



metrocluster commands

ONTAP 9.3 commands

NetApp
September 27, 2022

Table of Contents

- metrocluster commands 1
 - metrocluster configure 1
 - metrocluster heal 5
 - metrocluster modify 5
 - metrocluster show 6
 - metrocluster switchback 9
 - metrocluster switchover 10
 - metrocluster check commands 11
 - metrocluster config-replication commands 37
 - metrocluster configuration-settings commands 39
 - metrocluster interconnect commands 68
 - metrocluster node commands 78
 - metrocluster operation commands 83
 - metrocluster vservers commands 86

metrocluster commands

metrocluster configure

Configure MetroCluster and start DR mirroring for the node and its DR group

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

Description

The `metrocluster configure` command creates a MetroCluster configuration on either all the nodes in both MetroCluster clusters or solely on nodes in a DR group. The command configures a HA partner, DR partner, and a DR auxiliary partner for the nodes and also starts NVRAM mirroring between the configured DR partner nodes.

In MetroCluster, a DR group is a group of four nodes, two in each of the MetroCluster clusters:

- In the local cluster, a node and its HA partner,
- In the peer cluster, a node and its HA partner. These nodes are DR partners to the nodes in the local cluster.

There can be several DR groups in the MetroCluster configuration. MetroCluster provides synchronous DR protection to all data sets belonging to nodes within a properly configured DR group.

Without the `-node` parameter, the `metrocluster configure` command configures all the DR groups in both the MetroCluster clusters.

With the `-node`mynode` parameter, the command configures both the *mynode* node and its HA partner node from the local cluster, and its DR partner and DR auxiliary partner from the peer cluster.

Before running the `metrocluster configure` command, the aggregates and Vservers on each node must be prepared for the MetroCluster configuration. Each node should have:

- At least one non-root, mirrored aggregate of size greater than 10GB. This non-root aggregate should not have any volumes in it.
- No other non-root aggregates. Any other non-root, unmirrored aggregates and volumes should be deleted.
- No Vservers other than Vservers of type "node" or "admin." Any Vservers that are not of type "node" or "admin" should be deleted.
- A mirrored and healthy root aggregate.

After the command is successful all nodes in the local and remote clusters will have HA, DR, and DR auxiliary partners and NVRAM mirroring between the DR partners will be turned on. The same conditions apply for before running the `metrocluster configure-node`mynode` command, except that only one DR group is configured.

Parameters

[`-node-name {<nodename>|local}`] - Node to Configure

This optional parameter specifies the name of a single node in the local cluster. The command creates MetroCluster configuration on the local node specified by this parameter and the three other nodes

belonging to the same DR group.

`[-refresh {true|false}] - Refresh Configuration`

This optional parameter specifies if the node partner configuration steps should be done again. Not specifying this parameter will cause the MetroCluster configuration to continue using the current node partner information.

`[-allow-with-one-aggregate {true|false}] - Override the Two Data Aggregates Requirement`

This optional parameter specifies if MetroCluster configuration should be allowed with only one data aggregate in each cluster. This option has no effect if two or more aggregates are present.

Examples

The following example shows the creation of the MetroCluster configuration for a single DR group:

```

clusA::> metrocluster show
  Cluster                               Configuration State      Mode
  -----                               -
-----
  Local: clusA                          not-configured           -
  Remote: clusB                          not-configured           -
clusA::> metrocluster node show
DR                               Configuration  DR
Group Cluster Node              State          Mirroring Mode
-----
-      clusA   clusA-01          ready to configure      -      -
                               clusA-02          ready to configure      -      -
                               clusA-03          ready to configure      -      -
                               clusA-04          ready to configure      -      -

4 entries were displayed.
clusA::> metrocluster configure -node clusA-01
[Job 45] Job succeeded: Configure is successful
clusA::> metrocluster show
  Cluster                               Configuration State      Mode
  -----                               -
-----
  Local: clusA                          partially-configured     normal
  Remote: clusB                          partially-configured     normal
clusA::> metrocluster node show
DR                               Configuration  DR
Group Cluster Node              State          Mirroring Mode
-----
-----
-      clusA   clusA-03          ready to configure      -      -
                               clusA-04          ready to configure      -      -

1      clusA   clusA-01          configured             enabled   normal
                               clusA-02          configured             enabled   normal
      clusB   clusB-01          configured             enabled   normal
                               clusB-02          configured             enabled   normal

6 entries were displayed.

```

The following example shows the creation of the MetroCluster configuration for all DR groups:

```
clusA::> metrocluster show
```

Cluster	Configuration State	Mode
-----	-----	
Local: clusA	not-configured	-
Remote: clusB	not-configured	-

```
clusA::> metrocluster node show
```

DR	Configuration	DR
Group Cluster Node	State	Mirroring Mode
-----	-----	-----
- clusA clusA-01	ready to configure	- -
	clusA-02	ready to configure
		- -
	clusA-03	ready to configure
		- -
	clusA-04	ready to configure
		- -

4 entries were displayed.

```
clusA::> metrocluster configure
```

```
[Job 45] Job succeeded: Configure is successful
```

```
clusA::> metrocluster show
```

Cluster	Configuration State	Mode
-----	-----	
Local: clusA	configured	normal
Remote: clusB	configured	normal

```
clusA::> metrocluster node show
```

DR	Configuration	DR
Group Cluster Node	State	Mirroring Mode
-----	-----	-----
1 clusA clusA-01	configured	enabled normal
	clusA-02	configured enabled normal
	clusB clusB-01	configured enabled normal
	clusB-02	configured enabled normal
2 clusA clusA-03	configured	enabled normal
	clusA-04	configured enabled normal
	clusB clusB-03	configured enabled normal
	clusB-04	configured enabled normal

8 entries were displayed.

metrocluster heal

Heal DR data aggregates and DR root aggregates

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

Description

The `metrocluster heal` command heals DR data aggregates and DR root aggregates in preparation for a DR switchback. You must issue this command twice to complete the two phases of the healing process: first to heal the aggregates by resynchronizing the mirrored plexes and then to heal the root aggregates by switching them back to the disaster site. The DR partner nodes must be powered off and remote disk shelves must be powered on before running this command.

Parameters

-phase {aggregates|root-aggregates} - MetroCluster Healing Phase

This parameter specifies the healing phase. The first phase, *aggregates*, heals aggregates by resynchronizing mirrored plexes. The second phase, *root-aggregates*, heals the root aggregates of partner nodes. Healing root aggregates switches them back to the disaster site, allowing the site to boot up.

[-override-vetoes <true>] - Override All Soft Vetoes

This optional parameter overrides almost all heal operation soft vetoes. If this optional parameter is set to true, the system overrides subsystem soft vetoes that might prevent the heal operation. Hard vetoes cannot be overridden and can still prevent the switchback operation.

Examples

The following example performs the healing of both the aggregates and root aggregates:

```
cluster1::> metrocluster heal -phase aggregates
      [Job 136] Job succeeded: Heal Aggregates is successful
cluster1::> metrocluster heal -phase root-aggregates
      [Job 137] Job succeeded: Heal Root Aggregates is successful
```

metrocluster modify

Modify MetroCluster configuration options

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

Description

The `metrocluster modify` command modifies MetroCluster parameters for nodes in the MetroCluster configuration.

Parameters

{ **-auto-switchover-failure-domain** <MetroCluster AUSO Failure Domain> - Cluster Level AUSO Option

This parameter specifies configuration of the automatic switchover. Supported values are as follows:

- `auso-on-cluster-disaster` - triggers an unplanned switchover if all nodes in a DR cluster are down.
- `auso-on-dr-group-disaster` - triggers an unplanned switchover if both nodes of a DR group are down.
- `auso-disabled` - automatic switchover is disabled.

The default value is `auso-on-cluster-disaster`. It is set to default value if `metrocluster configure` command is issued.

The auto-switchover failure domain is set to `auso-disabled` if the `metrocluster unconfigure` command is issued.

This setting only affects the local cluster where the command is run.

| **-node-name** {<nodename>|local} - Node to Change the Option On }

This parameter is used to specify the node in the cluster for which the parameter needs to be modified.

[**-automatic-switchover-onfailure** <true>] - Node Level AUSO Option }

This parameter is used to enable automatic switchover on failure on a node when it is disabled because of internal errors. Possible value for this field is `true`. This option is available in `diag` mode and should be used only when warning ems message comes. All the nodes in MCC configuration must have this option enabled (the default state) to enable automatic switchover on failure.

Examples

The following example shows the output of Metrocluster modification done on a node:

```
clusA::*> metrocluster modify -node-name clusA-01 -node-object-limit on
[Job 168] Job succeeded: Modify is successful
clusA::*> metrocluster modify -node-name clusA-01 -automatic-switchover
-onfailure false
[Job 308] Job succeeded: Modify is successful
clusA::*> metrocluster modify -auto-switchover-failure-domain auso-on-
cluster-disaster
[Job 308] Job succeeded: Modify is successful
```

metrocluster show

Display MetroCluster configuration information

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

Description

The `metrocluster show` command displays configuration information for the pair of clusters configured in MetroCluster.

This command displays the following details about the local cluster and the DR partner cluster:

- Configuration State: This field specifies the configuration state of the cluster.
- Mode: This field specifies the operational mode of the cluster.
- AUSO Failure Domain: This field specifies the AUSO failure domain of the cluster.

Parameters

`[-periodic-check-status]`

If this option is used the MetroCluster periodic check status is displayed.

Examples

The following example shows the output of the command before MetroCluster configuration is done:

```
clusA::> metrocluster show
Cluster                               Entry Name                               State
-----                               -
-----
Local: clusA
      Configuration State  not-configured
      Mode                 -
      AUSO Failure Domain  -
Remote: clusB
      Configuration State  not-configured
      Mode                 -
      AUSO Failure Domain  -
```

The following example shows the output of the command after MetroCluster configuration is done only for some DR groups:

```

clusA::> metrocluster show
Cluster                               Entry Name                               State
-----                               -
-----
Local: clusA
      Configuration State  partially-configured
              Mode        -
      AUSO Failure Domain  -
Remote: clusB
      Configuration State  partially-configured
              Mode        -
      AUSO Failure Domain  -

```

The following example shows the output of the command after MetroCluster configuration is done:

```

clusA::> metrocluster show
Cluster                               Entry Name                               State
-----                               -
-----
Local: clusA
      Configuration State  configured
              Mode        normal
      AUSO Failure Domain  auso-on-cluster-disaster
Remote: clusB
      Configuration State  configured
              Mode        normal
      AUSO Failure Domain  auso-on-cluster-disaster

```

The following example shows the output of the command in switchover mode:

```

clusA::> metrocluster show
Cluster                               Entry Name                               State
-----                               -
-----
Local: clusA
      Configuration State  configured
              Mode        switchover
      AUSO Failure Domain  auso-on-cluster-disaster
Remote: clusB
      Configuration State  not-reachable
              Mode        -
      AUSO Failure Domain  not-reachable

```

The following example shows the output of the command when `-periodic-check-status` option is used:

```
clusA::> metrocluster show -periodic-check-status
Cluster                                Periodic Check Enabled
-----                                -
Local: clusA                           true
Remote: clusB                           true
```

metrocluster switchback

Switch back storage and client access

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

Description

The `metrocluster switchback` command initiates the switchback of storage and client access from nodes in the DR site to their home nodes. The home nodes and storage shelves must be powered on and reachable by nodes in the DR site. The `xref:{relative_path}metrocluster-heal.html[metrocluster heal]-phase``_aggregates_``` and `xref:{relative_path}metrocluster-heal.html[metrocluster heal]-phase``_root-aggregates_``` commands must have successfully completed before running the `metrocluster switchback` command.

Parameters

`[-f, -override-vetoes <true>] - Override All Soft Vetoes`

This optional parameter overrides all switchback operation soft vetoes. If this optional parameter is used, the system overrides subsystem soft vetoes that might prevent the switchback operation. Hard vetoes cannot be overridden and can still prevent the switchover operation.

`[-simulate <true>] - Simulate Switchback`

If this optional parameter is used, the system runs a simulation of the switchback operation to make sure all the prerequisites for the operation are met. This parameter cannot be used with switchback operations performed for switching back left-behind aggregates or for retrying a partially successful switchback.

Examples

The following is an example of how to start the switchback operation.

```
clusA::> metrocluster switchback
```

Related Links

- [metrocluster heal](#)

metrocluster switchover

Switch over storage and client access

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

Description

The `metrocluster switchover` command initiates the switchover of storage and client access from the source cluster to the disaster recovery (DR) site. This command is to be used after a disaster that renders all the nodes in the source cluster unreachable and powered off. It can also be used for negotiated switchover when the outage of the source cluster is anticipated as in cases such as disaster recovery testing or a site going offline for maintenance. If a switchover operation previously failed on certain nodes on the DR site then issuing the command retries the operation on all of those nodes.

Parameters

{ `[-simulate <true>]` - Simulate Negotiated Switchover

If this optional parameter is used, the system runs a simulation of the negotiated switchover operation to make sure all the prerequisites for the operation are met. This parameter cannot be used with switchover with the `-forced-on-disaster` parameter.

{ `[-forced-on-disaster <true>]` - Force Switchover on Disaster

This optional parameter forces a switchover on disaster. This parameter should be used if all the nodes on the disaster stricken site are powered off and unreachable. In the absence of this parameter, the command attempts to perform a negotiated switchover operation.

{ `[-force-nvfail-all <true>]` - Sets in-nvfailed-state on All Volumes

If this parameter is used, the switchover command will set the `in-nvfailed-state` parameter to true for all volumes being switched over and will set the `-dr-force-nvfail` parameter to true for any volumes that do not already have it enabled. This parameter has no effect when performing a negotiated switchover.

{ `[-retry-failed-nodes <Node name>,...]` - Nodes to Switchover

This optional parameter takes the list of nodes that previously failed the switchover operation and it retries the switchover operation on each of the nodes. This parameter is applicable only for a switchover with the `-forced-on-disaster` parameter.

{ `[-override-vetoes <true>]` - Override All Soft Vetoes

This optional parameter overrides all switchover operation soft vetoes. If this parameter is used, the system overrides all subsystem soft vetoes that might prevent the switchover operation. Hard vetoes cannot be overridden and can still prevent the switchover operation.

