

[◀ Prev](#)[Next ▶](#)

## 3.4. Global Cluster Properties

When a cluster is created, or if you select a cluster to configure, a cluster-specific page is displayed. The page provides an interface for configuring cluster-wide properties and detailed properties. You can configure cluster-wide properties with the tabbed interface below the cluster name. The interface provides the following tabs: **General**, **Fence**, **Multicast**, and **Quorum Partition**. To configure the parameters in those tabs, follow the steps in this section. If you do not need to configure parameters in a tab, skip the step for that tab.

1. **General** tab — This tab displays cluster name and provides an interface for configuring the configuration version and advanced cluster properties. The parameters are summarized as follows:
  - ✦ The **Cluster Name** text box displays the cluster name; it does not accept a cluster name change. You cannot change the cluster name. The only way to change the name of a Red Hat cluster is to create a new cluster configuration with the new name.
  - ✦ The **Configuration Version** value is set to 1 by default and is automatically incremented each time you modify your cluster configuration. However, if you need to set it to another value, you can specify it at the **Configuration Version** text box.
  - ✦ You can enter advanced cluster properties by clicking **Show advanced cluster properties**. Clicking **Show advanced cluster properties** reveals a list of advanced properties. You can click any advanced property for online help about the property.

Enter the values required and click **Apply** for changes to take effect.

2. **Fence** tab — This tab provides an interface for configuring these **Fence Daemon Properties** parameters: **Post-Fail Delay** and **Post-Join Delay**. The parameters are summarized as follows:
  - ✦ The **Post-Fail Delay** parameter is the number of seconds the fence daemon (fenced) waits before fencing a node (a member of the fence domain) after the node has failed. The **Post-Fail Delay** default value is 0. Its value may be varied to suit cluster and network performance.
  - ✦ The **Post-Join Delay** parameter is the number of seconds the fence daemon (fenced) waits before fencing a node after the node joins the fence domain. The **Post-Join Delay** default value is 3. A typical setting for **Post-Join Delay** is between 20 and 30 seconds, but can vary according to cluster and network performance.

Enter values required and Click **Apply** for changes to take effect.



### Note

For more information about **Post-Join Delay** and **Post-Fail Delay**, refer to the fenced(8) man page.

3. **Multicast** tab — This tab provides an interface for configuring these **Multicast Configuration** parameters: **Let cluster choose the multicast address** and **Specify the multicast address manually**. The default setting is **Let cluster choose the multicast address**. If you need to use a specific multicast address, click **Specify the multicast address manually**, enter a multicast address into the text box, and click **Apply** for changes to take effect.

If you do not specify a multicast address, `cman` (the cluster manager in Red Hat Cluster software) creates one based on the cluster ID. It generates the lower 16 bits of the address and appends them to the upper portion of the address according to whether the IP protocol is IPv4 or IPv6:

- ✦ For IPv4 — The address formed is 239.192. plus the lower 16 bits generated by `cman`.
- ✦ For IPv6 — The address formed is FF15:: plus the lower 16 bits generated by `cman`.

If you do specify a multicast address, you should use the 239.192.x.x series (or FF15:: for IPv6) that `cman` uses. Otherwise, using a multicast address outside that range may cause unpredictable results. For example, using 224.0.0.x (which is "All hosts on the network") may not be routed correctly, or even routed at all by some hardware.



#### Note

If you specify a multicast address, make sure that you check the configuration of routers that cluster packets pass through. Some routers may take a long time to learn addresses, seriously impacting cluster performance.



#### Note

The cluster ID is a unique identifier that `cman` generates for each cluster. To view the cluster ID, run the `cman_tool status` command on a cluster node.

4. **Quorum Partition** tab — This tab provides an interface for configuring these **Quorum Partition Configuration** parameters: **Do not use a Quorum Partition**, **Use a Quorum Partition**, **Interval**, **Votes**, **TKO**, **Minimum Score**, **Device**, **Label**, and **Heuristics**. The **Do not use a Quorum Partition** parameter is enabled by default. [Table 3.1, "Quorum-Disk Parameters"](#) describes the parameters. If you need to use a quorum disk, click **Use a Quorum Partition**, enter quorum disk parameters, click **Apply**, and restart the cluster for the changes to take effect.



#### Important

Quorum-disk parameters and heuristics depend on the site environment and the special requirements needed. To understand the use of quorum-disk parameters and heuristics, refer to the `qdisk(5)` man page. If you require assistance understanding and using quorum disk, contact an authorized Red Hat support representative.



#### Note

Clicking **Apply** on the **Quorum Partition** tab propagates changes to the cluster configuration file (`/etc/cluster/cluster.conf`) in each cluster node. However, for the quorum disk to operate, you must restart the cluster (refer to [Section 4.1, "Starting, Stopping, and Deleting Clusters"](#)).

Parameter	Description
<b>Do not use a Quorum Partition</b>	Disables quorum partition. Disables quorum-disk parameters in the <b>Quorum Partition</b> tab.
<b>Use a Quorum Partition</b>	Enables quorum partition. Enables quorum-disk parameters in the <b>Quorum Partition</b> tab.
<b>Interval</b>	The frequency of read/write cycles, in seconds.
<b>Votes</b>	The number of votes the quorum daemon advertises to CMAN when it has a high enough score.
<b>TKO</b>	The number of cycles a node must miss to be declared dead.
<b>Minimum Score</b>	The minimum score for a node to be considered "alive". If omitted or set to 0, the default function, $\text{floor}((n+1)/2)$ , is used, where $n$ is the sum of the heuristics scores. The <b>Minimum Score</b> value must never exceed the sum of the heuristic scores; otherwise, the quorum disk cannot be available.
<b>Device</b>	The storage device the quorum daemon uses. The device must be the same on all nodes.
<b>Label</b>	Specifies the quorum disk label created by the <code>mkqdisk</code> utility. If this field contains an entry, the label overrides the <b>Device</b> field. If this field is used, the quorum daemon reads <code>/proc/partitions</code> and checks for <code>qdisk</code> signatures on every block device found, comparing the label against the specified label. This is useful in configurations where the quorum device name differs among nodes.
<b>Heuristics</b>	<p><b>Path to Program</b> — The program used to determine if this heuristic is alive. This can be anything that can be executed by <code>/bin/sh -c</code>. A return value of 0 indicates success; anything else indicates failure. This field is required.</p> <p>.....</p> <p><b>Interval</b> — The frequency (in seconds) at which the heuristic is polled. The default interval for every heuristic is 2 seconds.</p> <p>.....</p> <p><b>Score</b> — The weight of this heuristic. Be careful when determining scores for heuristics. The default score for each heuristic is 1.</p>
<b>Apply</b>	Propagates the changes to the cluster configuration file ( <code>/etc/cluster/cluster.conf</code> ) in each cluster node.

**Table 3.1. Quorum-Disk Parameters**



Prev

3.3. Creating A Cluster



Up



Home

Next



3.5. Configuring Fence Devices