GOLINUXHUB
Sys Admin Tutorial Guide

VMware

Learn about VMware virtualization for its products like vsphere ESX and ESXi, vCenter Server, VMware View, VMware P2V and many more

Linux

Step by step configuration tutorials for many of the Linux services like DNS, DHCP, FTP, Samba4 etc including many tips and tricks in Red Hat Linux.

Database

Learn installation a databases like Oracle etc including many ot Linux.

A SECOND CHANCE ... ITS CALLED TOMORROW !!!

Configure Red Hat Cluster using VMware, Quorum Disk, GFS2, Openfiler

POSTED BY DEEPAK PRASAD WEDNESDAY, FEBRUARY 26, 2014 8 COMMENTS

In this article I will be showing you step by step guide to install and configure **Red Hat Cluster** using **VMware Workstation 10**.

These are the things which I would be using as per my lab setup:

- VMware Workstation 10 (any version is fine above 8)
- CentOS 6.5 - 64 bit (You can use either 32 or 64 bit and also if you use earlier versions, some rpm and packages would differ for any version below 6.0)
- Openfiler 2.99 - 64 bit

Brief intro of what we are trying to accomplish

1. Configure a 2 node Red Hat Cluster using CentOS 6.5 (64 bit)
2. One node will be used for management purpose of cluster with **luci** using CentOS 6.5 (64 bit)
3. **Openfiler** will be used to configure a shared iSCSI storage for the cluster
4. Configure **failver** for both the nodes
5. Configure a **Quorum disk** with 1 one vote to test the failover
6. Create a common **service GFS2** which will run on any one node of our cluster with failover policy

NOTE: I will not be able to configure fencing related settings as it is not supported on vmware. For more information please visit this site Fence Device and Agent Information for Red Hat Enterprise Linux

IMPORTANT NOTE: In this article I will not be able to explain properly all the terms used, for that you can always refer the Official Guide from Red Hat on Cluster Administration for further clarification

Lab Setup

2 nodes with CentOS 6.5 - 64 bit
Node 1
Hostname: node1.cluster
IP Address: 192.168.1.5

Node 2
Hostname: node2.cluster
IP Address: 192.168.1.6

1 Node for Management Interface with CentOS 6.5 - 64 bit
Node 1
Hostname: node3.mgmt
IP Address: 192.168.1.7

Openfiler
Hostname: of.storage
IP Address: 192.168.1.8

Search

SUBSCRIBE FOR MAIL ALERTS

Email address...

LIKE US ON FACEBOOK FOR ALL UPDATES

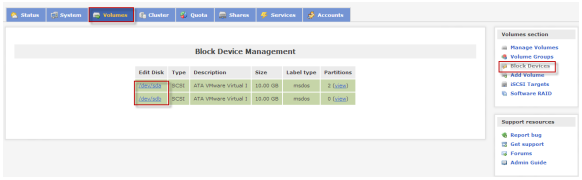
Before moving to start with the configuration of cluster and cluster nodes let us prepare our openfiler with iSCSI storage.

Login to the web console of your openfiler storage (*assuming* that you have successfully installed openfiler with sufficient free space for cluster storage)

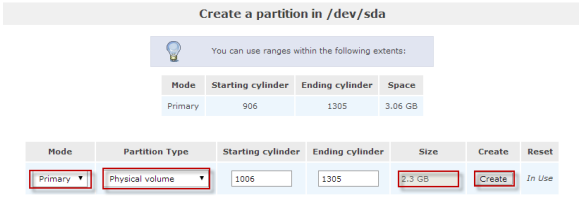
Here I have written one more article on configuration of openfiler which you can use for reference if you face any issues understanding me here as I will be very brief
Configuring iSCSI storage using openfiler

1. Configure iSCSI Target using Openfiler

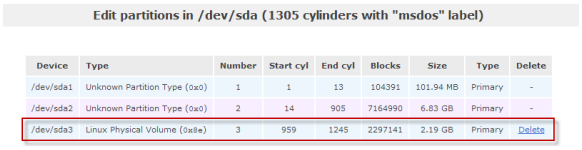
Click on **Block Management** and select the partition where you want to create **Physical Volume.**



Create a **new partition** with the below shown options for the available disk. Mention a cylinder value for the partition

