



# **snapmirror commands**

## **ONTAP 9.15.1 commands**

NetApp  
December 18, 2024

# Table of Contents

- snapmirror commands . . . . . 1
  - snapmirror abort . . . . . 1
  - snapmirror break . . . . . 4
  - snapmirror create . . . . . 7
  - snapmirror delete . . . . . 15
  - snapmirror flexgroup-epuuid-prefix-prepare-to-downgrade . . . . . 19
  - snapmirror initialize-ls-set . . . . . 19
  - snapmirror initialize . . . . . 21
  - snapmirror list-destinations . . . . . 27
  - snapmirror modify . . . . . 33
  - snapmirror promote . . . . . 37
  - snapmirror protect . . . . . 39
  - snapmirror quiesce . . . . . 42
  - snapmirror release . . . . . 45
  - snapmirror restore . . . . . 47
  - snapmirror resume . . . . . 56
  - snapmirror resync . . . . . 59
  - snapmirror set-options . . . . . 67
  - snapmirror set-preferred-cluster . . . . . 68
  - snapmirror show-history . . . . . 68
  - snapmirror show . . . . . 75
  - snapmirror update-ls-set . . . . . 100
  - snapmirror update . . . . . 102
  - snapmirror config-replication commands . . . . . 107
  - snapmirror failover commands . . . . . 113
  - snapmirror mediator commands . . . . . 115
  - snapmirror object-store commands . . . . . 122
  - snapmirror policy commands . . . . . 133
  - snapmirror snapshot-owner commands . . . . . 149

# snapmirror commands

## snapmirror abort

Abort an active transfer

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### Description

The `snapmirror abort` command stops SnapMirror transfers that might have started and not completed. A SnapMirror transfer is an operation on a given SnapMirror relationship and the relationship is identified by its destination endpoint, which can be a volume, a Vserver, an Application Consistency Group or a non-Data ONTAP endpoint. You identify the SnapMirror relationship with this command and the command aborts the transfer for the relationship. For load-sharing mirrors, the command also aborts transfers for other relationships that are part of the same load-sharing set. For SolidFire destination endpoints, the `snapmirror abort` command is only supported if the endpoint is in a SnapMirror relationship.

Load-sharing mirrors are either up to date and serving data to clients, or they are lagging and not serving data to clients. If the `snapmirror abort` command identifies an up-to-date load-sharing mirror, then SnapMirror transfers to the up-to-date load-sharing mirror and associated up-to-date load-sharing mirrors in the set of load-sharing mirrors are aborted. If the `snapmirror abort` command identifies a lagging load-sharing mirror, then only the SnapMirror transfer associated with the lagging load-sharing mirror is aborted.

After the `snapmirror abort` command successfully completes its operation, the volume on the receiving side of the transfer might contain a restart checkpoint. The restart checkpoint can be used by a subsequent transfer to restart and continue the aborted SnapMirror transfer.

This command is supported for SnapMirror relationships with the field *"Relationship Capability"* showing as either *"8.2 and above"* or *"Pre 8.2"* in the output of the `snapmirror show` command.

This command is not supported for SnapMirror active sync relationships with policy of type *automated-failover* or *automated-failover-duplex* in admin privilege level.

The use of wildcards in parameter values is not supported from the source Vserver or cluster for relationships with *"Relationship Capability"* of *"8.2 and above"*.

You can use this command from the source or the destination Vserver or cluster for FlexVol volume relationships.

For SnapMirror Synchronous relationships, this command aborts any ongoing transfer and takes the relationship *OutOfSync*. This can result in primary client IO failure for relationships with a policy of type *strict-sync-mirror*. Instead, the best practice recommendation is to use the `snapmirror quiesce` command.

Beginning with ONTAP 9.12.1, for SnapMirror Synchronous relationships on platforms that support NDO, `snapmirror abort` triggered in *"InSync"* state will not change the relationship status to *"OutOfSync"*. Instead, it will internally trigger the feature *"Fast Resync"* which will try to keep the relationship status *"InSync"*.

For Vserver SnapMirror relationships, this command must be run only from the cluster containing the destination Vserver.

## Parameters

### { [-S, -source-path

{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Source Path

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or the volume (volume). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (cluster) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form *hostip:/share/share-name*. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form *hostip:/lun/name*.

### | [-source-cluster <Cluster name>] - Source Cluster

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source`, `-vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

### [-source-vserver <vserver name>] - Source Vserver

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

### [-source-volume <volume name>] - Source Volume }

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

### { -destination-path

{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path

This parameter specifies the destination endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or volume (volume). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (cluster) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with AltaVault destinations, the destination endpoint is specified in the form *hostip:/share/share-name*. For relationships with SolidFire destinations, the destination endpoint is specified in the form *hostip:/lun/name*.

### | [-destination-cluster <Cluster name>] - Destination Cluster

Specifies the destination cluster of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and `-destination-volume` must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

### **-destination-vserver <vserver name> - Destination Vserver**

Specifies the destination Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-destination-volume` and for relationships with "*Relationship Capability*" of "*Pre 8.2*", `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

### **-destination-volume <volume name> - Destination Volume }**

Specifies the destination volume of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and for relationships with "*Relationship Capability*" of "*Pre 8.2*", `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

### **[-h, -hard <true>] - Discard Restart Checkpoint**

If this option is specified `true`, the restart checkpoint is discarded and the destination volume is restored to the last Snapshot copy that was successfully transferred. You can use the `-hard` option to discard the restart checkpoint of a previous transfer attempt which forces the subsequent transfer to start with a fresh Snapshot copy on the destination volume. This option can only be used from the destination Vserver or cluster. This parameter is not supported for relationships with non-Data ONTAP endpoints.

### **[-w, -foreground <true>] - Foreground Process**

This specifies whether the operation runs as a foreground process. If this parameter is specified, the default setting is `true` (the operation runs in the foreground). When set to `true`, the command will not return until the process completes. This parameter is only applicable to relationships with "*Relationship Capability*" of "*Pre 8.2*".

## **Examples**

To stop the active SnapMirror replication to the destination volume `vs2.example.com:dept_eng_dp_mirror1`, type the following command:

```
vs2.example.com::> snapmirror abort -destination-path
vs2.example.com:dept_eng_dp_mirror1
```

For relationships with "*Relationship Capability*" of "*Pre 8.2*", to stop the active SnapMirror replication to the destination volume `cluster2://vs2.example.com/dept_eng_dp_mirror1`, type the following command:

```
cluster2::> snapmirror abort -destination-path
cluster2://vs2.example.com/dept_eng_dp_mirror1
```

To stop the active SnapMirror replication to the destination Vserver `dvs1.example.com`, type the following command:

```
cluster2::> snapmirror abort -destination-path
dvs1.example.com:
```

To stop the active SnapMirror replication to the destination Application Consistency Group `cg_dst` in Vserver `vs2.example.com`, type the following command:

```
cluster2::> snapmirror abort -destination-path
vs2.example.com:/cg/cg_dst
```

## Related Links

- [snapmirror show](#)
- [snapmirror quiesce](#)

# snapmirror break

## Make SnapMirror destination writable

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror break` command breaks a SnapMirror relationship between a source and destination endpoint of a data protection mirror. The destination endpoint can be a Vserver, volume, Application Consistency Group or SolidFire endpoint. When Data ONTAP breaks the relationship, if the endpoint is a volume, Application Consistency Group or SolidFire endpoint, the destination is made read/write and can diverge from the source volume, client redirection is turned off on the destination, the restart checkpoint is cleared, and the clients can see the latest Snapshot copy. If the endpoint is a Vserver, the subtype of the destination Vserver is changed to `default`, volumes in the destination Vserver are made read/write and the clients can now access the Vserver namespace for modifications. For SolidFire destination endpoints, the `snapmirror break` command is only supported if the endpoint is in a SnapMirror relationship. For an Application Consistency Group the item volumes are changed to read/write volumes and can diverge from the corresponding items on the source.

Subsequent manual or scheduled SnapMirror updates to the broken relationship will fail until the SnapMirror relationship is reestablished using the [snapmirror resync](#) command.

This command applies to data protection mirrors. For vault relationships, this command is only intended for use when preparing for a Data ONTAP revert operation (see the `-delete-snapshots` parameter in advanced privilege level). This command is not intended for use with load-sharing mirrors.

For relationships with a policy of type `strict-sync-mirror` or `sync-mirror`, the relationship must be *Quiesced* before running the `snapmirror break` command.

This command is supported for SnapMirror relationships with the field `"Relationship Capability"` showing as either `"8.2 and above"` or `"Pre 8.2"` in the output of the [snapmirror show](#) command.

The `snapmirror break` command must be used from the destination Vserver or cluster.

## Parameters

**{ [-S, -source-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Source Path**

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or the volume (volume). To support relationships with "Relationship Capability" of "Pre 8.2", a format which also includes the name of the cluster (cluster) is provided. The "Pre 8.2" format cannot be used when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above". For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form *hostip:/share/share-name*. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form *hostip:/lun/name*.

**| [-source-cluster <Cluster name>] - Source Cluster**

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source`, `-vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with "Relationship Capability" of "Pre 8.2". This parameter cannot be specified when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above".

**[-source-vserver <vserver name>] - Source Vserver**

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with "Relationship Capability" of "Pre 8.2", `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

**[-source-volume <volume name>] - Source Volume }**

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with "Relationship Capability" of "Pre 8.2", `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

**{ -destination-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path**

This parameter specifies the destination endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or volume (volume). To support relationships with "Relationship Capability" of "Pre 8.2", a format which also includes the name of the cluster (cluster) is provided. The "Pre 8.2" format cannot be used when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above". For SnapMirror relationships with AltaVault destinations, the destination endpoint is specified in the form *hostip:/share/share-name*. For relationships with SolidFire destinations, the destination endpoint is specified in the form *hostip:/lun/name*.

**| [-destination-cluster <Cluster name>] - Destination Cluster**

Specifies the destination cluster of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and `-destination-volume` must also be specified. This parameter is only applicable for relationships with "Relationship Capability" of "Pre 8.2". This parameter cannot be specified when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above".

### **-destination-vserver <vserver name> - Destination Vserver**

Specifies the destination Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-destination-volume` and for relationships with "*Relationship Capability*" of "*Pre 8.2*", `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

### **-destination-volume <volume name> - Destination Volume }**

Specifies the destination volume of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and for relationships with "*Relationship Capability*" of "*Pre 8.2*", `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

### **[-f, -force <true>] - Force**

If this parameter is specified, the command proceeds without prompting for confirmation.

### **[-w, -foreground <true>] - Foreground Process**

This specifies whether the operation runs as a foreground process. If this parameter is specified, the default setting is *true* (the operation runs in the foreground). When set to *true*, the command will not return until the process completes. This parameter is only applicable to relationships with "*Relationship Capability*" of "*Pre 8.2*".

### **[-delete-snapshots <true>] - Delete Snapshots for Revert (privilege: advanced)**

Using this parameter causes break to delete Snapshot copies on a vault destination so that the system can be reverted. Note that the only Snapshot copies that will be deleted are those that were created with the current version of Data ONTAP. Any Snapshot copies that might be present created with a different version will not be deleted. This parameter is not supported for SnapLock Compliance destinations.

### **[-s, -restore-destination-to-snapshot <text>] - Restore Destination to Snapshot Copy**

This optional parameter specifies the Snapshot copy to which the destination volume is restored after a successful break operation. If the parameter is not specified, the destination is restored to the latest Snapshot copy. This parameter is not supported for Vserver, Application Consistency Group or FlexGroup relationships or SnapLock Compliance destinations.

### **[-recover <true>] - Recover (privilege: advanced)**

When a SnapMirror break operation fails on a FlexGroup or Application Consistency Group relationship, a subset of the destination FlexGroup or Application Consistency Group constituents could have been made writable and subsequently user data could have been written to these constituents. To recover from this failure, you can execute the `snapmirror break` command again specifying the `-recover` parameter. All constituents will be restored to the latest Snapshot copy, and any writes to the read-write constituents will be lost. This parameter is applicable only for SnapMirror relationships with FlexGroup and or Application Consistency Group endpoints.

## **Examples**

To stop the SnapMirror replication to the destination volume `vs2.example.com:dept_eng_dp_mirror1`, type the following command:

```
vs2.example.com::> snapmirror break -destination-path
vs2.example.com:dept_eng_dp_mirror1
```



For relationships with "*Relationship Capability*" of "*Pre 8.2*", to stop the SnapMirror replication to the destination volume `cluster2://vs2.example.com/dept_eng_dp_mirror1`, type the following command:

```
cluster2::> snapmirror break
      -destination-path cluster2://vs2.example.com/dept_eng_dp_mirror1
```

To stop replication to the destination Vserver `dvs1.example.com` of a Vserver SnapMirror relationship, type the following command:

```
cluster2::> snapmirror break -destination-path dvs1.example.com:
```

To stop SnapMirror replication to the destination Application Consistency Group `app_cg_dst` in Vserver `vs2.example.com`, type the following command:

```
vs2.example.com::> snapmirror break -destination-path
      vs2.example.com:/cg/app_cg_dst
```

## Related Links

- [snapmirror resync](#)
- [snapmirror show](#)

# snapmirror create

Create a new SnapMirror relationship

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror create` command creates a SnapMirror relationship between a source and destination endpoint. You can use this command to create a data protection relationship, an extended data protection relationship, or a load-sharing relationship between FlexVol volumes. You can also use it to create a data protection relationship between Vservers. A SnapMirror relationship between Vservers can only be created if the system containing the source Vserver is also running Data ONTAP 8.3 or later. You can also use the `snapmirror create` command to create an extended data protection relationship between FlexGroups. FlexGroups only support extended data protection relationships. A SnapMirror relationship between FlexGroups is only supported if the system containing the source FlexGroup volume is also running Data ONTAP 9.1.0 or later. The source or destination of a FlexGroup SnapMirror relationship cannot be the source or destination of any non-FlexGroup SnapMirror relationship.

You can use the `snapmirror create` command to create a data protection relationship or an extended data protection relationship between SnapLock source and SnapLock destination endpoints. When the cluster containing the source is running ONTAP 9.5.0 or later, the default relationship type is extended data protection (XDP), otherwise it is data protection (DP).

A SnapMirror relationship can also be created between a non-SnapLock source and a SnapLock destination to WORM-protect Snapshot copies. In this case, the relationship type is always set to extended data protection (XDP).

The `snapmirror create` command can be used to create an extended data protection (XDP) relationship between a Data ONTAP volume and a non-Data ONTAP endpoint that supports SnapMirror (AltaVault, Solidfire). AltaVault endpoints can only be used as destinations. SolidFire endpoints can be used as sources or destinations.

The `snapmirror create` command can be used to create a synchronous relationship between FlexVol volumes, which provides zero Recovery Point Objective (RPO) data protection. SnapMirror Synchronous supports two policy types, `sync-mirror` and `strict-sync-mirror`. Upon a permanent replication failure, the `strict-sync-mirror` variant restricts further client IO on the primary, whereas the `sync-mirror` variant does not.

The `snapmirror create` command can be used to create a SnapMirror active sync relationship between Consistency Groups that have FlexVol volumes as items. This provides zero RPO and zero Recovery Time Objective (RTO) data protection, also known as transparent application failover, for SAN workloads. An SnapMirror active sync relationship supports the `automated-failover` or `automated-failover-duplex` policy type and can only be created when the systems containing the source items and the destination items are running ONTAP 9.9.1 or later.

Asynchronous SnapMirror relationships can also be created between the source Application Consistency Groups and the destination Application Consistency Groups. In this case, the relationship type is always set to extended data protection (XDP). An Application Consistency Group consists of FlexVol volumes as items and the SnapMirror relationships can only be created when the systems containing the source items and the destination items are running ONTAP 9.13.1 or later.

Before using this command to create a SnapMirror relationship between Vservers, you typically create a source and destination Vserver using the `vserver create` command. The source Vserver should be of `subtypedefault` and the destination Vserver of `subtypedp-destination`. Also, before creating the relationship between Vservers, you must setup Vserver peer by using the `vserver peer create` command between the source and destination Vservers. A Vserver relationship cannot be created between two Vservers within the same cluster. The root volume of the destination Vserver will be read-write and data from the source Vserver's root volume will not be replicated. Therefore there will be no volume level relationship created between the root volumes of the two Vservers.

After creating the relationship, the destination Vserver must be initialized by using the `snapmirror initialize` command.

Before using this command to create a volume SnapMirror relationship, you typically create a source and destination volume using the `volume create` command. The source volume should be in the online state and a read-write (RW) type. The destination volume should be in the online state and a data protection (DP) type. For FlexGroup SnapMirror relationships, the source and destination FlexGroups must be spread over the same number of aggregates as specified in the `-aggr-list` parameter with the same number of constituents per aggregate as specified in the `-aggr-list-multiplier` parameter of the `volume create` command.

When a FlexGroup SnapMirror relationship is created, normally hidden relationships are also created for the constituent volumes. These relationships can be seen by using the `-expand` parameter of the `snapmirror show` command. Source information for these relationships can be seen using the `-expand` parameter of the `snapmirror list-destinations` command. Other SnapMirror commands are disabled for FlexGroup constituent relationships and FlexGroup constituent volumes.

If all systems involved are running Data ONTAP version 8.2 and later, a Vserver peering relationship must be set up using the `vserver peer create` command between the source and the destination Vservers to create a

relationship between the source and destination volumes. To enable interoperability with Data ONTAP 8.1, if the source volume is on a storage system running clustered Data ONTAP 8.1, the cluster administrator can create a data protection relationship between the source and destination volumes without a Vserver peering relationship between the source and destination Vservers. These relationships are managed the same way as on Data ONTAP 8.1 and the *"Relationship Capability"* field, as shown in the output of the `snapmirror show` command, is set to *"Pre 8.2"*.



SnapMirror relationships, except load-sharing relationships, which are created between two volumes which are both on a storage system running Data ONTAP version 8.2 and later have the *"Relationship Capability"* field set to *"8.2 and above"*.

Load-sharing mirrors must be confined to a single Vserver; they are not allowed to span Vservers. Load-sharing relationships are created with the *"Relationship Capability"* field set to *"Pre 8.2"* even if both the source and destination volumes are on a storage system running Data ONTAP version 8.2 and later. There is no *"8.2 and above"* implementation for load-sharing relationships.

A set of load-sharing mirrors can have one or more destination volumes. You create separate SnapMirror relationships between the common source volume and each destination volume to create the set of load-sharing mirrors.

The source or destination of a load-sharing relationship cannot be the source or destination of any other SnapMirror relationship.

After creating the relationship, the destination volume can be initialized using the `snapmirror initialize` command. The destination volumes in a set of load-sharing mirrors are initialized using the `snapmirror initialize-ls-set` command.

The `snapmirror create` command must be used from the destination Vserver or cluster.

## Parameters

**{ -S, -source-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Source Path**

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or the volume (volume). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (cluster) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form `hostip:/share/share-name`. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form `hostip:/lun/name`.

**| [-source-cluster <Cluster name>] - Source Cluster**

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source`, `-vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

### **-source-vserver <vserver name> - Source Vserver**

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

### **[-source-volume <volume name>] - Source Volume }**

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

### **{ -destination-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path**

This parameter specifies the destination endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or volume (volume). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (cluster) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with AltaVault destinations, the destination endpoint is specified in the form `hostip:/share/share-name`. For relationships with SolidFire destinations, the destination endpoint is specified in the form `hostip:/lun/name`.

### **[-destination-cluster <Cluster name>] - Destination Cluster**

Specifies the destination cluster of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and `-destination-volume` must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

### **-destination-vserver <vserver name> - Destination Vserver**

Specifies the destination Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-destination-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

### **-destination-volume <volume name> - Destination Volume }**

Specifies the destination volume of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

### **[-type <snapmirrorType>] - Relationship Type**

This parameter specifies the type of SnapMirror relationship that will be created. You can create a data protection (DP) relationship (deprecated), an extended data protection (XDP) relationship or a load-sharing (LS) relationship. The default value is XDP for all relationships except for Vserver DR relationships. FlexGroup volumes, Application Consistency Groups and non-Data ONTAP endpoints support only XDP relationships. For FlexVol volume relationships, when DP is specified as the type, an XDP relationship will be created unless a relationship of type DP in the opposite direction already exists. In that case the new relationship will be of type DP.

### **[-vserver <vserver name>] - Managing Vserver**

If this optional parameter is specified, it designates the managing Vserver. The managing Vserver is authorized to use SnapMirror commands to manage the SnapMirror relationship. The `-vserver` parameter is currently a reserved parameter.

### **[-schedule <text>] - SnapMirror Schedule**

This optional parameter designates the name of the schedule which is used to update the SnapMirror relationship. If you do not designate a schedule, updates are not scheduled, so you must update the SnapMirror relationship manually using the [snapmirror update](#) command or, in the case of a set of load-sharing mirrors, using the [snapmirror update-ls-set](#) command.



You define and name a schedule using the [job schedule cron create](#) command.

### **[-policy <sm\_policy>] - SnapMirror Policy**

This optional parameter designates the name of the SnapMirror policy which is associated with the SnapMirror relationship. For FlexVol volume relationships, the default policy when the data protection (DP) type is specified is *MirrorAllSnapshots* and the default policy when no type is specified or when the extended data protection (XDP) type is specified is *MirrorAndVault*. For FlexGroup volume and Application Consistency Group relationships, the *MirrorAndVault* policy is applied. For SnapMirror relationships between SnapLock volumes, if no policy is specified the default policy *MirrorAllSnapshots* is applied. For relationships with a SolidFire endpoint, there is no default policy. For these relationships a policy as described below must be specified. This parameter is not applicable to relationships with "Relationship Capability" of "Pre 8.2".

In clustered Data ONTAP 8.2 data protection (DP) relationships were used for mirroring, while extended data protection (XDP) relationships were used for vaulting. In clustered Data ONTAP 8.3 extended data protection (XDP) relationships support two more use cases, mirroring and unified mirror-vault. The exact behavior of an extended data protection (XDP) relationship is governed by the `snapmirror policy` associated with that relationship. In clustered Data ONTAP 8.3 the `snapmirror policy` has a new field `type` to indicate how the relationships with that policy will behave. The supported types are *async-mirror* (mirroring), *vault* (vaulting) and *mirror-vault* (unified mirroring and vault). For XDP relationships between a Data ONTAP source volume and an AltaVault destination endpoint, only policies of type *vault* are supported. For XDP relationships between a Data ONTAP source volume and a SolidFire destination endpoint, only policies of type *async-mirror* without an *all\_source\_snapshots* rule are supported. For XDP relationships between a SolidFire source endpoint and a Data ONTAP destination volume, only policies of type *async-mirror* without an *all\_source\_snapshots* rule, and policies of type *mirror-vault* are supported. SnapMirror policies of type *async-mirror* associated with FlexVol volume relationships when relationship type DP is specified or when no relationship type is specified, must include the label *all\_source\_snapshots*. Refer to the man page for the [snapmirror policy create](#) command for more information.



You define and name a policy using the [snapmirror policy create](#) command.

### **[-tries <integer\_or\_unlimited>] - Tries Limit**

This optional parameter specifies the maximum number of times to attempt each manual or scheduled transfer for a SnapMirror relationship. The default is eight times. The `-tries` parameter can be set to `0` to disable manual and scheduled updates for the SnapMirror relationship. This parameter is only applicable to relationships with "Relationship Capability" of "Pre 8.2". For relationships with "8.2 and above" capability, the tries limit is controlled by the value of tries in the SnapMirror policy that is associated with the relationship.

### **[-k, -throttle <throttleType>] - Throttle (KB/sec)**

This optional parameter limits the network bandwidth used for transfers. It configures for the relationship the maximum rate (in Kbytes/sec) at which data can be transferred. If no throttle is configured, by default the SnapMirror relationship fully utilizes the network bandwidth available. You can also configure the relationship to fully use the network bandwidth available by explicitly setting the throttle to *unlimited* or *0*. The minimum effective throttle value is four Kbytes/sec, so if you specify a throttle value between *1* and *4*, it will be treated as *4*. For FlexGroup volume relationships, the throttle value is applied individually to each constituent relationship. The `-throttle` parameter does not affect load-sharing mirrors and other SnapMirror relationships with "Relationship Capability" of "Pre 8.2" confined to a single cluster.

### **[-cg-item-mappings {<source volume>:@<destination volume>|<source item path>:@<destination item path>}] - Consistency Group Item Mappings**

This optional parameter specifies a list of the consistency group (CG) item mappings. It is considered only if the supplied source and destination path values denote valid CG paths. For CG level relationships, this parameter must be specified. The value for this parameter must contain mappings of one or more pairs of constituent source and destination volumes of the form *srcvol1:@dstvol1,...*.

### **[-identity-preserve {true|false}] - Identity Preserve Vserver DR**

Specifies whether or not the identity of the source Vserver is replicated to the destination Vserver of the Vserver SnapMirror relationship that will be created. If this parameter is set to true, the source Vserver's configuration will additionally be replicated to the destination. If the parameter is set to false, then only the source Vserver's volumes and RBAC configuration are replicated to the destination. This parameter is applicable only for SnapMirror relationships with Vserver endpoints. The default value is false.

### **[-is-auto-expand-enabled {true|false}] - Is Auto Expand Enabled**

This optional parameter specifies whether or not a FlexGroup SnapMirror relationship and its destination FlexGroup volume should be auto-expanded if the source FlexGroup volume is expanded. This parameter is supported only for FlexGroup SnapMirror relationships. The default value is true. For Application Consistency Groups, this parameter is always set to true.

### **[-backoff-level {high|medium|none}] - SM Backoff Level due to Client Ops**

This optional parameter specifies the SnapMirror backoff level due to client ops. This parameter is supported only for FlexVol SnapMirror relationships. The default value is *high*.

## **Examples**

To create an extended data protection relationship between the source endpoint

*vs1.example.com:dept\_eng*, and the destination endpoint *vs2.example.com:dept\_eng\_dp\_mirror2*, with the default policy of *MirrorAndVault*, type the following command:

```
vs2.example.com::> snapmirror create -destination-path
vs2.example.com:dept_eng_dp_mirror2 -source-path
vs1.example.com:dept_eng
```

To create an extended data protection relationship between the source FlexGroup

*vs1.example.com:fg\_src* and the destination FlexGroup *vs2.example.com:fg\_dst*, with the default policy of *MirrorAndVault*, type the following command:

```
vs2.example.com::> snapmirror create -destination-path
vs2.example.com:fg_dst -source-path
vs1.example.com:fg_src
```

To create a synchronous SnapMirror relationship between the source Flexvol *vs1.example.com:vol\_log*, and the destination Flexvol *vs2.example.com:vol\_log\_sync\_dp* when the source cluster is running ONTAP 9.5 or above, type the following command:

```
vs2.example.com::> snapmirror create -destination-path
vs2.example.com:vol_log_sync_dp -source-path
vs1.example.com:vol_log -policy Sync
```

To create a strict synchronous SnapMirror relationship between the source Flexvol *vs1.example.com:vol\_log*, and the destination Flexvol *vs2.example.com:vol\_log\_sync\_dp* when the source cluster is running ONTAP 9.5 or above, type the following command:

```
vs2.example.com::> snapmirror create -destination-path
vs2.example.com:vol_log_sync_dp -source-path
vs1.example.com:vol_log -policy StrictSync
```

To create a data protection mirror between the source endpoint

*cluster1://vs1.example.com/dept\_eng*, and the destination endpoint

*cluster2://vs2.example.com/dept\_eng\_dp\_mirror2* when the source cluster is running Data ONTAP 8.1 software, type the following command:

```
cluster2::> snapmirror create -destination-path
cluster2://vs2.example.com/dept_eng_dp_mirror2 -source-path
cluster1://vs1.example.com/dept_eng
-type DP
```

To create a load-sharing mirror between the source endpoint *cluster1://vs1.example.com/vs1\_root* which is a Vserver root volume, and the destination endpoint

*cluster1://vs1.example.com/vs1\_root\_ls1* with the schedule named *5min* used to update the relationship, type the following command:

```
cluster1::> snapmirror create
-destination-path cluster1://vs1.example.com/vs1_root_ls1
-source-path cluster1://vs1.example.com/vs1_root -type LS
-schedule 5min
```

To create a SnapMirror relationship between the source Vserver *vs1.example.com*, and the destination Vserver *dvs1.example.com* with the schedule named *hourly* used to update the relationship, type the following command:

```
cluster2::> snapmirror create
      -destination-path dvs1.example.com:
      -source-path vs1.example.com:
      -schedule hourly
```

To create an extended data protection (XDP) relationship between the Data ONTAP source endpoint *vs1.example.com:data\_ontap\_vol*, and the AltaVault destination endpoint *10.0.0.11:/share/share1*, type the following command:

```
vs1.example.com::> snapmirror create -destination-path
10.0.0.11:/share/share1 -source-path
vs1.example.com:data_ontap_vol
-type XDP
```

To create an extended data protection (XDP) relationship between the SolidFire source endpoint *10.0.0.12:/lun/0001*, and the Data ONTAP destination endpoint *vs2.example.com:data\_ontap\_vol2*, type the following command:

```
vs2.example.com::> snapmirror create -source-path
10.0.0.12:/lun/0001 -destination-path
vs2.example.com:data_ontap_vol2
-type XDP -policy MirrorLatest
```

To create an SnapMirror active sync relationship with the following attributes:

- It is between the source Consistency Group *cg\_src* in *Vserver vs1.example.com*, and the destination Consistency Group *cg\_dst* in *Vserver vs2.example.com*.
- It has item mappings between volumes *srcvol1* and *srcvol2* and volumes *dstvol1* and *dstvol2*.
- It can use the default policy named *AutomatedFailOver* that has a policy type of *automated-failover* or *AutomatedFailOverDuplex* that has a policy type of *automated-failover-duplex*.

Type the following command from the destination cluster:

```
destination::> snapmirror create -destination-path
vs2.example.com:/cg/cg_dst -source-path
vs1.example.com:/cg/cg_src
-policy AutomatedFailOver
-cg-item-mappings srcvol1:@dstvol1,srcvol2:@dstvol2
```

To create an extended data protection relationship between the source Application Consistency Group *vs1.example.com:/cg/cg\_src* and the destination Application Consistency Group *vs2.example.com:/cg/cg\_dst*, with the policy of *MirrorAllSnapshots*, type the following command:



```
destination::> snapmirror create -destination-path
                vs2.example.com:/cg/cg_dst -source-path
                vs1.example.com:/cg/cg_src
                -policy MirrorAllSnapshots
                -cg-item-mappings srcvol1:@dstvol1,srcvol2:@dstvol2
```

## Related Links

- [vserver create](#)
- [vserver peer create](#)
- [snapmirror initialize](#)
- [volume create](#)
- [snapmirror show](#)
- [snapmirror list-destinations](#)
- [snapmirror initialize-ls-set](#)
- [snapmirror update](#)
- [snapmirror update-ls-set](#)
- [job schedule cron create](#)
- [snapmirror policy create](#)

## snapmirror delete

Delete a SnapMirror relationship

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### Description

The `snapmirror delete` command removes the SnapMirror relationship between a source endpoint and a destination endpoint. The destination endpoint can be a Vserver, a volume, an Application Consistency Group, or a non-Data ONTAP endpoint. The Vservers, volumes, FlexVol volumes of the Consistency Group and non-Data ONTAP destinations are not destroyed and Snapshot copies on the volumes are not removed.

For relationships with SolidFire endpoints, the SnapMirror source commands [snapmirror release](#) and [snapmirror list-destinations](#) are not supported. Therefore, Snapshot copies that are locked by SnapMirror on the source container cannot be cleaned up using the [snapmirror release](#) command. If the source container is a Data ONTAP volume, in order to reclaim space captured in the base Snapshot copy on the volume, issue a `snapshot delete` command specifying the `-ignore-owners` parameter in `diag` privilege level. To reclaim space captured in a Snapshot copy locked by SnapMirror on a SolidFire system, refer to SolidFire documentation.

The `snapmirror delete` command fails if a SnapMirror transfer for the SnapMirror relationship is in progress for relationships with "*Relationship Capability*" of "*Pre 8.2*". For relationships with "*8.2 and above*" capability the delete will succeed even if a transfer is in progress and the transfer will ultimately stop.

A set of load-sharing mirrors can contain multiple destination volumes, each destination volume having a separate SnapMirror relationship with the common source volume. When used on one of the SnapMirror relationships from the set of load-sharing mirrors, the `snapmirror delete` command deletes the specified SnapMirror relationship from the set of load-sharing mirrors.

The `snapmirror delete` command preserves the read-write or read-only attributes of the volumes of a SnapMirror relationship after the relationship is deleted. Therefore, a read-write volume that was the source of a SnapMirror relationship retains its read-write attributes, and a data protection volume or a load-sharing volume that was a destination of a SnapMirror relationship retains its read-only attributes. Similarly, the `subtype` attribute of source and destination Vservers is not modified when a Vserver SnapMirror relationship is deleted.



When a SnapMirror relationship from a set of load-sharing mirrors is deleted, the destination volume becomes a data protection volume and retains the read-only attributes of a data protection volume.

For relationships with a policy of type `strict-sync-mirror` or `sync-mirror`, the relationship must be `Quiesced` before it can be deleted.

This command is supported for SnapMirror relationships with the field `"Relationship Capability"` showing as either `"8.2 and above"` or `"Pre 8.2"` in the output of the `snapmirror show` command.

For relationships with `"Relationship Capability"` of `"8.2 and above"`, the `snapmirror delete` command must be used from the destination Vserver or cluster. The SnapMirror relationship information is deleted from the destination Vserver, but no cleanup or deletion is performed on the source Vserver. The `snapmirror release` command must be issued on the source Vserver to delete the source relationship information.

For relationships with `"Relationship Capability"` of `"Pre 8.2"`, you can use this command from the source or from the destination cluster. When used from the destination cluster, the SnapMirror relationship information on the source and destination clusters is deleted. When used from the source cluster, only the SnapMirror relationship information on the source cluster is deleted.

## Parameters

**{ [-S, -source-path  
{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}] - Source Path**

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (`vserver`) and/or the volume (`volume`). To support relationships with `"Relationship Capability"` of `"Pre 8.2"`, a format which also includes the name of the cluster (`cluster`) is provided. The `"Pre 8.2"` format cannot be used when operating in a Vserver context on relationships with `"Relationship Capability"` of `"8.2 and above"`. For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form `hostip:/share/share-name`. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form `hostip:/lun/name`.

**| [-source-cluster <Cluster name>] - Source Cluster**

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source`, `-vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with `"Relationship Capability"` of `"Pre 8.2"`. This parameter cannot be specified when operating in a Vserver context on relationships with `"Relationship Capability"` of `"8.2 and`

above".

