



# **Manage SnapMirror relationships**

## **REST API reference**

NetApp

February 11, 2026

This PDF was generated from [https://docs.netapp.com/us-en/ontap-restapi/snapmirror\\_relationships\\_endpoint\\_overview.html](https://docs.netapp.com/us-en/ontap-restapi/snapmirror_relationships_endpoint_overview.html) on February 11, 2026. Always check [docs.netapp.com](https://docs.netapp.com) for the latest.

# Table of Contents

Manage SnapMirror relationships	1
Manage SnapMirror relationships	1
Overview	1
Retrieve information for SnapMirror relationships	1
Related ONTAP commands	2
Expensive properties	2
Examples	2
Learn more	3
Parameters	3
Response	10
Error	14
Definitions	15
Create a SnapMirror relationship	35
Required properties	35
Recommended optional properties	35
Default property values	36
Related ONTAP commands	36
Important notes	36
Examples	36
Learn more	40
Parameters	40
Request Body	41
Response	51
Response	52
Error	52
Definitions	58
Delete a SnapMirror relationship	79
Important notes	79
Related ONTAP commands	80
Examples	80
Learn more	81
Parameters	81
Response	83
Response	84
Error	84
Definitions	85
Retrieve a SnapMirror relationship	86
Related ONTAP commands	87
Expensive properties	87
Example	87
Learn more	87
Parameters	87
Response	87

Error .....	97
Definitions .....	98
Update a SnapMirror relationship .....	111
Examples .....	112
Related ONTAP commands .....	112
Important notes .....	113
Examples .....	113
Learn more .....	116
Parameters .....	117
Request Body .....	119
Response .....	127
Response .....	128
Error .....	128
Definitions .....	131

# Manage SnapMirror relationships

## Manage SnapMirror relationships

### Overview

This API manages asynchronous extended data protection (XDP) relationships for FlexVol volumes, FlexGroup volumes, or SVMs. It is also used to manage a synchronous relationship between FlexVol volumes, which provides zero RPO data protection and active synchronous relationship with automated failover between Application Consistency Group endpoints which provides zero RTO data protection.

To create an asynchronous extended data protection relationship with FlexVol volumes, FlexGroup volumes, Application Consistency Groups or SVMs, use the policy of type "async".

To create a synchronous relationship between FlexVol volumes, use the policy of type "sync" with sync\_type of either "sync" or "strict\_sync". To create an active synchronous relationship with automated failover between Application Consistency Group endpoints, use the policy of type "sync" with sync\_type "automated\_failover". You can create an asynchronous extended data protection relationship between the source and destination which can be used by the transfer APIs to perform SnapMirror "restore" operations.

To create FlexVol volume or FlexGroup volume SnapMirror relationships, the source volume must be in the "online" state and be a read\_write type; the destination volume must be in the "online" state and be a data protection type.

In the case of an asynchronous or synchronous SnapMirror relationship for an Application Consistency Group of FlexVol volumes, SnapMirror creation results in the creation of an Application Consistency Group on the source cluster if it did not already exist with the exact same name and set of FlexVol volumes specified in the current operation. Additionally, if the specified Application Consistency Group is already present and is already a part of an existing SnapMirror relationship, the process fails. Creating an Application Consistency Group on the destination cluster is part of the SnapMirror creation workflow.

To create SnapMirror relationships between SVMs, the source SVM must be of subtype "default" and the destination SVM of subtype "dp\_destination". Additionally, SVMs must be peered before a relationship can be established between them when the "create\_destination" property is not specified. When the "create\_destination" property is specified, the destination SVM is provisioned on the destination cluster and the SVM peer relationship is established between the source SVM and the new destination SVM, provided that the source SVM has SVM peering permissions for the destination cluster.

Data protection FlexVol volume SnapMirror relationships cannot be created using this API but existing relationships can be listed or managed.

The SnapMirror functionality is subdivided into relationship APIs and transfer APIs:

- SnapMirror relationship APIs are used to create and manage the SnapMirror relationships.
- SnapMirror transfer APIs are used to manage data transfers.

## Retrieve information for SnapMirror relationships

GET /snapmirror/relationships

Introduced In: 9.6

Retrieves information for SnapMirror relationships whose destination endpoints are in the current SVM or the current cluster, depending on the cluster context.

## Related ONTAP commands

- `snapmirror show`
- `snapmirror list-destinations`

## Expensive properties

- `source.consistency_group_volumes.name`
- `destination.consistency_group_volumes.name`
- `svmdr_volumes.name`

## Examples

The following examples show how to retrieve the list of SnapMirror relationships and the list of SnapMirror destinations.

1. Retrieving the list of SnapMirror relationships. This API must be run on the cluster containing the destination endpoint.

```
GET "/api/snapmirror/relationships/"
```

1. Retrieving the list of SnapMirror destinations on source. This must be run on the cluster containing the source endpoint.

```
GET "/api/snapmirror/relationships/?list_destinations_only=true"
```

1. Retrieving the relationship UUID of SnapMirror relationships with lag time greater than 2 days. This API must be run on the cluster containing the destination endpoint.

```
GET "/api/snapmirror/relationships/?fields=uuid&lag_time=>P2DT"
```

1. Retrieving the list of SnapMirror relationships with lag time less than 10 hours. This API must be run on the cluster containing the destination endpoint.

```
GET "/api/snapmirror/relationships/?lag_time=<PT10H"
```

1. Retrieving the list of constituent volumes for SVM DR Snapmirror relationships.

```
GET "/api/snapmirror/relationships/?fields=svmdr_volumes.name"
```

</private>

## Learn more

- [DOC /snapmirror/relationships](#)

## Parameters

Name	Type	In	Required	Description
list_destinations_only	boolean	query	False	Set to true to show relationships from the source only.
transfer.end_time	string	query	False	Filter by transfer.end_time <ul style="list-style-type: none"><li>• Introduced in: 9.9</li></ul>
transfer.uuid	string	query	False	Filter by transfer.uuid
transfer.bytes_transferred	integer	query	False	Filter by transfer.bytes_transferred
transfer.state	string	query	False	Filter by transfer.state
transfer.total_duration	string	query	False	Filter by transfer.total_duration <ul style="list-style-type: none"><li>• Introduced in: 9.9</li></ul>
transfer.type	string	query	False	Filter by transfer.type <ul style="list-style-type: none"><li>• Introduced in: 9.14</li></ul>
transfer.last_updated_time	string	query	False	Filter by transfer.last_updated_time <ul style="list-style-type: none"><li>• Introduced in: 9.14</li></ul>

Name	Type	In	Required	Description
state	string	query	False	Filter by state
restore	boolean	query	False	Filter by restore
policy.name	string	query	False	Filter by policy.name
policy.type	string	query	False	Filter by policy.type
policy.uuid	string	query	False	Filter by policy.uuid
source.cluster.name	string	query	False	Filter by source.cluster.name  • Introduced in: 9.7
source.cluster.uuid	string	query	False	Filter by source.cluster.uuid  • Introduced in: 9.7
source.consistency_group_volumes.name	string	query	False	Filter by source.consistency_group_volumes.name  • Introduced in: 9.8
source.path	string	query	False	Filter by source.path
source.luns.uuid	string	query	False	Filter by source.luns.uuid  • Introduced in: 9.16
source.luns.name	string	query	False	Filter by source.luns.name  • Introduced in: 9.16
source.svm.name	string	query	False	Filter by source.svm.name

Name	Type	In	Required	Description
source.svm.uuid	string	query	False	Filter by source.svm.uuid
lag_time	string	query	False	Filter by lag_time
consistency_group_failover.error.arguments.message	string	query	False	Filter by consistency_group_failover.error.arguments.message  • Introduced in: 9.8
consistency_group_failover.error.arguments.code	string	query	False	Filter by consistency_group_failover.error.arguments.code  • Introduced in: 9.8
consistency_group_failover.error.code	string	query	False	Filter by consistency_group_failover.error.code  • Introduced in: 9.8
consistency_group_failover.error.message	string	query	False	Filter by consistency_group_failover.error.message  • Introduced in: 9.8
consistency_group_failover.state	string	query	False	Filter by consistency_group_failover.state  • Introduced in: 9.14



Name	Type	In	Required	Description
consistency_group_failover.status.message	string	query	False	Filter by consistency_group_failover.status.message  • Introduced in: 9.8
consistency_group_failover.status.code	string	query	False	Filter by consistency_group_failover.status.code  • Introduced in: 9.8
consistency_group_failover.type	string	query	False	Filter by consistency_group_failover.type  • Introduced in: 9.14
group_type	string	query	False	Filter by group_type  • Introduced in: 9.11
master_bias_activated_site	string	query	False	Filter by master_bias_activated_site  • Introduced in: 9.14
destination.luns.uuid	string	query	False	Filter by destination.luns.uuid  • Introduced in: 9.16
destination.luns.name	string	query	False	Filter by destination.luns.name  • Introduced in: 9.16

Name	Type	In	Required	Description
destination.svm.name	string	query	False	Filter by destination.svm.name
destination.svm.uuid	string	query	False	Filter by destination.svm.uuid
destination.consistency_group_volumes.name	string	query	False	Filter by destination.consistency_group_volumes.name  • Introduced in: 9.8
destination.cluster.name	string	query	False	Filter by destination.cluster.name  • Introduced in: 9.7
destination.cluster.uuid	string	query	False	Filter by destination.cluster.uuid  • Introduced in: 9.7
destination.path	string	query	False	Filter by destination.path
io_serving_copy	string	query	False	Filter by io_serving_copy  • Introduced in: 9.14
throttle	integer	query	False	Filter by throttle  • Introduced in: 9.11
last_transfer_type	string	query	False	Filter by last_transfer_type  • Introduced in: 9.11

Name	Type	In	Required	Description
preferred_site	string	query	False	Filter by preferred_site  • Introduced in: 9.14
exported_snapshot	string	query	False	Filter by exported_snapshot
identity_preservation	string	query	False	Filter by identity_preservation  • Introduced in: 9.11
unhealthy_reason.message	string	query	False	Filter by unhealthy_reason.message
unhealthy_reason.arguments	string	query	False	Filter by unhealthy_reason.arguments  • Introduced in: 9.14
unhealthy_reason.code	string	query	False	Filter by unhealthy_reason.code
svmdr_volumes.name	string	query	False	Filter by svmdr_volumes.name  • Introduced in: 9.13
total_transfer_duration	string	query	False	Filter by total_transfer_duration  • Introduced in: 9.13
healthy	boolean	query	False	Filter by healthy
uuid	string	query	False	Filter by uuid

Name	Type	In	Required	Description
total_transfer_bytes	integer	query	False	Filter by total_transfer_bytes  • Introduced in: 9.13
backoff_level	string	query	False	Filter by backoff_level  • Introduced in: 9.14
transfer_schedule.name	string	query	False	Filter by transfer_schedule.name  • Introduced in: 9.11
transfer_schedule.uuid	string	query	False	Filter by transfer_schedule.uuid  • Introduced in: 9.11
last_transfer_network_compression_ratio	string	query	False	Filter by last_transfer_network_compression_ratio  • Introduced in: 9.13
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	<p>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.</p> <ul style="list-style-type: none"> <li>• Default value: 15</li> <li>• Max value: 120</li> <li>• Min value: 0</li> </ul>
order_by	array[string]	query	False	Order results by specified fields and optional [asc

## Response

Status: 200, Ok

Name	Type	Description
_links	<a href="#">_links</a>	
num_records	integer	Number of records
records	array[ <a href="#">snapmirror_relationship</a> ]	

## Example response

```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "backoff_level": "medium",
      "consistency_group_failover": {
        "error": {
          "arguments": [
            {
              "code": "string",
              "message": "string"
            }
          ],
          "code": "4",
          "message": "entry doesn't exist"
        },
        "state": "string",
        "status": {
          "code": "string",
          "message": "string"
        },
        "type": "string"
      },
      "destination": {
        "cluster": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "name": "cluster1",

```

```

    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "consistency_group_volumes": [
    {
      "name": "volume1"
    }
  ],
  "luns": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "/vol/volume1/lun1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "path": "svm1:volume1",
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "exported_snapshot": "string",
  "group_type": "consistency_group",
  "identity_preservation": "string",
  "io_serving_copy": "C1_sti85-vsim-ucs209a_cluster, C1_sti85-vsim-ucs209c_cluster",
  "lag_time": "PT8H35M42S",
  "last_transfer_network_compression_ratio": 61,
  "last_transfer_type": "initialize",
  "master_bias_activated_site": "C1_sti85-vsim-ucs209a_cluster",
  "policy": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "Asynchronous",
    "type": "string",
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },

```

```

"preferred_site": "C1_sti85-vsim-ucs209a_cluster",
"restore_to_snapshot": "string",
"source": {
  "cluster": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "cluster1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "consistency_group_volumes": [
    {
      "name": "volume1"
    }
  ],
  "luns": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "/vol/volume1/lun1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "path": "svm1:volume1",
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "state": "snapmirrored",
  "svmdr_volumes": [
    {
      "name": "volume1"
    }
  ],
  "throttle": 0,
  "total_transfer_bytes": 1098210312,
  "total_transfer_duration": "PT3M21S",

```



```

"transfer": {
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "end_time": "2020-12-02 18:36:19 -0800",
  "last_updated_time": "2023-09-14 15:39:19 -0700",
  "state": "string",
  "total_duration": "PT28M41S",
  "type": "initialize",
  "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
},
"transfer_schedule": {
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "name": "weekly",
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"unhealthy_reason": [
  {
    "arguments": [],
    "code": "6621444",
    "message": "Failed to complete update operation on one or
more item relationships."
  },
  {
    "arguments": [],
    "code": "6621445",
    "message": "Group Update failed"
  }
],
"uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
}
]
}

```

## Error

Status: Default

## ONTAP Error Response codes

Error code	Description
13303825	Could not retrieve information for the SnapMirror policy type
13303817	Unknown value for the Snapmirror State

Name	Type	Description
error	<a href="#">returned_error</a>	

### 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	<a href="#">href</a>	
self	<a href="#">href</a>	

\_links

Name	Type	Description
self	<a href="#">href</a>	

error\_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

SnapMirror Consistency Group failover error message.

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message

status

Name	Type	Description
code	string	Status code
message	string	SnapMirror Consistency Group failover status.

## snapmirror\_consistency\_group\_failover

SnapMirror Consistency Group failover information. The SnapMirror Consistency Group failover can be a planned or an unplanned operation. Only active SnapMirror Consistency Group failover operation progress can be monitored using this object. In case of an error during the failover operation, the property "consistency\_group\_failover.error" holds the reason for the error. ONTAP automatically retries any failed SnapMirror Consistency Group failover operation.

Name	Type	Description
error	<a href="#">error</a>	SnapMirror Consistency Group failover error message.
state	string	SnapMirror Consistency Group failover state.
status	<a href="#">status</a>	
type	string	SnapMirror Consistency Group failover type.

## bucket\_retention

Specifies the retention-mode and default retention period configured on the destination bucket.

Name	Type	Description
default_period	string	Specifies the default retention period that is applied to objects while committing them to the WORM state without an associated retention period. The retention period can be in years, or days. The retention period value represents a duration and must be specified in the ISO-8601 duration format. A period specified for years and days is represented in the ISO-8601 format as "P<num>Y" and "P<num>D" respectively, for example "P10Y" represents a duration of 10 years. The period string must contain only a single time element that is, either years, or days. A duration which combines different periods is not supported, for example "P1Y10D" is not supported.</num></num>

Name	Type	Description
mode	string	The lock mode of the bucket. compliance &dash; A SnapLock Compliance (SLC) bucket provides the highest level of WORM protection and an administrator cannot destroy a compliance bucket if it contains unexpired WORM objects. governance &dash; An administrator can delete a Governance bucket. no_lock &dash; Indicates the bucket does not support object locking.

#### storage\_service

Name	Type	Description
enabled	boolean	This property indicates whether to create the destination endpoint using storage service. This property is supported for Unified ONTAP destination endpoints only.  <ul style="list-style-type: none"> <li>• Introduced in: 9.7</li> <li>• x-nullable: true</li> </ul>
enforce_performance	boolean	Optional property to enforce storage service performance on the destination endpoint. This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. This property is supported for Unified ONTAP destination endpoints only.  <ul style="list-style-type: none"> <li>• Default value:</li> <li>• Introduced in: 9.7</li> <li>• x-nullable: true</li> </ul>

Name	Type	Description
name	string	<p>Optional property to specify the storage service name for the destination endpoint. This property is considered when the property "create_destination.storage_service.enabled" is set to "true". When the property "create_destination.storage_service.enabled" is set to "true" and the "create_destination.storage_service.name" for the endpoint is not specified, then ONTAP selects the highest storage service available on the cluster to provision the destination endpoint. This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>enum: ["extreme", "performance", "value"]</li> <li>Introduced in: 9.6</li> <li>x-nullable: true</li> </ul>

tiering



Name	Type	Description
policy	string	<p>Optional property to specify the destination endpoint's tiering policy when "create_destination.tiering.supported" is set to "true". This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. This property determines whether the user data blocks of the destination endpoint in a FabricPool will be tiered to the cloud store when they become cold. FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of the destination endpoint volume blocks increases if they are accessed frequently and decreases when they are not. all – This policy allows tiering of both destination endpoint snapshots and the user transferred data blocks to the cloud store as soon as possible by ignoring the temperature on the volume blocks. This tiering policy is not applicable for Consistency Group destination endpoints or for synchronous relationships. auto – This policy allows tiering of both destination endpoint snapshots and the active file system user data to the cloud store none – Destination endpoint volume blocks will not be tiered to the cloud store. snapshot_only – This policy allows tiering of only the destination endpoint volume snapshots not associated with the active file system. The default tiering policy is "snapshot_only" for a FlexVol volume and "none" for a FlexGroup volume. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>enum: ["all", "auto", "none", "snapshot_only"]</li> <li>Introduced in: 9.6</li> </ul> <p>x-nullable: true</p>



Name	Type	Description
supported	boolean	<p>Optional property to enable provisioning of the destination endpoint volumes on FabricPool aggregates. This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. Only FabricPool aggregates are used if this property is set to "true" and only non FabricPool aggregates are used if this property is set to "false". Tiering support for a FlexGroup volume can be changed by moving all of the constituents to the required aggregates. Note that in order to tier data, not only do the destination endpoint volumes need to support tiering by using FabricPools, the "create_destination.tiering.policy" must not be "none". A destination endpoint that uses FabricPools but has a tiering "policy" of "none" supports tiering but will not tier any data. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.6</li> <li>• x-nullable: true</li> </ul>

#### snapmirror\_destination\_creation

Use this object to provision the destination endpoint when establishing a SnapMirror relationship for a FlexVol volume, FlexGroup volume, SVM, Consistency Group or ONTAP S3 Bucket. Given a source endpoint, the destination endpoint is provisioned in the SVM specified in the "destination.path" property. While protecting an SVM, the SVM destination endpoint can only be provisioned on the local cluster. To provision the SVM destination endpoint use the optional "source.cluster.name" property to specify the remote cluster name or use the optional "source.cluster.uuid" property to specify the remote cluster UUID. When "create\_destination.enabled" option is specified while making a POST for a SnapMirror relationship, the relationship can be automatically initialized by setting the "state" either to "snapmirrored" when the policy is of type "async" or to "in\_sync" when the policy is of type "sync". The "destination.path" property must specify the destination endpoint path. For example, for FlexVol volume and FlexGroup volume, the "destination.path" can be specified as <destination-svm-name:dp-volume-name>, for SVM data protection, the "destination.path" must be specified as &lt;destination-svm-name:&gt;, and for Consistency Group, the "destination.path" must be specified as &lt;destination-svm-name:&gt; along with the "destination.consistency\_group\_volumes" or "destination.luns" property to indicate the list of destination volumes or LUNs of type "DP" in the Consistency Group. For a FlexVol volume, a FlexGroup volume, Consistency Group or a Bucket destination endpoint, the properties in this object can be specified either from the source or the destination cluster. For an SVM destination endpoint, the

properties in this object can be specified from the destination cluster only. This object is not supported for non ONTAP endpoints. While protecting a S3 Bucket, the optional "size" property can be used to create ONTAP S3 Bucket destination endpoint of the specified size.</destination-svm-name:></destination-svm-name:></destination-svm-name:dp-volume-name>

Name	Type	Description
bucket_retention	<a href="#">bucket_retention</a>	Specifies the retention-mode and default retention period configured on the destination bucket.
enabled	boolean	Optional property to create the destination endpoint when establishing a SnapMirror relationship. It is assumed to be "false" if no other property is set and assumed to be "true" if any other property is set.
size	integer	Optional property to specify the size of destination endpoint in bytes. This property is applicable only to ONTAP S3 Bucket endpoints. The minimum size for S3 bucket is 80MB and maximum size is 64TB. If not specified, system will create destination with default size of 800GB. <ul style="list-style-type: none"> <li>• Introduced in: 9.10</li> <li>• x-nullable: true</li> </ul>
snapshot_locking_enabled	boolean	Optional property to create the destination endpoint with snapshot locking enabled when establishing a SnapMirror relationship. This property is applicable to FlexVol volumes and FlexGroup volumes.
storage_service	<a href="#">storage_service</a>	
tiering	<a href="#">tiering</a>	

cluster

Name	Type	Description
_links	<a href="#">_links</a>	
name	string	
uuid	string	

#### consistency\_group\_volumes

Name	Type	Description
name	string	The name of the volume.

