ONTAP® 9

Disks and Aggregates Power Guide

October 2016 | [215-11589_A0]
doccomments@netapp.com

Updated for ONTAP 9.1
Release Candidate Documentation - Contents Subject To Change

NetApp®

Contents

Deciding whether to use the Disks and Aggregates Power Guide

This guide describes how to create, expand, and manage aggregates, taking into consideration performance and storage needs, disk drive types, partitioning, and RAID group considerations.

You should use this guide if you want to create, expand and manage your aggregates in the following way:

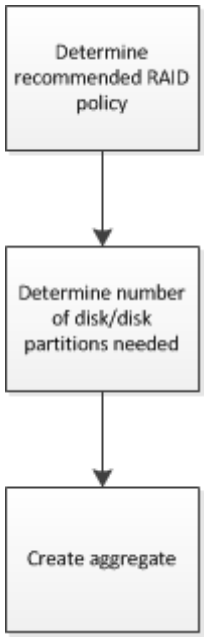
- You want to use the command-line interface (CLI), not an automated scripting tool.
- You want to use best practices, not explore every available option.
- You do not want to read a lot of conceptual background.
- You do not have a MetroCluster configuration.

If these assumptions are not correct for your situation, you should see the following resources:

- *ONTAP 9 Fabric-attached MetroCluster Installation and Configuration Guide*
- *Stretch MetroCluster installation and configuration*

Aggregate creation workflow

Creating an aggregate involves determining the recommended RAID policy, determining the number of disks to include in your aggregate, and creating the aggregate.



Default RAID policies for aggregates

Either RAID-DP or RAID-TEC is the default RAID policy for all new aggregates. The RAID policy determines the parity protection you have in the event of a disk failure.

RAID-DP provides double-parity protection in the event of a single or double disk failure. RAID-DP is the default RAID policy for the following aggregate types:

- All flash aggregates
- Flash Pool aggregates
- Performance hard disk drive (HDD) aggregates

Beginning with ONTAP 9.0, a new RAID policy called RAID-TEC is available. RAID-TEC is supported on all disk types and all platforms including All Flash FAS. Aggregates that contain larger disks have a higher possibility of concurrent disk failures. RAID-TEC helps to mitigate this risk by providing triple-parity protection so that your data can survive up to three simultaneous disk failures. RAID-TEC is the default RAID policy for capacity HDD aggregates with disks that are 6 TB or larger.

How to determine the number of disks or disk partitions required for an aggregate

You must have enough disks or disk partitions in your aggregate to meet system and business requirements. You should also have the recommended number of hot spare disks or hot spare disk partitions to minimize the potential of data loss.

root-data partitioning is enabled by default on certain configurations. Systems with root-data partitioning enabled use disk partitions to create aggregates. Systems that do not have root-data partitioning enabled use unpartitioned disks.

You must have enough disks or disk partitions to meet the minimum number required for your RAID policy and enough to meet your minimum capacity requirements. If the sum of usable space on the minimum number of disks or disk partitions required to meet your RAID policy does not meet your minimum capacity requirements, you must increase the number of disks or disk partitions until your capacity requirements are met. If your capacity requirements exceed the maximum number of RAID groups and disks allowed by your RAID group policy, you must create an additional aggregate.

Note: In ONTAP, the usable space of the drive is less than the physical capacity of the drive. You can find the usable space of a specific drive and the minimum number of disks or disk partitions required for each RAID policy in *Hardware Universe*. You can also use the `storage aggregate show-spare-disks` command to find the usable space of a specific disk.

In addition to the number of disks or disk partitions necessary to create your RAID group and meet your capacity requirements, you should also have the minimum number of hot spare disks or hot spare disk partitions recommended for your aggregate:

- For all flash aggregates, you should have a minimum of one hot spare disk or disk partition.
- For all AFF A700s systems, you should have a minimum of two hot spare disks or disk partitions.
- For non-flash homogenous aggregates, you should have a minimum of two hot spare disks or disk partitions.
- For Flash Pool aggregates, you should have a minimum of two hot spare disks or disk partitions for each disk type.
- To support the use of the Maintenance Center and to avoid issues caused by multiple concurrent disk failures, you should have a minimum of four hot spares in multi-disk carriers.

Related concepts

Configurations that support root-data partitioning on page 46

Related information

NetApp Hardware Universe
NetApp Technical Report 3838: Storage Subsystem Configuration Guide

Creating aggregates

You can create aggregates to provide storage to volumes on your system.

Before you begin

You must have determined the number of disks and the number of hot spare disks you need in the aggregate.

About this task

If root-data-data partitioning is enabled and you have 24 solid state drives (SSDs) or less in your configuration, it is recommended that your data partitions be assigned to different nodes. If root-data-data partitioning is enabled and you have more than 24 SSDs in your configuration, it is recommended that both of your data partitions be assigned to the same node.

The procedure for creating aggregates on systems with root-data partitioning and root-data-data partitioning enabled is the same as the procedure for creating aggregates on systems using unpartitioned disks. If root-data partitioning or root data-data partitioning is enabled on your system, you should use the number of disk partitions for the `-diskcount` option.

The name of your aggregate must conform to the following requirements:

- It must begin with either a letter or an underscore (_).
- It must contain only letters, digits, and underscores.
- It must contain 250 characters or less.

Steps

1. View the list of spare disk partitions to verify that you have enough to create your aggregate:

   storage aggregate show-spare-disks -original-owner *node_name*

   Data partitions are displayed under `Local Data Usable`. A root partition cannot be used as a spare.

2. Simulate the creation of the aggregate:

   storage aggregate create -aggregate *aggregate_name* -node *node_name* -raidtype raid_dp -diskcount *number_of_disks_or_partitions* -simulate true

3. If any warnings are displayed from the simulated command, adjust the command and repeat the simulation.

4. Create the aggregate:

   storage aggregate create -aggregate *aggr_name* -node *node_name* -raidtype raid_dp -diskcount *number_of_disks_or_partitions*

5. Display the aggregate to verify that it was created:

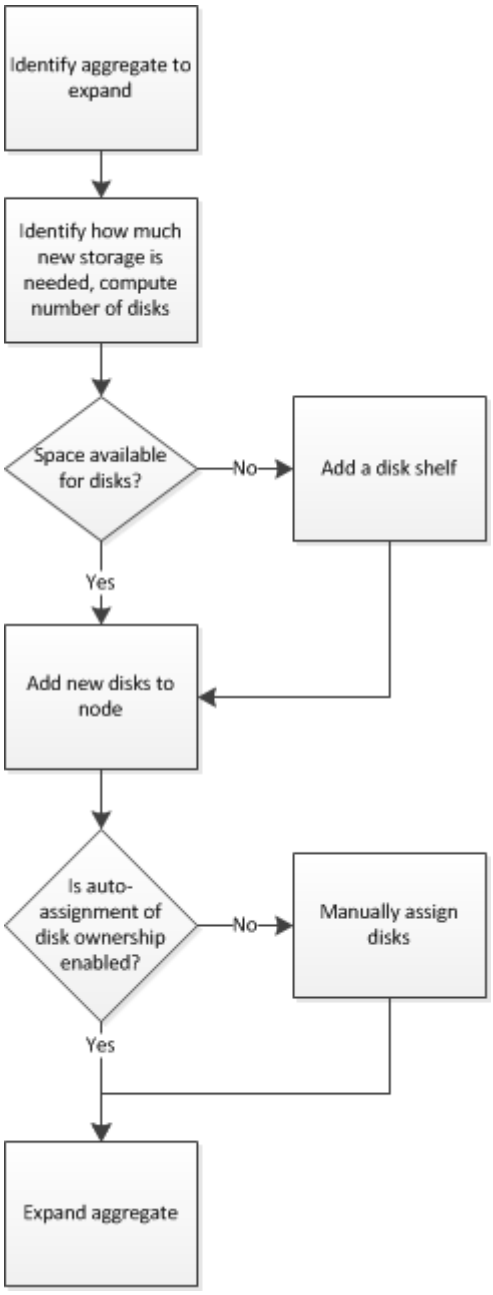
   storage aggregate show-status *aggregate_name*

Related concepts

Configurations that support root-data partitioning on page 46
How root-data partitioning works on page 48

Aggregate expansion workflow

Expanding an aggregate involves identifying the aggregate to expand, determining how much new storage is needed, installing new disks, assigning disk ownership, and creating new a RAID group if needed.



Adding disks to a node

You add disks to a node to increase the number of hot spares, to add space to an aggregate, or to replace disks.

Before you begin

You must have confirmed that your platform model supports the type of disk you want to add.

Steps

1. Check the NetApp Support Site for newer disk and shelf firmware and Disk Qualification Package files.

   If your node does not have the latest versions, you must update them before installing the new disk.

2. Install the disks according to the hardware guide for your disk shelf or the hardware and service guide for your platform.

   The new disks are not recognized until they are assigned to a node and pool. You can assign the new disks manually, or you can wait for Data ONTAP to automatically assign the new disks if your node follows the rules for disk autoassignment.

3. After the new disks have all been recognized, verify their addition and their ownership information:

   `storage aggregate show-spare-disks`

   You should see the new disks, owned by the correct node and in the correct pool.

4. Optional: Zero the newly added disks:

   `storage disk zerospares`

   Disks that have been used previously in a Data ONTAP aggregate must be zeroed before they can be added to another aggregate. Zeroing the disks now can prevent delays in case you need to quickly increase the size of an aggregate. The disk zeroing command runs in the background and can take hours to complete, depending on the size of the non-zeroed disks in the node.

Result

The new disks are ready to be added to an aggregate, used to replace an existing disk, or placed onto the list of hot spares.

Related concepts

When you need to update the Disk Qualification Package on page 43

Manually assigning disk ownership

Disks must be owned by a node before they can be used in an aggregate. If your cluster is not configured to use automatic disk ownership assignment, you must assign ownership manually. You cannot reassign ownership of a disk that is in use in an aggregate.

Steps

1. Display all unowned disks:

   `storage disk show -container-type unassigned`

2. Assign each disk:

   storage disk assign -disk *disk_name* -owner *owner_name*

   You can use the wildcard character to assign more than one disk at once. If you are reassigning a spare disk that is already owned by a different node, you must use the `-force` option

Expanding aggregates

You can add disks to an aggregate so that it can provide more storage to its associated volumes. The procedure for adding partitioned disks to an aggregate is similar to the procedure for adding unpartitioned disks.

Before you begin

You must know what the RAID group size is for the aggregate you are adding the storage to.

About this task

When you expand an aggregate, you should be aware of whether you are adding partition or unpartitioned disks to the aggregate. When you add unpartitioned drives to an existing aggregate, the size of the existing RAID groups is inherited by the new RAID group, which can affect the number of parity disks required. If an unpartitioned disk is added to a RAID group composed of partitioned disks, the new disk is partitioned, leaving an unused spare partition.

When you provision partitions, you must ensure that you do not leave the node without a drive with both partitions as spare. If you do, and the node experiences a controller disruption, valuable information about the problem (the core file) might not be available to provide to the technical support.

Steps

1. Show the available spare storage on the system that owns the aggregate:

   storage aggregate show-spare-disks -original-owner *node_name*

   You can use the `-is-disk-shared` parameter to show only partitioned drives or only unpartitioned drives.

   Example

```
cl1-s2::> storage aggregate show-spare-disks -original-owner cl1-s2 -is-disk-shared true

Original Owner: cl1-s2
 Pool0
  Shared HDD Spares
                                                    Local    Local
                                                     Data     Root Physical
 Disk                      Type     RPM Checksum    Usable   Usable     Size Status
 --------------------------- ----- ------ -------------- -------- -------- -------- --------
 1.0.1                     BSAS    7200 block          753.8GB  73.89GB  828.0GB zeroed
 1.0.2                     BSAS    7200 block          753.8GB       0B  828.0GB zeroed
 1.0.3                     BSAS    7200 block          753.8GB       0B  828.0GB zeroed
 1.0.4                     BSAS    7200 block          753.8GB       0B  828.0GB zeroed
 1.0.8                     BSAS    7200 block          753.8GB       0B  828.0GB zeroed
 1.0.9                     BSAS    7200 block          753.8GB       0B  828.0GB zeroed
 1.0.10                    BSAS    7200 block               0B  73.89GB  828.0GB zeroed
2 entries were displayed.
```

2. Show the current RAID groups for the aggregate:

   storage aggregate show-status *aggr_name*

   Example

```
cl1-s2::> storage aggregate show-status -aggregate data_1

Owner Node: cl1-s2
 Aggregate: data_1 (online, raid_dp) (block checksums)
  Plex: /data_1/plex0 (online, normal, active, pool0)
```

```
    RAID Group /data_1/plex0/rg0 (normal, block checksums)
                                                            Usable Physical
      Position Disk                      Pool Type     RPM     Size     Size Status
      -------- ------------------------- ---- ------ ------ -------- -------- ----------
      shared   1.0.10                     0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.5                      0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.6                      0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.11                     0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.0                      0   BSAS    7200  753.8GB  828.0GB (normal)
5 entries were displayed.
```

3. Simulate adding the storage to the aggregate:

   storage aggregate add-disks -aggregate *aggr_name* -diskcount *number_of_disks_or_partitions* -simulate true

   You can see the result of the storage addition without actually provisioning any storage. If any warnings are displayed from the simulated command, you can adjust the command and repeat the simulation.

   Example

```
cl1-s2::> storage aggregate add-disks data_1 -diskcount 5 -simulate true

Addition of disks would succeed for aggregate "data_1" on node "cl1-s2". The
following disks would be used to add to the aggregate: 1.0.2, 1.0.3, 1.0.4, 1.0.8, 1.0.9.
```

4. Add the storage to the aggregate:

   storage aggregate add-disks -aggregate *aggr_name* -raidgroup new -diskcount *number_of_disks_or_partitions*

   When creating a Flash Pool aggregate, if you are adding disks with a different checksum than the aggregate, or if you are adding disks to a mixed checksum aggregate, you must use the `-checksumstyle` parameter.

   If you are adding disks to a Flash Pool aggregate, you must use the `-disktype` parameter to specify the disk type.

   You can use the `-disksize` parameter to specify a size of the disks to add. Only disks with approximately the specified size are selected for addition to the aggregate.

   Example

```
cl1-s2::> storage aggregate add-disks -aggregate data_1 -raidgroup new -diskcount 5
```

5. Verify that the storage was added successfully:

   storage aggregate show-status -aggregate *aggr_name*

   Example

```
cl1-s2::> storage aggregate show-status -aggregate data_1

Owner Node: cl1-s2
 Aggregate: data_1 (online, raid_dp) (block checksums)
  Plex: /data_1/plex0 (online, normal, active, pool0)
   RAID Group /data_1/plex0/rg0 (normal, block checksums)
                                                            Usable Physical
      Position Disk                      Pool Type     RPM     Size     Size Status
      -------- ------------------------- ---- ------ ------ -------- -------- ----------
      shared   1.0.10                     0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.5                      0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.6                      0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.11                     0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.0                      0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.2                      0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.3                      0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.4                      0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.8                      0   BSAS    7200  753.8GB  828.0GB (normal)
      shared   1.0.9                      0   BSAS    7200  753.8GB  828.0GB (normal)
10 entries were displayed.
```

6. Verify that the node still has at least one drive with both the root partition and the data partition as spare:

   storage aggregate show-spare-disks -original-owner *node_name*

Example

```
cl1-s2::> storage aggregate show-spare-disks -original-owner cl1-s2 -is-disk-shared true

Original Owner: cl1-s2
 Pool0
  Shared HDD Spares
                                                            Local    Local
                                                             Data     Root Physical
 Disk                       Type     RPM Checksum         Usable   Usable     Size Status
 -------------------------- ----- ------ -------------- -------- -------- -------- --------
 1.0.1                      BSAS    7200 block           753.8GB  73.89GB  828.0GB zeroed
 1.0.10                     BSAS    7200 block                0B  73.89GB  828.0GB zeroed
2 entries were displayed.
```

Managing aggregates

You create and manage your aggregates so that they can provide storage to their associated volumes.

Correcting misaligned spare partitions

When you add partitioned disks to an aggregate, you must leave a disk with both the root and data partition available as spare for every node. If you do not and your node experiences a disruption, Data ONTAP might not be able to create a core file.

Before you begin

You must have both a spare data partition and a spare root partition on the same type of disk owned by the same node.

Steps

1. Display the spare partitions for the node:

   storage aggregate show-spare-disks -original-owner *node_name*

   Note which disk has a spare data partition (spare_data) and which disk has a spare root partition (spare_root). The spare partition will show a non-zero value under the `Local Data Usable` or `Local Root Usable` column.

2. Replace the disk with a spare data partition with the disk with the spare root partition:

   storage disk replace -disk *spare_data* -replacement *spare_root* -action start

   You can copy the data in either direction; however, copying the root partition takes less time to complete.

3. Monitor the progress of the disk replacement:

   storage aggregate show-status -aggregate *aggr_name*

4. After the replacement operation is complete, display the spares again to confirm that you have a full spare disk:

   storage aggregate show-spare-disks -original-owner *node_name*

   You should see a spare disk with usable space under both `Local Data Usable` and `Local Root Usable`.

