lun

The LUN for which the attribute is set.

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| uuid | string | The unique identifier of the LUN. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 a LUN attribute value

PATCH `/storage/luns/{lun.uuid}/attributes/{name}`

**Introduced In:** 9.10

Updates a LUN attribute value.

### Learn more

- DOC /storage/luns/{lun.uuid}/attributes

### Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | The unique identifier of the LUN. |
| name | string | path | True | The name of the attribute. |

### Request Body

| Name | Type | Description |
|---|---|---|
| lun | lun | The LUN for which the attribute is set. |
| value | string | The attribute value. Required in POST; valid in PATCH. |

### Example request

```
{
  "lun": {
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "value": "value1"
}
```

### Response

```
Status: 200, Ok
```

### Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
| --- | --- |
| 5374875 | The specified LUN was not found. |
| 5374929 | The combined sizes of an attribute name and value are too large. |
| 5374931 | The specified attribute was not found. |
| 5375060 | The move start operation failed. |

Also see the table of common errors in the Response body overview section of this documentation.

**Definitions**

**See Definitions**

href

| Name | Type | Description |
|---|---|---|
| href | string | |

_links

lun

The LUN for which the attribute is set.

| Name | Type | Description |
|---|---|---|
| uuid | string | The unique identifier of the LUN. |

lun_attribute

A name/value pair optionally stored with the LUN. Attributes are available to callers to persist small amounts of application-specific metadata. They are in no way interpreted by ONTAP.

Attribute names and values must be at least one byte and no more than 4091 bytes in length. The sum of the name and value lengths must be no more than 4092 bytes.

| Name | Type | Description |
|---|---|---|
| lun | lun | The LUN for which the attribute is set. |
| value | string | The attribute value. Required in POST; valid in PATCH. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_error

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

| Name | Type | Description |
|---|---|---|
| message | string | Error message |
| target | string | The target parameter that caused the error. |

# Retrieve historical performance metrics for a LUN

GET `/storage/luns/{lun.uuid}/metrics`

**Introduced In:** 9.14

Retrieves historical space and performance metrics for a LUN.

## Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | The unique identifier of the LUN. |

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| interval | string | query | False | The time range for the data. Examples can be 1h, 1d, 1m, 1w, 1y. The period for each time range is as follows:<br>• 1h: Metrics over the most recent hour sampled over 15 seconds.<br>• 1d: Metrics over the most recent day sampled over 5 minutes.<br>• 1w: Metrics over the most recent week sampled over 30 minutes.<br>• 1m: Metrics over the most recent month sampled over 2 hours.<br>• 1y: Metrics over the most recent year sampled over a day.<br>• Default value: 1<br>• enum: ["1h", "1d", "1w", "1m", "1y"] |
| throughput.other | integer | query | False | Filter by throughput.other |
| throughput.total | integer | query | False | Filter by throughput.total |
| throughput.read | integer | query | False | Filter by throughput.read |
| throughput.write | integer | query | False | Filter by throughput.write |
| status | string | query | False | Filter by status |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| uuid | string | query | False | Filter by uuid |
| iops.other | integer | query | False | Filter by iops.other |
| iops.total | integer | query | False | Filter by iops.total |
| iops.read | integer | query | False | Filter by iops.read |
| iops.write | integer | query | False | Filter by iops.write |
| timestamp | string | query | False | Filter by timestamp |
| duration | string | query | False | Filter by duration |
| latency.other | integer | query | False | Filter by latency.other |
| latency.total | integer | query | False | Filter by latency.total |
| latency.read | integer | query | False | Filter by latency.read |
| latency.write | integer | query | False | Filter by latency.write |
| 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.<br>• Default value: 15<br>• Max value: 120<br>• Min value: 0 |
| fields | array[string] | query | False | Specify the fields to return. |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| max_records | integer | query | False | Limit the number of records returned. |
| order_by | array[string] | query | False | Order results by specified fields and optional [asc |
| desc] direction. Default direction is 'asc' for ascending. | return_records | boolean | query | False |

## Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| num_records | integer | Number of records |
| records | array[records] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "duration": "PT15S",
      "iops": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "latency": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "status": "ok",
      "throughput": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "timestamp": "2017-01-25 06:20:13 -0500",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    }
  ]
}
```

## Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 8585947 | No metrics are available for the requested object. |
| 8586225 | An unexpected error occurred retrieving metrics for the requested object. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
|---|---|---|
| error | returned_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 | |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

records

Performance numbers, such as IOPS latency and throughput.

| Name | Type | Description |
|---|---|---|
| _links | _links | |

| Name | Type | Description |
| --- | --- | --- |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |
| status | string | Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |
| timestamp | string | The timestamp of the performance data. |
| uuid | string | The unique identifier of the LUN. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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 historical performance metrics for a LUN for a specific time

GET `/storage/luns/{lun.uuid}/metrics/{timestamp}`

**Introduced In:** 9.14

Retrieves historical space and performance metrics for a LUN for a specific time.

## Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| lun.uuid | string | path | True | The unique identifier of the LUN. |
| timestamp | string | path | True | The timestamp of the space and performance data.<br>• format: date-time |
| fields | array[string] | query | False | Specify the fields to return. |

## Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| duration | string | The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations: |
| iops | iops | The rate of I/O operations observed at the storage object. |
| latency | latency | The round trip latency in microseconds observed at the storage object. |
| status | string | Errors associated with the sample. For example, if the aggregation of data over multiple nodes fails, then any partial errors might return "ok" on success or "error" on an internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_ delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data. |
| throughput | throughput | The rate of throughput bytes per second observed at the storage object. |

| Name | Type | Description |
|---|---|---|
| timestamp | string | The timestamp of the performance data. |
| uuid | string | The unique identifier of the LUN. |

**Example response**


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "duration": "PT15S",
  "iops": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "latency": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "throughput": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "timestamp": "2017-01-25 06:20:13 -0500",
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
}
```


## Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
| --- | --- |
| 8585947 | No metrics are available for the requested object. |
| 8586225 | An unexpected error occurred retrieving metrics for the requested object. |

Also see the table of common errors in the Response body overview section of this documentation.

| Name | Type | Description |
| --- | --- | --- |
| error | returned_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 | |

iops

The rate of I/O operations observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

latency

The round trip latency in microseconds observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |

| Name | Type | Description |
|---|---|---|
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

throughput

The rate of throughput bytes per second observed at the storage object.

| Name | Type | Description |
|---|---|---|
| other | integer | Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on. |
| read | integer | Performance metric for read I/O operations. |
| total | integer | Performance metric aggregated over all types of I/O operations. |
| write | integer | Performance metric for write I/O operations. |

error_arguments

| Name | Type | Description |
|---|---|---|
| code | string | Argument code |
| message | string | Message argument |

returned_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. |

**Copyright information**

Copyright © 2026 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

**Trademark information**

NETAPP, the NETAPP logo, and the marks listed at http://www.netapp.com/TM are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.