



# Configuring the clusters into a MetroCluster configuration

## ONTAP MetroCluster

NetApp  
June 14, 2021

# Table of Contents

- Configuring the clusters into a MetroCluster configuration ..... 1
  - Configuring intercluster LIFs ..... 1
  - Mirroring the root aggregates ..... 9
  - Implementing the MetroCluster configuration ..... 10
  - Creating a mirrored data aggregate on each node ..... 12
  - Configuring FC-to-SAS bridges for health monitoring ..... 13
  - Moving a metadata volume in MetroCluster configurations ..... 14
  - Checking the MetroCluster configuration ..... 17

# Configuring the clusters into a MetroCluster configuration

You must peer the clusters, mirror the root aggregates, create a mirrored data aggregate, and then issue the command to implement the MetroCluster operations.

## Configuring intercluster LIFs

You must create intercluster LIFs on ports used for communication between the MetroCluster partner clusters. You can use dedicated ports or ports that also have data traffic.

### Configuring intercluster LIFs on dedicated ports

You can configure intercluster LIFs on dedicated ports. Doing so typically increases the available bandwidth for replication traffic.

#### Steps

1. List the ports in the cluster:

```
network port show
```

For complete command syntax, see the man page.

The following example shows the network ports in cluster01:

```

cluster01::> network port show

```

(Mbps)		Speed				
Node	Port	IPspace	Broadcast Domain	Link	MTU	Admin/Oper
-----						
cluster01-01						
	e0a	Cluster	Cluster	up	1500	auto/1000
	e0b	Cluster	Cluster	up	1500	auto/1000
	e0c	Default	Default	up	1500	auto/1000
	e0d	Default	Default	up	1500	auto/1000
	e0e	Default	Default	up	1500	auto/1000
	e0f	Default	Default	up	1500	auto/1000
cluster01-02						
	e0a	Cluster	Cluster	up	1500	auto/1000
	e0b	Cluster	Cluster	up	1500	auto/1000
	e0c	Default	Default	up	1500	auto/1000
	e0d	Default	Default	up	1500	auto/1000
	e0e	Default	Default	up	1500	auto/1000
	e0f	Default	Default	up	1500	auto/1000

2. Determine which ports are available to dedicate to intercluster communication:

```
network interface show -fields home-port,curr-port
```

For complete command syntax, see the man page.

The following example shows that ports e0e and e0f have not been assigned LIFs:

```

cluster01::> network interface show -fields home-port,curr-port
vserver lif                home-port curr-port
-----
Cluster cluster01-01_clus1 e0a       e0a
Cluster cluster01-01_clus2 e0b       e0b
Cluster cluster01-02_clus1 e0a       e0a
Cluster cluster01-02_clus2 e0b       e0b
cluster01
  cluster_mgmt             e0c       e0c
cluster01
  cluster01-01_mgmt1       e0c       e0c
cluster01
  cluster01-02_mgmt1       e0c       e0c

```

3. Create a failover group for the dedicated ports:

```
network interface failover-groups create -vserver system_SVM -failover-group failover_group -targets physical_or_logical_ports
```

The following example assigns ports e0e and e0f to the failover group intercluster01 on the system SVMcluster01:

```
cluster01::> network interface failover-groups create -vserver cluster01
-failover-group
intercluster01 -targets
cluster01-01:e0e,cluster01-01:e0f,cluster01-02:e0e,cluster01-02:e0f
```

4. Verify that the failover group was created:

```
network interface failover-groups show
```

For complete command syntax, see the man page.

```
cluster01::> network interface failover-groups show
Vserver          Group          Failover
-----          -
Targets
-----
Cluster
cluster01        Cluster
                  cluster01-01:e0a, cluster01-01:e0b,
                  cluster01-02:e0a, cluster01-02:e0b
cluster01        Default
                  cluster01-01:e0c, cluster01-01:e0d,
                  cluster01-02:e0c, cluster01-02:e0d,
                  cluster01-01:e0e, cluster01-01:e0f
                  cluster01-02:e0e, cluster01-02:e0f
cluster01        intercluster01
                  cluster01-01:e0e, cluster01-01:e0f
                  cluster01-02:e0e, cluster01-02:e0f
```

5. Create intercluster LIFs on the system SVM and assign them to the failover group.

ONTAP version	Command
9.6 and later	<pre>network interface create -vserver system_SVM -lif LIF_name -service-policy default-intercluster -home-node node -home -port port -address port_IP -netmask netmask -failover -group failover_group</pre>

9.5 and earlier	<pre>network interface create -vserver system_SVM -lif LIF_name -role intercluster -home-node node -home-port port -address port_IP -netmask netmask -failover-group failover_group</pre>
-----------------	---

For complete command syntax, see the man page.