Example

You display your spare partitions for node c1-01 and see that your spare partitions are not aligned:

```
c1::> storage aggregate show-spare-disks -original-owner c1-01

Original Owner: c1-01
 Pool0
  Shared HDD Spares
                                                    Local    Local
                                                     Data     Root Physical
 Disk                       Type     RPM Checksum  Usable   Usable     Size
 -------------------------- ----- ------ -------------- -------- -------- --------
 1.0.1                      BSAS    7200 block      753.8GB       0B  828.0GB
 1.0.10                     BSAS    7200 block           0B  73.89GB  828.0GB
```

You start the disk replacement job:

```
c1::> storage disk replace -disk 1.0.1 -replacement 1.0.10 -action start
```

While you are waiting for the replacement operation to finish, you display the progress of the operation:

```
c1::> storage aggregate show-status -aggregate aggr0_1

Owner Node: c1-01
 Aggregate: aggr0_1 (online, raid_dp) (block checksums)
  Plex: /aggr0_1/plex0 (online, normal, active, pool0)
   RAID Group /aggr0_1/plex0/rg0 (normal, block checksums)
                                                            Usable Physical
     Position Disk                        Pool Type     RPM     Size     Size Status
     -------- --------------------------- ---- ----- ------ -------- -------- ----------
     shared   1.0.1                        0   BSAS    7200  73.89GB  828.0GB (replacing, copy in
progress)
     shared   1.0.10                       0   BSAS    7200  73.89GB  828.0GB (copy 63% completed)
     shared   1.0.0                        0   BSAS    7200  73.89GB  828.0GB (normal)
     shared   1.0.11                       0   BSAS    7200  73.89GB  828.0GB (normal)
     shared   1.0.6                        0   BSAS    7200  73.89GB  828.0GB (normal)
     shared   1.0.5                        0   BSAS    7200  73.89GB  828.0GB (normal)
```

After the replacement operation is complete, you confirm that you have a full spare disk:

```
ie2220::> storage aggregate show-spare-disks -original-owner c1-01

Original Owner: c1-01
 Pool0
  Shared HDD Spares
                                                      Local    Local
                                                       Data     Root Physical
 Disk                       Type     RPM Checksum     Usable   Usable     Size
 -------------------------- ----- ------ -------------- -------- -------- --------
 1.0.1                      BSAS    7200 block          753.8GB  73.89GB  828.0GB
```

Determining drive and RAID group information for an aggregate

Some aggregate administration tasks require that you know what types of drives compose the aggregate, their size, checksum, and status, whether they are shared with other aggregates, and the size and composition of the RAID groups.

Step

1. Show the drives for the aggregate, by RAID group:

   `storage aggregate show-status` ***`aggr_name`***

   The drives are displayed for each RAID group in the aggregate.

   You can see the RAID type of the drive (data, parity, dparity) in the `Position` column. If the `Position` column displays `shared`, then the drive is shared: if it is an HDD, it is a partitioned disk; if it is an SSD, it is part of a storage pool.

Example: A Flash Pool aggregate using an SSD storage pool and data partitions

```
cluster1::> storage aggregate show-status nodeA_fp_1

Owner Node: cluster1-a
 Aggregate: nodeA_fp_1 (online, mixed_raid_type, hybrid) (block checksums)
  Plex: /nodeA_fp_1/plex0 (online, normal, active, pool0)
   RAID Group /nodeA_fp_1/plex0/rg0 (normal, block checksums, raid_dp)
                                                            Usable Physical
     Position Disk                        Pool Type     RPM     Size     Size Status
     -------- --------------------------- ---- ----- ------ -------- -------- -------
     shared   2.0.1                        0   SAS    10000  472.9GB  547.1GB (normal)
     shared   2.0.3                        0   SAS    10000  472.9GB  547.1GB (normal)
     shared   2.0.5                        0   SAS    10000  472.9GB  547.1GB (normal)
     shared   2.0.7                        0   SAS    10000  472.9GB  547.1GB (normal)
     shared   2.0.9                        0   SAS    10000  472.9GB  547.1GB (normal)
     shared   2.0.11                       0   SAS    10000  472.9GB  547.1GB (normal)

   RAID Group /nodeA_flashpool_1/plex0/rg1 (normal, block checksums, raid4) (Storage
Pool: SmallSP)
```

```
                                                                  Usable Physical
     Position Disk                        Pool Type     RPM     Size     Size Status
     -------- --------------------------- ---- ----- ------ -------- -------- -------
     shared   2.0.13                       0   SSD        -  186.2GB  745.2GB (normal)
     shared   2.0.12                       0   SSD        -  186.2GB  745.2GB (normal)
8 entries were displayed.
```

Relocating aggregate ownership within an HA pair

You can change the ownership of aggregates among the nodes in an HA pair without interrupting service from the aggregates.

Both nodes in an HA pair are physically connected to each other's disks or array LUNs. Each disk or array LUN is owned by one of the nodes. Although ownership of disks temporarily changes when a takeover occurs, the aggregate relocation operations either permanently (for example, if done for load balancing) or temporarily (for example, if done as part of takeover) change the ownership of all disks or array LUNs within an aggregate from one node to the other. The ownership changes without any data-copy processes or physical movement of the disks or array LUNs.

Relocating aggregate ownership

You can change the ownership of an aggregate only between the nodes within an HA pair.

About this task

- Because volume count limits are validated programmatically during aggregate relocation operations, it is not necessary to check for this manually.
  If the volume count exceeds the supported limit, the aggregate relocation operation fails with a relevant error message.
- You should not initiate aggregate relocation when system-level operations are in progress on either the source or the destination node; likewise, you should not start these operations during the aggregate relocation.
  These operations can include the following:
  - Takeover
  - Giveback
  - Shutdown
  - Another aggregate relocation operation
  - Disk ownership changes
  - Aggregate or volume configuration operations
  - Storage controller replacement
  - Data ONTAP upgrade
  - Data ONTAP revert
- If you have a MetroCluster configuration, you should not initiate aggregate relocation while disaster recovery operations (*switchover*, *healing*, or *switchback*) are in progress.
- If you have a MetroCluster configuration and initiate aggregate relocation on a switched-over aggregate, the operation might fail because it exceeds the DR partner's volume limit count.
- You should not initiate aggregate relocation on aggregates that are corrupt or undergoing maintenance.

- If the source node is used by an Infinite Volume with SnapDiff enabled, you must perform additional steps before initiating the aggregate relocation and then perform the relocation in a specific manner.
  You must ensure that the destination node has a namespace mirror constituent and make decisions about relocating aggregates that include namespace constituents.
  Infinite volumes management
- Before initiating the aggregate relocation, you should save any core dumps on the source and destination nodes.

Steps

1. View the aggregates on the node to confirm which aggregates to move and ensure they are online and in good condition:

   storage aggregate show -node *source-node*

   #### Example

   The following command shows six aggregates on the four nodes in the cluster. All aggregates are online. Node1 and Node3 form an HA pair and Node2 and Node4 form an HA pair.

   ```
   cluster::> storage aggregate show
   Aggregate     Size Available Used% State   #Vols  Nodes  RAID Status
   --------- -------- --------- ----- ------- ------ ------ -----------
   aggr_0     239.0GB   11.13GB   95% online       1 node1  raid_dp,
                                                            normal
   aggr_1     239.0GB   11.13GB   95% online       1 node1  raid_dp,
                                                            normal
   aggr_2     239.0GB   11.13GB   95% online       1 node2  raid_dp,
                                                            normal
   aggr_3     239.0GB   11.13GB   95% online       1 node2  raid_dp,
                                                            normal
   aggr_4     239.0GB   238.9GB    0% online       5 node3  raid_dp,
                                                            normal
   aggr_5     239.0GB   239.0GB    0% online       4 node4  raid_dp,
                                                            normal
   6 entries were displayed.
   ```

2. Issue the command to start the aggregate relocation:

   storage aggregate relocation start -aggregate-list *aggregate-1, aggregate-2*... -node *source-node* -destination *destination-node*

   The following command moves the aggregates aggr_1 and aggr_2 from Node1 to Node3. Node3 is Node1's HA partner. The aggregates can be moved only within the HA pair.

   ```
   cluster::> storage aggregate relocation start -aggregate-list aggr_1,
   aggr_2 -node node1 -destination node3
   Run the storage aggregate relocation show command to check relocation
   status.
   node1::storage aggregate>
   ```

3. Monitor the progress of the aggregate relocation with the `storage aggregate relocation show` command:

   storage aggregate relocation show -node *source-node*

   #### Example

   The following command shows the progress of the aggregates that are being moved to Node3:

```
cluster::> storage aggregate relocation show -node node1
Source Aggregate   Destination    Relocation Status
------ ----------- -------------  ------------------------
node1
       aggr_1      node3          In progress, module: wafl
       aggr_2      node3          Not attempted yet
2 entries were displayed.
node1::storage aggregate>
```

When the relocation is complete, the output of this command shows each aggregate with a relocation status of `Done`.

Commands for aggregate relocation

There are specific Data ONTAP commands for relocating aggregate ownership within an HA pair.

If you want to...	Use this command...	For more information...
Start the aggregate relocation process	`storage aggregate relocation start`	*ONTAP 9 man page: storage aggregate relocation start*
Monitor the aggregate relocation process	`storage aggregate relocation show`	*ONTAP 9 man page: storage aggregate relocation show*

Related information

ONTAP 9 Commands: Manual Page Reference

Assigning aggregates to SVMs

If you assign one or more aggregates to a Storage Virtual Machine (SVM, formerly known as Vserver), then you can use only those aggregates to contain volumes for that SVM. Assigning aggregates to your SVMs is particularly important in a multi-tenancy environment or when you use Infinite Volumes.

Before you begin

The SVM and the aggregates you want to assign to that SVM must already exist.

About this task

Assigning aggregates to your SVMs helps you keep your SVMs isolated from each other; this is especially important in a multi-tenancy environment. If you use Infinite Volumes, or plan to use them in the future, you must assign aggregates to your SVMs to keep your Infinite Volumes from impacting each other and any FlexVol volumes in your cluster.

Steps

1. Check the list of aggregates already assigned to the SVM:

   `vserver show -fields aggr-list`

   The aggregates currently assigned to the SVM are displayed. If there are no aggregates assigned, "-" is displayed.

2. Add or remove assigned aggregates, depending on your requirements:

If you want to...	Use this command...
Assign additional aggregates	`vserver add-aggregates`

If you want to...	Use this command...
Unassign aggregates	`vserver remove-aggregates`

The listed aggregates are assigned to or removed from the SVM. If the SVM already has volumes that use an aggregate that is not assigned to the SVM, a warning message is displayed, but the command is completed successfully. Any aggregates that were already assigned to the SVM and that were not named in the command are unaffected.

Example

In the following example, the aggregates aggr1 and aggr2 are assigned to SVM svm1:

```
vserver add-aggregates -vserver svm1 -aggregates aggr1,aggr2
```

Methods to create space in an aggregate

If an aggregate runs out of free space, various problems can result that range from loss of data to disabling a volume's guarantee. There are multiple ways to make more space in an aggregate.

All of the methods have various consequences. Prior to taking any action, you should read the relevant section in the documentation.

The following are some common ways to make space in an aggregate, in order of least to most consequences:

- Add disks to the aggregate.
- Move some volumes to another aggregate with available space.
- Shrink the size of volume-guaranteed volumes in the aggregate.
  You can do this manually or with the `autoshrink` option of the autosize capability.
- Change volume guarantee types to **none** on volumes that are using large amounts of space (large volume-guaranteed volumes with large reserved files) so that the volumes take up less space in the aggregate.
  A volume with a guarantee type of **none** has a smaller footprint in the aggregate than a volume with a guarantee type of **`volume`**.
- Delete unneeded volume Snapshot copies if the volume's guarantee type is **none**.
- Delete unneeded volumes.
- Enable space-saving features, such as deduplication or compression.
- (Temporarily) disable features that are using a large amount of metadata .

Determining which volumes reside on an aggregate

You might need to determine which FlexVol volumes or Infinite Volume constituents reside on an aggregate before performing operations on the aggregate, such as relocating it or taking it offline.

About this task

Infinite Volume constituents are somewhat similar to FlexVol volumes, but you usually do not manage them directly. For more information about Infinite Volumes and constituents, see the *Clustered Data ONTAP Infinite Volumes Management Guide*.

Step

1. Enter the appropriate command, depending on whether your system has Infinite Volumes:

If your system...	Then use this command...
Does not have Infinite Volumes	`volume show -aggregate aggregate_name`
Has Infinite Volumes	`volume show -is-constituent * -aggregate aggregate_name`

All volumes (and, if you have Infinite Volumes, constituents) that reside on the specified aggregate are displayed.

Determining whether a Flash Pool aggregate is using an SSD storage pool

You manage Flash Pool aggregates differently when they use SSD storage pools to provide their cache than when they use discrete SSDs.

Step

1. Display the aggregate's drives by RAID group:

   `storage aggregate show-status aggr_name`

   If the aggregate is using one or more SSD storage pools, the value for the `Position` column for the SSD RAID groups is displayed as `Shared`, and the name of the storage pool is displayed next to the RAID group name.

Commands for managing aggregates

You use the `storage aggregate` command to manage your aggregates.

If you want to...	Use this command...
Display the size of the cache for all Flash Pool aggregates	`storage aggregate show -fields hybrid-cache-size-total -hybrid-cache-size-total >0`
Display disk information and status for an aggregate	`storage aggregate show-status`
Display spare disks by node	`storage aggregate show-spare-disks`
Display the root aggregates in the cluster	`storage aggregate show -has-mroot true`
Display basic information and status for aggregates	`storage aggregate show`
Display the type of storage used in an aggregate	`storage aggregate show -fields storage-type`
Bring an aggregate online	`storage aggregate online`
Delete an aggregate	`storage aggregate delete`
Put an aggregate into the restricted state	`storage aggregate restrict`

If you want to...	Use this command...
Rename an aggregate	`storage aggregate rename`
Take an aggregate offline	`storage aggregate offline`
Change the RAID type for an aggregate	`storage aggregate modify -raidtype`

Related information

ONTAP 9 Commands: Manual Page Reference

What aggregates are

To support the differing security, backup, performance, and data sharing needs of your users, you can group the physical data storage resources on your storage system into one or more aggregates. You can then design and configure these aggregates to provide the appropriate level of performance and redundancy.

Each aggregate has its own RAID configuration, plex structure, and set of assigned drives or array LUNs. The aggregate provides storage, based on its configuration, to its associated FlexVol volumes or Infinite Volume.

Aggregates have the following characteristics:

- They can be composed of drives or array LUNs.
- They can be mirrored or unmirrored.
- If they are composed of drives, they can be single-tier (composed of only HDDs or only SSDs) or they can be Flash Pool aggregates, which include both HDD RAID groups and an SSD cache.

The cluster administrator can assign one or more aggregates to a Storage Virtual Machine (SVM), in which case you can use only those aggregates to contain volumes for that SVM.

Aggregate types

Before you create a new aggregate, you must decide the type of aggregate to create based on your application and business needs, the type of system you have, and the type of disks available in your system. Systems are generally set up to support one aggregate type.

There are four types of aggregates you can create:

- All flash aggregates
  All flash aggregates only use solid state drives (SSDs), so they are sometimes referred to as SSD-only aggregates. SSDs are flash media-based storage devices that provide better overall performance than hard disk drives (HDDs), which are mechanical devices using rotating media. You can create an all flash (SSD-only) aggregate on any FAS system if you have SSDs available. However, if you have an All Flash FAS (AFF) system, you must create an all flash aggregate. You cannot create any other aggregate type on an AFF system.
  All flash aggregates are best for running high performance, business critical applications that require the lowest possible latency and highest possible performance.
- Flash Pool aggregates
  Flash Pool aggregates provide a balance of price and performance by combining the speed of flash with the capacity of HDDs.
  You can create storage tiers within Flash Pool aggregates that allow you to deploy flash as high performance cache for your working data set and use HDDs for other needs.
- Performance HDD aggregates
  Performance HDD aggregates provide 10K and 15K RPM speeds. They do not leverage flash, but provide greater speed than capacity HDD aggregates. Performance HDD aggregates are a good option for capacity and less critical business applications.
- Capacity HDD aggregates
  Capacity HDD aggregates provide the slowest response rate for retrieving data at 7.2K RPM. Capacity HDD aggregates are good options for non-critical operations such as backups and archives.

How you use SSDs to increase storage performance

Solid-state drives (SSDs) are flash media-based storage devices that provide better overall performance than hard disk drives (HDDs), which are mechanical devices using rotating media. You should understand how Data ONTAP manages SSDs and the capability differences between SSDs and HDDs.

Depending on your storage system model, you can use SSDs in two ways:

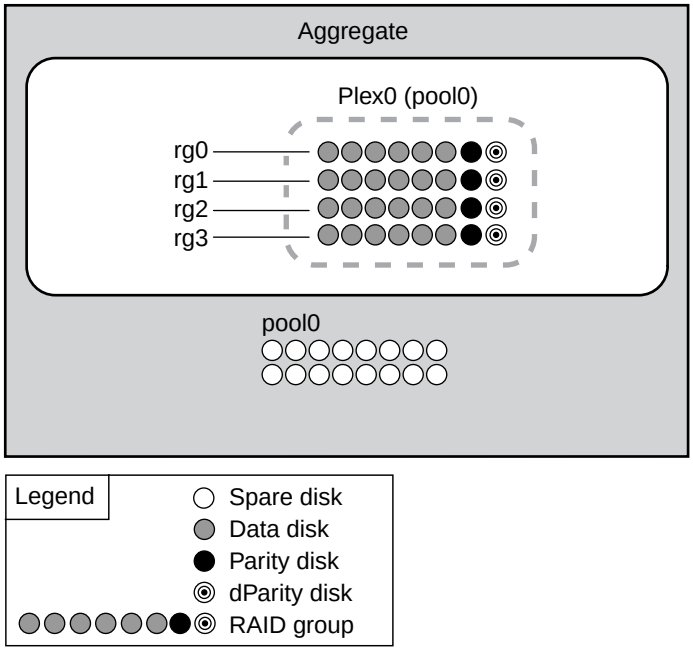
- You can create Flash Pool aggregates—aggregates composed mostly of HDDs, but with some SSDs that function as a high-performance cache for your working data set.
- You can create aggregates composed entirely of SSDs, where the SSDs function as the persistent storage for all data in the aggregate.

