



# 管理 **SnapMirror** 卷复制

## ONTAP 9

NetApp  
April 24, 2024

# 目录

- 管理 SnapMirror 卷复制 ..... 1
  - SnapMirror 复制工作流 ..... 1
  - 一步配置复制关系 ..... 2
  - 一次配置一个复制关系 ..... 4
  - 将现有 DP 类型关系转换为 XDP ..... 16
  - 转换 SnapMirror 关系的类型 ..... 21
  - 转换 SnapMirror 同步关系的模式 ..... 23
  - 创建和删除SnapMirror故障转移测试卷 ..... 24
  - 从 SnapMirror 灾难恢复目标卷提供数据 ..... 25
  - 从 SnapMirror 目标卷还原文件 ..... 30
  - 手动更新复制关系 ..... 34
  - 重新同步复制关系 ..... 35
  - 删除卷复制关系 ..... 35
  - 管理存储效率 ..... 36
  - 使用 SnapMirror 全局限制 ..... 38

# 管理 SnapMirror 卷复制

## SnapMirror 复制 workflow

SnapMirror 提供三种类型的数据保护关系： SnapMirror 灾难恢复，归档（以前称为 SnapVault ）和统一复制。您可以按照相同的基本 workflow 来配置每种类型的关系。

从 ONTAP 9.1.1 中的常规可用性开始， SnapMirror 业务连续性（ SM-BC ）可提供零恢复时间目标（零 RTO ）或透明应用程序故障转移（ TAF ），以便在 SAN 环境中自动对业务关键型应用程序进行故障转移。在配置两个 AFF 集群或两个全闪存 SAN 阵列 (ASA) 集群时、支持 SM-BC。

["NetApp 文档： SnapMirror 业务连续性"](#)

对于每种类型的 SnapMirror 数据保护关系， workflow 都是相同的：创建目标卷，创建作业计划，指定策略，创建和初始化关系。

从 ONTAP 9.3 开始、您可以使用 `snapmirror protect` 命令、一步配置数据保护关系。即使您使用，也是如此 `snapmirror protect`，您需要了解 workflow 中的每个步骤。



## 一步配置复制关系

从ONTAP 9.3开始、您可以使用 `snapmirror protect` 命令、一步配置数据保护关系。您可以指定要复制的卷列表，目标集群上的 SVM ，作业计划和 SnapMirror 策略。`snapmirror protect` 完成其余操作。

您需要的内容

- 源和目标集群以及 SVM 必须建立对等关系。

"集群和 SVM 对等"

- 目标卷上的语言必须与源卷上的语言相同。

关于此任务

。 `snapmirror protect` 命令可选择与指定SVM关联的聚合。如果没有与 SVM 关联的聚合，则它会从集群中的所有聚合中进行选择。根据聚合上的可用空间量和卷数选择聚合。

。 `snapmirror protect` 然后、命令将执行以下步骤：

- 为要复制的卷列表中的每个卷创建一个具有适当类型和预留空间量的目标卷。
- 配置适合您指定策略的复制关系。
- 初始化关系。

目标卷的名称格式为 `source_volume_name_dst`。如果与现有名称发生冲突，此命令会在卷名称后附加一个数字。您可以在命令选项中指定前缀和 / 或后缀。后缀将替换系统提供的 `dst` 后缀。

在 ONTAP 9.3 及更早版本中，目标卷最多可包含 251 个 Snapshot 副本。在 ONTAP 9.4 及更高版本中，目标卷最多可以包含 1019 个 Snapshot 副本。



初始化可能非常耗时。 `snapmirror protect` 在作业完成之前、不会等待初始化完成。因此、您应使用 `snapmirror show` 命令、而不是 `job show` 命令以确定初始化完成的时间。

从ONTAP 9.5开始、可以使用创建SnapMirror同步关系 `snapmirror protect` 命令：

#### 步骤

##### 1. 一步创建并初始化复制关系：

在运行此命令之前、必须将尖括号中的变量替换为所需的值。

```
snapmirror protect -path-list <SVM:volume> -destination-vserver  
<destination_SVM> -policy <policy> -schedule <schedule> -auto-initialize  
<true|false> -destination-volume-prefix <prefix> -destination-volume  
-suffix <suffix>
```



您必须从目标 SVM 或目标集群运行此命令。。 `-auto-initialize` 选项默认为“true”。

以下示例将使用默认值创建并初始化SnapMirror灾难恢复关系 `MirrorAllSnapshots` 策略：

```
cluster_dst::> snapmirror protect -path-list svm1:volA, svm1:volB  
-destination-vserver svm_backup -policy MirrorAllSnapshots -schedule  
replication_daily
```



如果愿意，您可以使用自定义策略。有关详细信息，请参见 ["创建自定义复制策略"](#)。

以下示例将使用默认值创建并初始化SnapVault关系 `XDPEndpointDefault` 策略：

```
cluster_dst:> snapmirror protect -path-list svm1:volA, svm1:volB
-destination-vserver svm_backup -policy XDPDefault -schedule
replication_daily
```

以下示例将使用默认值创建并初始化统一复制关系 MirrorAndVault 策略：

```
cluster_dst:> snapmirror protect -path-list svm1:volA, svm1:volB
-destination-vserver svm_backup -policy MirrorAndVault
```

以下示例将使用默认值创建并初始化SnapMirror同步关系 Sync 策略：

```
cluster_dst:> snapmirror protect -path-list svm1:volA, svm1:volB
-destination-vserver svm_sync -policy Sync
```



对于 SnapVault 和统一复制策略，您可能会发现，定义一个计划来为目标上最后一次传输的 Snapshot 副本创建副本会很有用。有关详细信息，请参见 ["定义在目标上创建本地副本的计划"](#)。

完成后

使用 `snapmirror show` 命令以验证是否已创建SnapMirror关系。有关完整的命令语法，请参见手册页。

## 一次配置一个复制关系

### 创建目标卷

您可以使用 `volume create` 命令创建目标卷。目标卷的大小应等于或大于源卷。

步骤

1. 创建目标卷：

```
volume create -vserver SVM -volume volume -aggregate aggregate -type DP -size
size
```

有关完整的命令语法，请参见手册页。

以下示例将创建一个名为的2 GB目标卷 volA\_dst：

```
cluster_dst:> volume create -vserver SVM_backup -volume volA_dst
-aggregate node01_aggr -type DP -size 2GB
```

## 创建复制作业计划

您可以使用 `job schedule cron create` 命令以创建复制作业计划。作业计划用于确定 SnapMirror 何时自动更新分配了该计划的数据保护关系。

关于此任务

您可以在创建数据保护关系时分配作业计划。如果不分配作业计划，则必须手动更新此关系。

### 步骤

#### 1. 创建作业计划：

```
job schedule cron create -name job_name -month month -dayofweek day_of_week  
-day day_of_month -hour hour -minute minute
```

适用于 `-month`，`-dayofweek`，和 `-hour`，您可以指定 `all` 以分别每月、每周的某一天和每小时运行此作业。

从 ONTAP 9.10.1 开始，您可以在作业计划中包含 Vserver：

```
job schedule cron create -name job_name -vserver Vserver_name -month month  
-dayofweek day_of_week -day day_of_month -hour hour -minute minute
```



对于卷 SnapMirror 关系中的 FlexVol 卷、支持的最小计划(RPO)为5分钟。对于卷 SnapMirror 关系中的 FlexGroup 卷、支持的最小计划(RPO)为30分钟。

以下示例将创建一个名为的作业计划 `my_weekly` 在星期六凌晨3:00运行：

```
cluster_dst::> job schedule cron create -name my_weekly -dayofweek  
"Saturday" -hour 3 -minute 0
```

## 自定义复制策略

### 创建自定义复制策略

如果关系的默认策略不适用，您可以创建自定义复制策略。例如，您可能希望在网络传输中压缩数据，或者修改 SnapMirror 尝试传输 Snapshot 副本的次数。

您可以在创建复制关系时使用默认策略或自定义策略。对于自定义归档（以前称为 SnapVault）或统一复制策略，您必须定义一个或多个 *rules* 来确定初始化和更新期间传输的 Snapshot 副本。您可能还需要定义一个计划，以便在目标上创建本地 Snapshot 副本。

复制策略的 *policy type* 决定了它支持的关系类型。下表显示了可用的策略类型。

Policy type	关系类型
异步镜像	SnapMirror 灾难恢复

存储	SnapVault
镜像存储	统一复制
strict-sync-mirror	StrictSync 模式下的 SnapMirror 同步（从 ONTAP 9.5 开始受支持）
sync-mirror	同步模式下的 SnapMirror 同步（从 ONTAP 9.5 开始受支持）