The following example creates intercluster LIFs `cluster01_icl01` and `cluster01_icl02` in the failover group `intercluster01`:

```
cluster01::> network interface create -vserver cluster01 -lif
cluster01_icl01 -service-
policy default-intercluster -home-node cluster01-01 -home-port e0e
-address 192.168.1.201
-netmask 255.255.255.0 -failover-group intercluster01

cluster01::> network interface create -vserver cluster01 -lif
cluster01_icl02 -service-
policy default-intercluster -home-node cluster01-02 -home-port e0e
-address 192.168.1.202
-netmask 255.255.255.0 -failover-group intercluster01
```

6. Verify that the intercluster LIFs were created:

**In ONTAP 9.6 and later:**

```
network interface show -service-policy default-intercluster
```

**In ONTAP 9.5 and earlier:**

```
network interface show -role intercluster
```

For complete command syntax, see the man page.

```

cluster01::> network interface show -service-policy default-intercluster
      Logical      Status      Network      Current
Current Is
Vserver      Interface  Admin/Oper  Address/Mask  Node      Port
Home
-----
-----
cluster01
      cluster01_icl01
      up/up      192.168.1.201/24  cluster01-01  e0e
true
      cluster01_icl02
      up/up      192.168.1.202/24  cluster01-02  e0f
true

```

7. Verify that the intercluster LIFs are redundant:

**In ONTAP 9.6 and later:**

```
network interface show -service-policy default-intercluster -failover
```

**In ONTAP 9.5 and earlier:**

```
network interface show -role intercluster -failover
```

For complete command syntax, see the man page.

The following example shows that the intercluster LIFs `cluster01_icl01` and `cluster01_icl02` on the SVMe0e port will fail over to the e0f port.

```

cluster01::> network interface show -service-policy default-intercluster
-failover
      Logical          Home          Failover          Failover
Vserver Interface      Node:Port        Policy            Group
-----
cluster01
      cluster01_icl01 cluster01-01:e0e local-only
intercluster01
                        Failover Targets: cluster01-01:e0e,
                                                cluster01-01:e0f
      cluster01_icl02 cluster01-02:e0e local-only
intercluster01
                        Failover Targets: cluster01-02:e0e,
                                                cluster01-02:e0f

```

## Configuring intercluster LIFs on shared data ports

You can configure intercluster LIFs on ports shared with the data network. Doing so reduces the number of ports you need for intercluster networking.

### Steps

1. List the ports in the cluster:

```
network port show
```

For complete command syntax, see the man page.

The following example shows the network ports in cluster01:



```
cluster01::> network port show
```

(Mbps)							Speed
Node	Port	IPspace	Broadcast Domain	Link	MTU	Admin/Oper	
-----							
cluster01-01							
	e0a	Cluster	Cluster	up	1500	auto/1000	
	e0b	Cluster	Cluster	up	1500	auto/1000	
	e0c	Default	Default	up	1500	auto/1000	
	e0d	Default	Default	up	1500	auto/1000	
cluster01-02							
	e0a	Cluster	Cluster	up	1500	auto/1000	
	e0b	Cluster	Cluster	up	1500	auto/1000	
	e0c	Default	Default	up	1500	auto/1000	
	e0d	Default	Default	up	1500	auto/1000	

## 2. Create intercluster LIFs on the system SVM:

### In ONTAP 9.6 and later:

```
network interface create -vserver system_SVM -lif LIF_name -service-policy default-intercluster -home-node node -home-port port -address port_IP -netmask netmask
```

### In ONTAP 9.5 and earlier:

```
network interface create -vserver system_SVM -lif LIF_name -role intercluster -home-node node -home-port port -address port_IP -netmask *netmask
```

For complete command syntax, see the man page.