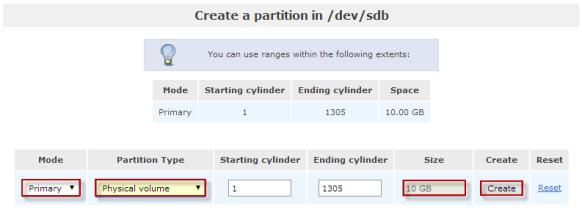


Once done you should see a **new partition** added

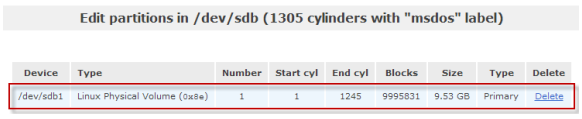


Similarly create a new partition for next disk `/dev/sdb`

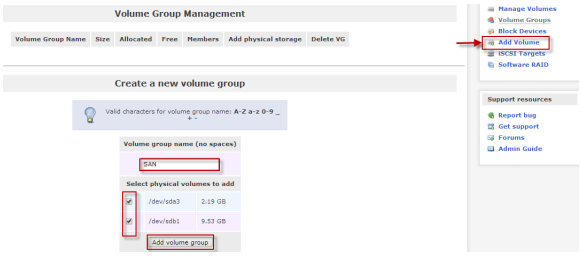
Select **Physical Volume** in the Partition Type



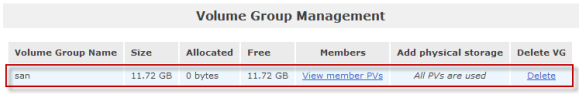
So our one more partition is created as you see below



Configure a **Volume Group** for both the partition you created



So, we have successfully create a new Volume Group **SAN**

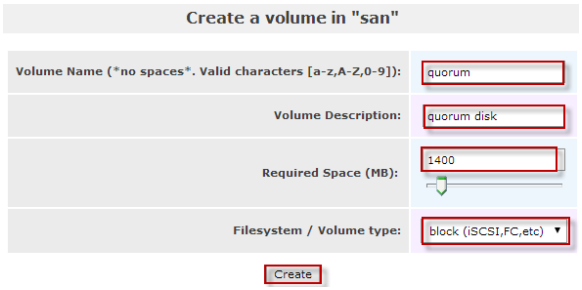


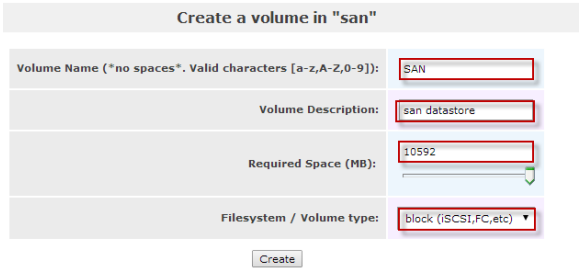
Next is to create a new Logical Volume. Create 2 Logical Volumes with custom size as per your requirement.

For my case I will create two volumes

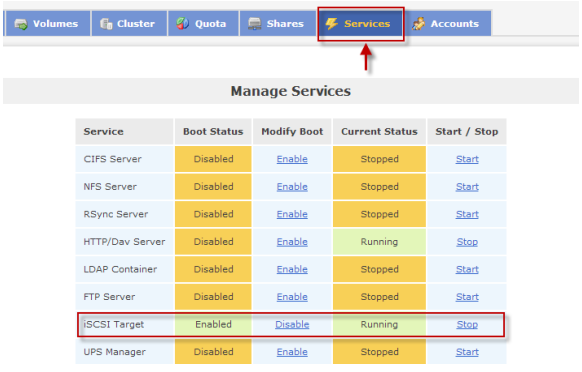
1. quorum with size 1400 MB (Quorum disk does not requires disk space more than 1GB)

2. SAN with all the left size which will be used for GFS2 filesystem in our cluster

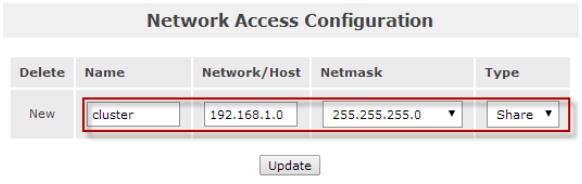




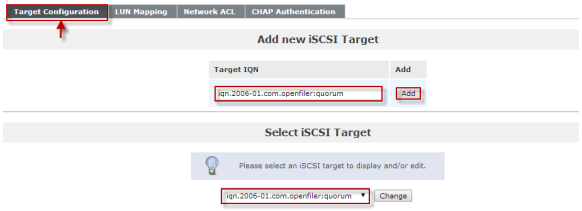
Start the **iSCSI Target** services



On the home page of **system** create a **ACL** for the subnet which will try to access the openfiler storage. For my case the subnet is **192.168.1.0** so I will add a new entry for the same with relative subnet mask.