创建自定义复制策略时，最好在默认策略之后对策略进行建模。

## 步骤

### 1. 创建自定义复制策略：

```
snapmirror policy create -vserver SVM -policy policy -type async-
mirror|vault|mirror-vault|strict-sync-mirror|sync-mirror -comment comment
-tries transfer_tries -transfer-priority low|normal -is-network-compression
-enabled true|false
```

有关完整的命令语法，请参见手册页。

从ONTAP 9.5开始、您可以使用指定为SnapMirror同步关系创建通用Snapshot副本计划的计划 `-common-snapshot-schedule` 参数。默认情况下， SnapMirror 同步关系的通用 Snapshot 副本计划为一小时。您可以为 SnapMirror 同步关系的 Snapshot 副本计划指定一个介于 30 分钟到两小时之间的值。

以下示例将为 SnapMirror 灾难恢复创建一个自定义复制策略，以便为数据传输启用网络压缩：

```
cluster_dst::> snapmirror policy create -vserver svm1 -policy
DR_compressed -type async-mirror -comment "DR with network compression
enabled" -is-network-compression-enabled true
```

以下示例将为 SnapVault 创建自定义复制策略：

```
cluster_dst::> snapmirror policy create -vserver svm1 -policy
my_snapvault -type vault
```

以下示例将为统一复制创建自定义复制策略：

```
cluster_dst::> snapmirror policy create -vserver svm1 -policy my_unified
-type mirror-vault
```

以下示例将在 StrictSync 模式下为 SnapMirror 同步关系创建自定义复制策略：



```
cluster_dst:> snapmirror policy create -vserver svml -policy
my_strictsync -type strict-sync-mirror -common-snapshot-schedule
my_sync_schedule
```

完成后

对于 "vault" 和 "mirror-vault" 策略类型，您必须定义规则来确定初始化和更新期间传输的 Snapshot 副本。

使用 `snapmirror policy show` 命令以验证是否已创建 SnapMirror 策略。有关完整的命令语法，请参见手册页。

为策略定义规则

对于策略类型为 "vault" 或 "mirror-vault" 的自定义策略，您必须至少定义一个规则来确定初始化和更新期间传输的 Snapshot 副本。您还可以为策略类型为 "vault" 或 "mirror-vault" 的默认策略定义规则。

关于此任务

每个策略类型为 "vault" 或 "mirror-vault" 的策略都必须具有一个规则，用于指定要复制的 Snapshot 副本。例如，规则 "bi-monthly" 表示仅应复制分配了 SnapMirror 标签 "bi-monthly" 的 Snapshot 副本。在源上配置 Snapshot 策略时，您可以指定 SnapMirror 标签。

每个策略类型都与一个或多个系统定义的规则相关联。指定策略类型时，系统会自动为策略分配这些规则。下表显示了系统定义的规则。

系统定义的规则	用于策略类型	结果
sm_created	异步镜像，镜像存储，同步，StrictSync	在初始化和更新时传输 SnapMirror 创建的 Snapshot 副本。
all_source_snapshots	异步镜像	初始化和更新时会传输源上的新 Snapshot 副本。
每天	存储，镜像存储	在初始化和更新时传输源上 SnapMirror 标签为 daily 的新 Snapshot 副本。
每周	存储，镜像存储	在初始化和更新时，会传输源上 SnapMirror 标签为 "weekly" 的新 Snapshot 副本。
每月	镜像存储	在初始化和更新时，会传输源上 SnapMirror 标签为 "m月 日" 的新 Snapshot 副本。

应用程序一致	Sync , StrictSync	源上 SnapMirror 标签为 "APP_consistent" 的 Snapshot 副本会同步复制到目标。从 ONTAP 9.7 开始支持。
--------	-------------------	--

除了 "async-mirror" 策略类型之外，您还可以根据需要为默认或自定义策略指定其他规则。例如：

- 对于默认值 MirrorAndVault 策略中、您可以创建一个名为 "bi-monthly" 的规则、以匹配源上具有 "bi-monthly" SnapMirror 标签的 Snapshot 副本。
- 对于策略类型为 mirror-vault 的自定义策略，您可以创建一个名为 "bi-weekly" 的规则，以匹配源上具有 "bi-weekly" SnapMirror 标签的 Snapshot 副本。

## 步骤

### 1. 为策略定义规则：

```
snapmirror policy add-rule -vserver SVM -policy policy_for_rule -snapmirror
-label snapmirror-label -keep retention_count
```

有关完整的命令语法，请参见手册页。

以下示例将添加一个 SnapMirror 标签为的规则 bi-monthly 设置为默认值 MirrorAndVault 策略：

```
cluster_dst:> snapmirror policy add-rule -vserver svm1 -policy
MirrorAndVault -snapmirror-label bi-monthly -keep 6
```

以下示例将添加一个 SnapMirror 标签为的规则 bi-weekly 自定义 my\_snapvault 策略：

```
cluster_dst:> snapmirror policy add-rule -vserver svm1 -policy
my_snapvault -snapmirror-label bi-weekly -keep 26
```

以下示例将添加一个 SnapMirror 标签为的规则 app\_consistent 自定义 Sync 策略：

```
cluster_dst:> snapmirror policy add-rule -vserver svm1 -policy Sync
-snapmirror-label app_consistent -keep 1
```

然后，您可以从与此 SnapMirror 标签匹配的源集群复制 Snapshot 副本：

```
cluster_src:> snapshot create -vserver vs1 -volume vol1 -snapshot
snapshot1 -snapmirror-label app_consistent
```

定义在目标上创建本地副本的计划

对于 SnapVault 和统一复制关系，您可以通过在目标上为上次传输的 Snapshot 副本创建一份副本来防止更新后的 Snapshot 副本损坏。无论源上的保留规则如何，此 "local copy" 都会保留，这样，即使最初由 SnapMirror 传输的 Snapshot 在源上不再可用，目标上也会提供该 Snapshot 的副本。

关于此任务

您可以在中指定创建本地副本的计划 `-schedule` 的选项 `snapmirror policy add-rule` 命令：

步骤

1. 定义在目标上创建本地副本的计划：

```
snapmirror policy add-rule -vserver SVM -policy policy_for_rule -snapmirror  
-label snapmirror-label -schedule schedule
```

有关完整的命令语法，请参见手册页。有关如何创建作业计划的示例，请参见 ["创建复制作业计划"](#)。

以下示例将创建本地副本的计划添加到默认值 `MirrorAndVault` 策略：

```
cluster_dst::> snapmirror policy add-rule -vserver svml -policy  
MirrorAndVault -snapmirror-label my_monthly -schedule my_monthly
```

以下示例将创建本地副本的计划添加到自定义中 `my_unified` 策略：

```
cluster_dst::> snapmirror policy add-rule -vserver svml -policy  
my_unified -snapmirror-label my_monthly -schedule my_monthly
```

## 创建复制关系

主存储中的源卷与二级存储中的目标卷之间的关系称为 `_data protection` 关系。\_您可以使用 `snapmirror create` 用于创建 SnapMirror 灾难恢复、SnapVault 或统一复制数据保护关系的命令。

您需要的内容

- 源和目标集群以及 SVM 必须建立对等关系。

["集群和 SVM 对等"](#)

- 目标卷上的语言必须与源卷上的语言相同。

关于此任务

在 ONTAP 9.3 之前，在 DP 模式下调用的 SnapMirror 和在 XDP 模式下调用的 SnapMirror 使用不同的复制引擎，对版本依赖性采用不同的方法：

- 在 DP 模式下调用的 SnapMirror 使用了一个 `_version-dependent_` 复制引擎，其中，主存储和二级存储上的 ONTAP 版本必须相同：

```
cluster_dst::> snapmirror create -type DP -source-path ... -destination
-path ...
```

- 在 XDP 模式下调用的 SnapMirror 使用 `_version-flexible` 复制引擎，该引擎支持主存储和二级存储上的不同 ONTAP 版本：

```
cluster_dst::> snapmirror create -type XDP -source-path ...
-destination-path ...
```

随着性能的提高，版本灵活的 SnapMirror 的显著优势大于使用版本相关模式获得的复制吞吐量的轻微优势。因此，从 ONTAP 9.3 开始，XDP 模式已成为新的默认模式，在命令行或新脚本或现有脚本中调用的任何 DP 模式都会自动转换为 XDP 模式。

现有关系不受影响。如果某个关系的类型已经是 DP，则它将继续为 DP 类型。下表显示了您可能会遇到的行为。

指定项	类型为 ...	默认策略（如果未指定策略）为 ...
数据	XDP	MirrorAllSnapshots （ SnapMirror 灾难恢复）
无	XDP	MirrorAllSnapshots （ SnapMirror 灾难恢复）
XDP	XDP	XDPDefault （ SnapVault ）

