



# **metrocluster check commands**

## **ONTAP 9.6 commands**

NetApp  
August 29, 2024

# Table of Contents

- metrocluster check commands . . . . . 1
  - metrocluster check disable-periodic-check . . . . . 1
  - metrocluster check enable-periodic-check . . . . . 1
  - metrocluster check run . . . . . 1
  - metrocluster check show . . . . . 3
  - metrocluster check aggregate show . . . . . 6
  - metrocluster check cluster show . . . . . 9
  - metrocluster check config-replication show-aggregate-eligibility . . . . . 13
  - metrocluster check config-replication show-capture-status . . . . . 14
  - metrocluster check config-replication show . . . . . 15
  - metrocluster check connection show . . . . . 16
  - metrocluster check lif repair-placement . . . . . 20
  - metrocluster check lif show . . . . . 21
  - metrocluster check node show . . . . . 23
  - metrocluster check volume show . . . . . 30

# 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 (privilege: advanced)**

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 the [metrocluster check run](#) command.

This command displays the high-level verification results for each of the components. If there are 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
connections    ok
s were displayed.
```

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

```
clusA::> metrocluster check show
Checked On: 4/9/2014 20:11:46
t          Result
-----
nodes      warning
clusters   ok
lifs       ok
config-replication ok
aggregates ok
connections 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:
Time of Check: 4/9/2014 20:12:36
          Name of the Component: connections
          Result of the Check: ok
          Additional Information/Recovery Steps:
6 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 consistent.
- `external-key-management`: This check verifies that the External Key Management configurations are consistent.

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|external-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
	external-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
	external-key-management	ok

```
14 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: external-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:
Type of Check: external-key-management
Cluster Name: clusB
Result of the Check: ok
```

Additional Information/Recovery Steps:  
14 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 connection show

Display the check results of connections for nodes in a MetroCluster over IP configuration

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

### Description

The `metrocluster check connection show` command displays the check results of connections for nodes in a MetroCluster over IP configuration.

### 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.

**[-check-result {ok|warning|not-run|not-applicable}] - Check Connection Result**

If this parameter is specified, the command displays information for the matching check-result.

**[-check-ping-error-info <text>] - Check Connection Ping Error Info**

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

**[-check-mtu-size-error-info <text>] - Check Connection MTU Size Error Info**

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

**[-check-storage-error-info <text>] - Check Connection Storage Error Info**

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

## Examples

The following example shows the output of the `metrocluster check connection show` command:

```

clusA::> metrocluster check connection show
DR
Group Cluster Node      Source      Destination
Network Address Network Address Partner Type Config
State
-----
1      cluster-A
      node-A1
      Home Port: e0f
      10.140.113.214  10.140.113.216  HA Partner
completed
      Check Result: ok
      Home Port: e0f
      10.140.113.214  10.140.113.218  DR Partner
completed
      Check Result: ok
      Home Port: e0f
      10.140.113.214  10.140.113.249  DR Auxiliary
completed
      Check Result: ok
      Home Port: e0g
      10.140.113.215  10.140.113.217  HA Partner
completed
      Check Result: ok
      Home Port: e0g
      10.140.113.215  10.140.113.248  DR Partner
completed
      Check Result: ok
      Home Port: e0g
      10.140.113.215  10.140.113.25   DR Auxiliary
completed
      Check Result: ok
      node-A2
      Home Port: e0f
      10.140.113.216  10.140.113.214  HA Partner
completed
      Check Result: ok
      Home Port: e0f
      10.140.113.216  10.140.113.249  DR Partner
completed
      Check Result: ok
      Home Port: e0f
      10.140.113.216  10.140.113.218  DR Auxiliary
completed
      Check Result: ok
      Home Port: e0g

```

```

10.140.113.217 10.140.113.215 HA Partner
completed
    Check Result: ok
    Home Port: e0g
10.140.113.217 10.140.113.25 DR Partner
completed
    Check Result: ok
    Home Port: e0g
10.140.113.217 10.140.113.248 DR Auxiliary
completed
    Check Result: ok
cluster-B
    node-B1
        Home Port: e0f
10.140.113.218 10.140.113.249 HA Partner
completed
    Check Result: ok
    Home Port: e0f
10.140.113.218 10.140.113.214 DR Partner
completed
    Check Result: ok
    Home Port: e0f
10.140.113.218 10.140.113.216 DR Auxiliary
completed
    Check Result: ok
    Home Port: e0g
10.140.113.248 10.140.113.25 HA Partner
completed
    Check Result: ok
    Home Port: e0g
10.140.113.248 10.140.113.215 DR Partner
completed
    Check Result: ok
    Home Port: e0g
10.140.113.248 10.140.113.217 DR Auxiliary
completed
    Check Result: ok
    node-B2
        Home Port: e0f
10.140.113.249 10.140.113.218 HA Partner
completed
    Check Result: ok
    Home Port: e0f
10.140.113.249 10.140.113.216 DR Partner
completed
    Check Result: ok

```

```

                Home Port: e0f
                10.140.113.249  10.140.113.214  DR Auxiliary
completed
                Check Result: ok
                Home Port: e0g
                10.140.113.25   10.140.113.248  HA Partner
completed
                Check Result: ok
                Home Port: e0g
                10.140.113.25   10.140.113.217  DR Partner
completed
                Check Result: ok
                Home Port: e0g
                10.140.113.25   10.140.113.215  DR Auxiliary
completed
                Check Result: ok
24 entries were displayed.

```

## 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
node-object-limit ok
automatic-uso ok

clusB-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

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 check volume show

Show results of the MetroCluster check for volumes

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

### Description

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

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

- `unmirrored-flexgroups`: This check looks for flexgroups residing on unmirrored aggregates.
- `mixed-flexgroups`: This check looks for flexgroups residing on a mix of mirrored and unmirrored aggregates.

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.

**[-vserver <vserver name>] - Vserver Name**

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

**[-volume <volume name>] - Volume Name**

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

**[-check <MetroCluster Volume 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.

**[-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 volume show
```

```
Last Checked On: 7/25/2018 10:04:07
```

Vserver	Volume	Check
Result		
-----		
vs1	unMirr	unmirrored-volumes
warning		
vs2	vs2UnMirrA	unmirrored-volumes
warning		

2 entries were displayed.

```
clusA::*> metrocluster check volume show -instance
```

```
Vserver Name: vs1
```

```
Volume Name: unMirr
```

```
Type of Check: unmirrored-volumes
```

```
Result of the Check: warning
```

```
Additional Information/Recovery Steps: FlexGroup "unMirr" resides on unmirrored aggregates. Parts of the FlexGroup may not be available after an un-planned switchover.
```

```
Vserver Name: vs2
```

```
Volume Name: vs2UnMirrA
```

```
Type of Check: unmirrored-volumes
```

```
Result of the Check: warning
```

```
Additional Information/Recovery Steps: FlexGroup "vs2UnMirrA" resides on unmirrored aggregates. Parts of the FlexGroup may not be available after an un-planned switchover.
```

```
2 entries were displayed.
```

```
clusA::>
```

## Related Links

- [metrocluster check run](#)

## Copyright information

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

## Trademark information

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