The following example creates intercluster LIFs `cluster01_icl01` and `cluster01_icl02`:

```

cluster01::> network interface create -vserver cluster01 -lif
cluster01_icl01 -service-
policy default-intercluster -home-node cluster01-01 -home-port e0c
-address 192.168.1.201
-netmask 255.255.255.0

cluster01::> network interface create -vserver cluster01 -lif
cluster01_icl02 -service-
policy default-intercluster -home-node cluster01-02 -home-port e0c
-address 192.168.1.202
-netmask 255.255.255.0

```

### 3. Verify that the intercluster LIFs were created:

#### In ONTAP 9.6 and later:

```
network interface show -service-policy default-intercluster
```

#### In ONTAP 9.5 and earlier:

```
network interface show -role intercluster
```

For complete command syntax, see the man page.

```

cluster01::> network interface show -service-policy default-intercluster

```

Current Is	Logical	Status	Network	Current
Vserver	Interface	Admin/Oper	Address/Mask	Node
Home				Port
cluster01	cluster01_icl01	up/up	192.168.1.201/24	cluster01-01 e0c
true				
	cluster01_icl02	up/up	192.168.1.202/24	cluster01-02 e0c
true				

### 4. Verify that the intercluster LIFs are redundant:

#### In ONTAP 9.6 and later:

```
network interface show -service-policy default-intercluster -failover
```

In ONTAP 9.6 and later:

In ONTAP 9.5 and earlier:

```
network interface show -role intercluster -failover
```

For complete command syntax, see the man page.

The following example shows that the intercluster LIFs `cluster01_icl01` and `cluster01_icl02` on the `e0c` port will fail over to the `e0d` port.

```
cluster01::> network interface show -service-policy default-intercluster
-failover
          Logical          Home          Failover          Failover
Vserver  Interface          Node:Port        Policy            Group
-----
cluster01
          cluster01_icl01 cluster01-01:e0c  local-only
192.168.1.201/24
                                     Failover Targets: cluster01-01:e0c,
                                                         cluster01-01:e0d
          cluster01_icl02 cluster01-02:e0c  local-only
192.168.1.201/24
                                     Failover Targets: cluster01-02:e0c,
                                                         cluster01-02:e0d
```

## Mirroring the root aggregates

You must mirror the root aggregates to provide data protection.

By default, the root aggregate is created as RAID-DP type aggregate. You can change the root aggregate from RAID-DP to RAID4 type aggregate. The following command modifies the root aggregate for RAID4 type aggregate:

```
storage aggregate modify -aggregate aggr_name -raidtype raid4
```



On non-ADP systems, the RAID type of the aggregate can be modified from the default RAID-DP to RAID4 before or after the aggregate is mirrored.

### Steps

1. Mirror the root aggregate:

```
storage aggregate mirror aggr_name
```

The following command mirrors the root aggregate for controller\_A\_1:

```
controller_A_1::> storage aggregate mirror aggr0_controller_A_1
```

This mirrors the aggregate, so it consists of a local plex and a remote plex located at the remote MetroCluster site.

2. Repeat the previous step for each node in the MetroCluster configuration.

## Implementing the MetroCluster configuration

You must run the `metrocluster configure -refresh true` command to start data protection on the nodes that you have added to a MetroCluster configuration.

You issue the `metrocluster configure -refresh true` command once, on one of the newly added nodes, to refresh the MetroCluster configuration. You do not need to issue the command on each of the sites or nodes.

The `metrocluster configure -refresh true` command automatically pairs the two nodes with the lowest system IDs in each of the two clusters as disaster recovery (DR) partners. In a four-node MetroCluster configuration, there are two DR partner pairs. The second DR pair is created from the two nodes with higher system IDs.

### Steps

1. Refresh the MetroCluster configuration:
  - a. Enter advanced privilege mode:  
`set -privilege advanced`
  - b. Refresh the MetroCluster configuration on one of the new nodes:  
`metrocluster configure -refresh true`

The following example shows the MetroCluster configuration refreshed on both DR groups:

```
controller_A_2::*> metrocluster configure -refresh true  
[Job 726] Job succeeded: Configure is successful.
```

```
controller_A_4::*> metrocluster configure -refresh true  
[Job 740] Job succeeded: Configure is successful.
```

- c. Return to admin privilege mode:  
`set -privilege admin`
2. Verify the networking status on site A:

```
network port show
```

The following example shows the network port usage on a four-node MetroCluster configuration:

```
cluster_A::> network port show
```

