SUSE Linux Enterprise Server 11 SP3 cluster configuration for Dell PowerEdge VRTX with NFS storage

Setting up an active-passive SUSE Linux Enterprise Server (SLES) 11 service pack 3 cluster configuration for PowerEdge VRTX entry shared configuration with a NFS storage

Samir Sachdev

January 2015

A Dell Technical White Paper

Revisions

Date	Description
January 2015	Initial release

THIS WHITE PAPER IS FOR INFORMATIONAL PURPOSES ONLY, AND MAY CONTAIN TYPOGRAPHICAL ERRORS AND TECHNICAL INACCURACIES. THE CONTENT IS PROVIDED AS IS, WITHOUT EXPRESS OR IMPLIED WARRANTIES OF ANY KIND.

PRODUCT WARRANTIES APPLICABLE TO THE DELL PRODUCTS DESCRIBED IN THIS DOCUMENT MAY BE FOUND AT: http://www.dell.com/learn/us/en/19/terms-of-sale-commercial-and-public-sector Performance of network reference architectures discussed in this document may vary with differing deployment conditions, network loads, and the like. Third party products may be included in reference architectures for the convenience of the reader. Inclusion of such third party products does not necessarily constitute Dell's recommendation of those products. Please consult your Dell representative for additional information.

Trademarks used in this text:

Dell™, the Dell logo, Dell Boomi™, Dell Precision™ ,OptiPlex™, Latitude™, PowerEdge™, PowerVault™, PowerConnect™, OpenManage™, EqualLogic™, Compellent™, KACE™, FlexAddress™, Force10™ and Vostro™ are trademarks of Dell Inc. Other Dell trademarks may be used in this document. Cisco Nexus®, Cisco MDS®, Cisco NX-0S®, and other Cisco Catalyst® are registered trademarks of Cisco System Inc. EMC VNX®, and EMC Unisphere® are registered trademarks of EMC Corporation. Intel®, Pentium®, Xeon®, Core® and Celeron® are registered trademarks of Intel Corporation in the U.S. and other countries. AMD® is a registered trademark and AMD Opteron™, AMD Phenom™ and AMD Sempron™ are trademarks of Advanced Micro Devices, Inc. Microsoft®, Windows®, Windows Server®, Internet Explorer®, MS-DOS®, Windows Vista® and Active Directory® are either trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries. Red Hat® and Red Hat® Enterprise Linux® are registered trademarks of Red Hat, Inc. in the United States and/or other countries. Novell® and SUSE® are registered trademarks of Novell Inc. in the United States and other countries. Oracle® is a registered trademark of Oracle Corporation and/or its affiliates. Citrix®, Xen®, XenServer® and XenMotion® are either registered trademarks or

Contents

1 Basic minimum cluster requirements

1.1 Hardware requirements

- Two blades on PowerEdge VRTX entry shared chassis.
- PowerEdge VRTX should be in entry-shared configuration.
- One isolated external network switch connected to one port of each blade (em2 port).
- A separate system installed with SLES 11 SP3 to act as NTP Server, which is part of the same network. The two nodes are required for cluster time synchronization. This system can also be used later to mount the NFS file system created on the cluster.

1.2 Software requirements

- SLES 11 SP3 must be deployed, with default pattern, on each blade.
- SLES HA package must be installed from the accompanying ISO image.
- All the steps must be performed as a root user on the systems.

2 Initial setup

Perfom the following procedures:

- Network Setup
- Plasma Shared Storage Setup
- MegaRAID Driver Installation for SPERC

2.1 Setting up the network

To set up the network:

1. Assign static address to the port in the same subnet range for **Node 1 (192.168.1.1)** and **Node 2 (192.168.1.2)**.
2. Assign Workstation Node IP (192.168.1.254).