**[-source-vserver <vserver name>] - Source Vserver**

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

**[-source-volume <volume name>] - Source Volume }**

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

**{ -destination-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path**

This parameter specifies the destination endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or volume (volume). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (cluster) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with AltaVault destinations, the destination endpoint is specified in the form *hostip:/share/share-name*. For relationships with SolidFire destinations, the destination endpoint is specified in the form *hostip:/lun/name*.

**[-destination-cluster <Cluster name>] - Destination Cluster**

Specifies the destination cluster of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and `-destination-volume` must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

**-destination-vserver <vserver name> - Destination Vserver**

Specifies the destination Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-destination-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

**-destination-volume <volume name> - Destination Volume }**

Specifies the destination volume of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

**[-f, -force <true>] - Force**

If specified, the delete proceeds even if it cannot clean up all artifacts of the relationship.

**[-w, -foreground <true>] - Foreground Process**

This specifies whether the operation runs as a foreground process. If this parameter is specified, the default setting is *true* (the operation runs in the foreground). When set to *true*, the command will not return until

the process completes. This parameter is only applicable to relationships with *"Relationship Capability"* of *"Pre 8.2"*.

## Examples

To delete the SnapMirror relationship with the destination endpoint `vs2.example.com:dept_eng_dp_mirror4`, type the following command:

```
vs2.example.com::> snapmirror delete -destination-path
vs2.example.com:dept_eng_dp_mirror4
```

For relationships with *"Relationship Capability"* of *"Pre 8.2"*, to delete the SnapMirror relationship with the destination endpoint `cluster2://vs2.example.com/dept_eng_dp_mirror4`, type the following command:

```
cluster2::> snapmirror delete -destination-path
cluster2://vs2.example.com/dept_eng_dp_mirror4
```

To delete the SnapMirror relationship with destination endpoint `dvs1.example.com:`, type the following command:

```
cluster2::> snapmirror delete -destination-path
dvs1.example.com:
```

To delete the SnapMirror active sync relationship with the destination Consistency Group `cg_dst` in Vserver `vs2.example.com`, type the following command from the destination cluster:

```
destination::> snapmirror delete -destination-path
vs2.example.com:/cg/cg_dst
```

To delete the Application Consistency Group SnapMirror relationship with the destination Application Consistency Group `app_cg_dst` in Vserver `vs2.example.com` and all item mappings, type the following command:

```
vs2.example.com::> snapmirror delete -destination-path
vs2.example.com:/cg/app_cg_dst
```

## Related Links

- [snapmirror release](#)
- [snapmirror list-destinations](#)
- [snapmirror show](#)

# snapmirror flexgroup-epuuid-prefix-prepare-to-downgrade

Prepares the system for downgrade

**Availability:** This command is available to *cluster* administrators at the *advanced* privilege level.

## Description

The `flexgroup-epuuid-prefix-prepare-to-downgrade` command prepares nodes to downgrade to a release without FlexGroup endpoint prefixed with the group endpoint uuid capability. Prior to disabling the capability, remove all FlexGroup SnapMirror object store relationships which have a source or destination endpoint prefixed with the group endpoint uuid.

## Examples

The following example disables the FlexGroup endpoint prefixed with group endpoint uuid capability in the local cluster:

```
cluster1::> snapmirror flexgroup-epuuid-prefix-prepare-to-downgrade
```

# snapmirror initialize-ls-set

Start a baseline load-sharing set transfer

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror initialize-ls-set` command initializes and updates a set of load-sharing mirrors. This command is usually used after the [snapmirror create](#) command is used to create a SnapMirror relationship for each of the destination volumes in the set of load-sharing mirrors. The initial transfers to empty load-sharing mirrors are baseline transfers done in parallel. During a baseline transfer Data ONTAP takes a Snapshot copy on the source volume to capture the current image of the source volume and transfers all of the Snapshot copies on the source volume to each of the destination volumes.

After the `snapmirror initialize-ls-set` command successfully completes, the last Snapshot copy transferred is made the exported Snapshot copy on the destination volumes.

The parameter that identifies the set of load-sharing mirrors is the source volume. Data and Snapshot copies are transferred from the source volume to all up-to-date destination volumes in the set of load-sharing mirrors.

Use the [snapmirror initialize](#) command to add and initialize a new destination volume to an existing set of load-sharing mirrors.



Even if the load-sharing set only has one mirror, you still need to use the `snapmirror initialize-ls-set` command to initialize the set. The [snapmirror initialize](#) command can only be used to initialize a new destination volume, if the load-sharing set has already been initialized.

This command is only supported for SnapMirror relationships with the field *"Relationship Capability"*

showing as *"Pre 8.2"* in the output of the `snapmirror show` command.

## Parameters

**{ -S, -source-path**

**{<[vserver:] [volume]>|<[cluster:]//vserver/[volume]>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Source Path**

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or the volume (volume). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (cluster) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form *hostip:/share/share-name*. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form *hostip:/lun/name*.

**[-source-cluster <Cluster name>] - Source Cluster**

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source`, `-vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

**-source-vserver <vserver name> - Source Vserver**

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

**-source-volume <volume name> - Source Volume }**

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

**[-w, -foreground <true>] - Foreground Process**

This specifies whether the operation runs as a foreground process. If this parameter is specified, the default setting is *true* (the operation runs in the foreground). When set to *true*, the command will not return until the process completes. This parameter is only applicable to relationships with *"Relationship Capability"* of *"Pre 8.2"*.

## Examples

To initialize the group of load-sharing mirrors for the source endpoint `//vs1.example.com/vs1_root`, type the following command:

```
cluster1::> snapmirror initialize-ls-set -source-path
//vs1.example.com/vs1_root
```

## Related Links

- [snapmirror create](#)
- [snapmirror initialize](#)
- [snapmirror show](#)

# snapmirror initialize

Start a baseline transfer

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror initialize` command initializes the destination Vserver, volume, Application Consistency Group or a non-Data ONTAP endpoint of a SnapMirror relationship. The command behaves differently between data protection (DP), extended data protection (XDP) and load-sharing (LS) relationships.

If you specify a *sync-mirror* or *strict-sync-mirror* type policy, the `snapmirror initialize` command creates and initializes a synchronous relationship and brings it InSync, providing zero RPO data protection.

If you are initializing a relationship with a policy of type *automated-failover* the `snapmirror initialize` command initializes a SnapMirror active sync relationship and brings the status to InSync, providing zero RPO and zero RTO data protection.

If you are initializing a relationship with a policy of type *automated-failover-duplex* the `snapmirror initialize` command initializes a SnapMirror active sync relationship and brings the status to InSync, providing zero RPO, zero RTO data protection and load balanced solution.

SnapMirror active sync is only supported for SAN. As a result, NAS access is not supported on SnapMirror active sync volumes and SnapMirror active sync initialization is not supported on volumes with NAS access.

For data protection (DP) and extended data protection (XDP) relationships, the `snapmirror initialize` command initializes the destination volume.

For load-sharing (LS) relationships, the `snapmirror initialize` command initializes a new load-sharing mirror in an existing set of load-sharing mirrors. If the command finishes before the start of a scheduled or manual transfer of the set of load-sharing mirrors, the load-sharing mirror is up to date with the set of load-sharing mirrors; otherwise, the load-sharing mirror will be brought up to date at the next scheduled or manual transfer of the set of load-sharing mirrors.

The initial transfer to an empty destination volume is called a baseline transfer. During a baseline transfer for a data protection (DP) or extended data protection (XDP) relationship, the `snapmirror initialize` command takes a Snapshot copy on the source volume to capture the current image of the source volume. For data protection relationships, the `snapmirror initialize` command transfers all of the Snapshot copies up to and including the Snapshot copy created by it from the source volume to the destination volume. For extended data protection (XDP) relationships, the `snapmirror initialize` command behavior depends on the `snapmirror policy` associated with the relationship. If the policy type is *async-mirror* then depending on the rules in the policy it can transfer either all the Snapshot copies up to and including the Snapshot copy created by it or only the Snapshot copy created by it from the source volume to the destination volume. For extended data protection (XDP) relationships with policy type *vault* or *mirror-vault* the

`snapmirror initialize` transfers only the Snapshot copy created by it.

After the `snapmirror initialize` command successfully completes, the last Snapshot copy transferred is made the exported Snapshot copy on the destination volume.

You can use the `snapmirror initialize` command to initialize a specific load-sharing mirror that is new to the set of load-sharing mirrors. An initialize of the new load-sharing mirror should bring it up to date with the other up-to-date destination volumes in the set of load-sharing mirrors.



Using the `snapmirror initialize` command to initialize a set of load-sharing mirrors will not work. Use the `snapmirror initialize-ls-set` command to initialize a set of load-sharing mirrors.

If a SnapMirror relationship does not already exist, that is, the relationship was not created using the `snapmirror create` command, the `snapmirror initialize` command will implicitly create the SnapMirror relationship, with the same behaviors as described for the `snapmirror create` command before initializing the relationship. This implicit create feature is not supported for Vservers.

This command is supported for SnapMirror relationships with the field *"Relationship Capability"* showing as either *"8.2 and above"* or *"Pre 8.2"* in the output of the `snapmirror show` command.

For relationships with *"Relationship Capability"* of *"8.2 and above"*, you can track the progress of the operation using the `snapmirror show` command.

For relationships with *"Relationship Capability"* of *"Pre 8.2"*, a job will be spawned to operate on the SnapMirror relationship, and the job id will be shown in the command output. The progress of the job can be tracked using the `job show` and `job history show` commands.

The `snapmirror initialize` command must be used from the destination Vserver or cluster.

## Parameters

**{ [-S, -source-path**  
**<[vserver:] [volume]>|<[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<vserver:]/cg/[app-cgname]>} - Source Path**

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (`vserver`) and/or the volume (`volume`). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (`cluster`) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form `hostip:/share/share-name`. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form `hostip:/lun/name`.

**| [-source-cluster <Cluster name>] - Source Cluster**

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source`, `-vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.



### **[`-source-vserver <vserver name>`] - Source Vserver**

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

### **[`-source-volume <volume name>`] - Source Volume }**

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

### **{ `-destination-path`**

**{`<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>`} - Destination Path**

This parameter specifies the destination endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or volume (volume). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (cluster) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with AltaVault destinations, the destination endpoint is specified in the form *hostip:/share/share-name*. For relationships with SolidFire destinations, the destination endpoint is specified in the form *hostip:/lun/name*.

### **[`-destination-cluster <Cluster name>`] - Destination Cluster**

Specifies the destination cluster of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and `-destination-volume` must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

### **`-destination-vserver <vserver name>` - Destination Vserver**

Specifies the destination Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-destination-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

### **`-destination-volume <volume name>` - Destination Volume }**

Specifies the destination volume of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

### **[`-s, -source-snapshot <text>`] - Source Snapshot**

This optional parameter specifies the Snapshot copy that `snapmirror initialize` will use for the baseline transfer. For data protection (DP) relationships, the baseline transfer will include all of the Snapshot copies up to and including the specified Snapshot copy. For extended data protection (XDP) relationships, the `snapmirror initialize` command behavior depends on the `snapmirror policy` associated with the relationship. If the policy type is *async-mirror* then depending on the rules in the policy it can transfer either all the Snapshot copies up to and including the specified Snapshot copy or only the specified Snapshot copy from the source volume to the destination volume. For extended data

protection (XDP) relationships with policy type `vault` or `mirror-vault` the `snapmirror initialize` transfers only the specified Snapshot copy. This parameter is not supported for SnapMirror Synchronous relationships and relationships with "*Relationship Capability*" of "*Pre 8.2*".

### **[`-type <snapmirrorType>`] - Snapmirror Relationship Type**

Specifies the type of SnapMirror relationship if a relationship is implicitly created. This parameter is the same as the one used in the `snapmirror create` command.

### **[`-policy <sm_policy>`] - SnapMirror Policy**

This optional parameter designates the name of the SnapMirror policy which is associated with the SnapMirror relationship. If you do not designate a policy, the current policy will be retained. This parameter is not applicable to relationships with "*Relationship Capability*" of "*Pre 8.2*".



You define and name a policy using the `snapmirror policy create` command.

### **[`-k, -throttle <throttleType>`] - Throttle (KB/sec)**

This optional parameter limits the network bandwidth used for the initialize transfer. It sets the maximum rate (in Kbytes/sec) at which data can be transferred during the operation. If this parameter is not specified, the throttle value configured for the relationship with the `snapmirror create` or `snapmirror modify` command will be used. To fully use the network bandwidth available, set the throttle value to `unlimited` or `0`. The minimum throttle value is four Kbytes/sec, so if you specify a throttle value between `1` and `4`, it will be treated as if you specified `4`. For FlexGroup relationships, the throttle value is applied individually to each constituent relationship. For SnapMirror Synchronous relationships the throttle value is applicable only for asynchronous transfers performed as part of user-initiated operations. The `-throttle` parameter does not affect load-sharing transfers and transfers for other relationships with "*Relationship Capability*" of "*Pre 8.2*" confined to a single cluster.

### **[`-transfer-priority {low|normal}`] - Transfer Priority**

This optional parameter specifies the priority at which the transfer runs. The default value for this parameter is the value in the SnapMirror policy associated with this relationship. This parameter is not applicable to relationships with a "*Relationship Capability*" of "*Pre 8.2*".

### **[`-cg-item-mappings {<source volume>:@<destination volume>|<source item path>:@<destination item path>}`] - Consistency Group Item Mappings**

This optional parameter specifies a list of the consistency group (CG) item mappings. It is considered only if the supplied source and destination path values denote valid CG paths. For CG level relationships, this parameter must be specified. The value for this parameter must contain mappings of one or more pairs of constituent source and destination volumes of the form `srcvol1:@dstvol1,...`.

### **[`-is-auto-expand-enabled {true|false}`] - Is Auto Expand Enabled**

This optional parameter specifies whether or not a FlexGroup SnapMirror relationship and its destination FlexGroup should be auto-expanded if the source FlexGroup is expanded. This parameter is supported only for FlexGroup SnapMirror relationships. If this initialize is creating a new Snapmirror relationship, the default value is `true`. If it is not creating a new relationship, if a value is specified, it must match the current value for the existing relationship. If the parameter is not specified the existing value will be retained.

### **[`-backoff-level {high|medium|none}`] - SM Backoff Level due to Client Ops**

This optional parameter specifies the SnapMirror backoff level due to client ops. This parameter is supported only for FlexVol SnapMirror relationships. The default value is `high`.

## **[-w, -foreground <true>] - Foreground Process**

This specifies whether the operation runs as a foreground process. If this parameter is specified, the default setting is *true* (the operation runs in the foreground). When set to *true*, the command will not return until the process completes. This parameter is only applicable to relationships with "Relationship Capability" of "Pre 8.2".

## **Examples**

To start the initial transfer for the SnapMirror relationship with the destination endpoint `vs2.example.com:dept_eng_dp_mirror2` after the relationship has been created with the [snapmirror create](#) command, type the following command:

```
vs2.example.com::> snapmirror initialize -destination-path
vs2.example.com:dept_eng_dp_mirror2
```

For relationships with "Relationship Capability" of "Pre 8.2", to start the initial transfer for the SnapMirror relationship with the destination endpoint `cluster2://vs2.example.com/dept_eng_dp_mirror2` after the relationship has been created with the [snapmirror create](#) command, type the following command:

```
cluster2::> snapmirror initialize -destination-path
cluster2://vs2.example.com/dept_eng_dp_mirror2
```

To create a data protection mirror relationship between the source endpoint `vs1.example.com:dept_mkt`, and the destination endpoint `vs2.example.com:dep_mkt_dp_mirror`, and start the initial transfer, type the following command:

```
vs2.example.com::> snapmirror initialize -destination-path
vs.example.com2:dept_mkt_dp_mirror
-source-path vs1.example.com:dept_mkt
```

To create a data protection mirror relationship between the source endpoint `cluster1://vs1.example.com/dept_mkt`, and the destination endpoint `cluster2://vs2.example.com/dep_mkt_dp_mirror`, and start the initial transfer when the source cluster is running Data ONTAP 8.1 software, type the following command:

```
cluster2::> snapmirror initialize -destination-path
cluster2://vs2.example.com/dept_mkt_dp_mirror
-source-path cluster1://vs1.example.com/dept_mkt
```

To create an extended data protection (XDP) relationship between the Data ONTAP source endpoint `vs1.example.com:data_ontap_vol`, and the AltaVault destination endpoint `10.0.0.11:/share/share1`, and start the initial transfer, type the following command:

```
vs1.example.com::> snapmirror initialize -destination-path
10.0.0.11:/share/share1
-source-path vs1.example.com:data_ontap_vol -type XDP
```

To start the initial transfer for the Vserver SnapMirror relationship with destination endpoint `dvs1.example.com`: after the relationship was created with the `snapmirror create` command, type the following command:

```
cluster2::> snapmirror initialize -destination-path
dvs1.example.com:
```

To initialize the SnapMirror Synchronous relationship between FlexVols `vol_log` and `vol_log_sync_dp` and bring it to InSync, after it is created using the `snapmirror create` command, type the following command:

```
vs2.example.com::> snapmirror initialize -destination-path
vs2.example.com:vol_log_sync_dp
```

To create a strict SnapMirror Synchronous relationship between FlexVols `vol_log` and `vol_log_sync_dp`, to initialize it and bring it to InSync, type the following command:

```
vs2.example.com::> snapmirror initialize -destination-path
vs2.example.com:vol_log_sync_dp -source-path
vs1.example.com:vol_log -policy StrictSync
```

To create and initialize an SnapMirror active sync relationship with the following attributes:

- It is between the source Consistency Group `cg_src` in Vserver `vs1.example.com`, and the destination Consistency Group `cg_dst` in Vserver `vs2.example.com`.
- It has item mappings between volumes `srcvol1` and `srcvol2` and volumes `dstvol1` and `dstvol2`.
- It uses the policy named `AutomatedFailOver` that has a policy type of `automated-failover`.

Type the following command from the destination cluster:

```
destination::> snapmirror initialize -destination-path
vs2.example.com:/cg/cg_dst -source-path
vs1.example.com:/cg/cg_src
-policy AutomatedFailOver
-cg-item-mappings srcvol1:@dstvol1,srcvol2:@dstvol2
```

To initialize the previously created SnapMirror active sync relationship, type the following command from the destination cluster:

```
destination:> snapmirror initialize -destination-path
vs2.example.com:/cg/cg_dst
```

To create and initialize an Application Consistency Group relationship with the following attributes:

- It is between the source Application Consistency Group *app\_cg\_src* in Vserver *vs1.example.com*, and the destination Application Consistency Group *app\_cg\_dst* in Vserver *vs2.example.com*. It has item mappings between volumes *srcvol1* and *dstvol2* and volumes *srcvol2* and *dstvol2*. It uses the default policy named *MirrorAndVault* that has a policy type of *mirror-vault*.

Type the following command from the destination cluster:

```
destination:> snapmirror initialize -destination-path
vs2.example.com:/cg/app_cg_dst -source-path
vs1.example.com:/cg/app_cg_src
-cg-item-mappings srcvol1:@dstvol1,srcvol2:@dstvol2
```

To initialize the previously created Application Consistency Group relationship, type the following command from the destination cluster:

```
destination:> snapmirror initialize -destination-path
vs2.example.com:/cg/app_cg_dst
```

## Related Links

- [snapmirror initialize-ls-set](#)
- [snapmirror create](#)
- [snapmirror show](#)
- [job show](#)
- [job history show](#)
- [snapmirror policy create](#)
- [snapmirror modify](#)

## snapmirror list-destinations

Display a list of destinations for SnapMirror sources

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### Description

The `snapmirror list-destinations` command displays information including the destination endpoints, the relationship status, and transfer progress, for SnapMirror relationships whose source endpoints are in the current Vserver if you are in a Vserver context, or the current cluster if you are in a cluster context. +

The command might display several relationships that have the same source and destination endpoints, but have different relationship IDs. If this is the case, some of the information is stale. It corresponds to relationships that have been deleted on the destination Vserver or cluster, and have not been released yet on the source Vserver or source cluster.

The relationships and the information displayed are controlled by the parameters that you specify. If no parameters are specified, the command displays the following information associated with each SnapMirror relationship whose source endpoint is in the current Vserver if you are in a Vserver context, or the current cluster if you are in a cluster context:

- Source path
- Relationship Type
- Destination Path
- Relationship Status
- Transfer Progress
- Progress Last Updated
- Relationship ID

Note the following limitations on the information displayed by the `snapmirror list-destinations` command:

- The *"Relationship Status"* field is not valid after the node hosting the source volume joins the cluster quorum, until at least one transfer is performed on the SnapMirror relationship.
- *"Transfer Progress"* and *"Progress Last Updated"* fields are only valid if a Snapshot copy transfer is in progress.
- The *"Relationship ID"* field is not valid for Vserver SnapMirror relationships.
- The *"Relationship Status"*, *"Transfer Progress"*, and *"Progress Last Updated"* fields are not valid for FlexGroup relationships, but they are valid for FlexGroup constituent relationships.

The `-instance` and `-fields` parameters are mutually exclusive and select the fields that are displayed. The `-instance` parameter if specified, displays detailed information about the relationships. The `-fields` parameter specifies what fields should be displayed. The other parameters of the `snapmirror list-destinations` command, select the SnapMirror relationships for which the information is displayed.

This command is not supported for SnapMirror relationships with non-Data ONTAP endpoints.

## Parameters

**{ [-fields <fieldname>,...]**

If you specify the `-fields <fieldname>,...` parameter, the command only displays the fields that you have specified.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays detailed information about all relationships selected.

**{ [-S, -source-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} } - Source Path**

Selects SnapMirror relationships that have a matching source path name.

**[`-source-vserver <vserver name>`] - Source Vserver**

Selects SnapMirror relationships that have a matching source Vserver name.

**[`-source-volume <volume name>`] - Source Volume }**

Selects SnapMirror relationships that have a matching source volume name.

**{ [`-destination-path`**

**{<[`vserver:`] [`volume`]>|<[[`cluster:`]//`vserver/`]`volume`>|<`hostip:/lun/name`>|<`hostip:/share/share-name`>|<[`vserver:`]/`cg/`[`app-cgname`]>}} - Destination Path**

Selects SnapMirror relationships that have a matching destination path name.

**[`-destination-vserver <vserver name>`] - Destination Vserver**

Selects SnapMirror relationships that have a matching destination Vserver name.

**[`-destination-volume <volume name>`] - Destination Volume }**

Selects SnapMirror relationships that have a matching destination volume name.

**[`-relationship-id <UUID>`] - Relationship ID**

Selects SnapMirror relationships that have a matching relationship identifier. This parameter is not supported for Vserver SnapMirror relationships.

**[`-type <snapmirrorType>`] - Relationship Type**

Selects SnapMirror relationships that have a matching relationship type. Possible values are:

- DP
- XDP
- RST

**[`-relationship-group-type`**

**{`none`|`vserver`|`consistencygroup`|`flexgroup`|`vserverflexgroup`|`vserverconsistencygroup` } - Relationship Group Type**

Selects SnapMirror relationships that have a matching relationship group type. Possible values are:

- none
- vserver
- flexgroup

**[`-policy-type {vault|async-mirror|mirror-vault|strict-sync-mirror|sync-mirror|automated-failover|automated-failover-duplex|continuous}`] - SnapMirror Policy Type**

Selects SnapMirror relationships that have a matching SnapMirror policy type. Possible values are:

- async-mirror
- vault
- mirror-vault

**[-cg-item-mappings {<source volume>:@<destination volume>|<source item path>:@<destination item path>}] - Consistency Group Item Mappings**

Selects SnapMirror relationships that have a matching list of Consistency Group (CG) item mappings. This parameter is applicable to CG level relationships only.

**[-status <mirror status>] - Relationship Status**

Selects SnapMirror relationships that have a matching relationship status. Possible values are:

- Idle
- Transferring

This parameter is not supported for FlexGroup and Application Consistency Group SnapMirror relationships, but it is supported for FlexGroup and Application Consistency Group constituent relationships.

**[-transfer-progress {<integer>[KB|MB|GB|TB|PB]}] - Transfer Progress**

Selects SnapMirror relationships that have a matching transfer progress. This parameter is not supported for FlexGroup and Application Consistency Group SnapMirror relationships, but it is supported for FlexGroup and Application Consistency Group constituent relationships.

**[-progress-last-updated <MM/DD HH:MM:SS>] - Timestamp of Last Progress Update**

Selects SnapMirror relationships that have a matching transfer progress last updated timestamp. This parameter is not supported for FlexGroup and Application Consistency Group SnapMirror relationships, but it is supported for FlexGroup and Application Consistency Group constituent relationships.

**[-source-volume-node <nodename>] - Source Volume Node Name**

Selects SnapMirror relationships that have a matching source volume node name. For FlexGroup relationships, it is the node which owns the root constituent source volume. This parameter is not supported for Vserver SnapMirror relationships.

**[-expand <true>] - Show Constituents of the Group**

Specifies whether to display constituent relationships of Vserver and FlexGroup SnapMirror relationships. By default, the constituents are not displayed.

## Examples

To display summary information about all relationships whose source endpoints are in the current cluster, type the following command:



```

cluster1::> snapmirror list-destinations
Progress
Source          Destination          Transfer  Last      Relationship
Path           Type  Path              Status    Progress  Updated   ID
-----
-----
vserver1.example.com:dp_s1
                DP    vserver2.example.com:dp_d1
                                Idle     -         -         06b4327b-954f-
11e1-af65-123478563412
vserver1.example.com:xdp_s1
                XDP   vserver2.example.com:xdp_d1
                                Idle     -         -         a9c1db0b-954f-
11e1-af65-123478563412
vserver2.example.com:
                DP    dvserver2.example2.com:
                                Idle     -         -         -
3 entries were displayed.

```

To display summary information about all relationships whose source endpoints are in the current Vserver, type the following command:

```

vserver1.example.com::> snapmirror list-destinations
Progress
Source          Destination          Transfer  Last      Relationship
Path           Type  Path              Status    Progress  Updated   ID
-----
-----
vserver1.example.com:dp_s1
                DP    vserver2.example.com:dp_d1
                                Idle     -         -         06b4327b-954f-
11e1-af65-123478563412
vserver1.example.com:xdp_s1
                XDP   vserver2.example.com:xdp_d1
                                Idle     -         -         a9c1db0b-954f-
11e1-af65-123478563412
2 entries were displayed.

```

To display detailed information about SnapMirror relationships whose source endpoints are in the current Vserver, type the following command:

```
vserver1.example.com::> snapmirror list-destinations -instance
Source Path: vserver1.example.com:dp_s1
    Destination Path: vserver2.example.com:dp_d1
    Relationship Type: DP
Relationship Group Type: none
    Relationship Status: Idle
    Transfer Progress: -
    Progress Last Updated: -
    Source Volume Node: node1
        Relationship ID: 06b4327b-954f-11e1-af65-123478563412
Source Path: vserver1.example.com:xdp_s1
    Destination Path: vserver2.example.com:xdp_d1
    Relationship Type: XDP
Relationship Group Type: none
    Relationship Status: Idle
    Transfer Progress: -
    Progress Last Updated: -
    Source Volume Node: node2
        Relationship ID: a9c1db0b-954f-11e1-af65-123478563412

2 entries were displayed.
```

To display summary information about all relationships including constituent relationships whose source endpoints are in the current Vserver, type the following command:

```
cluster-1::> snapmirror list-destinations -expand
```

Source		Destination		Transfer	Progress	Last	
Relationship							
Path	Type	Path	Status	Progress	Updated	Id	
vs1:fg_s1	XDP	vs1:fg_d1	-	-	-	504abc00-	
70a8-11e6-82be-0050568536d7							
vs1:fg_s1__0001							
	XDP	vs1:fg_d1__0001	-	-	-	5041f2aa-	
70a8-11e6-82be-0050568536d7							
vs1:fg_s1__0002							
	XDP	vs1:fg_d1__0002	-	-	-	50421733-	
70a8-11e6-82be-0050568536d7							
vs1:fg_s1__0003							
	XDP	vs1:fg_d1__0003	-	-	-	50421826-	
70a8-11e6-82be-0050568536d7							
vs1:fg_s1__0004							
	XDP	vs1:fg_d1__0004	-	-	-	504218f0-	
70a8-11e6-82be-0050568536d7							

5 entries were displayed.

To display information about Application Consistency Group SnapMirror relationships type the following command:

```
cluster-1::> snapmirror list-destinations
```

Source		Destination		Transfer	Progress	Last	
Relationship							
Path	Type	Path	Status	Progress	Updated	Id	
vs0:/cg/cgA	XDP	vs1:/cg/cg_dst_async	Idle	-	-	8cb8bc1a-	
c193-11ed-9f4c-005056a7c7a2							

## snapmirror modify

Modify a SnapMirror relationship

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror modify` command allows you to change one or more properties of SnapMirror relationships. The key parameter that identifies any SnapMirror relationship is the destination endpoint. The destination endpoint can be a Vserver, a volume, an Application Consistency Group or a non-Data ONTAP endpoint.

For load-sharing mirrors, a change to a property affects all of the SnapMirror relationships in the set of load-sharing mirrors. Destination volumes in a set of load-sharing mirrors do not have individual property settings.

Changes made by the `snapmirror modify` command do not take effect until the next manual or scheduled update of the SnapMirror relationship. Changes do not affect updates that have started and have not finished yet.

This command is supported for SnapMirror relationships with the field *"Relationship Capability"* showing as either *"8.2 and above"* or *"Pre 8.2"* in the output of the `snapmirror show` command.

The `snapmirror modify` command must be used from the destination Vserver or cluster.

## Parameters

**{ [-S, -source-path <[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}] - Source Path**

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (*vserver*) and/or the volume (*volume*). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (*cluster*) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form *hostip:/share/share-name*. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form *hostip:/lun/name*.

**| [-source-cluster <Cluster name>] - Source Cluster**

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source-vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

**[-source-vserver <vserver name>] - Source Vserver**

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

**[-source-volume <volume name>] - Source Volume }**

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP

source endpoints.

```
{ -destination-path  
{<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/  
share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path
```

This parameter specifies the destination endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or volume (volume). To support relationships with "Relationship Capability" of "Pre 8.2", a format which also includes the name of the cluster (cluster) is provided. The "Pre 8.2" format cannot be used when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above". For SnapMirror relationships with AltaVault destinations, the destination endpoint is specified in the form `hostip:/share/share-name`. For relationships with SolidFire destinations, the destination endpoint is specified in the form `hostip:/lun/name`.

```
[ -destination-cluster <Cluster name>] - Destination Cluster
```

Specifies the destination cluster of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and `-destination-volume` must also be specified. This parameter is only applicable for relationships with "Relationship Capability" of "Pre 8.2". This parameter cannot be specified when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above".

```
-destination-vserver <vserver name> - Destination Vserver
```

Specifies the destination Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-destination-volume` and for relationships with "Relationship Capability" of "Pre 8.2", `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

```
-destination-volume <volume name> - Destination Volume }
```

Specifies the destination volume of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and for relationships with "Relationship Capability" of "Pre 8.2", `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

```
[ -vserver <vserver name>] - Managing Vserver
```

If this optional parameter is specified, designates the managing Vserver. The managing Vserver is authorized to use some snapmirror commands to manage the SnapMirror relationship. The `-vserver` option is currently a reserved option.

```
[ -schedule <text>] - SnapMirror Schedule
```

This optional parameter designates the name of the schedule which is used to update the SnapMirror relationship. If you do not designate a schedule, updates are not scheduled, so you must update the SnapMirror relationship manually using the [snapmirror update](#) command or, in the case of a set of load-sharing mirrors, using the [snapmirror update-ls-set](#) command.



You define and name a schedule using the [job schedule cron create](#) command.

```
[ -policy <sm_policy>] - SnapMirror Policy
```

This optional parameter designates the name of the snapmirror policy which is associated with the SnapMirror relationship. If you do not designate a policy, the current policy will be retained. Modification of the policy is not supported for relationships with a policy of type `strict-sync-mirror`, `sync-mirror`, or `automated-failover-duplex`. If you want to use a different policy for SnapMirror Synchronous

relationships, you need to delete the relationship and create it again with the new policy. This parameter is not applicable to relationships with *"Relationship Capability"* of *"Pre 8.2"*.



You define and name a policy using the [snapmirror policy create](#) command.

#### **[`-tries <integer_or_unlimited>`] - Tries Limit**

This optional parameter specifies the maximum number of times to attempt each manual or scheduled transfer for a SnapMirror relationship. The default is eight times. The `-tries` parameter can be set to `0` to disable manual and scheduled updates for the SnapMirror relationship. This parameter is only applicable to relationships with *"Relationship Capability"* of *"Pre 8.2"*. For relationships with *"8.2 and above"* capability, the tries limit is controlled by the value of `tries` in the SnapMirror policy that is associated with the relationship.

#### **[`-k, -throttle <throttleType>`] - Throttle (KB/sec)**

This optional parameter limits the network bandwidth used for transfers. It configures for the relationship the maximum rate (in Kbytes/sec) at which data can be transferred. If no throttle is configured, by default the SnapMirror relationship fully utilizes the network bandwidth available. You can also configure the relationship to fully use the network bandwidth available by explicitly setting the throttle to *unlimited* or `0`. The minimum effective throttle value is four Kbytes/sec, so if you specify a throttle value between `1` and `4`, it will be treated as `4`. For FlexGroup volume relationships, the throttle value is applied individually to each constituent relationship. The `-throttle` parameter does not affect load-sharing mirrors and other SnapMirror relationships with *"Relationship Capability"* of *"Pre 8.2"* confined to a single cluster.

#### **[`-is-auto-expand-enabled {true|false}`] - Is Auto Expand Enabled**

This optional parameter specifies whether or not a FlexGroup SnapMirror relationship and its destination FlexGroup should be auto-expanded if the source FlexGroup is expanded. This parameter is supported only for FlexGroup SnapMirror relationships. If you do not specify the parameter, the current value of `auto expand` will be retained. Application Consistency Group destinations are always automatically expanded when the source Application Consistency Group is expanded.

#### **[`-backoff-level {high|medium|none}`] - SM Backoff Level due to Client Ops**

This optional parameter specifies the SnapMirror backoff level due to client ops. This parameter is supported only for FlexVol SnapMirror relationships. The default value is *high*.

#### **[`-w, -foreground <true>`] - Foreground Process**

This specifies whether the operation runs as a foreground process. If this parameter is specified, the default setting is *true* (the operation runs in the foreground). When set to *true*, the command will not return until the process completes. This parameter is only applicable to relationships with *"Relationship Capability"* of *"Pre 8.2"*.