#### luns

Optional property for a SnapMirror endpoint. Specifies the list of source LUNs and optionally list of destination LUNs during SnapMirror Consistency Group LUN restore operation.

Name	Type	Description
_links	<a href="#">_links</a>	
name	string	<p>The name of a LUN. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.</p> <p>LUN names are paths of the form "/vol/&lt;volume&gt;[/&lt;qtree&gt;]/&lt;namespace&gt;" where the qtree name is optional.</p>
uuid	string	The unique identifier of the LUN.

#### svm

SVM, applies only to SVM-scoped objects.

Name	Type	Description
_links	<a href="#">_links</a>	
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.

#### snapmirror\_endpoint

Endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to create the destination SVM endpoint or to establish an SVM DR relationship must have the property "cluster" populated with the remote cluster details. A POST request to create the destination FlexVol volume, FlexGroup volume, Consistency Group, ONTAP S3 bucket and NON-ONTAP object-store endpoints can optionally specify the "cluster" property when the source SVM and the destination SVM are peered. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination

SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
cluster	<a href="#">cluster</a>	
consistency_group_volumes	array[ <a href="#">consistency_group_volumes</a> ]	This property specifies the list of FlexVol volumes or LUNs of a Consistency Group. Optional on the ASA r2 platform. Mandatory for all other platforms.
luns	<a href="#">luns</a>	Optional property for a SnapMirror endpoint. Specifies the list of source LUNs and optionally list of destination LUNs during SnapMirror Consistency Group LUN restore operation.
path	string	ONTAP FlexVol/FlexGroup - svm1:volume1 ONTAP SVM - svm1: ONTAP Consistency Group - svm1:/cg/cg_name ONTAP S3 - svm1:/bucket/bucket1 NON-ONTAP - objstore1:/objstore <ul style="list-style-type: none"><li>• example: svm1:volume1</li><li>• Introduced in: 9.6</li><li>• x-nullable: true</li></ul>
svm	<a href="#">svm</a>	SVM, applies only to SVM-scoped objects.

## policy

Basic policy information of the relationship.

Name	Type	Description
_links	<a href="#">_links</a>	
name	string	Name of the SnapMirror policy.
type	string	
uuid	string	Unique identifier of the SnapMirror policy.

## snapmirror\_source\_endpoint

Source endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when

the endpoint is on a remote cluster. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
cluster	<a href="#">cluster</a>	
consistency_group_volumes	array[ <a href="#">consistency_group_volumes</a> ]	This property specifies the list of FlexVol volumes or LUNs of a Consistency Group. Optional on the ASA r2 platform. Mandatory for all other platforms.
luns	<a href="#">luns</a>	Optional property for a SnapMirror endpoint. Specifies the list of source LUNs and optionally list of destination LUNs during SnapMirror Consistency Group LUN restore operation.
path	string	ONTAP FlexVol/FlexGroup - svm1:volume1 ONTAP SVM - svm1: ONTAP Consistency Group - svm1:/cg/cg_name ONTAP S3 - svm1:/bucket/bucket1 NON-ONTAP - objstore1:/objstore <ul style="list-style-type: none"> <li>• example: svm1:volume1</li> <li>• Introduced in: 9.14</li> <li>• x-nullable: true</li> </ul>
svm	<a href="#">svm</a>	SVM, applies only to SVM-scoped objects.

#### svmdr\_volumes

Name	Type	Description
name	string	The name of the volume.

#### transfer

Basic information on the current transfer or the last transfer if there is no active transfer at the time of the request.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	

Name	Type	Description
bytes_transferred	integer	Total bytes transferred in the current or last successful transfer.
end_time	string	End time of the last transfer.
last_updated_time	string	Last updated time of the bytes transferred in the current transfer.
state	string	
total_duration	string	Elapsed time to transfer all snapshots for the last successful transfer.
type	string	Specifies the operation type of the current transfer on the relationship. The <i>initialize</i> transfer occurs when the relationship state changes from "uninitialized" to "snapmirrored" or "in_sync". The <i>update</i> transfer occurs when snapshots are being transferred from the source endpoint to the destination endpoint as part of a scheduled or manual update. The <i>resync</i> transfer occurs when the relationship state changes from "broken_off" to "snapmirrored" or "in_sync". The <i>restore</i> transfer occurs when a snapshot is being restored from a destination endpoint to another endpoint.
uuid	string	Transfer UUID. This property is applicable only for active transfers.

#### transfer\_schedule

Schedule used to update asynchronous relationships. This "transfer\_schedule" overrides the "transfer\_schedule" set on the SnapMirror relationship's policy. To remove the "transfer\_schedule", set its value to null (no-quotes). Only cron schedules are supported for SnapMirror.

Name	Type	Description
_links	<a href="#">_links</a>	
name	string	Job schedule name

Name	Type	Description
uuid	string	Job schedule UUID

snapmirror\_error

SnapMirror error

Name	Type	Description
arguments	array[string]	Arguments present in the error message encountered.
code	string	Error code
message	string	Error message

snapmirror\_relationship

SnapMirror relationship information. The SnapMirror relationship can be either "async" or "sync" based on the type of SnapMirror policy associated with the relationship. The source and destination endpoints of a SnapMirror relationship must be of the same type, for example, if the source endpoint is a FlexVol volume then the destination endpoint must be a FlexVol volume. The SnapMirror policy type "async" can be used when the SnapMirror relationship has FlexVol volume or FlexGroup volume or SVM as the endpoint. The SnapMirror policy type "sync" can be used when the SnapMirror relationship has FlexVol volume as the endpoint. The SnapMirror policy type "sync" with "sync\_type" as "automated\_failover" can be used when the SnapMirror relationship has Consistency Group as the endpoint.

Name	Type	Description
_links	<a href="#">_links</a>	
backoff_level	string	Specifies the SnapMirror backoff level due to Client Ops for FlexVol SnapMirror relationships.

Name	Type	Description
consistency_group_failover	<a href="#">snapmirror_consistency_group_failover</a>	<p>SnapMirror Consistency Group failover information. The SnapMirror Consistency Group failover can be a planned or an unplanned operation. Only active SnapMirror Consistency Group failover operation progress can be monitored using this object. In case of an error during the failover operation, the property "consistency_group_failover.error" holds the reason for the error. ONTAP automatically retries any failed SnapMirror Consistency Group failover operation.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.8</li> <li>• readOnly: 1</li> </ul>
destination	<a href="#">snapmirror_endpoint</a>	<p>Endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to create the destination SVM endpoint or to establish an SVM DR relationship must have the property "cluster" populated with the remote cluster details. A POST request to create the destination FlexVol volume, FlexGroup volume, Consistency Group, ONTAP S3 bucket and NON-ONTAP object-store endpoints can optionally specify the "cluster" property when the source SVM and the destination SVM are peered. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.6</li> </ul>
exported_snapshot	string	Snapshot exported to clients on destination.



Name	Type	Description
group_type	string	Specifies the group type of the top level SnapMirror relationship. The volume relationships are shown as <i>none</i> , the SVMDR relationships are shown as <i>svm_dr</i> , the Consistency Group relationships are shown as <i>consistency_group</i> , and the FlexGroup volume relationships are shown as <i>flexgroup</i> .
healthy	boolean	Is the relationship healthy?
identity_preservation	string	Specifies which configuration of the source SVM is replicated to the destination SVM. This property is applicable only for SVM data protection with "async" policy type. This "identity_preservation" overrides the "identity_preservation" set on the SnapMirror relationship's policy.
io_serving_copy	string	Specifies the sites serving I/O for the SnapMirror active sync relationship.
lag_time	string	Time since the exported snapshot was created.
last_transfer_network_compression_ratio	string	Specifies the compression ratio achieved for the data sent over the wire with network compression enabled for the last successful transfer.

Name	Type	Description
last_transfer_type	string	Specifies the operation type of the last transfer that occurred on the relationship. The <i>initialize</i> transfer occurs when the relationship state changes from uninitialized to snapmirrored or in_sync. The <i>update</i> transfer occurs when the snapshots are transferred from the source endpoint to the destination endpoint as part of scheduled or manual update. The <i>resync</i> transfer occurs when the relationship state changes from broken_off to snapmirrored or in_sync. The <i>restore</i> transfer occurs when the snapshot is restored from a destination endpoint to another endpoint.
master_bias_activated_site	string	Specifies the Master Bias Activated Site for the SnapMirror active sync relationship.
policy	<a href="#">policy</a>	Basic policy information of the relationship.
preferred_site	string	Specifies the Primary Site of the SnapMirror active sync relationship.
preserve	boolean	Set to true on resync to preserve snapshots on the destination that are newer than the latest common snapshot. This property is applicable only for relationships with FlexVol volume or FlexGroup volume endpoints and when the PATCH state is being changed to "snapmirrored".
quick_resync	boolean	Set to true to reduce resync time by not preserving storage efficiency. This property is applicable only for relationships with FlexVol volume endpoints and SVMDR relationships when the PATCH state is being changed to "snapmirrored".

Name	Type	Description
recover_after_break	boolean	Set to true to recover from a failed SnapMirror break operation on a FlexGroup volume relationship. This restores all destination FlexGroup constituent volumes to the latest snapshot, and any writes to the read-write constituents are lost. This property is applicable only for SnapMirror relationships with FlexGroup volume endpoints and when the PATCH state is being changed to "broken_off".
restore	boolean	Set to true to create a relationship for restore. To trigger restore-transfer, use transfers POST on the restore relationship. SnapMirror relationships with the policy type "async" can be restored. SnapMirror relationships with the policy type "sync" cannot be restored.
restore_to_snapshot	string	Specifies the snapshot to restore to on the destination during the break operation. This property is applicable only for SnapMirror relationships with FlexVol volume endpoints and when the PATCH state is being changed to "broken_off".
source	<a href="#">snapmirror_source_endpoint</a>	Source endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
state	string	<p>State of the relationship. To initialize the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". To break the relationship, PATCH the state to "broken_off" for relationships with a policy of type "async" or "sync".</p> <p>SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" cannot be "broken_off". To resync the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync".</p> <p>SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" can be in "broken_off" state due to a failed attempt of SnapMirror failover. To pause the relationship, suspending further transfers, PATCH the state to "paused" for relationships with a policy of type "async" or "sync". SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" cannot be "paused". To resume transfers for a paused relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". The entries "in_sync", "out_of_sync", "synchronizing", "expanding", and "shrinking" are only applicable to relationships with a policy of type "sync". A PATCH call on the state change only triggers the transition to the specified state. You must poll on the "state", "healthy" and "unhealthy_reason" properties using a GET request to determine if the transition is successful. To automatically initialize the relationship when specifying "create_destination" property, set the state to "snapmirrored" for</p>

Name	Type	Description
svmdr_volumes	array[ <a href="#">svmdr_volumes</a> ]	Specifies the list of constituent FlexVol volumes and FlexGroup volumes for an SVM DR SnapMirror relationship. FlexGroup constituents are not considered.
throttle	integer	Throttle, in KBs per second. This "throttle" overrides the "throttle" set on the SnapMirror relationship's policy. If neither of these are set, defaults to 0, which is interpreted as unlimited.
total_transfer_bytes	integer	Cumulative bytes transferred for the relationship.
total_transfer_duration	string	Indicates the cumulative duration of all transfers since the last aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship.
transfer	<a href="#">transfer</a>	Basic information on the current transfer or the last transfer if there is no active transfer at the time of the request.
transfer_schedule	<a href="#">transfer_schedule</a>	Schedule used to update asynchronous relationships. This "transfer_schedule" overrides the "transfer_schedule" set on the SnapMirror relationship's policy. To remove the "transfer_schedule", set its value to null (no-quotes). Only cron schedules are supported for SnapMirror.
unhealthy_reason	array[ <a href="#">snapmirror_error</a> ]	Reason the relationship is not healthy. It is a concatenation of up to four levels of error messages.
uuid	string	Unique identifier of the SnapMirror relationship.

returned\_error

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

## Create a SnapMirror relationship

POST /snapmirror/relationships

**Introduced In:** 9.6

Creates a SnapMirror relationship. This API can optionally provision the destination endpoint when it does not exist. This API must be executed on the cluster containing the destination endpoint unless the destination endpoint is being provisioned. When the destination endpoint is being provisioned, this API can also be executed from the cluster containing the source endpoint. Provisioning of the destination endpoint from the source cluster is supported for the FlexVol volume, FlexGroup volume and Application Consistency Group endpoints.

For SVM endpoints, provisioning the destination SVM endpoint is not supported from the source cluster. When the destination endpoint exists, the source SVM and the destination SVM must be in an SVM peer relationship. When provisioning the destination endpoint, the SVM peer relationship between the source SVM and the destination SVM is established as part of the destination provision, provided that the source SVM has SVM peering permissions for the destination cluster.

### Required properties

- `source.path` - Path to the source endpoint of the SnapMirror relationship.
- `destination.path` - Path to the destination endpoint of the SnapMirror relationship.
- `source.consistency_group_volumes` - List of FlexVol volumes of type "RW" that are constituents of an Application Consistency Group.
- `destination.consistency_group_volumes` - List of FlexVol volumes of type "DP" that are constituents of an Application Consistency Group.

### Recommended optional properties

- `policy.name` or `policy.uuid` - Policy governing the SnapMirror relationship.
- `state` - Set the state to "snapmirrored" to automatically initialize the relationship.
- `create_destination.enabled` - Enable this property to provision the destination endpoint.

## Default property values

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

- `policy.name` - *Asynchronous*
- `restore` - *false*
- `create_destination.tiering.policy` - *snapshot\_only* (when `create_destination.tiering.supported` is *true* for FlexVol volume)
- `create_destination.tiering.policy` - *none* (when `create_destination.tiering.supported` is *true* for FlexGroup volume)
- `create_destination.storage_service.enforce_performance` - *false*
- `destination.ipspace` - *Default*
- `throttle` - *0*
- `backoff_level` - *high*
- `policy.name` - *Asynchronous*
- `restore` - *false*
- `destination.ipspace` - *Default*
- `throttle` - *0*
- `backoff_level` - *high*

## Related ONTAP commands

- `snapmirror create`
- `snapmirror protect`

## Important notes

- The property "transfer\_schedule" if set on a SnapMirror relationship overrides the "transfer\_schedule" set on the policy being used with the SnapMirror relationship.
- The property "throttle" if set on a SnapMirror relationship overrides the "throttle" set on the policy being used with the SnapMirror relationship.
- The properties "transfer\_schedule" and "throttle" are not supported when "restore" is set to "true".
- The property "transfer\_schedule" cannot be set to null (no-quotes) during SnapMirror relationship POST.
- The property "throttle" is not supported when "create\_destination.enabled" is set to "true".
- The property "identity\_preservation" is applicable to only SnapMirror relationships with SVM endpoints and it indicates which configuration of the source SVM is replicated to the destination SVM.
- The property "backoff\_level" is not supported when "create\_destination.enabled" is set to "true".
- The property "backoff\_level" is only applicable for FlexVol SnapMirror relationships.

## Examples

The following examples show how to create FlexVol volumes, FlexGroup volumes, SVM and Application

Consistency Group SnapMirror relationships. Note that the source SVM name must be the local name of the peer SVM.