另请参见下面操作步骤中的示例。

转换的唯一例外情况如下：

- SVM 数据保护关系仍默认为 DP 模式。

明确指定 XDP 以使用默认值获取 XDP 模式 `MirrorAllSnapshots` 策略。

- 负载共享数据保护关系仍默认为 DP 模式。
- SnapLock 数据保护关系仍默认为 DP 模式。
- 如果设置了以下集群范围选项，则显式调用 DP 仍会默认为 DP 模式：

```
options replication.create_data_protection_rels.enable on
```

如果不显式调用 DP，则会忽略此选项。

在 ONTAP 9.3 及更早版本中，目标卷最多可包含 251 个 Snapshot 副本。在 ONTAP 9.4 及更高版本中，目标

卷最多可以包含 1019 个 Snapshot 副本。

从 ONTAP 9.5 开始，支持 SnapMirror 同步关系。

#### 步骤

##### 1. 从目标集群中，创建复制关系：

在运行此命令之前，必须将尖括号中的变量替换为所需的值。

```
snapmirror create -source-path <SVM:volume> -destination-path  
<SVM:volume> -type <DP|XDP> -schedule <schedule> -policy <policy>
```

有关完整的命令语法，请参见手册页。



。 schedule 参数在创建 SnapMirror 同步关系时不适用。

以下示例将使用默认值创建 SnapMirror 灾难恢复关系 MirrorLatest 策略：

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination  
-path svm_backup:volA_dst -type XDP -schedule my_daily -policy  
MirrorLatest
```

以下示例将使用默认值创建 SnapVault 关系 XDPDefault 策略：

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination  
-path svm_backup:volA_dst -type XDP -schedule my_daily -policy  
XDPDefault
```

以下示例将使用默认值创建统一复制关系 MirrorAndVault 策略：

```
cluster_dst:> snapmirror create -source-path svm1:volA -destination-path  
svm_backup:volA_dst -type XDP -schedule my_daily -policy MirrorAndVault
```

以下示例将使用自定义创建统一复制关系 my\_unified 策略：

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination  
-path svm_backup:volA_dst -type XDP -schedule my_daily -policy  
my_unified
```

以下示例将使用默认值创建 SnapMirror 同步关系 Sync 策略：

```
cluster_dst:> snapmirror create -source-path svm1:volA -destination  
-path svm_backup:volA_dst -type XDP -policy Sync
```

以下示例将使用默认值创建SnapMirror同步关系 StrictSync 策略：

```
cluster_dst:> snapmirror create -source-path svm1:volA -destination  
-path svm_backup:volA_dst -type XDP -policy StrictSync
```

以下示例将创建 SnapMirror 灾难恢复关系。如果DP类型自动转换为XDP且未指定任何策略、则此策略将默认为 MirrorAllSnapshots 策略：

```
cluster_dst:> snapmirror create -source-path svm1:volA -destination  
-path svm_backup:volA_dst -type DP -schedule my_daily
```

以下示例将创建 SnapMirror 灾难恢复关系。如果未指定任何类型或策略、则此策略将默认为 MirrorAllSnapshots 策略：

```
cluster_dst:> snapmirror create -source-path svm1:volA -destination  
-path svm_backup:volA_dst -schedule my_daily
```

以下示例将创建 SnapMirror 灾难恢复关系。如果未指定任何策略、则此策略默认为 XDPDefault 策略：

```
cluster_dst:> snapmirror create -source-path svm1:volA -destination  
-path svm_backup:volA_dst -type XDP -schedule my_daily
```

以下示例将使用预定义策略创建SnapMirror同步关系 SnapCenterSync：

```
cluster_dst:> snapmirror create -source-path svm1:volA -destination  
-path svm_backup:volA_dst -type XDP -policy SnapCenterSync
```



预定义的策略 SnapCenterSync 属于类型 Sync。此策略会复制使用创建的任何Snapshot副本 snapmirror-label "应用一致"。

完成后

使用 `snapmirror show` 命令以验证是否已创建SnapMirror关系。有关完整的命令语法，请参见手册页。

相关信息

- ["创建和删除SnapMirror故障转移测试卷"](#)。

在 **ONTAP** 中执行此操作的其他方法

要执行以下任务，请执行以下操作 ...	查看此内容 ...
重新设计的 System Manager（适用于 ONTAP 9.7 及更高版本）	<a href="#">"配置镜像和存储"</a>
System Manager 经典版（适用于 ONTAP 9.7 及更早版本）	<a href="#">"使用 SnapVault 进行卷备份概述"</a>

## 初始化复制关系

对于所有关系类型，初始化将执行 *baseline transfer*：它会为源卷创建 Snapshot 副本，然后将该副本及其引用的所有数据块传输到目标卷。否则，传输内容取决于策略。

您需要的内容

源和目标集群以及 SVM 必须建立对等关系。

["集群和 SVM 对等"](#)

关于此任务

初始化可能非常耗时。您可能希望在非高峰时段运行基线传输。

从 ONTAP 9.5 开始，支持 SnapMirror 同步关系。

步骤

### 1. 初始化复制关系：

```
snapmirror initialize -source-path SVM:volume|cluster://SVM/volume, ...  
-destination-path SVM:volume|cluster://SVM/volume, ...
```

有关完整的命令语法，请参见手册页。



您必须从目标 SVM 或目标集群运行此命令。

以下示例将初始化源卷之间的关系 volA 开启 svm1 和目标卷 volA\_dst 开启 svm\_backup：

```
cluster_dst::> snapmirror initialize -source-path svm1:volA -destination  
-path svm_backup:volA_dst
```

## 示例：配置存储 - 存储级联

示例将具体说明如何一次一步配置复制关系。您可以使用此示例中配置的存储 - 存储级联部署来保留超过 251 个标有 my-weekly 的 Snapshot 副本。

您需要的内容

- 源和目标集群以及 SVM 必须建立对等关系。

- 您必须运行 ONTAP 9.2 或更高版本。早期的 ONTAP 版本不支持存储 - 存储级联。

关于此任务

此示例假设以下条件：

- 您已在源集群上为 Snapshot 副本配置 SnapMirror 标签 my-daily，my-weekly 和 my-monthly。
- 您已在二级和三级目标集群上配置名为 "Vola" 的目标卷。
- 您已在二级和三级目标集群上配置名为 my\_SnapVault 的复制作业计划。

此示例显示了如何基于两个自定义策略创建复制关系：

- "snapvault\_secondary" 策略会在二级目标集群上保留 7 个每日 Snapshot 副本，52 个每周 Snapshot 副本和 180 个每月 Snapshot 副本。
- "snapvault\_tertiary policy" 会在三级目标集群上保留 250 个每周 Snapshot 副本。

步骤

1. 在二级目标集群上，创建 snapvault\_secondary 策略：

```
cluster_secondary::> snapmirror policy create -policy snapvault_secondary  
-type vault -comment "Policy on secondary for vault to vault cascade" -vserver  
svm_secondary
```

2. 在二级目标集群上，为策略定义 my-daily 规则：

```
cluster_secondary::> snapmirror policy add-rule -policy snapvault_secondary  
-snapmirror-label my-daily -keep 7 -vserver svm_secondary
```

3. 在二级目标集群上，为策略定义 my-weekly 规则：

```
cluster_secondary::> snapmirror policy add-rule -policy snapvault_secondary  
-snapmirror-label my-weekly -keep 52 -vserver svm_secondary
```

4. 在二级目标集群上，为策略定义 my-monthly 规则：

```
cluster_secondary::> snapmirror policy add-rule -policy snapvault_secondary  
-snapmirror-label my-monthly -keep 180 -vserver svm_secondary
```

5. 在二级目标集群上，验证策略：

```
cluster_secondary::> snapmirror policy show snapvault_secondary -instance
```



```

Vserver: svm_secondary
SnapMirror Policy Name: snapvault_secondary
SnapMirror Policy Type: vault
Policy Owner: cluster-admin
Tries Limit: 8
Transfer Priority: normal
Ignore accesstime Enabled: false
Transfer Restartability: always
Network Compression Enabled: false
Create Snapshot: false
Comment: Policy on secondary for vault to vault
cascade
Total Number of Rules: 3
Total Keep: 239
Rules: SnapMirror Label      Keep  Preserve Warn
Schedule Prefix
-----
-----
my-daily              7  false      0  -
-
my-weekly             52  false      0  -
-
my-monthly            180  false      0  -
-

```

#### 6. 在二级目标集群上，创建与源集群的关系：

```
cluster_secondary::> snapmirror create -source-path svm_primary:volA
-destination-path svm_secondary:volA -type XDP -schedule my_snapvault -policy
snapvault_secondary
```

#### 7. 在二级目标集群上，初始化与源集群的关系：

```
cluster_secondary::> snapmirror initialize -source-path svm_primary:volA
-destination-path svm_secondary:volA
```

#### 8. 在三级目标集群上，创建 snapvault\_tertiary 策略：