Examples

When a disaster strikes one site, the `metrocluster switchover` command is issued on the disaster recovery site as follows:

```
cluster1::> metrocluster switchover -forced-on-disaster true

Warning: MetroCluster switchover is a Disaster Recovery operation that
could
    cause some data loss. The cluster on the other site must either
be
    prevented from serving data or be simply powered off (nodes and
disk
    shelves)
    The following nodes ( cluster1-01 cluster1-02 ) will participate
in
    the switchover operation
Do you want to continue? {y|n}: y
Queued job. Use 'metrocluster operation show' to check status of the DR
operation.
cluster1::> metrocluster operation show
  Operation: switchover
    State: successful
  Start time: 10/3/2013 22:11:47
    End time: 10/3/2013 22:11:53
  Errors: -
```

metrocluster check commands

metrocluster check disable-periodic-check

Disable Periodic Check

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

Description

The `metrocluster check disable-periodic-check` command disables the periodic checking of the MetroCluster configuration.

After this command is run, the MetroCluster Check job will be prevented from periodically checking the configuration for errors.

Parameters

Examples

```
clusA::> metrocluster check disable-periodic-check
```

metrocluster check enable-periodic-check

Enable Periodic Check

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

Description

The `metrocluster check enable-periodic-check` command enables the periodic checking of the MetroCluster configuration.

After this command is run, the MetroCluster Check job will be able to run in the background and periodically check the configuration for errors.

Parameters

Examples

```
clusA::> metrocluster check enable-periodic-check
```

metrocluster check run

Check the MetroCluster setup

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

Description

The `metrocluster check run` command performs checks on the MetroCluster configuration and reports configuration errors if any.

To run this command, at least one DR group needs to be configured. The command checks the following parts of the configuration:

Node Configuration:

- `node-reachable`: This check verifies that the node is reachable.
- `metrocluster-ready`: This check verifies that the node is ready for MetroCluster configuration.
- `local-ha-partner`: This check verifies that the HA partner node is in the same cluster.
- `ha-mirroring-on`: This check verifies that HA mirroring for the node is configured.
- `symmetric-ha-relationship`: This check verifies that the relationship between the node and its HA partner is symmetric.
- `remote-dr-partner`: This check verifies that the DR partner node is in the remote cluster.

- `dr-mirroring-on`: This check verifies that DR mirroring for the node is configured.
- `symmetric-dr-relationship`: This check verifies that the relationship between the node and its DR partner is symmetric.
- `remote-dr-auxiliary-partner`: This check verifies that the DR auxiliary partner node is in the remote cluster.
- `symmetric-dr-auxiliary-relationship`: This check verifies that the relationship between the node and its DR auxiliary partner is symmetric.
- `storage-failover-enabled`: This check verifies that storage failover is enabled.
- `has-intercluster-lif`: This check verifies that the node has an intercluster LIF.
- `node-object-limit`: This check verifies that the node object limit option for the node is turned on.

Aggregate Configuration:

- `mirroring-status`: This check verifies that the aggregate is mirrored.
- `disk-pool-allocation`: This check verifies that the disks belonging to this aggregate have been correctly allocated to the right pools.

At the end of the check the command displays a summary of the results. This summary output can be viewed again by running `metrocluster check show`. If any of the rows in this output show any warnings more details can be viewed by running the `metrocluster check show` command for that component.

Parameters

`[-skip-dr-simulation {true|false}]` - Skip the DR Readiness Checks

If this optional parameter is set to true, the switchover and switchback simulations are not run.

Examples

The following example shows the execution of the command when there are no warnings:

```
clusA::> metrocluster check run

          Last Checked On: 4/9/2014 20:11:46
Component          Result
-----
nodes              ok
clusters          ok
lifs              ok
config-replication ok
aggregates        ok
5 entries were displayed.
Command completed. Use the "metrocluster check show -instance" command or
sub-commands in "metrocluster check" directory for detailed results.
```

The following example shows the execution of the command when there are some warnings:

```

clusA::> metrocluster check run
Last Checked On: 4/9/2014 20:11:46
Component          Result
-----
nodes              warning
clusters           ok
lifs               ok
config-replication ok
aggregates         ok
5 entries were displayed.
Command completed. Use the "metrocluster check show -instance" command or
sub-commands in "metrocluster check" directory for detailed results.

```

Related Links

- [metrocluster check show](#)

metrocluster check show

Show the results of the last instance of MetroCluster check

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

Description

The `metrocluster check show` command displays the results of [metrocluster check run](#) command.

This command displays the high-level verification results for each of the components. If there any errors for a component, running the show command for that component (for example [metrocluster check node show](#) or [metrocluster check aggregate show](#)) will display more information about the warning.



Please note that this command does not run the checks but only displays the results of checks. To look at the latest results, run the [metrocluster check run](#) command and then run this command.

Parameters

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

If you specify the `-fields <fieldname>, ...` parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

| [-instance] }

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

[-timestamp <MM/DD/YYYY HH:MM:SS>] - Time of Check

This is the time at which the [metrocluster check run](#) command was last run in this cluster and these results were produced. If this parameter is specified, only rows with this timestamp will be displayed.

[-component <MetroCluster Check Components>] - Name of the Component

This is the name of the component. If this parameter is specified, only rows with this component will be displayed.

[-result {ok|warning|not-run|not-applicable}] - Result of the Check

This is the result of the check for the component. If this parameter is specified, only rows with this result will be displayed.

[-additional-info <text>] - Additional Information/Recovery Steps

This is the additional info for the verification for this component. This field will have detailed information about the warning and recovery steps. If this parameter is specified, only rows with this additional info will be displayed.

Examples

The following example shows the execution of the command when there are no warnings:

```
clusA::> metrocluster check show
cked On: 4/9/2014 20:11:46
t          Result
-----
nodes          ok
clusters       ok
lifs           ok
config-replication ok
aggregates     ok
s were displayed.
```

The following example shows the execution of the command when there are some warnings:

```
clusA::> metrocluster check show
cked On: 4/9/2014 20:11:46
t          Result
-----
nodes          warning
clusters       ok
lifs           ok
config-replication ok
aggregates     ok
s were displayed.
```

The following example shows the execution of the command with -instance option:

```

clusA::> metrocluster check show -instance
Time of Check: 4/9/2014 20:12:36
      Name of the Component: nodes
      Result of the Check: warning
    Additional Information/Recovery Steps:
Time of Check: 4/9/2014 20:12:36
      Name of the Component: cluster
      Result of the Check: ok
    Additional Information/Recovery Steps:
Time of Check: 4/9/2014 20:12:36
      Name of the Component: lifs
      Result of the Check: ok
    Additional Information/Recovery Steps:
Time of Check: 4/9/2014 20:12:36
      Name of the Component: config-replication
      Result of the Check: ok
    Additional Information/Recovery Steps:
Time of Check: 4/9/2014 20:12:36
      Name of the Component: aggregates
      Result of the Check: warning
    Additional Information/Recovery Steps:
    5 entries were displayed.

```

Related Links

- [metrocluster check run](#)
- [metrocluster check node show](#)
- [metrocluster check aggregate show](#)

metrocluster check aggregate show

Show results of MetroCluster check for aggregates

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

Description

The `metrocluster check aggregate show` command displays the results of aggregate checks performed by the [metrocluster check run](#) command.

The command verifies the following aspects of the configuration of all aggregates in MetroCluster:

- `mirroring-status`: This check verifies that the aggregate is mirrored.
- `disk-pool-allocation`: This check verifies that the disks belonging to this aggregate have been correctly allocated to the right pools.

Additional information about the warnings (if any) and recovery steps can be viewed by running the command

with the `-instance` option.

Parameters

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

If you specify the `-fields <fieldname>`, ... parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

| [-instance] }

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

[-node <Node name>] - Node Name

This is the name of the node for which the check was run. If this parameter is specified, only rows with this node will be displayed.

[-aggregate <aggregate name>] - Name of the Aggregate

This is the name of the aggregate for which the check was run. If this parameter is specified, only rows with this aggregate will be displayed.

[-check <MetroCluster Aggregate Check>] - Type of Check

This is the type of the check performed. If this parameter is specified, only rows with this check will be displayed.

[-cluster <Cluster name>] - Name of Cluster

This is the name of the cluster the node belongs to. If this parameter is specified, only rows with this cluster will be displayed.

[-result {ok|warning|not-run|not-applicable}] - Result of the Check

This is the result of the check. If this parameter is specified, only rows with this result will be displayed.

[-additional-info <text>,...] - Additional Information/Recovery Steps

This is additional information about the check. This field has more information and recovery steps for the warning. If this parameter is specified, only rows with this additional info will be displayed.

Examples

The following example shows the execution of the command in a MetroCluster configuration with two nodes per cluster:

```
clusA::> metrocluster check aggregate show
```

```
Last Checked On: 4/9/2014 20:11:46
```

Node	Aggregate	Check	Result
clusA-01	a1_required_data_aggr	mirroring-status	ok
		disk-pool-allocation	ok
	aggr0_a1	mirroring-status	ok
		disk-pool-allocation	ok
clusA-02	a2_required_data_aggr	mirroring-status	ok
		disk-pool-allocation	ok
	aggr0_a2	mirroring-status	ok
		disk-pool-allocation	ok
clusB-01	b1_required_data_aggr	mirroring-status	ok
		disk-pool-allocation	ok
	aggr0_b1	mirroring-status	ok
		disk-pool-allocation	ok
clusB-02	aggr0_b2	mirroring-status	ok
		disk-pool-allocation	ok
	b2_required_data_aggr	mirroring-status	ok
		disk-pool-allocation	ok

```
16 entries were displayed.
```

The following example shows the execution of the command with `-instance` option:

```
clusA::> metrocluster check aggregate show -instance
```

```
Node Name: clusA-01
```

```
    Name of the Aggregate: a1_required_data_aggr_1
```

```
        Type of Check: mirroring-status
```

```
        Name of Cluster: clusA
```

```
        Result of the Check: ok
```

```
Additional Information/Recovery Steps: -
```

```
Node Name: clusA-01
```

```
    Name of the Aggregate: a1_required_data_aggr_1
```

```
        Type of Check: disk-pool-allocation
```

```

        Name of Cluster: clusA
        Result of the Check: ok
Additional Information/Recovery Steps: -
Node Name: clusA-01
        Name of the Aggregate: a1_required_data_aggr_2
        Type of Check: mirroring-status
        Name of Cluster: clusA
        Result of the Check: ok
Additional Information/Recovery Steps: -
Node Name: clusA-01
        Name of the Aggregate: a1_required_data_aggr_2
        Type of Check: disk-pool-allocation
        Name of Cluster: clusA
        Result of the Check: ok
Additional Information/Recovery Steps: -
Node Name: clusA-01
        Name of the Aggregate: aggr0_a1
        Type of Check: mirroring-status
        Name of Cluster: clusA
        Result of the Check: warning
Additional Information/Recovery Steps: Root aggregate "aggr0_a1" is un-
mirrored. Root aggregates should be mirrored in a MetroCluster
configuration.
Node Name: clusA-01
        Name of the Aggregate: aggr0_a1
        Type of Check: disk-pool-allocation
        Name of Cluster: clusA
        Result of the Check: ok
Additional Information/Recovery Steps: -
Node Name: clusB-01
        Name of the Aggregate: aggr0_b1
        Type of Check: mirroring-status
        Name of Cluster: clusB
        Result of the Check: ok
Additional Information/Recovery Steps: -
Node Name: clusB-01
        Name of the Aggregate: aggr0_b1
        Type of Check: disk-pool-allocation
        Name of Cluster: clusB
        Result of the Check: ok
Additional Information/Recovery Steps: -
Node Name: clusB-01
        Name of the Aggregate: b1_required_data_aggr_1
        Type of Check: mirroring-status
        Name of Cluster: clusB
        Result of the Check: ok

```

```

Additional Information/Recovery Steps: -
Node Name: clusB-01
    Name of the Aggregate: b1_required_data_aggr_1
    Type of Check: disk-pool-allocation
    Name of Cluster: clusB
    Result of the Check: ok
Additional Information/Recovery Steps: -
Node Name: clusB-01
    Name of the Aggregate: b1_required_data_aggr_2
    Type of Check: mirroring-status
    Name of Cluster: clusB
    Result of the Check: ok
Additional Information/Recovery Steps: -
Node Name: clusB-01
    Name of the Aggregate: b1_required_data_aggr_2
    Type of Check: disk-pool-allocation
    Name of Cluster: clusB
    Result of the Check: ok
Additional Information/Recovery Steps: -
12 entries were displayed.

```

Related Links

- [metrocluster check run](#)

metrocluster check cluster show

Show results of MetroCluster check for the cluster components

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

Description

The `metrocluster check cluster show` command displays the results of cluster checks performed by the [metrocluster check run](#) command.

The command displays the results of the following cluster configuration checks:

- `negotiated-switchover-ready`: This check verifies that the cluster is ready for a negotiated switchover operation.
- `switchback-ready`: This check verifies that the cluster is ready for a switchback operation.
- `job-schedules`: This check verifies that the job schedules between the local and remote clusters are consistent.
- `licenses`: This check verifies that the licenses between the local and remote clusters are consistent.
- `periodic-check-enabled`: This check verifies that the periodic MetroCluster Check Job is enabled.
- `onboard-key-management`: This check verifies that the Onboard Key Management hierarchies are synched.

Additional information about the warnings (if any) and recovery steps can be viewed by running the command with the `-instance` parameter.

Parameters

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

If you specify the `-fields <fieldname>, ...` parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

| [-instance] }

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

[-check {negotiated-switchover-ready|switchback-ready|job-schedules|licenses|periodic-check-enabled|onboard-key-management}] - Type of Check

This is the type of the check performed. If this parameter is specified, only rows with this check will be displayed.

[-cluster <Cluster name>] - Cluster Name

This is the name of the cluster the check results apply to. If this parameter is specified, only rows matching the specified cluster will be displayed.

[-result {ok|warning|not-run|not-applicable}] - Result of the Check

This is the result of the check. If this parameter is specified, only rows with this result will be displayed.

[-additional-info <text>] - Additional Information/Recovery Steps

This is additional information about the check. This field has more information and recovery steps for the warning. If this parameter is specified, only rows with this additional info will be displayed.