You manage Flash Pool aggregates and aggregates composed entirely of SSDs the same way you manage aggregates composed entirely of HDDs. However, there are some differences in the way you manage SSDs from the way you manage disks. In addition, some Data ONTAP capabilities are not available on SSDs and Flash Pool aggregates.

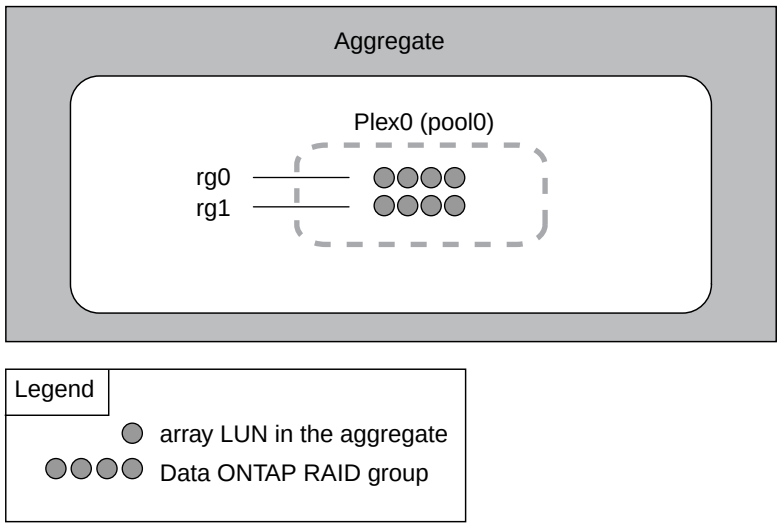
How unmirrored aggregates work

Unless you are using SyncMirror, all of your aggregates are unmirrored. Unmirrored aggregates have only one *plex* (copy of their data), which contains all of the RAID groups belonging to that aggregate.

The following diagram shows an unmirrored aggregate composed of disks, with its one plex. The aggregate has four RAID groups: rg0, rg1, rg2, and rg3. Each RAID group has 6 data disks, one parity disk, and one dparity (double parity) disk. All disks used by the aggregate come from the same pool, pool0.



The following diagram shows an unmirrored aggregate with array LUNs, with its one plex. It has two RAID groups, rg0 and rg1. All array LUNs used by the aggregate come from the same pool, pool0.



How mirrored aggregates work

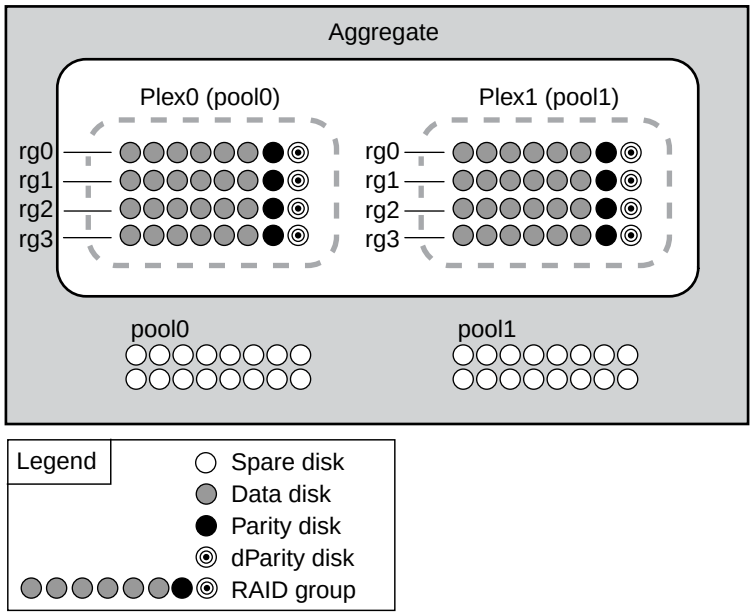
Mirrored aggregates have two *plexes* (copies of their data), which use the SyncMirror functionality to duplicate the data to provide redundancy.

When a mirrored aggregate is created (or when a second plex is added to an existing unmirrored aggregate), Data ONTAP copies the data in the original plex (plex0) to the new plex (plex1). The plexes are physically separated (each plex has its own RAID groups and its own pool), and the plexes are updated simultaneously. This provides added protection against data loss if more disks fail than the RAID level of the aggregate protects against or there is a loss of connectivity, because the unaffected plex continues to serve data while you fix the cause of the failure. After the plex that had a problem is fixed, the two plexes resynchronize and reestablish the mirror relationship.

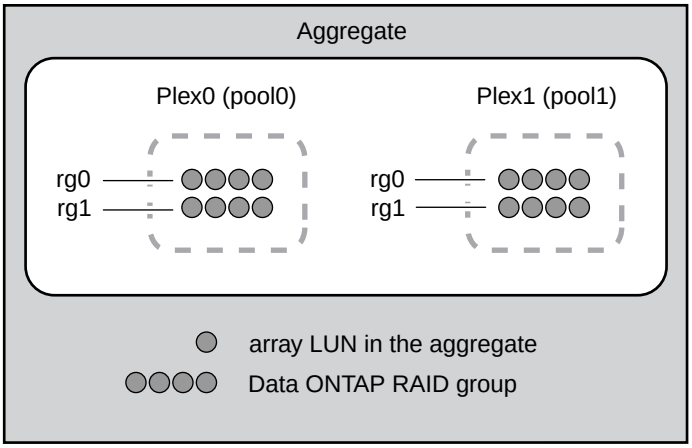
Note: The time for the two plexes to resynchronize can vary and depends on many variables such as aggregate size, system load, how much data has changed, and so on.

The disks and array LUNs on the system are divided into two pools: pool0 and pool1. Plex0 gets its storage from pool0 and plex1 gets its storage from pool1.

The following diagram shows an aggregate composed of disks with SyncMirror enabled and implemented. A second plex has been created for the aggregate, plex1. The data in plex1 is a copy of the data in plex0, and the RAID groups are also identical. The 32 spare disks are allocated to pool0 or pool1, 16 disks for each pool.



The following diagram shows an aggregate composed of array LUNs with SyncMirror enabled and implemented. A second plex has been created for the aggregate, plex1. Plex1 is a copy of plex0, and the RAID groups are also identical.



What a Flash Pool aggregate is

A Flash Pool aggregate combines both SSDs and HDDs (performance or capacity) to provide a high-performance aggregate more economically than an SSD aggregate.

The SSDs provide a high-performance cache for the active data set of the data volumes provisioned on the Flash Pool aggregate, offloading random read operations and repetitive random write operations to improve response times and overall throughput for disk I/O-bound data access operations. (Performance is not significantly increased for predominately sequential workloads.)

How Flash Pool aggregate caching policies work

Caching policies are applied to volumes that reside in Flash Pool aggregates. You should understand how caching policies work before changing them.

In most cases, the default caching policy of **`auto`** is the best caching policy to use. The caching policy should be changed only if a different policy provides better performance for your workload.

Configuring the wrong caching policy can severely degrade volume performance; the performance degradation could increase gradually over time. You should use caution when modifying caching policies, and if you experience performance issues with a volume for which the caching policy has been changed, you should return the caching policy to **`auto`**.

Caching policies combine a read caching policy and a write caching policy. The policy name concatenates the names of the read caching policy and the write caching policy, separated by a hyphen. Underscores are used within a read or write caching policy name. For example, the all_read_random_write-random_write policy combines the “all_read_random_write” read caching policy and the “random_write” write caching policy. If there is no hyphen in the policy name, the write caching policy is “none”, except for the **`auto`** policy.

Read caching policies optimize for future read performance by placing a copy of the data in the cache in addition to the stored data on HDDs. For read caching policies that insert data into the cache for write operations, the cache operates as a *write-through* cache.

Data inserted into the cache by using the write caching policy exists only in cache; there is no copy in HDDs. Flash Pool cache is RAID protected. Enabling write caching makes data from write operations available for reads from cache immediately, while deferring writing the data to HDDs until it ages out of the cache.

The following table provides a rough approximation of what types of data are inserted into cache for specific read and write caching policies.

Caching policy name	**Insertions using read caching policy**				**Insertions using write caching policy**	**Privilege level**
	Random reads	**Sequential reads**	**Random writes**	**Sequential writes**	**Random overwrites**	
auto	Yes	No	No	No	Yes	admin
none	No	No	No	No	No	admin
random_read	Yes	No	Yes	No	No	advanced
noread-random_write	No	No	No	No	Yes	advanced
meta	Metadata only	No	No	No	No	advanced
meta-random_write	Metadata only	No	No	No	Yes	advanced
random_read_write	Yes	No	Yes	No	No	advanced
random_read_write-random_write	Yes	No	Yes	No	Yes	advanced
all_read	Yes	Yes	No	No	No	advanced
all_read-random_write	Yes	Yes	No	No	Yes	advanced
all_read_random_write	Yes	Yes	Yes	No	No	advanced

Caching policy name	Insertions using read caching policy				Insertions using write caching policy	Privilege level
	Random reads	**Sequential reads**	**Random writes**	**Sequential writes**	**Random overwrites**	
all_read_random_write-random_write	Yes	Yes	Yes	No	Yes	advanced
all	Yes	Yes	Yes	Yes	No	advanced
all-random_write	Yes	Yes	Yes	Yes	Yes	advanced

Metadata is cached for all policies except **none**.

You can change the caching policy for a volume that resides on a Flash Pool aggregate by using the `-caching-policy` parameter with the `volume create` command. When you create a volume on a Flash Pool aggregate, by default, the **`auto`** caching policy is assigned to the volume.

If you move a volume from a Flash Pool aggregate to a single-tier aggregate, it loses its caching policy; if you later move it back to a Flash Pool aggregate, it is assigned the default caching policy of **`auto`**. If you move a volume between two Flash Pool aggregates, the caching policy is preserved.

Determining whether to modify the caching policy of Flash Pool aggregates

Beginning with ONTAP 9.0, you can assign cache-retention policies to volumes in Flash Pool aggregates to determine how long the volume data remains in the Flash Pool cache. However, in some cases changing the cache-retention policy might not impact the amount of time the volume's data remains in the cache.

About this task

If your data meets any of the following conditions, changing your cache-retention policy might not have an impact:

- Your workload is sequential.
- Your workload does not reread the random blocks cached in the solid state drives (SSDs).
- The cache size of the volume is too small.

The following steps check for these conditions. The task must be done in advanced privilege mode.

Steps

1. **`stats start`**
2. Determine the workload pattern of the volume:

   `statistics show -object workload_volume -instance *volume-workload* -counters sequential_reads`

3. Determine the hit rate of the volume:

   `statistics show -object wafl_hya_vvol -instance *volume* -counter read_ops_replaced_pwercent|wc_write_blks_overwritten_percent`

4. Determine the `Cacheable Read` and `Project Cache Alloc` of the volume:

   `wafl awa start *aggr_name*`

5. Compare the volume's hit rate to the `Cacheable Read`.

   If the hit rate of the volume is greater than the `Cacheable Read`, then your workload does not reread random blocks cached in the SSDs.

6. Compare the volume's current cache size to the `Project Cache Alloc`.

   If the current cache size of the volume is greater than the `Project Cache Alloc`, then the size of your volume cache is too small.

Modifying caching policies of Flash Pool aggregates

You should modify the caching policy of a volume only if a different caching policy is expected to provide better performance. You can modify the caching policy of a volume on a Flash Pool aggregate.

Before you begin

You must determine whether you want to modify your caching policy.

About this task

In most cases, the default caching policy of **`auto`** is the best caching policy that you can use. The caching policy should be changed only if a different policy provides better performance for your workload. Configuring the wrong caching policy can severely degrade volume performance; the performance degradation could increase gradually over time. You should use caution when modifying caching policies. If you experience performance issues with a volume for which the caching policy has been changed, you should return the caching policy to **`auto`**.

Step

1. Modify the volume's caching policy:

   `volume modify -volume *volume_name* -caching-policy *policy_name*`

Example

The following example modifies the caching policy of a volume named “vol2” to the policy **`none`**:

```
volume modify -volume vol2 -caching-policy none
```

Related tasks

Determining whether to modify the caching policy of Flash Pool aggregates on page 28

Related information

ONTAP 9 man page: volume modify

Setting the cache-retention policy for Flash Pool aggregates

Beginning with ONTAP 9.0, you can assign cache-retention policies to volumes in Flash Pool aggregates. Data in volumes with a high cache-retention policy remains in cache longer and data in volumes with a low cache-retention policy is removed sooner. This increases performance of your critical workloads by making high priority information accessible at a faster rate for a longer period of time.

Before you begin

You should know whether your system has any conditions that might prevent the cache-retention policy from having an impact on how long your data remains in cache.

About this task

The task must be done in advanced privilege mode.

Steps

1. Verify the volume's cache-retention policy:

   By default the cache retention policy is **normal**.

2. Set the cache-retention policy:

   priority hybrid-cache set *volume_name* read-cache=*read_cache_value* write-cache=*write_cache_value* cache-retention-priority=*cache_retention_policy*

   Set *cache_retention_policy* to **high** for data that you want to remain in cache longer. Set *cache_retention_policy* to **low** for data that you want to remove from cache sooner.

3. Verify that the volume's cache-retention policy is changed to the option you selected.

How Flash Pool SSD partitioning works for Flash Pool aggregates using storage pools

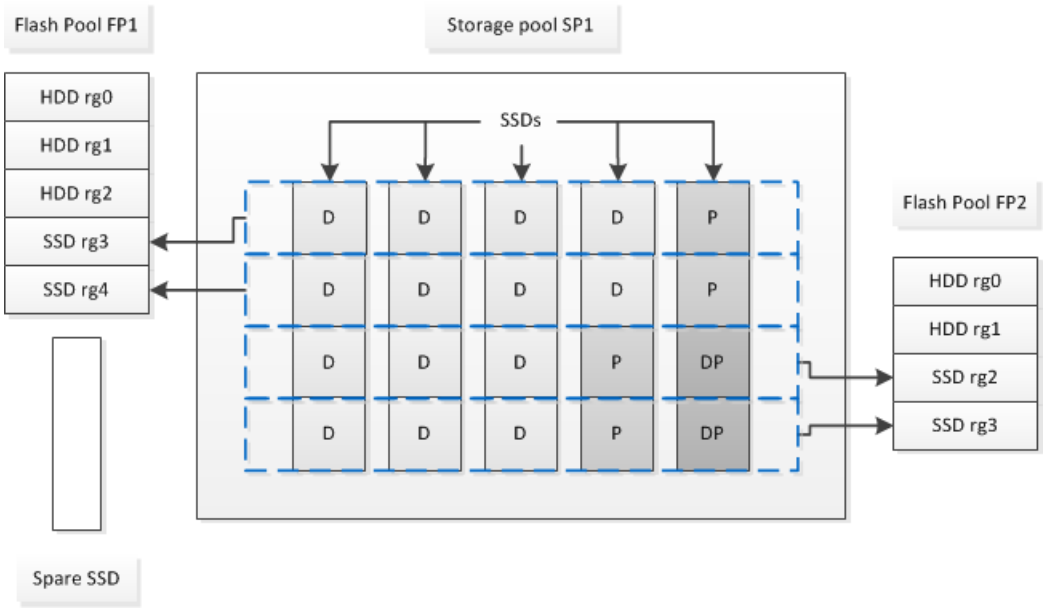
If you are providing cache to two or more Flash Pool aggregates, you should use Flash Pool Solid-State Drive (SSD) partitioning. Flash Pool SSD partitioning allows SSDs to be shared by all the aggregates using the Flash Pool. This spreads the cost of parity over multiple aggregates, increases SSD cache allocation flexibility, and maximizes SSD performance.

For an SSD to be used in a Flash Pool aggregate, the SSD must be placed in a storage pool. You cannot use SSDs that have been partitioned for root-data partitioning in a storage pool. After the SSD is placed in the storage pool, the SSD can no longer be managed as a stand-alone disk and cannot be removed from the storage pool unless you destroy the aggregates associated with the Flash Pool and you destroy the storage pool.

SSD storage pools are divided into four equal allocation units. SSDs added to the storage pool are divided into four partitions and one partition is assigned to each of the four allocation units. The SSDs in the storage pool must be owned by the same HA pair. By default, two allocation units are assigned to each node in the HA pair. Allocation units must be owned by the node that owns the aggregate it is serving. If more Flash cache is required for aggregates on one of the nodes, the default number of allocation units can be shifted to decrease the number on one node and increase the number on the partner node.

You can use only one spare SSD for a storage pool. If the storage pool provides allocation units to Flash Pool aggregates owned by both nodes in the HA pair, then the spare SSD can be owned by either node. However, if the storage pool provides allocation units only to Flash Pool aggregates owned by one of the nodes in the HA pair, then the SSD spare must be owned by that same node.

The following illustration is an example of Flash Pool SSD partitioning. The SSD storage pool provides cache to two Flash Pool aggregates:



Storage pool SP1 is composed of five SSDs and a hot spare SSD. Two of the storage pool's allocation units are allocated to Flash Pool FP1, and two are allocated to Flash Pool FP2. FP1 has a cache RAID type of RAID4. Therefore, the allocation units provided to FP1 contain only one partition designated for parity. FP2 has a cache RAID type of RAID-DP. Therefore, the allocation units provided to FP2 include a parity partition and a double-parity partition.

In this example, two allocation units are allocated to each Flash Pool aggregate. However, if one Flash Pool aggregate required a larger cache, you could allocate three of the allocation units to that Flash Pool aggregate, and only one to the other.