```
cluster_tertiary::> snapmirror policy create -policy snapvault_tertiary -type
vault -comment "Policy on tertiary for vault to vault cascade" -vserver
svm_tertiary
```

#### 9. 在三级目标集群上，为策略定义 my-weekly 规则：

```
cluster_tertiary::> snapmirror policy add-rule -policy snapvault_tertiary
-snapmirror-label my-weekly -keep 250 -vserver svm_tertiary
```

10. 在三级目标集群上，验证策略：

```
cluster_tertiary::> snapmirror policy show snapvault_tertiary -instance
```

```

                Vserver: svm_tertiary
SnapMirror Policy Name: snapvault_tertiary
SnapMirror Policy Type: vault
                Policy Owner: cluster-admin
                Tries Limit: 8
                Transfer Priority: normal
Ignore accesstime Enabled: false
                Transfer Restartability: always
Network Compression Enabled: false
                Create Snapshot: false
                Comment: Policy on tertiary for vault to vault
cascade
                Total Number of Rules: 1
                        Total Keep: 250
                                Rules: SnapMirror Label          Keep  Preserve Warn
Schedule Prefix
-----
-----
                                my-weekly          250   false      0  -
-
```

11. 在三级目标集群上，创建与二级集群的关系：

```
cluster_tertiary::> snapmirror create -source-path svm_secondary:volA
-destination-path svm_tertiary:volA -type XDP -schedule my_snapvault -policy
snapvault_tertiary
```

12. 在三级目标集群上，初始化与二级集群的关系：

```
cluster_tertiary::> snapmirror initialize -source-path svm_secondary:volA
-destination-path svm_tertiary:volA
```

## 将现有 DP 类型关系转换为 XDP

如果要升级到ONTAP 9.12.1或更高版本、则必须在升级之前将DP类型关系转换为XDP。ONTAP 9.12.1及更高版本不支持DP类型关系。您可以轻松地将现有 DP 类型关系转换为 XDP，以利用版本灵活的 SnapMirror。

### 关于此任务

- SnapMirror 不会自动将现有的 DP 类型关系转换为 XDP。要转换此关系，您需要中断并删除现有关系，创建新的 XDP 关系并重新同步此关系。有关背景信息，请参见 ["XDP 取代 DP 作为 SnapMirror 默认设置"](#)。

- 在规划转换时，您应了解 XDP SnapMirror 关系的后台准备和数据仓库阶段可能需要很长时间。在较长的一段时间内，SnapMirror 关系报告状态 "正在准备" 的情况并不少见。



将 SnapMirror 关系类型从 DP 转换为 XDP 后，自动调整大小和空间保证等空间相关设置将不再复制到目标。

#### 步骤

1. 在目标集群中、确保SnapMirror关系类型为DP、镜像状态为SnapMirrored、关系状态为Idle且关系运行状况良好：

```
snapmirror show -destination-path <SVM:volume>
```

以下示例显示了的输出 `snapmirror show` 命令：

```
cluster_dst::>snapmirror show -destination-path svm_backup:volA_dst

Source Path: svm1:volA
Destination Path: svm_backup:volA_dst
Relationship Type: DP
SnapMirror Schedule: -
Tries Limit: -
Throttle (KB/sec): unlimited
Mirror State: Snapmirrored
Relationship Status: Idle
Transfer Snapshot: -
Snapshot Progress: -
Total Progress: -
Snapshot Checkpoint: -
Newest Snapshot: snapmirror.10af643c-32d1-11e3-954b-
123478563412_2147484682.2014-06-27_100026
Newest Snapshot Timestamp: 06/27 10:00:55
Exported Snapshot: snapmirror.10af643c-32d1-11e3-954b-
123478563412_2147484682.2014-06-27_100026
Exported Snapshot Timestamp: 06/27 10:00:55
Healthy: true
```



您可能会发现保留的副本很有用 `snapmirror show` 用于跟踪现有关系设置的命令输出。

2. 在源卷和目标卷中、确保两个卷都具有通用Snapshot副本：

```
volume snapshot show -vserver <SVM> -volume <volume>
```

以下示例显示了 `volume snapshot show` 源卷和目标卷的输出：

```

cluster_src:> volume snapshot show -vserver vsml -volume volA
---Blocks---
Vserver Volume Snapshot State Size Total% Used%
-----
-----
svml volA
weekly.2014-06-09_0736 valid 76KB 0% 28%
weekly.2014-06-16_1305 valid 80KB 0% 29%
daily.2014-06-26_0842 valid 76KB 0% 28%
hourly.2014-06-26_1205 valid 72KB 0% 27%
hourly.2014-06-26_1305 valid 72KB 0% 27%
hourly.2014-06-26_1405 valid 76KB 0% 28%
hourly.2014-06-26_1505 valid 72KB 0% 27%
hourly.2014-06-26_1605 valid 72KB 0% 27%
daily.2014-06-27_0921 valid 60KB 0% 24%
hourly.2014-06-27_0921 valid 76KB 0% 28%
snapmirror.10af643c-32d1-11e3-954b-123478563412_2147484682.2014-06-
27_100026
valid 44KB 0% 19%
11 entries were displayed.

cluster_dest:> volume snapshot show -vserver svm_backup -volume volA_dst
---Blocks---
Vserver Volume Snapshot State Size Total% Used%
-----
-----
svm_backup volA_dst
weekly.2014-06-09_0736 valid 76KB 0% 30%
weekly.2014-06-16_1305 valid 80KB 0% 31%
daily.2014-06-26_0842 valid 76KB 0% 30%
hourly.2014-06-26_1205 valid 72KB 0% 29%
hourly.2014-06-26_1305 valid 72KB 0% 29%
hourly.2014-06-26_1405 valid 76KB 0% 30%
hourly.2014-06-26_1505 valid 72KB 0% 29%
hourly.2014-06-26_1605 valid 72KB 0% 29%
daily.2014-06-27_0921 valid 60KB 0% 25%
hourly.2014-06-27_0921 valid 76KB 0% 30%
snapmirror.10af643c-32d1-11e3-954b-123478563412_2147484682.2014-06-
27_100026

```

3. 要确保计划的更新不会在转换期间运行、请暂停现有的DP类型关系：

```
snapmirror quiesce -source-path <SVM:volume> -destination-path  
<SVM:volume>
```

有关完整的命令语法，请参见 ["手册页"](#)。



您必须从目标 SVM 或目标集群运行此命令。

以下示例将使源卷之间的关系处于静音状态 volA 开启 svm1 和目标卷 volA\_dst 开启 svm\_backup:

```
cluster_dst::> snapmirror quiesce -destination-path svm_backup:volA_dst
```

#### 4. 中断现有的 DP 类型关系:

```
snapmirror break -destination-path <SVM:volume>
```

有关完整的命令语法，请参见 ["手册页"](#)。



您必须从目标 SVM 或目标集群运行此命令。

以下示例将中断源卷之间的关系 volA 开启 svm1 和目标卷 volA\_dst 开启 svm\_backup:

```
cluster_dst::> snapmirror break -destination-path svm_backup:volA_dst
```

#### 5. 如果在目标卷上启用了Snapshot副本自动删除、请将其禁用:

```
volume snapshot autodelete modify -vserver _SVM_ -volume _volume_  
-enabled false
```

以下示例将在目标卷上禁用Snapshot副本自动删除 volA\_dst:

```
cluster_dst::> volume snapshot autodelete modify -vserver svm_backup  
-volume volA_dst -enabled false
```

#### 6. 删除现有 DP 类型关系:

```
snapmirror delete -destination-path <SVM:volume>
```

有关完整的命令语法，请参见 ["手册页"](#)。



您必须从目标 SVM 或目标集群运行此命令。

以下示例将删除源卷之间的关系 volA 开启 svm1 和目标卷 volA\_dst 开启 svm\_backup:

```
cluster_dst::> snapmirror delete -destination-path svm_backup:volA_dst
```

#### 7. 释放源上的初始SVM灾难恢复关系:

```
snapmirror release -destination-path <SVM:volume> -relationship-info  
-only true
```

以下示例将释放SVM灾难恢复关系:

```
cluster_src::> snapmirror release -destination-path svm_backup:volA_dst  
-relationship-info-only true
```

#### 8. 您可以使用中保留的输出 snapmirror show 用于创建新XDP类型关系的命令:

```
snapmirror create -source-path <SVM:volume> -destination-path  
<SVM:volume> -type XDP -schedule <schedule> -policy <policy>
```

新关系必须使用相同的源卷和目标卷。有关完整的命令语法, 请参见手册页。



您必须从目标 SVM 或目标集群运行此命令。

以下示例将在源卷之间创建SnapMirror灾难恢复关系 volA 开启 svm1 和目标卷 volA\_dst 开启 svm\_backup 使用默认值 MirrorAllSnapshots 策略:

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination  
-path svm_backup:volA_dst  
-type XDP -schedule my_daily -policy MirrorAllSnapshots
```

#### 9. 重新同步源卷和目标卷:

```
snapmirror resync -source-path <SVM:volume> -destination-path  
<SVM:volume>
```

要缩短重新同步时间、您可以使用 -quick-resync 选项、但您应注意、存储效率节省可能会丢失。有关完整的命令语法、请参见手册页: "[snapmirror resync命令](#)".