   Node 1 (hostname): **sles-ha**

   Node 2 (hostname): **sles-ha-1**

3. Network Settings can be configured by accessing in **/etc/hosts** on both nodes and adding following entries on both nodes:

   ```
   192.168.1.1        sles-ha        sles-ha
   192.168.1.2        sles-ha-1      sles-ha-1
   ```

4. Disable IPv6 from the blades by accessing **Network Settings > Global Options > Disable IPv6**.

2.2 Setting up the PowerEdge VRTX shared storage setup (entry-shared)

PowerEdge VRTX should be in an single controller entry-shared mode with at least two virtual disks configured and in **Multiple Assignment** mode. Both virtual disks must be assigned to both Node 1 and Node 2.

- VD1 is used as a `STONITH` drive
- VD2 is used as NFS storage managed by Cluster

2.3 Installing MegaRAID driver for Shared PERC Controller (SPERC)

Firmware on the SPERC should be (23.8.10-0059) for the Entry-Shared configuration.

1. Ensure that the drivers are exactly built for the kernel on the system for SLES 11 SP3 (3.0.76-0.11). This can be verified using the following command:

```
# uname –a
```

2. Deploy a unified driver (6.803.00.00) on both the nodes using the following command:

   ```
   # rpm –ivh lsi-megaraid_sas-kmp-default-v06.803.00.00_3.0.76-0.11-
   1.x86_64.rpm
   ```

Installation of the driver is completed. You can use the following command verify driver installation:

```
sles-ha-1:~ #lsscsi

[0:0:32:0]   enclosu DP       BP12G+          1.00  -

[0:2:0:0]    disk    DELL     PERC H310       2.12  /dev/sdd

[1:2:0:0]    disk    DELL     Shared PERC8    3.18  /dev/sda

[1:2:1:0]    disk    DELL     Shared PERC8    3.18  /dev/sdb
```

The command above should list the local PERC Controller and the SPERC with the virtual disks associated to the SPERC.

3 High availability and cluster setup

3.1 Installing SLE-HA extension

For these steps, we are using SLE-HA as distributed in the separate ISO `SLE-HA-11-SP3-x86_64-GM-CD1.iso`.

To install this SLE-HA add-on:

1. Start YaST.
2. Select **Software** > **Add-On Products**.
3. Select the local ISO Image and then specify the path to ISO image.

3.2 Setting up the automatic cluster

1. On Node 1, run the following command to start the bootstrap script:

```
# sleha-init
```

If NTP is not configured on the nodes, a warning is displayed. Ignore the warning and continue. It is not mandatory to specify the data in all the fields. You can specify the cluster settings manually at a later stage..

2. Specify a shared virtual disk's WWID in your node when prompted. For example, you can specify the MR VD's WWID in the following format:

```
/dev/disk/by-id/wwn-<id of the Shared VD>
```

3. On Node 2, run the following command to start the bootstrap script:

```
# sleha-join
```

4. Provide the IP of the initiating node, in this scenario Node-1 (192.168.1.1) to complete the cluster setup on Node 2.

3.3 Specifying cluster settings

Once the automatic cluster is setup using the bootstrap scripts, you must specify additional settings to the cluster, which has not been performed during bootstrap.