## Examples

To change the schedule to *halfhour* for the SnapMirror relationship with the destination endpoint `vs2.example.com:dept_eng_dp_mirror2`, type the following command:

```
vs2.example.com::> snapmirror modify -destination-path
vs2.example.com:dept_eng_dp_mirror2 -schedule halfhour
```

For relationships with *"Relationship Capability"* of *"Pre 8.2"*, to change the schedule to *halfhour*

for the SnapMirror relationship with the destination endpoint

`cluster2://vs2.example.com/dept_eng_dp_mirror2`, type the following command:

```
cluster2::> snapmirror modify -destination-path
cluster2://vs2.example.com/dept_eng_dp_mirror2
-schedule halfhour
```

To change the schedule to `halfhour` for the Vserver SnapMirror relationship with destination endpoint `dvs1.example.com:`, type the following command:

```
cluster2::> snapmirror modify -destination-path
dvs1.example.com: -schedule halfhour
```

To change the policy associated with the synchronous SnapMirror Consistency Group relationship with the destination Consistency Group `cg_dst` in Vserver `vs2.example.com` to the policy `Sync2`, type the following command:

```
vs2.example.com::> snapmirror modify -destination-path
vs2.example.com:/cg/cg_dst -policy Sync2
```

## Related Links

- [snapmirror show](#)
- [snapmirror update](#)
- [snapmirror update-ls-set](#)
- [job schedule cron create](#)
- [snapmirror policy create](#)

## snapmirror promote

Promote the destination to read-write

**Availability:** This command is available to `cluster` and `Vserver` administrators at the `admin` privilege level.

### Description

The `snapmirror promote` command performs a failover to the destination volume of a SnapMirror relationship. This command changes the destination volume from a read-only volume to a read-write volume and makes the destination volume assume the identity of the source volume. The command then destroys the original source volume. The destination volume must be a load-sharing volume. Note that you can promote a load-sharing volume that has been left in read-write state by a previously failed promote operation.

Client accesses are redirected from the original source volume to the promoted destination volume. The view clients see on the promoted destination volume is the latest transferred Snapshot copy, which might lag behind the view clients had of the original source volume before the promote.

The SnapMirror relationship is always deleted as part of the promotion process.

It is possible that the original source volume is the source of multiple SnapMirror relationships. For such a configuration, the promoted destination volume becomes the new source volume of the other SnapMirror relationships.

This command is only supported for SnapMirror relationships with the field *"Relationship Capability"* showing as *"Pre 8.2"* in the output of the [snapmirror show](#) command.

The `snapmirror promote` command fails if a SnapMirror transfer is in progress for any SnapMirror relationship with *"Relationship Capability"* of *"Pre 8.2"* involving the original source volume. It does not fail if a SnapMirror transfer is in progress for a relationship with *"Relationship Capability"* of *"8.2 and above"*.

## Parameters

**{ [-S, -source-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Source Path**

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or the volume (volume). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (cluster) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form *hostip:/share/share-name*. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form *hostip:/lun/name*.

**| [-source-cluster <Cluster name>] - Source Cluster**

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source-vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

**[-source-vserver <vserver name>] - Source Vserver**

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

**[-source-volume <volume name>] - Source Volume }**

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

**{ -destination-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path**

This parameter specifies the destination endpoint of the SnapMirror relationship in one of four path formats.



The normal format includes the names of the Vserver (vserver) and/or volume (volume). To support relationships with "Relationship Capability" of "Pre 8.2", a format which also includes the name of the cluster (cluster) is provided. The "Pre 8.2" format cannot be used when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above". For SnapMirror relationships with AltaVault destinations, the destination endpoint is specified in the form `hostip:/share/share-name`. For relationships with SolidFire destinations, the destination endpoint is specified in the form `hostip:/lun/name`.

#### **[`-destination-cluster <Cluster name>`] - Destination Cluster**

Specifies the destination cluster of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and `-destination-volume` must also be specified. This parameter is only applicable for relationships with "Relationship Capability" of "Pre 8.2". This parameter cannot be specified when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above".

#### **`-destination-vserver <vserver name>` - Destination Vserver**

Specifies the destination Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-destination-volume` and for relationships with "Relationship Capability" of "Pre 8.2", `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

#### **`-destination-volume <volume name>` - Destination Volume }**

Specifies the destination volume of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and for relationships with "Relationship Capability" of "Pre 8.2", `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

#### **[`-f, -force <true>`] - Force**

If this parameter is specified, the command proceeds without prompting for confirmation.

## Examples

To promote a mirror named `vs1_root_ls1` to be the source read-write volume for mirroring and client access, type the following command:

```
cluster1::> snapmirror promote -destination-path
//vs1.example.com/vs1_root_ls1
-source-path //vs1.example.com/vs1_root -f true
```

## Related Links

- [snapmirror show](#)

## snapmirror protect

Start protection for Vservers and volumes

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror protect` command establishes SnapMirror protection for the specified Vserver or a list of volumes. For each endpoint, the command creates a data protection destination endpoint in the Vserver specified by the `-destination-vservers` parameter, creates an extended data protection (XDP) SnapMirror relationship, and starts the initialization of the SnapMirror relationship. This command must be used from the destination Vserver or cluster. This command is not supported for FlexGroup volume constituents or non ONTAP endpoints. When source volumes are specified they must be read-write (RW) volumes.

## Parameters

### **[`-source-cluster` <Cluster name>] - Source Cluster**

This optional parameter specifies the source cluster name. This parameter is valid only when only a single Vserver is specified in the path-list parameter.

### **`-path-list`**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Path List**

This parameter specifies the list of source endpoints to be protected. The list is a comma separated list of paths of the form `vserver:volume` or `vserver:`, for example `vs1.example.com:dept_eng1`, `vs1.example.com:dept_eng2` or `vs1.example.com:`. If the list contains a Vserver endpoint, then the Vserver is the only endpoint that can be specified, and the list cannot contain a mixture of volume and Vserver endpoints.

### **[`-destination-vservers` <vserver name>] - Destination Vserver**

This parameter specifies the Vserver in which to create the destination volumes of the SnapMirror relationships. When protecting a single Vserver, this parameter specifies the destination Vserver endpoint for protection.

### **[`-destination-vservers-ip-space` <IPspace>] - Destination Vserver IPspace Name**

This optional parameter specifies the IPspace the Vserver will be assigned to. If left unspecified, the Vserver will be assigned to the default IPspace. This parameter is supported while protecting a Vserver.

### **[`-schedule` <text>] - SnapMirror Schedule**

This optional parameter designates the name of the schedule which is used to update the SnapMirror relationships.

### **`-policy` <sm\_policy> - SnapMirror Policy**

This parameter designates the name of the SnapMirror policy which is associated with the SnapMirror relationships.

### **[`-auto-initialize` {true|false}] - Auto Initialize**

This optional parameter specifies whether or not initializes of the SnapMirror relationships should be started after the relationships are created. The default value for this parameter is `true`.

### **[`-destination-volume-prefix` <text>] - Destination Volume Name Prefix**

This optional parameter designates the prefix for the destination volume name. For example if the source path is of the form `vserver:volume` and the destination-volume-prefix specified is `prefix` and no destination-volume-suffix is specified, then the destination volume name will be `_prefix_volume_dst` or possibly `prefix_volume_1_dst` if a name conflict is encountered. If both prefix and suffix are specified as `prefix` and `__suffix`,

then the destination volume name will be *prefix\_volume\_suffix* or *prefix\_volume\_1\_suffix*, if a name conflict is encountered. This parameter is not supported for Vserver endpoints.

#### **[`-destination-volume-suffix <text>`] - Destination Volume Name Suffix**

This optional parameter designates the suffix for the destination volume name. If you do not designate a suffix, a destination volume name with suffix *dst\_* will be used. For example if the source path is of the form *vserver:volume*, and the suffix specified is *DP\_*, the destination volume will be created with the name *volume\_DP* or *volume\_1\_DP* if a name conflict is encountered. If both prefix and suffix are specified as *prefix* and *suffix*, then the destination volume name will be *prefix\_volume\_suffix* or *prefix\_volume\_1\_suffix*, if a name conflict is encountered. This parameter is not supported for Vserver endpoints.

#### **[`-support-tiering {true|false}`] - Provision Destination Volumes on FabricPools**

This optional parameter specifies whether or not FabricPools are selected when provisioning a FlexVol volume or a FlexGroup volume during protection workflows. When this parameter is set to *true*, only FabricPools are used; when set to *false*, only non-FabricPools are used. Tiering support for a FlexVol volume can be changed by moving the volume to the required aggregate. Tiering support for a FlexGroup volume can be changed by moving all of the constituents to the required aggregates. The default value is *false*. This parameter is supported only for FlexVol volumes and FlexGroup volumes.

#### **[`-tiering-policy <Tiering Policy>`] - Destination Volume Tiering Policy**

This optional parameter specifies the tiering policy to apply to the destination FlexVol volume or FlexGroup volume. This policy determines whether the user data blocks of a FlexVol volume or FlexGroup volume in a FabricPool will be tiered to the capacity tier when they become cold. FabricPool combines flash (performance tier) with an object store (external capacity tier) into a single aggregate. The default tiering policy is 'snapshot-only' for a FlexVol volume and 'none' for a FlexGroup volume. The temperature of a FlexVol volume or FlexGroup volume block increases if it is accessed frequently and decreases when it is not.

The available tiering policies are:

- *snapshot-only* - This policy allows tiering of only the FlexVol volume or FlexGroup volume Snapshot copies not associated with the active file system. The default minimum cooling period is 2 days. The `-tiering-minimum-cooling-days` parameter can be used to override the default using the `volume modify` command after the destination FlexVol volume or FlexGroup volume has been created.
- *auto* - This policy allows tiering of both snapshot and active file system user data to the capacity tier. The default cooling period is 31 days. The `-tiering-minimum-cooling-days` parameter can be used to override the default using the `volume modify` command after the destination FlexVol volume or FlexGroup volume has been created.
- *none* - FlexVol volume or FlexGroup volume blocks will not be tiered to the capacity tier.
- *backup* - On a DP FlexVol volume or FlexGroup volume this policy allows all transferred user data blocks to start in the capacity tier.

This parameter is supported only for FlexVol volumes and FlexGroup volumes.

## **Examples**

To establish SnapMirror protection for the source volumes *vs1.example.com:dept\_eng1* and *vs1.example.com:dept\_eng2* using destination-vserver *vs2.example.com* and policy *MirrorAllSnapshots* type the following command:

```
vs2.example.com::> snapmirror protect -path-list
    vs1.example.com:dept_eng1,vs1.example.com:dept_eng2 -destination
-vserver
    vs2.example.com -policy MirrorAllSnapshots
```

To establish SnapMirror protection for the source Vserver *vs1.example.com* which is on cluster *cluster1* creating a destination-vserver named *vs1dp.example.com* and using policy *MirrorAllSnapshots* type the following command:

```
cluster2::> snapmirror protect -source-cluster cluster1 -path-list
vs1.example.com: -destination-vserver vs1dp.example.com -policy
MirrorAllSnapshots
```

## snapmirror quiesce

### Disable future transfers

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### Description

The `snapmirror quiesce` command disables future transfers for a SnapMirror relationship. If there is no transfer in progress, the relationship becomes *"Quiesced"*.

If there is a transfer in progress, it is not affected, and the relationship becomes *"Quiescing"* until the transfer completes. If the current transfer aborts, it will be treated like a future transfer and will not restart.

If applied to a load-sharing (LS) SnapMirror relationship, all the relationships in the load-sharing set will be quiesced.

If applied to a relationship with a policy of type *strict-sync-mirror* or *sync-mirror*, any ongoing operation is aborted, and if the relationship is InSync, synchronous replication is stopped. If the replication policy type is "strict-sync-mirror", then the primary client I/O disruption is not enforced. A new common Snapshot copy is created if the relationship is InSync, unless a recent one exists. The relationship becomes *"Quiescing"* until these operations complete.

When a SnapMirror relationship is quiesced, it remains quiesced across reboots and fail-overs.

This command is supported for SnapMirror relationships with the field *"Relationship Capability"* showing as either *"8.2 and above"* or *"Pre 8.2"* in the output of the `snapmirror show` command.

This command is not supported for SnapMirror active sync relationships with policy of type *automated-failover* or *automated-failover-duplex*.

The `snapmirror quiesce` command must be used from the destination Vserver or cluster.

The relationship must exist on the destination Vserver or cluster. When issuing `snapmirror quiesce`, you must specify the destination endpoint. The specification of the source endpoint of the relationship is optional.

## Parameters

### { [-S, -source-path

{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Source Path

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or the volume (volume). To support relationships with "Relationship Capability" of "Pre 8.2", a format which also includes the name of the cluster (cluster) is provided. The "Pre 8.2" format cannot be used when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above". For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form *hostip:/share/share-name*. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form *hostip:/lun/name*.

### | [-source-cluster <Cluster name>] - Source Cluster

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source`, `-vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with "Relationship Capability" of "Pre 8.2". This parameter cannot be specified when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above".

### [-source-vserver <vserver name>] - Source Vserver

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with "Relationship Capability" of "Pre 8.2", `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

### [-source-volume <volume name>] - Source Volume }

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with "Relationship Capability" of "Pre 8.2", `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

### { -destination-path

{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path

This parameter specifies the destination endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or volume (volume). To support relationships with "Relationship Capability" of "Pre 8.2", a format which also includes the name of the cluster (cluster) is provided. The "Pre 8.2" format cannot be used when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above". For SnapMirror relationships with AltaVault destinations, the destination endpoint is specified in the form *hostip:/share/share-name*. For relationships with SolidFire destinations, the destination endpoint is specified in the form *hostip:/lun/name*.

### | [-destination-cluster <Cluster name>] - Destination Cluster

Specifies the destination cluster of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and `-destination-volume` must also be specified. This parameter is only applicable for relationships with "Relationship Capability" of "Pre 8.2". This parameter cannot be specified when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above".

### **-destination-vserver <vserver name> - Destination Vserver**

Specifies the destination Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-destination-volume` and for relationships with *"Relationship Capability" of "Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

### **-destination-volume <volume name> - Destination Volume }**

Specifies the destination volume of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and for relationships with *"Relationship Capability" of "Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

## Examples

To quiesce the SnapMirror relationship with the destination endpoint `vs2.example.com:dept_eng_mirror2`, type the following command:

```
vs2.example.com::> snapmirror quiesce -destination-path
vs2.example.com:dept_eng_mirror2
```

For relationships with *"Relationship Capability" of "Pre 8.2"*, to quiesce the SnapMirror relationship with the destination endpoint `cluster2://vs2.example.com/dept_eng_mirror2`, type the following command:

```
cluster2::> snapmirror quiesce -destination-path
cluster2://vs2.example.com/dept_eng_mirror2
```

To quiesce the Vserver SnapMirror relationship with the destination endpoint `dvs1.example.com:`, type the following command:

```
cluster2::> snapmirror quiesce -destination-path
dvs1.example.com:
```

To quiesce the Application Consistency Group SnapMirror relationship with the destination Application Consistency Group `app_cg_dst` in Vserver `vs2.example.com`, type the following command:

```
vs2.example.com::> snapmirror quiesce -destination-path
vs2.example.com:/cg/app_cg_dst
```

## Related Links

- [snapmirror show](#)

# snapmirror release

Remove source information for a SnapMirror relationship

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror release` command removes the relationship information from the source Vserver. The command also removes any Snapshot copy owner tags and any Snapshot copies which were created for the specified relationship from the source volumes. It does not destroy any volumes or Vservers. This command must be used from the source Vserver or cluster.

For relationships with a policy of type *strict-sync-mirror*, it additionally resumes primary volume access if the IO was restricted because the relationship was *OutOfSync*.

For relationship having Application Consistency Groups, `snapmirror release` command used on the source of a Application Consistency Group relationship, removes the relationship information from the source side and deletes Snapshot copies which were created on the source volumes.

You can use the [snapmirror list-destinations](#) command to display source Vservers' relationship information. This information is populated during the first SnapMirror transfer, not when the [snapmirror create](#) command is issued.

This command is not supported for SnapMirror relationships with the field *"Relationship Capability"* showing as *"Pre 8.2"* in the output of the [snapmirror show](#) command.

This command is not supported for SnapMirror relationships with non-Data ONTAP endpoints.

The `snapmirror release` operation fails if a SnapMirror transfer for the SnapMirror relationship is in a data phase of the transfer.

## Parameters

**{ [-S, -source-path  
<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}] - Source Path**

Specifies the source endpoint of the SnapMirror relationship in one of three formats. The normal format includes the names of the Vserver (*vserver*), and/or volume (*volume*). A format which also includes the name of the cluster (*cluster*) is also provided for consistency with other `snapmirror` commands. The form of the pathname which includes the cluster name cannot be used when operating in a Vserver context. For relationships between Application Consistency Groups, the source endpoint is specified in the form *[vserver:]/cg/cg-name*.

**| [-source-vserver <vserver name>] - Source Vserver**

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameter `-source-volume` must also be specified.

**[-source-volume <volume name>] - Source Volume }**

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameter `-source-vserver` must also be specified.

### **{ -destination-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path**

Specifies the destination endpoint of the SnapMirror relationship in one of three formats. The normal format includes the names of the Vserver (vserver), and/or volume (volume). A format which also includes the name of the cluster (cluster) is also provided for consistency with other snapmirror commands. The form of the pathname which includes the cluster name cannot be used when operating in a Vserver context. For relationships between Application Consistency Groups the destination endpoint is specified in the form

*[vserver:]/cg/cg-name*

### **[-destination-vserver <vserver name> - Destination Vserver**

Specifies the destination Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameter `-destination-volume` must also be specified.

### **[-destination-volume <volume name>] - Destination Volume }**

Specifies the destination volume of the SnapMirror relationship. If this parameter is specified, parameter `-destination-vserver` must also be specified.

### **[-relationship-info-only <true>] - Remove relationship info only (skip cleanup of snapshots)**

If this parameter is specified, the cleanup of Snapshot copies is bypassed and only the source relationship information is removed. It is recommended to specify this parameter only when the source volume is not accessible.

### **[-relationship-id <UUID>] - Relationship ID**

This optional parameter specifies the relationship identifier of the relationship. It must be specified when information for more than one relationship with the same source and destination paths is present. This parameter is not supported for Vserver SnapMirror relationships.

### **[-f, -force <true>] - Force**

If this parameter is specified, the command proceeds without prompting for confirmation.

## **Examples**

To release the source information for the SnapMirror relationship with the destination endpoint *vs2.example.com:dept\_eng\_dp\_mirror4*, type the following command:

```
vs1.example.com::> snapmirror release
    -destination-path vs2.example.com:dept_eng_dp_mirror4
```

To release the source information for the SnapMirror relationship with the destination endpoint *vs2.example.com:dept\_eng\_dp\_mirror4*, and relationship-id *5f91a075-6a72-11e1-b562-123478563412*, type the following command:

```
vs1.example.com::> snapmirror release
    -destination-path vs2.example.com:dept_eng_dp_mirror4
    -relationship-id 5f91a075-6a72-11e1-b562-123478563412
```



To release the source information for the SnapMirror relationship with the destination endpoint `dvs1.example.com`, type the following command:

```
cluster1::> snapmirror release
      -destination-path dvs1.example.com:
```

To release the Application Consistency Group SnapMirror relationship with the destination Application Consistency Group `cg_dst` in Vserver `vs2.example.com` and the source Application Consistency Group `cg_src` in Vserver `vs1.example.com`, type the following command from the source cluster:

```
source::> snapmirror release
      vs2.example.com:/cg/cg_dst
```

To release just the source information but not remove the Snapshot copies that might be needed for a subsequent resync for the Asynchronous SnapMirror Application Consistency Group relationship with the destination Application Consistency Group `cg_dst` in Vserver `vs2.example.com`, type the following command:

```
vs2.example.com::> snapmirror release
      -destination-path vs2.example.com:/cg/cg_dst
      -relationship-info-only true
```

To release the SnapMirror active sync relationship with the destination Consistency Group `cg_dst` in Vserver `vs2.example.com` and the source Consistency Group `cg_src` in Vserver `vs1.example.com`, type the following command from the source cluster:

```
source::> snapmirror release
      vs2.example.com:/cg/cg_dst
```

## Related Links

- [snapmirror list-destinations](#)
- [snapmirror create](#)
- [snapmirror show](#)

## snapmirror restore

Restore a Snapshot copy from a source volume to a destination volume

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror restore` command restores the entire contents of a Snapshot copy or one or more files, LUNs or NVMe namespaces of a Snapshot copy from one volume to another volume.

The `snapmirror restore` command, in context to Application Consistency Groups, restores the entire content of a Snapshot copy from one Application Consistency Group to another Application Consistency Group. Each Application Consistency Group is considered as atomic entity in restore operation

The source of the restore can be a volume or an Application Consistency Group that is:

- the destination volume or destination Application Consistency Group of a extended data protection (XDP) relationship
- the destination volume of a data protection (DP) relationship with *"Relationship Capability"* of *"8.2 and above"*
- a data-protection volume or an Application Consistency Group which is not the destination endpoint of any SnapMirror relationship
- a read-write volume or an Application Consistency Group.
- an AltaVault endpoint. In this case the destination must be an empty Data ONTAP volume.

The following cannot be used as either the source or destination volume or destination Application Consistency Group of a restore:

- a volume or Application Consistency Group that is the source or destination endpoint of a SnapMirror load-sharing relationship.
- a volume that is the destination endpoint of a SnapMirror relationship with the *"Relationship Capability"* of *"Pre 8.2"*.
- a SolidFire endpoint.

A SnapMirror relationship of type *RST* is created from the source volume or source Application Consistency Group to the destination volume or destination Application Consistency Group by the `snapmirror restore` command. This relationship lasts for the duration of the restore operation and is deleted when the command completes successfully.

The following paragraphs describe the behavior when restoring the entire contents of a Snapshot copy to a destination volume or destination Application Consistency Group.

By default the `snapmirror restore` will copy the latest Snapshot copy from the source volume or source Application Consistency Group to the destination volume or destination Application Consistency Group. A specific Snapshot copy can be selected with the `-source-snapshot` parameter.

Any quota rules defined for the destination volume are deactivated prior to restoring the entire contents of a Snapshot copy. Run the command `volume quota modify -vserver`destination-volume-vserver -volume`destination-volume-name-state`on` to reactivate quota rules after the entire contents of the Snapshot copy have been restored.

If the destination Application Consistency Group contains empty data protection volumes, the `snapmirror restore` command performs a baseline restore.

If the destination volume is an empty data protection volume, the `snapmirror restore` command performs a baseline restore. For a baseline restore the following steps are performed:

- Create the *RST* SnapMirror relationship.
- The entire contents of the Snapshot copy selected to be restored are copied to the active file system of the destination volume or destination Application Consistency Group.
- The destination volume or destination Application Consistency Group is made read-write.
- The *RST* SnapMirror relationship is deleted.

If the destination volume is a read-write volume, an incremental restore is performed. The incremental restore fails if it cannot find a common Snapshot copy between the source and destination volumes. Restoring a Snapshot copy to an empty read-write volume is not supported. Incremental restore from a non-Data ONTAP endpoint to a Data ONTAP volume is not supported.

If the destination Application Consistency Group contains a read-write flex volume, an incremental restore is performed. The incremental restore fails if it cannot find a common Snapshot copy between the source and destination Application Consistency Group. Restoring a Snapshot copy to an empty read-write flex volumes which are the part of Application Consistency Group is not supported.

An incremental restore preserves all Snapshot copies on the destination volume but does not preserve changes to the active file system since the latest Snapshot copy. To preserve changes to the destination volume since the latest Snapshot copy use the [volume snapshot create](#) command. Restore is a disruptive operation so client access of the destination volume is not advised for the duration of the operation.

For an incremental restore the following steps are performed:

- Create the *RST* SnapMirror relationship.
- The active file system of the destination volume is reverted to the latest Snapshot copy on the destination volume and the destination volume is made read-only.
- In context to Application Consistency Group, the active file system of the destination Application Consistency Group is reverted to the latest Snapshot copy on the destination Application Consistency Group and the flex volumes inside destination Application Consistency Group is made read-only.
- This Snapshot copy is the exported Snapshot copy and it is the view to which clients are redirected when accessing the destination volume or destination Application Consistency Group.
- The contents of the Snapshot copy selected to be restored are copied to the active file system of the destination volume or destination Application Consistency Group.
- The destination volume or destination Application Consistency Group is made read-write.
- The *RST* SnapMirror relationship is deleted.

If `snapmirror restore` fails or is aborted, the *RST* relationship remains. Use the [snapmirror show](#) command with the destination volume name or destination Application Consistency Group name to display the reason for the error. An EMS is also generated when a failure occurs. There are two options to recover when restore fails or is aborted:

- Take corrective action suggested by the EMS and reissue the original command.
- Use the original command with `-clean-up-failure` to cancel the request.

When specifying `-clean-up-failure` to cancel an incremental restore request, the following steps are performed:

- If the Snapshot copy has not been restored to the destination volume or destination Application Consistency Group, all data copied to the active file system by `snapmirror restore` to the destination

volume or destination Application Consistency Group is reverted.

- The destination volume or destination Application Consistency Group is made read-write.
- The *RST* SnapMirror relationship is deleted.

When specifying `-clean-up-failure` to cancel a baseline restore request, the following steps are performed:

- If the Snapshot copy has been restored to the destination volume or destination Application Consistency Group, the volume or Application Consistency Group is made read-write.
- The *RST* SnapMirror relationship is deleted.

The following paragraphs describe the behavior and requirements when restoring one or more files, LUNs or NVMe namespaces to the destination volume.

The destination volume must be a read-write volume. Restoring files, LUNs or NVMe namespaces to a data protection volume is not supported. When restoring files, LUNs or NVMe namespaces the source and destination volumes are not required to have a common Snapshot copy. If a common Snapshot copy exists, an incremental restore is performed for those files, LUNs or NVMe namespaces being restored which exist in the common Snapshot copy.

The destination Application Consistency Group must contain a read-write volume. While restoring, source and destination Application Consistency Group are not required to have a common Snapshot copy. If a common Snapshot copy exists, an incremental restore is performed.

The contents of the files, LUNs or NVMe namespaces to which data is being restored on the destination volume are not preserved by this command. To preserve the contents of the destination files, LUNs or NVMe namespaces, create a Snapshot copy on the destination volume prior to running this command. Client I/O is not allowed to a file, LUN or NVMe namespace to which data is being restored on the destination volume.

The `-source-snapshot` parameter is required when restoring files, LUNs or NVMe namespaces. It identifies the Snapshot copy on the source volume from which the files, LUNs or NVMe namespaces to be restored are copied. If all files, LUNs or NVMe namespaces to be restored do not exist in this Snapshot copy the command fails.

The source path for each file, LUN or NVMe namespace being restored is required. The source path of a file, LUN or NVMe namespace is from the root of the source Snapshot copy of the source volume. The file is restored to the same path on the destination volume unless an optional destination path is specified. The destination path is from the root of the destination volume. If a file, LUN or NVMe namespace to which data is being restored on the destination volume does not exist, the file, LUN or NVMe namespace is created. If any directory in the path of the file, LUN or NVMe namespace being restored does not exist on the destination volume, the command fails. Overwriting the contents of an existing file with the contents of a different file is supported. Similarly, overwriting the contents of an existing LUN or NVMe namespace with the contents of a different LUN or NVMe namespace is supported. However, overwriting a file with the contents of a LUN or NVMe namespace is not supported. Overwriting a LUN with the contents of a file or NVMe namespace is not supported. Overwriting an NVMe namespace with the contents of a file or LUN is not supported. Client I/O is not allowed to all files, LUNs and NVMe namespaces to which data is being restored on the destination volume.

If quota rules have been defined for the destination volume, resource usage is updated during file restore, but limits of quota rules are not enforced. Therefore, resource limits might be exceeded during a file restore.

Multiple concurrent `snapmirror restore` commands, restoring one or more files, LUNs or NVMe namespaces to the same destination volume, are not supported. The destination volume of a `snapmirror`

`restore` to which one or more files, LUNs or NVMe namespaces are being restored, can simultaneously be the source volume of a [snapmirror update](#) .

For a file, LUN or NVMe namespace restore the following steps are performed:

- Create the *RST* SnapMirror relationship.
- If any file, LUN or NVMe namespace being restored does not exist on the destination volume, create all such files, LUNs or NVMe namespaces.
- Prevent client I/O to files, LUNs or NVMe namespaces to which data is being restored on the destination volume.
- Revoke locks and space reservations held by NAS clients for files being restored.
- Copy the contents of all source files, LUNs or NVMe namespaces to the corresponding file, LUN or NVMe namespace on the destination volume.
- Allow client I/O to files, LUNs or NVMe namespaces to which data has been restored on the destination volume.
- Delete the *RST* SnapMirror relationship.



Some file restore operations require a Snapshot copy to be created. This Snapshot copy is temporary, it is deleted before the operation completes.

Since client I/O is not allowed to files, LUNs or NVMe namespaces being restored, client I/O to files, LUNs or NVMe namespaces being restored should be quiesced. Mapped LUNs or NVMe namespaces remain mapped throughout the operation. SAN clients do not need to rediscover a mapped LUN that has been restored. Restoring an NVMe namespace on top of another NVMe namespace with a different attribute relevant to NVMe protocol accessibility (like size) is not supported.

If `snapmirror restore` fails or is aborted, the *RST* relationship remains. Use the [snapmirror show](#) command with the destination volume or destination Application Consistency Group to display the reason for the error. An EMS is also generated when a failure occurs. There are two options to recover when restore fails or is aborted:

- Take corrective action suggested by the EMS and reissue the original command.
- Use `snapmirror restore`-clean-up-failure`` along with specifying the destination volume or destination Application Consistency Group to cancel the request.

When specifying `-clean-up-failure` to cancel a file restore request, the following steps are performed:

- Any files to which client I/O is not allowed are removed.
- Any Snapshot copy created for use during a file restore operation is deleted.
- The *RST* SnapMirror relationship is deleted.



LUNs to which client I/O is not allowed remain. For LUNs to which client I/O is not allowed, do one of the following:

- Use the `snapmirror restore` command to restore data to the LUN. Once the command completes successfully, client I/O to the LUN is allowed.
- Delete the LUN using the [lun delete](#) command with the `-force-fenced` parameter.



Similarly, NVMe namespaces to which client I/O is not allowed remain. For NVMe namespaces to which client I/O is not allowed, do one of the following:

- Use the `snapmirror restore` command to restore data to the NVMe namespace. Once the command completes successfully, client I/O to the NVMe namespace is allowed.
- Delete the NVMe namespace using the `vserver nvme namespace delete` command with the `-skip -mapped-check` parameter.

The `snapmirror restore` command must be used from the destination Vserver or cluster.

## Parameters

**{ [-S, -source-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Source Path**

Specifies the source endpoint in one of four formats. The basic format includes the names of the Vserver (vserver) and volume (volume). A format which also includes the name of the cluster (cluster) is supported for consistency with other `snapmirror` commands. The form of the pathname which includes the cluster name is not valid when operating in a Vserver context. Source endpoint having Application Consistency Group includes vserver name followed by `cg` and Application Consistency Group name. A non-Data ONTAP source endpoint (for example, AltaVault) can be specified in the form `hostip:/share/share-name`.

**| [-source-cluster <Cluster name>] - Source Cluster**

Specifies the cluster in which the source volume resides. This parameter is not needed; it is provided for consistency with other `snapmirror` commands. If this parameter is specified, the `-source-vserver` and `-source-volume` parameters must also be specified. This parameter is not valid when operating in a Vserver context. This parameter is not supported if the source is a non-Data ONTAP endpoint.

**[-source-vserver <vserver name>] - Source Vserver**

Specifies the source Vserver of the SnapMirror relationship. If this parameter is specified, the `-source-volume` parameter must also be specified. This parameter is not supported if the source is a non-Data ONTAP endpoint.

**[-source-volume <volume name>] - Source Volume }**

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, the `-source-vserver` parameter must also be specified. This parameter is not supported if the source is a non-Data ONTAP endpoint.

**{ -destination-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path**

Specifies the destination endpoint in one of three formats. The basic format includes the names of the Vserver (vserver) and volume (volume). A format that also includes the name of the cluster (cluster) is supported for consistency with other `snapmirror` commands. The form of the pathname which includes the cluster name is not valid when operating in a Vserver context. Destination endpoint having Application Consistency Group includes vserver name followed by `cg` and Application Consistency Group name.

### **[`-destination-cluster` <Cluster name>] - Destination Cluster**

Specifies the cluster in which the destination volume resides. This parameter is not needed; it is provided for consistency with other `snapmirror` commands. If this parameter is specified, the `-destination-vserver` and `-destination-volume` parameters must also be specified. This parameter is not valid when operating in a Vserver context. This parameter is only applicable for relationships with "Relationship Capability" of "Pre 8.2".

### **`-destination-vserver` <vserver name> - Destination Vserver**

Specifies the destination Vserver. If this parameter is specified, the `-destination-volume` parameter must also be specified.

### **`-destination-volume` <volume name> - Destination Volume }**

Specifies the destination volume. If this parameter is specified, the `-destination-vserver` parameter must also be specified.

### **[`-s`, `-source-snapshot` <text>] - Source Snapshot**

When restoring the entire contents of a Snapshot copy, this optional parameter identifies the Snapshot copy to be restored from the source volume to the destination volume. The default value is the latest snapshot on the source volume. When restoring one or more files, LUNs or NVMe namespaces from a Snapshot copy, this parameter is required.

### **[`-k`, `-throttle` <throttleType>] - Throttle (KB/sec)**

This optional parameter limits the network bandwidth used for the restore transfer when the source and destination volumes belong to different clusters. It sets the maximum rate (in Kbytes/sec) at which data can be transferred between the clusters during the operation. To fully use the network bandwidth available between the clusters, set the throttle value to *unlimited* or *0*. The minimum throttle value is four Kbytes/sec, so if you specify a throttle value between *1* and *4*, it will be treated as if you specified *4*.

### **[`-transfer-priority` {*low*|*normal*}] - Transfer Priority**

This optional parameter specifies the priority at which the transfer runs. The default value for this parameter is *normal*.

### **[`-cg-item-mappings` {<source volume>:@<destination volume>|<source item path>:@<destination item path>}] - Consistency Group Item Mappings**

This optional parameter specifies a list of the consistency group (CG) item mappings. It is considered only if the supplied source and destination path values denote valid CG paths. For CG level relationships, this parameter must be specified. The value for this parameter must contain mappings of one or more pairs of constituent source and destination volumes of the form *srcv11:@dstv11,...*.

### **[`-disable-storage-efficiency` <true>] - Disable Storage Efficient Transfer**

The default behavior of restore is to preserve storage efficiency when possible. Use this optional parameter to turn off storage efficiency for data transferred over the wire and written to the destination volume.

### **[`-clean-up-failure` <true>] - Clean up after Failure**

Use this optional parameter to recover from an aborted or failed restore operation. Any temporary *RST* relationship is removed from the destination Vserver. An attempt is made to remove any temporary *RST* relationship from the source Vserver. If cleaning up an incomplete restore of the entire contents of a Snapshot copy and the destination volume was read-write prior to the failed or aborted restore operation, it is converted back to read-write if necessary, while removing all data transferred or copied during the restore operation. If cleaning up an incomplete restore of one or more files, LUNs or NVMe namespaces of a Snapshot copy, any file to which client I/O is not allowed is deleted.