Examples

The following example shows the execution of the command in a MetroCluster configuration:

```
clusA::> metrocluster check cluster show
```

```
Last Checked On: 11/29/2018 17:15:00
```

Cluster	Check	Result
clusA	negotiated-switchover-ready	not-applicable
	switchback-ready	not-applicable
	job-schedules	ok
	licenses	ok
	periodic-check-enabled	ok
	onboard-key-management	ok
clusB	negotiated-switchover-ready	not-applicable
	switchback-ready	not-applicable
	job-schedules	ok
	licenses	ok
	periodic-check-enabled	ok
	onboard-key-management	ok

```
12 entries were displayed.
```

The following example shows the execution of the command with the `-instance` parameter:

```
clusA::> metrocluster check cluster show -instance
```

```
Type of Check: negotiated-switchover-ready
```

```
Cluster Name: clusA
```

```
Result of the Check: not-applicable
```

```
Additional Information/Recovery Steps: Disaster recovery readiness checks  
are not performed as part of periodic metrocluster check. To run these  
checks, use the "metrocluster check run" command.
```

```
Type of Check: switchback-ready
```

```
Cluster Name: clusA
```

```
Result of the Check: not-applicable
```

```
Additional Information/Recovery Steps: Disaster recovery readiness checks  
are not performed as part of periodic metrocluster check. To run these  
checks, use the "metrocluster check run" command.
```

```
Type of Check: job-schedules
```

```
Cluster Name: clusA
```

```
Result of the Check: ok
```

```
Additional Information/Recovery Steps:
```

```
Type of Check: licenses
```

```
Cluster Name: clusA
```

```
Result of the Check: ok
```



```

Additional Information/Recovery Steps:
Type of Check: periodic-check-enabled
                Cluster Name: clusA
                Result of the Check: ok
Additional Information/Recovery Steps:
Type of Check: onboard-key-management
                Cluster Name: clusA
                Result of the Check: ok
Additional Information/Recovery Steps:
Type of Check: negotiated-switchover-ready
                Cluster Name: clusB
                Result of the Check: not-applicable
Additional Information/Recovery Steps: Disaster recovery readiness checks
are not performed as part of periodic metrocluster check. To run these
checks, use the "metrocluster check run" command.
Type of Check: switchback-ready
                Cluster Name: clusB
                Result of the Check: not-applicable
Additional Information/Recovery Steps: Disaster recovery readiness checks
are not performed as part of periodic metrocluster check. To run these
checks, use the "metrocluster check run" command.
Type of Check: job-schedules
                Cluster Name: clusB
                Result of the Check: ok
Additional Information/Recovery Steps:
Type of Check: licenses
                Cluster Name: clusB
                Result of the Check: ok
Additional Information/Recovery Steps:
Type of Check: periodic-check-enabled
                Cluster Name: clusB
                Result of the Check: ok
Additional Information/Recovery Steps:
Type of Check: onboard-key-management
                Cluster Name: clusB
                Result of the Check: ok
Additional Information/Recovery Steps:
12 entries were displayed.

```

Related Links

- [metrocluster check run](#)

metrocluster check config-replication show-aggregate-eligibility

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

level.

Description

The `metrocluster check config-replication show-aggregate-eligibility` command displays the MetroCluster configuration replication aggregate eligibility.

Parameters

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

If you specify the `-fields <fieldname>, ...` parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

| [-instance] }

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

[-aggregate <aggregate name>] - Aggregate

This is the aggregate name. If this parameter is specified, only rows with this aggregate will be displayed.

[-hosted-configuration-replication-volumes <volume name>,...] - Currently Hosted Configuration Replication Volumes

This is the list of the configuration replication volumes hosted on this aggregate. If this parameter is specified, only rows with these configuration replication volumes will be displayed.

[-is-eligible-to-host-additional-volumes {true|false}] - Eligibility to Host Another Configuration Replication Volume

This is the eligibility of the aggregate to host additional configuration replication volumes. If this parameter is specified, only rows with this eligibility will be displayed.

[-comment <text>] - Comment for Eligibility Status

This is a comment regarding the eligibility of the aggregate to host configuration replication volumes. If this parameter is specified, only rows with this comment will be displayed.

Examples

The following example shows the execution of the command in a MetroCluster configuration with thirteen aggregates in the cluster:

```

clusA::metrocluster check config-replication> show-aggregate-eligibility
Aggregate      Hosted Config Replication Vols      Eligible to
Comments                                             Host Addl Vols
-----
-----
a0              -                               false
Root Aggregate
a1             MDV_CRS_1bc7134a5ddf11e3b63f123478563412_A true      -
a2             MDV_CRS_1bc7134a5ddf11e3b63f123478563412_B true      -
a3              -                               false
Unable to determine available space of aggregate
a4              -                               false
Non-Local Aggregate
a5              -                               false
Non-Home Aggregate
a6              -                               false
Unable to determine mirror configuration
a7              -                               false
Mirror configuration does not match requirement
a8              -                               false
Disallowed Aggregate
a9              -                               false
Insufficient Space - 10GB required
a10            -                               false
Aggregate Offline
a11            -                               false
Inconsistent Aggregate
a12            -                               false
Aggregate Full
13 entries were displayed.

```

metrocluster check config-replication show-capture-status

Display MetroCluster capture status information

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

Description

The `metrocluster check config-replication show-capture-status` command indicates whether or not a configuration change that would prevent a negotiated switchover is currently being captured for replication.

Examples

The following example shows the execution of the command in a MetroCluster configuration when capture is not in progress:

```
cluster1::*> metrocluster check config-replication show-capture-status
Is Capture in Progress: false
```

metrocluster check config-replication show

Display MetroCluster config-replication status information

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

Description

The `metrocluster check config-replication show` command displays the results of MetroCluster configuration replication.

The command verifies the following aspects of MetroCluster configuration replication :

- **Enabled:** Verifies that MetroCluster configuration replication is enabled on the cluster.
- **Running:** Verifies that MetroCluster configuration replication is running on the cluster.
- **Remote Heartbeat:** Verifies that the MetroCluster configuration replication heartbeat with the remote cluster is healthy.
- **Last Heartbeat Sent:** Prints the timestamp of the last MetroCluster configuration replication heartbeat sent to the remote cluster.
- **Last Heartbeat Received:** Prints the timestamp of the last MetroCluster configuration replication heartbeat received from the remote cluster.
- **Storage Status:** Verifies that MetroCluster configuration replication storage is healthy.
- **Storage In Use:** Prints the location of MetroCluster configuration replication storage.
- **Storage Remarks:** Prints the underlying root cause for non healthy MetroCluster configuration storage.
- **Vserver Streams:** Verifies that MetroCluster configuration replication Vserver streams are healthy.
- **Cluster Streams:** Verifies that MetroCluster configuration replication Cluster streams are healthy.

Additional information about the warnings (if any) and recovery steps can be viewed by running the command with the `-instance` option.

Parameters

[`-instance`]

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

Examples

The following example shows the output of `metrocluster check config-replication show`:

```
clusA::metrocluster check config-replication> show
      Enabled: true
      Running: true
      Remote Heartbeat: ok
      Last Heartbeat Sent: 12/12/2013 14:24:59
      Last Heartbeat Received: 12/12/2013 14:25:00
      Storage Status: ok
      Storage In Use: Cluster-wide Volume:
MDV_CRS_1bc7134a5ddf11e3b63f123478563412_A
      Storage Remarks: -
      Vserver Streams: ok
      Cluster Streams: ok
```

metrocluster check lif repair-placement

Repair LIF placement for the sync-source Vserver LIFs in the destination cluster

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

Description

The `metrocluster check lif repair-placement` command reruns LIF placement for those LIFs displayed by the `metrocluster check lif show` command. This command is expected to be run after the admin manually rectifies the LIF placement failures displayed in the `metrocluster check lif show` command output. The command is successful if the LIF placement rerun does not encounter any LIF placement failure. This is to be confirmed by subsequent running of the `metrocluster check lif show` .

Parameters

-vserver <Vserver Name> - sync-source Vserver Name

This is the name of the sync source Vserver that has LIF placement failures as reported by the `metrocluster check lif show` command. This input ensures that the command is run on the specified Vserver.

[-lif <lif-name>] - Logical Interface Name

This is the Logical Interface name that belongs to the sync source Vserver that has a LIF placement failure in the destination cluster as reported by the `metrocluster check lif show` command. This input ensures that the command is run on the specified LIF only.

Examples

The following example shows the execution of the command with a sync source Vserver and a LIF specified:

```
clusA::> metrocluster check lif repair-placement -vserver vs1.example.com
-lif fcplif1
Command completed. Run the "metrocluster check lif show" command for
results.

clusA::> metrocluster check lif repair-placement -vserver vs1.example.com
-lif iscsilif1
Command completed. Run the "metrocluster check lif show" command for
results.
```

The following example shows the execution of the command with only a sync-source Vserver specified:

```
clusA::> metrocluster check lif repair-placement -vserver vs1.example.com

Command completed. Run the "metrocluster check lif show" command for
results.

clusA::>
```

Related Links

- [metrocluster check lif show](#)

metrocluster check lif show

Show results of MetroCluster check results for the data LIFs

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

Description

The `metrocluster check lif show` command displays the LIF placement failures in the MetroCluster configuration.

The command verifies the following aspects of the LIF placements of all the data LIFs in Metrocluster:

- `lif-placed-on-dr-node`: This check verifies that the LIF is placed on DR partner node.
- `port-selection`: This check verifies that the LIF is placed on correct port.

The LIF placement failures are mostly fabric/network connectivity issues that require manual intervention. Once the connectivity issues are resolved manually, the admin is expected to run [metrocluster check lif repair-placement](#) command to resolve the LIF placement issues for the sync source Vserver.

Additional information about the warnings (if any) and recovery steps can be viewed by running the command with the `-instance` option.

Parameters

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

If you specify the `-fields <fieldname>`, ... parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

| [-instance] }

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

[-cluster <Cluster name>] - Name of the Cluster

This is the name of the cluster the LIF belongs to. If this parameter is specified, only rows with this cluster will be displayed.

[-vserver <text>] - Name of the Vserver

This is the name of the Vserver in the MetroCluster configuration

[-lif <lif-name>] - Name of the Lif

This is the name of the LIF.

[-check <MetroCluster LIF placement Check>] - Description

This is the type of the check performed. If this parameter is specified, only rows with this check will be displayed.

[-result {ok|warning|not-run|not-applicable}] - Result of the Check

This is the result of the check performed. If this parameter is specified, only rows with this result will be displayed.

[-additional-info <text>] - Additional Information/Recovery Steps

This is additional information about the check. This field has more information and recovery steps for the warning. If this parameter is specified, only rows with this additional info will be displayed.

Examples

The following example shows the execution of the command in a MetroCluster configuration with two nodes per cluster:

```

clusA::>metrocluster check lif show
Cluster          Vserver      LIF          Check          Result
-----
ClusA            vs1          a_data1      lif-placed-on-dr-node  ok
                vs1          a_data1      port-selection      ok
                vs1          a_data1_inet6  lif-placed-on-dr-node  ok
                vs1          a_data1_inet6  port-selection      ok
ClusA            vs2-mc       b_data1      lif-placed-on-dr-node  ok
                vs2-mc       b_data1      port-selection
warning
                vs2-mc       b_data1_inet6  lif-placed-on-dr-node  ok
                vs2-mc       b_data1_inet6  port-selection
warning
ClusB            vs1-mc       a_data1      lif-placed-on-dr-node
warning
                vs1-mc       a_data1      port-selection      ok
                vs1-mc       a_data1_inet6  lif-placed-on-dr-node
warning
                vs1-mc       a_data1_inet6  port-selection      ok
ClusB            vs2          b_data1      lif-placed-on-dr-node  ok
                vs2          b_data1      port-selection      ok
                vs2          b_data1_inet6  lif-placed-on-dr-node  ok
                vs2          b_data1_inet6  port-selection      ok

16 entries were displayed.

```

Related Links

- [metrocluster check lif repair-placement](#)

metrocluster check node show

Show results of MetroCluster check for nodes

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

Description

The `metrocluster check node show` command displays the results of node checks performed by the [metrocluster check run](#) command.

The command displays the results of the following node configuration checks:

- `node-reachable`: This check verifies that the node is reachable.

- `metrocluster-ready`: This check verifies that the node is ready for MetroCluster configuration.
- `local-ha-partner`: This check verifies that the HA partner node is in the same cluster.
- `ha-mirroring-on`: This check verifies that HA mirroring for the node is configured.
- `ha-mirroring-op-state`: This check verifies that the HA mirroring operation is online.
- `symmetric-ha-relationship`: This check verifies that the relationship between the node and its HA partner is symmetric.
- `remote-dr-partner`: This check verifies that the DR partner node is in the remote cluster.
- `dr-mirroring-on`: This check verifies that DR mirroring for the node is configured.
- `dr-mirroring-op-state`: This check verifies that the DR mirroring operation is online.
- `symmetric-dr-relationship`: This check verifies that the relationship between the node and its DR partner is symmetric.
- `remote-dr-auxiliary-partner`: This check verifies that the DR auxiliary partner node is in the remote cluster.
- `symmetric-dr-auxiliary-relationship`: This check verifies that the relationship between the node and its DR auxiliary partner is symmetric.
- `storage-failover-enabled`: This check verifies that storage failover is enabled.
- `has-intercluster-lif`: This check verifies that the node has an intercluster LIF.
- `node-object-limit`: This check verifies that the node object limit option for the node is turned on.
- `automatic-uso`: This check verifies that the automatic USO option for the node is enabled.

Additional information about the warnings (if any) and recovery steps can be viewed by running the command with the `-instance` parameter.

Parameters

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

If you specify the `-fields <fieldname>, ...` parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

| [-instance] }

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

[-node <Node name>] - Node Name

This is the name of the node for which the check was run. If this parameter is specified, only rows with this node will be displayed.

[-check <MetroCluster Node Check>] - Type of Check

This is the type of the check performed. If this parameter is specified, only rows with this check will be displayed.

[-cluster <Cluster name>] - Cluster Name

This is the name of the cluster the node belongs to. If this parameter is specified, only rows with this cluster will be displayed.

[-result {ok|warning|not-run|not-applicable}] - Result of the Check

This is the result of the check. If this parameter is specified, only rows with this result will be displayed.

[-additional-info <text>] - Additional Information/Recovery Steps

This is additional information about the check. This field has more information and recovery steps for the warning. If this parameter is specified, only rows with this additional info will be displayed.

