A cluster is two or more computers (called *nodes* or *members*) that work together to perform a task. There are four major types of clusters:

* Storage
* High availability
* Load balancing
* High performance

1. Storage clusters provide a consistent file system image across servers in a cluster, allowing the servers to simultaneously read and write to a single shared file system
2. High-availability clusters provide continuous availability of services by eliminating single points of failure and by failing over services from one cluster node to another in case a node becomes inoperative.
3. Load-balancing clusters dispatch network service requests to multiple cluster nodes to balance the request load among the cluster nodes.
4. High-performance clusters use cluster nodes to perform concurrent calculations. A high-performance cluster allows applications to work in parallel, therefore enhancing the performance of the applications.

# Components of redhat cluster:

**Luci :**

luci service is the management service that presents the web based cluster interface via https at port 8084 and can be accessed in any browser at  
https://<cluster management node FQDN or hostname:8084>/

**Ricci:**

ricci service is the underlying daemon that helps in cluster configuration sync and file copy, service start, stop etc. and uses tcp port 11111.

**Cman, rgmanager, modclusterd:**

cman, rgmanager and modclusterd are the actual cluster services which futher start other services that actually make the clustering happen and keep it live.

**Cman:** Cluster Manager is distributed and runs in each cluster node, is responsible for monitoring, heartbeat, quorum, voting and communication between cluster nodes.

**Rgmanager: Resource Group Manager,** manages and provides failover capabilities for collections of cluster resources called services, resource groups, or resource trees. Commands used under rgmanager are clusvcadm, clustat, clurgmgrd (Daemon used to handle user service), clurmtabd (Daemon used to handle Clustered NFS mount tables).

**Fencing:** To ensure data integrity, only one node can run a cluster service and access cluster-servicedata at a time. The use of power switches in the cluster hardware configuration enables a nodeto power-cycle another node before restarting that node's HA services during a failoverprocess. T his prevents two nodes from simultaneously accessing the same data andcorrupting it. *Fence devices* (hardware or software solutions that remotely power, shutdown,

and reboot cluster nodes) are used to guarantee data integrity under all failure conditions.

**Clvmd:** Clustered LVM, In a shared storage cluster scenario where multiple nodes are writing data concurrently to the same storage there comes a probability of metadata being get overwritten or corrupted. So clvmd service is responsible for managing clustered shared storage to reflect the metadata changes instantly on all the cluster nodes using the same storage which removes the possibility of meta data corruption. With this option same volume group or logical volume can be mounted on multiple nodes of a cluster as long as clvmd service is [running](javascript:void(0);) and cluster is quorate.

**Ccsd:** Cluster configuration System Daemon, is responsible for loading /etc/cluster/cluster.conf and managing the configuration versions.

**Qdiskd:** This daemon is responsible for managing a quorum disk(qdisk).

It includes the concept of qdisk i.e. quorum disk. A Red Hat Cluster runs only if Quorate state is achieved which is determined by the no. of active votes representing the cluster. Each node by default counts as one vote. Now in extreme condition when you want your cluster to be running with minimum no. of nodes in those case we use quorum disk having predefined vote count. Now this may sound confusing so let me explain with an example:

For a 3 node cluster a quorum state is present untill 2 of the 3 nodes are active i.e. more than half. But what if due to some reasons the 2nd node also stops communicating with the the 3rd node? In that case under a normal architecture the cluster would dissolve and stop working. But for mission critical environments and such scenarios we use quorum disk in which an additional disk is configured which is mounted on all the nodes with qdiskd service running and a vote value is [assigned](javascript:void(0);) to it.

So suppose in above case I have assigned 1 vote to qdisk so even after 2 nodes stops communicating with 3rd node, the cluster would have 2 votes (1 qdisk + 1 from 3rd node) which is still more than half of vote count for a 3 node cluster. Now both the inactive nodes would be fenced and your 3rd node would be still up and running being a part of the cluster.

**Gfs and gfs2: Global File System,** the services responsible for mounting/unmounting GFS and GFS2 filesystems from /etc/fstab respectively. It is a cluster file system that [allows](http://a.yesadsrv.com/cpxcenter/dpop.php?nid=4&pid=1215&sid=58339&zone=80571&durl=&subid=%5bsid%5d&opt1=&opt2=) a cluster of nodes to simultaneously access ablock device that is shared among the nodes. To maintain file system integrity, GFS uses a lock manager to coordinate I/O. When one node changes data on a GFS file system, that change is immediately visible to the other cluster nodes using that file system.