### **[-tries <integer\_or\_unlimited>] - Tries Limit**

Specifies the total number of attempts to transfer data in cases where a transfer is interrupted by an error that SnapMirror can recover from. The value of this parameter must be a positive integer or *unlimited*.

### **[-f, -force <true>] - Force**

If this parameter is specified, the command proceeds without prompting for confirmation.

### **[-file-list <<source path>[,@<destination path>]>,...] - File List**

Specifies the files, LUNs or NVMe namespaces to be restored. The list can contain specifications for up to 8 files, LUNs or NVMe namespaces. Specification for each file, LUN or NVMe namespace consists of a *source\_path* and an optional *destination\_path*, and is of the form '*source\_path*[@*destination\_path*]. *source\_path* is required and is the path of the file from the source Snapshot copy, e.g. /dira/file1 or /lun1. The source path does not include the Snapshot name nor the source volume name. The path to each file to be restored in the active file system of the destination volume is the same as the path specified by *source\_path*, unless an optional *destination\_path* is specified. *destination\_path* begins with the @ symbol followed by the path of the file from the root of the active file system of the destination volume, e.g. @/file1 or @/dira/lun1. File restore is not supported if the *source\_path* or *destination\_path* contains commas in its directory or file name. Each *source\_path* and *destination\_path* is a separate entity in the list of paths. A *destination\_path* is associated with the *source\_path* that immediately precedes it. If this parameter is specified, `-source-snapshot` must also be specified. Examples:

```
/dira/file1
```

```
/dira/file1,@/dirb/file2
```

```
/dira/file1,@/dirb/file2,/dirc/file3
```

### **[-use-network-compression <true>] - Use Network Compression**

Use this optional parameter to use network compression for data transfer over the wire. This parameter is not supported for relationships with non-Data ONTAP endpoints.

### **[-backoff-level {high|medium|none}] - SM Backoff Level due to Client Ops**

This optional parameter specifies the SnapMirror backoff level due to client ops. This parameter is supported only for FlexVol SnapMirror relationships. The default value is *high*.

## **Examples**

The following example does an incremental restore between the restore source volume `vs2.example.com:dept_eng_dp_mirror2` and the restore destination volume `vs1.example.com:dept_eng`:



```

vs1.example.com::> snapmirror restore
  -destination-path vs1.example.com:dept_eng
  -source-path vs2.example.com:dept_eng_dp_mirror2
  -source-snapshot snap3
Warning: All data newer than Snapshot copy snap6 on volume
        vs1.example.com:dept_eng will be deleted.
Do you want to continue? {y|n}: y
[Job 34] Job is queued: snapmirror restore from source
vs2.example.com:dept_eng_dp_mirror2 for the snapshot snap3.
vs1.example.com::>

```

The following example does a restore between the source Application Consistency Group *vs2.example.com:/cg/src\_cg* and destination Application Consistency Group *vs1.example.com:/cg/dst\_cg*:

```

vs1.example.com::> snapmirror restore
  -destination-path vs1.example.com:/cg/dst_cg
  -source-path vs2.example.com:/cg/src_cg
  -source-snapshot snap3
[Job 34] Job is queued: snapmirror restore from source
vs2.example.com:/cg/src_cg for the snapshot snap3.
vs1.example.com::>

```

The following example restores */file3* from the source Snapshot copy *snap3* on the source volume *vs2.example.com:dept\_eng\_dp\_mirror2* to the active file system of the restore destination volume *vs1.example.com:dept\_eng*:

```

vs1.example.com::> snapmirror restore
  -destination-path vs1.example.com:dept_eng
  -source-path vs2.example.com:dept_eng_dp_mirror2
  -source-snapshot snap3
  -file-list /file3
Warning: This command will overwrite any file on destination
"vs1.example.com:dept_eng" that has the same path as any of
the files to be restored.
Do you want to continue? {y|n}: y
[Job 35] Job is queued: snapmirror restore from source
"vs2.example.com:dept_eng_dp_mirror2" for the snapshot snap3.
vs1.example.com::>

```

The following example restores */file3* from the source Snapshot copy *snap3* on the source volume *vs2.example.com:dept\_eng\_dp\_mirror2* to */file3.new* in the active file system of the restore destination volume *vs1.example.com:dept\_eng*:

```

vs1.example.com::> snapmirror restore
  -destination-path vs1.example.com:dept_eng
  -source-path vs2.example.com:dept_eng_dp_mirror2
  -source-snapshot snap3
  -file-list /file3,@/file3.new
Warning: This command will overwrite any file on destination
"vs1.example.com:dept_eng" that has the same path as any of
the files to be restored.
Do you want to continue? {y|n}: y
[Job 36] Job is queued: snapmirror restore from source
"vs2.example.com:dept_eng_dp_mirror2" for the snapshot snap3.
vs1.example.com::>

```

The following example restores `/file1`, `/file2`, and `/file3` from the source Snapshot copy `snap3` on the source volume `vs2.example.com:dept_eng_dp_mirror2` respectively to `/file1.new`, `/file2`, and `/file3.new` in the active file system of the restore destination volume `vs1.example.com:dept_eng`:

```

vs1.example.com::> snapmirror restore
  -destination-path vs1.example.com:dept_eng
  -source-path vs2.example.com:dept_eng_dp_mirror2
  -source-snapshot snap3
  -file-list /file1,@/file1.new,/file2,/file3,@/file3.new
Warning: This command will overwrite any file on destination
"vs1.example.com:dept_eng" that has the same path as any of
the files to be restored.
Do you want to continue? {y|n}: y
[Job 36] Job is queued: snapmirror restore from source
"vs2.example.com:dept_eng_dp_mirror2" for the snapshot snap3.
vs1.example.com::>

```

## Related Links

- [volume quota modify](#)
- [volume snapshot create](#)
- [snapmirror show](#)
- [snapmirror update](#)
- [lun delete](#)
- [vserver nvme namespace delete](#)

## snapmirror resume

Enable future transfers

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror resume` command enables future transfers for a SnapMirror relationship that has been quiesced.

If there is a scheduled transfer for the relationship, it will be triggered on the next schedule. If there is a restart checkpoint, it will be re-used if possible.

If applied on a load-sharing (LS) SnapMirror relationship, it enables future transfers for all the relationships in the load-sharing set.

If applied on a relationship with a policy of type *strict-sync-mirror* or *sync-mirror*, it enables future resync operations and initiates an Auto Resync.

When a quiesced SnapMirror relationship is resumed, future transfers remain enabled across reboots and fail-overs.

This command is supported for SnapMirror relationships with the field *Relationship Capability* showing as either *"8.2 and above"* or *"Pre 8.2"* in the output of the `snapmirror show` command.

The `snapmirror resume` command must be used from the destination Vserver or cluster.

The relationship must exist on the destination Vserver or cluster. When issuing `snapmirror resume`, you must specify the destination endpoint. The specification of the source endpoint of the relationship is optional.

## Parameters

**{ [-S, -source-path <[vserver:] [volume]>|<[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}] - Source Path**

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or the volume (volume). To support relationships with *Relationship Capability* of *"Pre 8.2"*, a format which also includes the name of the cluster (cluster) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *Relationship Capability* of *"8.2 and above"*. For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form `hostip:/share/share-name`. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form `hostip:/lun/name`.

**| [-source-cluster <Cluster name>] - Source Cluster**

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source`, `-vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with *Relationship Capability* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *Relationship Capability* of *"8.2 and above"*.

**[-source-vserver <vserver name>] - Source Vserver**

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with *Relationship Capability* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not

supported for relationships with non-Data ONTAP source endpoints.

#### **{ -source-volume <volume name> } - Source Volume }**

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

#### **{ -destination-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path**

This parameter specifies the destination endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or volume (volume). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (cluster) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with AltaVault destinations, the destination endpoint is specified in the form `hostip:/share/share-name`. For relationships with SolidFire destinations, the destination endpoint is specified in the form `hostip:/lun/name`.

#### **| [-destination-cluster <Cluster name>] - Destination Cluster**

Specifies the destination cluster of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and `-destination-volume` must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

#### **-destination-vserver <vserver name> - Destination Vserver**

Specifies the destination Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-destination-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

#### **-destination-volume <volume name> - Destination Volume }**

Specifies the destination volume of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

## Examples

To re-enable future transfers for the SnapMirror relationship with the destination endpoint `vs2.example.com:dept_eng_dp_mirror2` that has been previously quiesced, type the following command:

```
vs2.example.com::> snapmirror resume -destination-path
vs2.example.com:dept_eng_dp_mirror2
```

To re-enable future transfers for the SnapMirror relationship with the destination endpoint `cluster2://vs2.example.com/dept_eng_dp_mirror2` that has been previously quiesced, type the

following command:

```
cluster2::> snapmirror resume -destination-path
cluster2://vs2.example.com/dept_eng_dp_mirror2
```

To re-enable future transfers for the Vserver SnapMirror relationship with the destination endpoint *dvs1.example.com*: that has been previously quiesced, type the following command:

```
cluster2::> snapmirror resume -destination-path
dvs1.example.com:
```

To re-enable future transfers of the Application Consistency Group SnapMirror relationship with the destination Consistency Group *app\_cg\_dst* in Vserver *vs2.example.com*, type the following command:

```
vs2.example.com::> snapmirror resume -destination-path
vs2.example.com:/cg/app_cg_dst
```

## Related Links

- [snapmirror show](#)

# snapmirror resync

Start a resynchronize operation

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror resync` command establishes or reestablishes a mirroring relationship between a source and a destination endpoint. The endpoints can be Vservers, volumes, Application Consistency Groups or non-Data ONTAP endpoints that support SnapMirror. `snapmirror resync` for a SnapMirror relationship with volumes as endpoints is typically executed in the following cases:

- The destination mirror is broken (that is, the destination volume is a read-write volume and no longer a data protection mirror). After the `snapmirror resync` command completes, the destination volume is made a data protection mirror and the mirror can be manually updated or scheduled for updates.
- [snapmirror update](#) command failed because the required common Snapshot copy was deleted on the source volume.
- The volumes are the first and third endpoints in a cascade chain of relationships and they have a common Snapshot copy. In this case, `snapmirror resync` might implicitly create the SnapMirror relationship between them.



The `snapmirror resync` command can cause data loss on the destination volume because the command can remove the exported Snapshot copy on the destination volume.

The default behavior of the `snapmirror resync` command for volume relationships is defined as follows:

- Finds the most recent common Snapshot copy between the source and destination volumes, removes Snapshot copies on the destination volume that are newer than the common Snapshot copy and mounts the destination volume as a DP volume with the common Snapshot copy as the exported Snapshot copy.
- For data protection (DP) relationships, takes a Snapshot copy of the source volume to capture the current image and transfers Snapshot copies that are newer than the common Snapshot copy from the source volume to the destination volume. For extended data protection (XDP) relationships, transfers Snapshot copies newer than the common Snapshot copy according to the relationship policy, i.e., Snapshot copies will match rules associated with the policy as defined by the `snapmirror policy` commands. For relationships associated with `snapmirror policy` of type `async-mirror` and `mirror-vault` the `snapmirror resync` first takes a Snapshot copy of the source volume and includes it in the Snapshot copies selected for transfer.
- For a SnapLock Compliance volume in an XDP relationship with SnapMirror policy of type `async-mirror`, if SnapMirror resync operation detects data divergence between the common Snapshot copy and the AFS on the destination volume, the resync operation preserves the data changes in a locked Snapshot copy for the duration of the current volume expiry time. If the volume expiry time is in the past or has not been set, then the Snapshot copy is locked for a duration of 30 days. The common Snapshot copy is also locked for the same duration.

For Vserver SnapMirror relationships, a resync operation is typically executed when the relationship is broken-off, the `subtype` of the destination Vserver is `default` and the destination volumes are of type read-write. Once the command is queued, the subtype of the destination Vserver changes from `default` to `dp-destination`. A successful resync operation also makes the destination Vserver's volumes data protection volumes.

If the resync command is executed on a Vserver SnapMirror relationship, and the corresponding source and destination Vservers have volumes with volume level SnapMirror relationships, then the volume level SnapMirror relationships will be converted to volumes under the Vserver SnapMirror relationship. This conversion is supported only for source and destination Vservers which have been transitioned from a 7-Mode vFiler into a C-Mode Vserver. Some basic pre-requisites for the conversion are that the destination Vserver should be in a `stopped` state and all the destination Vserver volumes except the root volume should be in a volume level SnapMirror relationship with volumes of the source Vserver. The state of these volume level SnapMirror relationships should be `Snapmirrored` and status should be `Idle`.

When the destination Application Consistency Group relationship is broken-off, the `snapmirror resync` changes the destination Application Consistency Group to read-only and the relationship will be SnapMirrored.

`snapmirror resync` for a relationship with a policy of type `strict-sync-mirror` or `sync-mirror` is typically executed in the following case:

- The destination mirror is broken (that is, the destination volume is read-write and no longer read-only). After the `snapmirror resync` command completes, the destination volume changes to read-only and the relationship to InSync.



The `snapmirror resync` command is typically not required to return a relationship that has fallen out of sync due to an error condition to InSync because SnapMirror has Auto Resync for synchronous relationships. When SnapMirror detects that the relationship has fallen out of sync for any reason other than a `snapmirror quiesce`, `snapmirror break` or `snapmirror delete` command was executed on the relationship, it will automatically initiate a resync operation.

The default behavior of the `snapmirror resync` command for relationships with a policy of type `strict-`

`sync-mirror` or `sync-mirror` is defined as follows:

- Creates a Snapshot copy on the destination of the current image of the destination file system. This Snapshot copy becomes the exported Snapshot copy for the volume during the resync operation.
- Finds the most recent common Snapshot copy between the source and destination volumes. Performs a local rollback transfer to give the active file system the same data as the common Snapshot copy. It then loops through a sequence, creating a Snapshot copy on the source volume, transferring the data captured in that Snapshot copy, creating a Snapshot copy of the data on the destination, and repeating until the relationship is close to InSync. After the last transfer, it enters cutover to bring the relationship to InSync.
- User-created Snapshot copies are not replicated by a resync operation.
- At the conclusion of the resync operation, the exported Snapshot copy on the destination is removed and the client will then see the active file system on the destination volume. The relationship will be InSync and periodic creation of common Snapshot copies will resume.

The `snapmirror resync` command supports an optional parameter `"preserve"`. The parameter `"preserve"` is only supported for extended data protection (XDP) relationships. It is not supported for relationships with a non-Data ONTAP endpoint. It is not supported for relationships with a policy of type `strict-sync-mirror` and `sync-mirror`. When used, the parameter `"preserve"` changes the behavior of the `snapmirror resync` command. The changed behavior of the command can be described as follows:

- Finds the most recent common Snapshot copy between the source and destination volumes, preserves all Snapshot copies on the destination volume that are newer than the common Snapshot copy, and mounts the destination volume as a DP volume with the common Snapshot copy as the exported Snapshot copy.
- Performs a local rollback transfer to make a copy of the common Snapshot copy on the destination volume and establish it as the latest Snapshot copy on the destination volume. The command then transfers all Snapshot copies that are newer than the common Snapshot copy, from the source volume to the destination volume. The command only transfers Snapshot copies that match the relationship's policy, i.e., Snapshot copies will match rules associated with the policy as defined by the `snapmirror policy` commands.

If a SnapMirror relationship does not already exist, that is, the relationship was not created using the [snapmirror create](#) command, the `snapmirror resync` command will implicitly create the SnapMirror relationship, with the same behaviors as described for the [snapmirror create](#) command before resyncing it.

For Vservers, you must create SnapMirror relationships between Vservers by using the [snapmirror create](#) command before you run the `snapmirror resync` command. The `snapmirror resync` command does not implicitly create the relationship.

This command is supported for SnapMirror relationships with the field `"Relationship Capability"` showing as either `"8.2 and above"` or `"Pre 8.2"` in the output of the [snapmirror show](#) command.

For relationships with `"Relationship Capability"` of `"8.2 and above"`, you can track the progress of the operation using the [snapmirror show](#) command.

For relationships with `"Relationship Capability"` of `"Pre 8.2"`, a job will be spawned to operate on the SnapMirror relationship, and the job id will be shown in the command output. The progress of the job can be tracked using the [job show](#) and [job history show](#) commands.

The `snapmirror resync` command fails if the destination volume does not have a Snapshot copy in common with the source volume.

The `snapmirror resync` command does not work on load-sharing mirrors.

The `snapmirror resync` command must be used from the destination Vserver or cluster.

## Parameters

### { `-S, -source-path`

{<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Source Path

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or the volume (volume). To support relationships with "Relationship Capability" of "Pre 8.2", a format which also includes the name of the cluster (cluster) is provided. The "Pre 8.2" format cannot be used when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above". For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form `hostip:/share/share-name`. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form `hostip:/lun/name`.

### | `-source-cluster <Cluster name>` - Source Cluster

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source -vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with "Relationship Capability" of "Pre 8.2". This parameter cannot be specified when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above".

### | `-source-vserver <vserver name>` - Source Vserver

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with "Relationship Capability" of "Pre 8.2", `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

### | `-source-volume <volume name>` - Source Volume }

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with "Relationship Capability" of "Pre 8.2", `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

### { `-destination-path`

{<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path

This parameter specifies the destination endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or volume (volume). To support relationships with "Relationship Capability" of "Pre 8.2", a format which also includes the name of the cluster (cluster) is provided. The "Pre 8.2" format cannot be used when operating in a Vserver context on relationships with "Relationship Capability" of "8.2 and above". For SnapMirror relationships with AltaVault destinations, the destination endpoint is specified in the form `hostip:/share/share-name`. For relationships with SolidFire destinations, the destination endpoint is specified in the form `hostip:/lun/name`.

### | `-destination-cluster <Cluster name>` - Destination Cluster

Specifies the destination cluster of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and `-destination-volume` must also be specified. This parameter is only applicable for relationships with "Relationship Capability" of "Pre 8.2". This parameter cannot



be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

#### **-destination-vserver <vserver name> - Destination Vserver**

Specifies the destination Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-destination-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

#### **-destination-volume <volume name> - Destination Volume }**

Specifies the destination volume of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

#### **[-s, -source-snapshot <text>] - Source Snapshot**

This optional parameter specifies a Snapshot copy to transfer. The default behavior for a data protection relationship with a read-write source is that Data ONTAP creates a new Snapshot copy and uses it as the basis for determining what data are replicated; with this option, the specified Snapshot copy will be used instead. The default behavior for an extended data protection relationship depends on the relationship's policy type. For a data protection relationship, the specified Snapshot copy must be newer than the latest common Snapshot copy. For an extended data protection relationship, the specified Snapshot copy can be newer or older than the common Snapshot copy. This parameter is not supported for relationships with *"Relationship Capability"* of *"Pre 8.2"*.

#### **[-type <snapmirrorType>] - Snapmirror Relationship Type**

Specifies the type of SnapMirror relationship if a relationship is implicitly created. This parameter is the same as the one used in the [snapmirror create](#) command.

#### **[-policy <sm\_policy>] - SnapMirror Policy**

This optional parameter designates the name of the SnapMirror policy which is associated with the SnapMirror relationship. If you do not designate a policy, the current policy will be retained. This parameter is not applicable to relationships with *"Relationship Capability"* of *"Pre 8.2"*.



You define and name a policy using the [snapmirror policy create](#) command.

#### **[-f, -force <>true>] - Force**

If this parameter is specified, the command proceeds without prompting for confirmation.

#### **[-k, -throttle <throttleType>] - Throttle (KB/sec)**

This optional parameter limits the network bandwidth used for the resync transfer. It sets the maximum rate (in Kbytes/sec) at which data can be transferred during the operation. If this parameter is not specified, the throttle value configured for the relationship with the [snapmirror create](#) or [snapmirror modify](#) command will be used. To fully use the network bandwidth available, set the throttle value to *unlimited* or *0*. The minimum throttle value is four Kbytes/sec, so if you specify a throttle value between *1* and *4*, it will be treated as if you specified *4*. For FlexGroup and Application Consistency Group relationships, the throttle value is applied individually to each constituent relationship. For relationships with a policy of type *strict-sync-mirror* or *sync-mirror*, the throttle value is applicable only for the asynchronous transfers of the resync operation. The `-throttle` parameter does not affect load-sharing transfers and transfers for other relationships with *"Relationship Capability"* of *"Pre 8.2"* confined to a single cluster.

### **[-transfer-priority {low|normal}] - Transfer Priority**

This optional parameter specifies the priority at which the transfer runs. The default value for this parameter is the value in the SnapMirror policy associated with this relationship. This parameter is not applicable to relationships with a *Relationship Capability* of *Pre 8.2*.

### **[-preserve <true>] - Preserve**

This parameter is only supported for extended data protection (XDP) relationships with policies of type *vault*, and *mirror-vault*. It is not supported for relationships with a policy of type *async-mirror* and data protection and load-sharing relationships. This parameter is not supported for relationships with non-Data ONTAP endpoints. It is not supported for relationships with a policy of type *strict-sync-mirror* and *sync-mirror*. When specified, it changes the behavior of the `snapmirror resync` command to preserve Snapshot copies on the destination volume that are newer than the latest common Snapshot copy. This parameter is not supported for relationships with *Relationship Capability* of *Pre 8.2*.

### **[-quick-resync <true>] - Quick Resync**

This parameter is only supported for extended data protection (XDP) relationships. This parameter is not supported for relationships with non-Data ONTAP endpoints. It is not supported for relationships with a policy of type *strict-sync-mirror* and *sync-mirror*. Specifying this optional parameter reduces the resync time because the resync does not incur storage efficiency overhead before the transfer of new data. Specifying this parameter is recommended if the source of the resync does not have volume efficiency enabled or if reducing resync time is more important than preserving all possible storage efficiency. When this parameter is specified, resync does not preserve the storage efficiency of the new data with existing data over the wire and on the destination.

### **[-cg-item-mappings {<source volume>:@<destination volume>|<source item path>:@<destination item path>}] - Consistency Group Item Mappings**

This optional parameter specifies a list of the consistency group (CG) item mappings. It is considered only if the supplied source and destination path values denote valid CG paths. For CG level relationships, this parameter must be specified. The value for this parameter must contain mappings of one or more pairs of constituent source and destination volumes of the form *srcvol1:@dstvol1,...*.

### **[-is-auto-expand-enabled {true|false}] - Is Auto Expand Enabled**

This optional parameter specifies whether or not a FlexGroup SnapMirror relationship and its destination FlexGroup should be auto-expanded if the source FlexGroup is expanded. This parameter is supported only for FlexGroup SnapMirror relationships. If this resync is creating a new Snapmirror relationship, the default value is *true*. If it is not creating a new relationship, if a value is specified, it must match the current value for the existing relationship. If the parameter is not specified, the existing value will be retained.

### **[-backoff-level {high|medium|none}] - SM Backoff Level due to Client Ops**

This optional parameter specifies the SnapMirror backoff level due to client ops. This parameter is supported only for FlexVol SnapMirror relationships. The default value is *high*.

### **[-w, -foreground <true>] - Foreground Process**

This specifies whether the operation runs as a foreground process. If this parameter is specified, the default setting is *true* (the operation runs in the foreground). When set to *true*, the command will not return until the process completes. This parameter is only applicable to relationships with *Relationship Capability* of *Pre 8.2*.

## **Examples**

To reestablish mirroring for the destination endpoint `vs2.example.com:dept_mkt_mirror` that has been

previously broken off with the `snapmirror break` command, type the following command:

```
vs2.example.com::> snapmirror resync -destination-path
vs2.example.com:dept_mkt_dp_mirror
```

For relationships with "Relationship Capability" of "Pre 8.2", to reestablish mirroring for the destination endpoint `cluster2://vs2.example.com/dept_mkt_mirror` that has been previously broken off with the `snapmirror break` command, type the following command:

```
cluster2::> snapmirror resync -destination-path
cluster2//vs2.example.com/dept_mkt_dp_mirror
```

To create a SnapMirror relationship and reestablish mirroring between the destination endpoint named `vs2.example.com:dept_eng_dp_mirror2` and the source endpoint named `vs1.example.com:dept_eng`, type the following command:

```
vs2.example.com::> snapmirror resync -destination-path
vs2.example.com:dept_eng_dp_mirror2
-source-path vs1.example.com:dept_eng
```

To create a SnapMirror relationship and reestablish mirroring between the destination endpoint named `cluster2://vs2.example.com/dept_eng_dp_mirror2` and the source endpoint named `cluster1://vs1.example.com/dept_eng` when the source cluster is running Data ONTAP 8.1 software, type the following command:

```
cluster2::> snapmirror resync -destination-path
cluster2://vs2.example.com/dept_eng_dp_mirror2
-source-path cluster1://vs1.example.com/dept_eng
```

To create and reestablish an extended data protection (XDP) relationship between the Data ONTAP source endpoint `vs1.example.com:data_ontap_vol`, and the non-Data ONTAP (for example, AltaVault) destination endpoint `10.0.0.11:/share/share1`, and start the initial transfer, type the following command:

```
vs1.example.com::> snapmirror resync -destination-path
10.0.0.11:/share/share1
-source-path vs1.example.com:data_ontap_vol -type XDP
```

To reestablish mirroring for the destination endpoint `dvs1.example.com:` of a Vserver relationship that has been previously broken off with the `snapmirror break` command, type the following command:

```
cluster2::> snapmirror resync -destination-path
dvs1.example.com:
```

To resynchronize the SnapMirror active sync relationship with the source Consistency Group *cg\_src* in Vserver *vs1.example.com* and the destination Consistency Group *cg\_dst* in Vserver *vs2.example.com*, type the following command from the destination cluster:

```
destination::> snapmirror resync -destination-path
vs2.example.com:/cg/cg_dst
```

To resynchronize an SnapMirror active sync relationship with the following attributes:

- It is between the source Consistency Group *cg\_src* in Vserver *vs1.example.com*, and the destination Consistency Group *cg\_dst* in Vserver *vs2.example.com*.
- It has item mappings between volumes *srcvol1* and *srcvol2* and volumes *dstvol1* and *dstvol2*.

Type the following command from the destination cluster:

```
destination::> snapmirror resync -destination-path
vs2.example.com:/cg/cg_dst -source-path
vs1.example.com:/cg/cg_src
-cg-item-mappings srcvol1:@dstvol1,srcvol2:@dstvol2
```

To reestablish mirroring to the destination Application Consistency Group *app\_cg\_dst* in Vserver *vs2.example.com* that has been previously broken off with the [snapmirror break](#) command, type the following command:

```
vs2.example.com::> snapmirror resync -destination-path
vs2.example.com:/cg/app_cg_dst
```

## Related Links

- [snapmirror update](#)
- [snapmirror quiesce](#)
- [snapmirror break](#)
- [snapmirror delete](#)
- [snapmirror create](#)
- [snapmirror show](#)
- [job show](#)
- [job history show](#)
- [snapmirror policy create](#)
- [snapmirror modify](#)

# snapmirror set-options

Display/Set SnapMirror options

**Availability:** This command is available to *cluster* administrators at the *admin* privilege level.

## Description

The `snapmirror set-options` command can be used to display or set snapmirror options.

## Parameters

**[-dp-source-xfer-reserve-pct {0|25%|50%|75%|100%}] - Percentage Reserved for DP Source Transfers**

Specifies the percentage of maximum allowed concurrent transfers reserved for source DP transfers

**[-xdp-source-xfer-reserve-pct {0|25%|50%|75%|100%}] - Percentage Reserved for XDP Source Transfers**

Specifies the percentage of maximum allowed concurrent transfers reserved for source XDP transfers

**[-dp-destination-xfer-reserve-pct {0|25%|50%|75%|100%}] - Percentage Reserved for DP Destination Transfers**

Specifies the percentage of maximum allowed concurrent transfers reserved for destination DP transfers

**[-xdp-destination-xfer-reserve-pct {0|25%|50%|75%|100%}] - Percentage Reserved for XDP Destination Transfers**

Specifies the percentage of maximum allowed concurrent transfers reserved for destination XDP transfers

## Examples

The following example displays SnapMirror options:

```
cluster1::> snapmirror set-options
  Percentage Reserved for DP Source Transfers: 0
  Percentage Reserved for XDP Source Transfers: 0
  Percentage Reserved for DP Destination Transfers: 0
  Percentage Reserved for XDP Destination Transfers: 0

cluster1::> snapmirror set-options -dp-source-xfer-reserve-pct 25
  -xdp-source-xfer-reserve-pct 50 -dp-destination-xfer
  reserve-pct 0 -xdp-destination-xfer-reserve-pct 50

cluster1::> snapmirror set-options
  Percentage Reserved for DP Source Transfers: 25
  Percentage Reserved for XDP Source Transfers: 50
  Percentage Reserved for DP Destination Transfers: 0
  Percentage Reserved for XDP Destination Transfers: 50
```

# snapmirror set-preferred-cluster

Set this cluster as the preferred cluster

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror set-preferred-cluster` sets the local cluster as the preferred cluster for SnapMirror active sync relationships of policy-type 'automated-failover-duplex'. Note that the 'preferred-cluster' will be the one serving IO upon scenarios like a network partition between primary and secondary.

You should execute the `snapmirror set-preferred-cluster` command from the destination cluster.

Make sure the relationship status is InSync and the ONTAP Mediator is configured, connected, and in quorum before using this command.

## Parameters

### -destination-path

{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path

Specifies the destination endpoint of the SnapMirror relationship.

## Examples

To perform a set-preferred-cluster operation of an SnapMirror active sync relationship with source Consistency Group `cg_src` in Vserver `vs1.example.com`, and the destination Consistency Group `cg_dst` in Vserver `vs2.example.com`, type the following command from the destination cluster:

```
destination::> snapmirror set-preferred-cluster -destination-path
vs2.example.com:/cg/cg_dst
```

Note: The SnapMirror set-preferred-cluster operation internally triggers a planned failover operation. Hence, the SnapMirror active sync relationship information in the [snapmirror show](#) command will now be available on the new destination that is Vserver `vs2.example.com` with `cg_dst` as the new source and `cg_src` as the new destination Consistency Groups after the SnapMirror set-preferred-cluster operation is completed.

## Related Links

- [snapmirror show](#)

# snapmirror show-history

Displays history of SnapMirror operations.

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror show-history` command displays the history of the last 24 hours of SnapMirror operations. This command is not supported for relationships with non-Data ONTAP endpoints.

By default, the command displays the following information:

- Destination Path
- Source Path
- Operation
- Start Time
- End Time
- Result

The `snapmirror show-history` command displays - in reverse chronological order - the history of completed SnapMirror operations whose destination endpoints are in the current Vserver for Vserver administrators, or the current cluster for cluster administrators. This command does not return information on the following operations and relationships:

- Operations that happened prior to installing Data ONTAP 8.3.
- Relationships with the *"Relationship Capability"* field, as shown in the output of the SnapMirror show command, set to *"Pre 8.2"*.
- Operations on FlexGroup relationships that happened prior to installing Data ONTAP 9.5.
- Operations on FlexGroup constituent relationships.

The `-instance` parameter displays the following detailed information:

Destination Path: Path of the destination endpoint.

Source Path: Path of the source endpoint.

Relationship ID: The unique identifier of the relationship.  
This parameter is not supported for  
Vserver SnapMirror relationships.

Relationship Group Type: For FlexVol relationships, specifies the  
type of the group relationship that  
includes this FlexVol. For group  
relationships, specifies the type of the  
group relationship. Can be one of the  
following:

- none: No group relationship.
- vserver: Vserver relationship.
- flexgroup: FlexGroup relationship.

Operation: Type of the operation.

Can be one of the following:

- create
- modify
- quiesce
- resume
- delete
- initialize
- manual update
- scheduled update
- break
- resync
- abort
- restore

Operation ID: The unique identifier of the operation.

Start Time: Timestamp of the start of the operation.

End Time: Timestamp of the end of the operation.

Result: Result of the SnapMirror operation.

Can be one of the following:

- success
- failure

Transfer Size: Total amount of data transferred during the  
SnapMirror operation.

Additional Information: A message describing the cause of the  
failure or additional information about a  
successful operation, such as if a

checkpoint

was cleared as part of an abort operation.



## Parameters

**{ [-fields <fieldname>,...]**

If you specify the `-fields <fieldname>`, ... parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays detailed information about all fields.

**{ [-destination-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}} - Destination Path**

Select SnapMirror operations that have a matching destination path name.

**| [-destination-vserver <vserver name>] - Destination Vserver**

Select SnapMirror operations that have a matching destination Vserver name.

**[-destination-volume <volume name>] - Destination Volume }**

Select SnapMirror operations that have a matching destination volume name.

**[-operation-id <UUID>] - Operation ID**

Select SnapMirror operations that have a matching operation ID.

**[-source-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}} - Source Path**

Select SnapMirror operations that have a matching source path name.

**[-source-vserver <vserver name>] - Source Vserver**

Select SnapMirror operations that have a matching source Vserver name.

**[-source-volume <volume name>] - Source Volume**

Select SnapMirror operations that have a matching source volume name.

**[-operation-type {create|modify|quiesce|resume|delete|initialize|manual-update|scheduled-update|break|resync|abort|restore|failover|rsetup|baseline-gather}] - Operation Type**

Select SnapMirror operations that have a matching operation type. Possible values are:

- create
- modify
- quiesce
- resume
- delete
- initialize
- manual-update
- scheduled-update

- break
- resync
- abort
- restore

**[-start-time <MM/DD/YYYY HH:MM:SS>] - Start Time**

Select SnapMirror operations that have a matching start time.

**[-end-time <MM/DD/YYYY HH:MM:SS>] - End Time**

Select SnapMirror operations that have a matching end time.

**[-relationship-id <UUID>] - Relationship ID**

Select SnapMirror operations that have a matching relationship ID.

**[-relationship-group-type**

**{none|vserver|consistencygroup|flexgroup|vserverflexgroup|vserverconsistencygroup  
}] - Relationship Group Type**

Select SnapMirror relationships that have a matching relationship group type. Possible values are:

- none
- vserver
- flexgroup

**[-result {success|failure}] - Result of the Operation**

Select SnapMirror operations that have a matching result. Possible values are:

- success
- failure

**[-transfer-size {<integer>[KB|MB|GB|TB|PB]}] - Transfer Size**

Select SnapMirror operations that have a matching transfer size.

**[-additional-info <text>] - Additional Information**

Select SnapMirror operations that have matching additional information.

**[-max-rows-per-relationship <integer>] - Maximum Number of Rows per Relationship**

Select a matching number of SnapMirror operations per relationship.

**[-expand <true>] - Show Constituents of the Group.**

Select SnapMirror operations on relationships that are constituents and non-constituents of a group.

