Manage SAN vVol bindings

REST API reference

NetApp
September 12, 2025

This PDF was generated from https://docs.netapp.com/us-en/ontap-restapi-9101/protocols_san_vvol-bindings_endpoint_overview.html on September 12, 2025. Always check docs.netapp.com for the latest.

Table of Contents

Manage SAN vVol bindings . . . 1
- Protocols SAN vvol-bindings endpoint overview . . . 1
  - Overview . . . 1
  - Examples . . . 1
- Retrieve vVol bindings . . . 4
  - Related ONTAP commands . . . 5
  - Parameters . . . 5
  - Response . . . 6
  - Error . . . 8
  - Definitions . . . 8
- Create a vVol binding . . . 12
  - Required properties . . . 12
  - Related ONTAP commands . . . 12
  - Learn more . . . 12
  - Parameters . . . 12
  - Request Body . . . 12
  - Response . . . 14
  - Error . . . 15
  - Definitions . . . 16
- Delete a vVol binding . . . 19
  - Related ONTAP commands . . . 19
  - Learn more . . . 19
  - Parameters . . . 20
  - Response . . . 20
  - Error . . . 20
  - Definitions . . . 21
- Retrieve a vVol binding . . . 22
  - Related ONTAP commands . . . 22
  - Learn more . . . 22
  - Parameters . . . 22
  - Response . . . 22
  - Error . . . 24
  - Definitions . . . 25

Manage SAN vVol bindings

Protocols SAN vvol-bindings endpoint overview

Overview

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

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

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

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

Examples

Creating a vVol binding

```
# The API:
POST /api/protocols/san/vvol-bindings

# The call:
curl -X POST 'https://<mgmt-ip>/api/protocols/san/vvol-bindings' -H
'accept: application/hal+json' -d '{ "svm": { "name": "svm1" },
"protocol_endpoint": { "name": "/vol/vol1/pe1" }, "vvol" : { "name":
"/vol/vol1/vvol1" } }'
```

Retrieving all vVol bindings

```
# The API:
GET /api/protocols/san/vvol-bindings

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

# The response:
```

```
{
"records": [
  {
    "protocol_endpoint": {
      "uuid": "2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4",
      "name": "/vol/vol1/pe1",
      "_links": {
        "self": {
          "href": "/api/storage/luns/2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4"
        }
      }
    },
    "vvol": {
      "uuid": "28c02623-42fa-4f5f-a984-a02044bfc005",
      "name": "/vol/vol1/vvol1",
      "_links": {
        "self": {
          "href": "/api/storage/luns/28c02623-42fa-4f5f-a984-a02044bfc005"
        }
      }
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/vvol-bindings/2aab57f3-dc5d-491e-80d2-
15c7ed5dd5c4/28c02623-42fa-4f5f-a984-a02044bfc005"
      }
    }
  },
  {
    "protocol_endpoint": {
      "uuid": "2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4",
      "name": "/vol/vol1/pe1",
      "_links": {
        "self": {
          "href": "/api/storage/luns/2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4"
        }
      }
    },
    "vvol": {
      "uuid": "a8d4ba93-918f-40ad-a1e4-4d7b244bdcdf",
      "name": "/vol/vol1/vvol2",
      "_links": {
        "self": {
          "href": "/api/storage/luns/a8d4ba93-918f-40ad-a1e4-4d7b244bdcdf"
        }
      }
```

```
    },
    "_links": {
      "self": {
        "href": "/api/protocols/san/vvol-bindings/2aab57f3-dc5d-491e-80d2-
15c7ed5dd5c4/a8d4ba93-918f-40ad-a1e4-4d7b244bdcdf"
      }
    }
  }
],
"num_records": 2,
"_links": {
  "self": {
    "href": "/api/protocols/san/vvol-bindings"
  }
}
}
```