Creating a FlexVol SnapMirror relationship of type XDP.

```
POST "/api/snapmirror/relationships/" '{"source": {"path":  
"src_svm:src_vol"}, "destination": { "path": "dst_svm:dst_vol"}}'
```

Creating a FlexGroup SnapMirror relationship of type XDP.

```
POST "/api/snapmirror/relationships/" '{"source": {"path":  
"src_svm:source_flexgroup"}, "destination": { "path":  
"dst_svm:dest_flexgroup"}}'
```

Creating a SVM SnapMirror relationship of type XDP.

```
POST "/api/snapmirror/relationships/" '{"source": { "path": "src_svm:"},  
"destination": { "path": "dst_svm:"}}'
```

Creating a SnapMirror relationship in order to restore from a destination.

```
POST "/api/snapmirror/relationships/" '{"source": {"path":  
"src_svm:src_vol"}, "destination": { "path": "dst_svm:dst_vol"},  
"restore": "true"}'
```

Provision the destination FlexVol volume endpoint and create a SnapMirror relationship of type XDP.

```
POST "/api/snapmirror/relationships/" '{"source": {"path":  
"src_svm:src_vol"}, "destination": { "path": "dst_svm:dst_vol"},  
"create_destination": { "enabled": "true" }}'
```

Provision the destination FlexVol volume endpoint on a Fabricpool with a tiering policy and create a SnapMirror relationship of type XDP.

```
POST "/api/snapmirror/relationships/" '{"source": {"path":  
"src_svm:src_vol"}, "destination": { "path": "dst_svm:dst_vol"},  
"create_destination": { "enabled": "true", "tiering": { "supported":  
"true", "policy": "auto" } } }'
```

Provision the destination FlexVol volume endpoint using storage service and create a SnapMirror relationship of type XDP.



```
POST "/api/snapmirror/relationships/" '{"source": {"path":
"src_svm:src_vol"}, "destination": { "path": "dst_svm:dst_vol"},
"create_destination": { "enabled": "true", "storage_service": { "enabled":
"true", "name": "extreme", "enforce_performance": "true" } } }'
```

Provision the destination SVM endpoint and create a SnapMirror relationship of type XDP.

```
POST "/api/snapmirror/relationships/" '{"source": {"path": "src_svm:",
"cluster": { "name": "cluster_src" }}, "destination": { "path":
"dst_svm:", "create_destination": { "enabled": "true" } }'
```

Create an asynchronous SnapMirror relationship with Application Consistency Group endpoint.

```
POST "/api/snapmirror/relationships/" '{"source": { "path":
"src_svm:/cg/cg_src_vol", "consistency_group_volumes": [{ "name":
"src_vol_1" }, { "name": "src_vol_2" }] }, "destination": { "path":
"dst_svm:/cg/cg_dst_vol", "consistency_group_volumes": [{ "name":
"dst_vol_1" }, { "name": "dst_vol_2" }] }, "policy": "Asynchronous" }'
```

Create an asynchronous SnapMirror relationship with a child Application Consistency Group endpoint.

```
POST "/api/snapmirror/relationships/" '{"source": { "path":
"src_svm:/cg/parent_src_cg/sub_child_cg" }, "destination": { "path":
"dst_svm:/cg/cg_dst_vol" }, "policy": "MirrorAndVault" }'
```

```
POST "/api/snapmirror/relationships/" '{"source": {"path": "src_svm:/cg/cg_src_vol",
"consistency_group_volumes": [{ "name": "src_vol_1" }, { "name": "src_vol_2" } ]}, "destination": { "path":
"dst_svm:/cg/cg_dst_vol", "consistency_group_volumes": [{ "name": "dst_vol_1" }, { "name": "dst_vol_2" } ]},
"create_destination": { "enabled": "true", "tiering": { "supported": "true" } }, "policy": "Asynchronous", "state":
"snapmirrored" }'
```

Create a SnapMirror active sync relationship with the Application Consistency Group endpoint.

```
POST "/api/snapmirror/relationships/" '{"source": { "path": "src_svm:/cg/cg_src_vol",
"consistency_group_volumes": [{ "name": "src_vol_1" }, { "name": "src_vol_2" } ]}, "destination": { "path":
"dst_svm:/cg/cg_dst_vol", "consistency_group_volumes": [{ "name": "dst_vol_1" }, { "name": "dst_vol_2" } ]},
"policy": "AutomatedFailOver" }'
```

Provision the destination Application Consistency Group endpoint on a Fabricpool with a tiering policy, create a SnapMirror active sync relationship with a SnapMirror policy of type "sync" and sync\_type of "automated\_failover", and initialize the SnapMirror relationship with state as "in\_sync".

```
POST "/api/snapmirror/relationships/" '{"source": {"path": "src_svm:/cg/cg_src_vol",
"consistency_group_volumes": [{ "name": "src_vol_1" }, { "name": "src_vol_2" } ] }, "destination": { "path":
"dst_svm:/cg/cg_dst_vol", "consistency_group_volumes": [{ "name": "dst_vol_1" }, { "name": "dst_vol_2" } ] },
"create_destination": { "enabled": "true", "tiering": { "supported": "true" } }, "policy": "AutomatedFailOver",
"state": "in_sync" }'
```

Provision the destination Application Consistency Group endpoint with storage service, create a SnapMirror active sync relationship with a SnapMirror policy of type "sync" and sync\_type of "automated\_failover", and initialize the SnapMirror relationship with state as "in\_sync".

```
POST "/api/snapmirror/relationships/" '{"source": {"path": "src_svm:/cg/cg_src_vol",
"consistency_group_volumes": [{ "name": "src_vol_1" }, { "name": "src_vol_2" } ] }, "destination": { "path":
"dst_svm:/cg/cg_dst_vol", "consistency_group_volumes": [{ "name": "dst_vol_1" }, { "name": "dst_vol_2" } ] },
"create_destination": { "enabled": "true", "storage_service": { "enabled": "true", "name": "extreme",
"enforce_performance": "true" } }, "policy": "AutomatedFailOver", "state": "in_sync" }'
```

Provision the destination Application Consistency Group endpoint with storage service, create an asynchronous application consistency group relationship with a SnapMirror policy of type "async" and an async\_type of "XDPDefault", and initialize the SnapMirror relationship with state as "SnapMirrored".

```
POST "/api/snapmirror/relationships/" '{"source": {"path": "src_svm:/cg/cg_src_vol",
"consistency_group_volumes": [{ "name": "src_vol_1" }, { "name": "src_vol_2" } ] }, "destination": { "path":
"dst_svm:/cg/cg_dst_vol", "consistency_group_volumes": [{ "name": "dst_vol_1" }, { "name": "dst_vol_2" } ] },
"create_destination": { "enabled": "true", "storage_service": { "enabled": "true", "name": "extreme",
"enforce_performance": "true" } }, "policy": "XDPDefault", "state": "snapmirrored" }'
```

Creating a FlexVol volume SnapMirror relationship of type XDP with transfer\_schedule and throttle.

```
POST "/api/snapmirror/relationships/" '{"source": {"path": "src_svm:src_vol"}, "destination": { "path":
"dst_svm:dst_vol"}, "transfer_schedule":{"uuid":"817500fa-092d-44c5-9c10-7b54f7b2f20a", "name":"5min"},
"throttle":100}'
```

Creating an asynchronous SnapMirror relationship with backoff\_level set to medium.

```
POST "/api/snapmirror/relationships/" '{"source": {"path": "src_svm:src_vol"}, "destination": {"path": "dst_svm:dst_vol"}, "backoff_level": "medium"}' `
```

## Learn more

- [DOC /snapmirror/relationships](#)

## Parameters

Name	Type	In	Required	Description
return_records	boolean	query	False	The default is false. If set to true, the records are returned. <ul style="list-style-type: none"><li>• Default value:</li></ul>

Name	Type	In	Required	Description
return_timeout	integer	query	False	<p>The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.</p> <ul style="list-style-type: none"> <li>• Default value: 0</li> <li>• Max value: 120</li> <li>• Min value: 0</li> </ul>
validate_only	boolean	query	False	<p>Validate the operation and its parameters, without actually performing the operation.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.7</li> </ul>

## Request Body

Name	Type	Description
backoff_level	string	Specifies the SnapMirror backoff level due to Client Ops for FlexVol SnapMirror relationships.

Name	Type	Description
create_destination	<a href="#">snapmirror_destination_creation</a>	<p>Use this object to provision the destination endpoint when establishing a SnapMirror relationship for a FlexVol volume, FlexGroup volume, SVM, Consistency Group or ONTAP S3 Bucket. Given a source endpoint, the destination endpoint is provisioned in the SVM specified in the "destination.path" property. While protecting an SVM, the SVM destination endpoint can only be provisioned on the local cluster. To provision the SVM destination endpoint use the optional "source.cluster.name" property to specify the remote cluster name or use the optional "source.cluster.uuid" property to specify the remote cluster UUID. When "create_destination.enabled" option is specified while making a POST for a SnapMirror relationship, the relationship can be automatically initialized by setting the "state" either to "snapmirrored" when the policy is of type "async" or to "in_sync" when the policy is of type "sync". The "destination.path" property must specify the destination endpoint path. For example, for FlexVol volume and FlexGroup volume, the "destination.path" can be specified as &lt;destination-SVM-name:dp-volume-name&gt;, for SVM data protection, the "destination.path" must be specified as &amp;lt;destination-SVM-name:&amp;gt;, and for Consistency Group, the "destination.path" must be specified as &amp;lt;destination-SVM-name:/cg/consistency-group-name&amp;gt; along with the "destination.consistency_group_volumes" or "destination.luns" property to indicate the list of destination volumes or LUNs of type "DP" in the Consistency Group. For a FlexVol volume, a FlexGroup volume, Consistency Group or a Bucket destination endpoint, the properties in this object can be specified either from the source or the destination cluster. For an SVM destination</p>

Name	Type	Description
destination	<a href="#">snapmirror_endpoint</a>	<p>Endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to create the destination SVM endpoint or to establish an SVM DR relationship must have the property "cluster" populated with the remote cluster details. A POST request to create the destination FlexVol volume, FlexGroup volume, Consistency Group, ONTAP S3 bucket and NON-ONTAP object-store endpoints can optionally specify the "cluster" property when the source SVM and the destination SVM are peered. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.6</li> </ul>
exported_snapshot	string	Snapshot exported to clients on destination.
group_type	string	Specifies the group type of the top level SnapMirror relationship. The volume relationships are shown as <i>none</i> , the SVM DR relationships are shown as <i>svm_dr</i> , the Consistency Group relationships are shown as <i>consistency_group</i> , and the FlexGroup volume relationships are shown as <i>flexgroup</i> .
healthy	boolean	Is the relationship healthy?

Name	Type	Description
identity_preservation	string	Specifies which configuration of the source SVM is replicated to the destination SVM. This property is applicable only for SVM data protection with "async" policy type. This "identity_preservation" overrides the "identity_preservation" set on the SnapMirror relationship's policy.
io_serving_copy	string	Specifies the sites serving I/O for the SnapMirror active sync relationship.
lag_time	string	Time since the exported snapshot was created.
last_transfer_network_compression_ratio	string	Specifies the compression ratio achieved for the data sent over the wire with network compression enabled for the last successful transfer.
last_transfer_type	string	Specifies the operation type of the last transfer that occurred on the relationship. The <i>initialize</i> transfer occurs when the relationship state changes from uninitialized to snapmirrored or in_sync. The <i>update</i> transfer occurs when the snapshots are transferred from the source endpoint to the destination endpoint as part of scheduled or manual update. The <i>resync</i> transfer occurs when the relationship state changes from broken_off to snapmirrored or in_sync. The <i>restore</i> transfer occurs when the snapshot is restored from a destination endpoint to another endpoint.
master_bias_activated_site	string	Specifies the Master Bias Activated Site for the SnapMirror active sync relationship.
policy	policy	Basic policy information of the relationship.

Name	Type	Description
preferred_site	string	Specifies the Primary Site of the SnapMirror active sync relationship.
preserve	boolean	Set to true on resync to preserve snapshots on the destination that are newer than the latest common snapshot. This property is applicable only for relationships with FlexVol volume or FlexGroup volume endpoints and when the PATCH state is being changed to "snapmirrored".
quick_resync	boolean	Set to true to reduce resync time by not preserving storage efficiency. This property is applicable only for relationships with FlexVol volume endpoints and SVMDR relationships when the PATCH state is being changed to "snapmirrored".
recover_after_break	boolean	Set to true to recover from a failed SnapMirror break operation on a FlexGroup volume relationship. This restores all destination FlexGroup constituent volumes to the latest snapshot, and any writes to the read-write constituents are lost. This property is applicable only for SnapMirror relationships with FlexGroup volume endpoints and when the PATCH state is being changed to "broken_off".
restore	boolean	Set to true to create a relationship for restore. To trigger restore-transfer, use transfers POST on the restore relationship. SnapMirror relationships with the policy type "async" can be restored. SnapMirror relationships with the policy type "sync" cannot be restored.



Name	Type	Description
restore_to_snapshot	string	Specifies the snapshot to restore to on the destination during the break operation. This property is applicable only for SnapMirror relationships with FlexVol volume endpoints and when the PATCH state is being changed to "broken_off".
source	<a href="#">snapmirror_source_endpoint</a>	Source endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
state	string	<p>State of the relationship. To initialize the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". To break the relationship, PATCH the state to "broken_off" for relationships with a policy of type "async" or "sync". SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" cannot be "broken_off". To resync the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" can be in "broken_off" state due to a failed attempt of SnapMirror failover. To pause the relationship, suspending further transfers, PATCH the state to "paused" for relationships with a policy of type "async" or "sync". SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" cannot be "paused". To resume transfers for a paused relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". The entries "in_sync", "out_of_sync", "synchronizing", "expanding", and "shrinking" are only applicable to relationships with a policy of type "sync". A PATCH call on the state change only triggers the transition to the specified state. You must poll on the "state", "healthy" and "unhealthy_reason" properties using a GET request to determine if the transition is successful. To automatically initialize the relationship when specifying "create_destination" property, set the state to "snapmirrored" for relationships with a policy of type</p>

Name	Type	Description
svmdr_volumes	array[ <a href="#">svmdr_volumes</a> ]	Specifies the list of constituent FlexVol volumes and FlexGroup volumes for an SVM DR SnapMirror relationship. FlexGroup constituents are not considered.
throttle	integer	Throttle, in KBs per second. This "throttle" overrides the "throttle" set on the SnapMirror relationship's policy. If neither of these are set, defaults to 0, which is interpreted as unlimited.
total_transfer_bytes	integer	Cumulative bytes transferred for the relationship.
total_transfer_duration	string	Indicates the cumulative duration of all transfers since the last aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship.
transfer	<a href="#">transfer</a>	Basic information on the current transfer or the last transfer if there is no active transfer at the time of the request.
transfer_schedule	<a href="#">transfer_schedule</a>	Schedule used to update asynchronous relationships. This "transfer_schedule" overrides the "transfer_schedule" set on the SnapMirror relationship's policy. To remove the "transfer_schedule", set its value to null (no-quotes). Only cron schedules are supported for SnapMirror.
unhealthy_reason	array[ <a href="#">snapmirror_error</a> ]	Reason the relationship is not healthy. It is a concatenation of up to four levels of error messages.
uuid	string	Unique identifier of the SnapMirror relationship.

## Example request

```
{
  "backoff_level": "medium",
  "create_destination": {
    "bucket_retention": {
      "default_period": "P10Y",
      "mode": "governance"
    },
    "storage_service": {
      "name": "string"
    },
    "tiering": {
      "policy": "string"
    }
  },
  "destination": {
    "cluster": {
      "name": "cluster1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "consistency_group_volumes": [
      {
        "name": "volume1",
        "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
      }
    ],
    "ipspace": "Default",
    "luns": {
      "name": "/vol/volume1/lun1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "path": "svm1:volume1",
    "svm": {
      "name": "svm1",
      "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
    }
  },
  "exported_snapshot": "string",
  "group_type": "consistency_group",
  "identity_preservation": "string",
  "io_serving_copy": "C1_sti85-vsim-ucs209a_cluster, C1_sti85-vsim-ucs209c_cluster",
  "lag_time": "PT8H35M42S",
  "last_transfer_network_compression_ratio": 61,
  "last_transfer_type": "initialize",
}
```

```

"master_bias_activated_site": "C1_sti85-vsim-ucs209a_cluster",
"policy": {
  "name": "Asynchronous",
  "type": "string",
  "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
},
"preferred_site": "C1_sti85-vsim-ucs209a_cluster",
"restore_to_snapshot": "string",
"source": {
  "cluster": {
    "name": "cluster1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "consistency_group_volumes": [
    {
      "name": "volume1",
      "uuid": "028baa66-41bd-11e9-81d5-00a0986138f7"
    }
  ],
  "luns": {
    "name": "/vol/volume1/lun1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "path": "svm1:volume1",
  "svm": {
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  }
},
"state": "snapmirrored",
"svmdr_volumes": [
  {
    "name": "volume1"
  }
],
"throttle": 0,
"total_transfer_bytes": 1098210312,
"total_transfer_duration": "PT3M21S",
"transfer": {
  "end_time": "2020-12-02 18:36:19 -0800",
  "last_updated_time": "2023-09-14 15:39:19 -0700",
  "state": "string",
  "total_duration": "PT28M41S",
  "type": "initialize",
  "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
},

```

```

"transfer_schedule": {
  "name": "weekly",
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"unhealthy_reason": [
  {
    "arguments": [],
    "code": "6621444",
    "message": "Failed to complete update operation on one or more
item relationships."
  },
  {
    "arguments": [],
    "code": "6621445",
    "message": "Group Update failed"
  }
],
"uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
}

```

## Response

Status: 202, Accepted

Name	Type	Description
job	<a href="#">job_link</a>	

## Example response

```

{
  "job": {
    "uuid": "string"
  }
}

```

## Headers

Name	Description	Type
Location	Useful for tracking the resource location	string

## Response

Status: 201, Created

## Error

Status: Default

### ONTAP Error Response Codes

Error Code	Description
918657	Invalid duration value for the "retention_period" field.
	1115542
Invalid value given for the field.	
6619441	The source volume cannot be the same as the destination volume.
	6619599
Only the "none" tiering policy is supported when creating a destination volume with "snapshot_locking_enabled" set to true or if "snapshot_locking_enabled" is set to true on the source volume.	
6619637	Relationship with specified destination volume already exists.
	6619699
Schedule not found.	
6620374	Internal error. Failed to get SVM information.
	6620478
Internal error. Failed to check SnapMirror capability.	
6620608	Child consistency group cannot be a destination endpoint for SnapMirror relationship creation.
	6620654
Invalid SnapMirror Consistency Group name.	
6621125	The policy is not valid for relationships with FlexGroup volume endpoints. Only policies without snapshot creation schedules are supported for these relationships.
	6621128

Error Code	Description
Policy is not valid because it has snapshot creation schedules associated with it. Such policies cannot be used for relationships with Consistency Group endpoints.	
6621458	The destination Consistency Group is the source of a SnapMirror Synchronous (SM-S) relationship. Sources of SM-S relationships cannot be the destination of any other SnapMirror relationship.
	6621782
A property of the policy is not valid for relationships between these endpoints.	
6621834	Object store configuration does not exist for the specified SVM.
	6622088
SnapMirror Asynchronous relationship is not supported on a Consistency Group volume that has snapshot locking enabled.	
13303819	Could not retrieve SnapMirror policy information.
	13303821
Invalid SnapMirror policy UUID.	
13303841	This operation is not supported for SnapMirror relationships between these endpoints.
	13303852
destination.path provided does not contain ":".	
13303853	Restore relationships are not supported for SVM-DR endpoints.
	13303866
Associating the specified SnapMirror policy with this SnapMirror relationship is not supported.	
13303868	Create of destination endpoint and SnapMirror relationship failed.
	13303869
Creating a destination endpoint is not supported for restore relationships.	
13303870	A tiering policy cannot be specified if tiering is not being set to supported.
	13303871
Storage service properties cannot be specified if the storage service is not being enabled.	