## Examples

The example below displays summary information for all SnapMirror operations on relationships with destination endpoints in the current cluster:

```

cluster2::> snapmirror show-history
Destination Source          Start      End
Path         Path           Operation Time      Time      Result
-----
dvs2:        vs2:           create    8/15/2013 08:22:23
                                     8/15/2013 08:22:25
                                     success
vs1:vol1     vs1:aggr1     manual-update
                                     8/15/2013 08:22:44
                                     8/15/2013 08:22:44
                                     failure
vs1:vol1     vs1:aggr1     initialize
                                     8/15/2013 08:22:25
                                     8/15/2013 08:22:26
                                     success
vs1:vol1     vs1:aggr1     create    8/15/2013 08:22:15
                                     8/15/2013 08:22:16
                                     success
vs1:vol2     vs1:aggr1     initialize
                                     8/15/2013 08:23:23
                                     8/15/2013 08:23:23
                                     failure
vs1:vol2     vs1:aggr1     create    8/15/2013 08:23:10
                                     8/15/2013 08:23:10
                                     success

6 entries were displayed.

```

The example below displays detailed information for the SnapMirror operation with operation ID *dc158715-0583-11e3-89bd-123478563412*

```

cluster2::> snapmirror show-history -operation-id dc158715-0583-11e3-89bd-
123478563412
Destination Path: vs1:vol1
          Source Path: vs1:aggr1
          Relationship ID: cb3d30a0-0583-11e3-89bd-123478563412
          Relationship Group Type: none
          Operation: manual-update
          Operation ID: dc158715-0583-11e3-89bd-123478563412
          Start Time: 8/15/2013 08:22:44
          End Time: 8/15/2013 08:22:44
          Result: failure
          Transfer Size: -
          Additional Information: Volume vs1:vol1 is restricted. Use the
command "volume online" to bring the volume online.

```

The example below displays detailed information for all SnapMirror operations on relationships with the *Result* of "success" and whose destination endpoints are in the current cluster.

```
cluster2::> snapmirror show-history -result success -instance
  Destination Path: vs1:vol1
    Source Path: vs1:aggr1
      Relationship ID: cb3d30a0-0583-11e3-89bd-123478563412
Relationship Group Type: none
  Operation: initialize
    Operation ID: d03ce1db-0583-11e3-89bd-123478563412
      Start Time: 8/15/2013 08:22:25
        End Time: 8/15/2013 08:22:26
          Result: success
            Transfer Size: 1.09MB
  Additional Information: -
Destination Path: vs1:vol1
  Source Path: vs1:aggr1
    Relationship ID: cb3d30a0-0583-11e3-89bd-123478563412
Relationship Group Type: none
  Operation: create
    Operation ID: cb3d305d-0583-11e3-89bd-123478563412
      Start Time: 8/15/2013 08:22:15
        End Time: 8/15/2013 08:22:16
          Result: success
            Transfer Size: -
  Additional Information: -
Destination Path: vs1:vol2
  Source Path: vs1:aggr1
    Relationship ID: eb92c549-0583-11e3-89bd-123478563412
Relationship Group Type: none
  Operation: create
    Operation ID: eb92c506-0583-11e3-89bd-123478563412
      Start Time: 8/15/2013 08:23:10
        End Time: 8/15/2013 08:23:10
          Result: success
            Transfer Size: -
  Additional Information: -

3 entries were displayed.
```

The example below displays summary information for all SnapMirror operations on relationships with *max-rows-per-relationship* of 1 and whose destination endpoints are in the current cluster.

```
cluster2::> snapmirror show-history -max-rows-per-relationship 1
```

Destination Path	Source Path	Operation	Start Time	End Time	Result
vs1:vol1	vs1:aggr1	manual-update	8/15/2013 08:22:44	8/15/2013 08:22:44	failure
vs1:vol2	vs1:aggr1	initialize	8/15/2013 08:23:23	8/15/2013 08:23:23	failure

```
2 entries were displayed.
```

## snapmirror show

Display a list of SnapMirror relationships

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### Description

The `snapmirror show` command displays information associated with SnapMirror relationships. By default, the command displays the following information:

- Source path
- Relationship Type
- Destination Path
- Mirror State
- Relationship Status
- Total Progress
- Healthy
- Progress Last Updated

For backward compatibility with clustered Data ONTAP 8.1, and to accommodate load-sharing relationships which are only supported in a Data ONTAP 8.1 compatible way, SnapMirror relationships, which match one of the following conditions are managed as on clustered Data ONTAP 8.1: (1) The relationship is of type load-sharing; (2) The source endpoint of the relationship is on a remote Data ONTAP 8.1 cluster; (3) The local cluster was upgraded from clustered Data ONTAP 8.1, the relationship was created before the upgrade, and the relationship has not yet been converted to one with Data ONTAP 8.2 capabilities. These relationships have the same limitations as on clustered Data ONTAP 8.1. Especially, they support the same set of information fields. The *"Relationship Capability"* field is set to *"Pre 8.2"* for these relationships.

The `snapmirror show` command displays information for SnapMirror relationships whose destination endpoints are in the current Vserver if you are in a Vserver context, or in the current cluster if you are in a

cluster context, or on a non-Data ONTAP endpoint that supports SnapMirror (for example, AltaVault). For backward compatibility with clustered Data ONTAP 8.1, the command also displays information for SnapMirror relationships with the "Relationship Capability" of "Pre 8.2", and whose source endpoints are in the current Vserver or cluster, and destination endpoints are in different Vservers or clusters. You must use the [snapmirror list-destinations](#) command to display information for SnapMirror relationships whose source endpoints are in the current Vserver or current cluster.

Some of the SnapMirror relationship information is cached. The `snapmirror show` command only returns the cached information, therefore there is a delay after the information is changed before it is reflected in the `snapmirror show` output. Other information, such as progress metrics during a transfer, is only updated periodically and can be very delayed in the `snapmirror show` output.

The `-instance` and `-fields` parameters are mutually exclusive and select the information fields that are displayed. The other parameters to the `snapmirror show` command select the SnapMirror relationships for which information is displayed. The `-instance` displays detailed information fields including:

Source Path: Path of the source endpoint.

Destination Path: Path of the destination endpoint.

Relationship Type: Type of the SnapMirror relationship. Can be one of the following:

- DP: Data protection relationship.
- LS: Load-sharing relationship.
- XDP: Extended data protection

relationship.

- RST: Temporary relationship created during a restore operation, and deleted if the operation completes successfully.
- TDP: 7-mode to clustered Data ONTAP transition data protection

- relationship.

Relationship Group Type: For FlexVol relationships, specifies the type of the group relationship that includes this FlexVol. For group relationships, specifies the type of the group relationship. Can be one of the following:

- none: No group relationship.
- vservers: Vserver relationship.
- flexgroup: FlexGroup relationship.
- consistencygroup: Consistency Group relationship.

Only for relationships with

"Relationship Capability"  
of "8.2 and above" .

Relationship Status: Status of the SnapMirror relationship.

Can be one of the following:

- Idle: No transfer operation is in progress and future transfers are not disabled.
- Queued: A transfer operation has been accepted and queued in the system, and future transfers are not disabled.
- Transferring: A transfer operation is in progress and future transfers are not disabled.
- Preparing: Pre-transfer phase for Vault incremental transfers. For Vault relationships only.
- Finalizing: Post-transfer phase for Vault incremental transfers. Network traffic will be low as processing is primarily on the destination volume. For Vault relationships only.
- Aborting: A transfer abort operation that might include the removal of the checkpoint is underway. Future transfers are not disabled. Only for relationships with "Relationship Capability" of "8.2 and above" .
- Quiesced: No transfer operation is in progress and future transfers are disabled.
- Quiescing: A transfer operation is in progress and future transfers are disabled.
- Checking: Destination volume is undergoing a diagnostic check, no transfer is in progress, and future transfers are not disabled. Only for relationships with "Relationship Capability" of "Pre 8.2" .
- Breaking: The SnapMirror relationship is being broken off and no transfer is in progress.

The following values are only applicable to relationships with a policy of type sync-mirror, strict-sync-mirror, automated-failover, or automated-failover-duplex:

- OutOfSync: The SnapMirror relationship is not InSync and no async transfer operation is in progress.
- Transitioning: The SnapMirror relationship is transitioning to InSync.
- InSync: The SnapMirror relationship is InSync.

The following values are only applicable to relationships with a policy of type automated-failover:

- Modifying: One or more properties of the SnapMirror relationship are being modified.

Mirror State: State of the destination volume. Can be one of the following:

- Uninitialized: Destination volume has not been initialized.
- Snapmirrored: Destination volume has been initialized and is ready to receive SnapMirror updates.
- Broken-off: Destination volume is RW and snapshots are present.

----

The following values are applicable only to the relationships with a policy of type automated-failover or automated-failover-duplex:

- Expanding: Adding new volumes to the SnapMirror relationship is in progress.

----

Healthy: Condition of the relationship. Can be one of the following:

- true: The SnapMirror relationship is healthy. It has not missed a



scheduled transfer, or experienced a manual update failure.

- false: The SnapMirror relationship is not healthy. It has missed a scheduled transfer, or has experienced a manual update failure.

Unhealthy Reason: Reason the SnapMirror relationship is not healthy. Only for relationships with "Relationship Capability" of "8.2 and above"

Newest Snapshot: Name of the newest Snapshot copy on the destination volume.

Newest Snapshot Timestamp: Timestamp of the newest Snapshot copy.

Exported Snapshot: Name of the exported Snapshot copy on the destination volume. This field is not applicable to the policy of type automated-failover-duplex .

Exported Snapshot Timestamp: Timestamp of the exported Snapshot copy.

Lag Time: Time since the exported Snapshot copy was created. It is displayed in the format: hours:minutes:seconds. Only for relationships with "Relationship Capability" of "8.2 and above" .

Transfer Type: Type of the current transfer operation. Can be one of the following:

- initialize
- update
- resync
- restore

Only for relationships with "Relationship Capability" of "8.2 and above" .

Transfer Snapshot: Name of the Snapshot copy being transferred.

Snapshot Progress: Amount of data transferred for the transfer snapshot. This parameter is not supported for SnapMirror FlexGroup relationships, but it is supported for FlexGroup constituent relationships.

Total Progress: Total amount of data transferred for the current transfer operation.

Percent Complete for Current Status: Percent complete for the current value

of status. This field is only valid when the "Relationship Status" is "Finalizing" .

**Network Compression Ratio:** The compression ratio achieved for the data sent over the wire as a part of the current transfer operation. The ratio is not maintained across checkpoint restarts. If network compression is disabled for the transfer, the ratio will be set to 1:1. Only for relationships with "Relationship Capability" of "8.2 and above" . This parameter is not supported for Vserver or FlexGroup SnapMirror relationships, but it is supported for FlexGroup constituent relationships.

**Snapshot Checkpoint:** The amount of data transferred as recorded in most the restart checkpoint of the current or recent transfer snapshot. If a restart checkpoint is present the next transfer will continue from the checkpoint. This parameter is not supported for SnapMirror FlexGroup relationships, but it is supported for FlexGroup constituent relationships.

**Transfer Error:** Possible transient error condition if any, encountered by the current transfer operation. Only for relationships with "Relationship Capability" of "8.2 and above".

**Current Throttle:** The maximum transfer rate in Kilobytes per second, used for the current transfer between clusters. Only for relationships with "Relationship Capability" of "8.2 and above" .

**Current Transfer Priority:** Priority assigned to the current transfer. Possible values are:  
- low  
- normal  
Only for relationships with "Relationship Capability" of "8.2 and above" .

**Last Transfer Type:** Type of the previous transfer operation:  
- initialize  
- update  
- resync

- restore

Only for relationships with  
"Relationship Capability"  
of "8.2 and above" .

Last Transfer Size: Total amount of data transferred during the  
previous transfer operation if it was  
successful.

Only for relationships with  
"Relationship Capability"  
of "8.2 and above" .

This parameter is not supported for  
SnapMirror FlexGroup relationships, but it  
is supported for FlexGroup constituent  
relationships.

Last Transfer Network Compression Ratio: The compression ratio achieved  
for the data sent over the wire as a part of  
the previous transfer operation. If network  
compression was disabled for the transfer,  
the ratio will be set to 1:1.

Only for relationships with  
"Relationship Capability"  
of "8.2 and above" .

This parameter is not supported for Vserver  
or FlexGroup SnapMirror relationships, but

it

is supported for FlexGroup constituent  
relationships.

Last Transfer Duration: Duration of the previous transfer  
operation if it was successful.

Only for relationships with  
"Relationship Capability"  
of "8.2 and above" .

Last Transfer From: Source endpoint of the previous transfer  
operation.

Only for relationships with  
"Relationship Capability"  
of "8.2 and above" .

Last Transfer End Timestamp: Timestamp of the end of the previous  
transfer operation.

Only for relationships with  
"Relationship Capability"  
of "8.2 and above" .

Last Transfer Error: Cause of the failure of the previous  
transfer operation.

Only for relationships with  
"Relationship Capability"

of "8.2 and above" .

Relationship Capability: Management and control compatibility:

- "Pre 8.2": Management and control of the relationship is compatible with clustered Data ONTAP 8.1.
- "8.2 and above": Full support of clustered Data ONTAP 8.2 or later SnapMirror relationship management and control.

This parameter is not supported for Vserver SnapMirror relationships.

Relationship ID: The unique identifier of the relationship.

Only for relationships with

"Relationship Capability"

of "8.2 and above" .

This parameter is not supported for Vserver SnapMirror relationships.

Current Operation ID: Operation unique identifier of the currently executing SnapMirror operation.

Only for relationships with

"Relationship Capability"

of "8.2 and above" .

Throttle (KB/sec): Configured maximum transfer rate for cross-cluster transfers.

SnapMirror Policy Type: Type of the SnapMirror policy associated with the relationship. Can be one of the following:

- async-mirror
- vault
- mirror-vault

Refer to the man page for the

`xref:{relative_path}snapmirror-policy-create.html[snapmirror policy create]` command

for a description of what these types mean.

Only for relationships with

"Relationship Capability"

of "8.2 and above" .

SnapMirror Policy: Name of the SnapMirror policy associated with the relationship.

Only for relationships with

"Relationship Capability"

of "8.2 and above" .

SnapMirror Schedule: Name of the schedule (empty if there is no schedule) associated with the relationship.

Tries Limit: Maximum number of times a transfer will be tried.

Only for relationships with "Relationship Capability" of "Pre 8.2" .

This parameter is not supported for Vserver SnapMirror relationships.

Destination Volume Node: Node which owns the destination volume of the relationship. For FlexGroup relationships it is the node which owns the root constituent destination volume.

For object store relationships it is the node which owns the source volume.

Only for relationships with "Relationship Capability" of "8.2 and above" .

This parameter is not supported for Vserver SnapMirror relationships.

Identity Preserve Vserver DR: Whether or not the identity of the source Vserver is replicated to the destination Vserver. Can be:

- true: Source Vserver's configuration will additionally be replicated to the destination, along with the Vserver's volumes and RBAC configuration.
- false: Only volumes and RBAC configuration of the source Vserver is replicated to the destination.

This parameter is supported only for Vserver SnapMirror relationships.

Volume MSIDs Preserved: Whether or not the MSIDs of the source volumes are retained while creating destination volumes.

Can be:

- true: MSIDs of source Vserver volumes and destination Vserver volumes match.
- false: MSIDs of source Vserver volumes

and

destination Vserver volumes do not match.

This parameter is supported only for Vserver SnapMirror relationships.

Is Auto Expand Enabled: Whether or not the auto expand is enabled.

Can be:

- true: Auto Expand is enabled.

- false: Auto Expand is disabled.

This parameter is supported only for FlexGroup SnapMirror relationships.

Is Adaptive Enabled: Whether or not adaptive is enabled.

Can be:

- true: Adaptive is enabled.

- false: Adaptive is disabled.

This parameter is supported only for FlexVol SnapMirror relationships between Data ONTAP endpoints.

Backoff level: Level of backoff of SnapMirror relationship transfers in the presence of client ops.

Can be:

- high

- medium

- none

This parameter is supported only for FlexVol SnapMirror relationships.

Number of Successful Updates: The number of successful SnapMirror update operations for the relationship since the last reboot, aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship.

Only for relationships with "Relationship Capability" of "8.2 and above" .

This parameter is not supported for Vserver SnapMirror relationships.

Number of Failed Updates: The number of failed SnapMirror update operations for the relationship since the last reboot, aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship.

Only for relationships with "Relationship Capability" of "8.2 and above" .

This parameter is not supported for Vserver SnapMirror relationships.

Number of Successful Resyncs: The number of successful SnapMirror resync operations for the relationship since the last reboot, aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship.

Only for relationships with "Relationship Capability" of "8.2 and above" .

This parameter is not supported for Vserver SnapMirror relationships.

**Number of Failed Resyncs:** The number of failed SnapMirror resync operations for the relationship since the last reboot, aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship.

Only for relationships with "Relationship Capability" of "8.2 and above" .

This parameter is not supported for Vserver SnapMirror relationships.

**Number of Successful Breaks:** The number of successful SnapMirror break operations for the relationship since the last reboot, aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship.

Only for relationships with "Relationship Capability" of "8.2 and above" .

This parameter is not supported for Vserver SnapMirror relationships.

**Number of Failed Breaks:** The number of failed SnapMirror break operations for the relationship since the last reboot, aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship.

Only for relationships with "Relationship Capability" of "8.2 and above" .

This parameter is not supported for Vserver SnapMirror relationships.

**Total Transfer Bytes:** Cumulative bytes transferred for the relationship. Only for relationships with "Relationship Capability" of "8.2 and above" .

This parameter is not supported for Vserver SnapMirror relationships.

**Total Transfer Time:** Cumulative total transfer time in seconds for the relationship since the last reboot, aggregate relocation, takeover/giveback, or metrocluster switchover/switchback involving the node that hosts the relationship. Only for relationships with

"Relationship Capability" of "8.2 and above" .

This parameter is not supported for Vserver

## Parameters

**{ [-fields <fieldname>,...]**

If you specify the `-fields <fieldname>`, ... parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays detailed information about all fields.

**{ [-S, -source-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}} - Source Path**

Select SnapMirror relationships that have a matching source path name.

**| [-source-cluster <Cluster name>] - Source Cluster**

Select SnapMirror relationships that have a matching source cluster name. This parameter is not supported for relationships with non-Data ONTAP, FlexGroup volume, or Vserver source endpoints.

**[-source-vserver <vserver name>] - Source Vserver**

Select SnapMirror relationships that have a matching source Vserver name. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

**[-source-volume <volume name>] - Source Volume }**

Select SnapMirror relationships that have a matching source volume name. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

**{ [-destination-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}} - Destination Path**

Select SnapMirror relationships that have a matching destination path name.



Using wildcards with this parameter:

- To match all Vserver Snapmirror relationships, use: ``-destination-path` *`` :
- To match all the Snapmirror relationships except Vserver Snapmirror relationships in the cluster, use: ``-destination-path` *``

**| [-destination-cluster <Cluster name>] - Destination Cluster**

Select SnapMirror relationships that have a matching destination cluster name. This parameter is not supported for relationships with non-Data ONTAP, FlexGroup volume, or Vserver destination endpoints.

**[-destination-vserver <vserver name>] - Destination Vserver**

Select SnapMirror relationships that have a matching destination Vserver name. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.



**[-destination-volume <volume name>] - Destination Volume }**

Select SnapMirror relationships that have a matching destination volume name. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

**[-type <snapmirrorType>] - Relationship Type**

Select SnapMirror relationships that have a matching relationship type. Vservers support only DP SnapMirror relationships. Possible values are:

- DP
- LS
- XDP
- TDP
- RST

**[-relationship-group-type**

**{none|vserver|consistencygroup|flexgroup|vserverflexgroup|vserverconsistencygroup  
}] - Relationship Group Type**

Select SnapMirror relationships that have a matching relationship group type. Possible values are:

- none
- vserver
- flexgroup
- consistencygroup

**[-vserver <vserver name>] - Managing Vserver**

Select SnapMirror relationships that have a matching managing Vserver name. The `-vserver` option is currently a reserved option.

**[-schedule <text>] - SnapMirror Schedule**

Select SnapMirror relationships that have a matching schedule.

**[-policy-type {vault|async-mirror|mirror-vault|strict-sync-mirror|sync-mirror|automated-failover|automated-failover-duplex|continuous}] - SnapMirror Policy Type**

Selects SnapMirror relationships that have a matching SnapMirror policy type. Possible values are:

- async-mirror
- vault
- mirror-vault
- automated-failover
- automated-failover-duplex
- sync-mirror
- strict-sync-mirror

**[-policy <sm\_policy>] - SnapMirror Policy**

Select SnapMirror relationships that have a matching SnapMirror policy.

**[-tries <integer\_or\_unlimited>] - Tries Limit**

Select SnapMirror relationships that have a matching tries limit.

**[-k, -throttle <throttleType>] - Throttle (KB/sec)**

Select SnapMirror relationships that have a matching throttle.

**[-cg-item-mappings {<source volume>:@<destination volume>|<source item path>:@<destination item path>}] - Consistency Group Item Mappings**

Select SnapMirror relationships that have a matching cg-item-mappings.

**[-current-throttle <throttleType>] - Current Transfer Throttle (KB/sec)**

Select SnapMirror relationships that have a matching current throttle.

**[-state <mirror state>] - Mirror State**

Select SnapMirror relationships that have a matching mirror state. Possible values are:

- Uninitialized
- Snapmirrored
- Broken-off

**[-status <mirror status>] - Relationship Status**

Select SnapMirror relationships that have a matching relationship status. Possible values are:

- Idle
- Queued
- Transferring
- Preparing
- Finalizing
- Aborting
- Quiesced
- Quiescing
- Checking

Status values Finalizing, Checking, Waiting and Preparing are not supported for Vserver SnapMirror relationships.

**[-file-restore-file-count <integer>] - File Restore File Count**

The number of files being restored by file restore.

**[-file-restore-file-list <text>,...] - File Restore File List**

List of the destination file names of the files being restored by file restore.

**[-transfer-snapshot <text>] - Transfer Snapshot**

Select SnapMirror relationships that have a matching transfer Snapshot copy.

**[--snapshot-progress {<integer>[KB|MB|GB|TB|PB]}] - Snapshot Progress**

Select SnapMirror relationships that have a matching Snapshot progress.

**[--total-progress {<integer>[KB|MB|GB|TB|PB]}] - Total Progress**

Select SnapMirror relationships that have a matching total progress.

**[--network-compression-ratio <text>] - Network Compression Ratio**

Select SnapMirror relationships that have a matching network compression ratio. This parameter is not supported for Vserver SnapMirror relationships.

**[--snapshot-checkpoint {<integer>[KB|MB|GB|TB|PB]}] - Snapshot Checkpoint**

Select SnapMirror relationships that have a matching Snapshot copy checkpoint. This parameter is not supported for Vserver SnapMirror relationships.

**[--newest-snapshot <text>] - Newest Snapshot**

Select SnapMirror relationships that have a matching newest Snapshot copy.

**[--newest-snapshot-timestamp <MM/DD HH:MM:SS>] - Newest Snapshot Timestamp**

Select SnapMirror relationships that have a matching newest Snapshot copy timestamp.

**[--exported-snapshot <text>] - Exported Snapshot**

Select SnapMirror relationships that have a matching exported Snapshot copy name. For load-sharing mirror relationships, if the exported-snapshot field for a relationship has a dash (-), the load-sharing mirror is lagging behind the up-to-date mirrors in the set.

**[--exported-snapshot-timestamp <MM/DD HH:MM:SS>] - Exported Snapshot Timestamp**

Select SnapMirror relationships that have a matching exported Snapshot copy timestamp.

**[--healthy {true|false}] - Healthy**

Select SnapMirror relationships that have a matching healthy condition.

**[--relationship-id <UUID>] - Relationship ID**

Select SnapMirror relationships that have a matching relationship ID. This parameter is not supported for Vserver SnapMirror relationships.

**[--current-operation-id <UUID>] - Current Operation ID**

Select SnapMirror relationships that have a matching operation unique identifier of the currently executing SnapMirror operation.

**[--current-transfer-type**

**{initialize|update|resync|restore|check|file\_restore|cggrs\_initialize|cggrs\_resync|cg\_update|cg\_initialize|cg\_resync|cg\_restore|catalog\_metadata|cg\_file\_restore|failover|aufo|failover\_incapable|cg\_migrate}] - Transfer Type**

Select SnapMirror relationships that have a matching current transfer type.

**[--current-transfer-error <text>] - Transfer Error**

Select SnapMirror relationships that have a matching current transfer error.

**[-last-transfer-type {initialize|update|resync|restore|check|file\_restore|cggrs\_initialize|cggrs\_resync|cg\_update|cg\_initialize|cg\_resync|cg\_restore|catalog\_metadata|cg\_file\_restore|failover|aufo|failover\_incapable|cg\_migrate}] - Last Transfer Type**

Select SnapMirror relationships that have a matching last transfer type.

**[-last-transfer-error <text>] - Last Transfer Error**

Select SnapMirror relationships that have a matching last transfer error.

**[-last-transfer-size {<integer>[KB|MB|GB|TB|PB]}] - Last Transfer Size**

Select SnapMirror relationships that have a matching last transfer size.

**[-last-transfer-network-compression-ratio <text>] - Last Transfer Network Compression Ratio**

Select SnapMirror relationships that have a matching last transfer network compression ratio. This parameter is not supported for Vserver SnapMirror relationships.

**[-last-transfer-duration <[[<hours>:]<minutes>:]<seconds>>] - Last Transfer Duration**

Select SnapMirror relationships that have a matching last transfer duration.

**[-last-transfer-from <text>] - Last Transfer From**

Select SnapMirror relationships that have a matching last transfer source.

**[-last-transfer-end-timestamp <MM/DD HH:MM:SS>] - Last Transfer End Timestamp**

Select SnapMirror relationships that have a matching last transfer end timestamp.

**[-unhealthy-reason <text>] - Unhealthy Reason**

Select SnapMirror relationships that have a matching unhealthy reason.

**[-progress-last-updated <MM/DD HH:MM:SS>] - Progress Last Updated**

Select SnapMirror relationships that have a matching progress last updated.

**[-relationship-capability <text>] - Relationship Capability**

Select SnapMirror relationships that have a matching relationship capability. This parameter is not supported for Vserver SnapMirror relationships.

**[-lag-time <[[<hours>:]<minutes>:]<seconds>>] - Lag Time**

Select SnapMirror relationships that have a matching lag time.

**[-current-transfer-priority {low|normal}] - Current Transfer Priority**

Select SnapMirror relationships that have a matching current transfer priority.

**[-is-smtape-op {true|false}] - SMTape Operation**

Select SnapMirror relationships that have a matching smtape operation.

**[-destination-volume-node <nodename>] - Destination Volume Node Name**

Select SnapMirror relationships that have a matching destination volume node name. This parameter is not supported for Vserver SnapMirror relationships.

**[~~-identity-preserve~~ {true|false}] - Identity Preserve Vserver DR**

Select SnapMirror relationships that have a matching value for identity-preserve. This parameter is valid only for Vserver SnapMirror relationships.

**[~~-expand~~ <true>] - Show Constituents of the Group**

Specifies whether to display constituent relationships of Vserver, FlexGroup and Consistency Group SnapMirror relationships. By default, the constituents are not displayed.

**[~~-update-successful-count~~ <integer>] - Number of Successful Updates**

Select SnapMirror relationships that have a matching number of successful updates. This parameter is not supported for Vserver SnapMirror relationships.

**[~~-update-failed-count~~ <integer>] - Number of Failed Updates**

Select SnapMirror relationships that have a matching number of failed updates. This parameter is not supported for Vserver SnapMirror relationships.

**[~~-resync-successful-count~~ <integer>] - Number of Successful Resyncs**

Select SnapMirror relationships that have a matching number of successful resyncs. This parameter is not supported for Vserver SnapMirror relationships.

**[~~-resync-failed-count~~ <integer>] - Number of Failed Resyncs**

Select SnapMirror relationships that have a matching number of failed resyncs. This parameter is not supported for Vserver SnapMirror relationships.

**[~~-break-successful-count~~ <integer>] - Number of Successful Breaks**

Select SnapMirror relationships that have a matching number of successful breaks. This parameter is not supported for Vserver SnapMirror relationships.

**[~~-break-failed-count~~ <integer>] - Number of Failed Breaks**

Select SnapMirror relationships that have a matching number of failed breaks. This parameter is not supported for Vserver SnapMirror relationships.

**[~~-total-transfer-bytes~~ <integer>] - Total Transfer Bytes**

Select SnapMirror relationships that have a matching total transfer bytes. This parameter is not supported for Vserver SnapMirror relationships.

**[~~-total-transfer-time-secs~~ <integer>] - Total Transfer Time in Seconds**

Select SnapMirror relationships that have a matching total transfer time in seconds. This parameter is not supported for Vserver SnapMirror relationships.

**[~~-msid-preserve~~ {true|false}] - Source Volume MSIDs Preserved**

This parameter specifies whether the volume MSIDs are preserved at the destination. This parameter is applicable only for Vserver SnapMirror relationships.

**[~~-is-auto-expand-enabled~~ {true|false}] - Is Auto Expand Enabled**

Select SnapMirror relationships that have a matching value for auto expand. This parameter is supported only for FlexGroup SnapMirror relationships. Possible values are:

- true
- false

### **[`-percent-complete-cur-status <integer>`] - Percent Complete for Current Status**

Select SnapMirror relationships that have a matching percent complete for the current status.

### **[`-backoff-level {high|medium|none}`] - SM Backoff Level due to Client Ops**

Select SnapMirror relationships that have a matching value for backoff level. This parameter is supported only for FlexVol SnapMirror relationships. Possible values are:

- high
- medium
- none

## **Examples**

The example below displays summary information for all SnapMirror relationships with destination endpoints in the current cluster:

```
cluster2::> snapmirror show
Source          Destination  Mirror  Relationship  Total
Last
Path           Type  Path          State  Status        Progress  Healthy
Updated
-----
-----
cluster1-vs2.example1.com:
      DP  cluster2-dvs2.example2.com:
                Snapmirrored
                Idle          -          true  -
cluster2-vs1.example.com:dp_src1
      DP  cluster2-vs2.example.com:dp_dst1
                Snapmirrored
                Idle          -          true  -
cluster2-vs1.example.com:xdp_src1
      XDP cluster2-vs2.example.com:xdp_dst1
                Snapmirrored
                Idle          -          true  -
cluster2://cluster2-vs1.example.com/ls_src1
      LS  cluster2://cluster2-vs1.example.com/ls_mr1
                Snapmirrored
                Idle          -          true  -
                cluster2://cluster2-vs1.example.com/ls_mr2
                Snapmirrored
                Idle          -          true  -

5 entries were displayed.
```

The example below displays detailed information for the SnapMirror relationship with the destination endpoint `cluster2-vs2.example.com:dp_dst1`.

```
cluster2::> snapmirror show -destination-path cluster2-
vs2.example.com:dp_dst1
Source Path: cluster2-vs1.example.com:dp_src1
      Destination Path: cluster2-vs2.example.com:dp_dst1
      Relationship Type: DP
Relationship Group Type: none
      SnapMirror Schedule: -
      SnapMirror Policy Type: async-mirror
      SnapMirror Policy: DPDefault
      Tries Limit: -
      Throttle (KB/sec): unlimited
      Mirror State: Snapmirrored
      Relationship Status: Idle
      Transfer Snapshot: -
      Snapshot Progress: -
      Total Progress: -
Percent Complete for Current Status: -
      Network Compression Ratio: -
      Snapshot Checkpoint: -
      Newest Snapshot: snapmirror.3d19af37-8f5e-11e1-
8c83-123478563412_2147484676.2012-04-27_025137
      Newest Snapshot Timestamp: 04/27 02:51:42
      Exported Snapshot: snapmirror.3d19af37-8f5e-11e1-
8c83-123478563412_2147484676.2012-04-27_025137
      Exported Snapshot Timestamp: 04/27 02:51:42
      Healthy: true
      Unhealthy Reason: -
      Destination Volume Node: cluster2-node1
      Relationship ID: cdc70a81-8f5f-11e1-8392-
123478563412
      Current Operation ID: -
      Transfer Type: -
      Transfer Error: -
      Current Throttle: -
      Current Transfer Priority: -
      Last Transfer Type: update
      Last Transfer Error: -
      Last Transfer Size: 530.2MB
Last Transfer Network Compression Ratio: 111.7:1
      Last Transfer Duration: 0:2:53
      Last Transfer From: cluster2-vs1.example.com:dp_src1
Last Transfer End Timestamp: 04/27 02:51:45
      Progress Last Updated: -
      Relationship Capability: 8.2 and above
      Lag Time: 133:50:40
```

```

Identity Preserve Vserver DR: -
    Volume MSIDs Preserved: -
Is Auto Expand Enabled: -
    Is Adaptive: -
Number of Successful Updates: 1
    Number of Failed Updates: 0
Number of Successful Resyncs: 0
    Number of Failed Resyncs: 0
Number of Successful Breaks: 0
    Number of Failed Breaks: 0
        Total Transfer Bytes: 663552
Total Transfer Time in Seconds: 3

```

The example below displays detailed information for SnapMirror relationships with the *Relationship Capability* of "Pre 8.2" source or destination endpoints in the current cluster.

```

cluster2::> snapmirror show -relationship-capability "Pre 8.2" -instance
Source Path: cluster2://cluster2-vs1.example.com/ls_src1
    Destination Path: cluster2://cluster2-
vs1.example.com/ls_mr1
        Relationship Type: LS
    Relationship Group Type: -
        SnapMirror Schedule: -
    SnapMirror Policy Type: -
        SnapMirror Policy: -
            Tries Limit: 8
        Throttle (KB/sec): unlimited
            Mirror State: Snapmirrored
    Relationship Status: Idle
        Transfer Snapshot: -
        Snapshot Progress: -
            Total Progress: -
    Percent Complete for Current Status: -
        Network Compression Ratio: -
            Snapshot Checkpoint: -
                Newest Snapshot: snapmirror.3d4e52c5-8f5c-11e1-
8392-123478563412_3_2147484684.2012-05-02_163506
                Newest Snapshot Timestamp: 05/02 16:35:06
                Exported Snapshot: snapmirror.3d4e52c5-8f5c-11e1-
8392-123478563412_3_2147484684.2012-05-02_163506
                Exported Snapshot Timestamp: 05/02 16:35:06
                    Healthy: true
                Unhealthy Reason: -
            Destination Volume Node: -
                Relationship ID: -

```



```
Current Operation ID: -
    Transfer Type: -
    Transfer Error: -
    Last Transfer Type: -
    Last Transfer Error: -
    Last Transfer Size: -
Last Transfer Network Compression Ratio: -
    Last Transfer Duration: -
    Last Transfer From: -
    Last Transfer End Timestamp: -
    Progress Last Updated: -
    Relationship Capability: Pre 8.2
        Lag Time: -
        SnapMirror Policy: -
Identity Preserve Vserver DR: -
    Volume MSIDs Preserved: -
Is Auto Expand Enabled: -
    Is Adaptive: -
    Number of Successful Updates: -
    Number of Failed Updates: -
    Number of Successful Resyncs: -
    Number of Failed Resyncs: -
    Number of Successful Breaks: -
    Number of Failed Breaks: -
    Total Transfer Bytes: -
    Total Transfer Time in Seconds: -
Source Path: cluster2://cluster2-vs1.example.com/ls_src1
    Destination Path: cluster2://cluster2-
vs1.example.com/ls_mr2
    Relationship Type: LS
    Relationship Group Type: -
    SnapMirror Schedule: -
    SnapMirror Policy Type: -
    SnapMirror Policy: -
    Tries Limit: 8
    Throttle (KB/sec): unlimited
    Mirror State: Snapmirrored
    Relationship Status: Idle
    Transfer Snapshot: -
    Snapshot Progress: -
    Total Progress: -
    Percent Complete for Current Status: -
    Network Compression Ratio: -
    Snapshot Checkpoint: -
    Newest Snapshot: snapmirror.3d4e52c5-8f5c-11e1-
8392-123478563412_3_2147484684.2012-05-02_163506
```

```

Newest Snapshot Timestamp: 05/02 16:35:06
    Exported Snapshot: snapmirror.3d4e52c5-8f5c-11e1-
8392-123478563412_3_2147484684.2012-05-02_163506
    Exported Snapshot Timestamp: 05/02 16:35:06
        Healthy: true
        Unhealthy Reason: -
    Destination Volume Node: -
        Relationship ID: -
    Current Operation ID: -
        Transfer Type: -
        Transfer Error: -
    Last Transfer Type: -
    Last Transfer Error: -
    Last Transfer Size: -
Last Transfer Network Compression Ratio: -
    Last Transfer Duration: -
        Last Transfer From: -
    Last Transfer End Timestamp: -
        Progress Last Updated: -
    Relationship Capability: Pre 8.2
        Lag Time: -
        SnapMirror Policy: -
Identity Preserve Vserver DR: -
    Volume MSIDs Preserved: -
Is Auto Expand Enabled: -
    Is Adaptive: -
    Number of Successful Updates: -
        Number of Failed Updates: -
    Number of Successful Resyncs: -
        Number of Failed Resyncs: -
    Number of Successful Breaks: -
        Number of Failed Breaks: -
        Total Transfer Bytes: -
    Total Transfer Time in Seconds: -

```

2 entries were displayed.