Restrictions of Flash Pool aggregates using SSD storage pools

You must be aware of some restrictions of Flash Pool aggregates that use solid-state drive (SSD) storage pools.

- Reduced fault isolation
  The loss of a single SSD affects RAID groups that include one of its partitions. Every Flash Pool aggregate that has cache allocated from the SSD storage pool that contains the failed SSD has one or more RAID groups in reconstruction.
- Mirrored aggregates can coexist with Flash Pool aggregates that use storage pools, but Flash Pool aggregates cannot be mirrored.
- You cannot use the following technologies with Flash Pool aggregates that use SSD storage pools:
  - MetroCluster configurations
  - Sync Mirror
  - Physical SSDs
    Flash Pool aggregates can use SSD storage pools or physical SSDs, but not both.

Creating a Flash Pool aggregate using physical SSDs

You create a Flash Pool aggregate by enabling the feature on an existing aggregate composed of HDD RAID groups, and then adding one or more SSD RAID groups to that aggregate. This results in

two sets of RAID groups for that aggregate: SSD RAID groups (the SSD cache) and HDD RAID groups.

Before you begin

- You must have identified a valid aggregate composed of HDDs to convert to a Flash Pool aggregate.
- You must have determined write-caching eligibility of the volumes associated with the aggregate, and completed any required steps to resolve eligibility issues.
- You must have determined the SSDs you will be adding, and these SSDs must be owned by the node on which you are creating the Flash Pool aggregate.
- You must have determined the checksum types of both the SSDs you are adding and the HDDs already in the aggregate.
- You must have determined the number of SSDs you are adding and the optimal RAID group size for the SSD RAID groups.
  Using fewer RAID groups in the SSD cache reduces the number of parity disks required, but larger RAID groups require RAID-DP.
- You must have determined the RAID level you want to use for the SSD cache.
- You must have determined the maximum cache size for your system and determined that adding SSD cache to your aggregate will not cause you to exceed it.
- You must have familiarized yourself with the configuration requirements for Flash Pool aggregates.

About this task

After you add an SSD cache to an aggregate to create a Flash Pool aggregate, you cannot remove the SSD cache to convert the aggregate back to its original configuration.

By default, the RAID level of the SSD cache is the same as the RAID level of the HDD RAID groups. You can override this default selection by specifying the `raidtype` option when you add the first SSD RAID groups.

Steps

1. Mark the aggregate as eligible to become a Flash Pool aggregate:

   `storage aggregate modify -aggregate` *`aggr_name`* `-hybrid-enabled true`

   If this step does not succeed, determine write-caching eligibility for the target aggregate.

2. Add the SSDs to the aggregate by using the `storage aggregate add` command.

   You can specify the SSDs by ID or by using the `diskcount` and `disktype` parameters.

   If the HDDs and the SSDs do not have the same checksum type, or if the aggregate is a mixed-checksum aggregate, then you must use the `checksumstyle` parameter to specify the checksum type of the disks you are adding to the aggregate.

   You can specify a different RAID type for the SSD cache by using the `raidtype` parameter.

   If you want the cache RAID group size to be different from the default for the RAID type you are using, you should change it now, by using the `-cache-raid-group-size` parameter.

Creating a Flash Pool aggregate using SSD storage pools

You create a Flash Pool aggregate with SSD storage pools by enabling the feature on an existing aggregate composed of HDD RAID groups, and then adding one or more SSD storage pool allocation units to that aggregate.

Before you begin

- You must have identified a valid aggregate composed of HDDs to convert to a Flash Pool aggregate.
- You must have determined write-caching eligibility of the volumes associated with the aggregate, and completed any required steps to resolve eligibility issues.
- You must have created an SSD storage pool to provide the SSD cache to this Flash Pool aggregate.
  Any allocation unit from the storage pool that you want to use must be owned by the same node that owns the Flash Pool aggregate.
- You must have determined how much cache you want to add to the aggregate.
  You add cache to the aggregate by allocation units. You can increase the size of the allocation units later by adding SSDs to the storage pool if there is room.
- You must have determined the RAID type you want to use for the SSD cache.
  After you add a cache to the aggregate from SSD storage pools, you cannot change the RAID type of the cache RAID groups.
- You must have determined the maximum cache size for your system and determined that adding SSD cache to your aggregate will not cause you to exceed it.
  You can see the amount of cache that will be added to the total cache size by using the `storage pool show` command.
- You must have familiarized yourself with the configuration requirements for Flash Pool aggregates.

About this task

If you want the RAID type of the cache to different from that of the HDD RAID groups, you must specify the cache RAID type when you add the SSD capacity. After you add the SSD capacity to the aggregate, you can no longer change the RAID type of the cache.

After you add an SSD cache to an aggregate to create a Flash Pool aggregate, you cannot remove the SSD cache to convert the aggregate back to its original configuration.

Steps

1. Mark the aggregate as eligible to become a Flash Pool aggregate:

   `storage aggregate modify -aggregate` *`aggr_name`* `-hybrid-enabled true`

   If this step does not succeed, determine write-caching eligibility for the target aggregate.

2. Show the available SSD storage pool allocation units:

   `storage pool show-available-capacity`

3. Add the SSD capacity to the aggregate:

   `storage aggregate add` *`aggr_name`* `-storage-pool` *`sp_name`* `-allocation-units` *`number_of_units`*

   If you want the RAID type of the cache to be different from that of the HDD RAID groups, you must change it when you enter this command by using the `raidtype` parameter.

You do not need to specify a new RAID group; Data ONTAP automatically puts the SSD cache into separate RAID groups from the HDD RAID groups.

You cannot set the RAID group size of the cache; it is determined by the number of SSDs in the storage pool.

The cache is added to the aggregate and the aggregate is now a Flash Pool aggregate. Each allocation unit added to the aggregate becomes its own RAID group.

4. Optional: Confirm the presence and size of the SSD cache:

   `storage aggregate show` *`aggr_name`*

   The size of the cache is listed under `Total Hybrid Cache Size`.

Related information

NetApp Technical Report 4070: Flash Pool Design and Implementation Guide

Creating an SSD storage pool

You can create solid state drive (SSD) storage pools to provide SSD cache for two to four Flash Pool aggregates.

About this task

- You must supply a disk list when creating or adding disks to a storage pool.
  Storage pools do not support a `diskcount` parameter.
- The SSDs used in the storage pool should be the same size.

Steps

1. Determine the names of the available spare SSDs:

   `storage aggregate show-spare-disks -disk-type SSD`

   The SSDs used in a storage pool can be owned by either node of an HA pair.

2. Create the storage pool:

   `storage pool create -storage-pool` *`sp_name`* `-disk-list` *`disk1,disk2,...`*

3. Optional: Verify the newly created storage pool:

   `storage pool show -storage-pool` *`sp_name`*

Result

After the SSDs are placed into the storage pool, they no longer appear as spares on the cluster, even though the storage provided by the storage pool has not yet been allocated to any Flash Pool caches. You cannot add SSDs to a RAID group as discrete drives; their storage can be provisioned only by using the allocation units of the storage pool to which they belong.

Adding SSDs to an SSD storage pool

When you add solid state drives (SSDs) to an SSD storage pool, you increase the storage pool's physical and usable sizes and allocation unit size. The larger allocation unit size also affects allocation units that have already been allocated to Flash Pool aggregates.

Before you begin

You must have determined that this operation will not cause you to exceed the cache limit for your HA pair. Data ONTAP does not prevent you from exceeding the cache limit when you add SSDs to an SSD storage pool, and doing so can render the newly added storage capacity unavailable for use.

About this task

When you add SSDs to an existing SSD storage pool, the SSDs must be owned by one node or the other of the same HA pair that already owned the existing SSDs in the storage pool. You can add SSDs that are owned by either node of the HA pair.

The SSD you add to the storage pool must be the same size as disk currently used in the storage pool.

Steps

1. Optional: View the current allocation unit size and available storage for the storage pool:

   storage pool show -instance *sp_name*

2. Find available SSDs:

   storage disk show -container-type spare -type SSD

3. Add the SSDs to the storage pool:

   storage pool add -storage-pool *sp_name* -disk-list *disk1,disk2…*

   The system displays which Flash Pool aggregates will have their size increased by this operation and by how much, and prompts you to confirm the operation.

Determining Flash Pool candidacy and optimal cache size

Before converting an existing aggregate to a Flash Pool aggregate, you can determine whether the aggregate is I/O bound, and what would be the best Flash Pool cache size for your workload and budget. You can also check whether the cache of an existing Flash Pool aggregate is sized correctly.

Before you begin

You should know approximately when the aggregate you are analyzing experiences its peak load.

Steps

1. Enter advanced mode:

   set advanced

2. If you need to determine whether an existing aggregate would be a good candidate for conversion to a Flash Pool aggregate, determine how busy the disks in the aggregate are during a period of peak load, and how that is affecting latency:

   statistics show-periodic -object disk:raid_group -instance *raid_group_name* -counter disk_busy|user_read_latency -interval 1 -iterations 60

   You can decide whether reducing latency by adding Flash Pool cache makes sense for this aggregate.

Example

The following command shows the statistics for the first RAID group of the aggregate “aggr1”:

```
statistics show-periodic -object disk:raid_group -instance /aggr1/
plex0/rg0 -counter disk_busy|user_read_latency -interval 1 -iterations
60
```

3. Start Automated Workload Analyzer (AWA):

   system node run -node *node_name* wafl awa start *aggr_name*

   AWA begins collecting workload data for the volumes associated with the specified aggregate.

4. Exit advanced mode:

   set admin

   Allow AWA to run until one or more intervals of peak load have occurred. AWA collects workload statistics for the volumes associated with the specified aggregate, and analyzes data for up to one rolling week in duration. Running AWA for more than one week will report only on data collected from the most recent week. Cache size estimates are based on the highest loads seen during the data collection period; the load does not need to be high for the entire data collection period.

5. Enter advanced mode:

   set advanced

6. Display the workload analysis:

   system node run -node *node_name* wafl awa print

   You can use the `-t` option to show information about the volumes in the aggregate that are the best candidates for being on a Flash Pool aggregate.

   AWA displays the workload statistics and optimal Flash Pool cache size.

7. Stop AWA:

   system node run -node *node_name* wafl awa stop

   All workload data is flushed and is no longer available for analysis.

8. Exit advanced mode:

   set admin

Example

In the following example, AWA was run on aggregate “aggr1”. Here is the output of the `awa print` command after AWA had been running for about 3 days (442 10-minute intervals):

```
    ### FP AWA Stats ###

                    Host lada66a                    Memory 93788 MB
           ONTAP Version NetApp Release
R8_3_1x_awa_2809198_1504220853: Tue Apr 21 18:35:12 PDT 2015

Basic Information
               Aggregate lada66a_aggr1
            Current-time Thu Apr 23 16:42:17 PDT 2015
              Start-time Thu Apr 23 08:03:51 PDT 2015
      Total runtime (sec) 31103
    Interval length (sec) 600
          Total intervals 54
        In-core Intervals 1024

Summary of the past 20 intervals
                                max
```

```
                                    ------------
        Read Throughput (MB/s): 181.772
       Write Throughput (MB/s): 550.611
            Cacheable Read (%): 12
           Cacheable Write (%): 30
Max Projected Cache Size (GiB): 787.077

Summary Cache Hit Rate vs. Cache Size
Referenced Cache Size (GiB): 787.077
Referenced Interval: ID 53 starting at Thu Apr 23 16:33:07 PDT 2015
         Size        20%        40%        60%        80%
100%
 Read Hit (%)          9         20         28         32
35
Write Hit (%)         17         21         23         25         30
```

The results provide the following pieces of information:

- `Read Throughput` and `Write Throughput`
  The throughput measurements can help you identify an aggregate that is receiving a higher amount of traffic. Note that these numbers do not indicate whether that aggregate is I/O bound.
- `Max Projected Cache Size`
  The size at which the SSD cache would hold every eligible data block that was requested from disk during the AWA run. Note that this does not guarantee a hit for all future I/O operations, because they might request data that is not in the cache. However, if the workload during the AWA run was a typical one, and if your budget allows for it, this would be an ideal size for your Flash Pool cache.
- `Projected Read Offload` and `Projected Write Offload`
  The approximate percentages of read and write operations that would have been handled by a Flash Pool cache of the optimal size rather than going to disk (projected cache hit rate). Note that this number is related to the performance increase you would see by converting the aggregate to a Flash Pool aggregate, but not an exact prediction.
- `Summary Cache Hit Rate vs. Cache Size`
  This table can help you predict the performance impact of decreasing the size of the SSD cache from `Max Projected Cache Size`. These values are highly impacted by your workload. Depending on whether data that was aged out of the cache was ever accessed again, the impact of decreasing the size of the cache might be large or almost nonexistent. You can use this table to find the right balance between cost and performance for your workload and budget.

Determining the impact to cache size of adding SSDs to an SSD storage pool

If adding SSDs to a storage pool causes your platform model's cache limit to be exceeded, Data ONTAP does not allocate the newly added capacity to any Flash Pool aggregates. This can result in some or all of the newly added capacity being unavailable for use.

About this task

When you add SSDs to an SSD storage pool that has allocation units already allocated to Flash Pool aggregates, you increase the cache size of each of those aggregates and the total cache on the system. If none of the storage pool's allocation units have been allocated, adding SSDs to that storage pool does not affect the SSD cache size until one or more allocation units are allocated to a cache.

Steps

1. Determine the usable size of the SSDs you are adding to the storage pool:

   `storage disk show disk_name -fields usable-size`

2. Determine how many allocation units remain unallocated for the storage pool:

   `storage pool show-available-capacity sp_name`

   All unallocated allocation units in the storage pool are displayed.

3. Calculate the amount of cache that will be added by applying the following formula:

   (4 – number of unallocated allocation units) × 25% × usable size × number of SSDs

Commands for managing SSD storage pools

Data ONTAP provides the `storage pool` command for managing SSD storage pools.

If you want to...	Use this command...
Display how much storage a storage pool is providing to which aggregates	`storage pool show-aggregate`
Display how much cache would be added to the overall cache capacity for both RAID types (allocation unit data size)	`storage pool show -instance`
Display the disks in a storage pool	`storage pool show-disks`
Display the unallocated allocation units for a storage pool	`storage pool show-available-capacity`
Change the ownership of one or more allocation units of a storage pool from one HA partner to the other	`storage pool reassign`

Related information

ONTAP 9 Commands: Manual Page Reference

How the SVM affects which aggregates can be associated with a FlexVol volume

FlexVol volumes are always associated with one Storage Virtual Machines (SVMs), and one aggregate that supplies its storage. The SVM can limit which aggregates can be associated with that volume, depending on how the SVM is configured.

When you create a FlexVol volume, you specify which SVM the volume will be created on, and which aggregate that volume will get its storage from. All of the storage for the newly created FlexVol volume comes from that associated aggregate.

If the SVM for that volume has aggregates assigned to it, then you can use only one of those assigned aggregates to provide storage to volumes on that SVM. This can help you be sure that your SVMs are not sharing physical storage resources inappropriately. This segregation can be important in a multi-tenancy environment, because for some space management configurations, volumes that share the same aggregate can affect each other's access to free space when space is constrained for the aggregate. Aggregate assignment requirements apply to both cluster administrators and SVM administrators.

Volume move operation is not constrained by the SVM aggregate assignments, so if you are trying to keep your SVMs on separate aggregates, you must be sure that you do not violate your SVM aggregate assignments when you perform this operation.

If the SVM for that volume has no aggregates assigned to it, then the cluster administrator can use any aggregate in the cluster to provide storage to the new volume. However, the SVM administrator cannot create volumes for SVMs with no assigned aggregates. For this reason, if you want your SVM administrator to be able to create volumes for a specific SVM, then you must assign aggregates to that SVM.

Changing the aggregates assigned to an SVM does not affect any existing volumes. For this reason, the list of aggregates assigned to an SVM cannot be used to determine the aggregates associated with volumes for that SVM.

How to determine space usage in an aggregate

You can view space usage by all volumes in one or more aggregates with the `aggregate show-space` command. This helps you see which volumes are consuming the most space in their containing aggregates so that you can take actions to free more space.

The used space in an aggregate is directly affected by the space used in the FlexVol volumes and Infinite Volume constituents it contains. Measures that you take to increase space in a volume also affect space in the aggregate.

When the aggregate is offline, no values are displayed. Only non-zero values are displayed in the command output. However, you can use the `-instance` parameter to display all possible feature rows regardless of whether they are enabled and using any space. A value of `-` indicates that there is no data available to display.

The following rows are included in the `aggregate show-space` command output:

- `Volume Footprints`
  The total of all volume footprints within the aggregate. It includes all of the space that is used or reserved by all data and metadata of all volumes in the containing aggregate. It is also the amount of space that is freed if all volumes in the containing aggregate are destroyed. Infinite Volume constituents appear in the output of space usage commands as if the constituents were FlexVol volumes.
- `Aggregate Metadata`
  The total file system metadata required by the aggregate, such as allocation bitmaps and inode files.
- `Snapshot Reserve`
  The amount of space reserved for aggregate Snapshot copies, based on volume size. It is considered used space and is not available to volume or aggregate data or metadata.
- `Snapshot Reserve Unusable`
  The amount of space originally allocated for aggregate Snapshot reserve that is unavailable for aggregate Snapshot copies because it is being used by volumes associated with the aggregate. Can occur only for aggregates with a non-zero aggregate Snapshot reserve.
- `Total Used`
  The sum of all space used or reserved in the aggregate by volumes, metadata, or Snapshot copies.
- `Total Physical Used`
  The amount of space being used for data now (rather than being reserved for future use). Includes space used by aggregate Snapshot copies.