Examples

The following example shows the execution of the command in a MetroCluster configuration with two nodes per cluster:

```
clusA::> metrocluster check node show

Last Checked On: 9/12/2016 13:47:00

Node                Check                Result
-----
clusA-01
    node-reachable           ok
    metrocluster-ready       ok
    local-ha-partner         ok
    ha-mirroring-on          warning
    ha-mirroring-op-state    ok
    symmetric-ha-relationship warning
    remote-dr-partner        ok
    dr-mirroring-on          ok
    dr-mirroring-op-state    ok
    symmetric-dr-relationship ok
    remote-dr-auxiliary-partner ok
    symmetric-dr-auxiliary-relationship warning
    storage-failover-enabled ok
    has-intercluster-lif     ok
    node-object-limit        ok
    automatic-uso            ok

clusA-02
    node-reachable           ok
    metrocluster-ready       ok
    local-ha-partner         ok
    ha-mirroring-on          warning
    ha-mirroring-op-state    ok
    symmetric-ha-relationship warning
    remote-dr-partner        ok
    dr-mirroring-on          ok
    dr-mirroring-op-state    ok
    symmetric-dr-relationship ok
    remote-dr-auxiliary-partner ok
    symmetric-dr-auxiliary-relationship warning
    storage-failover-enabled ok
    has-intercluster-lif     ok
```

```

clusB-01
node-object-limit      ok
automatic-uso         ok

node-reachable        ok
metrocluster-ready    ok
local-ha-partner      ok
ha-mirroring-on      warning
ha-mirroring-op-state ok
symmetric-ha-relationship warning
remote-dr-partner     ok
dr-mirroring-on       ok
dr-mirroring-op-state ok
symmetric-dr-relationship ok
remote-dr-auxiliary-partner ok
symmetric-dr-auxiliary-relationship warning
storage-failover-enabled ok
has-intercluster-lif  ok
node-object-limit     ok
automatic-uso         ok

clusB-02
node-reachable        ok
metrocluster-ready    ok
local-ha-partner      ok
ha-mirroring-on      warning
ha-mirroring-op-state ok
symmetric-ha-relationship warning
remote-dr-partner     ok
dr-mirroring-on       ok
dr-mirroring-op-state ok
symmetric-dr-relationship ok
remote-dr-auxiliary-partner ok
symmetric-dr-auxiliary-relationship warning
storage-failover-enabled ok
has-intercluster-lif  ok
node-object-limit     ok
automatic-uso         ok

```

64 entries were displayed.

The following example shows the execution of the command with the `-instance` parameter:

```

clusA::> metrocluster check node show -instance
Node Name: clusA-01
           Type of Check: node-reachable
           Cluster Name: clusA
           Result of the Check: ok

```

Additional Information/Recovery Steps:

Node Name: clusA-01

Type of Check: metrocluster-ready

Cluster Name: clusA

Result of the Check: ok

Additional Information/Recovery Steps:

Node Name: clusA-01

Type of Check: local-ha-partner

Cluster Name: clusA

Result of the Check: ok

Additional Information/Recovery Steps:

Node Name: clusA-01

Type of Check: ha-mirroring-on

Cluster Name: clusA

Result of the Check: warning

Additional Information/Recovery Steps: Node's HA mirroring is not active.
Enable it on using "storage failover" commands.

Node Name: clusA-01

Type of Check: ha-mirroring-op-state

Cluster Name: clusA

Result of the Check: ok

Additional Information/Recovery Steps:

Node Name: clusA-01

Type of Check: symmetric-ha-relationship

Cluster Name: clusA

Result of the Check: warning

Additional Information/Recovery Steps: Partner not found. Check if node
"clusA-01's HA partner" is configured in MetroCluster.

Node Name: clusA-01

Type of Check: remote-dr-partner

Cluster Name: clusA

Result of the Check: ok

Additional Information/Recovery Steps:

Node Name: clusA-01

Type of Check: dr-mirroring-on

Cluster Name: clusA

Result of the Check: ok

Additional Information/Recovery Steps:

Node Name: clusA-01

Type of Check: dr-mirroring-op-state

Cluster Name: clusA

Result of the Check: ok

Additional Information/Recovery Steps:

Node Name: clusA-01

Type of Check: symmetric-dr-relationship

Cluster Name: clusA

```

                Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusA-01
                Type of Check: remote-dr-auxiliary-partner
                Cluster Name: clusA
                Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusA-01
                Type of Check: symmetric-dr-auxiliary-relationship
                Cluster Name: clusA
                Result of the Check: warning
Additional Information/Recovery Steps: Partner not found. Check if node
"clusA-01's DR auxiliary partner" is configured in MetroCluster.
Node Name: clusA-01
                Type of Check: storage-failover-enabled
                Cluster Name: clusA
                Result of the Check: warning
Additional Information/Recovery Steps: Node's storage failover is
disabled. Enable using "storage failover" commands.
Node Name: clusA-01
                Type of Check: has-intercluster-lif
                Cluster Name: clusA
                Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusA-01
                Type of Check: node-object-limit
                Cluster Name: clusA
                Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusB-01
                Type of Check: node-reachable
                Cluster Name: clusB
                Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusB-01
                Type of Check: metrocluster-ready
                Cluster Name: clusB
                Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusB-01
                Type of Check: local-ha-partner
                Cluster Name: clusB
                Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusB-01
                Type of Check: ha-mirroring-on

```

```

Cluster Name: clusB
Result of the Check: warning
Additional Information/Recovery Steps: Node's HA mirroring is not active.
Enable it on using "storage failover" commands.
Node Name: clusB-01
Type of Check: ha-mirroring-op-state
Cluster Name: clusB
Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusB-01
Type of Check: symmetric-ha-relationship
Cluster Name: clusB
Result of the Check: warning
Additional Information/Recovery Steps: Partner not found. Check if node
"clusB-01's HA partner" is configured in MetroCluster.
Node Name: clusB-01
Type of Check: remote-dr-partner
Cluster Name: clusB
Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusB-01
Type of Check: dr-mirroring-on
Cluster Name: clusB
Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusB-01
Type of Check: dr-mirroring-op-state
Cluster Name: clusB
Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusB-01
Type of Check: symmetric-dr-relationship
Cluster Name: clusB
Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusB-01
Type of Check: remote-dr-auxiliary-partner
Cluster Name: clusB
Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusB-01
Type of Check: symmetric-dr-auxiliary-relationship
Cluster Name: clusB
Result of the Check: warning
Additional Information/Recovery Steps: Partner not found. Check if node
"clusB-01's DR auxiliary partner" is configured in MetroCluster.

```

```

Node Name: clusB-01
                Type of Check: storage-failover-enabled
                Cluster Name: clusB
                Result of the Check: warning
Additional Information/Recovery Steps: Node's storage failover is
disabled. Enable using "storage failover" commands.
Node Name: clusB-01
                Type of Check: has-intercluster-lif
                Cluster Name: clusB
                Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusB-01
                Type of Check: node-object-limit
                Cluster Name: clusB
                Result of the Check: ok
Additional Information/Recovery Steps:
Node Name: clusB-01
                Type of Check: automatic-uso
                Cluster Name: clusB
                Result of the Check: ok
Additional Information/Recovery Steps:
32 entries were displayed.

```

Related Links

- [metrocluster check run](#)

metrocluster config-replication commands

metrocluster config-replication cluster-storage-configuration modify

Modify MetroCluster storage configuration information

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

Description

The `metrocluster config-replication cluster-storage-configuration modify` command modifies the configuration of storage used for configuration replication.

Parameters

[`-disallowed-aggregates <aggregate name>,...`] - Disallowed Aggregates

Use this parameter to set the list of storage aggregates that are not available to host storage for configuration replication.

Examples

The following example disallows two aggregates named *aggr1* and *aggr2*:

```
cluster1::*> metrocluster config-replication cluster-storage-configuration
modify -disallowed-aggregates aggr1,aggr2
```

metrocluster config-replication cluster-storage-configuration show

Display MetroCluster storage configuration information

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

Description

The `metrocluster config-replication cluster-storage-configuration show` command shows details of the configuration of the storage used for configuration replication.

The information displayed is the following:

- Disallowed Aggregates - The list of storage aggregates that are configured as not allowed to host storage areas.
- Auto-Repair - Displays *true* if the automatic repair of storage areas used by configuration replication is enabled.
- Auto-Recreate - Displays *true* if the automatic recreation of storage volumes used by configuration replication is enabled.
- Use Mirrored Aggregate - Displays *true* if storage areas for configuration replication are to be hosted on a mirrored aggregate.

Examples

The following is an example of the `metrocluster config-replication cluster-storage-configuration show` command:

```
cluster1::*> metrocluster config-replication cluster-storage-configuration
show
Disallowed Aggregates: -
    Auto-Repair: true
    Auto-Recreate: true
Use Mirrored Aggregate: true
```

metrocluster config-replication resync-status show

Display MetroCluster Configuration Resynchronization Status

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

Description

The `metrocluster config-replication resync-status show` command displays the state of the configuration synchronization operation between the two clusters in the MetroCluster configuration.

This command displays the following details about the local cluster and the peer cluster:

- Source: This is the source side whose configuration is being replicated to the destination side.
- Destination: This is the destination side where the configuration is being replicated to from the source side.
- State: This is the state of the synchronization operation.
- % Complete: This is completion percentage of the operation.

Examples

The following example shows the output of the command when synchronization is in progress:

```
clusterA::> metrocluster config-replication resync-status show
      Source                Destination                State                %
Complete
-----
-----
      clusterA                clusterB                complete                -
      clusterB                clusterA                complete                -
```

The following example shows the output of the command when synchronization from clusB to clusA is in progress:

```
clusA::> metrocluster config-replication resync-status show
      Source                Destination                State                %
Complete
-----
-----
      clusterA                clusterB                complete                -
      clusterB                clusterA                messaging                95
```

metrocluster configuration-settings commands

metrocluster configuration-settings show-status

Display the configuration settings status for a MetroCluster setup

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

Description

The `metrocluster configuration-settings show-status` command displays the configuration

settings status for nodes in a MetroCluster setup. If a DR group has not been created, then status for nodes in the local cluster only are displayed.

Parameters

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

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

| [-instance] }

If this parameter is specified, the command displays detailed information about all entries.

[-cluster-uuid <UUID>] - Cluster UUID

If this parameter is specified, the command displays detailed information about all nodes in the cluster matching the specified cluster-uuid.

[-cluster <Cluster name>] - Cluster Name

If this parameter is specified, the command displays detailed information about all the nodes in the specified cluster.

[-node <text>] - Node Name

If this parameter is specified, the command displays information for the matching nodes.

[-configuration-status <text>] - Configuration Settings Status

If this parameter is specified, the command displays detailed information about all nodes with the specified configuration status.

[-config-error-info <text>] - Configuration Error Information

If this parameter is specified, the command displays detailed information about all nodes with the specified configuration error information.

Examples

The following example shows the display of MetroCluster setup status:

```
Nodes do not have a valid platform-specific personality value (equivalent
to HAOSC parameter on non-Apollo platforms) for a MetroCluster setup.
clusA::> metrocluster configuration-settings show-status
Cluster                Node                Configuration Settings
Status
-----
clusA                   A1                not a MetroCluster setup
                       A2                not a MetroCluster setup
2 entries were displayed.
MetroCluster setup uses FC links rather than IP
xref:{relative_path}clusA::> metrocluster configuration-settings show-
status
Cluster                Node                Configuration Settings
```

Status

```
-----  
-----  
clusA          A1          not applicable for FC and  
SAS  
              A2          not applicable for FC and  
SAS
```

2 entries were displayed.

Output of the command when MetroCluster setup uses IP links and before
`"metrocluster configuration-settings dr-group create"` command is run:

```
clusA::> metrocluster configuration-settings show-status  
Cluster          Node          Configuration Settings  
Status  
-----  
-----
```

```
clusA          A1          ready for DR group create  
              A2          ready for DR group create
```

2 entries were displayed.

Output of the command after `"metrocluster configuration-settings dr-group
create"` command is run:

```
clusA::> metrocluster configuration-settings show-status  
Cluster          Node          Configuration Settings  
Status  
-----  
-----
```

```
clusA  
              A1          ready for interface create  
              A2          ready for interface create  
clusB  
              B1          ready for interface create  
              B2          ready for interface create
```

4 entries were displayed.

Output of the command after `"metrocluster configuration-settings
interface create"` command is run for every node:

```
clusA::> metrocluster configuration-settings show-status  
Cluster          Node          Configuration Settings  
Status  
-----  
-----
```

```
clusA  
create          A1          ready for next interface  
              A2          ready for connection connect  
clusB  
              B1          ready for connection connect  
              B2          ready for connection connect
```

```

4 entries were displayed.
Output of the command after ` "metrocluster configuration-settings
connection connect" ` command is run:
usA::> metrocluster configuration-settings show-status
Cluster                Node                Configuration Settings
Status
-----
clusA
                        A1                completed
                        A2                completed
clusB
                        B1                completed
                        B2                completed

4 entries were displayed.
Output of the command after ` "metrocluster configuration-settings
connection connect" ` command is run and there are connection errors:
clusA::> metrocluster configuration-settings show-status
Cluster                Node                Configuration Settings
Status
-----
clusA
                        A1                connection error
                        A2                completed
clusB
                        B1                connection error
                        B2                completed

4 entries were displayed.

```

metrocluster configuration-settings connection check

Check the network connections between partner nodes

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

Description

The `metrocluster configuration-settings connection check` command checks the settings of a MetroCluster over IP configuration.

This command is used for MetroCluster configurations that are connected through IP links.

Examples

The following example shows the output for the check command in MetroCluster over IP configurations:

```
clusA:*> metrocluster configuration-settings connection check
[Job 68] Job succeeded: Connect is successful.

Begin connection check.
Start checking the partner cluster.
    Check partner cluster: PASS.
Start checking the configuration settings.
    Check configuration settings: PASS.
Start pinging the network endpoints from cluster "clusA".
    Ping network endpoints: PASS.
Start pinging the network endpoints from cluster "clusB".
    Ping network endpoints: PASS.
Start checking the network MTU sizes from cluster "clusA".
    Check network MTU sizes: PASS.
Start checking the network MTU sizes from cluster "clusB".
    Check network MTU sizes: PASS.
Start checking the network subnets from cluster "clusA".
    Check network subnets: PASS.
Start checking the network subnets from cluster "clusB".
    Check network subnets: PASS.
Start checking the storage daemons on cluster "clusA".
    Check storage daemons: PASS.
Start checking the storage daemons on cluster "clusB".
    Check storage daemons: PASS.
End of connection check.
```

metrocluster configuration-settings connection connect

Configure the network connections between partner nodes

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

Description

The `metrocluster configuration-settings connection connect` command configures the connections that mirror NV logs and access remote storage between partner nodes in a MetroCluster setup.

This command is used for MetroCluster setups that are connected through IP links. MetroCluster setups that are connected through FC links will configure the FC connections automatically.

The `metrocluster configuration-settings` commands are run in the following order to set up MetroCluster:

- `metrocluster configuration-settings dr-group create` ,
- `metrocluster configuration-settings interface create` ,
- `metrocluster configuration-settings connection connect` .

Before this command is run

- The DR groups must have been configured. Run the `metrocluster configuration-settings dr-group show` command to verify that every node is partnered in a DR group.
- The network logical interfaces must have been configured on every node. Use the `metrocluster configuration-settings interface show` command to verify that every node has network logical interfaces configured to mirror NV logs and access remote storage.