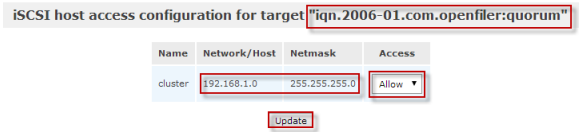


Next **Add iscsi target** for the first disk i.e. **quorum volume**. You can edit the iscsi target value with custom name as I have done for my case so that it becomes easier for me to understand



Next **map the volume to the iSCSI target**. For quorum target select quorum partition and click on **Map** as shown below

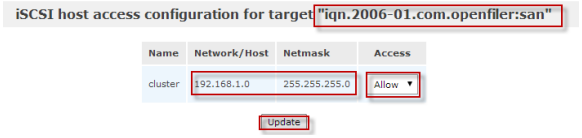
Next allow the **iSCSI target** in the **Network ACL** section



Do the same steps for **SAN volume** also as we did for quorum volume above. Edit the target value as shown below

Map the volume to the **iSCSI target** as shown in the figure below. Be sure to the map the correct volume

Allow the ACL for that particular target in **Network ACL** section



2. Let us start configuring our Cluster

We are going to use luci also known as Conga for Administering and management purpose for the cluster.

What is Conga?

Conga is an integrated set of software components that provides centralized configuration and management of Red Hat clusters and storage. Conga provides the following major features:

- One Web interface for managing cluster and storage
- Automated Deployment of Cluster Data and Supporting Packages
- Easy Integration with Existing Clusters
- No Need to Re-Authenticate

- Integration of Cluster Status and Logs
- Fine-Grained Control over User Permissions

The primary components in Conga are luci and ricci, which are separately installable. luci is a server that runs on one computer and communicates with multiple clusters and computers viaricci. ricci is an agent that runs on each computer (either a cluster member or a standalone computer) managed by Conga

On node3:
Run the below command to install all the Clustering related packages

```
[root@node3 ~]# yum groupinstall "High Availability Management" "High Availability"
```

On node1 and node2:
Install the below given packages to start building your cluster nodes and connect to the iSCSI Targets as we will create in openfiler

```
[root@node1 ~]# yum groupinstall "iSCSI Storage Client" "High Availability"

[root@node2 ~]# yum groupinstall "iSCSI Storage Client" "High Availability"
```

3. Add iSCSI targets using iSCSi initiator

Once the Clustering packages are installed let us move to next step to add iSCSi storage in our cluster nodes (Here **192.168.1.8** is the IP f my openfiler)

```
[root@node1 ~]# iscsiadm -m discovery -t sendtargets -p 192.168.1.8
Starting iscsid:                                           [  OK  ]
192.168.1.8:3260,1 iqn.2006-01.com.openfiler:san
192.168.1.8:3260,1 iqn.2006-01.com.openfiler:quorum
```

As you see as soon as we gave the discovery command with openfiler IP address, the iSCSi targets got discovered automatically as configured on openfiler

Now restart the iscsi service once again to refresh the settings

```
[root@node1 ~]# service iscsi restart
Stopping iscsi:                                            [  OK  ]
Starting iscsi:                                            [  OK  ]
```

Verify the added iSCSI storage on your **node1**

```
[root@node1 ~]# fdisk -l

Disk /dev/sdb: 1476 MB, 1476395008 bytes
46 heads, 62 sectors/track, 1011 cylinders
Units = cylinders of 2852 * 512 = 1460224 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Disk /dev/sdc: 11.1 GB, 11106516992 bytes
64 heads, 32 sectors/track, 10592 cylinders
Units = cylinders of 2048 * 512 = 1048576 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000
```

Now perform the same steps on **node2**

```
[root@node2 ~]# iscsiadm -m discovery -t sendtargets -p 192.168.1.8
192.168.1.8:3260,1 iqn.2006-01.com.openfiler:san
192.168.1.8:3260,1 iqn.2006-01.com.openfiler:quorum
```

Restart iscsi services

```
[root@node2 ~]# service iscsi restart
Stopping iscsi:                                            [  OK  ]
Starting iscsi:                                            [  OK  ]
```

Verify the added iscsi storage as reflected on node1

```
[root@node2 ~]# fdisk -l

Disk /dev/sdb: 1476 MB, 1476395008 bytes
```

```
46 heads, 62 sectors/track, 1011 cylinders
Units = cylinders of 2852 * 512 = 1460224 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Disk /dev/sdc: 11.1 GB, 11106516992 bytes
64 heads, 32 sectors/track, 10592 cylinders
Units = cylinders of 2048 * 512 = 1048576 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000
```

4. Configure Quorum disk

You need to do this step on both the nodes.
NOTE: Before you do this step be **VERY** sure of the partition you use as it should not be used by any one since the below step will destroy all the data in that partition