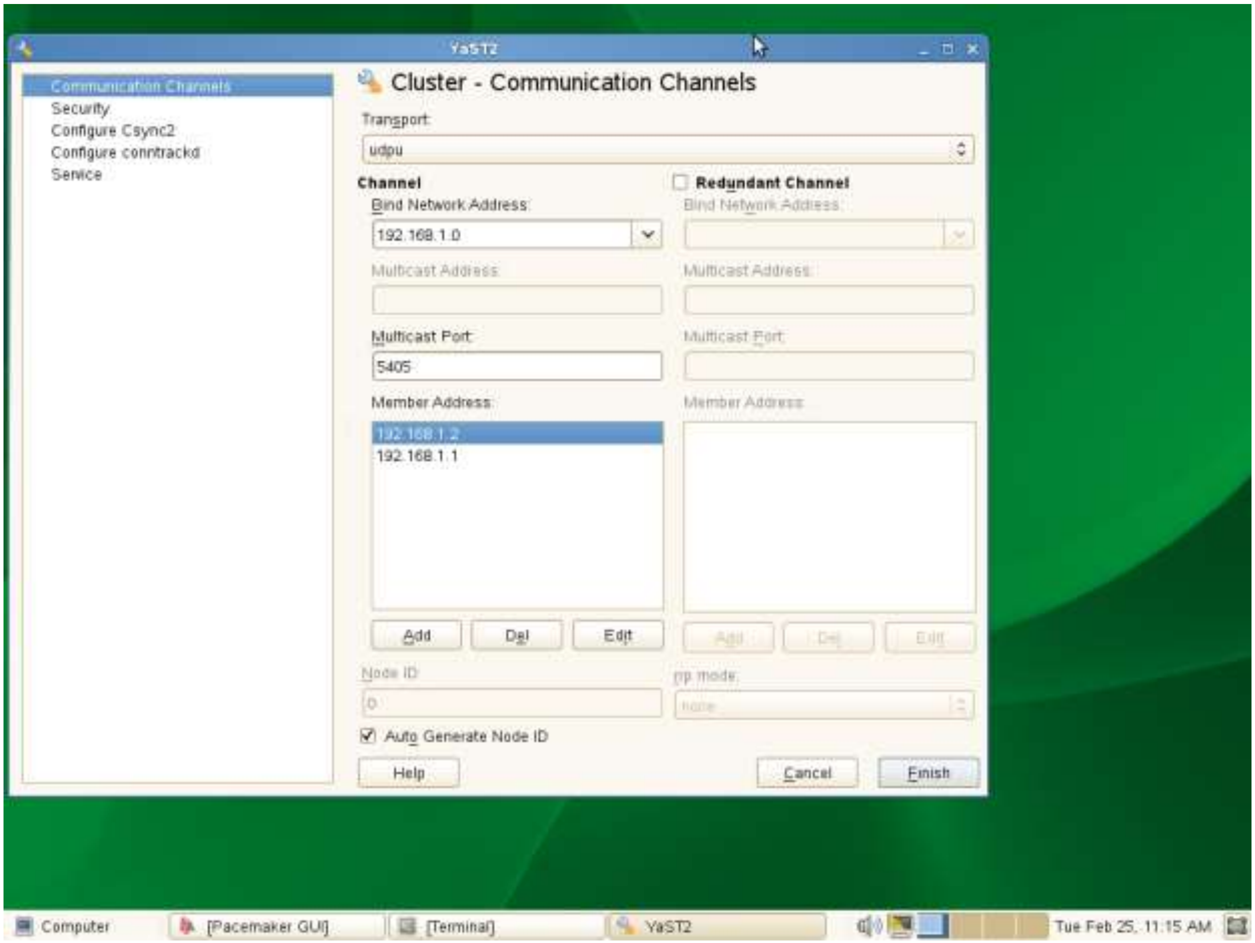
1. Start the cluster module using the following command:

```
# yast2 cluster
```

2. In Cluster - Communication Channel tab select `udpu` from the **Transport** drop-down list and clear the `Redundant Channel` field.
3. You can set the **Bind Network Address as 192.168.1.0,** because the range used is 192.168.1.x.

Ensure that the Multicast Port field is set to the default value of 5405.

4. Click **Add** and then specify the following IP address in **Member address** field:
   - **192.168.1.1**
   - **192.168.1.2**



Cluster - Communication Channels

Transport: udpu

Channel

Bind Network Address: 192.168.1.0

Multicast Address:

Multicast Port: 5405

Member Address:
192.168.1.2
192.168.1.1

☐ Redundant Channel

Bind Network Address:

Multicast Address:

Multicast Port:

Member Address:

Node ID: 0

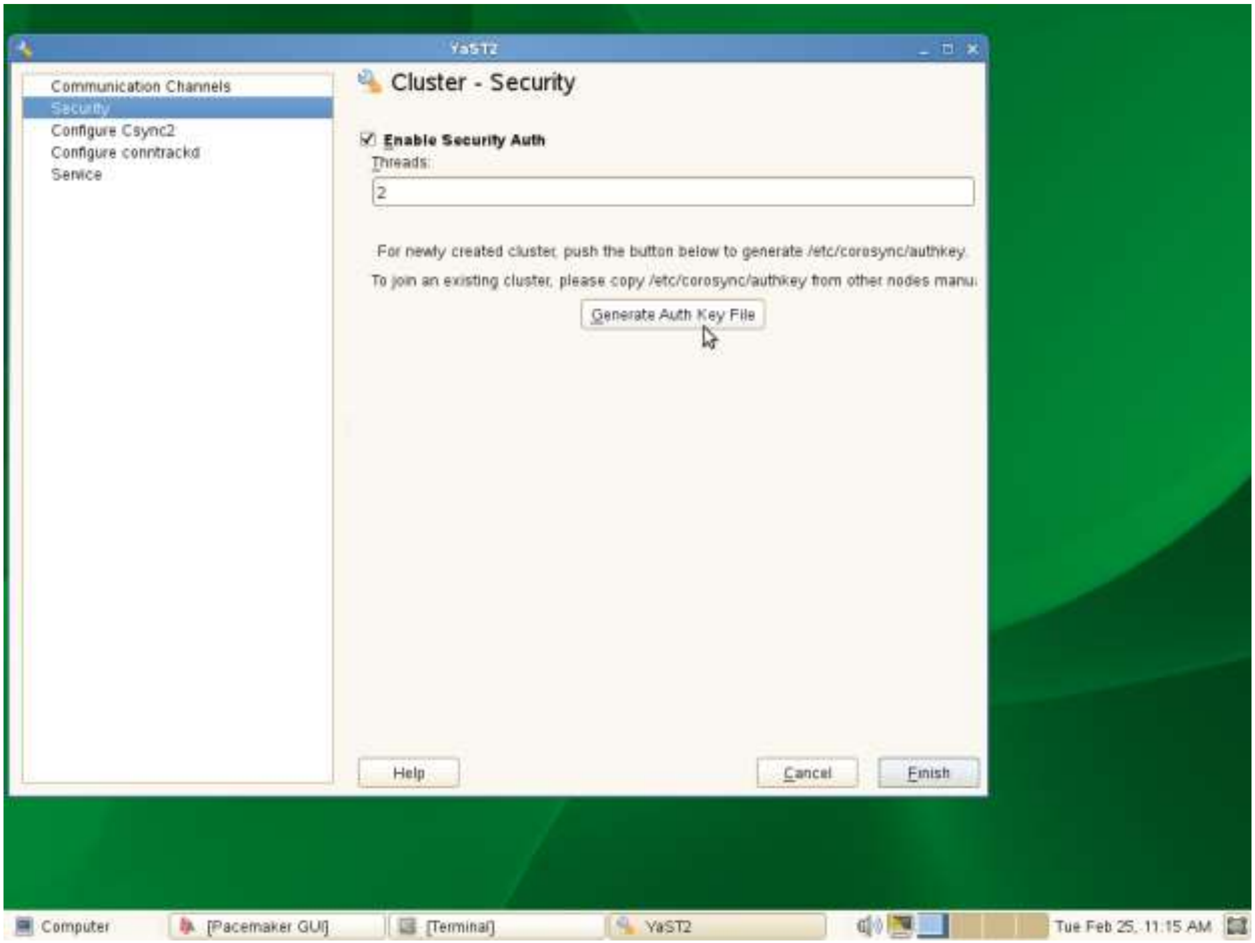
rrp mode: none

☑ Auto Generate Node ID

Figure 1 Cluster - Communication channel tab

5. Select the **Auto Generate Node ID** check box to automatically generate a unique ID for every cluster node.
6. Click **Finish** to confirm the changes for an existing cluster and close the cluster module. YaST writes the configuration to `/etc/corosync/corosync.conf` file.
7. In the Cluster - Security tab, select the **Enable Security Auth** check box.
8. Specify the threads value as 2 in the **Threads** field.
9. Click **Generate Auth Key File** to create an authentication key.

The key is written to the `etc/corosync/authkey` file. The key should be generated only on Node 1. To join the Node 2 to the cluster, the `authkey` file should be manually copied from Node 2 from Node 1.