After this command completes successfully, every node will:

- Have NV log mirroring configured and mirroring disabled. NV log mirroring will be enabled by the `metrocluster configure` command.
- Have access to remote storage. Use the `storage disk show -pool Pool1` command to view the remote disks that are hosted on DR partner nodes.

The DR groups and network logical interfaces that were configured by the `metrocluster configuration-settings` commands cannot be deleted after the connections have been configured. The `metrocluster configuration-settings connection disconnect` command must be run to remove the connections before the DR groups and network logical interfaces can be deleted.

Examples

The following example shows configuration of connections in a MetroCluster over IP setup:

```
clusA::> metrocluster configuration-settings connection connect
[Job 269] Job succeeded: Connect is successful.
clusA::> metrocluster configuration-settings connection show
DR
Group Cluster Node      Source      Destination
State                               Network Address Network Address Partner Type Config
-----
1      clusA A1
      Home Port: e0f
      10.140.113.214  10.140.113.216  HA Partner
completed
      Home Port: e0f
      10.140.113.214  10.140.113.218  DR Partner
completed
      Home Port: e0f
      10.140.113.214  10.140.113.249  DR Auxiliary
completed
      Home Port: e0g
      10.140.113.215  10.140.113.217  HA Partner
completed
      Home Port: e0g
```

```

10.140.113.215 10.140.113.248 DR Partner
completed
Home Port: e0g
10.140.113.215 10.140.113.25 DR Auxiliary
completed
A2
Home Port: e0f
10.140.113.216 10.140.113.214 HA Partner
completed
Home Port: e0f
10.140.113.216 10.140.113.249 DR Partner
completed
Home Port: e0f
10.140.113.216 10.140.113.218 DR Auxiliary
completed
Home Port: e0g
10.140.113.217 10.140.113.215 HA Partner
completed
Home Port: e0g
10.140.113.217 10.140.113.25 DR Partner
completed
Home Port: e0g
10.140.113.217 10.140.113.248 DR Auxiliary
completed
clusB B2
Home Port: e0f
10.140.113.249 10.140.113.218 HA Partner
completed
Home Port: e0f
10.140.113.249 10.140.113.216 DR Partner
completed
Home Port: e0f
10.140.113.249 10.140.113.214 DR Auxiliary
completed
Home Port: e0g
10.140.113.25 10.140.113.248 HA Partner
completed
Home Port: e0g
10.140.113.25 10.140.113.217 DR Partner
completed
Home Port: e0g
10.140.113.25 10.140.113.215 DR Auxiliary
completed
B1
Home Port: e0f
10.140.113.218 10.140.113.249 HA Partner

```

```

completed
      Home Port: e0f
      10.140.113.218  10.140.113.214  DR Partner
completed
      Home Port: e0f
      10.140.113.218  10.140.113.216  DR Auxiliary
completed
      Home Port: e0g
      10.140.113.248  10.140.113.25   HA Partner
completed
      Home Port: e0g
      10.140.113.248  10.140.113.215  DR Partner
completed
      Home Port: e0g
      10.140.113.248  10.140.113.217  DR Auxiliary
completed
24 entries were displayed.
clusA::> metrocluster configuration-settings show-status
Cluster          Node          Configuration Settings
Status
-----
clusA
              A1          completed
              A2          completed
clusB
              B1          completed
              B2          completed
4 entries were displayed.

```

Related Links

- [metrocluster configuration-settings dr-group create](#)
- [metrocluster configuration-settings interface create](#)
- [metrocluster configuration-settings dr-group show](#)
- [metrocluster configuration-settings interface show](#)
- [metrocluster configure](#)
- [storage disk show](#)
- [metrocluster configuration-settings connection disconnect](#)

metrocluster configuration-settings connection disconnect

Tear down the network connections between partner nodes

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

Description

The `metrocluster configuration-settings connection disconnect` command removes the connections between nodes in a DR group that are used to mirror NV logs and access remote storage.

This command cannot be run if a node in the DR group has remote disks assigned to the node. The assigned ownership of remote disks can be removed by running the `storage disk removeowner` command.

The `metrocluster configuration-settings` commands are run in the following order to remove MetroCluster over IP configuration:

- `metrocluster configuration-settings connection disconnect`,
- `metrocluster configuration-settings interface delete`,
- `metrocluster configuration-settings dr-group delete`.

Parameters

-dr-group-id <integer> - DR Group ID

This parameter identifies the DR group to be disconnected.

Examples

The following example illustrates removal of connections in a four-node MetroCluster setup:

```
clusA::> metrocluster configuration-settings connection disconnect -dr
-group-id 1
[Job 270] Job succeeded: Disconnect is successful.

clusA::> metrocluster configuration-settings show-status
Cluster          Node          Configuration Settings
Status
-----
clusA
              A1          ready for connection connect
              A2          ready for connection connect
clusB
              B1          ready for connection connect
              B2          ready for connection connect

4 entries were displayed.
clusA::> metrocluster configuration-settings connection show
DR              Source          Destination
Group Cluster Node   Network Address Network Address Partner Type Config
State
-----
1      clusA A1
              Home Port: e0f
```

disconnected	10.140.113.214	10.140.113.216	HA Partner
	Home Port: e0f		
disconnected	10.140.113.214	10.140.113.218	DR Partner
	Home Port: e0f		
disconnected	10.140.113.214	10.140.113.249	DR Auxiliary
	Home Port: e0g		
disconnected	10.140.113.215	10.140.113.217	HA Partner
	Home Port: e0g		
disconnected	10.140.113.215	10.140.113.248	DR Partner
	Home Port: e0g		
disconnected	10.140.113.215	10.140.113.25	DR Auxiliary
	A2		
disconnected	Home Port: e0f		
	10.140.113.216	10.140.113.214	HA Partner
	Home Port: e0f		
disconnected	10.140.113.216	10.140.113.249	DR Partner
	Home Port: e0f		
disconnected	10.140.113.216	10.140.113.218	DR Auxiliary
	Home Port: e0g		
disconnected	10.140.113.217	10.140.113.215	HA Partner
	Home Port: e0g		
disconnected	10.140.113.217	10.140.113.25	DR Partner
	Home Port: e0g		
disconnected	10.140.113.217	10.140.113.248	DR Auxiliary
	clusB B2		
disconnected	Home Port: e0f		
	10.140.113.249	10.140.113.218	HA Partner
	Home Port: e0f		
disconnected	10.140.113.249	10.140.113.216	DR Partner
	Home Port: e0f		
disconnected	10.140.113.249	10.140.113.214	DR Auxiliary

```

disconnected      Home Port: e0g
                  10.140.113.25   10.140.113.248   HA Partner
disconnected      Home Port: e0g
                  10.140.113.25   10.140.113.217   DR Partner
disconnected      Home Port: e0g
                  10.140.113.25   10.140.113.215   DR Auxiliary
                  B1
disconnected      Home Port: e0f
                  10.140.113.218  10.140.113.249   HA Partner
disconnected      Home Port: e0f
                  10.140.113.218  10.140.113.214   DR Partner
disconnected      Home Port: e0f
                  10.140.113.218  10.140.113.216   DR Auxiliary
disconnected      Home Port: e0g
                  10.140.113.248  10.140.113.25    HA Partner
disconnected      Home Port: e0g
                  10.140.113.248  10.140.113.215   DR Partner
disconnected      Home Port: e0g
                  10.140.113.248  10.140.113.217   DR Auxiliary
24 entries were displayed.

```

Related Links

- [storage disk removeowner](#)
- [metrocluster configuration-settings interface delete](#)
- [metrocluster configuration-settings dr-group delete](#)

metrocluster configuration-settings connection show

Display the connections between partner nodes in a MetroCluster setup

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

Description

The `metrocluster configuration-settings connection show` command displays the connection configuration information between the nodes in a MetroCluster setup.

Parameters

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

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

| [-instance] }

If this parameter is specified, the command displays detailed information about all entries.

[-dr-group-id <integer>] - DR Group ID

If this parameter is specified, the command displays information for the matching DR group.

[-cluster-uuid <UUID>] - Cluster UUID

If this parameter is specified, the command displays information for the matching cluster specified by uuid.

[-cluster <Cluster name>] - Cluster Name

If this parameter is specified, the command displays information for the matching cluster.

[-node-uuid <UUID>] - Node UUID

If this parameter is specified, the command displays information for the matching node specified by uuid.

[-node <text>] - Node Name

If this parameter is specified, the command displays information for the matching nodes.

[-home-port {<netport>|<ifgrp>}] - Home Port

If this parameter is specified, the command displays information for the matching home-port.

[-relationship-type <Roles of MetroCluster Nodes>] - Relationship Role Type

If this parameter is specified, the command displays information for the matching relationship-type.

[-source-address <IP Address>] - Source Network Address

If this parameter is specified, the command displays information for the matching source address.

[-destination-address <IP Address>] - Destination Network Address

If this parameter is specified, the command displays information for the matching destination address.

[-partner-cluster-uuid <UUID>] - Partner Cluster UUID

If this parameter is specified, the command displays information for the matching partner-cluster-uuid.

[-partner-node-uuid <UUID>] - Partner Node UUID

If this parameter is specified, the command displays information for the matching partner-node-uuid.

[-partner-node <text>] - Partner Node Name

If this parameter is specified, the command displays information for the matching partner-node.

[-partner-type <text>] - Partner Relationship Type

If this parameter is specified, the command displays information for the matching partner-type.

[-config-state <text>] - Configuration State

If this parameter is specified, the command displays information for the matching config-state.

[-config-error-info <text>] - Configuration Error Information

If this parameter is specified, the command displays information for the matching config-error-info.

Examples

The following example shows the output of `metrocluster configuration-settings connection connect` command:

```
Output of the command before the connections are established using the
xref:{relative_path}metrocluster-configuration-settings-connection-
connect.html[metrocluster configuration-settings connection connect]
command:
clusA::> metrocluster configuration-settings connection show
DR                               Source           Destination
Group Cluster Node   Network Address Network Address Partner Type Config
State
-----
1      clusA A1
      Home Port: e0f
      10.140.113.214  10.140.113.216  HA Partner
disconnected
      Home Port: e0f
      10.140.113.214  10.140.113.218  DR Partner
disconnected
      Home Port: e0f
      10.140.113.214  10.140.113.249  DR Auxiliary
disconnected
      Home Port: e0g
      10.140.113.215  10.140.113.217  HA Partner
disconnected
      Home Port: e0g
      10.140.113.215  10.140.113.248  DR Partner
disconnected
      Home Port: e0g
      10.140.113.215  10.140.113.25   DR Auxiliary
disconnected
      A2
      Home Port: e0f
      10.140.113.216  10.140.113.214  HA Partner
disconnected
      Home Port: e0f
      10.140.113.216  10.140.113.249  DR Partner
disconnected
```

disconnected	Home Port: e0f 10.140.113.216	10.140.113.218	DR Auxiliary
disconnected	Home Port: e0g 10.140.113.217	10.140.113.215	HA Partner
disconnected	Home Port: e0g 10.140.113.217	10.140.113.25	DR Partner
disconnected	Home Port: e0g 10.140.113.217	10.140.113.248	DR Auxiliary
	clusB B2		
disconnected	Home Port: e0f 10.140.113.249	10.140.113.218	HA Partner
disconnected	Home Port: e0f 10.140.113.249	10.140.113.216	DR Partner
disconnected	Home Port: e0f 10.140.113.249	10.140.113.214	DR Auxiliary
disconnected	Home Port: e0g 10.140.113.25	10.140.113.248	HA Partner
disconnected	Home Port: e0g 10.140.113.25	10.140.113.217	DR Partner
disconnected	Home Port: e0g 10.140.113.25	10.140.113.215	DR Auxiliary
	B1		
disconnected	Home Port: e0f 10.140.113.218	10.140.113.249	HA Partner
disconnected	Home Port: e0f 10.140.113.218	10.140.113.214	DR Partner
disconnected	Home Port: e0f 10.140.113.218	10.140.113.216	DR Auxiliary
disconnected	Home Port: e0g 10.140.113.248	10.140.113.25	HA Partner
disconnected	Home Port: e0g 10.140.113.248	10.140.113.215	DR Partner

disconnected

Home Port: e0g
10.140.113.248 10.140.113.217 DR Auxiliary

disconnected

24 entries were displayed.

Output of the command after the connections are established using the
xref:{relative_path}metrocluster-configuration-settings-connection-
connect.html[metrocluster configuration-settings connection connect]
command:

clusA::> metrocluster configuration-settings connection show

DR	Source	Destination		
Group	Cluster	Node	Network	Network
State			Address	Address
			Partner	Type
				Config

1 clusA A1

Home Port: e0f
10.140.113.214 10.140.113.216 HA Partner

completed

Home Port: e0f
10.140.113.214 10.140.113.218 DR Partner

completed

Home Port: e0f
10.140.113.214 10.140.113.249 DR Auxiliary

completed

Home Port: e0g
10.140.113.215 10.140.113.217 HA Partner

completed

Home Port: e0g
10.140.113.215 10.140.113.248 DR Partner

completed

Home Port: e0g
10.140.113.215 10.140.113.25 DR Auxiliary

completed

A2

Home Port: e0f
10.140.113.216 10.140.113.214 HA Partner

completed

Home Port: e0f
10.140.113.216 10.140.113.249 DR Partner

completed

Home Port: e0f
10.140.113.216 10.140.113.218 DR Auxiliary

completed

Home Port: e0g
10.140.113.217 10.140.113.215 HA Partner

```

completed
      Home Port: e0g
      10.140.113.217  10.140.113.25  DR Partner
completed
      Home Port: e0g
      10.140.113.217  10.140.113.248  DR Auxiliary
completed
      clusB B2
      Home Port: e0f
      10.140.113.249  10.140.113.218  HA Partner
completed
      Home Port: e0f
      10.140.113.249  10.140.113.216  DR Partner
completed
      Home Port: e0f
      10.140.113.249  10.140.113.214  DR Auxiliary
completed
      Home Port: e0g
      10.140.113.25   10.140.113.248  HA Partner
completed
      Home Port: e0g
      10.140.113.25   10.140.113.217  DR Partner
completed
      Home Port: e0g
      10.140.113.25   10.140.113.215  DR Auxiliary
completed
      B1
      Home Port: e0f
      10.140.113.218  10.140.113.249  HA Partner
completed
      Home Port: e0f
      10.140.113.218  10.140.113.214  DR Partner
completed
      Home Port: e0f
      10.140.113.218  10.140.113.216  DR Auxiliary
completed
      Home Port: e0g
      10.140.113.248  10.140.113.25   HA Partner
completed
      Home Port: e0g
      10.140.113.248  10.140.113.215  DR Partner
completed
      Home Port: e0g
      10.140.113.248  10.140.113.217  DR Auxiliary
completed
24 entries were displayed.