Error Code	Description
13303872	Specified property requires a later effective cluster version.
	13303873
Specifying a state when creating a relationship is only supported when creating a destination endpoint.	
13303874	Specified state is not supported when creating this relationship.
	13303875
Destination aggregates do not have sufficient space for hosting copies of source volumes.	
13303876	Destination cluster does not have composite aggregates.
	13303877
Source or destination cluster must be specified.	
13303878	The specified fields do not match.
	13303879
Source cluster name or UUID is needed to provision a destination SVM on the local cluster.	
13303880	Source cluster must be remote for provisioning a destination SVM on the local cluster.
	13303881
Network validation failed.	
13303882	SVM validation failed.
	13303883
Encryption is not enabled on the destination cluster.	
13303886	SVM peer permission not found.
	13303887
Synchronous SnapMirror relationships between FlexGroup volumes are not supported.	
13303888	Synchronous SnapMirror relationships require an effective cluster version of 9.5 or later on both the source and destination clusters.
	13303889
Asynchronous SnapMirror relationships between FlexGroup volumes require an effective cluster version of 9.5 or later on both the source and destination clusters.	

Error Code	Description
13303890	Asynchronous SnapMirror relationships between FlexVol volumes require an effective cluster version of 9.3, 9.5, or later on both the source and destination clusters.
	13303891
Creating a destination endpoint with storage service requires an effective cluster version of 9.7 or later.	
13303892	Fetching remote information from the destination cluster failed.
	13303893
Updating job description failed.	
13303894	Destination volume name is invalid. It must contain the source volume name and have a suffix when creating a destination endpoint on a cluster with an effective cluster version of 9.6 or earlier.
	13303895
Operation on the remote destination cluster is not supported.	
13303916	FlexGroup volumes are not supported on SnapLock aggregates.
	13303918
No suitable destination aggregate type is available.	
13303919	Only FabricPool enabled aggregates are available on the destination.
	13303920
Only SnapLock aggregates are available on the destination. FlexGroup volumes are not supported on SnapLock aggregates.	
13303921	Unable to retrieve the SnapMirror capabilities of the destination cluster.
	13303922
Specified source SVM is not a data SVM.	
13303923	Specified destination SVM is not a data SVM.
	13303924
Source SVM has an invalid snapshot policy.	
13303925	SnapMirror validation has failed.
	13303930
The specified tiering policy is not supported for destination volumes of Synchronous relationships.	

Error Code	Description
13303938	Fetching information from the local cluster failed.
	13303939
Could not create an SVM peer relationship.	
13303944	An SVM-DR relationship is not supported because the source SVM has CIFS configured and the associated SnapMirror policy has either the "identity_preservation" property not set or set to "exclude_network_and_protocol_config".
	13303949
This SnapMirror policy is only supported for relationships with object store destination endpoints.	
13303966	Consistency Group relationships require a policy of type "sync" with a sync_type of "automated_failover".
	13303967
Consistency Group volume is not a FlexVol volume.	
13303968	Unsupported volume type for the Consistency Group.
	13303969
SnapMirror relationships between SVM endpoints and object store endpoints are not supported.	
13303970	Unsupported policy type for the Consistency Group.
	13303971
SnapMirror relationships between Consistency Group endpoints and object store endpoints are not supported.	
13303976	Source or destination SVM is already part of an SVM-DR relation.
	13303977
Destination Consistency Group volume UUIDs are not expected while provisioning the destination volumes.	
13303978	Number of Consistency Group volume names and UUIDs does not match.
	13303979
Number of Consistency Group volumes exceeds the allowed limit.	
13303980	Number of source and destination Consistency Group volumes do not match.
	13303981
ISCSI or FCP protocol is not configured.	

Error Code	Description
13303982	SAN data interface is not configured on the SVM.
	13304021
No suitable storage can be found meeting the specified requirements. No FabricPool enabled aggregates are available on the destination.	
13304022	No suitable storage can be found meeting the specified requirements. No non-root, non-taken-over, non-SnapLock, non-composite aggregates are available on the destination.
	13304032
In an "All SAN Array", an SVM-DR relationship is not supported when the associated SnapMirror policy does not have the "identity_preservation" property set to "exclude_network_and_protocol_config".	
13304080	Specified UUID and name do not match.
	13304082
Specified properties are mutually exclusive.	
13304083	The specified property is not supported because all nodes in the cluster are not capable of supporting the property.
	13304093
The property specified is not supported for the specified relationships.	
13304098	This SnapMirror policy is not supported for SnapMirror relationships with SnapLock volumes.
	13304099
SnapLock Compliance Clock is not running on all nodes in the destination cluster.	
13304108	Schedule not found in the Administrative SVM or the SVM for the relationship.
	13304112
File restore from a Consistency Group asynchronous SnapMirror relationship endpoint is not supported.	
13304132	Creating a destination endpoint is not supported with the "backoff_level" property.
	13304144
SnapLock is not licensed on the destination cluster.	

Error Code	Description
13304155	Invalid SnapMirror consistency group name. The name must be from 1 to 30 characters in length. The first letter must be one of A-Z, a-z or "'". <i>The remaining characters must be A-Z, a-z, 0-9 or "'</i> .
	13304160
A relationship with the specified object store endpoint exists and a restore relationship cannot be created with the same object store source endpoint name as the destination of a backup relationship except when restoring to the source volume of the existing relationship. The "source.uuid", not the source endpoint name, establishes what object is restored from the cloud.</private>	
53411897	The specified source volumes do not match the volumes contained in the source consistency group.

## Definitions

## See Definitions

href

Name	Type	Description
href	string	

\_links

error\_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

SnapMirror Consistency Group failover error message.

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message

status

Name	Type	Description
code	string	Status code
message	string	SnapMirror Consistency Group failover status.

snapmirror\_consistency\_group\_failover

SnapMirror Consistency Group failover information. The SnapMirror Consistency Group failover can be a planned or an unplanned operation. Only active SnapMirror Consistency Group failover operation progress can be monitored using this object. In case of an error during the failover operation, the property "consistency\_group\_failover.error" holds the reason for the error. ONTAP automatically retries any failed SnapMirror Consistency Group failover operation.

Name	Type	Description
state	string	SnapMirror Consistency Group failover state.
status	<a href="#">status</a>	
type	string	SnapMirror Consistency Group failover type.

bucket\_retention

Specifies the retention-mode and default retention period configured on the destination bucket.

Name	Type	Description
default_period	string	Specifies the default retention period that is applied to objects while committing them to the WORM state without an associated retention period. The retention period can be in years, or days. The retention period value represents a duration and must be specified in the ISO-8601 duration format. A period specified for years and days is represented in the ISO-8601 format as "P<num>Y" and "P<num>D" respectively, for example "P10Y" represents a duration of 10 years. The period string must contain only a single time element that is, either years, or days. A duration which combines different periods is not supported, for example "P1Y10D" is not supported.</num></num>
mode	string	The lock mode of the bucket. compliance &dash; A SnapLock Compliance (SLC) bucket provides the highest level of WORM protection and an administrator cannot destroy a compliance bucket if it contains unexpired WORM objects. governance &dash; An administrator can delete a Governance bucket. no_lock &dash; Indicates the bucket does not support object locking.

## storage\_service

Name	Type	Description
enabled	boolean	<p>This property indicates whether to create the destination endpoint using storage service. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"><li>• Introduced in: 9.7</li><li>• x-nullable: true</li></ul>
enforce_performance	boolean	<p>Optional property to enforce storage service performance on the destination endpoint. This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"><li>• Default value: 1</li><li>• Introduced in: 9.7</li><li>• x-nullable: true</li></ul>



Name	Type	Description
name	string	<p>Optional property to specify the storage service name for the destination endpoint. This property is considered when the property "create_destination.storage_service.enabled" is set to "true". When the property "create_destination.storage_service.enabled" is set to "true" and the "create_destination.storage_service.name" for the endpoint is not specified, then ONTAP selects the highest storage service available on the cluster to provision the destination endpoint. This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>enum: ["extreme", "performance", "value"]</li> <li>Introduced in: 9.6</li> <li>x-nullable: true</li> </ul>

tiering



Name	Type	Description
policy	string	<p>Optional property to specify the destination endpoint's tiering policy when "create_destination.tiering.supported" is set to "true". This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. This property determines whether the user data blocks of the destination endpoint in a FabricPool will be tiered to the cloud store when they become cold. FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of the destination endpoint volume blocks increases if they are accessed frequently and decreases when they are not. all – This policy allows tiering of both destination endpoint snapshots and the user transferred data blocks to the cloud store as soon as possible by ignoring the temperature on the volume blocks. This tiering policy is not applicable for Consistency Group destination endpoints or for synchronous relationships. auto – This policy allows tiering of both destination endpoint snapshots and the active file system user data to the cloud store none – Destination endpoint volume blocks will not be tiered to the cloud store. snapshot_only – This policy allows tiering of only the destination endpoint volume snapshots not associated with the active file system. The default tiering policy is "snapshot_only" for a FlexVol volume and "none" for a FlexGroup volume. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>enum: ["all", "auto", "none", "snapshot_only"]</li> <li>Introduced in: 9.6</li> </ul> <p>x-nullable: true</p>

Name	Type	Description
supported	boolean	<p>Optional property to enable provisioning of the destination endpoint volumes on FabricPool aggregates. This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. Only FabricPool aggregates are used if this property is set to "true" and only non FabricPool aggregates are used if this property is set to "false". Tiering support for a FlexGroup volume can be changed by moving all of the constituents to the required aggregates. Note that in order to tier data, not only do the destination endpoint volumes need to support tiering by using FabricPools, the "create_destination.tiering.policy" must not be "none". A destination endpoint that uses FabricPools but has a tiering "policy" of "none" supports tiering but will not tier any data. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.6</li> <li>• x-nullable: true</li> </ul>

#### snapmirror\_destination\_creation

Use this object to provision the destination endpoint when establishing a SnapMirror relationship for a FlexVol volume, FlexGroup volume, SVM, Consistency Group or ONTAP S3 Bucket. Given a source endpoint, the destination endpoint is provisioned in the SVM specified in the "destination.path" property. While protecting an SVM, the SVM destination endpoint can only be provisioned on the local cluster. To provision the SVM destination endpoint use the optional "source.cluster.name" property to specify the remote cluster name or use the optional "source.cluster.uuid" property to specify the remote cluster UUID. When "create\_destination.enabled" option is specified while making a POST for a SnapMirror relationship, the relationship can be automatically initialized by setting the "state" either to "snapmirrored" when the policy is of type "async" or to "in\_sync" when the policy is of type "sync". The "destination.path" property must specify the destination endpoint path. For example, for FlexVol volume and FlexGroup volume, the "destination.path" can be specified as <destination-svm-name:dp-volume-name>, for SVM data protection, the "destination.path" must be specified as &lt;destination-svm-name:>, and for Consistency Group, the "destination.path" must be specified as &lt;destination-svm-name:> along with the "destination.consistency\_group\_volumes" or "destination.luns" property to indicate the list of destination volumes or LUNs of type "DP" in the Consistency Group. For a FlexVol volume, a FlexGroup volume, Consistency Group or a Bucket destination endpoint, the properties in this object can be specified either from the source or the destination cluster. For an SVM destination endpoint, the

properties in this object can be specified from the destination cluster only. This object is not supported for non ONTAP endpoints. While protecting a S3 Bucket, the optional "size" property can be used to create ONTAP S3 Bucket destination endpoint of the specified size.</destination-svm-name:></destination-svm-name:></destination-svm-name:dp-volume-name>

Name	Type	Description
bucket_retention	<a href="#">bucket_retention</a>	Specifies the retention-mode and default retention period configured on the destination bucket.
enabled	boolean	Optional property to create the destination endpoint when establishing a SnapMirror relationship. It is assumed to be "false" if no other property is set and assumed to be "true" if any other property is set.
size	integer	Optional property to specify the size of destination endpoint in bytes. This property is applicable only to ONTAP S3 Bucket endpoints. The minimum size for S3 bucket is 80MB and maximum size is 64TB. If not specified, system will create destination with default size of 800GB. <ul style="list-style-type: none"> <li>• Introduced in: 9.10</li> <li>• x-nullable: true</li> </ul>
snapshot_locking_enabled	boolean	Optional property to create the destination endpoint with snapshot locking enabled when establishing a SnapMirror relationship. This property is applicable to FlexVol volumes and FlexGroup volumes.
storage_service	<a href="#">storage_service</a>	
tiering	<a href="#">tiering</a>	

cluster

Name	Type	Description
name	string	
uuid	string	

## consistency\_group\_volumes

Name	Type	Description
name	string	The name of the volume.
uuid	string	Unique identifier of the volume. This corresponds to the instance-uuid that is exposed in the CLI and ONTAPI. It does not change due to a volume move. <ul style="list-style-type: none"><li>• example: 028baa66-41bd-11e9-81d5-00a0986138f7</li><li>• x-ntap-createOnly: true</li><li>• Introduced in: 9.8</li><li>• x-nullable: true</li></ul>

## luns

Optional property for a SnapMirror endpoint. Specifies the list of source LUNs and optionally list of destination LUNs during SnapMirror Consistency Group LUN restore operation.

Name	Type	Description
name	string	The name of a LUN. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.  LUN names are paths of the form "/vol/<volume>[/<qtree>]/<namespace>" where the qtree name is optional.
uuid	string	The unique identifier of the LUN.

## svm

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.

## snapmirror\_endpoint

Endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to create the destination SVM endpoint or to establish an SVM DR relationship must have the property "cluster" populated with the remote cluster details. A POST request to create the destination FlexVol volume, FlexGroup volume, Consistency Group, ONTAP S3 bucket and NON-ONTAP object-store endpoints can optionally specify the "cluster" property when the source SVM and the destination SVM are peered. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
cluster	<a href="#">cluster</a>	
consistency_group_volumes	array[ <a href="#">consistency_group_volumes</a> ]	This property specifies the list of FlexVol volumes or LUNs of a Consistency Group. Optional on the ASA r2 platform. Mandatory for all other platforms.
ipspace	string	Optional property to specify the IPspace of the SVM.
luns	<a href="#">luns</a>	Optional property for a SnapMirror endpoint. Specifies the list of source LUNs and optionally list of destination LUNs during SnapMirror Consistency Group LUN restore operation.
path	string	ONTAP FlexVol/FlexGroup - svm1:volume1 ONTAP SVM - svm1: ONTAP Consistency Group - svm1:/cg/cg_name ONTAP S3 - svm1:/bucket/bucket1 NON-ONTAP - objstore1:/objstore <ul style="list-style-type: none"><li>• example: svm1:volume1</li><li>• Introduced in: 9.6</li><li>• x-nullable: true</li></ul>
svm	<a href="#">svm</a>	SVM, applies only to SVM-scoped objects.

## policy

Basic policy information of the relationship.

Name	Type	Description
name	string	Name of the SnapMirror policy.
type	string	
uuid	string	Unique identifier of the SnapMirror policy.

#### snapmirror\_source\_endpoint

Source endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
cluster	<a href="#">cluster</a>	
consistency_group_volumes	array[ <a href="#">consistency_group_volumes</a> ]	This property specifies the list of FlexVol volumes or LUNs of a Consistency Group. Optional on the ASA r2 platform. Mandatory for all other platforms.
luns	<a href="#">luns</a>	Optional property for a SnapMirror endpoint. Specifies the list of source LUNs and optionally list of destination LUNs during SnapMirror Consistency Group LUN restore operation.
path	string	ONTAP FlexVol/FlexGroup - svm1:volume1 ONTAP SVM - svm1: ONTAP Consistency Group - svm1:/cg/cg_name ONTAP S3 - svm1:/bucket/bucket1 NON-ONTAP - objstore1:/objstore <ul style="list-style-type: none"> <li>• example: svm1:volume1</li> <li>• Introduced in: 9.14</li> <li>• x-nullable: true</li> </ul>
svm	<a href="#">svm</a>	SVM, applies only to SVM-scoped objects.

#### svmdr\_volumes



Name	Type	Description
name	string	The name of the volume.

#### transfer

Basic information on the current transfer or the last transfer if there is no active transfer at the time of the request.

Name	Type	Description
bytes_transferred	integer	Total bytes transferred in the current or last successful transfer.
end_time	string	End time of the last transfer.
last_updated_time	string	Last updated time of the bytes transferred in the current transfer.
state	string	
total_duration	string	Elapsed time to transfer all snapshots for the last successful transfer.
type	string	Specifies the operation type of the current transfer on the relationship. The <i>initialize</i> transfer occurs when the relationship state changes from "uninitialized" to "snapmirrored" or "in_sync". The <i>update</i> transfer occurs when snapshots are being transferred from the source endpoint to the destination endpoint as part of a scheduled or manual update. The <i>resync</i> transfer occurs when the relationship state changes from "broken_off" to "snapmirrored" or "in_sync". The <i>restore</i> transfer occurs when a snapshot is being restored from a destination endpoint to another endpoint.
uuid	string	Transfer UUID. This property is applicable only for active transfers.

#### transfer\_schedule

Schedule used to update asynchronous relationships. This "transfer\_schedule" overrides the

"transfer\_schedule" set on the SnapMirror relationship's policy. To remove the "transfer\_schedule", set its value to null (no-quotes). Only cron schedules are supported for SnapMirror.

Name	Type	Description
name	string	Job schedule name
uuid	string	Job schedule UUID

snapmirror\_error

SnapMirror error

Name	Type	Description
arguments	array[string]	Arguments present in the error message encountered.
code	string	Error code
message	string	Error message

snapmirror\_relationship

SnapMirror relationship information. The SnapMirror relationship can be either "async" or "sync" based on the type of SnapMirror policy associated with the relationship. The source and destination endpoints of a SnapMirror relationship must be of the same type, for example, if the source endpoint is a FlexVol volume then the destination endpoint must be a FlexVol volume. The SnapMirror policy type "async" can be used when the SnapMirror relationship has FlexVol volume or FlexGroup volume or SVM as the endpoint. The SnapMirror policy type "sync" can be used when the SnapMirror relationship has FlexVol volume as the endpoint. The SnapMirror policy type "sync" with "sync\_type" as "automated\_failover" can be used when the SnapMirror relationship has Consistency Group as the endpoint.