Cluster - Security

☑ Enable Security Auth

Threads:

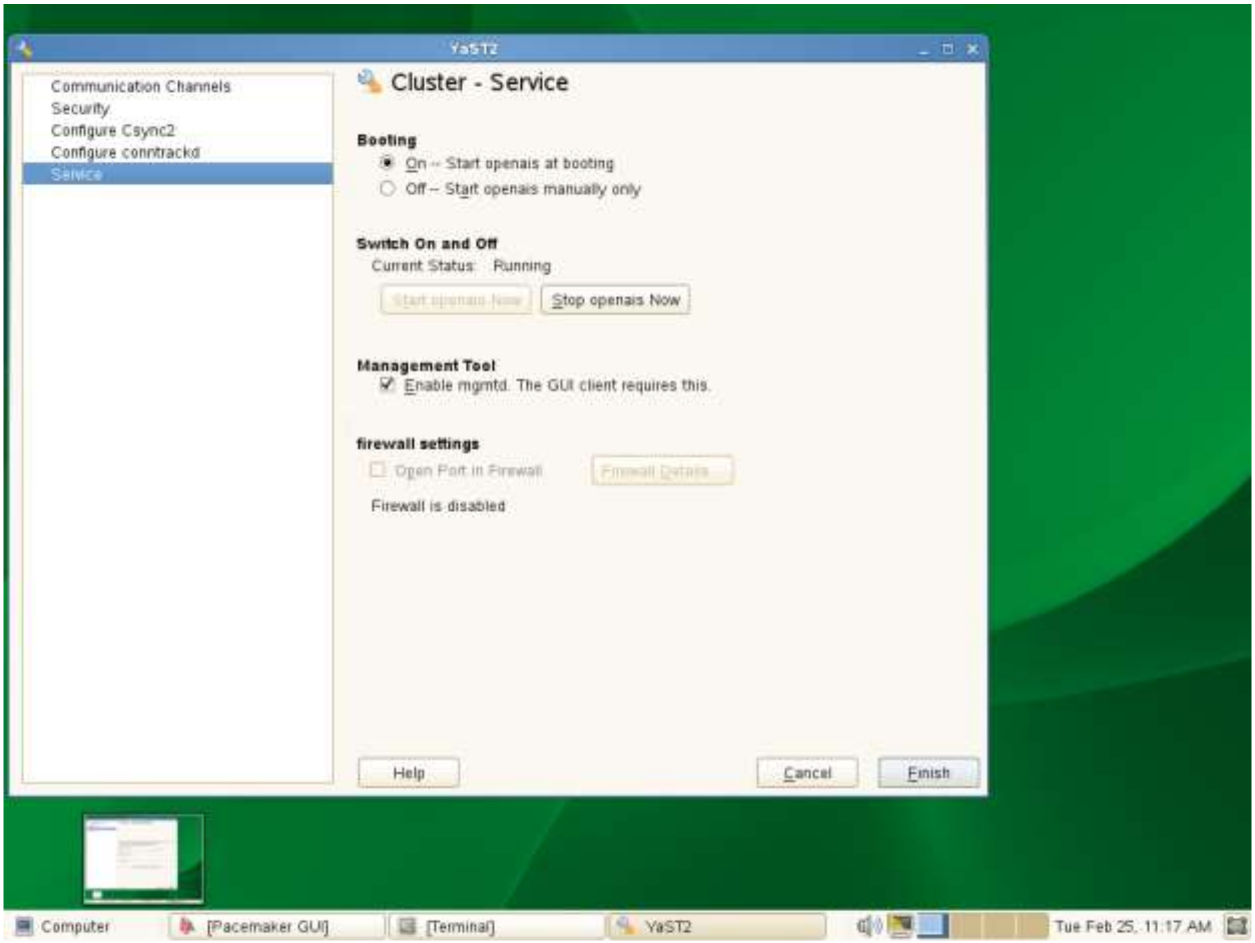
2

For newly created cluster, push the button below to generate /etc/corosync/authkey.