Node	Port	IPspace	Broadcast Domain	Link	MTU	Speed (Mbps) Admin/Oper
-----						
controller_A_1						
	e0a	Cluster	Cluster	up	9000	auto/1000
	e0b	Cluster	Cluster	up	9000	auto/1000
	e0c	Default	Default	up	1500	auto/1000
	e0d	Default	Default	up	1500	auto/1000
	e0e	Default	Default	up	1500	auto/1000
	e0f	Default	Default	up	1500	auto/1000
	e0g	Default	Default	up	1500	auto/1000
controller_A_2						
	e0a	Cluster	Cluster	up	9000	auto/1000
	e0b	Cluster	Cluster	up	9000	auto/1000
	e0c	Default	Default	up	1500	auto/1000
	e0d	Default	Default	up	1500	auto/1000
	e0e	Default	Default	up	1500	auto/1000
	e0f	Default	Default	up	1500	auto/1000
	e0g	Default	Default	up	1500	auto/1000

```
14 entries were displayed.
```

### 3. Verify the MetroCluster configuration from both sites in the MetroCluster configuration:

#### a. Verify the configuration from site A:

```
metrocluster show
```

```
cluster_A::> metrocluster show
```

```
Configuration: IP fabric
```

Cluster	Entry Name	State
-----		
Local: cluster_A	Configuration state	configured
	Mode	normal
Remote: cluster_B	Configuration state	configured
	Mode	normal

#### b. Verify the configuration from site B:

```
metrocluster show
```

```
cluster_B::> metrocluster show
```

```
Configuration: IP fabric
```

Cluster	Entry Name	State
-----	-----	-----
Local: cluster_B	Configuration state	configured
	Mode	normal
Remote: cluster_A	Configuration state	configured
	Mode	normal

## Creating a mirrored data aggregate on each node

You must create a mirrored data aggregate on each node in the DR group.

- You should know what drives will be used in the new aggregate.
- If you have multiple drive types in your system (heterogeneous storage), you should understand how you can ensure that the correct drive type is selected.
- Drives are owned by a specific node; when you create an aggregate, all drives in that aggregate must be owned by the same node, which becomes the home node for that aggregate.

In systems using ADP, aggregates are created using partitions in which each drive is partitioned in to P1, P2 and P3 partitions.

- Aggregate names should conform to the naming scheme you determined when you planned your MetroCluster configuration.

### Disk and aggregate management

1. Display a list of available spares:

```
storage disk show -spare -owner node_name
```

2. Create the aggregate by using the `storage aggregate create -mirror true` command.

If you are logged in to the cluster on the cluster management interface, you can create an aggregate on any node in the cluster. To ensure that the aggregate is created on a specific node, use the `-node` parameter or specify drives that are owned by that node.

You can specify the following options:

- Aggregate's home node (that is, the node that owns the aggregate in normal operation)
- List of specific drives that are to be added to the aggregate
- Number of drives to include



In the minimum supported configuration, in which a limited number of drives are available, you must use the `force-small-aggregate` option to allow the creation of a three disk RAID-DP aggregate.

- Checksum style to use for the aggregate
- Type of drives to use
- Size of drives to use
- Drive speed to use
- RAID type for RAID groups on the aggregate
- Maximum number of drives that can be included in a RAID group
- Whether drives with different RPM are allowed For more information about these options, see the `storage aggregate create` man page.

The following command creates a mirrored aggregate with 10 disks:

+

```
cluster_A::> storage aggregate create aggr1_node_A_1 -diskcount 10 -node
node_A_1 -mirror true
[Job 15] Job is queued: Create aggr1_node_A_1.
[Job 15] The job is starting.
[Job 15] Job succeeded: DONE
```

1. Verify the RAID group and drives of your new aggregate:

```
storage aggregate show-status -aggregate aggregate-name
```

## Configuring FC-to-SAS bridges for health monitoring

- Third-party SNMP monitoring tools are not supported for FibreBridge bridges.
- Starting with ONTAP 9.8, FC-to-SAS bridges are monitored via in-band connections by default, and additional configuration is not required.



Starting with ONTAP 9.8, the `storage bridge` command is replaced with `system bridge`. The following steps show the `storage bridge` command, but if you are running ONTAP 9.8 or later, the `system bridge` command is preferred.