There is never a row for Snapshot spill.

The following example shows the `aggregate show-space` command output for an aggregate whose Snapshot reserve is 5%. If the Snapshot reserve was 0, the row would not be displayed.

```
cluster1::> storage aggregate show-space

                       Aggregate : wqa_gx106_aggr1

      Feature                                 Used      Used%
      --------------------------------    ----------     ------
      Volume Footprints                     101.0MB         0%
      Aggregate Metadata                      300KB         0%
      Snapshot Reserve                       5.98GB         5%

      Total Used                             6.07GB         5%
      Total Physical Used                   34.82KB         0%
```

How you can determine and control a volume's space usage in the aggregate

You can determine which FlexVol volumes and Infinite Volume constituents are using the most space in the aggregate and specifically which features within the volume. The `volume show-footprint` command provides information about a volume's footprint, or its space usage within the containing aggregate.

The `volume show-footprint` command shows details about the space usage of each volume in an aggregate, including offline volumes. This command bridges the gap between the output of the `volume show-space` and `aggregate show-space` commands. All percentages are calculated as a percent of aggregate size.

Only non-zero values are displayed in the command output. However, you can use the `-instance` parameter to display all possible feature rows regardless of whether they are enabled and using any space. A value of **-** indicates that there is no data available to display.

Infinite Volume constituents appear in the output of space usage commands as if the constituents were FlexVol volumes.

The following example shows the `volume show-footprint` command output for a volume called testvol:

```
cluster1::> volume show-footprint testvol

      Vserver : thevs
      Volume  : testvol

      Feature                                 Used    Used%
      --------------------------------    ----------    -----
      Volume Data Footprint                 120.6MB       4%
      Volume Guarantee                       1.88GB      71%
      Flexible Volume Metadata              11.38MB       0%
      Delayed Frees                          1.36MB       0%
      Total Footprint                        2.01GB      76%
```

The following table explains some of the key rows of the output of the `volume show-footprint` command and what you can do to try to decrease space usage by that feature:

Row/feature name	Description/contents of row	Some ways to decrease
`Volume Data Footprint`	The total amount of space used in the containing aggregate by a volume's data in the active file system and the space used by the volume's Snapshot copies. This row does not include reserved space, so if volumes have reserved files, the volume's total used space in the `volume show-space` command output can exceed the value in this row.	• Deleting data from the volume. • Deleting Snapshot copies from the volume.
`Volume Guarantee`	The amount of space reserved by the volume in the aggregate for future writes. The amount of space reserved depends on the guarantee type of the volume.	Changing the type of guarantee for the volume to **none**. This row will go to **0**. If you configure your volumes with a volume guarantee of **none**, you should refer to Technical Report 3965 or 3483 for information about how a volume guarantee of **none** can affect storage availability.
`Flexible Volume Metadata`	The total amount of space used in the aggregate by the volume's metadata files.	No direct method to control.
`Delayed Frees`	Blocks that ONTAP used for performance and cannot be immediately freed. When ONTAP frees blocks in a FlexVol volume, this space is not always immediately shown as free in the aggregate because operations to free the space in the aggregate are batched for increased performance. Blocks that are declared free in the FlexVol volume but that are not yet free in the aggregate are called “delayed free blocks” until the associated delayed free blocks are processed. For SnapMirror destinations, this row has a value of **0** and is not displayed.	No direct method to control.
`File Operation Metadata`	The total amount of space reserved for file operation metadata. After space is used for file operation metadata, it is not returned as free space to the aggregate, but it is reused by subsequent file operations.	No direct method to control.
`Total Footprint`	The total amount of space that the volume uses in the aggregate. It is the sum of all of the rows.	Any of the methods used to decrease space used by a volume.

Related information

NetApp Technical Report 3483: Thin Provisioning in a NetApp SAN or IP SAN Enterprise Environment

NetApp Technical Report 3965: NetApp Thin Provisioning Deployment and Implementation Guide Data ONTAP 8.1 (7-Mode)

Managing disks

You can perform various tasks to manage your disks, including removing a failed disk, removing data from a disk, and removing ownership of a disk. There are also tasks you can perform related to managing disks using root-data partitioning and related to self-encrypting disks.

When you need to update the Disk Qualification Package

The Disk Qualification Package (DQP) adds full support for newly qualified drives. Before you update drive firmware or add new drive types or sizes to a cluster, you must update the DQP. A best practice is to also update the DQP regularly; for example, every quarter or semi-annually.

You need to download and install the DQP in the following situations:

- Whenever you add a new drive type or size to the node
  For example, if you already have 1-TB drives and add 2-TB drives, you need to check for the latest DQP update.
- Whenever you update the disk firmware
- Whenever newer disk firmware or DQP files are available
- Whenever you upgrade to a new version of Data ONTAP.
  The DQP is not updated as part of a Data ONTAP upgrade.

Related information

NetApp Downloads: Disk Qualification Package
NetApp Downloads: Disk Drive and Firmware

Configuring autoassignment of disk ownership

You can configure ONTAP to automatically assign disk ownership according to a disk's stack, shelf, or bay. If configured, automatic disk ownership assignments occur 10 minutes after system initialization and every five minutes during normal system operation.

Before you begin

- Your system must adhere to the requirements for automatic disk ownership.
- If you have multiple stacks or shelves that must have different ownership, one disk must have been manually assigned on each stack or shelf so that automatic ownership assignment works on each stack or shelf.

About this task

The behavior of the **`default`** autoassignment policy depends on the system model. For entry level models, the **`default`** policy is equivalent to the **`bay`** policy. For all other systems, it is equivalent to the **`stack`** policy.

Steps

1. Configure automatic disk assignment:

   `storage disk option modify -autoassign-policy` ***`autoassign_policy`***

- Use **stack** as the *autoassign_policy* to configure automatic ownership at the stack or loop level.
- Use **shelf** as the *autoassign_policy* to configure automatic ownership at the shelf level.
- Use **bay** as the *autoassign_policy* to configure automatic ownership at the bay level.

2. Verify the automatic assignment settings for the disks:

   storage disk option show

   Example

```
cluster1::> storage disk option show

Node           BKg. FW. Upd.  Auto Copy     Auto Assign    Auto
                                                           Assign
Policy
-------------  -------------  ------------  -------------  --------
cluster1-1     on             on            on             default
cluster1-2     on             on            on             default
```

Which disk autoassignment policy to use

You can typically use the default autoassignment policy, which is equivalent to the **stack** policy for most systems, and to the **bay** policy for entry level systems (FAS2xxx). However, for some configurations, you might need to change the autoassignment policy.

Use the appropriate autoassignment, based on your configuration:

If you are using...	Then use the autoassignment policy value of...
Stand-alone entry level system	**stack**
Entry level systems in an HA configuration with a single, shared shelf	**bay**
Entry level systems in an HA configuration with one stack of two or more shelves	**shelf**
MetroCluster configurations with one stack per node, two or more shelves	**shelf**
All other configurations	**stack**

Removing a failed disk

A disk that is completely failed is no longer counted by Data ONTAP as a usable disk, and you can immediately disconnect the disk from the disk shelf. However, you should leave a partially failed disk connected long enough for the Rapid RAID Recovery process to complete.

About this task

If you are removing a disk because it has failed or because it is producing excessive error messages, you should not use the disk again in this or any other storage system.

Steps

1. Find the disk ID of the failed disk by entering the following command:

   storage disk show -broken

If the disk does not appear in the list of failed disks, it might be partially failed, with a Rapid RAID Recovery in process. In this case, you should wait until the disk is present in the list of failed disks (which means that the Rapid RAID Recovery process is complete) before removing the disk.

2. Determine the physical location of the disk you want to remove by entering the following command:

   storage disk set-led -disk *disk_name* 2

   The fault LED on the face of the disk is lit for 2 minutes.

3. Remove the disk from the disk shelf, following the instructions in the hardware guide for your disk shelf model.

Removing ownership from a disk

Data ONTAP writes disk ownership information to the disk. Before you remove a spare disk or its shelf from a node, you should remove its ownership information so that it can be properly integrated into another node.

Before you begin

The disk you want to remove ownership from must meet the following requirements:

- It must be a spare disk.
  You cannot remove ownership from a disk that is being used in an aggregate.
- It cannot be in the maintenance center.
- It cannot be undergoing sanitization.
- It cannot be failed.
  It is not necessary to remove ownership from a failed disk.

About this task

If you have automatic disk assignment enabled, Data ONTAP could automatically reassign ownership before you remove the disk from the node. For this reason, you disable automatic ownership assignment until the disk is removed, and then reenable it.

Steps

1. If disk ownership automatic assignment is on, turn it off:

   storage disk option modify -node *node_name* -autoassign off

2. If needed, repeat the previous step for the node's HA partner.

3. Remove the software ownership information from the disk:

   storage disk removeowner *disk_name*

   To remove ownership information from multiple disks, use a comma-separated list:

   #### Example

   storage disk removeowner sys1:0a.23,sys1:0a.24,sys1:0a.25

4. If the disk is partitioned for root-data partitioning, remove ownership from the partitions by entering both of the following commands:

   storage disk removeowner *disk_name* -root true

```
storage disk removeowner disk_name -data true
```

Both partitions are no longer owned by any node.

5. If you turned off disk ownership automatic assignment previously, turn it on after the disk has been removed or reassigned:

```
storage disk option modify -node node_name -autoassign on
```

6. If needed, repeat the previous step for the node's HA partner.

Configurations that support root-data partitioning

Root-data partitioning helps to maximize storage utilization by using drive partitions to create aggregates instead of using entire drives. Root-data partitioning is supported only on certain configurations.

Beginning with ONTAP 9.0 there are two versions of root-data partitioning: root-data partitioning and root-data-data partitioning.

- Root-data partitioning is supported on the following configurations:
  - All Flash FAS (AFF) platforms
  - Entry-level FAS2xxx platforms
    Root-data partitioning of hard disk drives (HDDs) on entry-level FAS2xxx platforms is only allowed on internal shelves.
  - FAS platforms with only solid state drives (SSDs) attached
- Root-data-data partitioning is supported on the following configurations:
  - AFF platforms
  - FAS platforms with only solid state drives (SSDs) attached

Setting up an active-passive configuration on nodes using root-data partitioning

When an HA pair is configured to use root-data partitioning by the factory, ownership of the data partitions is split between both nodes in the pair, for use in an active-active configuration. If you want to use the HA pair in an active-passive configuration, you must update partition ownership before creating your data aggregate.

Before you begin

- You should have decided which node will be the active node and which node will be the passive node.
- Storage failover must be configured on the HA pair.

About this task

This task is performed on two nodes: Node A and Node B.

All commands are input at the clustershell.

This procedure is designed for nodes for which no data aggregate has been created from the partitioned disks.

Steps

1. View the current ownership of the data partitions:

   `storage aggregate show-spare-disks`

   #### Example

   You can see that half of the data partitions are owned by one node and half are owned by the other node. All of the data partitions should be spare.

```
cluster1::> storage aggregate show-spare-disks

Original Owner: cluster1-01
 Pool0
  Partitioned Spares
                                                            Local    Local
                                                             Data     Root Physical
 Disk                       Type     RPM Checksum         Usable   Usable     Size
 -------------------------- ----- ------ -------------- -------- -------- --------
 1.0.0                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.1                      BSAS    7200 block           753.8GB  73.89GB  828.0GB
 1.0.5                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.6                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.10                     BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.11                     BSAS    7200 block           753.8GB       0B  828.0GB

Original Owner: cluster1-02
 Pool0
  Partitioned Spares
                                                            Local    Local
                                                             Data     Root Physical
 Disk                       Type     RPM Checksum         Usable   Usable     Size
 -------------------------- ----- ------ -------------- -------- -------- --------
 1.0.2                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.3                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.4                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.7                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.8                      BSAS    7200 block           753.8GB  73.89GB  828.0GB
 1.0.9                      BSAS    7200 block           753.8GB       0B  828.0GB
12 entries were displayed.
```

2. Enter the advanced privilege level:

   `set advanced`

3. For each data partition owned by the node that will be the passive node, assign it to the active node:

   `storage disk assign -force -data true -owner` ***`active_node_name`*** **`-disk`** ***`disk_name`***

   You do not need to include the partition as part of the disk name.

   #### Example

   You would enter a command similar to the following example for each data partition you need to reassign:

   `storage disk assign -force -data true -owner cluster1-01 -disk 1.0.3`

4. Confirm that all of the partitions are assigned to the active node.

   #### Example

```
cluster1::*> storage aggregate show-spare-disks

Original Owner: cluster1-01
 Pool0
  Partitioned Spares
                                                            Local    Local
                                                             Data     Root Physical
 Disk                       Type     RPM Checksum         Usable   Usable     Size
 -------------------------- ----- ------ -------------- -------- -------- --------
 1.0.0                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.1                      BSAS    7200 block           753.8GB  73.89GB  828.0GB
 1.0.2                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.3                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.4                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.5                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.6                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.7                      BSAS    7200 block           753.8GB       0B  828.0GB
 1.0.8                      BSAS    7200 block           753.8GB       0B  828.0GB
```

```
  1.0.9                      BSAS    7200 block           753.8GB       0B  828.0GB
  1.0.10                     BSAS    7200 block           753.8GB       0B  828.0GB
  1.0.11                     BSAS    7200 block           753.8GB       0B  828.0GB

Original Owner: cluster1-02
 Pool0
  Partitioned Spares
                                                            Local    Local
                                                             Data     Root Physical
 Disk                       Type     RPM Checksum         Usable   Usable     Size
 -------------------------- ----- ------ -------------- -------- -------- --------
 1.0.8                      BSAS    7200 block                0B  73.89GB  828.0GB
13 entries were displayed.
```

Note that cluster1-02 still owns a spare root partition.

5. Return to administrative privilege:

   set admin

6. Create your data aggregate, leaving at least one data partition as spare:

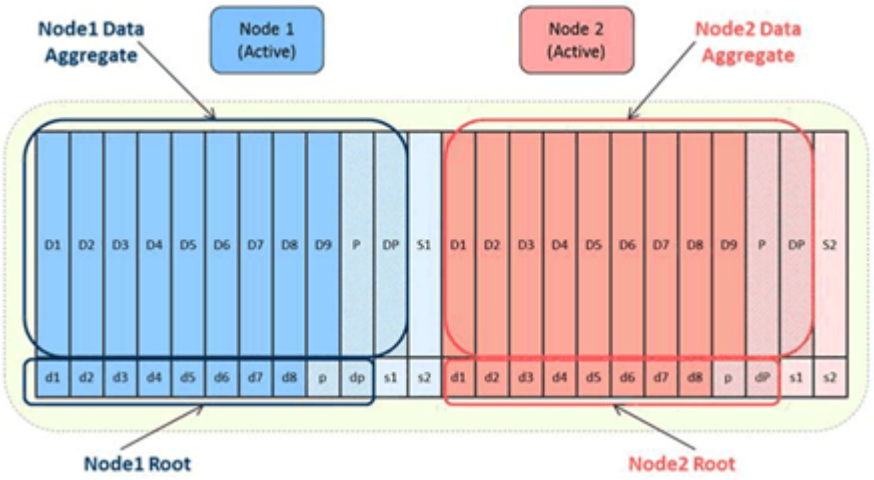
   storage aggregate create *new_aggr_name* -diskcount *number_of_partitions* -node *active_node_name*

   The data aggregate is created and is owned by the active node.

How root-data partitioning works

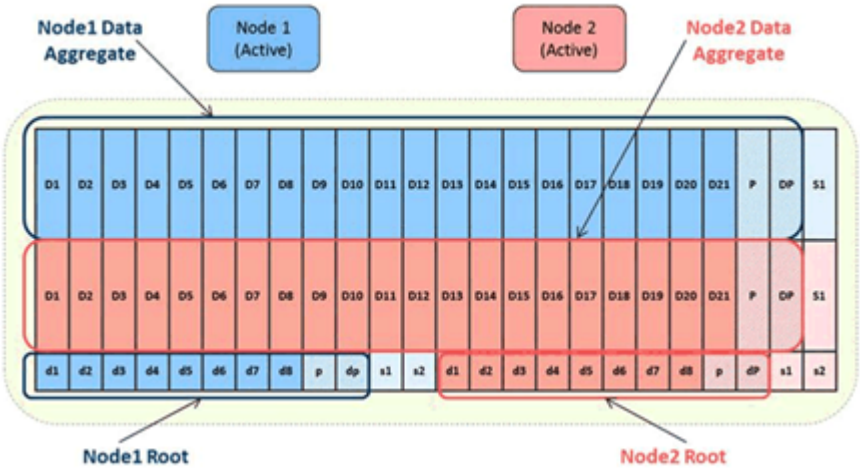
Beginning with ONTAP 9, a new version of root-data partitioning called root-data-data partitioning is available. Both root-data partitioning and root-data-data partitioning enable you to create aggregates using disk partitions instead of complete disks. Both versions are only available on certain configurations.

- Root-data partitioning creates one small partition as the root partition and one large partition for data as shown in the following illustration.



- Root-data-data partitioning creates one small partition as the root partition and two larger, equally sized partitions for data as shown in the following illustration.

  Creating two data partitions enables the same solid-state drive (SSD) to be shared between two nodes and two aggregates.