The example below displays detailed information for the Vserver SnapMirror relationship with the destination endpoint `cluster2-dvs2.example2.com`:

```

cluster2::> snapmirror show -destination-path cluster2-dvs2.example2.com:
Source Path: cluster1-vs2.example1.com:
    Destination Path: cluster2-dvs2.example2.com:
    Relationship Type: DP
    Relationship Group Type: -

```

```
    SnapMirror Schedule: -
    SnapMirror Policy Type: async-mirror
    SnapMirror Policy: DPDefault
    Tries Limit: -
    Throttle (KB/sec): unlimited
    Mirror State: Snapmirrored
    Relationship Status: Idle
    File Restore File Count: -
    File Restore File List: -
    Transfer Snapshot: -
    Snapshot Progress: -
    Total Progress: -
    Percent Complete for Current Status: -
    Network Compression Ratio: -
    Snapshot Checkpoint: -
    Newest Snapshot: vserverdr.1d519e9c-7838-11e3-
91fb-123478563412.2014-01-13_110707
    Newest Snapshot Timestamp: 01/13 11:07:07
    Exported Snapshot: vserverdr.1d519e9c-7838-11e3-
91fb-123478563412.2014-01-13_110707
    Exported Snapshot Timestamp: 01/13 11:07:07
    Healthy: true
    Unhealthy Reason: -
    Destination Volume Node: -
    Relationship ID: -
    Operation ID: -
    Transfer Type: -
    Transfer Error: -
    Current Throttle: -
    Current Transfer Priority: -
    Last Transfer Type: resync
    Last Transfer Error: -
    Last Transfer Size: -
    Last Transfer Network Compression Ratio: -
    Last Transfer Duration: -
    Last Transfer From: cluster1-vs2.example1.com:
    Last Transfer End Timestamp: -
    Progress Last Updated: -
    Relationship Capability: -
    Lag Time: 18:47:9
    Identity Preserve Vserver DR: false
    Volume MSIDs Preserved: true
    Is Auto Expand Enabled: -
    Is Adaptive: -
    Number of Successful Updates: -
    Number of Failed Updates: -
```

```
Number of Successful Resyncs: -
Number of Failed Resyncs: -
Number of Successful Breaks: -
Number of Failed Breaks: -
Total Transfer Bytes: -
Total Transfer Time in Seconds: -
```

The following example displays detailed information for the SnapMirror relationship with the AltaVault destination endpoint `10.0.0.11:/share/share1`:

```
cluster2::> snapmirror show -destination-path 10.0.0.11:/share/share1
Source Path: cluster2-vs1.example.com:data_ontap_vol
Destination Path: 10.0.0.11:/share/share1
Relationship Type: XDP
Relationship Group Type: none
SnapMirror Schedule: -
SnapMirror Policy Type: vault
SnapMirror Policy: XDPDefault
Tries Limit: -
Throttle (KB/sec): unlimited
Mirror State: Snapmirrored
Relationship Status: Idle
Transfer Snapshot: -
Snapshot Progress: -
Total Progress: -
Percent Complete for Current Status: -
Network Compression Ratio: -
Snapshot Checkpoint: -
Newest Snapshot: snapmirror.3d19af37-8f5e-11e1-8c83-
123478563412_2147484676.2012-04-27_025137
Newest Snapshot Timestamp: 04/27 02:51:42
Exported Snapshot: snapmirror.3d19af37-8f5e-11e1-8c83-
123478563412_2147484676.2012-04-27_025137
Exported Snapshot Timestamp: 04/27 02:51:42
Healthy: true
Unhealthy Reason: -
Destination Volume Node: -
Relationship ID: cdc70a81-8f5f-11e1-8392-123478563463
Current Operation ID: -
Transfer Type: -
Transfer Error: -
Current Throttle: -
Current Transfer Priority: -
Last Transfer Type: update
Last Transfer Error: -
```

```
    Last Transfer Size: 530.2MB
Last Transfer Network Compression Ratio: 1:1
    Last Transfer Duration: 0:2:53
        Last Transfer From: cluster2-vs1.example.com:data_ontap_vol
    Last Transfer End Timestamp: 04/27 02:51:45
    Progress Last Updated: -
Relationship Capability: 8.2 and above
    Lag Time: 133:50:40
Identity Preserve Vserver DR: -
Volume MSIDs Preserved: -
    Is Auto Expand Enabled: -
        Is Adaptive: -
Number of Successful Updates: 1
    Number of Failed Updates: 0
Number of Successful Resyncs: 0
    Number of Failed Resyncs: 0
    Number of Successful Breaks: 0
Number of Failed Breaks: 0
    Total Transfer Bytes: 663552
Total Transfer Time in Seconds: 3
```

The example shows the usage of the `-expand` parameter to additionally display the constituents of Vserver SnapMirror relationships with destination endpoints in the current cluster. Note that in the following example, since there is no volume level relationship for the root volume of a Vserver, it is not shown in the output:

```
cluster2::> snapmirror show -expand
```

```
Progress
```

```
Source          Destination  Mirror  Relationship  Total
Last
Path           Type  Path          State  Status          Progress  Healthy
Updated
```

```
-----
-----
cluster1-vs1.example1.com:
      DP  cluster2-dvs1.example2.com:
              Snapmirrored
              Idle          -          true  -
cluster1-vs1.example1.com:vol1
      DP  cluster2-dvs1.example2.com:vol1
              Snapmirrored
              Idle          -          true  -
cluster1-vs2.example1.com:
      DP  cluster2-dvs2.example2.com:
              Snapmirrored
              Idle          -          true  -
cluster1-vs2.example1.com:vol1
      DP  cluster2-dvs2.example2.com:vol1
              Snapmirrored
              Idle          -          true  -
4 entries were displayed.
```

## Related Links

- [snapmirror list-destinations](#)
- [snapmirror policy create](#)

## snapmirror update-ls-set

Start an incremental load-sharing set transfer

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### Description

The `snapmirror update-ls-set` command updates a set of load-sharing mirrors. The command makes destination volumes, in the group of load-sharing mirrors, up-to-date mirrors of the source volume.

The key parameter that identifies the set of load-sharing mirrors is the source volume. SnapMirror transfers are performed from the source volume to each of the up-to-date destination volumes in the set of load-sharing mirrors.

The `snapmirror update-ls-set` command performs an incremental transfer to each of the destination volumes. During an incremental transfer, Data ONTAP takes a Snapshot copy on the source volume to capture the current image of the source volume, finds the most recent common Snapshot copy between the source and destination volumes, and incrementally transfers Snapshot copies that are newer than the common Snapshot copy to the destination volume.



You still need to use the `snapmirror update-ls-set` command to manually update the set of load-sharing mirrors even if the set only has one destination mirror. The `snapmirror update` command can only be used to bring up to date a specific destination mirror that is lagging to the set.

After an update using the `snapmirror update-ls-set` command successfully completes, the last Snapshot copy transferred is made the new exported Snapshot copy on the destination volumes.

This command is only supported for SnapMirror relationships with the field *"Relationship Capability"* showing as *"Pre 8.2"* in the output of the `snapmirror show` command.

## Parameters

**{ -S, -source-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Source Path**

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or the volume (volume). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (cluster) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form `hostip:/share/share-name`. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form `hostip:/lun/name`.

**[ -source-cluster <Cluster name>] - Source Cluster**

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source -vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

**-source-vserver <vserver name> - Source Vserver**

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

**-source-volume <volume name> - Source Volume }**

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

## **[`-w`, `-foreground <true>`] - Foreground Process**

This specifies whether the operation runs as a foreground process. If this parameter is specified, the default setting is `true` (the operation runs in the foreground). When set to `true`, the command will not return until the process completes. This parameter is only applicable to relationships with "*Relationship Capability*" of "*Pre 8.2*".

## **Examples**

To update the group of load-sharing mirrors for the source endpoint named `//vs1.example.com/vs1_root`, type the following command:

```
cluster1::> snapmirror update-ls-set -source-path
//vs1.example.com/vs1_root
```

## **Related Links**

- [snapmirror update](#)
- [snapmirror show](#)

# **snapmirror update**

Start an incremental transfer

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## **Description**

The `snapmirror update` command updates the destination volume or non-Data ONTAP endpoint of a SnapMirror relationship. The `snapmirror update` command behaves differently for data protection (DP), extended data protection (XDP) and load-sharing (LS) relationships. Refer to the `-type` parameter of the [snapmirror create](#) command to understand different types of relationships supported by SnapMirror.

The `snapmirror update` command performs an incremental transfer.

Before using this command, the relationship must be initialized using the [snapmirror initialize](#) or [snapmirror initialize-ls-set](#) commands.

For data protection SnapMirror relationships with volumes as endpoints, the `snapmirror update` command makes the destination volume an up-to-date mirror of the source volume with the following steps:

- If the source volume is read-write, takes a Snapshot copy on the source volume to capture the current image of the source volume
- Finds the most recent Snapshot copy on the destination volume and validates that the corresponding Snapshot copy is still present on the source
- Incrementally transfers Snapshot copies that are newer than the corresponding Snapshot copy to the destination volume

You can use the `snapmirror update` command to update a specific load-sharing mirror that lags behind up-to-date destination volumes in the set of load-sharing mirrors. An update to the lagging load-sharing mirror



should bring it up to date with the other up-to-date destination volumes in the set of load-sharing mirrors.



Using the `snapmirror update` command to update a set of load-sharing mirrors will not work. Use the `snapmirror update-ls-set` command to update a set of load-sharing mirrors.

For extended data protection (XDP) relationships with a `snapmirror` policy of type `async-mirror`, a `snapmirror update` always creates a new Snapshot copy on the source volume. Depending on the rules in the policy, the command might transfer just the newly created Snapshot copy or all Snapshot copies that are newer than the common Snapshot copy including the newly created Snapshot copy to the destination volume.

For extended data protection (XDP) relationships with a `snapmirror` policy of type `vault`, a `snapmirror update` does not create a new Snapshot copy on the source volume but transfers only selected Snapshot copies that are newer than the common Snapshot copy to the destination volume. (Those older than the common copy can be transferred by using the `-source-snapshot` parameter.) Snapshot copies are selected by matching the value of `-snapmirror-label` of a Snapshot copy with the value of `-snapmirror-label` of one of the rules from the corresponding SnapMirror policy associated with the SnapMirror relationship. All matching Snapshot copies are incrementally transferred to the destination volume.

For extended data protection (XDP) relationships with a `snapmirror` policy of type `mirror-vault`, a `snapmirror update` always creates a new Snapshot copy on the source volume and transfers only selected Snapshot copies that are newer than the common snapshot copy. The newly created Snapshot copy is always selected.

For extended data protection (XDP) relationships with a `snapmirror` policy of type `vault` or `mirror-vault`, the `snapmirror update` command also manages expiration of Snapshot copies on the destination volume. It does so by deleting Snapshot copies that have exceeded the value of `-keep` for the matching rule from the corresponding SnapMirror policy associated with the SnapMirror relationship. Snapshot copies that match the same `-snapmirror-label` will be deleted in oldest-first order.

For relationships with a policy of type `strict-sync-mirror` or `sync-mirror`, this command creates a new common Snapshot copy and designates it as the exported Snapshot copy on the destination volume. It updates the destination read-only view because IO is redirected to the new exported Snapshot copy. Clients could experience a brief latency spike during this process as the primary IO is temporarily fenced. This command is allowed only when the relationship-status is InSync. The command retains two pairs of common Snapshot copies and deletes the older ones.

For SnapMirror active sync relationships with Consistency Groups of FlexVol volumes as endpoints, the `snapmirror update` command creates a new coordinated common Snapshot copy. The command retains two coordinated common Snapshot copies and deletes the oldest one.

For data protection relationships, the parameter `-source-snapshot` is optional and only allows for the transfer of Snapshot copies newer than the common Snapshot copy up to the specified `-source-snapshot`.

For extended data protection (XDP) relationships the parameter `-source-snapshot` is optional.

For extended data protection (XDP) relationships with a `snapmirror` policy of type `vault` or `mirror-vault`, the parameter `-source-snapshot` allows transfer of a Snapshot copy that is older than the common Snapshot copy and/or might not be selected for transfer based on policy-based selection of a scheduled update transfer.

For extended data protection (XDP) relationships with a `snapmirror` policy of type `async-mirror`, the `snapmirror update` with parameter `-source-snapshot` does not create a new Snapshot copy on the source volume. Depending on the rules in the policy, the command might transfer just the specified Snapshot

copy or Snapshot copies that are newer than the common Snapshot copy upto and including the specified Snapshot copy to the destination volume.

After the `snapmirror update` command successfully completes, the last Snapshot copy transferred is designated as the new exported Snapshot copy on the destination volume. If an update to an extended data protection (XDP) relationship specifies a Snapshot copy using the `-source-snapshot` parameter that is older than the common snapshot, after the `snapmirror update` successfully completes, the exported Snapshot copy on the destination volume will remain unchanged.

If the `snapmirror update` does not finish successfully—for example, due to a network failure or because a `snapmirror abort` command was issued—a restart checkpoint might be recorded on the destination volume. If a restart checkpoint is recorded, the next update restarts and continues the transfer from the restart checkpoint. For extended data protection (XDP) relationships, the next update will restart and continue the old transfer regardless of whether the Snapshot copy being transferred is a matching Snapshot copy or not.

This command is supported for SnapMirror relationships with the field *"Relationship Capability"* showing as either *"8.2 and above"* or *"Pre 8.2"* in the output of the `snapmirror show` command.

For relationships with *"Relationship Capability"* of *"8.2 and above"*, you can track the progress of the operation using the `snapmirror show` command.

For relationships with *"Relationship Capability"* of *"Pre 8.2"*, a job will be spawned to operate on the SnapMirror relationship, and the job id will be shown in the command output. The progress of the job can be tracked using the `job show` and `job history show` commands.

For Vserver SnapMirror relationships, the `snapmirror update` command makes the destination Vserver an up-to-date mirror of the source Vserver.

The `snapmirror update` command must be used from the destination Vserver or cluster.

## Parameters

**{ [-S, -source-path <[vserver:] [volume]>|<[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<vserver:/cg/[app-cgname]>}] - Source Path**

This parameter specifies the source endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (vserver) and/or the volume (volume). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (cluster) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with an AltaVault source, the source endpoint is specified in the form `hostip:/share/share-name`. For SnapMirror relationships with a SolidFire source, the source endpoint is specified in the form `hostip:/lun/name`.

**| [-source-cluster <Cluster name>] - Source Cluster**

Specifies the source cluster of the SnapMirror relationship. If this parameter is specified, the `-source`, `-vserver` and `-source-volume` parameters must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

### **[`-source-vserver <vserver name>`] - Source Vserver**

Specifies the source Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-source-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

### **[`-source-volume <volume name>`] - Source Volume }**

Specifies the source volume of the SnapMirror relationship. If this parameter is specified, parameters `-source-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-source-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP source endpoints.

### **{ `-destination-path`**

**{`<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>`} - Destination Path**

This parameter specifies the destination endpoint of the SnapMirror relationship in one of four path formats. The normal format includes the names of the Vserver (`vserver`) and/or volume (`volume`). To support relationships with *"Relationship Capability"* of *"Pre 8.2"*, a format which also includes the name of the cluster (`cluster`) is provided. The *"Pre 8.2"* format cannot be used when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*. For SnapMirror relationships with AltaVault destinations, the destination endpoint is specified in the form `hostip:/share/share-name`. For relationships with SolidFire destinations, the destination endpoint is specified in the form `hostip:/lun/name`.

### **[`-destination-cluster <Cluster name>`] - Destination Cluster**

Specifies the destination cluster of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and `-destination-volume` must also be specified. This parameter is only applicable for relationships with *"Relationship Capability"* of *"Pre 8.2"*. This parameter cannot be specified when operating in a Vserver context on relationships with *"Relationship Capability"* of *"8.2 and above"*.

### **`-destination-vserver <vserver name>` - Destination Vserver**

Specifies the destination Vserver of the SnapMirror relationship. For relationships with volumes as endpoints, if this parameter is specified, parameters `-destination-volume` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

### **`-destination-volume <volume name>` - Destination Volume }**

Specifies the destination volume of the SnapMirror relationship. If this parameter is specified, parameters `-destination-vserver` and for relationships with *"Relationship Capability"* of *"Pre 8.2"*, `-destination-cluster` must also be specified. This parameter is not supported for relationships with non-Data ONTAP destination endpoints.

### **[`-s, -source-snapshot <text>`] - Source Snapshot**

This optional parameter specifies a Snapshot copy to transfer. The default behavior for a data protection relationship with a read-write source is that Data ONTAP creates a new Snapshot copy and uses it as the basis for determining what data are replicated; with this option, the specified Snapshot copy will be used instead. The default behavior for an extended data protection relationship depends on the relationship's policy type. For a data protection relationship, the specified Snapshot copy must be newer than the latest common Snapshot copy. For an extended data protection relationship, the specified Snapshot copy can be newer or older than the common Snapshot copy. This parameter is not supported for relationships with

"*Relationship Capability*" of "*Pre 8.2*".

### **[-k, -throttle <throttleType>] - Throttle (KB/sec)**

This optional parameter limits the network bandwidth used for the update transfer. It sets the maximum rate (in Kbytes/sec) at which data can be transferred during the operation. If this parameter is not specified, the throttle value configured for the relationship with the [snapmirror create](#) or [snapmirror modify](#) command will be used. To fully use the network bandwidth available, set the throttle value to *unlimited* or *0*. The minimum throttle value is four Kbytes/sec, so if you specify a throttle value between *1* and *4*, it will be treated as if you specified *4*. For FlexGroup and Application Consistency Group relationships, the throttle value is applied individually to each constituent relationship. The `-throttle` parameter does not affect load-sharing transfers and transfers for other relationships with "*Relationship Capability*" of "*Pre 8.2*" confined to a single cluster.

### **[-transfer-priority {low|normal}] - Transfer Priority**

This optional parameter specifies the priority at which the transfer runs. The default value for this parameter is the value in the SnapMirror policy associated with this relationship. This parameter is not applicable to relationships with a "*Relationship Capability*" of "*Pre 8.2*".

### **[-enable-storage-efficiency <true>] - Enable Storage Efficient Transfers**

This is an optional parameter. For an extended data protection (XDP) relationship that is currently not storage efficient, set this parameter to *true* to enable storage efficient transfers. Storage efficient in this context refers to both over the wire efficiency and how the data is written to the destination volume. The transfer fails if storage efficiency cannot be achieved. If the transfer succeeds, future transfers will continue being storage efficient as long as it is still feasible, but will not fail if the transfer is not storage efficient. The default value is *false*. This parameter is not supported for relationships with non-Data ONTAP endpoints.

### **[-w, -foreground <true>] - Foreground Process**

This specifies whether the operation runs as a foreground process. If this parameter is specified, the default setting is *true* (the operation runs in the foreground). When set to *true*, the command will not return until the process completes. This parameter is only applicable to relationships with "*Relationship Capability*" of "*Pre 8.2*".

## **Examples**

To update the mirror relationship between the destination endpoint

`vs2.example.com:dept_eng_dp_mirror3` and its source endpoint, type the following command:

```
vs2.example.com::> snapmirror update -destination-path
vs2.example.com:dept_eng_dp_mirror3
```

For relationships with "*Relationship Capability*" of "*Pre 8.2*", to update the mirror relationship between the destination endpoint `cluster2://vs2.example.com/dept_eng_dp_mirror3` and its source endpoint, type the following command:

```
cluster2::> snapmirror update -destination-path
cluster2://vs2.example.com/dept_eng_dp_mirror3
```

To update the Vserver SnapMirror relationship between destination endpoint `dvs1.example.com:` and its

source endpoint, type the following command:

```
cluster2::> snapmirror update -destination-path  
dvs1.example.com:
```

To update the SnapMirror active sync relationship with the destination Consistency Group *cg\_dst* in Vserver *vs2.example.com*, type the following command on the destination cluster:

```
destination::> snapmirror update -destination-path  
vs2.example.com:/cg/cg_dst
```

To update the Application Consistency Group relationship with the destination Application Consistency Group *app\_cg\_dst* in Vserver *vs2.example.com*, type the following command on the destination cluster:

```
destination::> snapmirror update -destination-path  
vs2.example.com:/cg/app_cg_dst
```

## Related Links

- [snapmirror create](#)
- [snapmirror initialize](#)
- [snapmirror initialize-ls-set](#)
- [snapmirror update-ls-set](#)
- [snapmirror abort](#)
- [snapmirror show](#)
- [job show](#)
- [job history show](#)
- [snapmirror modify](#)

## snapmirror config-replication commands

### snapmirror config-replication cluster-storage-configuration modify

Modify SnapMirror storage configuration information

**Availability:** This command is available to *cluster* administrators at the *advanced* privilege level.

#### Description

The `snapmirror config-replication cluster-storage-configuration modify` command modifies the configuration of storage used for configuration replication.

## Parameters

**[~~-disallowed-aggregates~~ <aggregate name>,...] - Disallowed Aggregates (privilege: advanced)**

Use this parameter to set the list of storage aggregates that are not available to host storage for configuration replication.

## Examples

The following example disallows two aggregates named *aggr1* and *aggr2*:

```
cluster1::*> snapmirror config-replication cluster-storage-configuration
modify -disallowed-aggregates aggr1,aggr2
```

## snapmirror config-replication cluster-storage-configuration show

Display SnapMirror storage configuration information

**Availability:** This command is available to *cluster* administrators at the *advanced* privilege level.

### Description

The `snapmirror config-replication cluster-storage-configuration show` command shows details of the configuration of the storage used for configuration replication.

The information displayed is the following:

- Disallowed Aggregates - The list of storage aggregates that are configured as not allowed to host storage areas.
- Auto-Repair - Displays *true* if the automatic repair of storage areas used by configuration replication is enabled.
- Auto-Recreate - Displays *true* if the automatic recreation of storage volumes used by configuration replication is enabled.
- Use Mirrored Aggregate - Displays *true* if storage areas for configuration replication are to be hosted on a mirrored aggregate.

## Examples

The following is an example of the `snapmirror config-replication cluster-storage-configuration show` command:

```
cluster1::*> snapmirror config-replication cluster-storage-configuration
show
Disallowed Aggregates: -
    Auto-Repair: true
    Auto-Recreate: true
Use Mirrored Aggregate: true
```

## snapmirror config-replication status show-aggregate-eligibility

Display the SnapMirror configuration replication aggregate eligibility

**Availability:** This command is available to *cluster* administrators at the *admin* privilege level.

### Description

The `snapmirror config-replication status show-aggregate-eligibility` command displays the SnapMirror configuration replication aggregate eligibility.

### Parameters

**{ [-fields <fieldname>,...]**

If you specify the `-fields <fieldname>`, ... parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays detailed information about all fields.

**[-aggregate <aggregate name>] - Aggregate**

Display only rows that have a matching aggregate name.

**[-hosted-configuration-replication-volumes <volume name>,...] - Currently Hosted Configuration Replication Volumes**

Display only rows that have matching configuration replication volumes hosted on this aggregate.

**[-is-eligible-to-host-additional-volumes {true|false}] - Eligibility to Host Another Configuration Replication Volume**

Display only rows that have a matching eligibility of the aggregate to host additional configuration replication volumes.

**[-comment <text>] - Comment for Eligibility Status**

Display only rows that have a matching comment regarding the eligibility of the aggregate to host configuration replication volumes.

### Examples

The following example shows the execution of the command in a SnapMirror configuration with thirteen aggregates in the cluster:

```
clusA::snapmirror config-replication status> show-aggregate-eligibility
Aggregate      Hosted Config Replication Vols      Eligible to
Comments                                             Host Addl Vols
-----
-----
a0              -                               false
Root Aggregate
a1             MDV_CRS_1bc7134a5ddf11e3b63f123478563412_A true      -
a2             MDV_CRS_1bc7134a5ddf11e3b63f123478563412_B true      -
a3              -                               false
Unable to determine available space of aggregate
a4              -                               false
Non-Local Aggregate
a5              -                               false
Non-Home Aggregate
a6              -                               false
Unable to determine mirror configuration
a7              -                               false
Mirror configuration does not match requirement
a8              -                               false
Disallowed Aggregate
a9              -                               false
Insufficient Space - 10GB required
a10            -                               false
Aggregate Offline
a11            -                               false
Inconsistent Aggregate
a12            -                               false
Aggregate Full
13 entries were displayed.
```

## snapmirror config-replication status show-communication

Display SnapMirror configuration replication communication status information

**Availability:** This command is available to *cluster* administrators at the *admin* privilege level.

### Description

The `snapmirror config-replication status show-communication` command displays the current SnapMirror configuration replication communication status.

The command displays the following aspects of SnapMirror configuration replication for each peer cluster:

- **Remote Heartbeat:** Verifies that the SnapMirror configuration replication heartbeat with the remote cluster is healthy.



- Last Heartbeat Sent: Prints the timestamp of the last SnapMirror configuration replication heartbeat sent to the remote cluster.
- Last Heartbeat Received: Prints the timestamp of the last SnapMirror configuration replication heartbeat received from the remote cluster.

Additional information about the warnings (if any) and recovery steps can be viewed by running the command with the `-instance` option.

## Parameters

**{ [-fields <fieldname>,...]**

If you specify the `-fields <fieldname>`, ... parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays detailed information about all fields.

**[-cluster-uuid <UUID>] - Remote Cluster**

Display only rows that have a matching peer cluster UUID.

**[-cluster <text>] - Peer Cluster Name**

Display only rows that have matching peer cluster name.

**[-remote-heartbeat {ok|warning|not-run|not-applicable}] - Remote Heartbeat**

Display only rows that have a matching remote heartbeat status.

**[-last-heartbeat-sent <MM/DD/YYYY HH:MM:SS>] - Last Heartbeat Sent Time**

Display only rows that have a matching timestamp of the last heartbeat sent.

**[-last-heartbeat-received <MM/DD/YYYY HH:MM:SS>] - Last Heartbeat Received Time**

Display only rows that have a matching timestamp of the last heartbeat received.

**[-heartbeat-recovery-steps <text>] - Heartbeat Recovery Steps**

Display only rows that have matching heartbeat recovery steps.

## Examples

The following example shows the execution of the command in a SnapMirror configuration with two peer clusters:

```
clus1::*> snapmirror config-replication status show-communication
      Peer Cluster: clus2
      Remote Heartbeat: ok
      Last Heartbeat Sent: 11/11/2014 11:11:45
      Last Heartbeat Received: 11/11/2014 11:11:46
      Peer Cluster: clus3
      Remote Heartbeat: ok
      Last Heartbeat Sent: 11/11/2014 11:11:26
      Last Heartbeat Received: 11/11/2014 11:11:27

2 entries were displayed.
```

## snapmirror config-replication status show

Display SnapMirror configuration replication status information

**Availability:** This command is available to *cluster* administrators at the *admin* privilege level.

### Description

The `snapmirror config-replication status show` command displays the current SnapMirror configuration replication status.

The command displays the following aspects of SnapMirror configuration replication:

- **Enabled:** Verifies that SnapMirror configuration replication is enabled on the cluster.
- **Running:** Verifies that SnapMirror configuration replication is running on the cluster.
- **Storage Status:** Verifies that SnapMirror configuration replication storage is healthy.
- **Storage In Use:** Prints the location of SnapMirror configuration replication storage.
- **Storage Remarks:** Prints the underlying root cause for non-healthy SnapMirror configuration storage.
- **Vserver Streams:** Verifies that SnapMirror configuration replication Vserver streams are healthy.

Additional information about the warnings (if any) and recovery steps can be viewed by running the command with the `-instance` option.

### Parameters

**[`-instance` ]**

If you specify the `-instance` parameter, the command displays detailed information about all fields.

### Examples

The following example shows the execution of the command:

```

clus1::*> snapmirror config-replication status show
      Enabled: true
      Running: true
      Storage Status: ok
      Storage In Use: Cluster-wide Volume:
MDV_CRS_3d47e9106b7d11e4a77b000c29f810a2_A
      Storage Remarks: -
      Vserver Streams: ok

```

## snapmirror failover commands

### snapmirror failover show

Display failover status

**Availability:** This command is available to *cluster* administrators at the *admin* privilege level.

#### Description

The `snapmirror failover show` command is used to monitor the progress of the failover operation on the SnapMirror active sync relationship. You must execute this command from the destination cluster of the SnapMirror active sync relationship.

#### Parameters

**{ [-fields <fieldname>,...]**

If you specify the `-fields <fieldname>`, ... parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays detailed information about all fields.

**[-start-time <MM/DD/YYYY HH:MM:SS>] - Start Time**

Start operation time of the failover.

**[-source-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}} - Source Path**

Specifies the source endpoint of the SnapMirror relationship.

**[-destination-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}} - Destination Path**

Specifies the destination endpoint of the SnapMirror relationship.

**[-status {started|failed|completed|completed\_with\_warnings}] - Failover Status**

The status of snapmirror failover.

**[`-error-reason <text>`] - Error Reason**

The reason for failure occurred during snapmirror failover.

**[`-end-time <MM/DD/YYYY HH:MM:SS>`] - End Time**

End operation time of the failover.

**[`-primary-data-cluster <text>`] - Primary Data Cluster**

Primary data cluster.

**[`-last-progress-update <text>`] - Last Progress Update**

Last progress update.

**[`-type {planned|unplanned|incapable}`] - Failover Type**

Whether the SnapMirror failover was planned or unplanned.

**[`-error-reason-codes <integer>,...`] - Error Reason codes**

The reason codes for failure occurred during snapmirror failover.

**Examples**

To see the progress of the failover operation performed on an SnapMirror active sync relationship with destination Consistency Group `cg_dst` in Vserver `vs2.example.com`, type the following command from the destination cluster:

```
destination::> snapmirror failover show -destination-path  
vs2.example.com:/cg/cg_dst
```

**snapmirror failover start**

Start planned failover for SnapMirror relationships with 'automated-failover' policy-type

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

**Description**

The `snapmirror failover start` command initiates a planned failover operation to change the roles of copies in a SnapMirror active sync relationship without disrupting the client operations. The primary writable volumes become the secondary data protection volumes while the secondary data protection volumes become the primary writable volumes. The new primary volumes can begin processing I/O requests locally without disrupting the client operations. Asymmetric Logical Unit Access (ALUA) reporting will also change as a part of this role change.

You should execute the `snapmirror failover start` command from the destination cluster.

Make sure the relationship status is InSync and the ONTAP Mediator is configured, connected, and in quorum before using this command to switch the primary and secondary roles.

## Parameters

### **-destination-path**

{<[vserver:] [volume]>|<[[cluster:]//vserver/] volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>} - Destination Path

Specifies the destination endpoint of the SnapMirror relationship.

## Examples

To perform a planned failover operation of an SnapMirror active sync relationship with source Consistency Group *cg\_src* in Vserver *vs1.example.com*, and the destination Consistency Group *cg\_dst* in Vserver *vs2.example.com*, type the following command from the destination cluster:

```
destination::> snapmirror failover start -destination-path
                vs2.example.com:/cg/cg_dst
```

Note: The SnapMirror active sync relationship information in the [snapmirror show](#) command will now be available on the new destination that is Vserver *vs2.example.com* with *cg\_dst* as the new source and *cg\_src* as the new destination Consistency Groups after the SnapMirror failover operation is completed.

## Related Links

- [snapmirror show](#)

# snapmirror mediator commands

## snapmirror mediator add

Create mediator config entry

**Availability:** This command is available to *cluster* administrators at the *admin* privilege level.

## Description

The `snapmirror mediator add` command configures the connection between a pair of clusters and an ONTAP Mediator. It is mandatory to initialize the ONTAP Mediator on one of the cluster peers before the SnapMirror active sync relation with policy type *automated-failover* performs a planned or an unplanned failover. You can initialize the ONTAP Mediator from either cluster. When you issue the ONTAP Mediator add command on one cluster, the ONTAP Mediator is automatically added on the other cluster.

## Parameters

### **-mediator-address <IP Address> - Mediator Ip Address**

Specifies the IP address of the mediator.

### **-peer-cluster <text> - Peer cluster**

Peer cluster with AutomatedFailOver SnapMirror relationships.

**-username <text> - Username**

User account at the mediator.

