# Manage SAN igroups

## REST API reference

NetApp
February 06, 2026

This PDF was generated from https://docs.netapp.com/us-en/ontap-restapi/protocols_san_igroups_endpoint_overview.html on February 06, 2026. Always check docs.netapp.com for the latest.

# Table of Contents

# 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 |
|---|---|---|---|---|
| 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 |
| target.firmware_revision | string | query | False | Filter by target.firmware_revision<br>• Introduced in: 9.11 |

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

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| 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 |
| protocol | string | query | False | Filter by protocol |
| uuid | string | query | False | Filter by uuid |
| comment | string | query | False | Filter by comment<br>• Introduced in: 9.9<br>• maxLength: 254<br>• minLength: 0 |
| connectivity_tracking.connection_state | string | query | False | Filter by connectivity_tracking.connection_state<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.required_nodes.uuid | string | query | False | Filter by connectivity_tracking.required_nodes.uuid<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.arguments.code | string | query | False | Filter by connectivity_tracking.alerts.summary.arguments.code<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 |
| connectivity_tracking.alerts.summary.message | string | query | False | Filter by connectivity_tracking.alerts.summary.message<br>• Introduced in: 9.11 |
| initiators.proximity.peer_svms.name | string | query | False | Filter by initiators.proximity.peer_svms.name<br>• Introduced in: 9.15 |

| 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.local_svm | boolean | query | False | Filter by initiators.proximity.local_svm<br>• Introduced in: 9.15 |
| initiators.name | string | query | False | Filter by initiators.name<br>• maxLength: 96<br>• minLength: 1 |
| initiators.comment | string | query | False | Filter by initiators.comment<br>• Introduced in: 9.9<br>• maxLength: 254<br>• minLength: 0 |
| initiators.connectivity_tracking.connection_state | string | query | False | Filter by initiators.connectivity_tracking.connection_state<br>• Introduced in: 9.11 |
| initiators.igroup.name | string | query | False | Filter by initiators.igroup.name<br>• Introduced in: 9.9<br>• maxLength: 96<br>• minLength: 1 |
| initiators.igroup.uuid | string | query | False | Filter by initiators.igroup.uuid |

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

| Name | Type | In | Required | Description |
| --- | --- | --- | --- | --- |
| parent_igroups.comment | string | query | False | Filter by parent_igroups.comment<br>• Introduced in: 9.9<br>• maxLength: 254<br>• minLength: 0 |
| parent_igroups.name | string | query | False | Filter by parent_igroups.name<br>• Introduced in: 9.9<br>• maxLength: 96<br>• minLength: 1 |
| parent_igroups.uuid | string | query | False | Filter by parent_igroups.uuid<br>• Introduced in: 9.9 |
| fields | array[string] | query | False | Specify the fields to return. |
| max_records | integer | query | False | Limit the number of records returned. |
| return_records | boolean | query | False | The default is true for GET calls. When set to false, only the number of records is returned.<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[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 |

| 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[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. |

| Error Code | Description |
| --- | --- |
| 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 05:19:00 +0000"
              }
            ],
            "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 05:19:00 +0000"
          }
        ],
        "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 05:19:00 +0000"
              }
            ],
            "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 05:19:00 +0000"
          }
        ],
        "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. |
| 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 05:19:00 +0000"
          }
        ],
        "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>If the requested igroup is part of a remote, non-local, MetroCluster SVM, the LUN maps are not retrieved.<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. |

## 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.