**Dlm\_controld: Distributed Lock Manager, GFS** and CLVM use locks from the lock manager to simultanously synchronize the metadata on the shared storage (LVM/Volume Groups).Without a lock manager, there would be no control over access to your shared storage, and the nodes in the cluster would corrupt each other's data. In addition, rgmanager uses DLM to synchronize service states

**Heartbeat:** Cluster heartbeat delivers 2 functions:

* 1. The efficient delivery of messages to other nodes in the cluster
  2. The detection of when a node has left the cluster.

Cluster communications using the default multicast transport occurs through a combination of unicast and multicast traffic. All nodes pass a unicast heartbeat token in a circular fashion. Token starts on the node with the lowest ip address and proceeds to nodes with higher ip addresses. Only the node that holds the token is allowed to transmit multicast messages.

Every multicast message has a sequence number, and the sequence number increases with every new multicast message transmitted. When a node has finished transmitting, it records in the token the sequence number of the last multicast message it sent and passes the token to the next node. The next node then checks to make sure it has received all messages up to that sequence number, and requests retransmission of any messages that have not been received. In later versions of cluster, each node will wait for five failures to receive before requesting a retransmit which decreases the number of transmissions if there are multicast delays.

The Protocol used between the nodes for communication is called **Totem** and its provided by **corosync**. For more details on Totem protocol, read <https://access.redhat.com/articles/390433>.

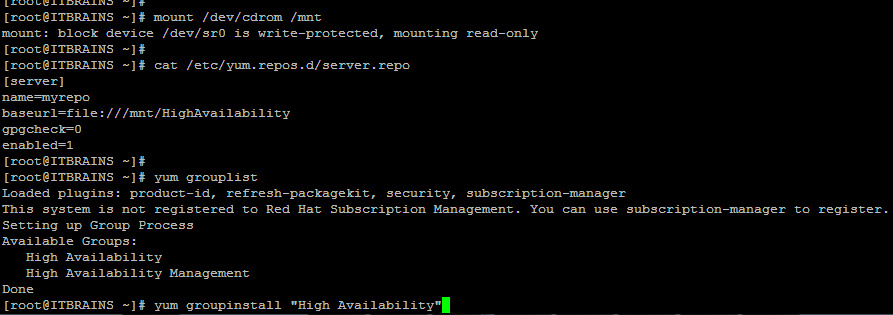
# Steps to configure a 2 node “High Availability” active backup cluster:

1. Create the Yum repositories capable to run groupinstall and install the packages as below on both the cluster. You can make a third cluster management node, you just need to install “High Availability Management” group on that node. If you want you can make each node a cluster management node by installing this group and starting luci service.

Mount the cdrom and make yum repository.

**# yum groupinstall "High Availability"**

**# yum groupinstall "High Availability Management"**



1. Start/Stop the services on both the nodes:

**service iptables stop  
service ip6tables stop  
service ricci start  
service rgmanager start  
service modclusterd start**

**service NetworkManager stop**

1. Enable/Disable the below services on both the nodes

**#chkconfig iptables off  
#chkconfig ip6tables off  
#chkconfig ricci on  
#chkconfig NetworkManager off**

**#chkconfig rgmanager on  
#chkconfig modclusterd on**

1. Make sure the selinux is disabled on both the nodes.
2. Make the passwordless ssh authentication on both nodes
3. Make sure the /etc/hosts file is updated with names of both/all the machines.
4. Create the password for ricci user on both the nodes

passwd ricci

1. Start luci service if it is not running and point your browser to https://yourserver:8084
2. Check if the cman service is started, it should be started automatically once the nodes are connected with cluster.
3. Now create a new cluster and add the nodes in it. Once the nodes are added, the cman service should be started automatically on the nodes. If not, start it manually.
4. Configure Fencing (For testing purpose, you can ignore fencing)
5. Configure the Failover Domain
6. Configure Resources
   1. Add ip address as the resource which is going to be floating on the nodes
   2. Add the script /etc/init.d/httpd as another resource
7. Add resources in the service group
8. Validate cluster configuration with below command”