```

Related Links

- [metrocluster configuration-settings connection connect](#)

metrocluster configuration-settings dr-group create

Create a DR group in a MetroCluster setup

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

Description

The `metrocluster configuration-settings dr-group create` command partners the nodes that will comprise a DR group in a MetroCluster setup.

This command is used for MetroCluster setups that are connected through IP links. MetroCluster setups that are connected through FC links will configure DR groups automatically and do not require the `metrocluster configuration-settings` commands.

The `metrocluster configuration-settings` commands are run in the following order to set up MetroCluster:

- `metrocluster configuration-settings dr-group create`,
- [metrocluster configuration-settings interface create](#),
- [metrocluster configuration-settings connection connect](#).

Before running this command, cluster peering must be configured between the local and partner clusters. Run the `cluster peer show` command to verify that peering is available between the local and partner clusters.

This command configures a local node and a remote node as DR partner nodes. The command also configures the HA partner of the local node and the HA partner of the remote node as the other DR partner nodes in the DR group.

Parameters

-partner-cluster <Cluster name> - Partner Cluster Name

Use this parameter to specify the name of the partner cluster.

-local-node {<nodename>|local} - Local Node Name

Use this parameter to specify the name of a node in the local cluster.

-remote-node <text> - Remote Node Name

Use this parameter to specify the name of a node in the partner cluster that is to be the DR partner of the specified local node.

Examples

The following example shows the creation of the MetroCluster DR group:

```

clusA::> metrocluster configuration-settings dr-group create -partner
-cluster clusB -local-node A1 -remote-node B1
[Job 268] Job succeeded: DR Group Create is successful.

clusA::> metrocluster configuration-settings dr-group show
DR Group ID Cluster Node DR Partner Node
-----
-----
1 clusA
A1 B1
A2 B2
clusB
B2 A2
B1 A1

4 entries were displayed.
clusA::> metrocluster configuration-settings show-status
Cluster Node Configuration Settings
Status
-----
-----
clusA
A1 ready for interface create
A2 ready for interface create
clusB
B1 ready for interface create
B2 ready for interface create

4 entries were displayed.

```

Related Links

- [metrocluster configuration-settings interface create](#)
- [metrocluster configuration-settings connection connect](#)
- [cluster peer show](#)

metrocluster configuration-settings dr-group delete

Delete a DR group in a MetroCluster setup

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

Description

The `metrocluster configuration-settings dr-group delete` command deletes a DR group and its node partnerships that were configured using the [metrocluster configuration-settings dr-group create](#) command.

This command cannot be run if the [metrocluster configuration-settings interface create](#) command has

configured a network logical interface on a network port provisioned for MetroCluster. The `metrocluster configuration-settings interface delete` command must then be run to delete the network logical interfaces on every node in the DR group.

The `metrocluster configuration-settings` commands are run in the following order to remove the MetroCluster over IP configuration:

- `metrocluster configuration-settings connection disconnect` ,
- `metrocluster configuration-settings interface delete` ,
- `metrocluster configuration-settings dr-group delete` .

Parameters

`-dr-group-id <integer>` - Dr group Id

This parameter identifies the DR group to be deleted.

Examples

The following example shows the deletion of the MetroCluster DR group:

```
clusA::> metrocluster configuration-settings dr-group delete -dr-group-id 1
```

```
Warning: This command deletes the existing DR group relationship. Are you sure
```

```
        you want to proceed ? {y|n}: y
```

```
[Job 279] Job succeeded: DR Group Delete is successful.
```

```
clusA::> metrocluster configuration-settings dr-group show
No DR groups exist.
```

```
clusA::> metrocluster configuration-settings show-status
```

```
Cluster          Node          Configuration Settings
```

```
Status
```

```
-----
-----
clusA
      A1          ready for DR group create
      A2          ready for DR group create
clusB
      B1          ready for DR group create
      B2          ready for DR group create
```

```
4 entries were displayed.
```

Related Links

- [metrocluster configuration-settings dr-group create](#)
- [metrocluster configuration-settings interface create](#)
- [metrocluster configuration-settings interface delete](#)
- [metrocluster configuration-settings connection disconnect](#)

metrocluster configuration-settings dr-group show

Display the DR groups in a MetroCluster setup

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

Description

The `metrocluster configuration-settings dr-group show` command displays the DR groups and their nodes.

Parameters

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

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

| [-instance] }

If this parameter is specified, the command displays detailed information about all entries.

[-dr-group-id <integer>] - DR Group ID

If this parameter is specified, the command displays information for the matching DR group.

[-cluster-uuid <UUID>] - Cluster UUID

If this parameter is specified, the command displays information for the matching cluster uuid.

[-cluster <Cluster name>] - Cluster Name

If this parameter is specified, the command displays information for the specified cluster.

[-node-uuid <UUID>] - Node UUID

If this parameter is specified, the command displays information for the matching nodes uuid.

[-node <text>] - Node Name

If this parameter is specified, the command displays information for the matching nodes.

[-dr-partner-node-uuid <UUID>] - DR Partner Node UUID

If this parameter is specified, the command displays information for the matching DR partner node uuid.

[-dr-partner-node <text>] - DR Partner Node Name

If this parameter is specified, the command displays information for the matching DR partner nodes.

Examples

The following example illustrates the display of DR group configuration in a four-node MetroCluster setup:

```
clusA::> metrocluster configuration-settings dr-group show
DR Group ID Cluster                               Node                               DR Partner Node
-----
-----
1          clusA
          A1                               B1
          A2                               B2
          clusB
          B2                               A2
          B1                               A1

4 entries were displayed.
```

metrocluster configuration-settings interface create

Create a MetroCluster interface

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

Description

The `metrocluster configuration-settings interface create` command configures the network logical interfaces that will be used on a node in a MetroCluster setup to mirror NV logs and access remote storage.

This command is used for MetroCluster setups that are connected through IP links. MetroCluster setups that are connected through FC links do not require the user to provision network logical interfaces to mirror NV logs and access remote storage.

The `metrocluster configuration-settings` commands are run in the following order to set up MetroCluster:

- [metrocluster configuration-settings dr-group create](#) ,
- `metrocluster configuration-settings interface create` ,
- [metrocluster configuration-settings connection connect](#) .

Before running this command , the node's DR group must be configured using the [metrocluster configuration-settings dr-group create](#) command. Run the [metrocluster configuration-settings dr-group show](#) command to verify that the node's DR group has been configured.

Parameters

-cluster-name <Cluster name> - Cluster Name

Use this parameter to specify the name of the local or partner cluster.

-home-node <text> - Home Node

Use this parameter to specify the home node in the cluster which hosts the interface.

-home-port {<netport>|<ifgrp>} - Home Port

Use this parameter to specify the home port provisioned for MetroCluster.

-address <IP Address> - Network Address

Use this parameter to specify the network address to be assigned to the home port.

-netmask <Contiguous IP Mask> - Netmask

Use this parameter to specify the network mask to be assigned to the interface.

Examples

This example shows configuring logical interface on MetroCluster IP capable port:

```

clusA::> metrocluster configuration-settings interface create -cluster
-name clusA -home-node A1 -home-port e0f -address 10.140.113.214 -netmask
255.255.192.0
[Job 281] Job succeeded: Interface Create is successful.

clusA::> metrocluster configuration-settings interface show
DR
Config
Group Cluster Node      Network Address Netmask      Gateway
State
-----
-----
1      clusA A1
                Home Port: e0f
                                10.140.113.214 255.255.192.0  -
completed
Output after configuring all the interfaces:
clusA::> metrocluster configuration-settings interface show
DR
Config
Group Cluster Node      Network Address Netmask      Gateway
State
-----
-----
1      clusA A1
                Home Port: e0f
                                10.140.113.214 255.255.192.0  -
completed
                Home Port: e0g
                                10.140.113.215 255.255.192.0  -
completed

```

```

                A2
      Home Port: e0f
                10.140.113.216  255.255.192.0  -
completed
      Home Port: e0g
                10.140.113.217  255.255.192.0  -
completed
      clusB B2
      Home Port: e0f
                10.140.113.249  255.255.192.0  -
completed
      Home Port: e0g
                10.140.113.25   255.255.192.0  -
completed
                B1
      Home Port: e0f
                10.140.113.218  255.255.192.0  -
completed
      Home Port: e0g
                10.140.113.248  255.255.192.0  -
completed
8 entries were displayed.

```

```
clusA::> metrocluster configuration-settings show-status
```

Cluster	Node	Configuration	Settings
clusA	A1	ready for connection	connect
	A2	ready for connection	connect
clusB	B1	ready for connection	connect
	B2	ready for connection	connect

```
4 entries were displayed.
```

```
clusA::> metrocluster configuration-settings connection show
```

DR	Source	Destination	Partner	Type	Config
Group	Cluster	Node	Network Address	Network Address	Partner
1	clusA	A1	10.140.113.214	10.140.113.216	HA Partner

disconnected

```

disconnected      Home Port: e0f
                  10.140.113.214  10.140.113.218  DR Partner

disconnected      Home Port: e0f
                  10.140.113.214  10.140.113.249  DR Auxiliary

disconnected      Home Port: e0g
                  10.140.113.215  10.140.113.217  HA Partner

disconnected      Home Port: e0g
                  10.140.113.215  10.140.113.248  DR Partner

disconnected      Home Port: e0g
                  10.140.113.215  10.140.113.25   DR Auxiliary

disconnected      A2
                  Home Port: e0f
                  10.140.113.216  10.140.113.214  HA Partner

disconnected      Home Port: e0f
                  10.140.113.216  10.140.113.249  DR Partner

disconnected      Home Port: e0f
                  10.140.113.216  10.140.113.218  DR Auxiliary

disconnected      Home Port: e0g
                  10.140.113.217  10.140.113.215  HA Partner

disconnected      Home Port: e0g
                  10.140.113.217  10.140.113.25   DR Partner

disconnected      Home Port: e0g
                  10.140.113.217  10.140.113.248  DR Auxiliary

disconnected      clusB B2
                  Home Port: e0f
                  10.140.113.249  10.140.113.218  HA Partner

disconnected      Home Port: e0f
                  10.140.113.249  10.140.113.216  DR Partner

disconnected      Home Port: e0f
                  10.140.113.249  10.140.113.214  DR Auxiliary

disconnected      Home Port: e0g
                  10.140.113.25   10.140.113.248  HA Partner

```



```

disconnected
    Home Port: e0g
    10.140.113.25    10.140.113.217    DR Partner
disconnected
    Home Port: e0g
    10.140.113.25    10.140.113.215    DR Auxiliary
disconnected
    B1
    Home Port: e0f
    10.140.113.218    10.140.113.249    HA Partner
disconnected
    Home Port: e0f
    10.140.113.218    10.140.113.214    DR Partner
disconnected
    Home Port: e0f
    10.140.113.218    10.140.113.216    DR Auxiliary
disconnected
    Home Port: e0g
    10.140.113.248    10.140.113.25     HA Partner
disconnected
    Home Port: e0g
    10.140.113.248    10.140.113.215    DR Partner
disconnected
    Home Port: e0g
    10.140.113.248    10.140.113.217    DR Auxiliary
disconnected
24 entries were displayed.

```

Related Links

- [metrocluster configuration-settings dr-group create](#)
- [metrocluster configuration-settings connection connect](#)
- [metrocluster configuration-settings dr-group show](#)

metrocluster configuration-settings interface delete

Delete a MetroCluster interface

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

Description

The `metrocluster configuration-settings interface delete` command deletes the network logical interface that was configured on a network port provisioned for MetroCluster.

This command cannot be run if the [metrocluster configuration-settings connection connect](#) command has set up the connections between the nodes in a DR group. The [metrocluster configuration-settings connection](#)

`disconnect` command must then be run to remove the connections.

The `metrocluster configuration-settings` commands are run in the following order to remove the MetroCluster over IP configuration:

- `metrocluster configuration-settings connection disconnect` ,
- `metrocluster configuration-settings interface delete` ,
- `metrocluster configuration-settings dr-group delete` .

Parameters

-cluster-name <Cluster name> - Cluster Name

Use this parameter to specify the name of the local or partner cluster.

-home-node <text> - Home Node

Use this parameter to specify the home node in the cluster which hosts the interface.

-home-port {<netport>|<ifgrp>} - Home Port

Use this parameter to specify the home port provisioned for MetroCluster.

Examples

The following example shows the deletion of interface in a MetroCluster setup:

```
clusA::> metrocluster configuration-settings interface delete -cluster
-name clusA -home-node A1 -home-port e0f
[Job 271] Job succeeded: Interface Delete is successful.

clusA::> metrocluster configuration-settings interface show
DR
Config
Group Cluster Node      Network Address Netmask      Gateway
State
-----
-----
1      clusA A1
      Home Port: e0g
      10.140.113.215  255.255.192.0  -
completed
      A2
      Home Port: e0f
      10.140.113.216  255.255.192.0  -
completed
      Home Port: e0g
      10.140.113.217  255.255.192.0  -
completed
      clusB B2
```

```

Home Port: e0f
10.140.113.249 255.255.192.0 -
completed

Home Port: e0g
10.140.113.25 255.255.192.0 -
completed

B1
Home Port: e0f
10.140.113.218 255.255.192.0 -
completed

Home Port: e0g
10.140.113.248 255.255.192.0 -
completed
7 entries were displayed.

```

```
clusA::> metrocluster configuration-settings show-status
```

Cluster	Node	Configuration Settings
clusA	A1	ready for next interface
create	A2	ready for connection connect
clusB	B1	ready for connection connect
	B2	ready for connection connect

4 entries were displayed.

Output of the command after deleting all the interfaces:

```
clusA::> metrocluster configuration-settings interface show
No interfaces exist.
```

```
clusA::> metrocluster configuration-settings show-status
```

Cluster	Node	Configuration Settings
clusA	A1	ready for interface create
	A2	ready for interface create
clusB	B1	ready for interface create
	B2	ready for interface create

4 entries were displayed.

Related Links

- [metrocluster configuration-settings connection connect](#)
- [metrocluster configuration-settings connection disconnect](#)
- [metrocluster configuration-settings dr-group delete](#)

metrocluster configuration-settings interface show

Display the network logical interfaces provisioned for MetroCluster

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

Description

The `metrocluster configuration-settings interface show` command displays the network logical interfaces that were provisioned for MetroCluster.

Parameters

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

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

| [-instance] }

If this parameter is specified, the command displays detailed information about all entries.

[-dr-group-id <integer>] - DR Group ID

If this parameter is specified, the command displays information for the matching DR group.