您必须从目标 SVM 或目标集群运行此命令。虽然重新同步不需要基线传输，但它可能非常耗时。您可能希望在非高峰时段运行重新同步。

以下示例将重新同步源卷之间的关系 volA 开启 svm1 和目标卷 volA\_dst 开启 svm\_backup:

```
cluster_dst::> snapmirror resync -source-path svm1:volA -destination  
-path svm_backup:volA_dst
```

10. 如果您禁用了Snapshot副本自动删除、请重新启用它:

```
volume snapshot autodelete modify -vserver <SVM> -volume <volume>  
-enabled true
```

完成后

1. 使用 `snapmirror show` 命令以验证是否已创建SnapMirror关系。
2. 一旦SnapMirror XDP目标卷开始按照SnapMirror策略的定义更新Snapshot副本、请使用的输出 `snapmirror list-destinations` 命令以显示新的SnapMirror XDP关系。

## 转换 SnapMirror 关系的类型

从 ONTAP 9.5 开始，支持 SnapMirror 同步。您可以将异步 SnapMirror 关系转换为 SnapMirror 同步关系，反之亦然，而无需执行基线传输。

关于此任务

您不能通过更改 SnapMirror 策略将异步 SnapMirror 关系转换为 SnapMirror 同步关系，反之亦然

步骤

- \* 将异步 SnapMirror 关系转换为 SnapMirror 同步关系 \*

a. 从目标集群中，删除异步 SnapMirror 关系:

```
snapmirror delete -destination-path SVM:volume
```

```
cluster2::>snapmirror delete -destination-path vs1_dr:vol1
```

b. 在源集群中，释放 SnapMirror 关系而不删除通用 Snapshot 副本:

```
snapmirror release -relationship-info-only true -destination-path  
dest_SVM:dest_volume
```

```
cluster1::>snapmirror release -relationship-info-only true  
-destination-path vs1_dr:vol1
```

c. 从目标集群中，创建 SnapMirror 同步关系：

```
snapmirror create -source-path src_SVM:src_volume -destination-path  
dest_SVM:dest_volume -policy sync-mirror
```

```
cluster2::>snapmirror create -source-path vs1:vol1 -destination-path  
vs1_dr:vol1 -policy sync
```

d. 重新同步 SnapMirror 同步关系：

```
snapmirror resync -destination-path dest_SVM:dest_volume
```

```
cluster2::>snapmirror resync -destination-path vs1_dr:vol1
```

• \* 将 SnapMirror 同步关系转换为异步 SnapMirror 关系 \*

a. 从目标集群中，暂停现有的 SnapMirror 同步关系：

```
snapmirror quiesce -destination-path dest_SVM:dest_volume
```

```
cluster2::> snapmirror quiesce -destination-path vs1_dr:vol1
```

b. 从目标集群中，删除异步 SnapMirror 关系：

```
snapmirror delete -destination-path SVM:volume
```

```
cluster2::>snapmirror delete -destination-path vs1_dr:vol1
```

c. 在源集群中，释放 SnapMirror 关系而不删除通用 Snapshot 副本：

```
snapmirror release -relationship-info-only true -destination-path  
dest_SVM:dest_volume
```

```
cluster1::>snapmirror release -relationship-info-only true  
-destination-path vs1_dr:vol1
```

d. 从目标集群中，创建异步 SnapMirror 关系：

```
snapmirror create -source-path src_SVM:src_volume -destination-path
```



```
dest_SVM:dest_volume -policy MirrorAllSnapshots
```

```
cluster2::>snapmirror create -source-path vs1:vol1 -destination-path  
vs1_dr:vol1 -policy sync
```

e. 重新同步 SnapMirror 同步关系：

```
snapmirror resync -destination-path dest_SVM:dest_volume
```

```
cluster2::>snapmirror resync -destination-path vs1_dr:vol1
```

## 转换 SnapMirror 同步关系的模式

从 ONTAP 9.5 开始，支持 SnapMirror 同步关系。您可以将 SnapMirror 同步关系的模式从 StrictSync 转换为 Sync，反之亦然。

关于此任务

您不能修改 SnapMirror 同步关系的策略来转换其模式。

步骤

1. 从目标集群中，暂停现有的 SnapMirror 同步关系：

```
snapmirror quiesce -destination-path dest_SVM:dest_volume
```

```
cluster2::> snapmirror quiesce -destination-path vs1_dr:vol1
```

2. 从目标集群中，删除现有的 SnapMirror 同步关系：

```
snapmirror delete -destination-path dest_SVM:dest_volume
```

```
cluster2::> snapmirror delete -destination-path vs1_dr:vol1
```

3. 在源集群中，释放 SnapMirror 关系而不删除通用 Snapshot 副本：

```
snapmirror release -relationship-info-only true -destination-path  
dest_SVM:dest_volume
```

```
cluster1::> snapmirror release -relationship-info-only true -destination  
-path vs1_dr:vol1
```

4. 在目标集群中，通过指定要将 SnapMirror 同步关系转换到的模式来创建 SnapMirror 同步关系：

```
snapmirror create -source-path vs1:vol1 -destination-path dest_SVM:dest_volume  
-policy Sync|StrictSync
```

```
cluster2::> snapmirror create -source-path vs1:vol1 -destination-path  
vs1_dr:vol1 -policy Sync
```

#### 5. 从目标集群重新同步 SnapMirror 关系：

```
snapmirror resync -destination-path dest_SVM:dest_volume
```

```
cluster2::> snapmirror resync -destination-path vs1_dr:vol1
```

## 创建和删除SnapMirror故障转移测试卷

从ONTAP 9.14.1开始、您可以使用System Manager创建卷克隆、以便在不中断活动SnapMirror关系的情况下测试SnapMirror故障转移和灾难恢复。完成测试后、您可以清理关联数据并删除测试卷。

### 创建SnapMirror故障转移测试卷



#### 关于此任务

- 您可以对同步和异步SnapMirror关系执行故障转移测试。
- 创建卷克隆以执行灾难恢复测试。
- 此时将在与SnapMirror目标相同的Storage VM上创建克隆卷。
- 您可以使用FlexVol和FlexGroup SnapMirror关系。
- 如果选定关系已存在测试克隆、则无法为此关系创建其他克隆。
- 不支持SnapLock存储关系。

#### 开始之前

- 您必须是集群管理员。
- 必须在源集群和目标集群上安装SnapMirror许可证。

#### 步骤


1. 在目标集群上、选择\*保护>关系\*。
2. 选择 ...  在关系源旁边，选择\*Test Failover\*。
3. 在\*Test Failover (测试故障转移)窗口中，选择\***Test Failover** (测试故障转移)\*。
4. 选择\*存储>卷\*，然后验证是否列出了测试故障转移卷。
5. 选择\*存储>共享\*。
6. 单击  **Add** 并选择\*共享\*。

7. 在\*添加共享\*窗口的\*共享名称\*字段中键入共享的名称。
8. 在\*文件夹\*字段中，选择\*浏览\*，选择测试克隆卷，然后选择\*保存\*。
9. 在\*添加共享\*窗口底部，选择\*保存\*。
10. 在客户端上打开共享、并验证测试卷是否具有读写功能。

## 清理故障转移数据并删除测试卷

完成故障转移测试后、您可以清理与测试卷关联的所有数据并将其删除。

### 步骤

1. 在目标集群上、选择\*保护>关系\*。
2. 选择 ...  在关系源旁边，然后选择\*清理测试故障转移\*。
3. 在\*清理测试故障转移\*窗口中，选择\*清理\*。
4. 选择\*存储>卷\*并确认已删除测试卷。

## 从 SnapMirror 灾难恢复目标卷提供数据

### 使目标卷可写

您需要先使目标卷可写，然后才能将数据从该卷提供给客户端。您可以使用 `snapmirror quiesce` 命令以停止向目标(即)进行的计划传输 `snapmirror abort` 命令以停止正在进行的传输、以及 `snapmirror break` 命令以使目标可写。

### 关于此任务

您必须从目标 SVM 或目标集群执行此任务。

### 步骤

1. 停止向目标进行的计划传输：

```
snapmirror quiesce -source-path SVM:volume|cluster://SVM/volume, ...  
-destination-path SVM:volume|cluster://SVM/volume, ...
```