For me I will use the iSCSI quorum partition which is `/dev/sdb`
Here we are using label name as "**quorum**"

NOTE: This label will be used in further steps so remember the name which you use. Also once you run the command on any of one of the node, the same would automatically be reflected on other nodes sharing the same partition.

```
[root@node1 ~]# mkqdisk -c /dev/sdb -l quorum
mkqdisk v3.0.12.1

Writing new quorum disk label 'quorum' to /dev/sdb.
WARNING: About to destroy all data on /dev/sdb; proceed [N/y] ? y
Warning: Initializing previously initialized partition
Initializing status block for node 1...
Initializing status block for node 2...
Initializing status block for node 3...
Initializing status block for node 4...
Initializing status block for node 5...
Initializing status block for node 6...
Initializing status block for node 7...
Initializing status block for node 8...
Initializing status block for node 9...
Initializing status block for node 10...
Initializing status block for node 11...
Initializing status block for node 12...
Initializing status block for node 13...
Initializing status block for node 14...
Initializing status block for node 15...
Initializing status block for node 16...
```

5. Format a GFS2 partition

Since we want GFS services to be running on our cluster so let us format the iSCSI san target which we mapped on the cluster nodes i.e. /dev/sdc

Explanation:
Formatting filesystem: GFS2
Locking Protocol: lock_dlm
Cluster Name: cluster1
FileSystem name: GFS
Journal: 2
Partition: /dev/sdc

Run the below command on any one of the nodes as the same would be reflected on all other nodes for the same partition which in our case is `/dev/sdc`

```
[root@node1 ~]# mkfs.gfs2 -p lock_dlm -t cluster1:GFS -j 2 /dev/sdc
This will destroy any data on /dev/sdc.
It appears to contain: Linux GFS2 Filesystem (blocksize 4096, lockproto lock_dlm)

Are you sure you want to proceed? [y/n] y

Device:                    /dev/sdc
```

```
Blocksize:                 4096
Device Size                10.34 GB (2711552 blocks)
Filesystem Size:           10.34 GB (2711552 blocks)
Journals:                  2
Resource Groups:           42
Locking Protocol:          "lock_dlm"
Lock Table:                "cluster1:GFS"
UUID:                      2ff81375-31f9-c57d-59d1-7573cdfaff42
```

For more information on GFS2 partition follow the below link

How to configure GFS2 partition in Red Hat Cluster

6. Assign password to ricci

As explained earlier ricci is the agent which is used by luci to connect to each cluster node. So we need to assign a password to the same. This has to be performed on both node1 and node2

```
[root@node1 ~]# passwd ricci
Changing password for user ricci.
New password:
BAD PASSWORD: it is based on a dictionary word
BAD PASSWORD: is too simple
Retype new password:
passwd: all authentication tokens updated successfully.
```

Restart the ricci services to take the changes affect

```
[root@node1 ~]# /etc/init.d/ricci start
Starting oddjobd:                                          [  OK  ]
generating SSL certificates...  done
Generating NSS database...  done
Starting ricci:                                            [  OK  ]
```

Make sure the ricci services comes up after reboot

```
[root@node1 ~]# chkconfig ricci on
```

```
[root@node2 ~]# passwd ricci
Changing password for user ricci.
New password:
BAD PASSWORD: it is based on a dictionary word
BAD PASSWORD: is too simple
Retype new password:
passwd: all authentication tokens updated successfully.

[root@node2 ~]# /etc/init.d/ricci start
Starting oddjobd:                                          [  OK  ]
generating SSL certificates...  done
Generating NSS database...  done
Starting ricci:                                            [  OK  ]
[root@node2 ~]# chkconfig ricci on
```

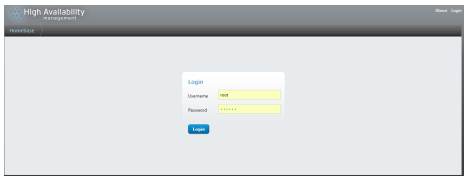
7. Starting conga services

Since **node3** is your **management server**, start luci services on it using the below command

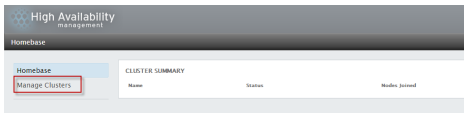
```
[root@node3 ~]# /etc/init.d/luci start
Adding following auto-detected host IDs (IP addresses/domain names), corresponding to
`node3.example' address, to the configuration of self-managed certificate
`/var/lib/luci/etc/cacert.config' (you can change them by editing
`/var/lib/luci/etc/cacert.config', removing the generated certificate
`/var/lib/luci/certs/host.pem' and restarting luci):         (none suitable found, you
can still do it manually as mentioned above)

Generating a 2048 bit RSA private key
writing new private key to '/var/lib/luci/certs/host.pem'
Starting saslauthd:                                        [  OK  ]
Start luci...                                              [  OK  ]
Point your web browser to https://node3.mgmt:8084 (or equivalent) to access luci
```