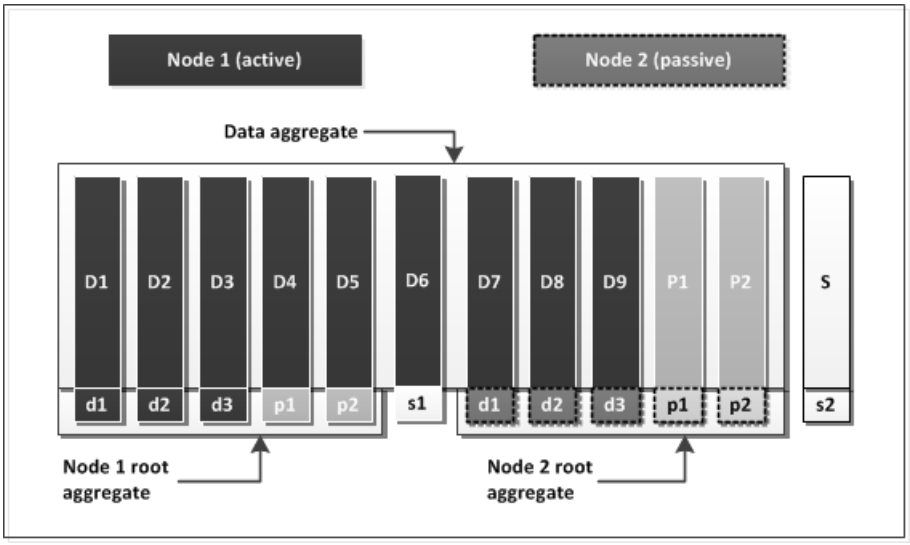
The size of the partitions is set by ONTAP, and depends on the number of disks used to compose the root aggregate when the system is initialized. The more disks used to compose the root aggregate, the smaller the root partition. The data partitions are used to create aggregates. The two data partitions created in root-data-data partitioning are of the same size. After system initialization, the partition sizes are fixed. Adding partitions or disks to the root aggregate after system initialization increases the size of the root aggregate, but does not change the root partition size.

For root-data partitioning and root-data-data partitioning, the partitions are used by RAID in the same manner as physical disks. If a partitioned disk is moved to another node or used in another aggregate, the partitioning persists. You can use the disk only in RAID groups composed of partitioned disks. If you add an unpartitioned drive to a RAID group consisting of partitioned drives, the unpartitioned drive is partitioned to match the partition size of the drives in the RAID group and the rest of the disk is unused.

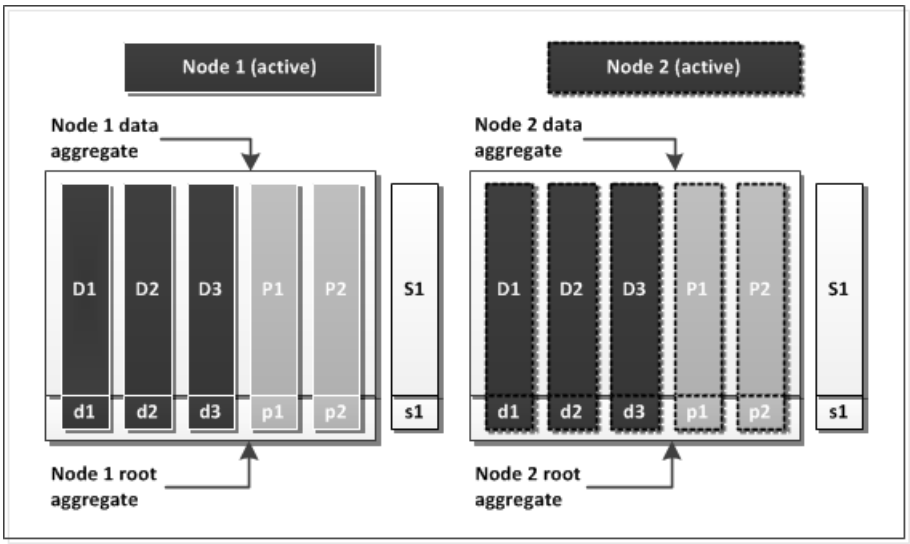
Standard root-data partitioning disk layouts

The root aggregate is configured by the factory. You should not change it. However, you can use the data partitions in a few different configurations, depending on your requirements.

The following diagram shows one way to configure the partitions for an active-passive configuration with 12 partitioned disks using root-data partitioning. This configuration applies in the same way to root-data-data partitioning. There are two root aggregates, one for each node, composed of the small partitions. Each root aggregate has a spare partition. There is one RAID-DP data aggregate with two parity disk partitions and one spare partition.



The following diagram shows one way to configure the partitions for an active-active configuration with 12 partitioned disks using root-data partitioning. This configuration applies in the same way to root-data-data partitioning. In this case, there are two RAID-DP data aggregates, each with their own data partitions, parity partitions, and spares. Each disk is allocated to only one node. This is a best practice that prevents the loss of a single disk from affecting both nodes.



The disks used for data, parity, and spare partitions might not be exactly as shown in these diagrams. For example, the parity partitions might not always align on the same disk.

In ONTAP 9.0, AFF systems initialized with advanced disk partitioning (ADP) will have root aggregates that are provisioned across a maximum of 48 SSD partitions.

Commands for managing disks

You can use the `storage disk` and `storage aggregate` commands to manage your disks.

If you want to...	Use this command...
Display a list of spare disks, including partitioned disks, by owner	`storage aggregate show-spare-disks`
Display the disk RAID type, current usage, and RAID group by aggregate	`storage aggregate show-status`
Display the RAID type, current usage, aggregate, and RAID group, including spares, for physical disks	`storage disk show -raid`
Display a list of failed disks	`storage disk show -broken`
Display the pre-cluster (nodescope) drive name for a disk	`storage disk show -primary-paths` (advanced)
Illuminate the LED for a particular disk or shelf	`storage disk set-led`
Display the checksum type for a specific disk	`storage disk show -fields checksum-compatibility`
Display the checksum type for all spare disks	`storage disk show -fields checksum-compatibility -container-type spare`
Display disk connectivity and placement information	`storage disk show -fields disk,primary-port,secondary-name,secondary-port,shelf,bay`
Display the pre-cluster disk names for specific disks	`storage disk show -disk -fields diskpathnames`
Display the list of disks in the maintenance center	`storage disk show -maintenance`
Display SSD wear life	`storage disk show -ssd-wear`
Unpartition a disk	`system node run -node local -command disk unpartition`
Zero all non-zeroed disks	`storage disk zerospares`
Stop an ongoing sanitization process on one or more specified disks	`disk sanitize abort disk_list`
Display storage encryption disk information	`storage encryption disk show`
Retrieve authentication keys from all linked key management servers	`security key-manager restore`

Related information

ONTAP 9 Commands: Manual Page Reference

Commands for displaying space usage information

You use the `storage aggregate` and `volume` commands to see how space is being used in your aggregates and volumes and their Snapshot copies.

To display information about...	Use this command...	For more information...
Aggregates, including details about used and available space percentages, Snapshot reserve size, and other space usage information	`storage aggregate show` `storage aggregate show-space -fields snap-size-total,used-including-snapshot-reserve`	*ONTAP 9 man page: storage aggregate show* *ONTAP 9 man page: storage aggregate show-space*
How disks and RAID groups are used in an aggregate, and RAID status	`storage aggregate show-status`	*ONTAP 9 man page: storage aggregate show-status*
The amount of disk space that would be reclaimed if you deleted a specific Snapshot copy	`volume snapshot compute-reclaimable`(advanced)	*ONTAP 9 man page: volume snapshot compute-reclaimable*
The amount of space used by a volume	`volume show -fields size,used,available,percent-used` `volume show-space`	*ONTAP 9 man page: volume show* *ONTAP 9 man page: volume show-space*
The amount of space used by a volume in the containing aggregate	`volume show-footprint`	*ONTAP 9 man page: volume show-footprint*

Related information

ONTAP 9 Commands: Manual Page Reference

Commands for displaying information about storage shelves

You use the `storage shelf show` command to display configuration and error information for your disk shelves.

If you want to display...	Use this command...
General information about shelf configuration and hardware status	`storage shelf show`
Detailed information for a specific shelf, including stack ID	`storage shelf show -shelf`
Unresolved, customer actionable, errors by shelf	`storage shelf show -errors`

If you want to display...	Use this command...
Bay information	`storage shelf show -bay`
Connectivity information	`storage shelf show -connectivity`
Cooling information, including temperature sensors and cooling fans	`storage shelf show -cooling`
Information about I/O modules	`storage shelf show -module`
Port information	`storage shelf show -port`
Power information, including PSUs (power supply units), current sensors, and voltage sensors	`storage shelf show -power`

Related information

ONTAP 9 Commands: Manual Page Reference

Support for Storage Encryption

Storage Encryption provides extra data protection by allowing data at rest to only be accessed with an authentication key. It is supported on certain storage controllers and disk shelves that contain self-encrypting disks (SEDs).

Storage Encryption can be supported by external key management servers or by onboard key management functionality. Storage Encryption using external key management servers supports Key Management Interoperability Protocol (KMIP) 1.0 and 1.1 for communication with key management servers. See the Interoperability Matrix for the latest information about supported key management servers, storage systems, and disk shelves.

Storage Encryption does not support the following:

- MetroCluster configurations
- 10 Gb network interfaces for communication with key management servers
  This limitation does not apply to serving data.
- Non-homogenous disk sets
  All disks of an individual node or HA pair in the cluster must have encryption functionality to be able to use Storage Encryption. However, Storage Encryption-enabled HA pairs can coexist with non-Storage Encryption-enabled HA pairs in the same cluster.
- More than 128 authentication keys per cluster.
  You receive a warning when the number of stored authentication keys reaches 100. You cannot create new authentication keys when the number of stored authentication keys reaches the limit of 128. You must then delete unused authentication keys before you can create new ones.
- Various ports for communication with the key management server
  You should use the network interface e0m for communication with key management servers. Depending on the storage controller model, certain network interfaces might not be available during the boot process for communication with key management servers.

Related information

NetApp Interoperability Matrix Tool

How to determine whether you need an external key management server

The Onboard Key Manager is included in ONTAP 9.0 and later for Storage Encryption. Onboard key management is easier to deploy and less expensive to set up than an external key management server. However, there are several reasons why you still might need to use an external key management server.

You should set up an external key management server if any of the following are true for your environment:

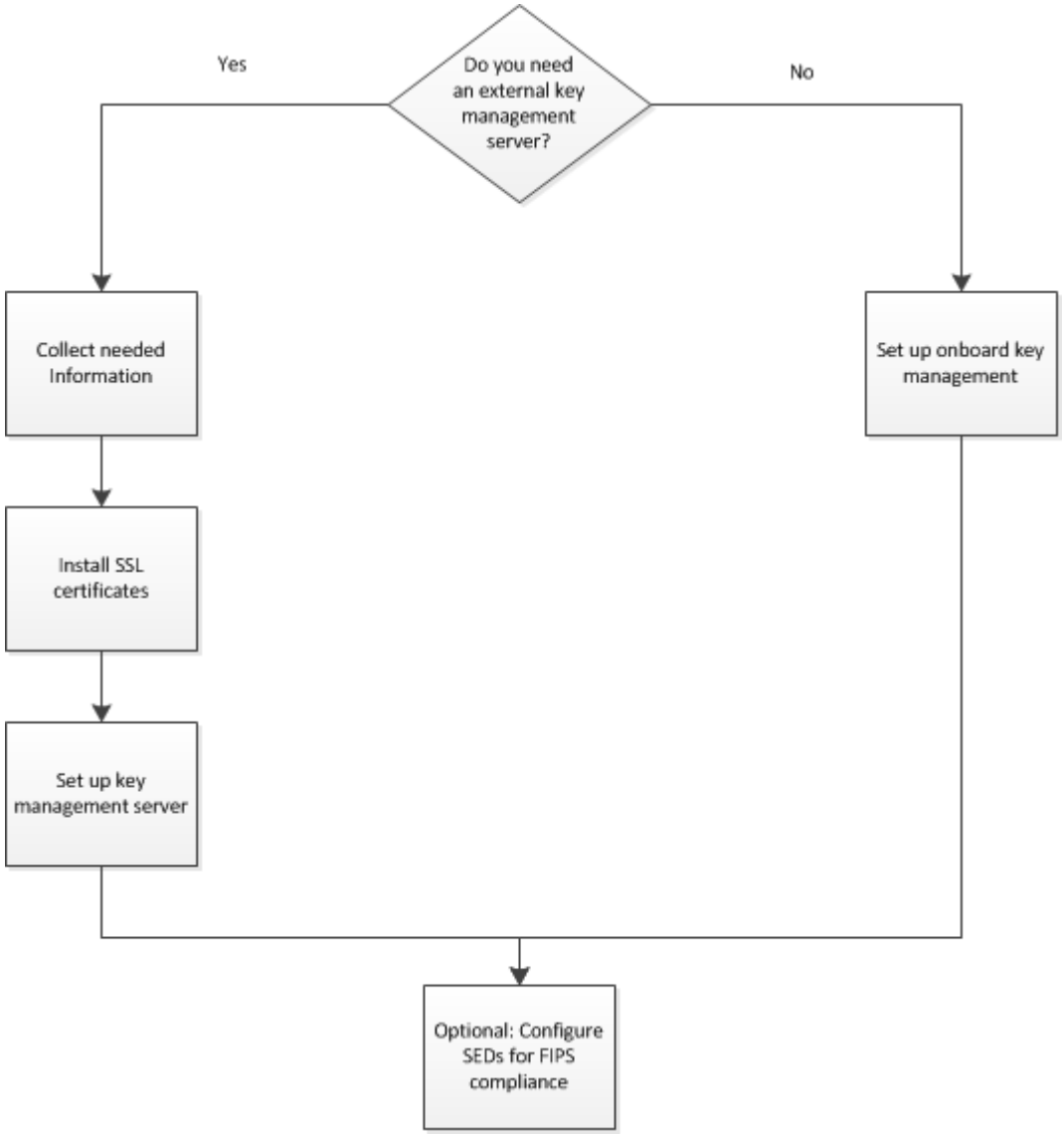
- Your encryption key management solution must comply with Federal Information Processing Standards (FIPS) 140-2.
- You need a multi-cluster solution.
  With an external key management server, you can have a single key server solution that supports multiple clusters with centralized management of all encryption keys.

- You need support of industry OASIS Key Management Interoperability Protocol (KMIP) standard.
- Your business requires the added security of storing your encryption keys on a system or in a location different from the data.

  With onboard key management, the data and the encryption keys are stored on the same system. With an external key server, the data and keys are stored separately.

Key management setup workflow

The steps necessary to set up key management for Storage Encryption vary depending on whether you need external key management. If you need external key management, you must collect information, install SSL certificates, and set up a key management server. If you do not need external key management, you can set up onboard key management.



Related information

NetApp Technical Report 3954: NetApp Technical Report 3954: NetApp Storage Encryption Preinstallation Requirements and Procedures for IBM Tivoli Lifetime Key Manager

NetApp Technical Report 4074: NetApp Technical Report 4074: NetApp Storage Encryption Preinstallation Requirements and Procedures for SafeNet KeySecure

Setting up external key management for Storage Encryption

Setting up external key management involves the following tasks.

Steps

1. *Collect information for configuring Storage Encryption with external key management servers* on page 56.
2. *Install SSL certificates on the cluster* on page 57.
3. *Set up external key management servers* on page 58.
4. *Optional: Configure self-encrypting disks for FIPS 140-2 compliance* on page 59.

Information to collect before configuring Storage Encryption with external key management servers

Certain information is required to successfully configure Storage Encryption on your cluster with an external key management server. Other optional information might also be needed based upon your preferences.

The following information is required to configure Storage Encryption.

- Network interface name
  The name of the network interface the cluster should use to communicate with external key management servers
- Network interface IP address in IPv4 or IPv6 format.
- IPv6 network prefix length
  This is only needed if you specify an IPv6 network interface address.
- Network interface subnet mask
- Network interface gateway IP address
- IP addresses for a minimum of two external key management servers
  You should link the cluster to two or more external key management servers to prevent loss of data access in the event of a single server failure.
- IPv6 address for the cluster network interface
  This is only needed if you specify an IPv6 address for the external key management servers.
- Public KMIP client SSL certificate for the cluster
- Private KMIP client SSL certificate for the cluster
- Public Certificate authority (CA) SSL certificate for the key management server

The following information is optional depending on your configuration preferences.

- Port number for each external key management server
  The port number must be the same for all key management servers. If you do not provide a port number, it defaults to port 5696, which is the Internet Assigned Numbers Authority (IANA) assigned port for the Key Management Interoperability Protocol (KMIP).

- Key tag name
  The key tag name is used to identify all keys belonging to a particular cluster. The default key tag name is the cluster name.

Installing SSL certificates on the cluster

The cluster and key management servers use SSL connections for secure communications. Before you can set up and configure Storage Encryption, you must install public and private KMIP SSL certificates for the cluster and each key management server to verify each other's identity and to establish the needed SSL connections.

Before you begin

- The time must be synchronized between the following systems:
  - The server creating the certificates
  - The key management servers
  - The cluster
- You must have the public SSL KMIP client certificate (`client.pem`).
- You must have the private SSL KMIP client certificate (`client_private.pem`).
  This SSL KMIP client certificate must not be password protected.
- You must have the necessary certificate authority (CA) public key management server SSL certificate (`key_management_server_ipaddress_CA.pem`).
  The IP address of `key_management_server_ipaddress` must be identical to the IP address of the key management server that you use to identify it when running the Storage Encryption setup wizard.

About this task

In an HA pair, both nodes must use the same public and private KMIP SSL certificates. If you want multiple HA pairs that are connected to the same key management server to have access to each other's keys, all nodes in all HA pairs must use the same public and private KMIP SSL certificates.

Steps

1. Install the SSL KMIP client certificate for the cluster:

   `security certificate install -vserver` *`admin_svm_name`* `-type client -subtype kmip-cert`

   It is important to use the `-subtype kmip-cert` parameter so that the certificate is installed properly for use with key management servers.