Retrieving a specific vVol binding

```
# The API:
GET /api/protocols/san/vvol-bindings/{protocol_endpoint.uuid}/{vvol.uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/san/vvol-bindings/2aab57f3-
dc5d-491e-80d2-15c7ed5dd5c4/28c02623-42fa-4f5f-a984-a02044bfc005' -H
'accept: application/hal+json'

# The response:
{
"protocol_endpoint": {
  "uuid": "2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4",
  "name": "/vol/vol1/pe1",
  "_links": {
    "self": {
      "href": "/api/storage/luns/2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4"
    }
  }
},
"vvol": {
  "uuid": "28c02623-42fa-4f5f-a984-a02044bfc005",
  "name": "/vol/vol1/vvol1",
  "_links": {
    "self": {
      "href": "/api/storage/luns/28c02623-42fa-4f5f-a984-a02044bfc005"
```

```
    }
  }
},
"svm": {
  "uuid": "bf295ccc-a6bb-11eb-93e8-005056bb470f",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/bf295ccc-a6bb-11eb-93e8-005056bb470f"
    }
  }
},
"id": 2411392,
"is_optimal": true,
"count": 1,
"_links": {
  "self": {
    "href": "/api/protocols/san/vvol-bindings/2aab57f3-dc5d-491e-80d2-
15c7ed5dd5c4/28c02623-42fa-4f5f-a984-a02044bfc005"
  }
}
}
```


Deleting a vVol binding

```
# The API:
DELETE /api/protocols/san/vvol-
bindings/{protocol_endpoint.uuid}/{vvol.uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/san/vvol-
bindings/2aab57f3-dc5d-491e-80d2-15c7ed5dd5c4/28c02623-42fa-4f5f-a984-
a02044bfc005' -H 'accept: application/hal+json'
```

Retrieve vVol bindings

GET `/protocols/san/vvol-bindings`

Introduced In: 9.10

Retrieves vVol bindings.

Related ONTAP commands

- `lun bind show`
- DOC /protocols/san/vvol-bindings

Parameters

Name	Type	In	Required	Description
svm.uuid	string	query	False	Filter by svm.uuid
svm.name	string	query	False	Filter by svm.name
count	integer	query	False	Filter by count
id	integer	query	False	Filter by id
vvol.uuid	string	query	False	Filter by vvol.uuid
vvol.name	string	query	False	Filter by vvol.name
protocol_endpoint.uuid	string	query	False	Filter by protocol_endpoint.uuid
protocol_endpoint.name	string	query	False	Filter by protocol_endpoint.name
is_optimal	boolean	query	False	Filter by is_optimal
fields	array[string]	query	False	Specify the fields to return.
max_records	integer	query	False	Limit the number of records returned.
return_records	boolean	query	False	The default is true for GET calls. When set to false, only the number of records is returned. • Default value: 1

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. • Default value: 1 • Max value: 120 • Min value: 0
order_by	array[string]	query	False	Order results by specified fields and optional [asc

Response

```
Status: 200, Ok
```

Name	Type	Description
_links	_links	
num_records	integer	Number of records.
records	array[vvol_binding]	

Example response


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "count": 1,
      "id": 1,
      "is_optimal": 1,
      "protocol_endpoint": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "/vol/volume1/lun1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "svm": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "vvol": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "/vol/volume1/lun1",
```

```
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      }
    }
  ]
}
```


Error

```
Status: Default, Error
```

Name	Type	Description
error	error	

Example error


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


Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
next	href	
self	href	

_links

Name	Type	Description
self	href	

protocol_endpoint

The class `protocol_endpoint` LUN in the vVol binding. Required in POST.

Name	Type	Description
_links	_links	
name	string	The fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the (optional) qtree name, and base name of the LUN. Valid in POST and PATCH.
uuid	string	The unique identifier of the LUN.

svm

The SVM in which the vVol binding and its LUNs are located. Required in POST.

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

vvol

The class `vvol` LUN in the vVol binding. Required in POST.

Name	Type	Description
_links	_links	
name	string	The fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the (optional) qtree name, and base name of the LUN. Valid in POST and PATCH.
uuid	string	The unique identifier of the LUN.

vvol_binding

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

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

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

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

Name	Type	Description
id	integer	The identifier assigned to the vVol binding. The bind identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.
is_optimal	boolean	Indicates if the class `procotol_endpoint` LUN and the class `vvol` LUN are on the same cluster node.
protocol_endpoint	protocol_endpoint	The class `protocol_endpoint` LUN in the vVol binding. Required in POST.
svm	svm	The SVM in which the vVol binding and its LUNs are located. Required in POST.
vvol	vvol	The class `vvol` LUN in the vVol binding. Required in POST.

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

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

Create a vVol binding

POST `/protocols/san/vvol-bindings`

Introduced In: 9.10

Creates a vVol binding. The binding between any two specific LUNs is reference counted. When a binding is created that already exists, the binding count is incremented.

Required properties

- `svm.uuid` or `svm.name` - Existing SVM in which to create the vVol binding.
- `protocol_endpoint.uuid` or `protocol_endpoint.name` - Existing class `protocol_endpoint` LUN to bind to the specified class `vvol` LUN.
- `vvol.uuid` or `vvol.name` - Existing class `vvol` LUN to bind to the specified class `protocol_endpoint` LUN.

Related ONTAP commands

- `lun bind create`

Learn more

- DOC /protocols/san/vvol-bindings

Parameters

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

Request Body

Name	Type	Description
count	integer	The vVol binding between any two specific LUNs is reference counted. When a REST POST is executed for a vVol binding that already exists, the vVol binding reference count is incremented. When a REST DELETE is executed, the vVol binding reference count is decremented. Only when the vVol binding count reaches zero, or the query parameter `delete_all_references` is supplied, is the vVol binding destroyed.
id	integer	The identifier assigned to the vVol binding. The bind identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.
is_optimal	boolean	Indicates if the class `procotol_endpoint` LUN and the class `vvol` LUN are on the same cluster node.
protocol_endpoint	protocol_endpoint	The class `protocol_endpoint` LUN in the vVol binding. Required in POST.
svm	svm	The SVM in which the vVol binding and its LUNs are located. Required in POST.
vvol	vvol	The class `vvol` LUN in the vVol binding. Required in POST.

Example request