8. Accessing luci console

The default login credential will be your **node3** username/password i.e.
username: *root*
password: *Your root password*

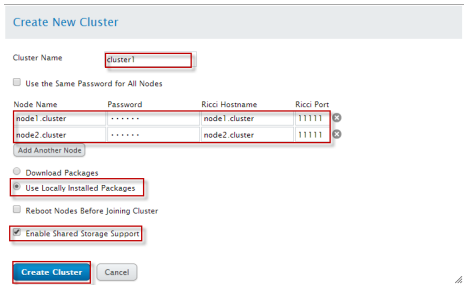


Click on **Manage Clusters** to create a new cluster

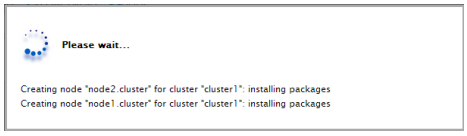


Click on **Create**

*Provide the following details for the cluster***Cluster name:** Cluster1(As provided above)
Node Name: node1.cluster (192.168.1.5) Make sure that hostname is resolvable
node2.cluster (192.168.1.6) Make sure that hostname is resolvable
Password: As provided for agent ricci in **Step 6**
Check Shared storage box as we are using GFS2



Once you click on **submit**, the nodes will start the procedure to add the nodes *(if everything goes correct or else it will throw the error)*



Now the nodes are added but they are shown in red color. Let us check the reason behind it. Click on any of the nodes for more details

So the reason looks like most of the services are not running . Let us login to the console and start the services


```
[root@node1 ~]# /etc/init.d/cman start
Starting cluster:
   Checking if cluster has been disabled at boot...        [  OK  ]
   Checking Network Manager...
Network Manager is either running or configured to run. Please disable it in the cluster.
                                                           [FAILED]
Stopping cluster:
   Leaving fence domain...                                 [  OK  ]
   Stopping gfs_controld...                                [  OK  ]
   Stopping dlm_controld...                                [  OK  ]
```

```
   Stopping fenced...                                      [  OK  ]
   Stopping cman...                                        [  OK  ]
   Unloading kernel modules...                             [  OK  ]
   Unmounting configfs...                                  [  OK  ]
```

IMPORTANT NOTE: If you are planning to configure Red Hat Cluster then make sure NetworkManager service is not running

```
[root@node1 ~]# service NetworkManager stop
Stopping NetworkManager daemon:                            [  OK  ]

[root@node1 ~]# chkconfig NetworkManager off
```

start the cman services

```
[root@node1 ~]# /etc/init.d/cman start
Starting cluster:
   Checking if cluster has been disabled at boot...        [  OK  ]
   Checking Network Manager...                             [  OK  ]
   Global setup...                                         [  OK  ]
   Loading kernel modules...                               [  OK  ]
   Mounting configfs...                                    [  OK  ]
   Starting cman...                                        [  OK  ]
   Waiting for quorum...                                   [  OK  ]
   Starting fenced...                                      [  OK  ]
   Starting dlm_controld...                                [  OK  ]
   Tuning DLM kernel config...                             [  OK  ]
   Starting gfs_controld...                                [  OK  ]
   Unfencing self...                                       [  OK  ]
   Joining fence domain...                                 [  OK  ]
```

start clvmd service

```
[root@node1 ~]# /etc/init.d/clvmd start
Activating VG(s):   2 logical volume(s) in volume group "VolGroup" now active
                                                           [  OK  ]
[root@node1 ~]# chkconfig  clvmd on
```

Start rgmanager and modclusterd service

```
[root@node1 ~]# /etc/init.d/rgmanager start
Starting Cluster Service Manager:                          [ OK ]

[root@node1 ~]# chkconfig rgmanager on

[root@node1 ~]# /etc/init.d/modclusterd start
Starting Cluster Module - cluster monitor:

[root@node1 ~]# chkconfig modclusterd on
```