[-cluster-uuid <UUID>] - Cluster UUID

If this parameter is specified, the command displays information for the matching cluster specified by uuid.

[-cluster <Cluster name>] - Cluster Name

If this parameter is specified, the command displays information for the matching cluster..

[-node-uuid <UUID>] - Node UUID

If this parameter is specified, the command displays information for the matching nodes uuid.

[-node <text>] - Node Name

If this parameter is specified, the command displays information for the matching nodes.

[-home-port {<netport>|<ifgrp>}] - Home Port

If this parameter is specified, all interfaces with home-port set to this value are displayed.

[-address <IP Address>] - Network Address

If this parameter is specified, the command displays information for the matching network address.

[-netmask <Contiguous IP Mask>] - Netmask

If this parameter is specified, all interfaces with netmask set to this value are displayed.

[-gateway <IP Address>] - Gateway

If this parameter is specified, all interfaces with gateway set to this value are displayed.

[-config-state <text>] - Configuration State

If this parameter is specified, all interfaces with this field set to the specified value are displayed.

[-config-error-info <text>] - Configuration Error Information

If this parameter is specified, all interfaces with this field set to the specified value are displayed.

Examples

The following example illustrates display of logical interfaces configured in a four-node MetroCluster setup:

```

clusA::> metrocluster configuration-settings interface show
DR
Config
Group Cluster Node      Network Address Netmask      Gateway
State
-----
1      clusA A1
      Home Port: e0f
      10.140.113.214 255.255.192.0 -
completed
      Home Port: e0g
      10.140.113.215 255.255.192.0 -
completed
      A2
      Home Port: e0f
      10.140.113.216 255.255.192.0 -
completed
      Home Port: e0g
      10.140.113.217 255.255.192.0 -
completed
      clusB B2
      Home Port: e0f
      10.140.113.249 255.255.192.0 -
completed
      Home Port: e0g
      10.140.113.25  255.255.192.0 -
completed
      B1
      Home Port: e0f
      10.140.113.218 255.255.192.0 -
completed
      Home Port: e0g
      10.140.113.248 255.255.192.0 -
completed
8 entries were displayed.

```

metrocluster interconnect commands

metrocluster interconnect adapter modify

Modify MetroCluster interconnect adapter settings

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

Description

The `metrocluster interconnect adapter modify` command enables you to modify settings of the MetroCluster interconnect adapter.

Parameters

-node {<nodename>|local} - Node Name

This parameter specifies the node name.

-is-ood-enabled {true|false} - Is Out-of-Order Delivery Enabled?

This parameter specifies the out-of-order delivery setting on the adapter.

Examples

The following example enables out-of-order delivery for the port 'fcvi_device_0' on the node 'clusA-01':

```
clusA::*> metrocluster interconnect adapter modify -node clusA-01 -adapter
-port-name fcvi_device_0 -is-ood-enabled true
```

metrocluster interconnect adapter show

Display MetroCluster interconnect adapter information

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

Description

The `metrocluster interconnect adapter show` command displays interconnect adapter information for the nodes in a MetroCluster configuration.

This command displays the following details about the local node and the HA partner node:

- Node: This field specifies the name of the node in the cluster.
- Adapter Name: This field specifies the name of the interconnect adapter.
- Adapter Type: This field specifies the type of the interconnect adapter.
- Link Status: This field specifies the physical link status of the interconnect adapter.
- Is OOD Enabled: This field specifies the out-of-order delivery status of the interconnect adapter.
- IP Address: This field specifies the IP address assigned to the interconnect adapter.
- Port Number: This field specifies the port number of the interconnect adapter.

Parameters

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

If you specify the `-fields <fieldname>, ...` parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

| **[-connectivity]**

Displays the connectivity information from all the interconnect adapters to the connected nodes.

| **[-switch]**

Displays details of switches connected to all the interconnect adapters.

| **[-connectivity-hidden]**

Displays additional connectivity information (IP address, Area ID, Port ID) from all the interconnect adapters to the connected nodes.

| **[-instance] }**

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

[-node {<nodename>|local}] - Node Name

Displays information only about the interconnect adapters that are hosted by the specified node.

[-adapter <text>] - Adapter

Displays information only about the interconnect adapters that match the specified name.

[-port-name <text>] - Port Name

Displays information only about the interconnect adapters that host the specified port name.

[-type <text>] - Adapter Type

Displays information only about the interconnect adapters that match the specified adapter type.

[-physical-status <text>] - Physical Status

Displays information only about the interconnect adapters that match the specified physical status.

[-wwn <text>] - Adapter Port World Wide Name

Displays information only about the interconnect adapters that match the specified world wide name.

[-address <text>] - IP Address

Displays information only about the interconnect adapters that match the specified IP address.

[-firmware-version <text>] - Firmware Version

Displays information only about the interconnect adapters that match the specified firmware version.

[-link-speed <text>] - Link Speed

Displays information only about the interconnect adapters that match the specified link speed.

[-link-speed-neg-type <text>] - Link Speed Negotiation Type

Displays information only about the interconnect adapters that match the specified negotiated link speed type.

[-switch-name <text>] - Switch Name

Displays information only about the interconnect adapters that are connected to the specified switch.

[-switch-model <text>] - Switch Model

Displays information only about the interconnect adapters that are connected to the switch with the specified model.

[-switch-wwn <text>] - Switch WWName

Displays information only about the interconnect adapters that are connected to the switch with the specified world wide name.

[-switch-vendor <text>] - Switch Vendor

Displays information only about the interconnect adapters that are connected to the switch with the specified vendor.

[-switch-status <text>] - Switch Status

Displays information only about the interconnect adapters that are connected to the switch with the specified operational status.

[-switch-port-number <text>] - Switch Port Number

Displays information only about the interconnect adapters that are connected to the switch with the specified port number.

[-switch-port-wwpn <text>] - Switch Port WWPN

Displays information only about the interconnect adapters that are connected to the switch with the specified word wide port name.

[-remote-adapter-name-list <text>,...] - Remote Adapter Name List

Displays information only about the interconnect adapters that are connected to the specified remote adapters.

[-remote-adapter-wwn-list <text>,...] - Remote Adapter WWName List

Displays information only about the interconnect adapters that are connected to the remote adapters with the specified world wide names.

[-remote-adapter-address-list <text>,...] - Remote Adapter IP Address List

Displays information only about the interconnect adapters that are connected to the remote adapters with the specified IP addresses.

[-remote-adapter-port-id-list <Hex Integer>,...] - Remote Adapter Port ID List

Displays information only about the interconnect adapters that are connected to the remote adapters with the specified port IDs.

[-remote-adapter-domain-id-list <integer>,...] - Remote Adapter Domain ID List

Displays information only about the interconnect adapters that are connected to the remote adapters with the specified domain IDs.

[-remote-adapter-area-id-list <integer>,...] - Remote Adapter Area ID List

Displays information only about the interconnect adapters that are connected to the remote adapters with the specified Area IDs.

[-remote-partner-system-id-list <integer>,...] - Remote Partner System ID List

Displays information only about the interconnect adapters that are connected to the remote nodes with the

specified System IDs.

[-remote-partner-name-list {<nodename>|local}] - Remote Partner Name List

Displays information only about the interconnect adapters that are connected to the specified remote nodes.

[-is-ood-enabled {true|false}] - Is Out-of-Order Delivery Enabled?

Displays information only about the interconnect adapters that match the specified out-of-order delivery setting.

Examples

The following example shows the output of the command during normal operation (neither cluster is in switchover state):

```
clusA::> metrocluster interconnect adapter show
```

Node	Adapter	Type	Link Status	Is OOD Enabled?	IP Address	Port
clusA-01	cxgb3_0	iWARP	Up	false	10.0.1.1	c0a
clusA-01	cxgb3_0	iWARP	Down	false	10.0.2.1	c0b
clusA-01	fcvi_device_0	FC-VI	Up	false	1.0.0.1	1a
clusA-01	fcvi_device_1	FC-VI	Up	false	2.0.0.3	1b
clusA-02	cxgb3_0	iWARP	Up	false	10.0.1.2	c0a
clusA-02	cxgb3_0	iWARP	Down	false	10.0.2.2	c0b
clusA-02	fcvi_device_0	FC-VI	Up	false	1.0.1.1	1a
clusA-02	fcvi_device_1	FC-VI	Up	false	2.0.1.3	1b

The following example shows the output of the command after MetroCluster switchover is performed:

```
clusA::> metrocluster interconnect adapter show
```

Node	Adapter	Type	Link Status	Is OOD Enabled?	IP Address	Port
clusA-01	cxgb3_0	iWARP	Up	false	10.0.1.1	c0a
clusA-01	cxgb3_0	iWARP	Down	false	10.0.2.1	c0b
clusA-01	fcvi_device_0	FC-VI	Down	false	1.0.0.1	1a
clusA-01	fcvi_device_1	FC-VI	Down	false	2.0.0.3	1b
clusA-02	cxgb3_0	iWARP	Up	false	10.0.1.2	c0a
clusA-02	cxgb3_0	iWARP	Down	false	10.0.2.2	c0b
clusA-02	fcvi_device_0	FC-VI	Down	false	1.0.1.1	1a
clusA-02	fcvi_device_1	FC-VI	Down	false	2.0.1.3	1b

The following example shows the output of the command with connectivity field during normal operation (neither cluster is in switchover state):

```
clusA::> metrocluster interconnect adapter show -connectivity -node local
-type FC-VI
Adapter Name: fcvi_device_0
                WWName: 21:00:00:24:ff:32:01:68
                PortNo: 1a

Remote Adapters:
Adapter Name Partner Node Name World Wide Name          PortId
-----
fcvi_device_0
                clusA-01          21:00:00:24:ff:32:01:80  65536
fcvi_device_0
                clusB-01          21:00:00:24:ff:32:01:54 131072
fcvi_device_0
                clusB-02          21:00:00:24:ff:32:01:60 131328

Adapter Name: fcvi_device_1
                WWName: 21:00:00:24:ff:32:01:69
                PortNo: 1b

Remote Adapters:
Adapter Name Partner Node Name World Wide Name          PortId
-----
fcvi_device_1
                clusA-01          21:00:00:24:ff:32:01:81 196608
fcvi_device_1
                clusB-01          21:00:00:24:ff:32:01:55 262144
fcvi_device_1
                clusB-02          21:00:00:24:ff:32:01:61 262400
```

The following example shows the output of the command with connectivity field after MetroCluster switchover is performed.

```

clusA::> metrocluster interconnect adapter show -connectivity -node local
-type FC-VI
Adapter Name: fcvi_device_0
                WWName: 21:00:00:24:ff:32:01:68
                PortNo: 1a

Remote Adapters:
Adapter Name Partner Node Name World Wide Name          PortId
-----
fcvi_device_0
                clusA-01          21:00:00:24:ff:32:01:80    65536
Adapter Name: fcvi_device_1
                WWName: 21:00:00:24:ff:32:01:69
                PortNo: 1b

Remote Adapters:
Adapter Name Partner Node Name World Wide Name          PortId
-----
fcvi_device_1
                clusA-01          21:00:00:24:ff:32:01:81    196608

```

metrocluster interconnect mirror show

Display MetroCluster interconnect mirror information

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

Description

The `metrocluster interconnect mirror show` command displays NVRAM mirror information for the nodes configured in a MetroCluster.

This command displays the following details about the local node and the HA partner node:

- **Node:** This field specifies the name of the node in the cluster.
- **Partner Name:** This field specifies the name of the partner node.
- **Partner Type:** This field specifies the type of the partner.
- **Mirror Admin Status:** This field specifies the administrative status of the NVRAM mirror between partner nodes.
- **Mirror Oper Status:** This field specifies the operational status of the NVRAM mirror between partner nodes.
- **Adapter:** This field specifies the name of the interconnect adapter used for NVRAM mirroring.
- **Type:** This field specifies the type of the interconnect adapter used for NVRAM mirroring.
- **Status:** This field specifies the physical status of the interconnect adapter used for NVRAM mirroring.

Parameters

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

If you specify the `-fields <fieldname>, ...` parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

| [-instance] }

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

[-node {<nodename>|local}] - Node Name

If this parameter is specified, mirror details of the specified node are displayed.

[-partner-type {HA|DR|AUX}] - Partner Type

If this parameter is specified, mirror details of the specified partner type are displayed.

[-adapter <text>] - Adapter

If this parameter is specified, mirror details of the specified adapter are displayed.

[-type <text>] - Adapter Type

If this parameter is specified, mirror details of the specified adapter type are displayed.

[-status <text>] - Status

If this parameter is specified, mirror details of the adapter with the specified status are displayed.

[-mirror-oper-status {unknown|online|offline}] - Mirror Operational Status

If this parameter is specified, only mirror details with the specified operational status are displayed.

[-partner-name <text>] - Partner Name

If this parameter is specified, mirror details of the specified partner are displayed.

[-mirror-admin-status {enabled|disabled}] - Mirror Administrative Status

If this parameter is specified, only mirror details with the specified administrative status are displayed.

Examples

The following example shows the output of the command during normal operation (neither cluster is in switchover state):

```

clusA::> metrocluster interconnect mirror show

```

Node	Partner	Name	Partner Type	Mirror		Adapter	Type	Status
				Admin Status	Oper Status			
clusA-01								
	clusA-02		HA	enabled	online			
						cxgb3_0	iWARP	Up
						cxgb3_0	iWARP	Up
	clusB-01		DR	enabled	online			
						fcvi_device_0		
							FC-VI	Up
						fcvi_device_1		
							FC-VI	Up
clusA-02								
	clusA-01		HA	enabled	online			
						cxgb3_0	iWARP	Up
						cxgb3_0	iWARP	Up
	clusB-02		DR	enabled	online			
						fcvi_device_0		
							FC-VI	Up
						fcvi_device_1		
							FC-VI	Up

The following example shows the output of the command after MetroCluster switchover is performed:

```
clusA::> metrocluster interconnect mirror show
```

Node	Partner	Name	Type	Mirror		Adapter	Type	Status
				Admin Status	Oper Status			
clusA-01								
	clusA-02		HA	enabled	online			
						cxgb3_0	iWARP	Up
						cxgb3_0	iWARP	Up
clusB-01								
			DR	disabled	offline			
						fcvi_device_0		
							FC-VI	Up
						fcvi_device_1		
							FC-VI	Up
clusA-02								
	clusA-01		HA	enabled	online			
						cxgb3_0	iWARP	Up
						cxgb3_0	iWARP	Up
clusB-02								
			DR	disabled	offline			
						fcvi_device_0		
							FC-VI	Up
						fcvi_device_1		
							FC-VI	Up

metrocluster interconnect mirror multipath show

Display multipath information

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

Description

The `metrocluster interconnect mirror multipath show` command displays the NVRAM mirror multipath policy for the nodes configured in a MetroCluster.