有关完整的命令语法，请参见手册页。

以下示例将停止源卷之间的计划传输 `volA` 开启 `svm1` 和目标卷 `volA_dst` 开启 `svm_backup`：

```
cluster_dst::> snapmirror quiesce -source-path svm1:volA -destination  
-path svm_backup:volA_dst
```

2. 停止正在向目标传输的数据：

```
snapmirror abort -source-path SVM:volume|cluster://SVM/volume, ... -destination  
-path SVM:volume|cluster://SVM/volume, ...
```

有关完整的命令语法，请参见手册页。



SnapMirror 同步关系不需要执行此步骤（从 ONTAP 9.5 开始受支持）。

以下示例将停止源卷之间正在进行的传输 volA 开启 svm1 和目标卷 volA\_dst 开启 svm\_backup:

```
cluster_dst::> snapmirror abort -source-path svm1:volA -destination-path
svm_backup:volA_dst
```

3. 中断 SnapMirror 灾难恢复关系:

```
snapmirror break -source-path SVM:volume|cluster://SVM/volume, ... -destination
-path SVM:volume|cluster://SVM/volume, ...
```

有关完整的命令语法，请参见手册页。

以下示例将中断源卷之间的关系 volA 开启 svm1 和目标卷 volA\_dst 开启 svm\_backup:

```
cluster_dst::> snapmirror break -source-path svm1:volA -destination-path
svm_backup:volA_dst
```

在 ONTAP 中执行此操作的其他方法

要执行以下任务，请执行以下操作 ...	查看此内容 ...
重新设计的 System Manager（适用于 ONTAP 9.7 及更高版本）	<a href="#">"从 SnapMirror 目标提供数据"</a>
System Manager 经典版（适用于 ONTAP 9.7 及更早版本）	<a href="#">"卷灾难恢复概述"</a>

配置用于数据访问的目标卷

在使目标卷可写后，您必须为该卷配置数据访问。NAS 客户端， NVMe 子系统和 SAN 主机可以从目标卷访问数据，直到源卷重新激活为止。

NAS 环境:

- 1. 使用与源 SVM 中的源卷挂载到的相同接合路径将 NAS 卷挂载到命名空间。
- 2. 将适当的ACL应用于目标卷上的SMB共享。
- 3. 将 NFS 导出策略分配给目标卷。
- 4. 将配额规则应用于目标卷。
- 5. 将客户端重定向到目标卷。
- 6. 在客户端上重新挂载NFS和SMB共享。

SAN环境：

1. 将卷中的 LUN 映射到相应的启动程序组。
2. 对于 iSCSI，创建从 SAN 主机启动程序到 SAN LIF 的 iSCSI 会话。
3. 在 SAN 客户端上，执行存储重新扫描以检测已连接的 LUN。

有关 NVMe 环境的信息，请参见 ["SAN 管理"](#)。

## 重新激活原始源卷

当您不再需要从目标提供数据时，可以在源卷和目标卷之间重新建立初始数据保护关系。

关于此任务

- 以下操作步骤假定原始源卷中的基线完好无损。如果基线不完好，则必须在执行操作步骤之前创建并初始化提供数据的卷与原始源卷之间的关系。
- XDP SnapMirror 关系的后台准备和数据仓库阶段可能需要很长时间。在较长的一段时间内，SnapMirror 关系报告状态 "正在准备" 的情况并不少见。

步骤

1. 反转原始数据保护关系：

```
snapmirror resync -source-path SVM:volume -destination-path SVM:volume
```

有关完整的命令语法，请参见手册页。



您必须从原始源SVM或原始源集群运行此命令。虽然重新同步不需要基线传输，但它可能非常耗时。您可能希望在非高峰时段运行重新同步。如果源和目标上不存在通用 Snapshot 副本，则此命令将失败。使用 ... snapmirror initialize 重新初始化此关系。

以下示例将反转初始源卷之间的关系：volA 开启 svm1 和提供数据的卷、`volA\_dst` 开启 svm\_backup：

```
cluster_src::> snapmirror resync -source-path svm_backup:volA_dst  
-destination-path svm1:volA
```

2. 准备好重新建立对原始源的数据访问后、停止对原始目标卷的访问。要执行此操作、一种方法是停止原始目标SVM：

```
vserver stop -vserver SVM
```

有关完整的命令语法，请参见手册页。



您必须从原始目标SVM或原始目标集群运行此命令。此命令将停止用户对整个原始目标SVM的访问。您可能希望使用其他方法停止对原始目标卷的访问。

以下示例将停止初始目标SVM：

```
cluster_dst::> vserver stop svm_backup
```

### 3. 更新已反转的关系：

```
snapmirror update -source-path SVM:volume -destination-path SVM:volume
```

有关完整的命令语法，请参见手册页。



您必须从原始源SVM或原始源集群运行此命令。

以下示例将更新提供数据的卷之间的关系、volA\_dst 开启 svm\_backup`和原始源卷、`volA 开启 svm1：

```
cluster_src::> snapmirror update -source-path svm_backup:volA_dst  
-destination-path svm1:volA
```

### 4. 从初始源SVM或初始源集群停止已反转关系的计划传输：

```
snapmirror quiesce -source-path SVM:volume -destination-path SVM:volume
```

有关完整的命令语法，请参见手册页。



您必须从原始源SVM或原始源集群运行此命令。

以下示例将停止初始目标卷之间的计划传输、volA\_dst 开启 svm\_backup`和原始源卷、`volA 开启 svm1：

```
cluster_src::> snapmirror quiesce -source-path svm_backup:volA_dst  
-destination-path svm1:volA
```

### 5. 当最终更新完成且关系状态指示"已暂停"时、从原始源SVM或原始源集群运行以下命令以中断已反转的关系：

```
snapmirror break -source-path SVM:volume -destination-path SVM:volume
```

有关完整的命令语法，请参见手册页。



您必须从原始源SVM或源集群运行此命令。

以下示例将中断初始目标卷之间的关系、volA\_dst 开启 svm\_backup`和原始源卷、`volA 开启 svm1：

```
cluster_src::> snapmirror break -source-path svm_backup:volA_dst  
-destination-path svm1:volA
```

6. 从原始源SVM或原始源集群中、删除已反转的数据保护关系：

```
snapmirror delete -source-path SVM:volume -destination-path SVM:volume
```

有关完整的命令语法，请参见手册页。



您必须从原始源SVM或原始源集群运行此命令。

以下示例将删除初始源卷之间的已反转关系： volA 开启 svm1 和提供数据的卷、 `volA\_dst` 开启 svm\_backup:

```
cluster_src::> snapmirror delete -source-path svm_backup:volA_dst  
-destination-path svm1:volA
```

7. 从初始目标SVM或初始目标集群释放已反转的关系。

```
snapmirror release -source-path SVM:volume -destination-path SVM:volume
```



您必须从原始目标SVM或原始目标集群运行此命令。

以下示例将释放初始目标卷之间的已反转关系、 volA\_dst 开启 svm\_backup 和原始源卷、 `volA` 开启 svm1:

```
cluster_dst::> snapmirror release -source-path svm_backup:volA_dst  
-destination-path svm1:volA
```

8. 从原始目标重新建立初始数据保护关系：

```
snapmirror resync -source-path SVM:volume -destination-path SVM:volume
```

有关完整的命令语法，请参见手册页。

以下示例将重新建立初始源卷之间的关系、 volA 开启 svm1 和初始目标卷、 `volA\_dst` 开启 svm\_backup:

```
cluster_dst::> snapmirror resync -source-path svm1:volA -destination  
-path svm_backup:volA_dst
```

9. 如果需要、启动初始目标SVM：

```
vserver start -vserver SVM
```

有关完整的命令语法，请参见手册页。

以下示例将启动初始目标SVM：

```
cluster_dst::> vserver start svm_backup
```

完成后

使用 `snapmirror show` 命令以验证是否已创建 SnapMirror 关系。有关完整的命令语法，请参见手册页。

## 从 SnapMirror 目标卷还原文件

从 **SnapMirror** 目标还原单个文件，**LUN** 或 **NVMe** 命名空间

您可以从 Snapshot 副本还原单个文件，LUN，一组文件或 LUN，也可以从 SnapMirror 目标卷还原 NVMe 命名空间。从 ONTAP 9.7 开始，您还可以从 SnapMirror 同步目标还原 NVMe 命名空间。您可以将文件还原到原始源卷或其他卷。

您需要的内容

要从 SnapMirror 同步目标（从 ONTAP 9.5 开始受支持）还原文件或 LUN，必须先删除并释放此关系。

关于此任务

要将文件或 LUN 还原到的卷（目标卷）必须为读写卷：

- 如果源卷和目标卷具有通用 Snapshot 副本（通常情况下，还原到原始源卷时），SnapMirror 将执行 `_incremental restore`。
- 否则，SnapMirror 将执行 `_baseline restore`，其中指定的 Snapshot 副本及其引用的所有数据块将传输到目标卷。

步骤

1. 列出目标卷中的 Snapshot 副本：

```
volume snapshot show -vserver SVM -volume volume
```

有关完整的命令语法，请参见手册页。

以下示例显示了上的 Snapshot 副本 `vserverB:secondary1` 目标：



```
cluster_dst:> volume snapshot show -vserver vserverB -volume secondary1
```

Vserver	Volume	Snapshot	State	Size	Total% Used%
-----	-----	-----	-----	-----	-----
vserverB	secondary1	hourly.2013-01-25_0005	valid	224KB	0%
0%		daily.2013-01-25_0010	valid	92KB	0%
0%		hourly.2013-01-25_0105	valid	228KB	0%
0%		hourly.2013-01-25_0205	valid	236KB	0%
0%		hourly.2013-01-25_0305	valid	244KB	0%
0%		hourly.2013-01-25_0405	valid	244KB	0%
0%		hourly.2013-01-25_0505	valid	244KB	0%

7 entries were displayed.