We need to start all these services on **node2** as well

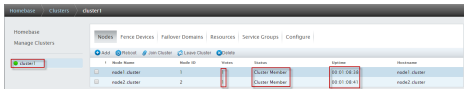
```
[root@node2 ~]# /etc/init.d/cman start
Starting cluster:
   Checking if cluster has been disabled at boot...        [  OK  ]
   Checking Network Manager...                             [  OK  ]
   Global setup...                                         [  OK  ]
   Loading kernel modules...                               [  OK  ]
   Mounting configfs...                                    [  OK  ]
   Starting cman...                                        [  OK  ]
   Waiting for quorum...                                   [  OK  ]
   Starting fenced...                                      [  OK  ]
   Starting dlm_controld...                                [  OK  ]
   Tuning DLM kernel config...                             [  OK  ]
   Starting gfs_controld...                                [  OK  ]
   Unfencing self...                                       [  OK  ]
   Joining fence domain...                                 [  OK  ]

[root@node2 ~]# chkconfig cman on
```

```
[root@node2 ~]# /etc/init.d/clvmd start
Starting clvmd:
Activating VG(s):   2 logical volume(s) in volume group "VolGroup" now active
                                                           [  OK  ]
```

```
[root@node2 ~]# /etc/init.d/rgmanager start
Starting Cluster Service Manager:                          [  OK  ]

[root@node2 ~]# chkconfig rgmanager on

[root@node2 ~]# chkconfig modclusterd on

[root@node2 ~]# /etc/init.d/modclusterd start
Starting Cluster Module - cluster monitor:
```

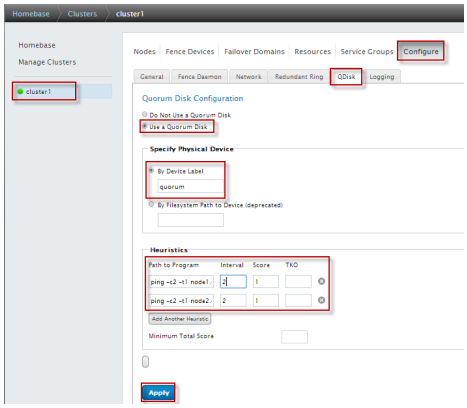
Now once all the services have started, let us refresh the web console and see the changes

So all the services are running and there is no more warning message on either cluster or the nodes

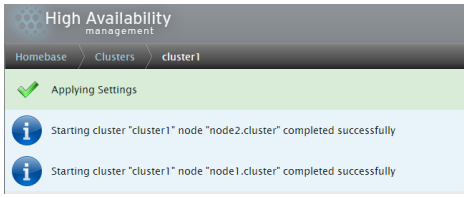


9. Configure Quorum Disk

Click on **Configure** from the **TAB** menu as shown below and select **QDisk**
Fill in the details as shown below
Check the box with "**Use a Quorum Disk**"
Provide the **label name** used in above steps while formatting Quorum disk in **Step 4**
Provide the **command** to be run to check the quorum status between all the nodes and the **interval time**
Click on **Apply** once done

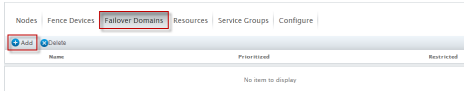


If everything goes fine you should be able to see the below message

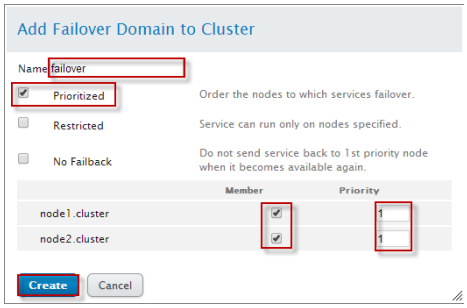


10. Configure Failover Domain

Select **Failover Domain** option from the **TAB** menu and Add a new Failover Domain

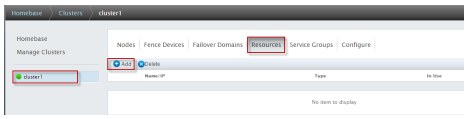


Give a name to your **failover domain** and follow the setting as shown below



11. Create Resources

Click on **Resources** TAB from the top menu and select **Add**

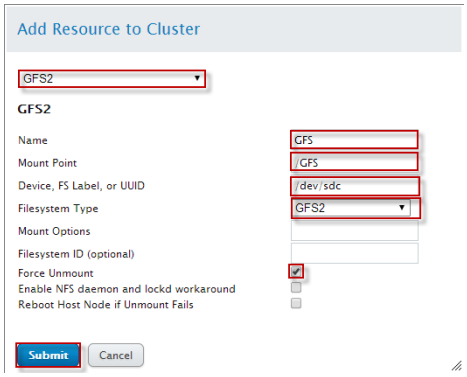


Select **GFS2** from the drop down menu and fill in the details
Name: *Give any name*
Mount Point: *Before giving the mount point make sure it exists on both the nodes*

Let us create these mount points on **node1** and **node2**

```
[root@node1 ~]# mkdir /GFS
[root@node2 ~]# mkdir /GFS
```