```
{
  "count": 1,
  "id": 1,
  "is_optimal": 1,
  "protocol_endpoint": {
    "name": "/vol/volume1/lun1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "svm": {
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "vvol": {
    "name": "/vol/volume1/lun1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  }
}
```


Response

```
Status: 201, Created
```

Name	Type	Description
num_records	integer	Number of records.
records	array[vvol_binding]	

Example response


```
{
  "records": [
    {
      "count": 1,
      "id": 1,
      "is_optimal": 1,
      "protocol_endpoint": {
        "name": "/vol/volume1/lun1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "svm": {
        "name": "svm1",
        "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
      },
      "vvol": {
        "name": "/vol/volume1/lun1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      }
    }
  ]
}
```


Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
2621462	The specified SVM does not exist or is not accessible to the caller.
2621706	Both the SVM UUID and SVM name were supplied, but don't refer to the same SVM.
2621707	No SVM was specified. Either `svm.name` or `svm.uuid` must be supplied.
5374238	A LUN in a Snapshot copy was specified.
5374323	The LUN specified as the protocol endpoint LUN is not of class `protocol_endpoint`.
5374325	The LUN specified as the vVol LUN is not of class `vvol`.

Error Code	Description
5374874	The UUID and name supplied for the protocol endpoint of Vvol LUN do not refer to the same LUN. Use to the `target` property of the error object to differentiate between the protocol endpoint LUN and the vVol LUN.
5374875	The protocol endpoint or vVol LUN was not found or is not accessible to the caller. Use to the `target` property of the error object to differentiate between the protocol endpoint LUN and the vVol LUN.
5374876	The protocol endpoint or vVol LUN was not found in the SVM. Use to the `target` property of the error object to differentiate between the protocol endpoint LUN and the vVol LUN.
5374924	No protocol endpoint LUN was supplied.
5374925	No vVol LUN was supplied.

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

protocol_endpoint

The class `protocol_endpoint` LUN in the vVol binding. Required in POST.

Name	Type	Description
name	string	The fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the (optional) qtree name, and base name of the LUN. Valid in POST and PATCH.
uuid	string	The unique identifier of the LUN.

svm

The SVM in which the vVol binding and its LUNs are located. Required in POST.

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

vvol

The class `vvol` LUN in the vVol binding. Required in POST.

Name	Type	Description
name	string	The fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the (optional) qtree name, and base name of the LUN. Valid in POST and PATCH.
uuid	string	The unique identifier of the LUN.

vvol_binding

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

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

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

Name	Type	Description
count	integer	The vVol binding between any two specific LUNs is reference counted. When a REST POST is executed for a vVol binding that already exists, the vVol binding reference count is incremented. When a REST DELETE is executed, the vVol binding reference count is decremented. Only when the vVol binding count reaches zero, or the query parameter `delete_all_references` is supplied, is the vVol binding destroyed.
id	integer	The identifier assigned to the vVol binding. The bind identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.
is_optimal	boolean	Indicates if the class `procotol_endpoint` LUN and the class `vvol` LUN are on the same cluster node.
protocol_endpoint	protocol_endpoint	The class `protocol_endpoint` LUN in the vVol binding. Required in POST.

Name	Type	Description
svm	svm	The SVM in which the vVol binding and its LUNs are located. Required in POST.
vvol	vvol	The class `vvol` LUN in the vVol binding. Required in POST.

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

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

Delete a vVol binding

DELETE `/protocols/san/vvol-bindings/{protocol_endpoint.uuid}/{vvol.uuid}`

Introduced In: 9.10

Deletes a vVol binding. The binding between any two specific LUNs is reference counted. When a binding is deleted, the binding count is decremented, but the LUNs remain bound if the resultant reference count is greater than zero. When the binding count reaches zero, the binding is destroyed.

Related ONTAP commands

- `lun bind destroy`

Learn more

- DOC /protocols/san/vvol-bindings

Parameters

Name	Type	In	Required	Description
protocol_endpoint.uuid	string	path	True	The unique identifier of the class `protocol_endpoint` LUN.
vvol.uuid	string	path	True	The unique identifier of the class `vvol` LUN.
delete_all_references	boolean	query	False	Forces deletion of the binding regardless of the reference count value. • Default value:

Response