## 2. 从 SnapMirror 目标卷中的 Snapshot 副本还原单个文件或 LUN 或一组文件或 LUN：

```
snapmirror restore -source-path SVM:volume|cluster://SVM/volume, ...
-destination-path SVM:volume|cluster://SVM/volume, ... -source-snapshot snapshot
-file-list source_file_path,@destination_file_path
```

有关完整的命令语法，请参见手册页。



您必须从目标 SVM 或目标集群运行此命令。

以下命令将还原这些文件 file1 和 file2 从 Snapshot 副本 daily.2013-01-25\_0010 在初始目标卷中 secondary1，将其保存到初始源卷的活动文件系统中的同一位置 primary1：

```
cluster_dst:> snapmirror restore -source-path vserverB:secondary1
-destination-path vserverA:primary1 -source-snapshot daily.2013-01-
25_0010 -file-list /dir1/file1,/dir2/file2
```

```
[Job 3479] Job is queued: snapmirror restore for the relationship with
destination vserverA:primary1
```

以下命令将还原这些文件 file1 和 file2 从 Snapshot 副本 daily.2013-01-25\_0010 在初始目标卷中 secondary1，并将其移至原始源卷的活动文件系统中的其他位置 primary1。

目标文件路径以 @ 符号开头，后跟原始源卷根中文件的路径。在此示例中、file1 将还原到 /dir1/file1.new 和file2将还原到 /dir2.new/file2 开启 primary1:

```
cluster_dst:> snapmirror restore -source-path vserverB:secondary1
-destination-path vserverA:primary1 -source-snapshot daily.2013-01-
25_0010 -file-list
/dir/file1,@/dir1/file1.new,/dir2/file2,@/dir2.new/file2

[Job 3479] Job is queued: snapmirror restore for the relationship with
destination vserverA:primary1
```

以下命令将还原这些文件 file1 和 file3 从Snapshot副本 daily.2013-01-25\_0010 在初始目标卷中 secondary1，并将其保存到初始源卷的活动文件系统中的不同位置 primary1`和还原 `file2 from snap1 到的活动文件系统中的同一位置 primary1。

在此示例中、为文件 file1 将还原到 /dir1/file1.new 和 file3 将还原到 /dir3.new/file3:

```
cluster_dst:> snapmirror restore -source-path vserverB:secondary1
-destination-path vserverA:primary1 -source-snapshot daily.2013-01-
25_0010 -file-list
/dir/file1,@/dir1/file1.new,/dir2/file2,/dir3/file3,@/dir3.new/file3

[Job 3479] Job is queued: snapmirror restore for the relationship with
destination vserverA:primary1
```

## 从 SnapMirror 目标还原卷的内容

您可以从 SnapMirror 目标卷中的 Snapshot 副本还原整个卷的内容。您可以将卷的内容还原到原始源卷或其他卷。

关于此任务

还原操作的目标卷必须为以下卷之一：

- 读写卷，在这种情况下，如果源卷和目标卷具有通用 Snapshot 副本（通常情况下，还原到原始源卷时），SnapMirror 将执行 \_incremental restore"。



如果没有通用 Snapshot 副本，则此命令将失败。您不能将卷的内容还原到空的读写卷。

- 空数据保护卷，在这种情况下，SnapMirror 会执行 \_baseline restore"，其中指定的 Snapshot 副本及其引用的所有数据块都会传输到源卷。

还原卷的内容会造成系统中断。还原操作正在运行时、SnapVault主卷上不能运行SMB流量。

如果还原操作的目标卷已启用数据压缩，而源卷未启用数据压缩，请在目标卷上禁用数据压缩。还原操作完成后，您需要重新启用数据压缩。

在执行还原之前，为目标卷定义的任何配额规则都将被停用。您可以使用 `volume quota modify` 命令以在还原操作完成后重新激活配额规则。

步骤

1. 列出目标卷中的 Snapshot 副本：

```
volume snapshot show -vserver <SVM> -volume <volume>
```

有关完整的命令语法，请参见手册页。

以下示例显示了上的Snapshot副本 `vserverB:secondary1` 目标：

```
cluster_dst::> volume snapshot show -vserver vserverB -volume secondary1
```

Vserver	Volume	Snapshot	State	Size	Total%
Used%					
-----	-----	-----	-----	-----	-----
-----					
vserverB	secondary1	hourly.2013-01-25_0005	valid	224KB	0%
0%		daily.2013-01-25_0010	valid	92KB	0%
0%		hourly.2013-01-25_0105	valid	228KB	0%
0%		hourly.2013-01-25_0205	valid	236KB	0%
0%		hourly.2013-01-25_0305	valid	244KB	0%
0%		hourly.2013-01-25_0405	valid	244KB	0%
0%		hourly.2013-01-25_0505	valid	244KB	0%
0%					

7 entries were displayed.

2. 从 SnapMirror 目标卷中的 Snapshot 副本还原卷的内容：

```
snapmirror restore -source-path <SVM:volume>|<cluster://SVM/volume>  
-destination-path <SVM:volume>|<cluster://SVM/volume> -source-snapshot  
<snapshot>
```

有关完整的命令语法，请参见手册页。



您必须从原始源SVM或原始源集群运行此命令。

以下命令将还原初始源卷的内容 `primary1` 从Snapshot副本 `daily.2013-01-25_0010` 在初始目标卷中 `secondary1`：

```
cluster_src::> snapmirror restore -source-path vserverB:secondary1
-destination-path vserverA:primary1 -source-snapshot daily.2013-01-
25_0010
```

Warning: All data newer than Snapshot copy daily.2013-01-25\_0010 on volume vserverA:primary1 will be deleted.

Do you want to continue? {y|n}: y

[Job 34] Job is queued: snapmirror restore from source vserverB:secondary1 for the snapshot daily.2013-01-25\_0010.

3. 重新挂载已还原的卷并重新启动使用该卷的所有应用程序。

在 **ONTAP** 中执行此操作的其他方法

要执行以下任务，请执行以下操作 ...	查看此内容 ...
重新设计的 System Manager（适用于 ONTAP 9.7 及更高版本）	<a href="#">"从早期的 Snapshot 副本还原卷"</a>
System Manager 经典版（适用于 ONTAP 9.7 及更早版本）	<a href="#">"使用 SnapVault 还原卷概述"</a>

## 手动更新复制关系

如果由于源卷已移动而导致更新失败，您可能需要手动更新复制关系。

关于此任务

SnapMirror 会中止从移动的源卷进行的所有传输，直到您手动更新复制关系为止。

从 ONTAP 9.5 开始，支持 SnapMirror 同步关系。尽管源卷和目标卷在这些关系中始终处于同步状态，但二级集群中的视图只会每小时与主卷同步一次。如果要查看目标上的时间点数据、则应运行来执行手动更新 `snapmirror update` 命令：

步骤

1. 手动更新复制关系：

```
snapmirror update -source-path SVM:volume|cluster://SVM/volume, ... -destination
-path SVM:volume|cluster://SVM/volume, ...
```

有关完整的命令语法，请参见手册页。



您必须从目标 SVM 或目标集群运行此命令。如果源和目标上不存在通用 Snapshot 副本，则此命令将失败。使用 ... `snapmirror initialize` 重新初始化此关系。

以下示例将更新源卷之间的关系 volA 开启 svm1 和目标卷 volA\_dst 开启 svm\_backup：

```
cluster_src::> snapmirror update -source-path svm1:volA -destination  
-path svm_backup:volA_dst
```