### Steps

1. From the ONTAP cluster prompt, add the bridge to health monitoring:
  - a. Add the bridge, using the command for your version of ONTAP:

ONTAP version	Command
9.5 and later	<code>storage bridge add -address 0.0.0.0 -managed-by in-band -name bridge-name</code>

ONTAP version	Command
9.4 and earlier	<code>storage bridge add -address <i>bridge-ip-address</i> -name <i>bridge-name</i></code>

b. Verify that the bridge has been added and is properly configured:

`storage bridge show`

It might take as long as 15 minutes to reflect all data because of the polling interval. The ONTAP health monitor can contact and monitor the bridge if the value in the **Status** column is **ok**, and other information, such as the worldwide name (WWN), is displayed.

The following example shows that the FC-to-SAS bridges are configured:

```

controller_A_1::> storage bridge show

Bridge              Symbolic Name Is Monitored  Monitor Status
Vendor Model        Bridge WWN
-----
ATTO_10.10.20.10  atto01         true          ok            Atto
FibreBridge 7500N  20000010867038c0
ATTO_10.10.20.11  atto02         true          ok            Atto
FibreBridge 7500N  20000010867033c0
ATTO_10.10.20.12  atto03         true          ok            Atto
FibreBridge 7500N  20000010867030c0
ATTO_10.10.20.13  atto04         true          ok            Atto
FibreBridge 7500N  2000001086703b80

4 entries were displayed

controller_A_1::>

```

## Moving a metadata volume in MetroCluster configurations

You can move a metadata volume from one aggregate to another aggregate in a MetroCluster configuration. You might want to move a metadata volume when the source aggregate is decommissioned or unmirrored, or for other reasons that make the aggregate ineligible.

- You must have cluster administrator privileges to perform this task.
- The target aggregate must be mirrored and should not be in the degraded state.
- The available space in the target aggregate must be larger than the metadata volume that you are moving.

### Steps



1. Set the privilege level to advanced:

```
set -privilege advanced
```

2. Identify the metadata volume that should be moved:

```
volume show MDV_CRS*
```

```
Cluster_A::*> volume show MDV_CRS*
Vserver   Volume                               Aggregate      State      Type      Size
Available Used%
-----  -----
Cluster_A
          MDV_CRS_14c00d4ac9f311e7922800a0984395f1_A
          Node_A_1_aggr1
          online      RW        10GB
9.50GB   5%
Cluster_A
          MDV_CRS_14c00d4ac9f311e7922800a0984395f1_B
          Node_A_2_aggr1
          online      RW        10GB
9.50GB   5%
Cluster_A
          MDV_CRS_15035e66c9f311e7902700a098439625_A
          Node_B_1_aggr1
          -          RW        -
-        -
Cluster_A
          MDV_CRS_15035e66c9f311e7902700a098439625_B
          Node_B_2_aggr1
          -          RW        -
-        -
4 entries were displayed.

Cluster_A::>
```

3. Identify an eligible target aggregate:

```
metrocluster check config-replication show-aggregate-eligibility
```

The following command identifies the aggregates in cluster\_A that are eligible to host metadata volumes:

```
Cluster_A::*> metrocluster check config-replication show-aggregate-
eligibility
```

```
Aggregate Hosted Config Replication Vols Host Addl Vols Comments
-----
-----
Node_A_1_aggr0 - false Root Aggregate
Node_A_2_aggr0 - false Root Aggregate
Node_A_1_aggr1 MDV_CRS_1bc7134a5ddf11e3b63f123478563412_A true -
Node_A_2_aggr1 MDV_CRS_1bc7134a5ddf11e3b63f123478563412_B true -
Node_A_1_aggr2 - true
Node_A_2_aggr2 - true
Node_A_1_Aggr3 - false Unable to determine available space of aggregate
Node_A_1_aggr5 - false Unable to determine mirror configuration
Node_A_2_aggr6 - false Mirror configuration does not match requirement
Node_B_1_aggr4 - false NonLocal Aggregate
```



In the previous example, Node\_A\_1\_aggr2 and Node\_A\_2\_aggr2 are eligible.