```
Status: 200, Ok
```

Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
5374875	The vVol binding was not found because the protocol endpoint or vVol LUN was not found. Use to the `target` property of the error object to differentiate between the protocol endpoint LUN and the vVol LUN.
5374926	The vVol binding was not found.

Name	Type	Description
error	error	

Example error

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

Definitions

See Definitions

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

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

Retrieve a vVol binding

GET `/protocols/san/vvol-bindings/{protocol_endpoint.uuid}/{vvol.uuid}`

Introduced In: 9.10

Retrieves a vVol binding.

Related ONTAP commands

- `lun bind show`

Learn more

- DOC /protocols/san/vvol-bindings

Parameters

Name	Type	In	Required	Description
protocol_endpoint.uuid	string	path	True	The unique identifier of the class `protocol_endpoint` LUN.
vvol.uuid	string	path	True	The unique identifier of the class `vvol` LUN.
fields	array[string]	query	False	Specify the fields to return.

Response

```
Status: 200, Ok
```

Name	Type	Description
_links	_links	

Name	Type	Description
count	integer	The vVol binding between any two specific LUNs is reference counted. When a REST POST is executed for a vVol binding that already exists, the vVol binding reference count is incremented. When a REST DELETE is executed, the vVol binding reference count is decremented. Only when the vVol binding count reaches zero, or the query parameter `delete_all_references` is supplied, is the vVol binding destroyed.
id	integer	The identifier assigned to the vVol binding. The bind identifier is unique amongst all class `vvol` LUNs bound to the same class `protocol_endpoint` LUN.
is_optimal	boolean	Indicates if the class `procotol_endpoint` LUN and the class `vvol` LUN are on the same cluster node.
protocol_endpoint	protocol_endpoint	The class `protocol_endpoint` LUN in the vVol binding. Required in POST.
svm	svm	The SVM in which the vVol binding and its LUNs are located. Required in POST.
vvol	vvol	The class `vvol` LUN in the vVol binding. Required in POST.

Example response


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "count": 1,
  "id": 1,
  "is_optimal": 1,
  "protocol_endpoint": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "/vol/volume1/lun1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "vvol": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "/vol/volume1/lun1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  }
}
```


Error

```
Status: Default
```

ONTAP Error Response Codes

Error Code	Description
4	The vVol binding was not found.

Name	Type	Description
error	error	

Example error

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

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
self	href	

protocol_endpoint

The class `protocol_endpoint` LUN in the vVol binding. Required in POST.

Name	Type	Description
_links	_links	
name	string	The fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the (optional) qtree name, and base name of the LUN. Valid in POST and PATCH.
uuid	string	The unique identifier of the LUN.

svm

The SVM in which the vVol binding and its LUNs are located. Required in POST.

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

vvol

The class `vvol` LUN in the vVol binding. Required in POST.

Name	Type	Description
_links	_links	

Name	Type	Description
name	string	The fully qualified path name of the LUN composed of the "/vol" prefix, the volume name, the (optional) qtree name, and base name of the LUN. Valid in POST and PATCH.
uuid	string	The unique identifier of the LUN.

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

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

Copyright information

Copyright © 2025 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.