To join an existing cluster, please copy /etc/corosync/authkey from other nodes manu

Figure 2 Cluster-Security tab

10. In the Cluster – Service tab, select the **On – Start openais at booting** radio button as the **Booting** option.
11. Click **Finish**.



Cluster - Service

Booting

● On -- Start openais at booting

○ Off -- Start openais manually only

Switch On and Off

Current Status: Running

Management Tool

☑ Enable mgmtd. The GUI client requires this.

firewall settings

□ Open Port in Firewall

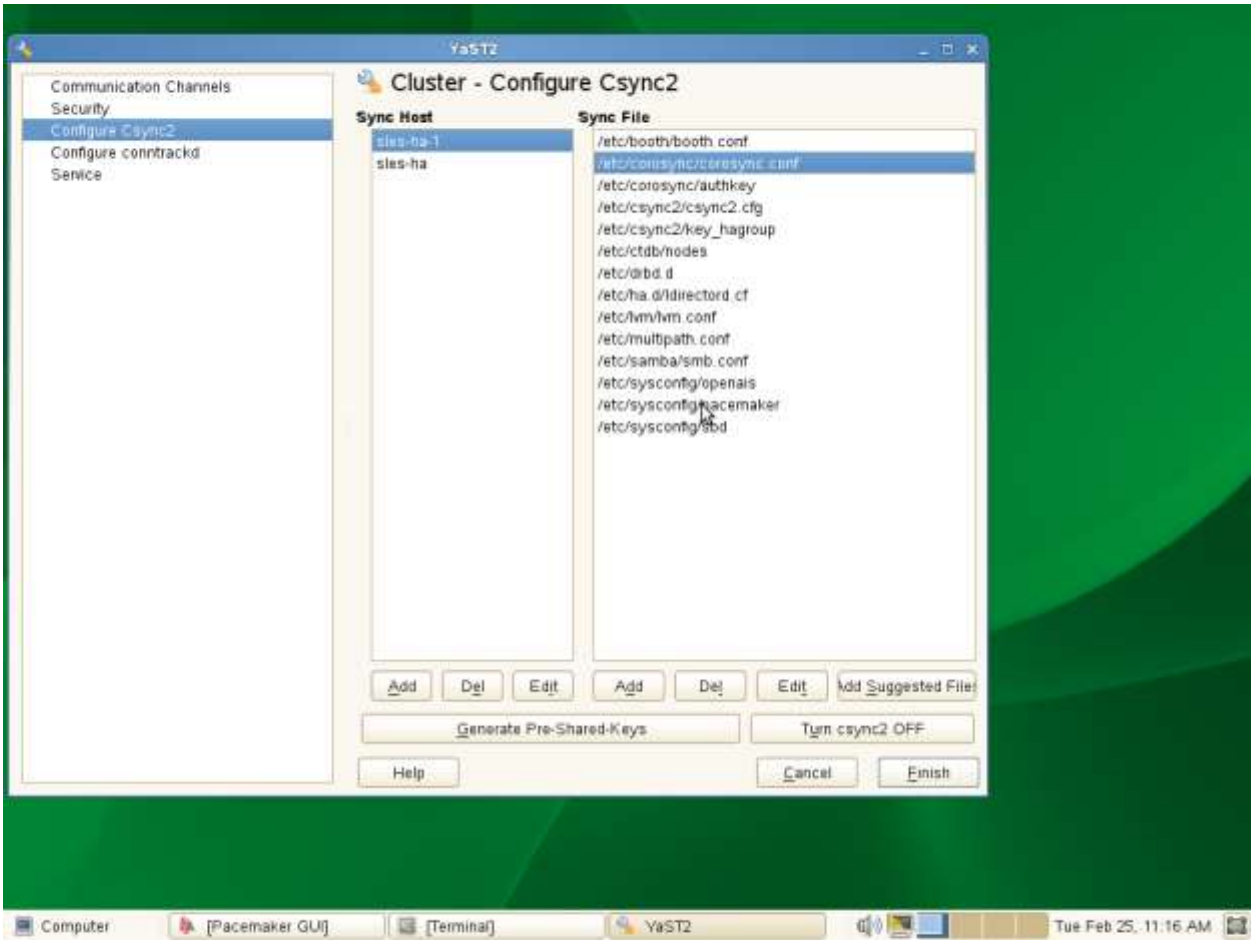
Firewall is disabled

Figure 3 Cluster-Service

12. In the Cluster - Configure Csync2 tab, click **Add** in the **Sync Host group** field and specify the local host names of all nodes in the cluster, to specify the synchronization group.