#### 4. Start the volume move operation:

```
volume move start -vserver svm_name -volume metadata_volume_name -destination
-aggregate destination_aggregate_name
```

The following command moves metadata volume MDV\_CRS\_14c00d4ac9f311e7922800a0984395f1 from aggregate Node\_A\_1\_aggr1 to aggregate Node\_A\_1\_aggr2:

```
Cluster_A::*> volume move start -vserver svm_cluster_A -volume
MDV_CRS_14c00d4ac9f311e7922800a0984395f1
-destination-aggregate aggr_cluster_A_02_01

Warning: You are about to modify the system volume
"MDV_CRS_9da04864ca6011e7b82e0050568be9fe_A". This may cause
severe
performance or stability problems. Do not proceed unless
directed to
do so by support. Do you want to proceed? {y|n}: y
[Job 109] Job is queued: Move
"MDV_CRS_9da04864ca6011e7b82e0050568be9fe_A" in Vserver
"svm_cluster_A" to aggregate "aggr_cluster_A_02_01".
Use the "volume move show -vserver svm_cluster_A -volume
MDV_CRS_9da04864ca6011e7b82e0050568be9fe_A" command to view the status
of this operation.
```

5. Verify the state of the volume move operation:

```
volume move show -volume vol_constituent_name
```

6. Return to the admin privilege level:

```
set -privilege admin
```

## Checking the MetroCluster configuration

You can check that the components and relationships in the MetroCluster configuration are working correctly. You should do a check after initial configuration and after making any changes to the MetroCluster configuration. You should also do a check before a negotiated (planned) switchover or a switchback operation.

If the `metrocluster check run` command is issued twice within a short time on either or both clusters, a conflict can occur and the command might not collect all data. Subsequent `metrocluster check show` commands do not show the expected output.

### Steps

1. Check the configuration:

```
metrocluster check run
```

The command runs as a background job and might not be completed immediately.

```
cluster_A::> metrocluster check run
The operation has been started and is running in the background. Wait
for
it to complete and run "metrocluster check show" to view the results. To
check the status of the running metrocluster check operation, use the
command,
"metrocluster operation history show -job-id 2245"
```

```
cluster_A::> metrocluster check show
Last Checked On: 9/13/2018 20:41:37
```

Component	Result
nodes	ok
lifs	ok
config-replication	ok
aggregates	ok
clusters	ok
connections	ok

6 entries were displayed.

2. Display more detailed results from the most recent `metrocluster check run` command:

```
metrocluster check aggregate show
```

```
metrocluster check cluster show
```

```
metrocluster check config-replication show
```

```
metrocluster check lif show
```

```
metrocluster check node show
```

The `metrocluster check show` commands show the results of the most recent `metrocluster check run` command. You should always run the `metrocluster check run` command prior to using the `metrocluster check show` commands so that the information displayed is current.

The following example shows the `metrocluster check aggregate show` command output for a healthy four-node MetroCluster configuration:

```
cluster_A::> metrocluster check aggregate show
```

```
Last Checked On: 8/5/2014 00:42:58
```

Node	Aggregate	Check
Result		
-----	-----	-----
controller_A_1	controller_A_1_aggr0	mirroring-status
ok		disk-pool-allocation
ok		ownership-state
ok		

```

controller_A_1_aggr1
ok      mirroring-status
ok      disk-pool-allocation
ok      ownership-state

controller_A_1_aggr2
ok      mirroring-status
ok      disk-pool-allocation
ok      ownership-state

controller_A_2      controller_A_2_aggr0
ok      mirroring-status
ok      disk-pool-allocation
ok      ownership-state

controller_A_2      controller_A_2_aggr1
ok      mirroring-status
ok      disk-pool-allocation
ok      ownership-state

controller_A_2      controller_A_2_aggr2
ok      mirroring-status
ok      disk-pool-allocation
ok      ownership-state

18 entries were displayed.

```

The following example shows the `metrocluster check cluster show` command output for a healthy four-node MetroCluster configuration. It indicates that the clusters are ready to perform a negotiated switchover if necessary.

Last Checked On: 9/13/2017 20:47:04

Cluster	Check	Result
-----	-----	-----
mccint-fas9000-0102	negotiated-switchover-ready	not-applicable
	switchback-ready	not-applicable
	job-schedules	ok
	licenses	ok
	periodic-check-enabled	ok
mccint-fas9000-0304	negotiated-switchover-ready	not-applicable
	switchback-ready	not-applicable
	job-schedules	ok
	licenses	ok
	periodic-check-enabled	ok

10 entries were displayed.

## Copyright Information

Copyright © 2021 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.