Name	Type	Description
backoff_level	string	Specifies the SnapMirror backoff level due to Client Ops for FlexVol SnapMirror relationships.

Name	Type	Description
create_destination	<a href="#">snapmirror_destination_creation</a>	<p>Use this object to provision the destination endpoint when establishing a SnapMirror relationship for a FlexVol volume, FlexGroup volume, SVM, Consistency Group or ONTAP S3 Bucket. Given a source endpoint, the destination endpoint is provisioned in the SVM specified in the "destination.path" property. While protecting an SVM, the SVM destination endpoint can only be provisioned on the local cluster. To provision the SVM destination endpoint use the optional "source.cluster.name" property to specify the remote cluster name or use the optional "source.cluster.uuid" property to specify the remote cluster UUID. When "create_destination.enabled" option is specified while making a POST for a SnapMirror relationship, the relationship can be automatically initialized by setting the "state" either to "snapmirrored" when the policy is of type "async" or to "in_sync" when the policy is of type "sync". The "destination.path" property must specify the destination endpoint path. For example, for FlexVol volume and FlexGroup volume, the "destination.path" can be specified as &lt;destination-SVM-name:dp-volume-name&gt;, for SVM data protection, the "destination.path" must be specified as &amp;lt;destination-SVM-name:&amp;gt;, and for Consistency Group, the "destination.path" must be specified as &amp;lt;destination-SVM-name:/cg/consistency-group-name&amp;gt; along with the "destination.consistency_group_volumes" or "destination.luns" property to indicate the list of destination volumes or LUNs of type "DP" in the Consistency Group. For a FlexVol volume, a FlexGroup volume, Consistency Group or a Bucket destination endpoint, the properties in this object can be specified either</p>

Name	Type	Description
destination	<a href="#">snapmirror_endpoint</a>	<p>Endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to create the destination SVM endpoint or to establish an SVM DR relationship must have the property "cluster" populated with the remote cluster details. A POST request to create the destination FlexVol volume, FlexGroup volume, Consistency Group, ONTAP S3 bucket and NON-ONTAP object-store endpoints can optionally specify the "cluster" property when the source SVM and the destination SVM are peered. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.6</li> </ul>
exported_snapshot	string	Snapshot exported to clients on destination.
group_type	string	Specifies the group type of the top level SnapMirror relationship. The volume relationships are shown as <i>none</i> , the SVM DR relationships are shown as <i>svm_dr</i> , the Consistency Group relationships are shown as <i>consistency_group</i> , and the FlexGroup volume relationships are shown as <i>flexgroup</i> .
healthy	boolean	Is the relationship healthy?

Name	Type	Description
identity_preservation	string	Specifies which configuration of the source SVM is replicated to the destination SVM. This property is applicable only for SVM data protection with "async" policy type. This "identity_preservation" overrides the "identity_preservation" set on the SnapMirror relationship's policy.
io_serving_copy	string	Specifies the sites serving I/O for the SnapMirror active sync relationship.
lag_time	string	Time since the exported snapshot was created.
last_transfer_network_compression_ratio	string	Specifies the compression ratio achieved for the data sent over the wire with network compression enabled for the last successful transfer.
last_transfer_type	string	Specifies the operation type of the last transfer that occurred on the relationship. The <i>initialize</i> transfer occurs when the relationship state changes from uninitialized to snapmirrored or in_sync. The <i>update</i> transfer occurs when the snapshots are transferred from the source endpoint to the destination endpoint as part of scheduled or manual update. The <i>resync</i> transfer occurs when the relationship state changes from broken_off to snapmirrored or in_sync. The <i>restore</i> transfer occurs when the snapshot is restored from a destination endpoint to another endpoint.
master_bias_activated_site	string	Specifies the Master Bias Activated Site for the SnapMirror active sync relationship.

Name	Type	Description
policy	<a href="#">policy</a>	Basic policy information of the relationship.
preferred_site	string	Specifies the Primary Site of the SnapMirror active sync relationship.
preserve	boolean	Set to true on resync to preserve snapshots on the destination that are newer than the latest common snapshot. This property is applicable only for relationships with FlexVol volume or FlexGroup volume endpoints and when the PATCH state is being changed to "snapmirrored".
quick_resync	boolean	Set to true to reduce resync time by not preserving storage efficiency. This property is applicable only for relationships with FlexVol volume endpoints and SVMDR relationships when the PATCH state is being changed to "snapmirrored".
recover_after_break	boolean	Set to true to recover from a failed SnapMirror break operation on a FlexGroup volume relationship. This restores all destination FlexGroup constituent volumes to the latest snapshot, and any writes to the read-write constituents are lost. This property is applicable only for SnapMirror relationships with FlexGroup volume endpoints and when the PATCH state is being changed to "broken_off".
restore	boolean	Set to true to create a relationship for restore. To trigger restore-transfer, use transfers POST on the restore relationship. SnapMirror relationships with the policy type "async" can be restored. SnapMirror relationships with the policy type "sync" cannot be restored.

Name	Type	Description
restore_to_snapshot	string	Specifies the snapshot to restore to on the destination during the break operation. This property is applicable only for SnapMirror relationships with FlexVol volume endpoints and when the PATCH state is being changed to "broken_off".
source	<a href="#">snapmirror_source_endpoint</a>	Source endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
state	string	<p>State of the relationship. To initialize the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". To break the relationship, PATCH the state to "broken_off" for relationships with a policy of type "async" or "sync".</p> <p>SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" cannot be "broken_off". To resync the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync".</p> <p>SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" can be in "broken_off" state due to a failed attempt of SnapMirror failover. To pause the relationship, suspending further transfers, PATCH the state to "paused" for relationships with a policy of type "async" or "sync". SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" cannot be "paused". To resume transfers for a paused relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". The entries "in_sync", "out_of_sync", "synchronizing", "expanding", and "shrinking" are only applicable to relationships with a policy of type "sync". A PATCH call on the state change only triggers the transition to the specified state. You must poll on the "state", "healthy" and "unhealthy_reason" properties using a GET request to determine if the transition is successful. To automatically initialize the relationship when specifying "create_destination" property, set the state to "snapmirrored" for</p>



Name	Type	Description
svmdr_volumes	array[ <a href="#">svmdr_volumes</a> ]	Specifies the list of constituent FlexVol volumes and FlexGroup volumes for an SVM DR SnapMirror relationship. FlexGroup constituents are not considered.
throttle	integer	Throttle, in KBs per second. This "throttle" overrides the "throttle" set on the SnapMirror relationship's policy. If neither of these are set, defaults to 0, which is interpreted as unlimited.
total_transfer_bytes	integer	Cumulative bytes transferred for the relationship.
total_transfer_duration	string	Indicates the cumulative duration of all transfers since the last aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship.
transfer	<a href="#">transfer</a>	Basic information on the current transfer or the last transfer if there is no active transfer at the time of the request.
transfer_schedule	<a href="#">transfer_schedule</a>	Schedule used to update asynchronous relationships. This "transfer_schedule" overrides the "transfer_schedule" set on the SnapMirror relationship's policy. To remove the "transfer_schedule", set its value to null (no-quotes). Only cron schedules are supported for SnapMirror.
unhealthy_reason	array[ <a href="#">snapmirror_error</a> ]	Reason the relationship is not healthy. It is a concatenation of up to four levels of error messages.
uuid	string	Unique identifier of the SnapMirror relationship.

job\_link

Name	Type	Description
uuid	string	The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation.

returned\_error

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

## Delete a SnapMirror relationship

DELETE /snapmirror/relationships/{uuid}

**Introduced In:** 9.6

Deletes a SnapMirror relationship.

### Important notes

- The "destination\_only", "source\_only", and "source\_info\_only" flags are mutually exclusive. If no flag is specified, the relationship is deleted from both the source and destination and all common snapshots between the source and destination are also deleted.
- For a restore relationship, the call must be executed on the cluster containing the destination endpoint without specifying the destination\_only, source\_only, or source\_info\_only parameters.
- Additionally, ensure that there are no ongoing transfers on a restore relationship before calling this API.
- The "failover", "force-failover" and "failback" query parameters are only applicable for SVM-DR SnapMirror relationships.
- When a SnapMirror relationship associated with a pair of source and destination Consistency Groups is deleted, the corresponding Consistency Groups on the source and destination clusters are not automatically deleted and remain in place.
- The "delete\_lun\_maps\_in\_destination" query parameter is applicable only for SnapMirror active sync relationships.
- The "unmap\_namespace" query parameter is applicable only for NVMe SnapMirror active sync relationships.

## Related ONTAP commands

- `snapmirror delete`
- `snapmirror release`

## Examples

The following examples show how to delete the relationship from both the source and destination, the destination only, and the source only.

Deleting the relationship from both the source and destination. This API must be run on the cluster containing the destination endpoint.

```
DELETE "/api/snapmirror/relationships/4512b2d2-fd60-11e8-8929-005056bbfe52"
```

Deleting the relationship on the destination only. This API must be run on the cluster containing the destination endpoint.

```
DELETE "/api/snapmirror/relationships/fd1e0697-02ba-11e9-acc7-005056a7697f/?destination_only=true"
```

Deleting the relationship on the source only. This API must be run on the cluster containing the source endpoint.

```
DELETE "/api/snapmirror/relationships/93e828ba-02bc-11e9-acc7-005056a7697f/?source_only=true"
```

Deleting the source information only. This API must be run on the cluster containing the source endpoint. This does not delete the common snapshots between the source and destination.

```
DELETE "/api/snapmirror/relationships/caf545a2-fc60-11e8-aa13-005056a707ff/?source_info_only=true"
```

Deleting the relationship from source and destination cluster along with deleting the LUN maps for the volumes of the CG in destination cluster. This API must be run on the cluster containing the destination endpoint.

```
DELETE "/api/snapmirror/relationships/feda8f5e-e29e-11ed-94aa-005056a78ce2/?delete_lun_maps_in_destination=true"
```

Deleting the relationship from destination cluster along with deleting the namespace maps for the volumes of the CG in destination cluster. This API must be run on the cluster containing the destination endpoint.

```
DELETE "/api/snapmirror/relationships/feda8f5e-e29e-11ed-94aa-005056a78ce2/?unmap_namespace=true"
```

## Learn more

- [DOC /snapmirror/relationships](#)

## Parameters

Name	Type	In	Required	Description
uuid	string	path	True	SnapMirror relationship UUID
destination_only	boolean	query	False	Deletes a relationship on the destination only. This parameter is applicable only when the call is executed on the cluster that contains the destination endpoint.
source_only	boolean	query	False	Deletes a relationship on the source only. This parameter is applicable only when the call is executed on the cluster that contains the source endpoint. In the case of a relationship that uses SnapMirror policy of type 'continuous', this parameter is used to delete the source end of the relationship while ignoring errors when the destination endpoint is not reachable.

Name	Type	In	Required	Description
source_info_only	boolean	query	False	Deletes relationship information on the source only. This parameter is applicable only when the call is executed on the cluster that contains the source endpoint.
delete_lun_maps_in_destination	boolean	query	False	<p>Deletes LUN mapping for the volumes of the CG in destination.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.14</li> <li>• Default value:</li> </ul>
unmap_namespace	boolean	query	False	<p>Deletes namespace mapping and then deletes the subsystems.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.17</li> <li>• Default value:</li> </ul>

Name	Type	In	Required	Description
return_timeout	integer	query	False	<p>The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.</p> <ul style="list-style-type: none"> <li>• Default value: 0</li> <li>• Max value: 120</li> <li>• Min value: 0</li> </ul>

## Response

Status: 200, Ok

Name	Type	Description
job	<a href="#">job_link</a>	

## Example response

```
{
  "job": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "string"
  }
}
```

## Response

Status: 202, Accepted

## Error

Status: Default

### ONTAP Error Response codes

Error code	Description
13303825	Could not retrieve information for the SnapMirror policy type
13303814	Could not retrieve the source or destination SVM UUID
13303815	Could not retrieve information for the peer cluster
13303822	SnapMirror release has failed
13303813	SnapMirror release was successful but delete has failed
13303854	Cleanup of restore relationship failed
13303855	DELETE call on a restore relationship does not support the given flags
13303865	Deleting the specified SnapMirror policy is not supported.
6619715	Modification of relationship is in progress. Retry the command after a few minutes.

Error code	Description
6621470	SnapMirror delete failed on the relationship because "unmap-namespace" is not set. Set the "unmap-namespace" to true for NVMe SnapMirror active sync relationships.

Name	Type	Description
error	<a href="#">returned_error</a>	

### 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	<a href="#">href</a>	

job\_link

Name	Type	Description
_links	<a href="#">_links</a>	
uuid	string	The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation.

error\_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

returned\_error

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

## Retrieve a SnapMirror relationship

GET /snapmirror/relationships/{uuid}

**Introduced In:** 9.6

Retrieves a SnapMirror relationship.

## Related ONTAP commands

- `snapmirror show`
- `snapmirror list-destinations`

## Expensive properties

- `source.consistency_group_volumes.name`
- `destination.consistency_group_volumes.name`

## Example

```
GET "/api/snapmirror/relationships/caf545a2-fc60-11e8-aa13-005056a707ff/"
```

## Learn more

- [DOC /snapmirror/relationships](#)

## Parameters

Name	Type	In	Required	Description
uuid	string	path	True	SnapMirror relationship UUID
list_destinations_only	boolean	query	False	Set to true to show relationships from the source only.
fields	array[string]	query	False	Specify the fields to return.

## Response

```
Status: 200, Ok
```

Name	Type	Description
<code>_links</code>	<a href="#">_links</a>	

Name	Type	Description
backoff_level	string	Specifies the SnapMirror backoff level due to Client Ops for FlexVol SnapMirror relationships.
consistency_group_failover	<a href="#">snapmirror_consistency_group_failover</a>	<p>SnapMirror Consistency Group failover information. The SnapMirror Consistency Group failover can be a planned or an unplanned operation. Only active SnapMirror Consistency Group failover operation progress can be monitored using this object. In case of an error during the failover operation, the property "consistency_group_failover.error" holds the reason for the error. ONTAP automatically retries any failed SnapMirror Consistency Group failover operation.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.8</li> <li>• readOnly: 1</li> </ul>
destination	<a href="#">snapmirror_endpoint</a>	<p>Endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to create the destination SVM endpoint or to establish an SVM DR relationship must have the property "cluster" populated with the remote cluster details. A POST request to create the destination FlexVol volume, FlexGroup volume, Consistency Group, ONTAP S3 bucket and NON-ONTAP object-store endpoints can optionally specify the "cluster" property when the source SVM and the destination SVM are peered. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.6</li> </ul>

Name	Type	Description
exported_snapshot	string	Snapshot exported to clients on destination.
group_type	string	Specifies the group type of the top level SnapMirror relationship. The volume relationships are shown as <i>none</i> , the SVMDR relationships are shown as <i>svm_dr</i> , the Consistency Group relationships are shown as <i>consistency_group</i> , and the FlexGroup volume relationships are shown as <i>flexgroup</i> .
healthy	boolean	Is the relationship healthy?
identity_preservation	string	Specifies which configuration of the source SVM is replicated to the destination SVM. This property is applicable only for SVM data protection with "async" policy type. This "identity_preservation" overrides the "identity_preservation" set on the SnapMirror relationship's policy.
io_serving_copy	string	Specifies the sites serving I/O for the SnapMirror active sync relationship.
lag_time	string	Time since the exported snapshot was created.
last_transfer_network_compression_ratio	string	Specifies the compression ratio achieved for the data sent over the wire with network compression enabled for the last successful transfer.

Name	Type	Description
last_transfer_type	string	Specifies the operation type of the last transfer that occurred on the relationship. The <i>initialize</i> transfer occurs when the relationship state changes from uninitialized to snapmirrored or in_sync. The <i>update</i> transfer occurs when the snapshots are transferred from the source endpoint to the destination endpoint as part of scheduled or manual update. The <i>resync</i> transfer occurs when the relationship state changes from broken_off to snapmirrored or in_sync. The <i>restore</i> transfer occurs when the snapshot is restored from a destination endpoint to another endpoint.
master_bias_activated_site	string	Specifies the Master Bias Activated Site for the SnapMirror active sync relationship.
policy	<a href="#">policy</a>	Basic policy information of the relationship.
preferred_site	string	Specifies the Primary Site of the SnapMirror active sync relationship.
preserve	boolean	Set to true on resync to preserve snapshots on the destination that are newer than the latest common snapshot. This property is applicable only for relationships with FlexVol volume or FlexGroup volume endpoints and when the PATCH state is being changed to "snapmirrored".
quick_resync	boolean	Set to true to reduce resync time by not preserving storage efficiency. This property is applicable only for relationships with FlexVol volume endpoints and SVMDR relationships when the PATCH state is being changed to "snapmirrored".

Name	Type	Description
recover_after_break	boolean	Set to true to recover from a failed SnapMirror break operation on a FlexGroup volume relationship. This restores all destination FlexGroup constituent volumes to the latest snapshot, and any writes to the read-write constituents are lost. This property is applicable only for SnapMirror relationships with FlexGroup volume endpoints and when the PATCH state is being changed to "broken_off".
restore	boolean	Set to true to create a relationship for restore. To trigger restore-transfer, use transfers POST on the restore relationship. SnapMirror relationships with the policy type "async" can be restored. SnapMirror relationships with the policy type "sync" cannot be restored.
restore_to_snapshot	string	Specifies the snapshot to restore to on the destination during the break operation. This property is applicable only for SnapMirror relationships with FlexVol volume endpoints and when the PATCH state is being changed to "broken_off".
source	<a href="#">snapmirror_source_endpoint</a>	Source endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
state	string	<p>State of the relationship. To initialize the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". To break the relationship, PATCH the state to "broken_off" for relationships with a policy of type "async" or "sync". SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" cannot be "broken_off". To resync the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" can be in "broken_off" state due to a failed attempt of SnapMirror failover. To pause the relationship, suspending further transfers, PATCH the state to "paused" for relationships with a policy of type "async" or "sync". SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" cannot be "paused". To resume transfers for a paused relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". The entries "in_sync", "out_of_sync", "synchronizing", "expanding", and "shrinking" are only applicable to relationships with a policy of type "sync". A PATCH call on the state change only triggers the transition to the specified state. You must poll on the "state", "healthy" and "unhealthy_reason" properties using a GET request to determine if the transition is successful. To automatically initialize the relationship when specifying "create_destination" property, set the state to "snapmirrored" for relationships with a policy of type</p>

Name	Type	Description
svmdr_volumes	array[ <a href="#">svmdr_volumes</a> ]	Specifies the list of constituent FlexVol volumes and FlexGroup volumes for an SVM DR SnapMirror relationship. FlexGroup constituents are not considered.
throttle	integer	Throttle, in KBs per second. This "throttle" overrides the "throttle" set on the SnapMirror relationship's policy. If neither of these are set, defaults to 0, which is interpreted as unlimited.
total_transfer_bytes	integer	Cumulative bytes transferred for the relationship.
total_transfer_duration	string	Indicates the cumulative duration of all transfers since the last aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship.
transfer	<a href="#">transfer</a>	Basic information on the current transfer or the last transfer if there is no active transfer at the time of the request.
transfer_schedule	<a href="#">transfer_schedule</a>	Schedule used to update asynchronous relationships. This "transfer_schedule" overrides the "transfer_schedule" set on the SnapMirror relationship's policy. To remove the "transfer_schedule", set its value to null (no-quotes). Only cron schedules are supported for SnapMirror.
unhealthy_reason	array[ <a href="#">snapmirror_error</a> ]	Reason the relationship is not healthy. It is a concatenation of up to four levels of error messages.
uuid	string	Unique identifier of the SnapMirror relationship.