2. Enter the public and private certificate as prompted.

3. Install the SSL public certificate for the root certificate authority (CA) of the KMIP server:

   `security certificate install -vserver` *`admin_svm_name`* `-type server-ca -subtype kmip-cert -kmip-server-ip` *`key_management_server_ipaddress`*

   If you are using the same root CA for multiple key management servers with IPv4 addresses, enter the subnet address that covers all key management server IP addresses. If they are on completely different networks, you can use the subnet address 0.0.0.0 as a wildcard.

   If your key management servers use IPv6 addresses, you must use a separate root CA for each one.

Example

If your key management server IP addresses are 172.18.12.9, 172.18.248.123, and 172.18.99.97, and they all use the same root CA, add them at the same time by using the subnet address 172.18.0.0 instead:

```
security certificate install -vserver svm1 -type server-ca -subtype
kmip-cert -kmip-server-ip 172.18.0.0
```

4. If you are linking multiple key management servers to the cluster and they use individual root CAs, repeat the previous step for each public certificate of each key management server.

   You can link up to four key management servers.

Related information

ONTAP 9 System Administration Reference

Setting up external key management servers

You can set up an external key management servers so that your storage system can securely store and retrieve authentication keys for self-encrypting disks (SEDs) in a location separate from your data. You can link up to four key management servers. A minimum of two is recommended for redundancy and disaster recovery.

Before you begin

- All disks in the node or HA pair must have encryption functionality.
  See the Interoperability Matrix for a lists of disks that support Storage Encryption.
- You must have collected the information necessary to configure Storage Encryption.
- The necessary SSL KMIP client and server certificates must be installed.

About this task

External key management servers are set up using the `security key-manager setup` command. This command is only available to cluster administrators.

Steps

1. Launch the key management setup wizard from the storage system prompt:

   ```
   security key-manager setup
   ```

   You can use the `-node` parameter to specify the name of the node you want to configure key management settings on.

2. Use the information you collected in advance to complete the steps in the wizard.
   - Select `no`, when prompted for whether to use onboard key management.
   - Use `e0m` for the network interface.

Example

```
hpcl::> security key-manager setup

Would you like to configure onboard key-management? {yes, no} [no]:
Would you like to use KMIP server configuration? {yes, no} [yes]:

Enter the TCP port number for KMIP server [5696]:
Enter the network interface [e0c]:
Would you like to configure an IPv4 address? {yes, no} [yes]:
```

```
Enter the IP address: [20.1.1.1]:
Enter the netmask: [255.255.1.1]:
Enter the gateway: [20.1.1.5]:
Would you like to configure an IPv6 address? {yes, no} [no]:
```

3. Repeat these steps for each node in the cluster.
4. Add an additional key management server for redundancy:

   security key-manager add -address *key_management_server_ipaddress*

   You can add up to four key management servers.
5. Verify that all the added key management servers are linked:

   security key-manager show -status

   All linked key management servers are displayed and listed as **available** under the **Status** column.

Related information

NetApp Interoperability Matrix Tool

Configuring self-encrypting disks for FIPS 140-2 compliance

Federal Information Processing Standards (FIPS) 140-2 compliance is required for some industries. If needed, you can configure your self-encrypting disks (SEDs) to run in FIPS 140-2 compliance mode.

Before you begin

Your drive firmware must support FIPS 140-2 compliance. The Interoperability Matrix contains information about supported drive firmware versions.

About this task

Configuring SEDs to run in FIPS 140-2 compliance mode automatically enables power-on lock protection. This means that the disks require authentication after being power-cycled.

Step

1. Configure SEDs to run in FIPS 140-2 compliance mode:

   storage encryption disk modify -disk *disk_id* -fips-key-id *fips_authentication_key_id*

   The FIPS authentication key can be different from the data encryption authentication key.

The following example enables FIPS 140-2 compliance mode on all SEDs in disk stack 2.10:

```
cluster1::> storage encryption disk modify -disk 2.10.* -fips-key-id
6A1E21D8000000000100000000000005A1FB4EE8F62FD6D8AE6754C9019F35A

Info: Starting modify on 14
disks.
            |
      View the status of the operation by using
the                                                                    |
      storage encryption disk show-status command.
```

Related information

NetApp Interoperability Matrix Tool

Enabling cluster-wide FIPS-compliant mode

Cluster-wide FIPS-compliant mode is disabled by default. You can modify your cluster-wide security configuration to enable FIPS-compliant mode on all nodes in your cluster.

Before you begin

Your Key Management Interoperability Protocol (KMIP) server must support TLSv1.2. TLSv1.2 is required for ONTAP to complete the connection to the external KMIP server when cluster-wide FIPS-compliant mode is enabled.

Steps

1. Verify that TLSv1.2 is supported:

   `security config show -supported-protocols`

2. Enable cluster-wide FIPS-compliant mode:

   `security config modify -is-fips-enabled true -interface SSL`

3. Verify that cluster-wide FIPS-compliant mode is enabled:

   `security config show`

Enabling onboard key management

The Onboard Key Manager secures the keys created on the cluster when you enable encryption on a volume. You must enable Onboard Key Manager on each cluster on which you plan to encrypt data.

Before you begin

You must be a cluster administrator to perform this task.

About this task

If you are using NSE with an external key management (KMIP) server, you must delete the external key manager configuration before you can enable the Onboard Key Manager.

Steps

1. Start the key manager setup wizard:

   `security key-manager setup`

Example

The following command starts the key manager setup wizard on **`cluster1`**:

```
cluster1::> security key-manager setup
Welcome to the key manager setup wizard, which will lead you through
the steps to add boot information.

Enter the following commands at any time
"help" or "?" if you want to have a question clarified,
"back" if you want to change your answers to previous questions, and
"exit" if you want to quit the key manager setup wizard. Any changes
you made before typing "exit" will be applied.

Restart the key manager setup wizard with "security key-manager
setup". To accept a default or omit a question, do not enter a value.

Would you like to use onboard key-management? {yes, no} [yes]:
Enter the cluster-wide passphrase:    <32..256 UTF8 characters long text>
Reenter the cluster-wide passphrase:    <32..256 UTF8 characters long text>
```

2. Enter `yes` at the prompt to configure onboard key management.
3. Enter a passphrase between 32 and 256 characters at the passphrase prompt and reenter when prompted.

After you finish

Copy the passphrase to a secure location outside the storage system for future use.

All key management information is automatically backed up in the replicated database (RDB) for the cluster. You should also back up the information manually, for use in case of a disaster.

Updating onboard key manager passphrase

For security, you should periodically change your onboard key manager passphrase.

Steps

1. Enter

   `security key-manager update-passphrase`

2. When prompted if you'd like to continue, enter

   `Y`

3. Enter your current passphrase.
4. Enter your new passphrase. When prompted, reenter the new passphrase to confirm.

```
Warning: This command will reconfigure the cluster passphrase for
onboard key management.
Do you want to continue? {y|n}:

Enter current passphrase:    <32..256 UTF8 characters long text>

Enter new passphrase:    <32..256 UTF8 characters long text>

Reenter the new passphrase:    <32..256 UTF8 characters long text>
```

Deleting existing key manager configuration

You can manually switch from onboard key management to an external key management configuration without any loss of data or access. To do this, you have to delete the existing key manager configuration. This command is used to delete the onboard key management configuration and can be used to go back to a non-VGE or non-NSE environment for any reason.

Before you begin

- Effective cluster version must be ONTAP 9.0 or higher.
- If the cluster has encrypted volumes, the volumes must be moved to plain-text.
- Change the authentication key for all SEDs on the storage system back to the default MSID.

Steps

1. Reset the authentication key and FIPS authentication key on all drives to the default value (0x0).
2. Enter

   `security key-manager delete-key-database`

```
Warning: This command will permanently delete all keys from onboard
key management.
Do you want to continue? {y|n}:

Enter the passphrase:    <32..256 UTF8 characters long text>
```

3. Re-run the Onboard Key Manager setup wizard:

   security key-manager setup

Managing self-encrypting disks

You can perform various tasks to manage your self-encrypting disks, including replacing SSL certificates, changing authentication keys, replacing disks, returning to service or unprotect mode, and destroying data.

Replacing SSL certificates before expiration

All SSL certificates have an expiration date. You should update your SSL certificates before they expire to prevent loss of data access to self-encrypting disks.

Before you begin

- You must have obtained the replacement public and private certificates for the cluster.
- You must have obtained the replacement public certificate for the key management server
- You must have installed the appropriate new certificates on the key management server.
  For more information, see the documentation for your key management server.

Steps

1. Remove the IP address of the key management server of the SSL certificate that you want to replace:

   security key-manager delete -address *key_management_server_ipaddress*

2. Remove the cluster's client certificates:

   security certificate delete -type client -vserver *admin_svm_name* -common-name *fqdn_or_custom_common_name* -ca *certificate_authority* -subtype kmip-cert

3. Remove all installed key management server certificates:

   security certificate delete -type server-ca -vserver *admin_svm_name* -common-name *fqdn_or_custom_common_name* -ca *certificate_authority* -subtype kmip-cert

   Repeat this step for each key management server.

4. Install the SSL client certificate for the cluster:

   security certificate install -vserver *admin_svm_name* -type client -subtype kmip-cert

5. Enter the public and private certificate as prompted.

6. Install the public certificate of the key management server:

   security certificate install -vserver *admin_svm_name* -type server-ca -subtype kmip-cert -kmip-server-ip *key_management_server_ipaddress*

   If you are linking multiple key management servers to the cluster, repeat this step for each public certificate of each key management server.

7. Add a key management server:

   security key-manager add -address *key_management_server_ipaddress*

   You can add up to four key management servers.

8. Verify connectivity between the cluster and key management servers:

   security key-manager show -status

Changing the authentication key

You can change the authentication key of self-encrypting disks (SEDs) at any time by using the `security key-manager create-key` command to create a new key and the `storage encryption disk modify` command to assign the new key to SEDs. You might want to change the authentication key as part of your security protocol or when moving an aggregate to another cluster.

Steps

1. Verify that key management servers are configured and available on all nodes in the cluster:

   security key-manager show -status

2. If you want to create a new authentication key, perform one of the following actions; otherwise, to change to an existing authentication key, skip to step 3.

If you want to...	Then...
Create a new authentication key manually	**a.** Enter the following command: **security key-manager create-key -prompt-for-key true -key_tag *key_tag*** **b.** When prompted, enter the new authentication key. It must be 20 to 32 characters long.
Create a new authentication key automatically	Enter the following command: **security key-manager create-key -key_tag *key_tag***

   key_tag is the label used to associate keys with a particular cluster. If you do not specify a key tag, Data ONTAP uses the cluster name by default.

3. Make a note of the new authentication key ID.

4. Assign a new authentication key to disks:

   storage encryption disk modify -disk *disk_id* -data-key-id *authentication_key_id*

5. Verify that the authentication key was changed successfully:

   storage encryption disk show

6. Optional: Assign a new FIPS authentication key:

   storage encryption disk modify -disk *disk_id* -fips-key-id *fips_authentication_key_id*

   The FIPS authentication key ID can be different from the data authentication key ID above.

7. Verify that the FIPS authentication key was changed successfully:

   storage encryption disk show -fips

Example

The following command creates a new authentication key automatically:

```
cluster1::> security key-manager create-key

Node: node0
Creating authentication key...
Authentication key creation successful.
Key ID:
6A7A9FEC6A7A9C3C01010000000000006B2CC4A92B8F6EB1FEBB6887C78D82A7

Node: node0
Key manager restore operation initialized.
Successfully restored key information.

Node: node1
Key manager restore operation initialized.
Successfully restored key information.
```

The following command changes the authentication key on all SEDs in disk stack 1 to the new authentication key:

```
cluster1::> storage encryption disk modify -disk 1.* -data-key-id
6A7A9FEC6A7A9C3C01010000000000006B2CC4A92B8F6EB1FEBB6887C78D82A7
```

Replacing a self-encrypting disk

Replacing a self-encrypting disk (SED) is similar to replacing a regular disk, except that there are some extra steps you must take to reenable Storage Encryption after you replace the disk.

Before you begin

You must be aware of the key used by the SEDs on your storage system so that you can configure the replacement SED to use the same key.

Steps

1. Ensure that reconstruction has started by entering the following command:

   aggr status -r

   The status of the disk should display as "Reconstructing".

2. Remove the failed disk and replace it with a new SED, following the instructions in the hardware guide for your disk shelf model.

3. Assign ownership of the newly replaced SED by entering the following command:

   disk assign *disk_name*

4. Confirm that the new disk has been properly assigned by entering the following command:

   disk encrypt show

   The newly added disk is displayed in the output.

5. Encrypt the disk by entering the following command:

   disk encrypt rekey *key_id disk_name*

6. Finalize the replacement process by entering the following command:

   disk encrypt lock *disk_name*

   The newly replaced SED is ready for use, and Storage Encryption is enabled and working on this system.

Returning self-encrypting disks to service when authentication keys are no longer available

If one or more self-encrypting disks (SEDs) that are set to non-MSID authentication keys become inaccessible due to the permanent loss of the authentication keys, you can recover the SEDs (though not the data stored on them) and return them to service.

About this task

Such SEDs will be in a broken state, and regardless of the method you use to return them to service, whatever data is currently stored on them will be lost afterward. Only use this process if you are absolutely sure that the authentication keys for the SEDs are permanently lost and that there is no chance to recover them.

There are several methods for returning such SEDs to service, depending on the following criteria:

- If the SEDs have not been put into FIPS compliance mode, or if the SEDs are in FIPS compliance mode and the FIPS authentication key is still available (only the data authentication key is lost), you can use the `storage encryption disk sanitize` command.
- If the SEDs are in FIPS compliance mode and the FIPS authentication key is not available either, you can use the `storage encryption disk revert-to-original-state` command, provided that the disk has a PSID printed on its label.
  This requires accessing the physical disk to obtain the PSID from its printed label.

Step

1. Return the SEDs to service by using one of the following methods:

If the SEDS are...	Use these steps...
Not in FIPS compliance mode	**a.** Sanitize the broken disk: **`storage encryption disk sanitize -disk *disk_id*`** **b.** Set the privilege level to advanced: **`set -privilege advanced`** **c.** Fail the sanitized disk: **`storage disk fail -immediate true -disk *disk_id*`** **d.** Unfail the sanitized disk: **`storage disk unfail -spare true -disk *disk_id*`** **e.** Verify that the SED is now a spare and ready to be reused in an aggregate: **`storage disk show -disk *disk_id*`**
In FIPS compliance mode and the FIPS key is available	**a.** Sanitize the broken disk: **`storage encryption disk sanitize -disk *disk_id*`**

If the SEDS are...	Use these steps...

In FIPS compliance mode, the FIPS key is not available, and the SEDs have a PSID printed on the label	**a.** Obtain the SED's PSID from its disk label.
	b. Set the privilege level to advanced:
	`set -privilege advanced`
	c. Reset the SED to its factory configured settings:
	`storage encryption disk revert-to-original-state -disk disk_id -psid disk_physical_secure_id`
	d. Fail the sanitized disk:
	`storage disk fail -immediate true -disk disk_id`
	e. Unfail the sanitized disk:
	`storage disk unfail -spare true -disk disk_id`
	f. Verify that the SED is now a spare and ready to be reused in an aggregate:
	`storage disk show -disk disk_id`

Returning SEDs to unprotected mode

If your storage system is configured to use Storage Encryption but you decide to stop using this feature, you can do so by returning the SEDs to unprotected mode. You cannot disable Storage Encryption altogether because SEDs always encrypt data for storage. However, you can return them to unprotected mode where they no longer use secret authentication keys, and use the default MSID instead.

Steps

1. Set the privilege level to advanced:

   `set -privilege advanced`

2. Optional: If SEDs are running in FIPS-compliance mode, you must first change the FIPS authentication key back to the default MSID:

   `storage encryption disk modify -disk * -fips-key-id 0x0`

   Otherwise, skip to the next step.

3. Change the authentication key for all SEDs on the storage system back to the default MSID:

   `storage encryption disk modify -disk * -data-key-id 0x0`

4. If you expect to operate the storage system in unprotected mode permanently, you should also remove all key management servers by entering the following command for each one:

   `security key-manager delete -address key_management_server_ipaddress`

   The storage system displays two `kmip_init` errors during every bootup after you remove all key management servers. These errors are normal in this situation and you can disregard them.

5. If you expect to operate the storage system in unprotected mode permanently and you removed all key management servers in the preceding step, you should view the list of installed Storage Encryption related SSL certificates, and then remove all key management server SSL certificates:

   `security certificate show -subtype kmip-cert`

```
security certificate delete -type server-ca -vserver admin_svm_name -
common-name fqdn_or_custom_common_name -ca certificate_authority -
subtype kmip-cert
```

If you had multiple key management servers linked to the storage system, repeat the last command for each public certificate of each key management server.

Tips for creating and backing up aggregates containing data to be sanitized

If you are creating or backing up aggregates to contain data that might need to be sanitized, following some simple guidelines will reduce the time it takes to sanitize your data.

- Make sure your aggregates containing sensitive data are not larger than they need to be.
  If they are larger than needed, sanitization requires more time, disk space, and bandwidth.
- When you back up aggregates containing sensitive data, avoid backing them up to aggregates that also contain large amounts of nonsensitive data.
  This reduces the resources required to move nonsensitive data before sanitizing sensitive data.

Methods for making data on SEDs inaccessible

If you want to make data on a disk permanently inaccessible, but keep the disk's unused space available for new data, you can sanitize the disk. If you want to permanently make data inaccessible and you do not need to reuse the disk, you can destroy it.