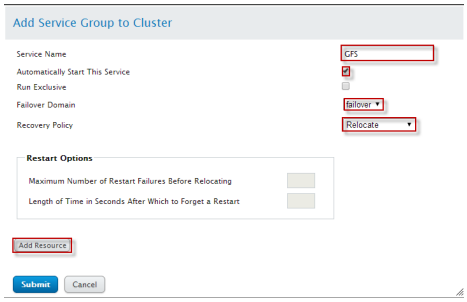
Next fill in the device details which we formatted for **GFS2** i.e. **`/dev/sdc`**
Check the **Force Unmount** box and click on **Submit**



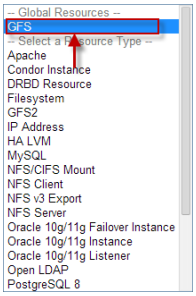
12. Create Service Group

Select **Service Group** TAB from the top menu and click on **Add**

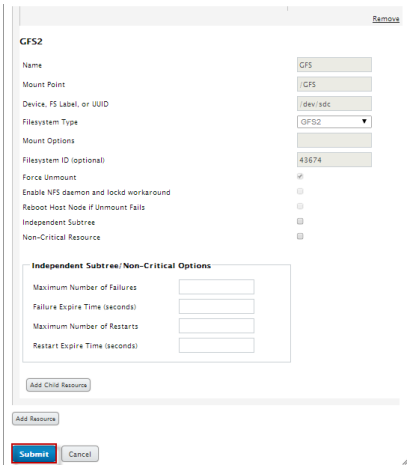
Give a **name** to your service
Check the box to **automatically** start your service
Select the **failover** which we created in **Step 10**
Select **relocate** from the drop down menu for **Recovery Policy**
Once done click on "**Add resource**"



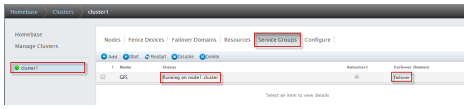
You will see the below box on your screen. Select the **Resource** we created in **Step 11**.



As soon as you select GFS, all the saved setting under GFS resource will be visible under service group section as shown below. Click on **Submit** to save the changes



Once you click on **submit**, refresh the web console and you should be able to see the **GFS** service running on your cluster on any of the node as shown below



You can verify the same from CLI also

13. Verification

On **node1**

```
[root@node1 ~]# clustat
Cluster Status for cluster1 @ Wed Feb 26 00:49:04 2014
Member Status: Quorate

 Member Name                                              ID   Status
 ------ ----                                              ---- ------
 node1.cluster                                             1 Online,  Local, rgmanager
 node2.cluster                                             2 Online,  rgmanager
 /dev/block/8:16                                           0 Online, Quorum Disk

 Service Name                                    State            Owner (Last)

 ------- ----                                    -----            ----- ------
```

```
 service:GFS                                started           node1.cluster
```

So, if GFS is running on node1 then GFS should be mounted on **/GFS** on **node1**. Let us verify

```
[root@node1 ~]# df -h
Filesystem                Size  Used Avail Use% Mounted on
/dev/mapper/VolGroup-root  8.7G  3.4G  5.0G  41% /
tmpfs                     495M   32M  464M   7% /dev/shm
/dev/sda1                 194M   30M  155M  16% /boot
/dev/sr0                  4.2G  4.2G     0 100% /media/CentOS_6.5_Final
/dev/sdc                   11G  518M  9.9G   5% /GFS
```

Now let me try to relocate the **GFS** service on **node2**

```
[root@node1 ~]# clusvcadm -r GFS -m node2
'node2' not in membership list
Closest match: 'node2.cluster'
Trying to relocate service:GFS to node2.cluster...Success
service:GFS is now running on node2.cluster
```

Let us see if the changes are reflected on cluster

```
[root@node1 ~]# clustat
Cluster Status for cluster1 @ Wed Feb 26 00:50:42 2014
Member Status: Quorate

 Member Name                                              ID   Status
 ------ ----                                              ---- ------
 node1.cluster                                             1 Online, Local, rgmanager
 node2.cluster                                             2 Online, rgmanager
 /dev/block/8:16                                           0 Online, Quorum Disk

 Service Name                                 State                Owner (Last)

 ------- ----                                 -----                ----- ------

 service:GFS                                 started              node2.cluster
```