## Example response

```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "backoff_level": "medium",
  "consistency_group_failover": {
    "error": {
      "arguments": [
        {
          "code": "string",
          "message": "string"
        }
      ],
      "code": "4",
      "message": "entry doesn't exist"
    },
    "state": "string",
    "status": {
      "code": "string",
      "message": "string"
    },
    "type": "string"
  },
  "destination": {
    "cluster": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "cluster1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "consistency_group_volumes": [
      {
        "name": "volume1"
      }
    ],
    "luns": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      }
    }
  }
}
```

```

    }
  },
  "name": "/vol/volume1/lun1",
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"path": "svm1:volume1",
"svm": {
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "name": "svm1",
  "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
}
},
"exported_snapshot": "string",
"group_type": "consistency_group",
"identity_preservation": "string",
"io_serving_copy": "C1_sti85-vsim-ucs209a_cluster, C1_sti85-vsim-ucs209c_cluster",
"lag_time": "PT8H35M42S",
"last_transfer_network_compression_ratio": 61,
"last_transfer_type": "initialize",
"master_bias_activated_site": "C1_sti85-vsim-ucs209a_cluster",
"policy": {
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "name": "Asynchronous",
  "type": "string",
  "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
},
"preferred_site": "C1_sti85-vsim-ucs209a_cluster",
"restore_to_snapshot": "string",
"source": {
  "cluster": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "cluster1",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  }
}

```

```

},
"consistency_group_volumes": [
  {
    "name": "volume1"
  }
],
"luns": {
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "name": "/vol/volume1/lun1",
  "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
},
"path": "svm1:volume1",
"svm": {
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "name": "svm1",
  "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
}
},
"state": "snapmirrored",
"svmdr_volumes": [
  {
    "name": "volume1"
  }
],
"throttle": 0,
"total_transfer_bytes": 1098210312,
"total_transfer_duration": "PT3M21S",
"transfer": {
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  }
},
"end_time": "2020-12-02 18:36:19 -0800",
"last_updated_time": "2023-09-14 15:39:19 -0700",
"state": "string",
"total_duration": "PT28M41S",
"type": "initialize",

```

```

    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "transfer_schedule": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "weekly",
    "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
  },
  "unhealthy_reason": [
    {
      "arguments": [],
      "code": "6621444",
      "message": "Failed to complete update operation on one or more
item relationships."
    },
    {
      "arguments": [],
      "code": "6621445",
      "message": "Group Update failed"
    }
  ],
  "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
}

```

## Error

Status: Default

### ONTAP Error Response codes

Error code	Description
13303825	Could not retrieve information for the SnapMirror policy type
13303817	Unknown value for the Snapmirror State

Name	Type	Description
error	<a href="#">returned_error</a>	

### 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	<a href="#">href</a>	

error\_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

SnapMirror Consistency Group failover error message.

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message

status

Name	Type	Description
code	string	Status code
message	string	SnapMirror Consistency Group failover status.

snapmirror\_consistency\_group\_failover

SnapMirror Consistency Group failover information. The SnapMirror Consistency Group failover can be a planned or an unplanned operation. Only active SnapMirror Consistency Group failover operation progress can be monitored using this object. In case of an error during the failover operation, the property "consistency\_group\_failover.error" holds the reason for the error. ONTAP automatically retries any failed SnapMirror Consistency Group failover operation.

Name	Type	Description
error	<a href="#">error</a>	SnapMirror Consistency Group failover error message.
state	string	SnapMirror Consistency Group failover state.
status	<a href="#">status</a>	
type	string	SnapMirror Consistency Group failover type.

#### bucket\_retention

Specifies the retention-mode and default retention period configured on the destination bucket.

Name	Type	Description
default_period	string	Specifies the default retention period that is applied to objects while committing them to the WORM state without an associated retention period. The retention period can be in years, or days. The retention period value represents a duration and must be specified in the ISO-8601 duration format. A period specified for years and days is represented in the ISO-8601 format as "P<num>Y" and "P<num>D" respectively, for example "P10Y" represents a duration of 10 years. The period string must contain only a single time element that is, either years, or days. A duration which combines different periods is not supported, for example "P1Y10D" is not supported.</num></num>

Name	Type	Description
mode	string	The lock mode of the bucket. compliance &dash; A SnapLock Compliance (SLC) bucket provides the highest level of WORM protection and an administrator cannot destroy a compliance bucket if it contains unexpired WORM objects. governance &dash; An administrator can delete a Governance bucket. no_lock &dash; Indicates the bucket does not support object locking.

#### storage\_service

Name	Type	Description
enabled	boolean	This property indicates whether to create the destination endpoint using storage service. This property is supported for Unified ONTAP destination endpoints only.  <ul style="list-style-type: none"> <li>• Introduced in: 9.7</li> <li>• x-nullable: true</li> </ul>
enforce_performance	boolean	Optional property to enforce storage service performance on the destination endpoint. This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. This property is supported for Unified ONTAP destination endpoints only.  <ul style="list-style-type: none"> <li>• Default value: 1</li> <li>• Introduced in: 9.7</li> <li>• x-nullable: true</li> </ul>



Name	Type	Description
name	string	<p>Optional property to specify the storage service name for the destination endpoint. This property is considered when the property "create_destination.storage_service.enabled" is set to "true". When the property "create_destination.storage_service.enabled" is set to "true" and the "create_destination.storage_service.name" for the endpoint is not specified, then ONTAP selects the highest storage service available on the cluster to provision the destination endpoint. This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>enum: ["extreme", "performance", "value"]</li> <li>Introduced in: 9.6</li> <li>x-nullable: true</li> </ul>

tiering



Name	Type	Description
policy	string	<p>Optional property to specify the destination endpoint's tiering policy when "create_destination.tiering.supported" is set to "true". This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. This property determines whether the user data blocks of the destination endpoint in a FabricPool will be tiered to the cloud store when they become cold. FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of the destination endpoint volume blocks increases if they are accessed frequently and decreases when they are not. all – This policy allows tiering of both destination endpoint snapshots and the user transferred data blocks to the cloud store as soon as possible by ignoring the temperature on the volume blocks. This tiering policy is not applicable for Consistency Group destination endpoints or for synchronous relationships. auto – This policy allows tiering of both destination endpoint snapshots and the active file system user data to the cloud store none – Destination endpoint volume blocks will not be tiered to the cloud store. snapshot_only – This policy allows tiering of only the destination endpoint volume snapshots not associated with the active file system. The default tiering policy is "snapshot_only" for a FlexVol volume and "none" for a FlexGroup volume. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>enum: ["all", "auto", "none", "snapshot_only"]</li> <li>Introduced in: 9.6</li> </ul> <p>x-nullable: true</p>

Name	Type	Description
supported	boolean	<p>Optional property to enable provisioning of the destination endpoint volumes on FabricPool aggregates. This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. Only FabricPool aggregates are used if this property is set to "true" and only non FabricPool aggregates are used if this property is set to "false". Tiering support for a FlexGroup volume can be changed by moving all of the constituents to the required aggregates. Note that in order to tier data, not only do the destination endpoint volumes need to support tiering by using FabricPools, the "create_destination.tiering.policy" must not be "none". A destination endpoint that uses FabricPools but has a tiering "policy" of "none" supports tiering but will not tier any data. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.6</li> <li>• x-nullable: true</li> </ul>

#### snapmirror\_destination\_creation

Use this object to provision the destination endpoint when establishing a SnapMirror relationship for a FlexVol volume, FlexGroup volume, SVM, Consistency Group or ONTAP S3 Bucket. Given a source endpoint, the destination endpoint is provisioned in the SVM specified in the "destination.path" property. While protecting an SVM, the SVM destination endpoint can only be provisioned on the local cluster. To provision the SVM destination endpoint use the optional "source.cluster.name" property to specify the remote cluster name or use the optional "source.cluster.uuid" property to specify the remote cluster UUID. When "create\_destination.enabled" option is specified while making a POST for a SnapMirror relationship, the relationship can be automatically initialized by setting the "state" either to "snapmirrored" when the policy is of type "async" or to "in\_sync" when the policy is of type "sync". The "destination.path" property must specify the destination endpoint path. For example, for FlexVol volume and FlexGroup volume, the "destination.path" can be specified as <destination-svm-name:dp-volume-name>, for SVM data protection, the "destination.path" must be specified as &lt;destination-svm-name:&gt;, and for Consistency Group, the "destination.path" must be specified as &lt;destination-svm-name:&gt; along with the "destination.consistency\_group\_volumes" or "destination.luns" property to indicate the list of destination volumes or LUNs of type "DP" in the Consistency Group. For a FlexVol volume, a FlexGroup volume, Consistency Group or a Bucket destination endpoint, the properties in this object can be specified either from the source or the destination cluster. For an SVM destination endpoint, the

properties in this object can be specified from the destination cluster only. This object is not supported for non ONTAP endpoints. While protecting a S3 Bucket, the optional "size" property can be used to create ONTAP S3 Bucket destination endpoint of the specified size.</destination-svm-name:></destination-svm-name:></destination-svm-name:dp-volume-name>

Name	Type	Description
bucket_retention	<a href="#">bucket_retention</a>	Specifies the retention-mode and default retention period configured on the destination bucket.
enabled	boolean	Optional property to create the destination endpoint when establishing a SnapMirror relationship. It is assumed to be "false" if no other property is set and assumed to be "true" if any other property is set.
size	integer	Optional property to specify the size of destination endpoint in bytes. This property is applicable only to ONTAP S3 Bucket endpoints. The minimum size for S3 bucket is 80MB and maximum size is 64TB. If not specified, system will create destination with default size of 800GB. <ul style="list-style-type: none"> <li>• Introduced in: 9.10</li> <li>• x-nullable: true</li> </ul>
snapshot_locking_enabled	boolean	Optional property to create the destination endpoint with snapshot locking enabled when establishing a SnapMirror relationship. This property is applicable to FlexVol volumes and FlexGroup volumes.
storage_service	<a href="#">storage_service</a>	
tiering	<a href="#">tiering</a>	

#### cluster

Name	Type	Description
_links	<a href="#">_links</a>	
name	string	
uuid	string	

### consistency\_group\_volumes

Name	Type	Description
name	string	The name of the volume.

### luns

Optional property for a SnapMirror endpoint. Specifies the list of source LUNs and optionally list of destination LUNs during SnapMirror Consistency Group LUN restore operation.

Name	Type	Description
_links	<a href="#">_links</a>	
name	string	<p>The name of a LUN. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.</p> <p>LUN names are paths of the form "/vol/&lt;volume&gt;[/&lt;qtree&gt;]/&lt;names pace&gt;" where the qtree name is optional.</p>
uuid	string	The unique identifier of the LUN.

### svm

SVM, applies only to SVM-scoped objects.

Name	Type	Description
_links	<a href="#">_links</a>	
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.

### snapmirror\_endpoint

Endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to create the destination SVM endpoint or to establish an SVM DR relationship must have the property "cluster" populated with the remote cluster details. A POST request to create the destination FlexVol volume, FlexGroup volume, Consistency Group, ONTAP S3 bucket and NON-ONTAP object-store endpoints can optionally specify the "cluster" property when the source SVM and the destination SVM are peered. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination

SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
cluster	<a href="#">cluster</a>	
consistency_group_volumes	array[ <a href="#">consistency_group_volumes</a> ]	This property specifies the list of FlexVol volumes or LUNs of a Consistency Group. Optional on the ASA r2 platform. Mandatory for all other platforms.
luns	<a href="#">luns</a>	Optional property for a SnapMirror endpoint. Specifies the list of source LUNs and optionally list of destination LUNs during SnapMirror Consistency Group LUN restore operation.
path	string	ONTAP FlexVol/FlexGroup - svm1:volume1 ONTAP SVM - svm1: ONTAP Consistency Group - svm1:/cg/cg_name ONTAP S3 - svm1:/bucket/bucket1 NON-ONTAP - objstore1:/objstore <ul style="list-style-type: none"><li>• example: svm1:volume1</li><li>• Introduced in: 9.6</li><li>• x-nullable: true</li></ul>
svm	<a href="#">svm</a>	SVM, applies only to SVM-scoped objects.

## policy

Basic policy information of the relationship.

Name	Type	Description
_links	<a href="#">_links</a>	
name	string	Name of the SnapMirror policy.
type	string	
uuid	string	Unique identifier of the SnapMirror policy.

## snapmirror\_source\_endpoint

Source endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when

the endpoint is on a remote cluster. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
cluster	<a href="#">cluster</a>	
consistency_group_volumes	array[ <a href="#">consistency_group_volumes</a> ]	This property specifies the list of FlexVol volumes or LUNs of a Consistency Group. Optional on the ASA r2 platform. Mandatory for all other platforms.
luns	<a href="#">luns</a>	Optional property for a SnapMirror endpoint. Specifies the list of source LUNs and optionally list of destination LUNs during SnapMirror Consistency Group LUN restore operation.
path	string	ONTAP FlexVol/FlexGroup - svm1:volume1 ONTAP SVM - svm1: ONTAP Consistency Group - svm1:/cg/cg_name ONTAP S3 - svm1:/bucket/bucket1 NON-ONTAP - objstore1:/objstore <ul style="list-style-type: none"><li>• example: svm1:volume1</li><li>• Introduced in: 9.14</li><li>• x-nullable: true</li></ul>
svm	<a href="#">svm</a>	SVM, applies only to SVM-scoped objects.

#### svmdr\_volumes

Name	Type	Description
name	string	The name of the volume.

#### transfer

Basic information on the current transfer or the last transfer if there is no active transfer at the time of the request.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	



Name	Type	Description
bytes_transferred	integer	Total bytes transferred in the current or last successful transfer.
end_time	string	End time of the last transfer.
last_updated_time	string	Last updated time of the bytes transferred in the current transfer.
state	string	
total_duration	string	Elapsed time to transfer all snapshots for the last successful transfer.
type	string	Specifies the operation type of the current transfer on the relationship. The <i>initialize</i> transfer occurs when the relationship state changes from "uninitialized" to "snapmirrored" or "in_sync". The <i>update</i> transfer occurs when snapshots are being transferred from the source endpoint to the destination endpoint as part of a scheduled or manual update. The <i>resync</i> transfer occurs when the relationship state changes from "broken_off" to "snapmirrored" or "in_sync". The <i>restore</i> transfer occurs when a snapshot is being restored from a destination endpoint to another endpoint.
uuid	string	Transfer UUID. This property is applicable only for active transfers.

#### transfer\_schedule

Schedule used to update asynchronous relationships. This "transfer\_schedule" overrides the "transfer\_schedule" set on the SnapMirror relationship's policy. To remove the "transfer\_schedule", set its value to null (no-quotes). Only cron schedules are supported for SnapMirror.

Name	Type	Description
_links	<a href="#">_links</a>	
name	string	Job schedule name

Name	Type	Description
uuid	string	Job schedule UUID

snapmirror\_error

SnapMirror error

Name	Type	Description
arguments	array[string]	Arguments present in the error message encountered.
code	string	Error code
message	string	Error message

returned\_error

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

## Update a SnapMirror relationship

PATCH /snapmirror/relationships/{uuid}

**Introduced In:** 9.6

Updates a SnapMirror relationship. This API is used to initiate SnapMirror operations such as "initialize", "resync", "break", "quiesce", and "resume" by specifying the appropriate value for the "state" field. It is also used to modify the SnapMirror policy associated with the specified relationship. Additionally, a SnapMirror relationship can be failed over to the destination endpoint or a failed over SnapMirror relationship can be failed back to the original state or a SnapMirror relationship direction can be reversed using this API. This API can also be used to expand the SnapMirror active sync relationship with the specified source and destination volumes.

To initialize the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or "in\_sync" for relationships with a policy of type "sync".

To break the relationship or to failover to the destination endpoint and start serving data from the destination

endpoint, PATCH the state to "broken\_off" for relationships with a policy of type "async" or "sync". SnapMirror relationships with the policy type as "sync" and sync\_type as "automated\_failover" cannot be "broken\_off".

To resync the broken relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or "in\_sync" for relationships with a policy of type "sync".

To failback the failed over relationship and start serving data from the source endpoint, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or "in\_sync" for relationships with a policy of type "sync" and set the query flag "failback" as "true". SnapMirror relationships with the policy type as "sync" and sync\_type as "automated\_failover" can be in "broken\_off" state due to a failed attempt of automated SnapMirror failover operation.

To pause the relationship, suspending further transfers, PATCH the state to "paused" for relationships with a policy of type "async" or "sync". SnapMirror relationships with the policy type as "sync" and sync\_type as "automated\_failover" cannot be "paused".

To resume transfers for a paused relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or "in\_sync" for relationships with a policy of type "sync".

To reverse the direction of the relationship, PATCH the "source.path" with the destination endpoint and the "destination.path" with the source endpoint and the relationship state to "snapmirrored" for relationships with a policy of type "async" or "in\_sync" for relationships with a policy of type "sync". For relationships with a policy of type "async" and relationship state as "snapmirrored", stop IO on the source endpoint and perform a SnapMirror transfer POST operation before reversing the direction of the relationship to prevent any loss of data.

The values "in\_sync", "out\_of\_sync", and "synchronizing" are only applicable to relationships with a policy of type "sync".

When "transfer\_schedule" is specified along with "state" during PATCH, first the schedule is modified on the relationship and then the respective SnapMirror operation is initiated. The "transfer\_schedule" specified is used to update asynchronous relationships.

When "throttle" is specified along with "state" during PATCH, first the throttle is modified on the relationship, which will be used by any upcoming transfers and then the respective SnapMirror operation is initiated. If the SnapMirror operation initiated a transfer then it will also use the new throttle. If "throttle" needs to be applied for a specific transfer use SnapMirror Transfer REST API.