**[-port-number <integer>] - Port Number**

This optional parameter specifies the mediator service port number to communicate with the mediator. The port number must be in the range 1025 to 65535 inclusive. The default port number is 31784.

**Examples**

The following example configures the connection between a mediator and a pair of clusters.

```
clusA::> snapmirror mediator add -mediator-address 10.234.102.227 -peer
-cluster clusB -username admin
Notice: Enter the mediator password.
Enter the password:
      Enter the password again:
Info: [Job: 114] 'mediator add' job queued
```

**snapmirror mediator remove**

Remove mediator config entry

**Availability:** This command is available to *cluster* administrators at the *admin* privilege level.

**Description**

The `snapmirror mediator remove` command deletes a mediator configuration entry.

**Parameters****-mediator-address <IP Address> - Mediator Ip Address**

Specifies the IP address of the mediator.

**-peer-cluster <text> - Peer cluster**

Peer cluster with AutomatedFailOver SnapMirror relationships.

**Examples**

The following example removes a mediator configuration entry.

```
clusA::> snapmirror mediator remove -mediator-address 10.140.102.227 -peer
-cluster clusB
Info: [Job 36] 'mediator remove' job queued
```

## snapmirror mediator show

### Show mediator information

**Availability:** This command is available to *cluster* administrators at the *admin* privilege level.

### Description

The `snapmirror mediator show` command shows the status of the ONTAP Mediator configuration.

### Parameters

**{ [-fields <fieldname>,...]**

If you specify the `-fields <fieldname>`, ... parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays detailed information about all fields.

**[-mediator-address <IP Address>] - Mediator IP Address**

Ip address of the mediator.

**[-peer-cluster <text>] - Peer Cluster Name**

Peer cluster name.

**[-connection-status**

**{connected|unreachable|suspended|unusable|misconfigured|removing|not-configured|unknown|adding}] - Connection Status**

Connection status of the cluster with the mediator.

**[-quorum-status {unknown|true|false}] - Mediator Quorum Status**

Shows whether the SnapMirror Consistency Group relationships are synchronized with mediator.

**[-health-fetch-timeout <integer>] - Peer Health Fetch Timeout**

Timeout value (in seconds) for peer-cluster health fetch via mediator.

**[-connection-timeout <integer>] - Connection Timeout**

Timeout value (in seconds) for mediator connection.

### Examples

The following example shows the list of mediator configurations.

```
clusA:*> snapmirror mediator show
  Mediator Address Peer Cluster      Connection Status Quorum Status
  -----
  10.140.102.227  clusB      unreachable      true
clusA:*> snapmirror mediator show -instance
Mediator Uuid: 416fbdee-c982-11e9-9034-005056a7124c
Mediator IP Address: 10.140.102.227
Peer Cluster: clusB
Peer Cluster Uuid: 771d9b13-c973-11e9-928e-005056a7a882
Connection Status: unreachable
Quorum Status: true
Health Fetch Timeout: 5
Connection Timeout: 5
```

## snapmirror mediator primary-bias show

### Show Primary Bias Status

**Availability:** This command is available to *cluster* administrators at the *admin* privilege level.

### Description

The `snapmirror mediator primary-bias show` command shows the status of primary-bias. This command is applicable when mediator status is shown as unreachable/misconfigured from both primary and secondary cluster.

This will also show the state corresponding to each CG endpoint residing on the given cluster. Slave CG endpoints will show the state of "Failover disabled". Master CG endpoints will show the state of Primary-bias activation.

The status of primary-bias can be one of the following. + primary-bias-activated - represents the state that allows master endpoints to assume the I/O serving authority + primary-bias-not-activated - represents the state that mediator is reachable from primary and secondary clusters and IO serving authority set to default + mediator-engaged-for-failover - represents the state that mediator is engaged and Failover on slave endpoints is possible + mediator-disengaged-for-failover - represents the setting of the intermediate state preventing Failover on slave endpoints of a CG +

### Parameters

{ [-fields <fieldname>,...]

If you specify the `-fields <fieldname>, ...` parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

| [-instance ] }

If you specify the `-instance` parameter, the command displays detailed information about all fields.

[-cg-path

{<[vserver:][volume]>|<[[cluster:]/vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}} - Cg Path



CG path.

**[-cg-role {destination|source}] - Role**

Indicates if CG is primary or secondary on local cluster.

**[-mb-state {Primary bias activated. Mediator Disengaged|Primary bias not-activated. Mediator engaged|Mediator disengaged for failover|Mediator engaged for failover}] - MB state**

Primary bias State.

### Examples

The following example shows the status of primary bias

```
cluster1::> snapmirror mediator primary-bias show
CG Path           Role           Status
-----
vs1:/cg/dcg       destination    mediator-engaged-for-failover
```

## snapmirror mediator primary-bias history show

### Show Primary Bias History

**Availability:** This command is available to *cluster* administrators at the *admin* privilege level.

### Description

The `snapmirror mediator primary-bias history show` command shows the history of Primary bias events.

The setting/clearing of the Primary Bias states (including the intermediate state) will be logged as event history per CG endpoint in the "snapmirror mediator primary bias history" command. The "mediator-engaged/disengaged-for-failover" actions will represent the setting/clearing of the intermediate state preventing Failover on slave endpoints of a CG. The "primary-bias-activated/not-activated" will represent the state that allows master endpoints to assume the I/O serving authority.

### Parameters

**{ [-fields <fieldname>,...]**

If you specify the `-fields <fieldname>`, ... parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays detailed information about all fields.

**[-cg-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}] - Cg Path**

CG path.

**[`-start-time` <MM/DD/YYYY HH:MM:SS>] - TBA Task Start Time**

Time of event

**[`-action` {Primary bias activated. Mediator Disengaged|Primary bias not-activated. Mediator engaged|Mediator disengaged for failover|Mediator engaged for failover}]**

**- Event**

Primary bias event.

## Examples

The following example shows the history output

```
C1_sti7-vsimgucs569m_cluster::*> snapmirror mediator primary-bias history
show -cg-path vs0:/cg/scg
  CG Path          Time                Action
  -----
vs0:/cg/scg      7/25/2022 10:33:28
                  primary-bias-activated
vs0:/cg/scg      7/25/2022 22:42:33
                  primary-bias-not-activated
  2 entries were displayed.
C1_sti7-vsimgucs569m_cluster::*>
```

## snapmirror mediator tba-history show

### Show Mediator Agent Event History

**Availability:** This command is available to *cluster* administrators at the *admin* privilege level.

### Description

The `snapmirror mediator tba-history show` command shows the history of the ONTAP Mediator Agent events..

### Parameters

**{ [`-fields` <fieldname>,...]**

If you specify the `-fields <fieldname>`, ... parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

**| [`-instance` ] }**

If you specify the `-instance` parameter, the command displays detailed information about all fields.

**[-cg-rel-uuid <UUID>] - Cg Relationship Uuid**

Cg Relationship Uuid.

**[-start-time <MM/DD/YYYY HH:MM:SS>] - TBA Task Start Time**

Mediator Agent Task Start Time.

**[-token-num <integer>] - Token Number**

Token Number.

**[-cg-path**

**{<[vserver:] [volume]>|<[[cluster:]//vserver/]volume>|<hostip:/lun/name>|<hostip:/share/share-name>|<[vserver:]/cg/[app-cgname]>}] - Cg Path**

Cg Path.

**[-aux-generation-num <integer>] - Auxillary Generation Number**

Auxillary Generation Number.

**[-cg-rel-task <text>] - TBA Task Type**

Mediator Agent Task Type.

**[-cg-rel-state {invalid-state|waiting|task-processing|task-complete}] - Cg Management State**

Cg Management State.

**[-end-time <MM/DD/YYYY HH:MM:SS>] - TBA Task End Time**

Mediator Agent Task End Time.

**[-cg-rel-action {invalid|none|continue|stop|failover-incapable|auto-failover|success|failed|mismatch|retry|success-pending|stop-pending|continue-pending|failover-incapable-pending|try-failover-incapable-ic|try-failover-incapable-med|try-auto-failover}] - Cg Management Action**

Cg Management Action.

**[-errCode <integer>] - Error Code for Task Failure**

Error Code for Task Failure.

## Examples

The following example shows the list of mediator configurations.

```

C1_sti89-vsimgucs525q_cluster::> snapmirror mediator tba-history show
      (snapmirror mediator tba-history show)
      Path          Token API   Event   Start Time  State      End Time
Action
-----
vs0:/cg/cg1_src 5   4      Process Set Resync Context 4/28/2021
11:09:52 task-complete 4/28/2021 11:09:52 continue
vs0:/cg/cg1_src 6   1      Process Master InSync 4/28/2021 11:10:08
task-complete 4/28/2021 11:10:08 success
2 entries were displayed.
C1_sti89-vsimgucs525q_cluster::> snapmirror mediator tba-history
show -instance
Cg Relationship Uuid: ad49a4f1-a833-11eb-9846-005056a7c906
      Mediator Agent Task Start Time: 4/28/2021 11:09:52
      Token Number: 5
      Cg Path: vs0:/cg/cg1_src
Auxillary Generation Number: 4
      Mediator Agent Task Type: Process Set Resync Context
      Cg Management State: task-complete
Mediator Agent Task End Time: 4/28/2021 11:09:52
      Cg Management Action: continue
Error Code for Task Failure: 0
Cg Relationship Uuid: ad49a4f1-a833-11eb-9846-005056a7c906
      Mediator Agent Task Start Time: 4/28/2021 11:10:08
      Token Number: 6
      Cg Path: vs0:/cg/cg1_src
Auxillary Generation Number: 1
      Mediator Agent Task Type: Process Master InSync
      Cg Management State: task-complete
Mediator Agent Task End Time: 4/28/2021 11:10:08
      Cg Management Action: success
Error Code for Task Failure: 0

```

## snapmirror object-store commands

### snapmirror object-store config create

Define the configuration for a SnapMirror object store

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

#### Description

The `snapmirror object-store config create` command is used by a cluster administrator to tell Data ONTAP how to connect to an object store. Following pre-requisites must be met before creating an object store

configuration in Data ONTAP.

- A valid data bucket or container must be created with the object store provider. This assumes that the user has valid account credentials with the object store provider to access the data bucket.
- The Data ONTAP node must be able to connect to the object store. This includes
- Fast, reliable connectivity to the object store.
- An inter-cluster LIF (Logical Interface) must be configured on the cluster.
- If SSL/TLS authentication is required, then valid certificates must be installed.

An object-store configuration once created must not be reassociated with a different object-store or container. See [snapmirror object-store config modify](#) command for more information.

## Parameters

### **-vserver <vserver name> - Vserver Name**

This parameter specifies the vservers on which the object store configuration needs to be created.

### **-object-store-name <text> - Object Store Configuration Name**

This parameter specifies the name that will be used to identify the object store configuration. The name can contain the following characters: `"`, `-`, `A-Z`, `a-z`, and `0-9`. *The first character must be one of the following: `"`, `A-Z`, or `a-z`.*

### **-usage {data|metadata} - Object Store Use**

This parameter specifies the usage for an object store configuration.

### **-provider-type <providerType> - Type of the Object Store Provider**

This parameter specifies the type of object store provider that will be attached to the aggregate. Valid options are: `AWS_S3` (Amazon S3 storage), `Azure_Cloud` (Microsoft Azure Cloud), `SGWS` (StorageGrid WebScale), `IBM_COS` (IBM Cloud Object Storage), `AliCloud` (Alibaba Cloud Object Storage Service), `GoogleCloud` (Google Cloud Storage) and `ONTAP_S3`.

### **-server <Remote InetAddress> - Fully Qualified Domain Name of the Object Store Server**

This parameter specifies the Fully Qualified Domain Name (FQDN) of the remote object store server. For Amazon S3, server name must be an AWS regional endpoint in the format `s3.amazonaws.com` or `s3-<region>.amazonaws.com`, for example, `s3-us-west-2.amazonaws.com`. The region of the server and the bucket must match. For more information on AWS regions, refer to 'Amazon documentation on AWS regions and endpoints'. For Azure, if the `-server` is a `"blob.core.windows.net"` or a `"blob.core.usgovcloudapi.net"`, then a value of `-azure-account` followed by a period will be added in front of the server.

### **[-is-ssl-enabled {true|false}] - Is SSL/TLS Enabled**

This parameter indicates whether a secured SSL/TLS connection will be used during data access to the object store. The default value is `true`.

### **[-port <integer>] - Port Number of the Object Store**

This parameter specifies the port number on the remote server that Data ONTAP will use while establishing connection to the object store.

**-container-name <text> - Data Bucket/Container Name**

This parameter specifies the data bucket or container that will be used for read and write operations.



This name cannot be modified once a configuration is created.

**[-access-key <text>] - Access Key ID for S3 Compatible Provider Types**

This parameter specifies the access key (access key ID) required to authorize requests to the AWS S3, SGWS, IBM COS object stores and ONTAP\_S3. For an Azure object store see `-azure-account`.

**[-ipspace <IPspace>] - IPspace to Use in Order to Reach the Object Store**

This optional parameter specifies the IPspace to use to connect to the object store. Default value: *Default*.

**[-use-iam-role {true|false}] - (DEPRECATED)-Use IAM Role for AWS Cloud Volumes ONTAP**

This optional parameter is deprecated. Please use `-auth-type` instead. Note, that `-auth-type EC2-IAM` is an equivalent of `-use-iam-role true`, and `-auth-type key` is an equivalent of `-use-iam-role false`.

**[-secret-password <text>] - Secret Access Key for S3 Compatible Provider Types**

This parameter specifies the password (secret access key) to authenticate requests to the AWS S3, SGWS, IBM COS object stores and ONTAP\_S3. If the `-access-key` is specified but the `-secret-password` is not, then one will be asked to enter the `-secret-password` without echoing the input.

**[-is-certificate-validation-enabled {true|false}] - Is SSL/TLS Certificate Validation Enabled**

This parameter indicates whether an SSL/TLS certificate of an object store server is validated whenever an SSL/TLS connection to an object store server is established. This parameter is only applicable when `is-ssl-enabled` is `true`. The default value is `true`. It is recommended to use the default value to make sure that Data ONTAP connects to a trusted object store server, otherwise identities of an object store server are not verified.

**[-azure-account <text>] - Azure Account**

This parameter specifies the account required to authorize requests to the Azure object store. For other object store providers see `access-key`.



The value of this field cannot be modified once a configuration is created.

**[-ask-azure-private-key {true|false}] - Ask to Enter the Azure Access Key without Echoing**

If this parameter is `true` then one will be asked to enter `-azure-private-key` without echoing the input. Default value: *true*.

**[-azure-private-key <text>] - Azure Access Key**

This parameter specifies the access key required to authenticate requests to the Azure object store. See also `ask-azure-private-key`. For other object store providers see `-secret-password`.

**[-server-side-encryption {none | SSE-S3}] - Encryption of Data at Rest by the Object Store Server (privilege: advanced)**

This parameter specifies if AWS or other S3 compatible object store server must encrypt data at rest. The available choices depend on `provider-type`. `none` encryption (no encryption required) is supported by all

types of S3 (non-Azure) object store servers. SSE-S3 encryption is supported by and is a default for all types of S3 (non-Azure) object store servers except ONTAP\_S3. This is an advanced property. In most cases it is best not to change default value of "sse\_s3" for object store servers which support SSE-S3 encryption. The encryption is in addition to any encryption done by ONTAP at a volume or at an aggregate level.

#### **[`-url-style {path-style | virtual-hosted-style}`] - URL Style Used to Access S3 Bucket**

This parameter specifies the URL style used to access S3 bucket. This option is only available for non-Azure object store providers. The available choices and default value depend on provider-type.

#### **[`-iamra-session-token <text>`] - IAMRA Session Token for Authentication**

This parameter specifies a temporary token for S3 snapmirror which will expire periodically. This will increase security.

### **Examples**

The following example creates a cluster scoped object store configuration named *objectStoreName*.

```
cluster1::*> snapmirror object-store config create
  -object-store-name objectStoreName -usage data -owner snapmirror
  -provider-type SGWS -server objectStoreServer.com
  -container-name containerName -is-ssl-enabled true
  -is-certificate-validation-enabled false
  -ipspace Default -access-key userAccessKey
  -secret-password userSecretPassWord
```

### **Related Links**

- [snapmirror object-store config modify](#)

## **snapmirror object-store config delete**

Delete SnapMirror object store configuration

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### **Description**

The `snapmirror object-store config delete` command removes an existing object store configuration in Data ONTAP.

### **Parameters**

#### **`-vserver <vserver name>` - Vserver Name**

This parameter specifies the vservers name on which the object store configuration has been configured.

#### **`-object-store-name <text>` - Object Store Configuration Name**

This parameter specifies the object store configuration to be deleted.

## Examples

The following example deletes an object store configuration named *objectStoreName*.

```
cluster1::*> snapmirror object-store config delete
      -object-store-name objectStoreName
```

## snapmirror object-store config modify

Modify SnapMirror object store configuration attributes

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### Description

The [storage aggregate object-store config modify](#) command is used to update one or more of object store configuration parameters.

### Parameters

**-vserver <vserver name> - Vserver Name**

This parameter specifies the vserver on which the object store configuration needs to be created.

**-object-store-name <text> - Object Store Configuration Name**

This parameter identifies the configuration to be modified.

**[-usage {data|metadata}] - Object Store Use**

This parameter specifies the usage for an object store configuration.

**[-new-object-store-name <text>] - Object Store Configuration New Name**

This optional parameter specifies the new name for the object store configuration.

**[-server <Remote InetAddress>] - Fully Qualified Domain Name of the Object Store Server**

This optional parameter specifies the new Fully Qualified Domain Name (FQDN) of the same object store server. For Amazon S3, server name must be an AWS regional endpoint in the format `s3.amazonaws.com` or `s3-<region>.amazonaws.com`, for example, `s3-us-west-2.amazonaws.com`. The region of the server and the bucket must match. For more information on AWS regions, refer to 'Amazon documentation on AWS regions and endpoints'. For Azure, if the `-server` is a "blob.core.windows.net" or a "blob.core.usgovcloudapi.net", then the value of `azure-account` in the configuration followed by a period will be added in front of the server. Note that the value of `azure-account` cannot be modified.

**[-is-ssl-enabled {true|false}] - Is SSL/TLS Enabled**

This optional parameter indicates whether a secured SSL/TLS connection will be used during data access to the object store.

**[-port <integer>] - Port Number of the Object Store**

This optional parameter specifies a new port number to connect to the object store server indicated in the `-server` parameter.



**[-access-key <text>] - Access Key ID for S3 Compatible Provider Types**

This optional parameter specifies a new access key (access key ID) for the AWS S3, SGWS, IBM COS object stores and ONTAP S3.

**[-ipspace <IPspace>] - IPspace to Use in Order to Reach the Object Store**

This optional parameter specifies new ipspace values for the configuration.

**[-use-iam-role {true|false}] - (DEPRECATED)-Use IAM Role for AWS Cloud Volumes ONTAP**

This optional parameter is deprecated. Please use `-auth-type` instead. Note, that `-auth-type EC2-IAM` is an equivalent of `-use-iam-role true`, and `-auth-type key` is an equivalent of `-use-iam-role false`.

**[-secret-password <text>] - Secret Access Key for S3 Compatible Provider Types**

This optional parameter specifies a new password (secret access key) for the AWS S3, SGWS, IBM COS object stores and ONTAP S3. For an Azure object store see `-azure-private-key`. If the `-access-key` is specified but the `-secret-password` is not then one will be asked to enter the `-secret-password` without echoing the input.

**[-is-certificate-validation-enabled {true|false}] - Is SSL/TLS Certificate Validation Enabled**

This optional parameter indicates whether an SSL/TLS certificate of an object store server is validated whenever an SSL/TLS connection to an object store server is established. This parameter is only applicable when `is-ssl-enabled` is `true`. It is recommended to keep the default value which is `true` to make sure that Data ONTAP connects to a trusted object store server, otherwise identities of an object store server are not verified.

**[-ask-azure-private-key {true|false}] - Ask to Enter the Azure Access Key without Echoing**

If this optional parameter is `true` then one will be asked to enter the `-azure-private-key` without echoing the input.

**[-azure-private-key <text>] - Azure Access Key**

This optional parameter specifies a new access key for Azure object store. For other object store providers see `secret-password`. See also `ask-azure-private-key`.

**[-server-side-encryption {none | SSE-S3}] - Encryption of Data at Rest by the Object Store Server (privilege: advanced)**

This parameter specifies if AWS or other S3 compatible object store server must encrypt data at rest. The available choices depend on `provider-type`. `none` encryption (no encryption required) is supported by all S3 (non-Azure) object store servers. `SSE-S3` encryption is supported by all S3 (non-Azure) object store servers except `ONTAP_S3`. This is an advanced property. In most cases it is best not to change default value of `"sse_s3"` for object store servers which support SSE-S3 encryption. The encryption is in addition to any encryption done by ONTAP at a volume or at an aggregate level. Note that changing this option does not change encryption of data which already exist in the object store.

**[-url-style {path-style | virtual-hosted-style}] - URL Style Used to Access S3 Bucket**

This parameter specifies the URL style used to access S3 bucket. This option is only available for non-Azure object store providers. The available choices and default value depend on `provider-type`.

**[-iamra-session-token <text>] - IAMRA Session Token for Authentication**

This parameter specifies a temporary token for S3 snapmirror which will expire periodically. This will increase security.

## Examples

The following example modifies an object-store configuration named *objectStoreName* to a new name *newName*.

```
cluster:*> snapmirror object-store config modify
      -object-store-name objectStoreName
      -new-object-store-name newName
```

## Related Links

- [storage aggregate object-store config modify](#)

## snapmirror object-store config show

Display a list of SnapMirror object store configurations

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror object-store config show` command displays information about all existing object store configurations in the cluster.

## Parameters

**{ [-fields <fieldname>,...]**

If you specify the `-fields <fieldname>, ...` parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays detailed information about all fields.

**[-vserver <vserver name>] - Vserver Name**

If this parameter is specified, the command displays information only about object store configurations which are configured within this vservers.

**[-object-store-name <text>] - Object Store Configuration Name**

If this parameter is specified, the command displays information only about object store configurations whose name matches the specified names.

**[-usage {data|metadata}] - Object Store Use**

Specifies what this object store configuration is used for.

**[-vsUuid <UUID>] - Vserver UUID**

Vserver UUID.

**[-config-id <integer>] - Object Store Configuration ID**

If this parameter is specified, the command displays information only about object store configurations

whose configuration ID matches the specified value.

**[`-provider-type <providerType>`] - Type of the Object Store Provider**

If this parameter is specified, the command displays information only about object store configurations whose provider type matches the specified value.

**[`-server <Remote InetAddress>`] - Fully Qualified Domain Name of the Object Store Server**

If this parameter is specified, the command displays information only about object store configurations whose server name matches the specified value. The server name is specified as a Fully Qualified Domain Name (FQDN).

**[`-is-ssl-enabled {true|false}`] - Is SSL/TLS Enabled**

If this parameter is specified, the command displays information only about object store configurations whose status about the use of secured communication over the network matches the specified value.

**[`-port <integer>`] - Port Number of the Object Store**

If this parameter is specified, the command displays information only about object store configurations whose port numbers matches the specified value.

**[`-container-name <text>`] - Data Bucket/Container Name**

If this parameter is specified, the command displays information only about object store configurations whose container name matches the specified value. Data ONTAP uses this container name or object store data bucket while accessing data from the object store.

**[`-access-key <text>`] - Access Key ID for S3 Compatible Provider Types**

If this parameter is specified, the command displays information only about AWS S3, SGWS, IBM COS object store configurations and ONTAP S3 whose access key matches the specified value. Data ONTAP requires the access key for authorized access to the object store.

**[`-ipspace <IPspace>`] - IPspace to Use in Order to Reach the Object Store**

If this parameter is specified, the command displays information only about object store configurations whose IPspace matches the specified value. Data ONTAP uses the IPspace value to connect to the object store.

**[`-use-iam-role {true|false}`] - (DEPRECATED)-Use IAM Role for AWS Cloud Volumes ONTAP**

If this parameter is specified, the command displays information only about object store configurations whose IAM role status flag matches the specified value. The `-iam-role` and `-use-iam-role` parameters are relevant only in the context of AWS object store and indicates whether IAM role must be used for accessing it. The IAM credentials can be obtained only through AWS Cloud Volumes ONTAP.

**[`-is-certificate-validation-enabled {true|false}`] - Is SSL/TLS Certificate Validation Enabled**

If this parameter is specified, the command displays information only about object store configurations whose status about the validation of SSL/TLS certificate matches the specified value.

**[`-azure-account <text>`] - Azure Account**

If this parameter is specified, the command displays information only about Azure object store configurations whose account matches the specified value. Data ONTAP requires the Azure account for authorized access to the Azure object store.

### **[`-server-side-encryption` {`none` | `SSE-S3`}] - Encryption of Data at Rest by the Object Store Server (privilege: advanced)**

If this parameter is specified, the command displays information only about object store configurations whose server-side encryption matches the specified value.

### **[`-url-style` {`path-style` | `virtual-hosted-style`}] - URL Style Used to Access S3 Bucket**

If this parameter is specified, the command displays information only about object store configurations whose URL style matches the specified value.

## **Examples**

The following example displays information about all existing object store configurations in the cluster.

```
cluster1::*> snapmirror object-store config show
```

## **snapmirror object-store profiler abort**

Abort object store profiler

**Availability:** This command is available to *cluster* administrators at the *advanced* privilege level.

### **Description**

The `snapmirror object-store profiler abort` command will abort an ongoing object store profiler run. This command requires two parameters - an object store configuration and a node on which the profiler is currently running.

### **Parameters**

**`-node` {<nodename>|`local`}** - Node on Which the Profiler Should Run (privilege: advanced)

This parameter specifies the node on which the object store profiler is running.

**`-object-store-name` <text>** - Object Store Configuration Name (privilege: advanced)

This parameter specifies the object store configuration that describes the object store. The object store configuration has information about the object store server name, port, access credentials, and provider type.

## **Examples**

The following example aborts the object store profiler :

```
cluster1::> snapmirror object-store profiler abort -object-store-name my-store -node my-node
```

## **snapmirror object-store profiler show**

**Availability:** This command is available to *cluster* administrators at the *advanced* privilege level.

## Description

The `snapmirror object-store profiler show` command is used to monitor progress and results of the `snapmirror object-store profiler start` command.

## Parameters

**{ [-fields <fieldname>,...]**

If you specify the `-fields <fieldname>`, ... parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays detailed information about all fields.

**[-node <nodename>] - Node Name (privilege: advanced)**

This parameter specifies the node on which the profiler was started.

**[-object-store-name <text>] - ONTAP Name for this Object Store Configuration (privilege: advanced)**

This parameter specifies the object store configuration that describes the object store. The object store configuration has information about the object store server name, port, access credentials, and provider type.

**[-object-name-prefix <UUID>] - Bin UUID (privilege: advanced)**

This parameter specifies the object name prefix.

**[-object-prefix <text>] - Prefix Added to Each Object (privilege: advanced)**

If this parameter is specified, the command displays information only about the objects whose prefix matches the specified prefix.

**[-profiler-status <text>] - Profiler Status (privilege: advanced)**

Current status of the profiler.

**[-start-time <MM/DD/YYYY HH:MM:SS>] - Profiler Start Time (privilege: advanced)**

Time at which profiler run started.

**[-op-name <text>] - Operation Name - PUT/GET (privilege: advanced)**

Name of the operation. Possible values are PUT or GET.

**[-op-size {<integer>[KB|MB|GB|TB|PB]}] - Size of Operation (privilege: advanced)**

Size of the PUT or GET operation.

**[-op-count <integer>] - Number of Operations Performed (privilege: advanced)**

Number of operations issued to the object store.

**[-op-failed <integer>] - Number of Operations Failed (privilege: advanced)**

Number of operations that failed.

**[-op-latency-minimum <integer>] - Minimum Latency for Operation in Milliseconds (privilege: advanced)**

Minimum latency of the operation in milliseconds, as measured from the filesystem layer.

**[-op-latency-maximum <integer>] - Maximum Latency for Operation in Milliseconds (privilege: advanced)**

Maximum latency of the operation in milliseconds, as measured from the filesystem layer.

**[-op-latency-average <integer>] - Average Latency for Operation in Milliseconds (privilege: advanced)**

Average latency of the operation in milliseconds, as measured from the filesystem layer.

**[-op-throughput {<integer>[KB|MB|GB|TB|PB]}] - Throughput per Second for the operation (privilege: advanced)**

Throughput per second for the operation.

**[-op-errors <text>,...] - Error Reasons and Count (privilege: advanced)**

Error reasons and count for failed operation.

**[-op-latency-histogram <text>,...] - Latency Histogram (privilege: advanced)**

Latency histogram for the operation.

## Examples

The following example displays the results of [snapmirror object-store profiler start](#) :

```
cluster1::>snapmirror object-store profiler show
```

## Related Links

- [snapmirror object-store profiler start](#)

## snapmirror object-store profiler start

Start the object store profiler to measure latency and throughput

**Availability:** This command is available to *cluster* administrators at the *advanced* privilege level.

## Description

The `snapmirror object-store profiler start` command writes objects to an object store and reads those objects to measure latency and throughput of an object store. This command requires two parameters - an object store configuration and node from which to send the PUT/GET/DELETE operations. This command verifies whether the object store is accessible through the intercluster LIF of the node on which it runs. The command fails if the object store is not accessible. The command will create a 10GB dataset by doing 2500 PUTs for a maximum time period of 60 seconds. Then it will issue GET operations of different sizes - 4KB, 8KB, 32KB, 256KB for a maximum time period of 180 seconds. Finally it will delete the objects it created. This command can result in additional charges to your object store account. This is a CPU intensive command. It is recommended to run this command when the system is under 50% CPU utilization.

## Parameters

### **-node {<nodename>|local} - Node on Which the Profiler Should Run (privilege: advanced)**

This parameter specifies the node from which PUT/GET/DELETE operations are sent.

### **-object-store-name <text> - Object Store Configuration Name (privilege: advanced)**

This parameter specifies the object store configuration that describes the object store. The object store configuration has information about the object store server name, port, access credentials, and provider type.

### **[-object-prefix <text>] - Prefix Added to Each Object (privilege: advanced)**

This parameter specifies the prefix attached to each object. There is a prefix length restriction of 80 characters. In addition to this restriction, follow any specific prefix length or character restrictions that are imposed by the cloud store they plan to run this command. Refer to the respective cloud store documentation for details.

## Examples

The following example starts the object store profiler :

```
cluster1::>snapmirror object-store profiler start -object-store-name my-  
store -node my-node
```

# snapmirror policy commands

## snapmirror policy add-rule

Add a new rule to SnapMirror policy

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror policy add-rule` command adds a rule to a SnapMirror policy. Rules define which Snapshot copies are protected by vault relationships or define the schedule at which Snapshot copies are created on the SnapMirror destination. Rules which do not include a schedule are rules for protecting Snapshot copies. Rules which include a schedule are rules for creating Snapshot copies on the SnapMirror destination. A rule with a schedule can only be added to SnapMirror policies of type *vault* or *mirror-vault*. A rule must not be added to a policy that will be associated with a SnapMirror data protection relationship. A policy that will be associated with a SnapMirror vault relationship must have at least one rule and at most ten rules. A SnapMirror policy with rules must have at least one rule without a schedule.

## Parameters

### **-vserver <vserver name> - Vserver Name**

Specifies the Vserver for the SnapMirror policy.

### **-policy <sm\_policy> - SnapMirror Policy Name**

Specifies the SnapMirror policy name.

### **-snapmirror-label <text> - Snapshot Copy Label**

This parameter is primarily used for the purpose of Snapshot copy selection for extended data protection (XDP) relationships. Only Snapshot copies that have a SnapMirror label that matches this parameter will be transferred to the SnapMirror destination. However, when this parameter is associated with a rule containing a schedule, Snapshot copies will be created on the SnapMirror destination using this `snapmirror-label` parameter. The label can be 31 or fewer characters in length. SnapMirror policies of type `async-mirror` and `mirror-vault` have a rule added for label `sm_created` at the time of policy creation. This rule cannot be removed or modified by the user. This rule when coupled with `create-snapshot` field set to `true` indicates that the SnapMirror relationship using this policy shall create a new Snapshot copy and transfer it as part of a [snapmirror update](#) operation. SnapMirror policies of type `async-mirror` support one additional rule with SnapMirror label `all_source_snapshots`. This rule along with the rule for SnapMirror label `sm_created` indicates that *all* new Snapshot copies on the primary volume along with the newly created Snapshot copy are transferred as a part of a [snapmirror update](#) or [snapmirror initialize](#) operation. Rules with any other SnapMirror labels cannot be added to SnapMirror policies of type `async-mirror`. The rule for label `sm_created` when added to a `snapmirror` policy of type `vault` indicates that all SnapMirror created Snapshot copies of the primary volume are selected for transfer.

### **-keep <text> - Snapshot Copy Retention Count**

Specifies the maximum number of Snapshot copies that are retained on the SnapMirror destination volume for a rule. The total number of Snapshot copies retained for all the rules in a policy cannot exceed 1019. For all the rules in SnapMirror policies of type `async-mirror`, this parameter must be set to value `1`.

### **[-preserve {true|false}] - Snapshot Copy Preserve Enabled**

Specifies the behavior when the Snapshot copy retention count is reached on the SnapMirror vault destination for the rule. The default value is `false`, which means that the oldest Snapshot copy will be deleted to make room for new ones only if the number of Snapshot copies has exceeded the retention count specified in the "keep" parameter. When set to `true`, and when the Snapshot copies have reached the retention count, then an incremental SnapMirror vault update transfer will fail or if the rule has a schedule, Snapshot copies will no longer be created on the SnapMirror destination. For all the rules in SnapMirror policies of type `async-mirror` this parameter must be set to value `false`.

### **[-warn <integer>] - Warning Threshold Count**

Specifies the warning threshold count for the rule. The default value is `0`. When set to a value greater than zero, an event is generated after the number of Snapshot copies (for the particular rule) retained on a SnapMirror vault destination reaches the specified warn limit. The preserve parameter for the rule must be `true` to set the warn parameter to a value greater than zero.

### **[-schedule <text>] - Snapshot Copy Creation Schedule**

This optional parameter specifies the name of the schedule associated with a rule. This parameter is allowed only for rules associated with SnapMirror policies of type `vault` or `mirror-vault`. When this parameter is specified, Snapshot copies are directly created on the SnapMirror destination. The Snapshot copies created will have the same content as the latest Snapshot copy already present on the SnapMirror destination. Snapshot copies on the source that have a SnapMirror label matching this rule will not be selected for transfer. The default value is `-`.



You define and name a schedule using the [job schedule cron create](#) command.