#ccs\_config\_validate

**Service cman cannot be started if the file cluster.conf not found**

If the file cluster.conf not found, it can be created with below steps in CLI

ccs -h *host* --createcluster *clustername*

**To add a host**

ccs -h node-01.example.com --addnode node-01.example.com

ccs -h node-01.example.com --addnode node-02.example.com

ccs -h node-01.example.com --addnode node-03.example.com

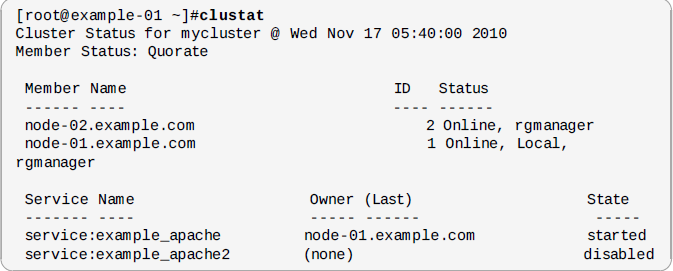
**Copy the file cluster.conf to all the nodes of cluster**

**To copy the configuration file from one node to another node setup the passwordless authentication between all the nodes.**

**#service cman start**

# Commands to check/manage cluster in CLI:

1. How to check status of cluster?

He cluster

1. Start/enable a service on a node

clusvcadm –e <service name> -m <cluster member>

1. Disable a service on a node

clusvcadm –d <service name>

1. Relocate the service to another node

clusvcadm –r <service\_name> -m <member>

**To stop the cluster follow the below steps:-**

1. service modclusterd stop
2. service cman stop
3. service rgmanager stop
4. What is quorum in cluster ?

Ans. A cluster quorum disk is the [storage](http://searchstorage.techtarget.com/definition/storage) medium on which the [configuration](http://searchexchange.techtarget.com/definition/configuration) [database](http://searchsqlserver.techtarget.com/definition/database) is stored for a [cluster computing](http://searchdatacenter.techtarget.com/definition/cluster-computing) network. The cluster configuration database, also called the quorum, tells the cluster which physical server(s) should be active at any given time. The quorum disk comprises a shared block device that allows concurrent read/write access by all nodes in a cluster.

1. What is DLM in redhat cluster ?
2. What is ACPI and how to disable ACPI soft off in server ?
3. What is the role of rgmanager in redhat cluster ?

Ans. rgmanager, implements cold failover for off-the-shelf applications. In the Red Hat High Availability Add-On, an application is configured with other cluster resources to form an HA service that can fail over from one cluster node to another with no apparent interruption to cluster clients. HA-service failover can occur if a cluster node fails or if a cluster system administrator moves the service from one cluster node to another.

1. **What is split-brain**  
     
   It is a condition where two instances of the same cluster are running and trying to access same resource at the same time, resulting in corrupted cluster integrity  
   Cluster must maintain quorum to prevent split-brain issues  
     
   It's necessary for a cluster to maintain quorum to prevent 'split-brain' problems. If we didn't enforce quorum, a communication error on that same thirteen-node cluster  
   may cause a situation where six nodes are operating on the shared disk, and another six were also operating on it, independently. Because of the communication error,  
   the two partial-clusters would overwrite areas of the disk and corrupt the file system. With quorum rules enforced, only one of the partial clusters can use the shared storage, thus protecting data integrity.  
     
   Quorum doesn't prevent split-brain situations, but it does decide who is dominant and allowed to function in the cluster. Should split-brain occur, quorum prevents more than one cluster group from doing anything.

**What is Quorum disk**

* In case of a 2 node cluster, quorum disk acts as a tie-breaker and prevents split-brain issue
* If a node has access to network and quorum disk, it is active
* If a node has lost access to network or quorum disk, it is inactive and can be fenced
* A Quorum disk, known as a qdisk is small partition on SAN storage used to enhance quorum. It generally carries enough votes to allow even a single node to take quorum during a cluster partition.
* It does this by using configured heuristics, that is custom tests, to decided which which node or partition is best suited for providing clustered services during a cluster reconfiguration