Again to reverify on the available partitions

```
[root@node1 ~]# df -h
Filesystem                Size  Used Avail Use% Mounted on
/dev/mapper/VolGroup-root  8.7G  3.4G  5.0G  41% /
tmpfs                     495M   26M  470M   6% /dev/shm
/dev/sda1                 194M   30M  155M  16% /boot
/dev/sr0                  4.2G  4.2G     0 100% /media/CentOS_6.5_Final

On node2

[root@node2 ~]# df -h
Filesystem                Size  Used Avail Use% Mounted on
/dev/mapper/VolGroup-root  8.7G  3.4G  5.0G  41% /
tmpfs                     495M   32M  464M   7% /dev/shm
/dev/sda1                 194M   30M  155M  16% /boot
/dev/sr0                  4.2G  4.2G     0 100% /media/CentOS_6.5_Final
/dev/sdc                   11G  518M  9.9G   5% /GFS
```

Restarting Cluster Manually

In case you plan to restart your cluster manually then there is a pre-defined order in which you will have to stop and start the services

Stopping Cluster services

```
On one node only, disable services listed in clustat # clusvcadm -d GFS

Verify that all services in clustat are showing disabled
On each node: # service rgmanager stop
On each node: # service clvmd stop
On each node: # service cman stop
```

Starting Cluster services

```
On each node: # service cman start
On each node: # service clvmd start
On each node: # service rgmanager start
```

Enable the **GFS** service in any of the node

```
# clusvcadm -e GFS
```

References

Red Hat Enterprise Cluster

Related Articles

Configuring iSCSI storage using openfiler
How to install openfiler
Overview of services used in Red Hat Cluster
How to configure a Clustered Samba share using ctdb in Red Hat Cluster

Follow the below links for more tutorials

Step by Step Linux Boot Process Explained In Detail
Tutorial for Monitoring Tools SAR and KSAR with examples in Linux
How to configure Samba 4 Secondary Domain Controller
How does a DNS query works when you type a URL on your browser?
How to create password less ssh connection for multiple non-root users
How to secure Apache web server in Linux using password (.htaccess)
How to register Red Hat Linux with RHN (Red Hat Network)
Red hat Enterprise Linux 5.5 Installation Guide (Screenshots)
15 tips to enhance security of your Linux machine
Why is Linux more secure than windows and any other OS
What is the difference between "su" and "su -" in Linux?
What is swappiness and how do we change its value?
How to log iptables messages in different log file
What are the s and k scripts in the etc rcx.d directories
How to check all the currently running services in Linux
How to auto start service after reboot in Linux
What is virtual memory, paging and swap space?

8 comments:

dipanjan mukherjee 16 May 2014, 16:48:00

This is excellent tutorial regarding step by step guide of RHEL cluster setup.

Reply

Samim 29 May 2014, 17:36:00

Hi..It's really very nice post..I followed all the steps as you told, but facing few problem :

1. I'm getting two set of Disk on each node, means total two set of "quorum" & "san" disk, total four disk on each node. And that's also unordered way, if quorum showing sdb on node1 then for node2 it's showing sdc like that.

2. After created quorum disk none of the node is starting due to cman service is bringing up.

3. Where is server log and Configuration file for Cluster on node3, how can we change the configuration and check log..what's happening in back end.

Please help to understand those points and clear some doubts.

Reply

Replies

Deepak Prasad 1 Jun 2014, 22:26:00

Hello Samim,

It also happened with me, in that case try to re-discover the iscsi targets and repeat

step 3 above a few times. Also restarting the iscsi services on the openfiler will help you.

Any particular error you are getting for cman service?

You can configure log using conga. I will try to write an article on the same.

Thanks
Deepak

Reply

Muhammad Iqbal 29 May 2014, 19:15:00

Simple and Excellent

Reply

Varun Jain 17 Aug 2014, 23:06:00

Thank you ! was looking for this kind of tutorial, see how lucky am I. :)

Reply

Louis Delossantos 13 Oct 2014, 19:41:00

Why do you need to write the quorum disk twice? Doesn't the second write just re-write anything that was placed on the disk at time of the node 1's write to the quorum?

Little confused about this, since it's a shared disk, aren't you just writing data to the quorum on disk 1, then just re-writing all of it over when performing the operation with node 2?

Thanks.

Reply

Replies

Deepak Prasad 13 Oct 2014, 20:24:00

My mistake, the time I had configured this for the first time was a bit new and later forgot to update this article though I had updated the same in my new article.

But thanks for marking the mistake, I have updated the article and yes the quorum as well as GFS2 partition is formatted only on any one of the cluster node since the partitions are configured on a common shared storage the changes are reflected on all the nodes of the same cluster.

Reply

Sai Raj 23 Oct 2014, 01:35:00

nice one , your setup working for me smootly..... thanks for your intiative to prepare this beautifullll work..... billions thanks to you.....no words to expresss

Reply

Enter your comment...

Comment as: ccverak (Goog ▼ Sign out

Publish Preview Notify me