- Disk sanitization
  When a disk is sanitized, the disk encryption key is changed to a new random value, the power-on lock state is reset to false, and the data authentication key is set to the default manufacture secure ID (MSID). This causes the data on the disk to becomes inaccessible and impossible to retrieve. Disks that are sanitized can be reused as non-zeroed spare disks.
- Disk destroy
  When a disk is destroyed, the encryption key is set to an unknown random value and the disk is irreversibly locked. This renders the disk permanently unusable and the data on it permanently inaccessible.

You can sanitize or destroy individual disks or simultaneously sanitize or destroy all disks on your node.

Sanitizing NSE disks

NetApp Storage Encryption disks can be sanitized using the `storage encryption disk sanitize` command. A different process must be followed to sanitize NetApp Storage Encryption (NSE) disks versus non- NSE disks. Non-NSE disks cannot use the `storage encryption disk sanitize` command.

Steps

1. Migrate any data that needs to be preserved to a different aggregate.
2. Destroy the aggregate on the disk to be sanitized.
3. Identify the *`disk ID`* for the disk to be sanitized:

   ```
   storage encryption disk show
   ```

4. Sanitize the disk:

- To sanitize specific disks enter the following command. Only hot spare or broken disks can be sanitized using this command.

  storage encryption disk sanitize -disk disk_id

- To sanitize all disks regardless of type enter the following command:

  storage encryption disk sanitize -disk * -force-all-state true

Disk_id specifies the disks you want to sanitized.

5. After the sanitization process is complete, return each disk to spare status:

   disk sanitize release disk_name

6. Verify that the sanitized disks are listed as spare disks:

   storage aggregate show-spare-disks

Sanitizing non-NSE disks

If you want to make data on a disk permanently inaccessible, but keep the disk's unused space available for new data, you can sanitize the disk.

About this task

A different process must be followed to sanitize non-NetApp Storage Encryption disks (NSEs) disks versus NSE disks. Non-NSE disks must be sanitized through the nodeshell using the disk sanitize command.

You cannot sanitize disks that are part of a storage pool or that are used in an aggregate. Disk sanitization disables some Data ONTAP commands. When disk sanitization is enabled, it cannot be disabled.

Steps

1. Enter the nodeshell for the node that owns the disks you want to sanitize:

   system node run -node *node_name*

2. Enable disk sanitization:

   options licensed_feature.disk_sanitization.enable on

   Once you enable disk sanitization, you cannot disable it.

3. If the disks you want to sanitize are partitioned, unpartition each disk:

   disk unpartition *disk_name*

4. Sanitize the disk:

   - To sanitize specific disks enter the following command. Only hot spare or broken disks can be sanitized using this command.

     disk sanitize start [-p pattern1|-r [-p pattern2|-r [-p pattern3|-r]]] [-c cycle_count] disk_list

   - To sanitize all disks regardless of type enter the following command:

     disk sanitize start [-p pattern1|-r [-p pattern2|-r [-p pattern3|-r]]] [-c cycle_count] * -force-all-state true

5. After the sanitization process is complete, return the disks to spare status:

   disk sanitize release disk_list

6. Exit node shell:

```
exit
```

7. Verify that the sanitized disks are returned to spare status:

   ```
   storage aggregate show-spare-disks
   ```

Destroying NSE disks

If you need to quickly destroy data on a disk and you are not concerned with saving the disk for future use, you can destroy the disk. When a disk is destroyed the data becomes permanently inaccessible and the disk can no longer be used.

Steps

1. Migrate any data that needs to be preserved to a different aggregate.
2. Destroy the aggregate.
3. Destroy the disk:
   - To destroy specific disks enter the following command.

     ```
     storage encryption disk destroy -disk disk_id
     ```

   - To simultaneously destroy all disks enter the following command:

     ```
     storage encryption disk destroy -disk * -force-all-state true
     ```

   Disk_id specifies the disks you want to destroy.

Emergency shredding of data on NSE disks

In case of a security emergency, you can instantly prevent access to data NetApp Storage Encryption (NSE) disks, even if power is not available to the storage system or the external key server.

Before you begin

- You are using an external key management server.
- The external key server must be configured so that it only operates if an easily destroyed authentication item (for example, a smart card or USB drive) is present.

See the documentation for the external key management server for details.

About this task

The steps for emergency shredding vary depending on whether power is available to the storage system and the external key server.

Step

1. Perform one of the following actions:

If...	Then...

Power is available to the storage system and you have time to gracefully take the storage system offline	**a.** If the storage system is a node in an HA pair, disable takeover.
	b. Take all aggregates offline and destroy them.
	c. Halt the storage system.
	d. Boot into maintenance mode.
	e. Sanitize or destroy the disks:
	• If you want to make the data on the disks inaccessible and still be able to reuse the disks, santize the disks:
	`storage encryption disk sanitize -disk * -force-all-states true`
	• If you want to make the data on the disks inaccessible and you do not need to save the disks, destroy the disks:
	`storage encryption disk destroy -disk * -force-all-states true`
	This leaves the storage system in a permanently disabled state with all data erased. To use the storage system again, you must set it up from the beginning.

Power is available to the storage system and you must shred the data immediately; time is critical	**a.** If the storage system is a node in an HA pair, disable takeover.
	b. Set the privilege level to advanced.
	c. Sanitize or destroy the disks:
	• If you want to make the data on the disks inaccessible and still be able to reuse the disks, santize the disks:
	`storage encryption disk sanitize -disk * -force-all-states true`
	• If you want to make the data on the disks inaccessible and you do not need to save the disks, destroy the disks:
	`storage encryption disk destroy -disk * -force-all-states true`
	The storage system panics, which is expected due to the abrupt nature of the procedure. It leaves the storage system in a permanently disabled state with all data erased. To use the storage system again, you must set it up from the beginning.

Power is available to the external key server but not to the storage system	**a.** Log in to the external key server. **b.** Destroy all keys associated with the disks containing data to protect.
Power is not available to the external key server or the storage system	Destroy the authentication item for the key server (for example, the smart card). If power to the systems is restored, the external key server cannot operate due to the missing authentication item. This prevents access to the disk encryption keys by the storage system, and therefore access to the data on the disks.

Manually enabling drive authentication on replacement drives for Storage Encryption

By default, replacement encryption drives have a Manufacture Secure ID (MSID) of 0x0. A drive with an MSID of 0x0 does not require a Key ID or passphrase authentication for data access. When you replace an encryption drive, you must manually rekey it with the current system Key ID and enable the authentication passphrase so that your data is protected.

Steps

1. Identify the disk name:

   disk show -n

2. Assign the replacement drive to the node:

   disk assign *disk_name*

3. Display the current system Key ID:

   disk encrypt show

4. Rekey the replacement drive with the current system Key ID:

   disk encrypt rekey *key_id disk_name*

5. Lock the replacement drive:

   disk encrypt lock *disk_name*

6. Verify that the replacement drive has the current system Key ID:

   disk encrypt show

How RAID is used to protect your data and data availability

Understanding how RAID protects your data and data availability can help you to administer your storage systems more effectively.

RAID-DP (double-parity) protection or RAID-TEC (triple-parity) protection is used to maintain data integrity within a RAID group. Parity disks provide redundancy for the data that is stored in the data disks. If up to two disks fail for RAID-DP or up two three disks fail for RAID-TEC, the RAID subsystem can use the parity disks to reconstruct the data in the disk that failed.

RAID protection levels for disks

ONTAP supports three levels of RAID protection for aggregates. Your level of RAID protection determines the number of parity disks available for data recovery in the event of disk failures.

With RAID protection, if there is a data disk failure in a RAID group, ONTAP can replace the failed disk with a spare disk and use parity data to reconstruct the data of the failed disk.

- RAID4
  With RAID4 protection, ONTAP can use one spare disk to replace and reconstruct the data from one failed disk within the RAID group.
- RAID-DP
  With RAID-DP protection, ONTAP can use up to two spare disks to replace and reconstruct the data from up to two simultaneously failed disks within the RAID group.
- RAID-TEC
  With RAID-TEC protection, ONTAP can use up to three spare disks to replace and reconstruct the data from up to three simultaneously failed disks within the RAID group.

Related information

NetApp Technical Report 3437: Storage Subsystem Resiliency Guide

Converting from RAID-DP to RAID-TEC

If you want the added protection of triple-parity, you can convert from RAID-DP to RAID-TEC. RAID-TEC is recommended if the size of the disks used in your aggregate is greater than 4 TiB.

Before you begin

The aggregate that is to be converted must have a minimum of six disks.

About this task

Hard disk drive (HDD) aggregates can be converted from RAID-DP to RAID-TEC. This includes HDD tiers in Flash Pool aggregates.

Steps

1. Verify that the aggregate is online and has a minimum of six disks:

   `storage aggregate show-status -aggregate` ***`aggregate_name`***

2. Convert the aggregate from RAID-DP to RAID-TEC:

   storage aggregate modify -aggregate *aggregate_name* -raidtype raid_tec

3. Verify that the aggregate RAID policy is RAID-TEC:

   storage aggregate show *aggregate_name*

Converting RAID-TEC to RAID-DP

If you reduce the size of your aggregate and no longer need triple parity, you can convert your RAID policy from RAID-TEC to RAID-DP and reduce the number of disks you need for RAID parity.

About this task

The maximum RAID group size for RAID-TEC is larger than the maximum RAID group size for RAID-DP. If the largest RAID-TEC group size is not within the RAID-DP limits, you cannot convert to RAID-DP.

Steps

1. Verify that the aggregate is online and has a minium of six disks:

   storage aggregate show-status -aggregate *aggregate_name*

2. Convert the aggregate from RAID-TEC to RAID-DP:

   storage aggregate modify -aggregate *aggregate_name* -raidtype raid_dp

3. Verify that the aggregate RAID policy is RAID-DP:

   storage aggregate show *aggregate_name*

Considerations for sizing RAID groups

Configuring an optimum RAID group size requires a trade-off of factors. You must decide which factors—speed of RAID rebuild, assurance against risk of data loss due to drive failure, optimizing I/O performance, and maximizing data storage space—are most important for the aggregate that you are configuring.

When you create larger RAID groups, you maximize the space available for data storage for the same amount of storage used for parity (also known as the “parity tax”). On the other hand, when a disk fails in a larger RAID group, reconstruction time is increased, impacting performance for a longer period of time. In addition, having more disks in a RAID group increases the probability of a multiple disk failure within the same RAID group.

HDD or array LUN RAID groups

You should follow these guidelines when sizing your RAID groups composed of HDDs or array LUNs:

- All RAID groups in an aggregate should have a similar number of disks.
  The RAID groups do not have to be exactly the same size, but you should avoid having any RAID group that is less than one half the size of other RAID groups in the same aggregate when possible.
- The recommended range of RAID group size is between 12 and 20.
  The reliability of performance disks can support a RAID group size of up to 28, if needed.
- If you can satisfy the first two guidelines with multiple RAID group sizes, you should choose the larger size.

SSD RAID groups in Flash Pool aggregates

The SSD RAID group size can be different from the RAID group size for the HDD RAID groups in a Flash Pool aggregate. Usually, you should ensure that you have only one SSD RAID group for a Flash Pool aggregate, to minimize the number of SSDs required for parity.

SSD RAID groups in SSD aggregates

You should follow these guidelines when sizing your RAID groups composed of SSDs:

- All RAID groups in an aggregate should have a similar number of drives.
  The RAID groups do not have to be exactly the same size, but you should avoid having any RAID group that is less than one half the size of other RAID groups in the same aggregate when possible.
- For RAID-DP, the recommended range of RAID group size is between 20 and 28.

Customizing the size of your RAID groups

You can customize the size of your RAID groups to ensure that your RAID group sizes are appropriate for the amount of storage you plan to include for an aggregate.

About this task

For standard aggregates, you change the size of RAID groups on a per-aggregate basis. For Flash Pool aggregates, you can change the RAID group size for the SSD RAID groups and the HDD RAID groups independently.

The following list outlines some facts about changing the RAID group size:

- By default, if the number of disks or array LUNs in the most recently created RAID group is less than the new RAID group size, disks or array LUNs will be added to the most recently created RAID group until it reaches the new size.
- All other existing RAID groups in that aggregate remain the same size, unless you explicitly add disks to them.
- You can never cause a RAID group to become larger than the current maximum RAID group size for the aggregate.
- You cannot decrease the size of already created RAID groups.
- The new size applies to all RAID groups in that aggregate (or, in the case of a Flash Pool aggregate, all RAID groups for the affected RAID group type—SSD or HDD).

Step

1. Use the applicable command:

If you want to...	Enter the following command...
Change the maximum RAID group size for the SSD RAID groups of a Flash Pool aggregate	**storage aggregate modify -aggregate *aggr_name* -cache-raid-group-size *size***
Change the maximum size of any other RAID groups	**storage aggregate modify -aggregate *aggr_name* -maxraidsize *size***

Examples

The following command changes the maximum RAID group size of the aggregate n1_a4 to 20 disks or array LUNs:

```
storage aggregate modify -aggregate n1_a4 -maxraidsize 20
```

The following command changes the maximum RAID group size of the SSD cache RAID groups of the Flash Pool aggregate n1_cache_a2 to 24:

```
storage aggregate modify -aggregate n1_cache_a2 -cache-raid-group-size
24
```

Related concepts

Considerations for sizing RAID groups on page 74

How hot spare disks work

A hot spare disk is a disk that is assigned to a storage system and is ready for use, but is not in use by a RAID group and does not hold any data.

If a disk failure occurs within a RAID group, the hot spare disk is automatically assigned to the RAID group to replace the failed disks. The data of the failed disk is reconstructed on the hot spare replacement disk in the background from the RAID parity disk. The reconstruction activity is logged in the `/etc/message` file and an AutoSupport message is sent.

If the available hot spare disk is not the same size as the failed disk, a disk of the next larger size is chosen and then downsized to match the size of the disk that it is replacing.

How low spare warnings can help you manage your spare disks

By default, warnings are issued to the console and logs if you have fewer than one hot spare drive that matches the attributes of each drive in your storage system. You can change the threshold value for these warning messages to ensure that your system adheres to best practices.

You should set the `min_spare_count` RAID option to **`2`** to ensure that you always have the minimum recommended number of spare disks.

Setting the `min_spare_count` RAID option to **`0`** disables low spare warnings. You might want to do this if you do not have enough drives to provide hot spares (for example, if your storage system does not support external disk shelves). You can disable the warnings only if the following requirements are met:

- Your system has 16 or fewer drives.
- You have no RAID groups that use RAID4.

  Note: You cannot create aggregates that use RAID4 protection while the `raid.min_spare_count` option is set to **`0`**. If either of these requirements is no longer met after this option has been set to **`0`**, the option is automatically set back to **`1`**.

Where to find additional information

After you have setup the disks and aggregates for your storage system, you can perform additional tasks such as setting up volumes to contain your data.

- *ONTAP 9 Commands: Manual Page Reference*
  Describe commands for creating and managing aggregates in reference format.
- *ONTAP 9 Logical Storage Management Guide*
  Describes how to manage logical storage resources in clusters, including FlexVol volumes, FlexClone volumes, FlexCache volumes, files, and LUNs.
- *ONTAP 9 SAN Administration Guide*
  Describes how to configure and manage iSCSI, FCoE, and FC protocols including configuration of LUNs, igroups and targets.
- *NetApp Technical Report 3483: Thin Provisioning in a NetApp SAN or IP SAN Enterprise Environment*
  Describes thin provisioning in a SAN or IP SAN enterprise environment.

.

Copyright information

Copyright © 1994–2016 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.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademark information

Active IQ, AltaVault, Arch Design, ASUP, AutoSupport, Campaign Express, Clustered Data ONTAP, Customer Fitness, Data ONTAP, DataMotion, Fitness, Flash Accel, Flash Cache, Flash Pool, FlexArray, FlexCache, FlexClone, FlexGroup, FlexPod, FlexScale, FlexShare, FlexVol, FPolicy, Fueled by SolidFire, GetSuccessful, Helix Design, LockVault, Manage ONTAP, MetroCluster, MultiStore, NetApp, NetApp Insight, OnCommand, ONTAP, ONTAPI, RAID DP, RAID-TEC, SANscreen, SANshare, SANtricity, SecureShare, Simplicity, Simulate ONTAP, Snap Creator, SnapCenter, SnapCopy, SnapDrive, SnapIntegrator, SnapLock, SnapManager, SnapMirror, SnapMover, SnapProtect, SnapRestore, Snapshot, SnapValidator, SnapVault, SolidFire, SolidFire Helix, StorageGRID, SyncMirror, Tech OnTap, Unbound Cloud, and WAFL and other names are trademarks or registered trademarks of NetApp, Inc., in the United States, and/or other countries. All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such. A current list of NetApp trademarks is available on the web.

http://www.netapp.com/us/legal/netapptmlist.aspx

How to send comments about documentation and receive update notifications

You can help us to improve the quality of our documentation by sending us your feedback. You can receive automatic notification when production-level (GA/FCS) documentation is initially released or important changes are made to existing production-level documents.

If you have suggestions for improving this document, send us your comments by email.

doccomments@netapp.com

To help us direct your comments to the correct division, include in the subject line the product name, version, and operating system.

If you want to be notified automatically when production-level documentation is released or important changes are made to existing production-level documents, follow Twitter account @NetAppDoc.

You can also contact us in the following ways:

- NetApp, Inc., 495 East Java Drive, Sunnyvale, CA 94089 U.S.
- Telephone: +1 (408) 822-6000
- Fax: +1 (408) 822-4501
- Support telephone: +1 (888) 463-8277

Index

A

C

D

E

F

H

I

K

L

M

N

O

P

R

T

U

V

W