For each node, the strings should be exactly as returned by `hostname` command.

In the example displayed in the figure, the strings are sles-ha-1 and sles-ha for Node 2 and Node 1 respectively.



Sync Host

sles-ha-1
sles-ha

Sync File

/etc/booth/booth.conf
/etc/corosync/corosync.conf
/etc/corosync/authkey
/etc/csync2/csync2.cfg
/etc/csync2/key_hagroup
/etc/ctdb/nodes
/etc/drbd.d
/etc/ha.d/ldirectord.cf
/etc/lvm/lvm.conf
/etc/multipath.conf
/etc/samba/smb.conf
/etc/sysconfig/openais
/etc/sysconfig/pacemaker
/etc/sysconfig/sbd

Figure 4 Cluster-Configure Csync2

13. Click **Generate Pre-Shared-Keys** to create a key file for the synchronization group. The key file is written to `/etc/csync2/key_hagroup`. This file should also be copied to Node 2 from Node 1.
14. Click **Csync2** to activate **Turn csync2 ON**. The **chkconfig csync2** starts the **Csync2** automatically at boot time.

3.3.1 Synchronizing the configuration file with Csync2

To successfully synchronize the files with Csync2, ensure that the same csync2 configuration should be available on all nodes.

1. Copy the file `/etc/csync2/csync2.cfg` manually to Node 2 after the cluster configuration is completed and the files mentioned in the *Specifying cluster settings* section are generated.

   Summary of files to be copied for Synchronization from Node 1 to Node 2 in the corresponding directories is listed below:

   ```
   /etc/corosync/corosync.conf
   ```

   ```
   /etc/corosync/authkey
   ```

```
/etc/csync2/key_hagroup

/etc/csync2/csync2.cfg
```