### **[-prefix <text>] - Snapshot Copy Creation Prefix**

This optional parameter specifies the prefix for the Snapshot copy name to be created as per the schedule. If no value is specified, then the `snapmirror-label` will be used as the prefix. The prefix parameter can only be specified for rules which have a schedule.



**[`-retention-period` {<integer> `seconds` | `minutes` | `hours` | `days` | `months` | `years`} | `infinite`}] - Snapshot Copy Retention Period**

This optional parameter specifies the duration for which the Snapshot copies associated with this rule will be locked. The parameter is specified as a number followed by a suffix. The valid suffixes are *seconds*, *minutes*, *hours*, *days*, *months*, and *years*. For example, a value of *6months* represents a retention period of 6 months. It can also be set to a special value of *infinite* for infinite retention. The destination volume must have the *snapshot-locking-enabled* parameter set to true for this field to take effect else it will be ignored.

## Examples

The following example adds a rule named *nightly* to the SnapMirror policy named *TieredBackup* on Vserver *vs0.example.com*. The rule will retain a maximum of 5 *nightly* Snapshot copies.

```
vs0.example.com::> snapmirror policy add-rule -vserver vs0.example.com
                    -policy TieredBackup -snapmirror-label nightly -keep 5
```

## Related Links

- [snapmirror update](#)
- [snapmirror initialize](#)
- [job schedule cron create](#)

## snapmirror policy create

### Create a new SnapMirror policy

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

## Description

The `snapmirror policy create` command creates a SnapMirror policy. When applied to a SnapMirror relationship, the SnapMirror policy controls the behavior of the relationship and specifies the configuration attributes for that relationship. The policies *DPDefault*, *MirrorAllSnapshots*, *MirrorAndVault*, *MirrorLatest*, *Unified7year*, and *XDPDefault* are created by the system for asynchronous replication. The policies *Sync*, *StrictSync*, *AutomatedFailOver*, and *AutomatedFailOverDuplex* are created by the system for synchronous replication.



All SnapMirror policies have a field *create-snapshot*. This field specifies whether SnapMirror creates a new Snapshot copy on the primary volume at the beginning of a [snapmirror update](#) or [snapmirror resync](#) operation. Currently this field cannot be set or modified by the user. It is set to *true* for SnapMirror policies of type *async-mirror* and *mirror-vault* at the time of creation. SnapMirror policies of type *vault* have *create-snapshot* set to *false* at the time of creation.



Use the [snapmirror policy add-rule](#) command to add a rule to a policy.

## Parameters

### **-vserver <vserver name> - Vserver Name**

Specifies the Vserver for the SnapMirror policy.

### **-policy <sm\_policy> - SnapMirror Policy Name**

This parameter specifies the SnapMirror policy name. A policy name can be made up of the characters A to Z, a to z, 0 to 9, ".", "-", and "\_". The name can be up to 256 characters in length.

### **[-type {vault|async-mirror|mirror-vault|strict-sync-mirror|sync-mirror|automated-failover|automated-failover-duplex|continuous}] - Snapmirror Policy Type**

This parameter specifies the SnapMirror policy type. The supported values are *async-mirror*, *vault*, *mirror-vault*, *sync-mirror* and *strict-sync-mirror*. Data protection (DP) relationships support only *async-mirror* policy type, while extended data protection (XDP) relationships support all policy types.

If the type is set to *async-mirror* then the policy is for Disaster Recovery. When the policy type is associated with extended data protection (XDP) relationships, [snapmirror update](#) and [snapmirror resync](#) operations transfer selected Snapshot copies from the primary volume to the secondary volume. The selection of Snapshot copies is governed by the rules in the policy. However [snapmirror initialize](#) and [snapmirror update](#) operations on data protection (DP) relationships ignore the rules in the policy and transfer *all* Snapshot copies of the primary volume which are newer than the common Snapshot copy on the destination. For both data protection (DP) and extended data protection (XDP) relationships, the Snapshot copies are kept on the secondary volume as long as they exist on the primary volume. Once a protected Snapshot copy is deleted from the primary volume, it is deleted from the secondary volume as part of the next transfer. The policy type supports rules with certain pre-defined label names only. Refer to the man page for the [snapmirror policy add-rule](#) command for the details.

If the type is set to *vault* then the policy is used for Backup and Archive. The rules in this policy type determine which Snapshot copies are protected and how long they are retained on the secondary volume. This policy type is supported by extended data protection (XDP) relationships only.

If the type is set to *mirror-vault* then the policy is used for unified data protection which provides both Disaster Recovery and Backup using the same secondary volume. This policy type is supported by extended data protection (XDP) relationships only.

If the type is set to *sync-mirror* or *strict-sync-mirror* then the policy is used for synchronous Disaster Recovery. These are supported only by extended data protection (XDP) relationships between FlexVol volumes. Once the relationship is initialized with [snapmirror initialize](#), the relationship will be InSync such that all writes to the primary will be replicated to the secondary before the write is acknowledged to the client. Upon a replication failure, relationship falls OutOfSync. Upon an OutOfSync event, the *strict-sync-mirror* variant restricts further client IO on the primary, whereas the *sync-mirror* variant does not. SnapMirror will automatically trigger resync to bring the OutOfSync relationships back InSync as soon as it can, unless the relationship is *Quiesced* or *Broken-off*. Once a relationship is initialized, you normally use the [snapmirror quiesce](#) command to stop synchronous replication and the [snapmirror resume](#) command to resume synchronous replication. These policy types do not support replication of user Snapshot copies.

### **[-comment <text>] - Comment**

Specifies a text comment for the SnapMirror policy. If the comment contains spaces, it must be enclosed within quotes.

**[-tries <integer\_or\_unlimited>] - Tries Limit**

Determines the maximum number of times to attempt each manual or scheduled transfer for a SnapMirror relationship. The value of this parameter must be a positive integer or *unlimited*. The default value is *8*.

**[-transfer-priority {low|normal}] - Transfer Scheduling Priority**

Specifies the priority at which a transfer runs. The supported values are *normal* or *low*. The *normal* transfers are scheduled before the *low* priority transfers. The default is *normal*.

**[-ignore-atime {true|false}] - Ignore File Access Time**

This parameter applies only to extended data protection (XDP) relationships. It specifies whether incremental transfers will ignore files which have only their access time changed. The supported values are *true* or *false*. The default is *false*.

**[-restart {always|never|default}] - Restart Behavior**

This parameter applies only to data protection relationships. It defines the behavior of SnapMirror if an interrupted transfer exists. The supported values are *always*, *never*, or *default*. If the value is set to *always*, an interrupted SnapMirror transfer always restarts provided it has a restart checkpoint and the conditions are the same as they were before the transfer was interrupted. In addition, a new SnapMirror Snapshot copy is created which will then be transferred. If the value is set to *never*, an interrupted SnapMirror transfer will never restart, even if a restart checkpoint exists. A new SnapMirror Snapshot copy will still be created and transferred. Data ONTAP version 8.2 will interpret a value of *default* as being the same as *always*. Vault transfers will always resume based on a restart checkpoint, provided the Snapshot copy still exists on the source volume.

**[-is-network-compression-enabled {true|false}] - Is Network Compression Enabled**

Specifies whether network compression is enabled for transfers. The supported values are *true* or *false*. The default is *false*.

**[-rpo <integer>] - Recovery Point Objective (seconds)**

Specifies the time for recovery point objective, in seconds. This parameter is only supported for a policy of type *continuous*.

**[-always-replicate-snapshots {true|false}] - This prioritizes replication of app-consistent snapshots over synchronous replication**

If this parameter is set to *true*, it specifies that SnapMirror Synchronous relationships will lose the zero RPO protection upon failure in replicating application created snapshots. The default value is *false*.

**[-common-snapshot-schedule <text>] - Common Snapshot Copy Creation Schedule for SnapMirror Synchronous (privilege: advanced)**

Specifies the common Snapshot creating schedule. This parameter is only supported for SnapMirror Synchronous relationships.

**[-are-data-ops-sequentially-split {true|false}] - Is Sequential Splitting of Data Operations Enabled? (privilege: advanced)**

This parameter specifies whether I/O, such as write, copy-offload and punch-holes, are split sequentially, rather than being run in parallel on the source and destination. Splitting I/O sequentially will make the system more robust, and less prone to I/O errors. Starting 9.11.1, enabling this feature improves the performance when the workload is NAS based and is metadata heavy. However, it will make the IO performance slower for large file workloads like LUNs, databases, virtualization containers, etc. The default value for the parameter `-are-data-ops-sequentially-split` is *false*. The parameter `-are-data-ops-sequentially-split` should only be used if frequent I/O timeout or "OutOfSync" has happened.

The parameter `-are-data-ops-sequentially-split` requires an effective cluster version of Data ONTAP 9.6.0 or later on both the source and destination clusters.

**[`-sequential-split-op-timeout-secs <integer>`] - Sequential Split Op Timeout in Seconds (privilege: advanced)**

This parameter specifies the op timeout value used when the splitting mode is sequential. This parameter is used only for Sync relationships. Supported values are from 15 to 25 seconds.

**[`-disable-fast-resync {true|false}`] - Disable Fast Resync (privilege: advanced)**

Specifies whether fast resync for a SnapMirror Synchronous relationship is disabled or not. The default value for this parameter is `false`.

**[`-discard-configs <network>,...`] - Configurations Not Replicated During Identity Preserve Vserver DR**

Specifies the configuration to be dropped during replication. The supported values are:

- `network` - Drops network interfaces, routes, and kerberos configuration.

This parameter is supported only for policies of type `async-mirror` and applicable only for identity-preserve Vserver SnapMirror relationships.

**[`-transfer-schedule-name <text>`] - Transfer Schedule Name**

This optional parameter specifies the schedule which is used to update the SnapMirror relationships.

**[`-throttle <throttleType>`] - Throttle (KB/sec)**

This optional parameter limits the network bandwidth used for transfers. It configures for the relationships the maximum rate (in Kbytes/sec) at which data can be transferred. If no throttle is configured, by default the SnapMirror relationships fully utilize the network bandwidth available. You can also configure the relationships to fully use the network bandwidth available by explicitly setting the throttle to `unlimited` or `0`. The minimum effective throttle value is four Kbytes/sec, so if you specify a throttle value between `1` and `4`, it will be treated as `4`. For FlexGroup volume relationships, the throttle value is applied individually to each constituent relationship.

## Examples

The following example creates a SnapMirror policy named `TieredBackup` on a Vserver named `vs0.example.com`.

```
vs0.example.com::> snapmirror policy create -vserver vs0.example.com
                        -policy TieredBackup -type vault -tries 10 -restart never
```

## Related Links

- [snapmirror update](#)
- [snapmirror resync](#)
- [snapmirror policy add-rule](#)
- [snapmirror initialize](#)
- [snapmirror quiesce](#)

- [snapmirror resume](#)

## snapmirror policy delete

Delete a SnapMirror policy

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### Description

The `snapmirror policy delete` command deletes a SnapMirror policy. A policy that is to be deleted must not be associated with any SnapMirror relationship. The built-in policies cannot be deleted.

### Parameters

**-vserver <vserver name> - Vserver Name**

Specifies the Vserver for the SnapMirror policy.

**-policy <sm\_policy> - SnapMirror Policy Name**

Specifies the SnapMirror policy name.

### Examples

The following example deletes a SnapMirror policy named *TieredBackup* on Vserver *vs0.example.com*:

```
vs0.example.com::> snapmirror policy delete -vserver vs0.example.com
-policy TieredBackup
```

## snapmirror policy modify-rule

Modify an existing rule in SnapMirror policy

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### Description

The `snapmirror policy modify-rule` command can be used to modify the retention count, preserve setting, warning threshold count, schedule, and prefix for a rule in a SnapMirror policy. Reducing the retention count or disabling the preserve setting for a rule in a SnapMirror policy might result in the deletion of Snapshot copies on the vault destination when the next transfer by the [snapmirror update](#) command occurs or when the next scheduled Snapshot copy creation on the destination for the rule occurs. Modifying a rule to add a schedule will enable creation of Snapshot copies on the SnapMirror destination. Snapshot copies on the source that have a SnapMirror label matching this rule will not be selected for transfer. Schedule and prefix can only be modified for rules associated with SnapMirror policies of type *vault* or *mirror-vault*. A SnapMirror policy with rules must have at least one rule without a schedule.



The rules in SnapMirror policies of type *async-mirror* cannot be modified.

## Parameters

### **-vserver <vserver name> - Vserver Name**

Specifies the Vserver for the SnapMirror policy.

### **-policy <sm\_policy> - SnapMirror Policy Name**

Specifies the SnapMirror policy name.

### **-snapmirror-label <text> - Snapshot Copy Label**

This parameter specifies the rule that is to be modified in a SnapMirror policy.

### **[-keep <text>] - Snapshot Copy Retention Count**

Specifies the maximum number of Snapshot copies that are retained on the SnapMirror destination volume for a rule. The total number of Snapshot copies retained for all the rules in a policy cannot exceed 1019. For all the rules in SnapMirror policies of type *async-mirror*, this parameter must be set to value *1*.

### **[-preserve {true|false}] - Snapshot Copy Preserve Enabled**

Specifies the behavior when the Snapshot copy retention count is reached on the SnapMirror vault destination for the rule. The default value is *false*, which means that the oldest Snapshot copy will be deleted to make room for new ones only if the number of Snapshot copies has exceeded the retention count specified in the "keep" parameter. When set to *true*, and when the Snapshot copies have reached the retention count, then an incremental SnapMirror vault update transfer will fail or if the rule has a schedule, Snapshot copies will no longer be created on the SnapMirror destination. For all the rules in SnapMirror policies of type *async-mirror* this parameter must be set to value *false*.

### **[-warn <integer>] - Warning Threshold Count**

Specifies the warning threshold count for the rule. The default value is *0*. When set to a value greater than zero, an event is generated after the number of Snapshot copies (for the particular rule) retained on a SnapMirror vault destination reaches the specified warn limit. The preserve parameter for the rule must be *true* to set the warn parameter to a value greater than zero.

### **[-schedule <text>] - Snapshot Copy Creation Schedule**

This optional parameter specifies the name of the schedule associated with a rule. This parameter is allowed only for rules associated with SnapMirror policies of type *vault* or *mirror-vault*. When this parameter is specified, Snapshot copies are directly created on the SnapMirror destination. The Snapshot copies created will have the same content as the latest Snapshot copy already present on the SnapMirror destination. Snapshot copies on the source that have a SnapMirror label matching this rule will not be selected for transfer. The default value is *-*.



You define and name a schedule using the [job schedule cron create](#) command.

### **[-prefix <text>] - Snapshot Copy Creation Prefix**

This optional parameter specifies the prefix for the Snapshot copy name to be created as per the schedule. If no value is specified, then the *snapmirror-label* will be used as the prefix. The prefix parameter can only be specified for rules which have a schedule.

### **[-retention-period {{<integer> seconds|minutes|hours|days|months|years} | infinite}] - Snapshot Copy Retention Period**

This optional parameter specifies the duration for which the Snapshot copies associated with this rule will be locked. The parameter is specified as a number followed by a suffix. The valid suffixes are *seconds*,

*minutes*, *hours*, *days*, *months*, and *years*. For example, a value of *6months* represents a retention period of 6 months. It can also be set to a special value of *infinite* for infinite retention. The destination volume must have the *snapshot-locking-enabled* parameter set to true for this field to take effect else it will be ignored.

## Examples

The following example changes the retention count for *nightly* Snapshot copies to *6* for a rule named *nightly* on a SnapMirror policy named *TieredBackup* on Vserver *vs0.example.com*:

```
vs0.example.com::> snapmirror policy modify-rule -vserver vs0.example.com
                    -policy TieredBackup -snapmirror-label nightly -keep 6
```

## Related Links

- [snapmirror update](#)
- [job schedule cron create](#)

## snapmirror policy modify

### Modify a SnapMirror policy

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### Description

The `snapmirror policy modify` command can be used to modify the policy attributes.



Use the [snapmirror policy modify-rule](#) command to modify a rule in a SnapMirror policy.

### Parameters

**-vserver <vserver name> - Vserver Name**

Specifies the Vserver for the SnapMirror policy.

**-policy <sm\_policy> - SnapMirror Policy Name**

Specifies the SnapMirror policy name.

**[-comment <text>] - Comment**

Specifies a text comment for the SnapMirror policy. If the comment contains spaces, it must be enclosed within quotes.

**[-tries <integer\_or\_unlimited>] - Tries Limit**

Determines the maximum number of times to attempt each manual or scheduled transfer for a SnapMirror relationship. The value of this parameter must be a positive integer or *unlimited*. The default value is *8*.

**[-transfer-priority {low|normal}] - Transfer Scheduling Priority**

Specifies the priority at which a transfer runs. The supported values are *normal* or *low*. The *normal* transfers are scheduled before the *low* priority transfers. The default is *normal*.

### **[`-ignore-atime {true|false}`] - Ignore File Access Time**

This parameter applies only to extended data protection (XDP) relationships. It specifies whether incremental transfers will ignore files which have only their access time changed. The supported values are *true* or *false*. The default is *false*.

### **[`-restart {always|never|default}`] - Restart Behavior**

This parameter applies only to data protection relationships. It defines the behavior of SnapMirror if an interrupted transfer exists. The supported values are *always*, *never*, or *default*. If the value is set to *always*, an interrupted SnapMirror transfer always restarts provided it has a restart checkpoint and the conditions are the same as they were before the transfer was interrupted. In addition, a new SnapMirror Snapshot copy is created which will then be transferred. If the value is set to *never*, an interrupted SnapMirror transfer will never restart, even if a restart checkpoint exists. A new SnapMirror Snapshot copy will still be created and transferred. Data ONTAP version 8.2 will interpret a value of *default* as being the same as *always*. Vault transfers will always resume based on a restart checkpoint, provided the Snapshot copy still exists on the source volume.

### **[`-is-network-compression-enabled {true|false}`] - Is Network Compression Enabled**

Specifies whether network compression is enabled for transfers. The supported values are *true* or *false*. The default is *false*.

### **[`-rpo <integer>`] - Recovery Point Objective (seconds)**

Specifies the time for recovery point objective, in seconds. This parameter is only supported for a policy of type *continuous*.

### **[`-always-replicate-snapshots {true|false}`] - This prioritizes replication of app-consistent snapshots over synchronous replication**

If this parameter is set to *true*, it specifies that SnapMirror Synchronous relationships will lose the zero RPO protection upon failure in replicating application created snapshots. The default value is *false*.

### **[`-common-snapshot-schedule <text>`] - Common Snapshot Copy Creation Schedule for SnapMirror Synchronous (privilege: advanced)**

Specifies the common Snapshot creating schedule. This parameter is only supported for Snapmirror Synchronous relationships.

### **[`-are-data-ops-sequentially-split {true|false}`] - Is Sequential Splitting of Data Operations Enabled? (privilege: advanced)**

This parameter specifies whether I/O, such as write, copy-offload and punch-holes, are split sequentially, rather than being run in parallel on the source and destination. Splitting the I/O sequentially will make the system more robust, and less prone to I/O errors. Starting 9.11.1, enabling this feature improves the performance when the workload is NAS based and is metadata heavy. However, it will make the IO performance slower for large file workloads like LUNs, databases, virtualization containers, etc. The default value of parameter `-sequential-split-data-ops` is *false*. The parameter `-are-data-ops-sequentially-split` should only be used if frequent I/O timeout or "OutOfSync" has happened. Changes made by the `snapmirror policy modify -sequential-split-data-ops` command do not take effect until the next resync. Changes do not affect resync or initialize operations that have started and have not finished yet. The parameter `-are-data-ops-sequentially-split` requires an effective cluster version of Data ONTAP 9.6.0 or later on both the source and destination clusters.

### **[`-sequential-split-op-timeout-secs <integer>`] - Sequential Split Op Timeout in Seconds (privilege: advanced)**

This parameter specifies the op timeout value used when the splitting mode is sequential. This parameter is



used only for Sync relationships. Supported values are from 15 to 25 seconds. The default value of the parameter `-sequential-split-op-timeout` is `15 seconds`. Changes made by the `snapmirror policy modify -sequential-split-op-timeout-secs` command do not take effect until the next resync. Changes do not affect resync or initialize operations that have started and have not finished yet.

#### **[`-disable-fast-resync {true|false}`] - Disable Fast Resync (privilege: advanced)**

Specifies whether fast resync for a SnapMirror Synchronous relationship is disabled or not. The default value for this parameter when a policy is created is `false`.

#### **[`-discard-configs <network>,...`] - Configurations Not Replicated During Identity Preserve Vserver DR**

Specifies the configuration to be dropped during replication. The supported values are:

- `network` - Drops network interfaces, routes, and kerberos configuration.

This parameter is supported only for policies of type `async-mirror` and applicable only for identity-preserve Vserver SnapMirror relationships.

#### **[`-transfer-schedule-name <text>`] - Transfer Schedule Name**

This optional parameter specifies the schedule which is used to update the SnapMirror relationships.

#### **[`-throttle <throttleType>`] - Throttle (KB/sec)**

This optional parameter limits the network bandwidth used for transfers. It configures for the relationships the maximum rate (in Kbytes/sec) at which data can be transferred. If no throttle is configured, by default the SnapMirror relationships fully utilize the network bandwidth available. You can also configure the relationships to fully use the network bandwidth available by explicitly setting the throttle to `unlimited` or `0`. The minimum effective throttle value is four Kbytes/sec, so if you specify a throttle value between `1` and `4`, it will be treated as `4`. For FlexGroup volume relationships, the throttle value is applied individually to each constituent relationship.

## Examples

The following example changes the "transfer-priority" and the "comment" text of a snapmirror policy named `TieredBackup` on Vserver `vs0.example.com`:

```
vs0.example.com::> snapmirror policy modify -vserver vs0.example.com
-policy TieredBackup -transfer-priority low -comment "Use for tiered
backups"
```

## Related Links

- [snapmirror policy modify-rule](#)

## snapmirror policy remove-rule

Remove a rule from SnapMirror policy

**Availability:** This command is available to `cluster` and `Vserver` administrators at the `admin` privilege level.

## Description

The `snapmirror policy remove-rule` command removes a rule from a SnapMirror policy. On the destination of a SnapMirror relationship with `snapmirror policy` of type `vault` or `mirror-vault`, all Snapshot copies with a SnapMirror label matching the rule being removed are no longer processed by SnapMirror and might need to be deleted manually. A `snapmirror policy` of type `vault` must have at least one rule if that policy is associated with a SnapMirror relationship. A SnapMirror policy with rules must have at least one rule without a schedule.

## Parameters

**-vserver <vserver name> - Vserver Name**

Specifies the Vserver for the SnapMirror policy.

**-policy <sm\_policy> - SnapMirror Policy Name**

Specifies the SnapMirror policy name.

**-snapmirror-label <text> - Snapshot Copy Label**

This parameter specifies the rule that is removed from the SnapMirror policy.

The rule for SnapMirror label `sm_created` cannot be removed from SnapMirror policies of type `async-mirror` or `mirror-vault`.

## Examples

The following example removes a rule named `nightly` from a SnapMirror policy named `TieredBackup` on Vserver `vs0.example.com`:

```
vs0.example.com::> snapmirror policy remove-rule -vserver vs0.example.com
-policy TieredBackup -snapmirror-label nightly
```

## snapmirror policy show

Show SnapMirror policies

**Availability:** This command is available to `cluster` and `Vserver` administrators at the `admin` privilege level.

## Description

The `snapmirror policy show` command displays the following information about SnapMirror policies:

- Vserver Name
- SnapMirror Policy Name
- SnapMirror Policy Type
- Number of Rules in the policy
- Tries
- Transfer Priority
- Comment for the policy

- Individual Rule Names
- Keep value for the Rule
- Total of Keep values across all Rules in the policy

## Parameters

**{ [-fields <fieldname>,...]**

If you specify the `-fields <fieldname>`, ... parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays detailed information about all fields.

**[-vserver <vserver name>] - Vserver Name**

Selects the policies that match this parameter value.

**[-policy <sm\_policy>] - SnapMirror Policy Name**

Selects the policies that match this parameter value.

**[-type {vault|async-mirror|mirror-vault|strict-sync-mirror|sync-mirror|automated-failover|automated-failover-duplex|continuous}] - Snapmirror Policy Type**

Selects the policies that match this parameter value. A policy can be of type `async-mirror`, `vault` or `mirror-vault`.

**[-owner {cluster-admin|vserver-admin}] - Owner of the Policy**

Selects the policies that match this parameter value. A policy can be owned by either the `"Cluster Admin"` or a `"Vserver Admin"`.

**[-comment <text>] - Comment**

Selects the policies that match this parameter value.

**[-tries <integer\_or\_unlimited>] - Tries Limit**

Selects the policies that match this parameter value.

**[-transfer-priority {low|normal}] - Transfer Scheduling Priority**

Selects the policies that match this parameter value.

**[-ignore-atime {true|false}] - Ignore File Access Time**

Selects the policies that match this parameter value.

**[-restart {always|never|default}] - Restart Behavior**

Selects the policies that match this parameter value.

**[-is-network-compression-enabled {true|false}] - Is Network Compression Enabled**

Selects the policies that match this parameter value.

**[-create-snapshot {true|false}] - Create a New Snapshot Copy**

Selects the policies that match this parameter value.

**[-rpo <integer>] - Recovery Point Objective (seconds)**

Selects the policies that match this parameter value.

**[-always-replicate-snapshots {true|false}] - This prioritizes replication of app-consistent snapshots over synchronous replication**

Selects the policies that match this parameter value.

**[-snapmirror-label <text>,...] - Snapshot Copy Label**

Selects the policies that match this parameter value.

**[-keep <text>,...] - Snapshot Copy Retention Count**

Selects the policies that match this parameter value.

**[-preserve {true|false}] - Snapshot Copy Preserve Enabled**

Selects the policies that match this parameter value.

**[-warn <integer>,...] - Warning Threshold Count**

Selects the policies that match this parameter value.

**[-schedule <text>,...] - Snapshot Copy Creation Schedule**

Selects the policies that match this parameter value.

**[-prefix <text>,...] - Snapshot Copy Creation Prefix**

Selects the policies that match this parameter value.

**[-retention-period {{<integer> seconds|minutes|hours|days|months|years} | infinite}] - Snapshot Copy Retention Period**

This optional parameter specifies the duration for which the Snapshot copies associated with this rule will be locked. The parameter is specified as a number followed by a suffix. The valid suffixes are *seconds*, *minutes*, *hours*, *days*, *months*, and *years*. For example, a value of *6months* represents a retention period of 6 months. It can also be set to a special value of *infinite* for infinite retention. The destination volume must have the *snapshot-locking-enabled* parameter set to true for this field to take effect else it will be ignored.

**[-total-rules <integer>] - Total Rules in the Policy**

Selects the policies that match this parameter value.

**[-total-keep <integer>] - Total Retention Count for All Rules in the Policy**

Selects the policies that match this parameter value.

**[-common-snapshot-schedule <text>] - Common Snapshot Copy Creation Schedule for SnapMirror Synchronous (privilege: advanced)**

Selects the policies that match this parameter value.

**[-are-data-ops-sequentially-split {true|false}] - Is Sequential Splitting of Data Operations Enabled? (privilege: advanced)**

Selects the policies that match this parameter value.

**[-sequential-split-op-timeout-secs <integer>] - Sequential Split Op Timeout in Seconds (privilege: advanced)**

Selects the policies that match this parameter value.

**[-disable-fast-resync {true|false}] - Disable Fast Resync (privilege: advanced)**

Selects the policies that match this parameter value.

**[-discard-configs <network>,...] - Configurations Not Replicated During Identity Preserve Vserver DR**

Selects the policies that match this parameter value.

**[-transfer-schedule-name <text>] - Transfer Schedule Name**

Selects the policies that match this parameter value.

**[-throttle <throttleType>] - Throttle (KB/sec)**

Selects the policies that match this parameter value.

## Examples

The following example displays information about all SnapMirror policies:

```
cs::> snapmirror policy show
Vserver Policy          Policy Number      Transfer
Name      Name              Type   Of Rules Tries Priority Comment
-----
cs      DPDefault          async-mirror  2      8  normal  Asynchronous
SnapMirror policy for mirroring all Snapshot copies and the latest active
file system.
  SnapMirror Label: sm_created          Keep:          1
                    all_source_snapshots      1
                    Total Keep:            2

cs      MirrorAllSnapshots async-mirror  2      8  normal  Asynchronous
SnapMirror policy for mirroring all Snapshot copies and the latest active
file system.
  SnapMirror Label: sm_created          Keep:          1
                    all_source_snapshots      1
                    Total Keep:            2

cs      MirrorAndVault    mirror-vault  3      8  normal  A unified
Asynchronous SnapMirror and SnapVault policy for mirroring the latest
active file system and daily and weekly Snapshot copies.
  SnapMirror Label: sm_created          Keep:          1
                    daily                7
                    weekly                52
                    Total Keep:        60
```

```

cs      MirrorLatest      async-mirror  1      8  normal  Asynchronous
SnapMirror policy for mirroring the latest active file system.
  SnapMirror Label: sm_created                                Keep:      1
                                                            Total Keep: 1

vs0.example.com
  TieredBackup      vault      0      8  normal  Use for tiered
backups
  Snapmirror-label: -                                       Keep:      -
                                                            Total Keep: 0

cs  Unified7year      mirror-vault  4      8  normal  Unified SnapMirror
policy with 7year retention.
  SnapMirror Label: sm_created                                Keep:      1
                    daily                                    7
                    weekly                                   52
                    monthly                                  84
                                                            Total Keep: 144

cs      XDPDefault      vault      2      8  normal  Vault policy
with daily and weekly rules.
  SnapMirror Label: daily                                    Keep:      7
                    weekly                                   52
                                                            Total Keep: 59

7 entries were displayed.

```

The following example shows all the policies with the following fields - vserver (default), policy (default) and transfer-priority:

```

cs::> snapmirror policy show -fields transfer-priority
vserver      policy      transfer-priority
-----
cs           DPDefault  normal
cs           MirrorAllSnapshots
              normal
cs           MirrorAndVault
              normal
cs           MirrorLatest
              normal
vs0.example.com
  TieredBackup
              normal
cs           Unified7year
              normal
cs           XDPDefault
              normal
7 entries were displayed.

```

## snapmirror snapshot-owner commands

### snapmirror snapshot-owner create

Add an owner to preserve a Snapshot copy for a SnapMirror mirror-to-vault cascade configuration

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

#### Description

The `snapmirror snapshot-owner create` command adds an owner on the specified Snapshot copy. A Snapshot copy can have at most one owner. An owner can only be added to a Snapshot copy on a read-write volume. The Snapshot copy must have a valid SnapMirror label.



Refer to the *ONTAP Data Protection Guide* for valid use cases to add an owner on a Snapshot copy.

#### Parameters

**-vserver <vserver name> - Vserver Name**

This parameter specifies the Vserver on which the volume is located.

**-volume <volume name> - Volume Name**

This parameter specifies the name of the volume.

### **-snapshot <snapshot name> - Snapshot Copy Name**

This parameter specifies the name of the Snapshot copy.

### **[-owner <owner name>] - Snapshot Copy Owner Name**

This parameter specifies the name of the owner for the Snapshot copy. The owner name can be made up of the characters A to Z, a to z, 0 to 9, and "\_". The name can be up to 32 characters in length. When not specified, an owner will be added with a system-generated default name.

## **Examples**

The following example adds owner *app1* on Snapshot copy *snap1* on volume *vol1* in Vserver *vs0.example.com*.

```
cluster1::> snapmirror snapshot-owner create -vserver vs0.example.com
           -volume vol1 -snapshot snap1 -owner app1
```

The following example adds a default owner on Snapshot copy *snap2* on volume *vol1* in Vserver *vs0.example.com*.

```
cluster1::> snapmirror snapshot-owner create -vserver vs0.example.com
           -volume vol1 -snapshot snap2
```

## **snapmirror snapshot-owner delete**

Delete an owner used to preserve a Snapshot copy for a SnapMirror mirror-to-vault cascade configuration

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### **Description**

The `snapmirror snapshot-owner delete` command removes an owner on the specified Snapshot copy, which was added using the [snapmirror snapshot-owner create](#) command.

### **Parameters**

#### **-vserver <vserver name> - Vserver Name**

This parameter specifies the Vserver on which the volume is located.

#### **-volume <volume name> - Volume Name**

This parameter specifies the name of the volume.

#### **-snapshot <snapshot name> - Snapshot Copy Name**

This parameter specifies the name of the Snapshot copy.

#### **[-owner <owner name>] - Snapshot Copy Owner Name**

This parameter specifies the name of the owner for the Snapshot copy. When not specified, the owner with



the system-generated default name will be removed.

## Examples

The following example removes owner *app1* on Snapshot copy *snap1* on volume *vol1* in Vserver *vs0.example.com*.

```
cluster1::> snapmirror snapshot-owner delete -vserver vs0.example.com
             -volume vol1 -snapshot snap1 -owner app1
```

The following example removes the default owner on Snapshot copy *snap2* on volume *vol1* in Vserver *vs0.example.com*.

```
cluster1::> snapmirror snapshot-owner delete -vserver vs0.example.com
             -volume vol1 -snapshot snap2
```

## Related Links

- [snapmirror snapshot-owner create](#)

## snapmirror snapshot-owner show

### Display Snapshot Copies with Owners

**Availability:** This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

### Description

The `snapmirror snapshot-owner show` command is used to list all Snapshot copies with owners that were added using the [snapmirror snapshot-owner create](#) command.

### Parameters

{ [-fields <fieldname>,...]

If this parameter is specified, the command displays information about the specified fields.

| [-instance ] }

If this parameter is specified, the command displays detailed information about all fields.

**-vserver <vserver name> - Vserver Name**

This parameter specifies the Vserver on which the volume is located.

**-volume <volume name> - Volume Name**

This parameter specifies the name of the volume.

**[-snapshot <snapshot name>] - Snapshot Copy Name**

If this parameter is specified, the command displays the owner name for the specified Snapshot copy.

## Examples

The following example lists all Snapshot copies with owners on volume `vol1` in Vserver `vs0`. The system-generated default owner name is displayed as "-".

```
cluster1::> snapmirror snapshot-owner show
      -vserver vs0.example.com -volume vol1
Vserver  Volume          Snapshot          Owner Names
-----  -
vs0.example.com
         vol1          snap2            -
                        snap1            appl
```

The following example displays the owner name for Snapshot copy `snap1` on volume `vol1` in Vserver `vs0.example.com`.

```
cluster1::> snapmirror snapshot-owner show
      -vserver vs0.example.com -volume vol1 -snapshot snap1
Vserver: vs0.example.com
      Volume: vol1
      Snapshot: snap1
      Owner Names: appl
```

## Related Links

- [snapmirror snapshot-owner create](#)

## Copyright information

Copyright © 2024 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.