This command displays the following details about the local node and the HA partner node:

- **Node:** This field specifies the name of the node in the cluster.
- **Multipath Policy:** This field specifies the multipath policy used for NVRAM mirroring.

Parameters

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

If you specify the `-fields <fieldname>, ...` parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

| [-instance] }

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

[-node {<nodename>|local}] - Node Name

If this parameter is specified, mirror details of the specified node are displayed.

[-multipath-policy {no-mp|static-map|dynamic-map|round-robin}] - Multipath Policy

If this parameter is specified, nodes with the specified multipath policy are displayed.

Examples

The following example shows the output of the command:

```
clusA::> metrocluster interconnect mirror multipath show
Node                Multipath Policy
-----
clusA-1             static-map
clusA-2             static-map
```

metrocluster node commands

metrocluster node show

Display MetroCluster node configuration information

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

Description

The `metrocluster node show` command displays configuration information for the nodes in the MetroCluster configuration.

Parameters

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

If you specify the `-fields <fieldname>, ...` parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

| [-partners]

If this option is used the MetroCluster node partnership view will be displayed.

| [-instance] }

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

[-dr-group-id <integer>] - DR Group ID

If this parameter is specified, all nodes belonging to the specified DR group are displayed.

[-cluster <Cluster name>] - Cluster Name

If this parameter is specified, all nodes belonging to the specified cluster are displayed.

[-node <Node name>] - Node Name

If this parameter is specified, the specified node is displayed.

[-ha-partner <Node name>] - HA Partner Name

If this parameter is specified, the node with the specified HA partner is displayed.

[-dr-cluster <Cluster name>] - DR Cluster Name

If this parameter is specified, all nodes belonging to the specified cluster are displayed.

[-dr-partner <Node name>] - DR Partner Name

If this parameter is specified, the node with the specified DR partner is displayed.

[-dr-auxiliary <Node name>] - DR Auxiliary Name

If this parameter is specified, the node with the specified DR auxiliary partner is displayed.

[-node-uuid <UUID>] - Node UUID

If this parameter is specified, the node with the specified Uuid is displayed.

[-ha-partner-uuid <UUID>] - HA Partner UUID

If this parameter is specified, the nodes with the specified HA partner is displayed.

[-dr-partner-uuid <UUID>] - DR Partner UUID

If this parameter is specified, the node with the specified DR partner is displayed.

[-dr-auxiliary-uuid <UUID>] - DR Auxiliary UUID

If this parameter is specified, the node with the specified DR auxiliary partner is displayed.

[-node-cluster-uuid <UUID>] - Node Cluster UUID

If this parameter is specified, all nodes belonging to the specified cluster are displayed.

[-ha-partner-cluster-uuid <UUID>] - HA Partner Cluster UUID

If this parameter is specified, all nodes whose HA partner belong to the specified cluster are displayed.

[-dr-partner-cluster-uuid <UUID>] - DR Partner Cluster UUID

If this parameter is specified, all nodes whose DR partner belong to the specified cluster are displayed.

[-dr-auxiliary-cluster-uuid <UUID>] - DR Auxiliary Cluster UUID

If this parameter is specified, all nodes whose DR auxiliary partner belong to the specified cluster are displayed.

[-node-systemid <integer>] - Node System ID

If this parameter is specified, all nodes with the specified system ID are displayed.

[-ha-partner-systemid <integer>] - HA Partner System ID

If this parameter is specified, all nodes with an HA partner with the specified system ID are displayed.

[-dr-partner-systemid <integer>] - DR Partner System ID

If this parameter is specified, all nodes with a DR partner with the specified system ID are displayed.

[-dr-auxiliary-systemid <integer>] - DR Auxiliary System ID

If this parameter is specified, all nodes with a DR auxiliary partner with the specified system ID are displayed.

[-dr-mirroring-state <text>] - State of DR Mirroring Config

If this parameter is specified, all nodes with this field set to the specified value are displayed. This field specifies if the NVRAM mirroring to the DR partner is enabled through the [metrocluster configure](#) command. This field needs to be set to "enabled" for the DR mirroring to be active.

[-configuration-state <text>] - Configuration State of Node

If this parameter is specified, all nodes with this field set to the specified value are displayed.

[-additional-configuration-info <text>] - Additional Configuration Info

If this parameter is specified, all nodes with this field set to the specified value are displayed.

[-dr-operation-state <text>] - DR Operation State

If this parameter is specified, all nodes with this field set to the specified value are displayed.

[-dr-operation-time <integer>] - Time to Complete Operation (secs)

If this parameter is specified, all nodes with this field set to the specified value are displayed.

[-node-object-limit {on|off}] - Specifies if the Node Object Limits are Enforced

If this parameter is specified, all nodes with this field set to the specified value are displayed.

[-node-ha-partner <text>] - Node and its HA Partner

If this parameter is specified, all nodes with this field set to the specified value are displayed.

[-automatic-uso {true|false}] - Automatic USO

If this parameter is specified, all nodes with this field set to the specified value are displayed.

Examples

The following example shows the output of the command before the MetroCluster configuration is done:

```

clusA::> metrocluster node show
DR
Group Cluster Node          Configuration State          DR
Mirroring Mode
-----
-----
-      clusA  clusA-01          ready to configure -      -
                clusA-02          ready to configure -      -
                clusA-03          ready to configure -      -
                clusA-04          ready to configure -      -

4 entries were displayed.
clusA::> metrocluster node show -partners
Node (HA Partner) DR Partner (DR Auxiliary)
-----
-----
Cluster:                clusA -
                clusA-01 (-) - (-)
                clusA-02 (-) - (-)
                clusA-03 (-) - (-)
                clusA-04 (-) - (-)

4 entries were displayed.

```

The following example shows the output of the command when some DR groups in the MetroCluster configuration are not yet configured:

```

clusA::> metrocluster node show
DR
Group Cluster Node Configuration State DR Mirroring Mode
-----
-----
-----
- clusA clusA-03 ready to configure - -
      clusA-04 ready to configure - -
1 clusA clusA-01 configured enabled normal
      clusA-02 configured enabled normal
  clusB clusB-01 configured enabled normal
      clusB-02 configured enabled normal
6 entries were displayed.
clusA::> metrocluster node show -partners
Node (HA Partner) DR Partner (DR Auxiliary)
-----
-----
Cluster: clusA -
          clusA-03 (-) - (-)
          clusA-04 (-) - (-)
Cluster: clusA clusB
          clusA-01 (clusA-02) clusB-01 (clusB-02)
          clusA-02 (clusA-01) clusB-02 (clusB-01)
Cluster: clusB clusA
          clusB-01 (clusB-02) clusA-01 (clusA-02)
          clusB-02 (clusB-01) clusA-02 (clusA-01)
6 entries were displayed.

```

The following example shows the output of the command after after all DR groups in the MetroCluster configuration are configured:

```

clusA::> metrocluster node show
DR
Group Cluster Node Configuration State DR Mirroring Mode
-----
-----
-----
1 clusA clusA-01 configured enabled normal
  clusA clusA-02 configured enabled normal
  clusB clusB-01 configured enabled normal
  clusB clusB-02 configured enabled normal
2 clusA clusA-03 configured enabled normal
  clusA clusA-04 configured enabled normal
  clusB clusB-03 configured enabled normal
  clusB clusB-04 configured enabled normal

8 entries were displayed.
clusA::> metrocluster node show -partners
Node (HA Partner) DR Partner (DR Auxiliary)
-----
-----
Cluster: clusA clusB
clusA-01 (clusA-02) clusB-01 (clusB-02)
clusA-02 (clusA-01) clusB-02 (clusB-01)
Cluster: clusB clusA
clusB-01 (clusB-02) clusA-01 (clusA-02)
clusB-02 (clusB-01) clusA-02 (clusA-01)
Cluster: clusA clusB
clusA-03 (clusA-04) clusB-03 (clusB-04)
clusA-04 (clusA-03) clusB-04 (clusB-03)
Cluster: clusB clusA
clusB-03 (clusB-04) clusA-03 (clusA-04)
clusB-04 (clusB-03) clusA-04 (clusA-03)

8 entries were displayed.

```

Related Links

- [metrocluster configure](#)

metrocluster operation commands

metrocluster operation show

Display details of the last MetroCluster operation

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

Description

The `metrocluster operation show` command displays information about the most recent MetroCluster operation run on the local cluster.

This command will display information about all MetroCluster commands except for the commands in the `metrocluster check` directory. This command will not display any information after MetroCluster has been completely unconfigured using the `metrocluster unconfigure` command.

Examples

The following example shows the output of `metrocluster operation show` after running a [metrocluster configure](#) command was successful:

```
clusA::> metrocluster operation show
      Operation: configure
      State: successful
Start time: 2/15/2013 18:22:46
      End time: 2/15/2013 18:25:18
      Errors: -
```

Related Links

- [metrocluster configure](#)

metrocluster operation history show

Display details of all MetroCluster operations

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

Description

The `metrocluster operation history show` command displays information about all the MetroCluster operations run on the local cluster.

This command will display information about all MetroCluster commands except for the commands in the `metrocluster check` directory. This command will not display any information after MetroCluster has been completely unconfigured using the `metrocluster unconfigure` command.

Parameters

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

If you specify the `-fields <fieldname>, ...` parameter, the command output also includes the specified field or fields. You can use `'-fields ?'` to display the fields to specify.

| [-instance] }

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

[`-operation-uuid <UUID>`] - Identifier for the Operation

This is the UUID of the operation. If this parameter is specified, only the operation with this UUID is displayed.

[`-cluster <Cluster name>`] - Cluster Where the Command Was Run

This is the name of the cluster where the command was run. If this parameter is specified, only the operations that were run in this cluster are displayed.

[`-node-name <Node name>`] - Node Where the Command Was run

This is the name of the node where the command was run. If this parameter is specified, only the operations that were run on this node are displayed.

[`-operation <MetroCluster Operation Name>`] - Name of the Operation

This is the name of the operation. If this parameter is specified, only the operations with this name are displayed.

[`-start-time <MM/DD/YYYY HH:MM:SS>`] - Start Time

This is the time the operation started execution. If this parameter is specified, only the operations that were started at this time are displayed.

[`-state <MetroCluster Operation state>`] - State of the Operation

This is the state of the operation. If this parameter is specified, only the operations that are in this state are displayed.

[`-end-time <MM/DD/YYYY HH:MM:SS>`] - End Time

This is the time the operation completed. If this parameter is specified, only the operations that completed at this time are displayed.

[`-error-list <text>,...`] - Error List For the Operation

This is the list of errors that were encountered during an operation's execution. If this parameter is specified, only the operations that have the matching errors are displayed.

[`-job-id <integer>`] - Identifier for the Job

This is the job id for the operation. If this parameter is specified, only the operation that has the matching job id displayed.

Examples

The following example shows the output of `metrocluster operation history show` after some MetroCluster operations have been performed:

```
clusA::> metrocluster operation history show
Operation          State          Start time      End time
-----
configure          successful     2/15/2013 18:22:46
                   2/15/2013 18:25:18
configure          failed        2/15/2013 18:13:45
                   2/15/2013 18:13:45
2 entries were displayed.
```

metrocluster vservers commands

metrocluster vservers recover-from-partial-switchback

Recover vservers from partial switchback

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

Description

The `metrocluster vservers recover-from-partial-switchback` command executes the necessary steps needed for a Vserver to be in healthy state after partial completion of the Switchback.

Parameters

Examples

```
cluster::> metrocluster vservers recover-from-partial-switchback
```

metrocluster vservers recover-from-partial-switchover

Recover vservers from partial switchover

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

Description

The `metrocluster vservers recover-from-partial-switchover` command executes the necessary steps needed for a Vserver to be in healthy state after partial completion of the Switchover.

Parameters

Examples

```
cluster::> metrocluster vservers recover-from-partial-switchover
```


metrocluster vserver resync

Resynchronize Vserver with it's partner Vserver

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

Description

The `metrocluster vserver resync` command resynchronizes the Vserver with it's partner Vserver

Parameters

-cluster <Cluster name> - Cluster Name

Name of the cluster where the Vserver belongs

-vserver <vserver> - Vserver

Name of the Vserver to be resynchronized

Examples

```
cluster::> metrocluster vserver resync -cluster clus1 -vserver vs1
```

metrocluster vserver show

Display MetroCluster Vserver relationships

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

Description

The `metrocluster vserver show` command displays configuration information for all pairs of Vservers in MetroCluster.

Parameters

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

The command output includes the specified field or fields

| [-creation-time]

Shows the last configuration modification time on the Vserver

| [-instance] }

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

[-cluster <Cluster name>] - Cluster Name

Name of the cluster where the Vserver belongs

[-vserver <vserver>] - Vserver

Name of the Vserver

[-partner-vserver <vserver>] - Partner Vserver

Name of the partner Vserver

[-configuration-state {healthy|unhealthy|degraded|pending-setup|syncing|replication-paused|pending-switchback}] - Configuration State

Configuration states include:

- *healthy*
- *unhealthy*
- *degraded* indicates that Vservers are not in sync
- *syncing* indicates that the Vserver configuration is being synchronized
- *replication-paused* indicates that the configuration replication was manually paused
- *pending-setup* indicates that partner Vserver creation is pending

[-corrective-action <text>] - Corrective Action

Corrective action which can be followed to successfully create the partner Vserver

[-creation-time-of-last-applied-change <MM/DD/YYYY HH:MM:SS>] - Creation Time on the source

Last configuration modification time on the Vserver

[-out-of-sync <true>] - Is out of sync

Indicates that the Vserver configuration replication is not in sync

[-config-resume-time <MM/DD/YYYY HH:MM:SS>] - configuration Resume Time

Displays the resume time of the Vserver configuration replication

Examples

The following example shows the output of the command when partner Vservers are created

```

clusA::> metrocluster vserver show
Cluster: clusA
Configuration
Vserver
State
Partner
Vserver
-----
clusA
healthy
vs1
Cluster: clusB
Configuration
Vserver
State
Partner
Vserver
-----
clusB
healthy
clusA
3 entries were displayed.

```

The following example shows the output of the command when the partner Vserver creation is pending

```

clusA::> metrocluster vserver show
Cluster: clusA
Configuration
Vserver
State
Partner
Vserver
-----
clusA
healthy
vs1
pending-setup
Corrective Action: Create Ipspace ips1 on the partner cluster.
2 entries were displayed.

```

Copyright Information

Copyright © 2022 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

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.