For SnapMirror active sync relationships, when "consistency\_group\_volumes" is specified during PATCH, first the existing FlexVol volume relationship is deleted and released and then the volumes are added to the SnapMirror active sync relationship.

## Examples

### Related ONTAP commands

- `snapmirror modify`
- `snapmirror initialize`
- `snapmirror resync`
- `snapmirror break`
- `snapmirror quiesce`

- `snapmirror resume`

## Important notes

- The property "transfer\_schedule" if set on a SnapMirror relationship overrides the "transfer\_schedule" set on the policy being used with the SnapMirror relationship.
- The property "throttle" if set on a SnapMirror relationship overrides the "throttle" set on the policy being used with the SnapMirror relationship.
- The properties "transfer\_schedule" and "throttle" are not supported when "failback" is set to "true".
- The properties "transfer\_schedule" and "throttle" are not supported when "failover" is set to "true".
- The properties "transfer\_schedule" and "throttle" are not supported when "force\_failover" is set to "true".
- The properties "transfer\_schedule" and "throttle" are not supported when the direction of the relationship is being reversed.
- To remove a transfer\_schedule on a SnapMirror relationship set the "transfer\_schedule" to null (no-quotes) during SnapMirror relationship PATCH.
- The property "identity\_preservation" value can be changed from a higher "identity\_preservation" threshold value to a lower "identity\_preservation" threshold value but not vice-versa. For example, the threshold value of the "identity\_preservation" property can be changed from "full" to "exclude\_network\_config", but cannot be increased from "exclude\_network\_and\_protocol\_config" to "exclude\_network\_config" to "full". The threshold value of the "identity\_preservation" cannot be changed to "exclude\_network\_and\_protocol\_config" for IDP SVMMDR.
- The property "backoff\_level" is only applicable for FlexVol SnapMirror relationships.

## Examples

The following examples show how to perform the SnapMirror "resync", "initialize", "resume", "quiesce", and "break" operations. In addition, a relationship can be failed over to the destination endpoint and start serving data from the destination endpoint. A failed over relationship can be failed back to the source endpoint and serve data from the source endpoint. Also a relationship can be reversed by making the source endpoint as the new destination endpoint and the destination endpoint as the new source endpoint.

To update an associated SnapMirror policy.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"policy": { "name" : "MirrorAndVaultDiscardNetwork"}}'
```

To perform SnapMirror "resync" for an asynchronous SnapMirror relationship.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"state":"snapmirrored"}'
```

To perform SnapMirror "initialize" for an asynchronous SnapMirror relationship.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"state":"snapmirrored"}'
```

To perform SnapMirror "resume" for an asynchronous SnapMirror relationship.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"state":"snapmirrored"}'
```

To perform SnapMirror "quiesce" for an asynchronous SnapMirror relationship.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff" '{"state":"paused"}'
```

To perform SnapMirror "break" for an asynchronous SnapMirror relationship. This operation does a failover to the destination endpoint. After a the failover, data can then be served from the destination endpoint.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff" '{"state":"broken_off"}'
```

To forcefully failover to the destination endpoint and start serving data from the destination endpoint.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/?force=true" '{"state":"broken_off"}'
```

To fallback to the source endpoint and start serving data from the source endpoint for an asynchronous relationship.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/?fallback=true" '{"state":"snapmirrored"}'
```

To fallback to the source endpoint and start serving data from the source endpoint for a synchronous relationship.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/?fallback=true" '{"state":"in_sync"}'
```

To reverse the direction of an asynchronous relationship, that is, make the source endpoint as the new destination endpoint and make the destination endpoint as the new source endpoint.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"source": {"path": "dst_svm:dst_vol"}, "destination": {"path": "src_svm:src_vol"}, "state": "snapmirrored"}'
```

To reverse the direction of a synchronous relationship, that is, make the source endpoint as the new destination endpoint and make the destination endpoint as the new source endpoint.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"source": {"path": "dst_svm:dst_vol"}, "destination": {"path": "src_svm:src_vol"}, "state": "in_sync"}'
```

Updating SnapMirror transfer\_schedule and throttle for an asynchronous SnapMirror relationship. Transfer\_schedule can be specified as UUID or name or both.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"transfer_schedule":{"uuid":"817500fa-092d-44c5-9c10-7b54f7b2f20a", "name":"5min"}, "throttle":100}'
```

Removing the SnapMirror transfer\_schedule for an asynchronous SnapMirror relationship.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"transfer_schedule":{"uuid":null, "name":null}}'
```

Removing the SnapMirror throttle for an asynchronous SnapMirror relationship.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"throttle":0}'
```

To perform SnapMirror "resync" and update the SnapMirror transfer\_schedule for an asynchronous SnapMirror relationship. First the transfer\_schedule is modified and then the resync transfer is initiated.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"state":"snapmirrored", "transfer_schedule":{"uuid":"817500fa-092d-44c5-9c10-7b54f7b2f20a", "name":"5min"}}'
```

To perform SnapMirror "initialize" and update the SnapMirror throttle for an asynchronous SnapMirror relationship. First the throttle is modified and then the initialize transfer is initiated. The initialize transfer will use this new throttle.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"state":"snapmirrored", "throttle":100}'
```

To perform SnapMirror "resync" and update the SnapMirror throttle for an asynchronous SnapMirror relationship. First the throttle is modified and then the resync transfer is initiated. The resync transfer will use this new throttle.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"state":"snapmirrored", "throttle":100}'
```

To perform a SnapMirror active sync or Asynchronous Consistency Group expansion.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{ "source" :  
{"consistency_group_volumes":[{"name":"vol"}]}, "destination" :  
{"consistency_group_volumes":[{"name":"voldp"}]} }'
```

Updating SnapMirror backoff\_level for an asynchronous SnapMirror relationship.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"backoff_level": "none}"'
```

To perform SnapMirror "initialize" and update the SnapMirror backoff\_level for an asynchronous SnapMirror relationship. First the backoff\_level is modified and then the initialize transfer is initiated. The initialize transfer will use this new backoff\_level.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"state":"snapmirrored", "backoff_level": "medium}"'
```

To perform SnapMirror "resync" and update the SnapMirror backoff\_level for an asynchronous SnapMirror relationship. First the backoff\_level is modified and then the resync transfer is initiated. The resync transfer will use this new backoff\_level.

```
PATCH "/api/snapmirror/relationships/98bb2608-fc60-11e8-aa13-005056a707ff/" '{"state":"snapmirrored", "backoff_level": "medium}"'
```

## Learn more

- [DOC /snapmirror/relationships](#)

## Parameters

Name	Type	In	Required	Description
uuid	string	path	True	SnapMirror relationship UUID
failover	boolean	query	False	<p>If this parameter is set, validation and failover will occur to the SVM-DR SnapMirror relationship destination endpoint. Any other fields specified with this parameter will be ignored. This parameter is supported only for SVM-DR SnapMirror relationships.</p> <ul style="list-style-type: none"><li>• Introduced in: 9.7</li><li>• Default value:</li></ul>
force-failover	boolean	query	False	<p>If this parameter is set, failover will occur to the SVM-DR SnapMirror relationship destination endpoint, overriding the validation errors. Any other fields specified with this parameter will be ignored. This parameter is supported only for SVM-DR SnapMirror relationships.</p> <ul style="list-style-type: none"><li>• Introduced in: 9.7</li><li>• Default value:</li></ul>



Name	Type	In	Required	Description
force	boolean	query	False	<p>If this parameter is set while specifying the state as "broken_off", indicates a forced failover overriding the validation errors.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.8</li> <li>• Default value:</li> </ul>
failback	boolean	query	False	<p>If this parameter is set while specifying the state as "snapmirrored", indicates recovery of the failed over SnapMirror relationship by preserving the data written on the destination endpoint when the SnapMirror relationship was in failed over state. This flag is only applicable to SVM-DR SnapMirror relationships.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.8</li> <li>• Default value:</li> </ul>

Name	Type	In	Required	Description
return_timeout	integer	query	False	<p>The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.</p> <ul style="list-style-type: none"> <li>• Default value: 0</li> <li>• Max value: 120</li> <li>• Min value: 0</li> </ul>
validate_only	boolean	query	False	<p>Validate the operation and its parameters, without actually performing the operation.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.7</li> </ul>

## Request Body

Name	Type	Description
backoff_level	string	Specifies the SnapMirror backoff level due to Client Ops for FlexVol SnapMirror relationships.

Name	Type	Description
destination	<a href="#">snapmirror_endpoint</a>	<p>Endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to create the destination SVM endpoint or to establish an SVM DR relationship must have the property "cluster" populated with the remote cluster details. A POST request to create the destination FlexVol volume, FlexGroup volume, Consistency Group, ONTAP S3 bucket and NON-ONTAP object-store endpoints can optionally specify the "cluster" property when the source SVM and the destination SVM are peered. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.6</li> </ul>
exported_snapshot	string	Snapshot exported to clients on destination.
group_type	string	Specifies the group type of the top level SnapMirror relationship. The volume relationships are shown as <i>none</i> , the SVM DR relationships are shown as <i>svm_dr</i> , the Consistency Group relationships are shown as <i>consistency_group</i> , and the FlexGroup volume relationships are shown as <i>flexgroup</i> .
healthy	boolean	Is the relationship healthy?

Name	Type	Description
identity_preservation	string	Specifies which configuration of the source SVM is replicated to the destination SVM. This property is applicable only for SVM data protection with "async" policy type. This "identity_preservation" overrides the "identity_preservation" set on the SnapMirror relationship's policy.
io_serving_copy	string	Specifies the sites serving I/O for the SnapMirror active sync relationship.
lag_time	string	Time since the exported snapshot was created.
last_transfer_network_compression_ratio	string	Specifies the compression ratio achieved for the data sent over the wire with network compression enabled for the last successful transfer.
last_transfer_type	string	Specifies the operation type of the last transfer that occurred on the relationship. The <i>initialize</i> transfer occurs when the relationship state changes from uninitialized to snapmirrored or in_sync. The <i>update</i> transfer occurs when the snapshots are transferred from the source endpoint to the destination endpoint as part of scheduled or manual update. The <i>resync</i> transfer occurs when the relationship state changes from broken_off to snapmirrored or in_sync. The <i>restore</i> transfer occurs when the snapshot is restored from a destination endpoint to another endpoint.
master_bias_activated_site	string	Specifies the Master Bias Activated Site for the SnapMirror active sync relationship.
policy	<a href="#">policy</a>	Basic policy information of the relationship.

Name	Type	Description
preferred_site	string	Specifies the Primary Site of the SnapMirror active sync relationship.
preserve	boolean	Set to true on resync to preserve snapshots on the destination that are newer than the latest common snapshot. This property is applicable only for relationships with FlexVol volume or FlexGroup volume endpoints and when the PATCH state is being changed to "snapmirrored".
quick_resync	boolean	Set to true to reduce resync time by not preserving storage efficiency. This property is applicable only for relationships with FlexVol volume endpoints and SVMDR relationships when the PATCH state is being changed to "snapmirrored".
recover_after_break	boolean	Set to true to recover from a failed SnapMirror break operation on a FlexGroup volume relationship. This restores all destination FlexGroup constituent volumes to the latest snapshot, and any writes to the read-write constituents are lost. This property is applicable only for SnapMirror relationships with FlexGroup volume endpoints and when the PATCH state is being changed to "broken_off".
restore_to_snapshot	string	Specifies the snapshot to restore to on the destination during the break operation. This property is applicable only for SnapMirror relationships with FlexVol volume endpoints and when the PATCH state is being changed to "broken_off".

Name	Type	Description
source	<a href="#">snapmirror_source_endpoint</a>	Source endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
state	string	<p>State of the relationship. To initialize the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". To break the relationship, PATCH the state to "broken_off" for relationships with a policy of type "async" or "sync". SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" cannot be "broken_off". To resync the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" can be in "broken_off" state due to a failed attempt of SnapMirror failover. To pause the relationship, suspending further transfers, PATCH the state to "paused" for relationships with a policy of type "async" or "sync". SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" cannot be "paused". To resume transfers for a paused relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". The entries "in_sync", "out_of_sync", "synchronizing", "expanding", and "shrinking" are only applicable to relationships with a policy of type "sync". A PATCH call on the state change only triggers the transition to the specified state. You must poll on the "state", "healthy" and "unhealthy_reason" properties using a GET request to determine if the transition is successful. To automatically initialize the relationship when specifying "create_destination" property, set the state to "snapmirrored" for relationships with a policy of type</p>

Name	Type	Description
throttle	integer	Throttle, in KBs per second. This "throttle" overrides the "throttle" set on the SnapMirror relationship's policy. If neither of these are set, defaults to 0, which is interpreted as unlimited.
total_transfer_bytes	integer	Cumulative bytes transferred for the relationship.
total_transfer_duration	string	Indicates the cumulative duration of all transfers since the last aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship.
transfer	<a href="#">transfer</a>	Basic information on the current transfer or the last transfer if there is no active transfer at the time of the request.
transfer_schedule	<a href="#">transfer_schedule</a>	Schedule used to update asynchronous relationships. This "transfer_schedule" overrides the "transfer_schedule" set on the SnapMirror relationship's policy. To remove the "transfer_schedule", set its value to null (no-quotes). Only cron schedules are supported for SnapMirror.
unhealthy_reason	array[ <a href="#">snapmirror_error</a> ]	Reason the relationship is not healthy. It is a concatenation of up to four levels of error messages.
uuid	string	Unique identifier of the SnapMirror relationship.



## Example request

```
{
  "backoff_level": "medium",
  "destination": {
    "cluster": {
      "name": "cluster1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "luns": {
      "name": "/vol/volume1/lun1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "path": "svm1:volume1",
    "svm": {
      "name": "svm1",
      "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
    }
  },
  "exported_snapshot": "string",
  "group_type": "consistency_group",
  "identity_preservation": "string",
  "io_serving_copy": "C1_sti85-vsim-ucs209a_cluster, C1_sti85-vsim-ucs209c_cluster",
  "lag_time": "PT8H35M42S",
  "last_transfer_network_compression_ratio": 61,
  "last_transfer_type": "initialize",
  "master_bias_activated_site": "C1_sti85-vsim-ucs209a_cluster",
  "policy": {
    "name": "Asynchronous",
    "type": "string",
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  },
  "preferred_site": "C1_sti85-vsim-ucs209a_cluster",
  "restore": true,
  "restore_to_snapshot": "string",
  "source": {
    "cluster": {
      "name": "cluster1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "luns": {
      "name": "/vol/volume1/lun1",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "path": "svm1:volume1",
```

```

    "svm": {
      "name": "svm1",
      "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
    },
    "state": "snapmirrored",
    "throttle": 0,
    "total_transfer_bytes": 1098210312,
    "total_transfer_duration": "PT3M21S",
    "transfer": {
      "end_time": "2020-12-02 18:36:19 -0800",
      "last_updated_time": "2023-09-14 15:39:19 -0700",
      "state": "string",
      "total_duration": "PT28M41S",
      "type": "initialize",
      "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
    },
    "transfer_schedule": {
      "name": "weekly",
      "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
    },
    "unhealthy_reason": [
      {
        "arguments": [],
        "code": "6621444",
        "message": "Failed to complete update operation on one or more
item relationships."
      },
      {
        "arguments": [],
        "code": "6621445",
        "message": "Group Update failed"
      }
    ],
    "uuid": "4ea7a442-86d1-11e0-ae1c-123478563412"
  }

```

## Response

Status: 200, Ok

Name	Type	Description
job	<a href="#">job_link</a>	

## Example response

```
{
  "job": {
    "uuid": "string"
  }
}
```

## Response

Status: 202, Accepted

## Error

Status: Default

### ONTAP Error Response codes

Error code	Description
13303825	Could not retrieve information for the SnapMirror policy type
13303817	Unknown value for the SnapMirror state
13303829	Invalid state
13303830	Transient state
13303831	Invalid state for async SnapMirror relationship
13303834	Given input valid only for FlexGroup SnapMirror relationship
13303835	Given flag is valid only when PATCH state is broken_off
13303836	Given flag is valid only when PATCH state is snapmirrored or in_sync
13303818	Invalid state transition requested
13303828	Given state change is not possible for SVM SnapMirror relationship
13303833	Requested state change is not possible
13303832	SnapMirror relationship is already initialized
13303824	Quiescing the SnapMirror relationship has failed
13303826	Required environment variables are not set

Error code	Description
13303827	Internal Error
13303823	Quiesce operation timed out
13303821	Invalid SnapMirror policy name/UUID
13303819	Could not retrieve SnapMirror policy information
13303851	Cannot modify attributes of SnapMirror restore relationship
13303816	Could not retrieve the state or status values for the relationship. Wait a few minutes and try the operation again
13303837	Given flags are valid only if SnapMirror state change is requested
6619546	Destination must be a dp volume
13303808	Transition to broken_off state failed
13303809	Transition to paused state failed
13303810	Transition to snapmirrored state failed
13303811	Transition from paused state failed
13303820	SnapMirror policy, transfer_schedule, and throttle, if specified were successfully updated, state transition failed
13303856	SVM is not configured with any data protocol
13303857	SVM is not configured with any network interface
13303858	Internal error. Failed to check LIF and protocols details for SVM
13303859	Internal error. SVM Failover operation failed. SVM operational state is unavailable.
13303865	Modifying the specified SnapMirror policy is not supported.
13303866	Cannot use the specified policy to modify the policy of the relationship.
13303867	Modifying the policy of an async-mirror or a vault relationship is not supported.
13303884	LIF and protocols details are configured incorrectly for SVM.
13303996	The source and destination clusters both have a policy with the same name, but they have different properties.
13304062	Cannot reverse the direction of a SnapMirror DP relationship when the source cluster version is earlier than the destination cluster version.

Error code	Description
13304070	Remote peer cluster requires the dp_rest_support capability to support reversing the direction of a DP relationship.
13304071	Failed to access capabilities on remote cluster.
13304080	Specified uuid and name do not match.
13304081	Modifying a property during the operation is not supported.
13304082	The specified properties are mutually exclusive.
13304083	The specified property is not supported because all nodes in the cluster are not capable of supporting the property.
13304086	Reversing the direction of a SnapMirror relationship associated with a policy containing the property create_snapshot_on_source set to false is not supported.
6619715	Modification of relationship is in progress. Retry the command after a few minutes.
6619699	Schedule not found.
13304108	Schedule not found in the Administrative SVM or the SVM for the relationship.
13304111	The SnapMirror active sync relationship consistency groups are nested. Expanding an SnapMirror active sync relationship with a pre-existing DP volume is only supported for flat consistency groups.
6621458	The destination Consistency Group is the source of a SnapMirror Synchronous (SM-S) relationship. Sources of SM-S relationships cannot be the destination of any other SnapMirror relationship.
6621782	A property of the policy is not valid for relationships between these endpoints.

| 13304120 | Values specified for the source.path and destination.path properties do not match the relationship's source.path or destination.path properties. | 13304003 | IntraCluster flip operation is not supported. | | 6621125 | The policy is not valid for relationships with FlexGroup volume endpoints. Only policies without snapshot creation schedules are supported for these relationships. | | 13304093 | The property specified is not supported for the specified relationships. | | 6622077 | The expand operation has failed on the SnapMirror active sync relationship with specified destination path. |

6619720	Relationship information has been updated and is being propagated. Wait a few minutes and try the operation again.
13304156	Either use state as broken-off or failover as true for SVM-DR fail-over operation.