## 重新同步复制关系

在将目标卷设为可写之后，在因源卷和目标卷上不存在通用 Snapshot 副本而导致更新失败之后，或者如果要更改此关系的复制策略，则需要重新同步复制关系。

关于此任务

- 虽然重新同步不需要基线传输，但它可能非常耗时。您可能希望在非高峰时段运行重新同步。
- 属于扇出或级联配置的卷可能需要较长时间才能重新同步。在较长的一段时间内，SnapMirror 关系报告状态 "正在准备" 的情况并不少见。

步骤

1. 重新同步源卷和目标卷：

```
snapmirror resync -source-path SVM:volume|cluster://SVM/volume, ... -destination  
-path SVM:volume|cluster://SVM/volume, ... -type DP|XDP -policy policy
```

有关完整的命令语法，请参见手册页。



您必须从目标 SVM 或目标集群运行此命令。

以下示例将重新同步源卷之间的关系 volA 开启 svm1 和目标卷 volA\_dst 开启 svm\_backup：

```
cluster_dst::> snapmirror resync -source-path svm1:volA -destination  
-path svm_backup:volA_dst
```

## 删除卷复制关系

您可以使用 `snapmirror delete` 和 `snapmirror release` 用于删除卷复制关系的命令。然后，您可以手动删除不需要的目标卷。

关于此任务

。 `snapmirror release` 命令将从源中删除 SnapMirror 创建的任何 Snapshot 副本。您可以使用 `-relationship-info-only` 选项以保留 Snapshot 副本。

步骤

1. 暂停复制关系：

```
snapmirror quiesce -destination-path SVM:volume|cluster://SVM/volume
```

```
cluster_dst:> snapmirror quiesce -destination-path svm_backup:volA_dst
```

2. (可选)如果您要求目标卷为读/写卷、请中断复制关系。如果您计划删除目标卷或不需要对卷进行读/写、则可以跳过此步骤:

```
snapmirror break -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...
```

```
cluster_dst:> snapmirror break -source-path svm1:volA -destination-path svm_backup:volA_dst
```

3. 删除复制关系:

```
snapmirror delete -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...
```

有关完整的命令语法, 请参见手册页。



您必须从目标集群或目标 SVM 运行此命令。

以下示例将删除源卷之间的关系 volA 开启 svm1 和目标卷 volA\_dst 开启 svm\_backup:

```
cluster_dst:> snapmirror delete -source-path svm1:volA -destination-path svm_backup:volA_dst
```

4. 从源 SVM 释放复制关系信息:

```
snapmirror release -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...
```

有关完整的命令语法, 请参见手册页。



您必须从源集群或源 SVM 运行此命令。

以下示例将从源SVM释放指定复制关系的信息 svm1:

```
cluster_src:> snapmirror release -source-path svm1:volA -destination-path svm_backup:volA_dst
```

## 管理存储效率

SnapMirror 可保留源卷和目标卷上的存储效率, 但在目标上启用后处理数据压缩时除外。

在这种情况下，目标上的所有存储效率都将丢失。要更正此问题描述，您需要在目标上禁用后处理压缩，手动更新关系并重新启用存储效率。

您需要的内容

- 源和目标集群以及 SVM 必须建立对等关系。

#### "集群和 SVM 对等"

- 必须在目标上禁用后处理压缩。

关于此任务

您可以使用 `volume efficiency show` 命令以确定是否已在卷上启用效率。有关详细信息，请参见手册页。

您可以通过查看 SnapMirror 审核日志并找到传输问题描述来检查 SnapMirror 是否保持存储效率。如果显示传输问题描述 `transfer_desc=Logical Transfer`、SnapMirror 不会保持存储效率。如果显示传输问题描述 `transfer_desc=Logical Transfer with Storage Efficiency`、SnapMirror 正在保持存储效率。例如：

```
Fri May 22 02:13:02 CDT 2020 ScheduledUpdate[May 22 02:12:00]:cc0fbc29-  
b665-11e5-a626-00a09860c273 Operation-Uid=39fbcf48-550a-4282-a906-  
df35632c73a1 Group=none Operation-Cookie=0 action=End source=<sourcepath>  
destination=<destpath> status=Success bytes_transferred=117080571  
network_compression_ratio=1.0:1 transfer_desc=Logical Transfer - Optimized  
Directory Mode
```

使用存储进行逻辑传输

从 ONTAP 9.3 开始，不再需要手动更新即可重新启用存储效率。如果 SnapMirror 检测到已禁用后处理压缩，则会在下次计划更新时自动重新启用存储效率。源和目标都必须运行 ONTAP 9.3。

从 ONTAP 9.3 开始，在将目标卷设为可写后，AFF 系统对存储效率设置的管理方式与 FAS 系统不同：

- 使用将目标卷设置为可写之后 `snapmirror break` 命令时、卷上的缓存策略会自动设置为 `"auto"` (默认值)。



此行为仅适用于 FlexVol 卷，不适用于 FlexGroup 卷。

- 重新同步时，缓存策略会自动设置为 `"none"`，并且无论您的原始设置如何，重复数据删除和实时压缩都会自动禁用。您必须根据需要手动修改这些设置。



启用存储效率后手动更新可能会非常耗时。您可能希望在非高峰时段运行此操作。

步骤

1. 更新复制关系并重新启用存储效率：

```
snapmirror update -source-path SVM:volume|cluster://SVM/volume, ... -destination  
-path SVM:volume|cluster://SVM/volume, ... -enable-storage-efficiency true
```

有关完整的命令语法，请参见手册页。



您必须从目标 SVM 或目标集群运行此命令。如果源和目标上不存在通用 Snapshot 副本，则此命令将失败。使用 `... snapmirror initialize` 重新初始化此关系。

以下示例将更新源卷之间的关系 volA 开启 svm1 和目标卷 volA\_dst 开启 `svm\_backup` 并重新启用存储效率：

```
cluster_dst::> snapmirror update -source-path svm1:volA -destination
-path svm_backup:volA_dst -enable-storage-efficiency true
```

## 使用 SnapMirror 全局限制

全局网络限制可用于每个节点级别的所有 SnapMirror 和 SnapVault 传输。

关于此任务

SnapMirror 全局限制用于限制传入和 / 或传出 SnapMirror 和 SnapVault 传输所使用的带宽。此限制会在集群范围内对集群中的所有节点强制实施。

例如、如果传出限制设置为100 Mbps、则集群中的每个节点的传出带宽都将设置为100 Mbps。如果禁用了全局限制，则会在所有节点上禁用它。

尽管数据传输速率通常以每秒位数（bps）表示，但必须以千字节 / 秒（KBps）为单位输入限制值。



在ONTAP 9.9.1及更早版本中、此限制对不起作用 `volume move` 传输或负载共享镜像传输。从ONTAP 9.10.0开始、您可以指定一个选项来限制卷移动操作。有关详细信息，请参见 ["如何在ONTAP 9.10及更高版本中限制卷移动。"](#)

全局限制与 SnapMirror 和 SnapVault 传输的每个关系限制功能结合使用。系统将强制实施每个关系限制，直到每个关系传输的总带宽超过全局限制的值为止，之后将强制实施全局限制。限制值 0 表示已禁用全局限制。



SnapMirror 全局限制对同步的 SnapMirror 同步关系没有影响。但是，在 SnapMirror 同步关系执行异步传输阶段（例如初始化操作）或发生 "不同步" 事件后，此限制会影响这些关系。因此，建议不要对 SnapMirror 同步关系启用全局限制。

步骤

1. 启用全局限制：

```
options -option-name replication.throttle.enable on|off
```

以下示例显示了如何在上启用SnapMirror全局限制 cluster\_dst：

```
cluster_dst::> options -option-name replication.throttle.enable on
```

2. 指定目标集群上传入传输所使用的最大总带宽：



```
options -option-name replication.throttle.incoming.max_kbs KBps
```

建议的最小限制带宽为 4 KBps，最大限制为 2 Tbps。此选项的默认值为 `unlimited`，表示已用总带宽无限制。

以下示例显示了如何将传入传输使用的最大总带宽设置为 100 Mbps：

```
cluster_dst::> options -option-name  
replication.throttle.incoming.max_kbs 12500
```



100 Mbps = 12500 KBps

### 3. 指定源集群上传出传输使用的最大总带宽：

```
options -option-name replication.throttle.outgoing.max_kbs KBps
```

建议的最小限制带宽为 4 KBps，最大限制为 2 Tbps。此选项的默认值为 `unlimited`，表示已用总带宽无限制。参数值以KBps为单位。

以下示例显示了如何将传出传输使用的最大总带宽设置为 100 Mbps：

```
cluster_src::> options -option-name  
replication.throttle.outgoing.max_kbs 12500
```

## 版权信息

版权所有 © 2024 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

## 商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。