13304159	Could not retrieve the state or status values for the relationship. Wait a few minutes and try the operation again. If the error persists, address the error condition shown.
----------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## Definitions

## See Definitions

href

Name	Type	Description
href	string	

\_links

error\_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

SnapMirror Consistency Group failover error message.

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message

status

Name	Type	Description
code	string	Status code
message	string	SnapMirror Consistency Group failover status.

snapmirror\_consistency\_group\_failover

SnapMirror Consistency Group failover information. The SnapMirror Consistency Group failover can be a planned or an unplanned operation. Only active SnapMirror Consistency Group failover operation progress can be monitored using this object. In case of an error during the failover operation, the property "consistency\_group\_failover.error" holds the reason for the error. ONTAP automatically retries any failed SnapMirror Consistency Group failover operation.

Name	Type	Description
state	string	SnapMirror Consistency Group failover state.
status	<a href="#">status</a>	
type	string	SnapMirror Consistency Group failover type.

bucket\_retention

Specifies the retention-mode and default retention period configured on the destination bucket.

Name	Type	Description
default_period	string	Specifies the default retention period that is applied to objects while committing them to the WORM state without an associated retention period. The retention period can be in years, or days. The retention period value represents a duration and must be specified in the ISO-8601 duration format. A period specified for years and days is represented in the ISO-8601 format as "P<num>Y" and "P<num>D" respectively, for example "P10Y" represents a duration of 10 years. The period string must contain only a single time element that is, either years, or days. A duration which combines different periods is not supported, for example "P1Y10D" is not supported.</num></num>

storage\_service



Name	Type	Description
enabled	boolean	<p>This property indicates whether to create the destination endpoint using storage service. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.7</li> <li>• x-nullable: true</li> </ul>
enforce_performance	boolean	<p>Optional property to enforce storage service performance on the destination endpoint. This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>• Default value: 1</li> <li>• Introduced in: 9.7</li> <li>• x-nullable: true</li> </ul>

Name	Type	Description
name	string	<p>Optional property to specify the storage service name for the destination endpoint. This property is considered when the property "create_destination.storage_service.enabled" is set to "true". When the property "create_destination.storage_service.enabled" is set to "true" and the "create_destination.storage_service.name" for the endpoint is not specified, then ONTAP selects the highest storage service available on the cluster to provision the destination endpoint. This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>enum: ["extreme", "performance", "value"]</li> <li>Introduced in: 9.6</li> <li>x-nullable: true</li> </ul>

tiering



Name	Type	Description
policy	string	<p>Optional property to specify the destination endpoint's tiering policy when "create_destination.tiering.supported" is set to "true". This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. This property determines whether the user data blocks of the destination endpoint in a FabricPool will be tiered to the cloud store when they become cold. FabricPool combines flash (performance tier) with a cloud store into a single aggregate. Temperature of the destination endpoint volume blocks increases if they are accessed frequently and decreases when they are not. all – This policy allows tiering of both destination endpoint snapshots and the user transferred data blocks to the cloud store as soon as possible by ignoring the temperature on the volume blocks. This tiering policy is not applicable for Consistency Group destination endpoints or for synchronous relationships. auto – This policy allows tiering of both destination endpoint snapshots and the active file system user data to the cloud store none – Destination endpoint volume blocks will not be tiered to the cloud store. snapshot_only – This policy allows tiering of only the destination endpoint volume snapshots not associated with the active file system. The default tiering policy is "snapshot_only" for a FlexVol volume and "none" for a FlexGroup volume. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>enum: ["all", "auto", "none", "snapshot_only"]</li> <li>Introduced in: 9.6</li> </ul> <p>x-nullable: true</p>

Name	Type	Description
supported	boolean	<p>Optional property to enable provisioning of the destination endpoint volumes on FabricPool aggregates. This property is applicable to FlexVol volume, FlexGroup volume, and Consistency Group endpoints. Only FabricPool aggregates are used if this property is set to "true" and only non FabricPool aggregates are used if this property is set to "false". Tiering support for a FlexGroup volume can be changed by moving all of the constituents to the required aggregates. Note that in order to tier data, not only do the destination endpoint volumes need to support tiering by using FabricPools, the "create_destination.tiering.policy" must not be "none". A destination endpoint that uses FabricPools but has a tiering "policy" of "none" supports tiering but will not tier any data. This property is supported for Unified ONTAP destination endpoints only.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.6</li> <li>• x-nullable: true</li> </ul>

#### snapmirror\_destination\_creation

Use this object to provision the destination endpoint when establishing a SnapMirror relationship for a FlexVol volume, FlexGroup volume, SVM, Consistency Group or ONTAP S3 Bucket. Given a source endpoint, the destination endpoint is provisioned in the SVM specified in the "destination.path" property. While protecting an SVM, the SVM destination endpoint can only be provisioned on the local cluster. To provision the SVM destination endpoint use the optional "source.cluster.name" property to specify the remote cluster name or use the optional "source.cluster.uuid" property to specify the remote cluster UUID. When "create\_destination.enabled" option is specified while making a POST for a SnapMirror relationship, the relationship can be automatically initialized by setting the "state" either to "snapmirrored" when the policy is of type "async" or to "in\_sync" when the policy is of type "sync". The "destination.path" property must specify the destination endpoint path. For example, for FlexVol volume and FlexGroup volume, the "destination.path" can be specified as <destination-svm-name:dp-volume-name>, for SVM data protection, the "destination.path" must be specified as &lt;destination-svm-name:&gt;, and for Consistency Group, the "destination.path" must be specified as &lt;destination-svm-name:&gt; along with the "destination.consistency\_group\_volumes" or "destination.luns" property to indicate the list of destination volumes or LUNs of type "DP" in the Consistency Group. For a FlexVol volume, a FlexGroup volume, Consistency Group or a Bucket destination endpoint, the properties in this object can be specified either from the source or the destination cluster. For an SVM destination endpoint, the

properties in this object can be specified from the destination cluster only. This object is not supported for non ONTAP endpoints. While protecting a S3 Bucket, the optional "size" property can be used to create ONTAP S3 Bucket destination endpoint of the specified size.</destination-svm-name:></destination-svm-name:></destination-svm-name:dp-volume-name>

Name	Type	Description
bucket_retention	<a href="#">bucket_retention</a>	Specifies the retention-mode and default retention period configured on the destination bucket.
enabled	boolean	Optional property to create the destination endpoint when establishing a SnapMirror relationship. It is assumed to be "false" if no other property is set and assumed to be "true" if any other property is set.
size	integer	Optional property to specify the size of destination endpoint in bytes. This property is applicable only to ONTAP S3 Bucket endpoints. The minimum size for S3 bucket is 80MB and maximum size is 64TB. If not specified, system will create destination with default size of 800GB.  <ul style="list-style-type: none"> <li>• Introduced in: 9.10</li> <li>• x-nullable: true</li> </ul>
snapshot_locking_enabled	boolean	Optional property to create the destination endpoint with snapshot locking enabled when establishing a SnapMirror relationship. This property is applicable to FlexVol volumes and FlexGroup volumes.
storage_service	<a href="#">storage_service</a>	
tiering	<a href="#">tiering</a>	

cluster

Name	Type	Description
name	string	
uuid	string	

## consistency\_group\_volumes

Name	Type	Description
name	string	The name of the volume.

## luns

Optional property for a SnapMirror endpoint. Specifies the list of source LUNs and optionally list of destination LUNs during SnapMirror Consistency Group LUN restore operation.

Name	Type	Description
name	string	<p>The name of a LUN. A LUN is located within a volume. Optionally, it can be located within a qtree in a volume.</p> <p>LUN names are paths of the form "/vol/&lt;volume&gt;[/&lt;qtree&gt;]/&lt;namespace&gt;" where the qtree name is optional.</p>
uuid	string	The unique identifier of the LUN.

## 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.

## snapmirror\_endpoint

Endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to create the destination SVM endpoint or to establish an SVM DR relationship must have the property "cluster" populated with the remote cluster details. A POST request to create the destination FlexVol volume, FlexGroup volume, Consistency Group, ONTAP S3 bucket and NON-ONTAP object-store endpoints can optionally specify the "cluster" property when the source SVM and the destination SVM are peered. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
cluster	<a href="#">cluster</a>	
luns	<a href="#">luns</a>	Optional property for a SnapMirror endpoint. Specifies the list of source LUNs and optionally list of destination LUNs during SnapMirror Consistency Group LUN restore operation.
path	string	ONTAP FlexVol/FlexGroup - svm1:volume1 ONTAP SVM - svm1: ONTAP Consistency Group - svm1:/cg/cg_name ONTAP S3 - svm1:/bucket/bucket1 NON-ONTAP - objstore1:/objstore <ul style="list-style-type: none"> <li>• example: svm1:volume1</li> <li>• Introduced in: 9.6</li> <li>• x-nullable: true</li> </ul>
svm	<a href="#">svm</a>	SVM, applies only to SVM-scoped objects.

## policy

Basic policy information of the relationship.

Name	Type	Description
name	string	Name of the SnapMirror policy.
type	string	
uuid	string	Unique identifier of the SnapMirror policy.

## snapmirror\_source\_endpoint

Source endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
cluster	<a href="#">cluster</a>	



Name	Type	Description
luns	<a href="#">luns</a>	Optional property for a SnapMirror endpoint. Specifies the list of source LUNs and optionally list of destination LUNs during SnapMirror Consistency Group LUN restore operation.
path	string	ONTAP FlexVol/FlexGroup - svm1:volume1 ONTAP SVM - svm1: ONTAP Consistency Group - svm1:/cg/cg_name ONTAP S3 - svm1:/bucket/bucket1 NON-ONTAP - objstore1:/objstore <ul style="list-style-type: none"> <li>• example: svm1:volume1</li> <li>• Introduced in: 9.14</li> <li>• x-nullable: true</li> </ul>
svm	<a href="#">svm</a>	SVM, applies only to SVM-scoped objects.

#### svmdr\_volumes

Name	Type	Description
name	string	The name of the volume.

#### transfer

Basic information on the current transfer or the last transfer if there is no active transfer at the time of the request.

Name	Type	Description
bytes_transferred	integer	Total bytes transferred in the current or last successful transfer.
end_time	string	End time of the last transfer.
last_updated_time	string	Last updated time of the bytes transferred in the current transfer.
state	string	

Name	Type	Description
total_duration	string	Elapsed time to transfer all snapshots for the last successful transfer.
type	string	Specifies the operation type of the current transfer on the relationship. The <i>initialize</i> transfer occurs when the relationship state changes from "uninitialized" to "snapmirrored" or "in_sync". The <i>update</i> transfer occurs when snapshots are being transferred from the source endpoint to the destination endpoint as part of a scheduled or manual update. The <i>resync</i> transfer occurs when the relationship state changes from "broken_off" to "snapmirrored" or "in_sync". The <i>restore</i> transfer occurs when a snapshot is being restored from a destination endpoint to another endpoint.
uuid	string	Transfer UUID. This property is applicable only for active transfers.

#### transfer\_schedule

Schedule used to update asynchronous relationships. This "transfer\_schedule" overrides the "transfer\_schedule" set on the SnapMirror relationship's policy. To remove the "transfer\_schedule", set its value to null (no-quotes). Only cron schedules are supported for SnapMirror.

Name	Type	Description
name	string	Job schedule name
uuid	string	Job schedule UUID

#### snapmirror\_error

SnapMirror error

Name	Type	Description
arguments	array[string]	Arguments present in the error message encountered.

Name	Type	Description
code	string	Error code
message	string	Error message

#### snapmirror\_relationship

SnapMirror relationship information. The SnapMirror relationship can be either "async" or "sync" based on the type of SnapMirror policy associated with the relationship. The source and destination endpoints of a SnapMirror relationship must be of the same type, for example, if the source endpoint is a FlexVol volume then the destination endpoint must be a FlexVol volume. The SnapMirror policy type "async" can be used when the SnapMirror relationship has FlexVol volume or FlexGroup volume or SVM as the endpoint. The SnapMirror policy type "sync" can be used when the SnapMirror relationship has FlexVol volume as the endpoint. The SnapMirror policy type "sync" with "sync\_type" as "automated\_failover" can be used when the SnapMirror relationship has Consistency Group as the endpoint.

Name	Type	Description
backoff_level	string	Specifies the SnapMirror backoff level due to Client Ops for FlexVol SnapMirror relationships.
destination	<a href="#">snapmirror_endpoint</a>	<p>Endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to create the destination SVM endpoint or to establish an SVM DR relationship must have the property "cluster" populated with the remote cluster details. A POST request to create the destination FlexVol volume, FlexGroup volume, Consistency Group, ONTAP S3 bucket and NON-ONTAP object-store endpoints can optionally specify the "cluster" property when the source SVM and the destination SVM are peered. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.</p> <ul style="list-style-type: none"> <li>• Introduced in: 9.6</li> </ul>

Name	Type	Description
exported_snapshot	string	Snapshot exported to clients on destination.
group_type	string	Specifies the group type of the top level SnapMirror relationship. The volume relationships are shown as <i>none</i> , the SVMDR relationships are shown as <i>svm_dr</i> , the Consistency Group relationships are shown as <i>consistency_group</i> , and the FlexGroup volume relationships are shown as <i>flexgroup</i> .
healthy	boolean	Is the relationship healthy?
identity_preservation	string	Specifies which configuration of the source SVM is replicated to the destination SVM. This property is applicable only for SVM data protection with "async" policy type. This "identity_preservation" overrides the "identity_preservation" set on the SnapMirror relationship's policy.
io_serving_copy	string	Specifies the sites serving I/O for the SnapMirror active sync relationship.
lag_time	string	Time since the exported snapshot was created.
last_transfer_network_compression_ratio	string	Specifies the compression ratio achieved for the data sent over the wire with network compression enabled for the last successful transfer.

Name	Type	Description
last_transfer_type	string	Specifies the operation type of the last transfer that occurred on the relationship. The <i>initialize</i> transfer occurs when the relationship state changes from uninitialized to snapmirrored or in_sync. The <i>update</i> transfer occurs when the snapshots are transferred from the source endpoint to the destination endpoint as part of scheduled or manual update. The <i>resync</i> transfer occurs when the relationship state changes from broken_off to snapmirrored or in_sync. The <i>restore</i> transfer occurs when the snapshot is restored from a destination endpoint to another endpoint.
master_bias_activated_site	string	Specifies the Master Bias Activated Site for the SnapMirror active sync relationship.
policy	<a href="#">policy</a>	Basic policy information of the relationship.
preferred_site	string	Specifies the Primary Site of the SnapMirror active sync relationship.
preserve	boolean	Set to true on resync to preserve snapshots on the destination that are newer than the latest common snapshot. This property is applicable only for relationships with FlexVol volume or FlexGroup volume endpoints and when the PATCH state is being changed to "snapmirrored".
quick_resync	boolean	Set to true to reduce resync time by not preserving storage efficiency. This property is applicable only for relationships with FlexVol volume endpoints and SVMDR relationships when the PATCH state is being changed to "snapmirrored".

Name	Type	Description
recover_after_break	boolean	Set to true to recover from a failed SnapMirror break operation on a FlexGroup volume relationship. This restores all destination FlexGroup constituent volumes to the latest snapshot, and any writes to the read-write constituents are lost. This property is applicable only for SnapMirror relationships with FlexGroup volume endpoints and when the PATCH state is being changed to "broken_off".
restore_to_snapshot	string	Specifies the snapshot to restore to on the destination during the break operation. This property is applicable only for SnapMirror relationships with FlexVol volume endpoints and when the PATCH state is being changed to "broken_off".
source	<a href="#">snapmirror_source_endpoint</a>	Source endpoint of a SnapMirror relationship. For a GET request, the property "cluster" is populated when the endpoint is on a remote cluster. A POST request to establish a SnapMirror relationship between the source endpoint and destination endpoint and when the source SVM and the destination SVM are not peered, must specify the "cluster" property for the remote endpoint.

Name	Type	Description
state	string	<p>State of the relationship. To initialize the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". To break the relationship, PATCH the state to "broken_off" for relationships with a policy of type "async" or "sync".</p> <p>SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" cannot be "broken_off". To resync the relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync".</p> <p>SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" can be in "broken_off" state due to a failed attempt of SnapMirror failover. To pause the relationship, suspending further transfers, PATCH the state to "paused" for relationships with a policy of type "async" or "sync". SnapMirror relationships with the policy type as "sync" and "sync_type" as "automated_failover" cannot be "paused". To resume transfers for a paused relationship, PATCH the state to "snapmirrored" for relationships with a policy of type "async" or to state "in_sync" for relationships with a policy of type "sync". The entries "in_sync", "out_of_sync", "synchronizing", "expanding", and "shrinking" are only applicable to relationships with a policy of type "sync". A PATCH call on the state change only triggers the transition to the specified state. You must poll on the "state", "healthy" and "unhealthy_reason" properties using a GET request to determine if the transition is successful. To automatically initialize the relationship when specifying "create_destination" property, set the state to "snapmirrored" for</p>

Name	Type	Description
throttle	integer	Throttle, in KBs per second. This "throttle" overrides the "throttle" set on the SnapMirror relationship's policy. If neither of these are set, defaults to 0, which is interpreted as unlimited.
total_transfer_bytes	integer	Cumulative bytes transferred for the relationship.
total_transfer_duration	string	Indicates the cumulative duration of all transfers since the last aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship.
transfer	<a href="#">transfer</a>	Basic information on the current transfer or the last transfer if there is no active transfer at the time of the request.
transfer_schedule	<a href="#">transfer_schedule</a>	Schedule used to update asynchronous relationships. This "transfer_schedule" overrides the "transfer_schedule" set on the SnapMirror relationship's policy. To remove the "transfer_schedule", set its value to null (no-quotes). Only cron schedules are supported for SnapMirror.
unhealthy_reason	array[ <a href="#">snapmirror_error</a> ]	Reason the relationship is not healthy. It is a concatenation of up to four levels of error messages.
uuid	string	Unique identifier of the SnapMirror relationship.

job\_link



Name	Type	Description
uuid	string	The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation.

returned\_error

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	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.