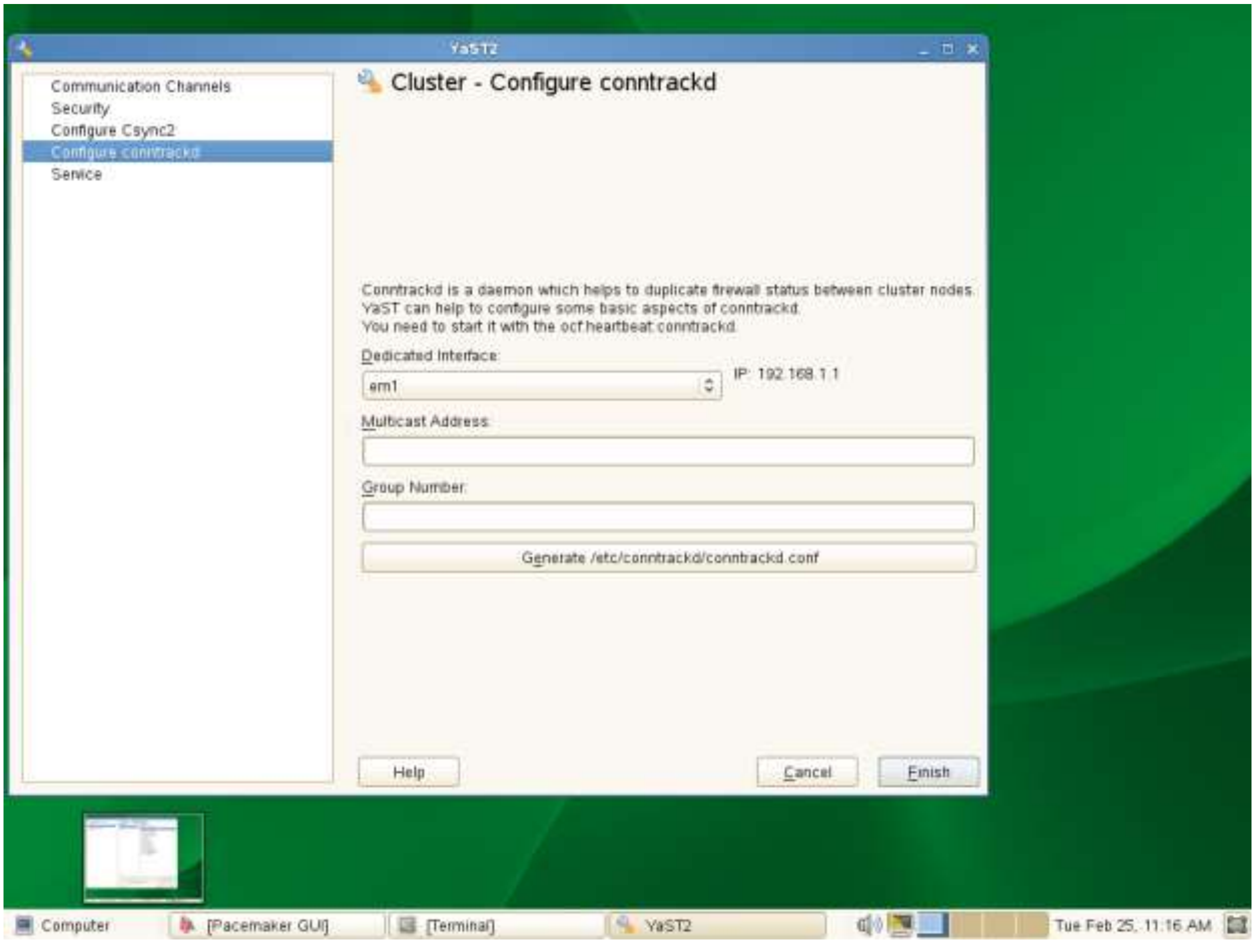
2. Both must be running on both nodes. Run the following command on all nodes to start both `csync2` and `xinetd` services automatically on both the nodes, when the system is started.

```
# chkconfig csync2 on

# chkconfig xinetd on

# rcxinetd start
```

The services start.



Cluster - Configure conntrackd

Conntrackd is a daemon which helps to duplicate firewall status between cluster nodes.
YaST can help to configure some basic aspects of conntrackd.
You need to start it with the ocf:heartbeat:conntrackd.

Dedicated Interface: em1 IP: 192.168.1.1

Multicast Address:

Group Number:

Figure 5 Cluster-Configure conntracked

Do not modify any of the default options in the Cluster-Configure conntracked tab as the firewall is disabled.

3.4 Bringing the cluster online

1. Check if `openais` service is currently running by running the following command:

   ```
   # rcopenais status
   ```

2. If the `openais` service is not running then start OpenAIS/Corosync by running the following command:

   ```
   # rcopenais start
   ```

3. Repeat steps 1 and 2 for all nodes.
4. To check the status of cluster, run following command:

   ```
   # crm_mon
   ```

   If all nodes are online, the output should be similar to the following:

   ```
   Last updated: Wed Feb 26 13:27:04 2014

   Last change: Mon Feb 24 11:23:55 2014 by root via cibadmin on sles-ha

   Stack: classic openais (with plugin)

   Version: 1.1.9-2db99f1

   2 Nodes configured, 2 expected votes

   Online: [ sles-ha sles-ha-1 ]
   ```

3.5 Configuring management workstation as NTP server for time synchronization

Configure both the nodes to use the NTP server in the network for synchronizing the time.

1. Click **YaST** -> **Network Services.**
2. Add the following entry to all the nodes (**Node 1** and **Node 2** in our case) in the `/etc/ntp.conf` to point to the NTP server in the network:

   ```
   server 192.168.1.254
   ```

3. Comment out rest of the lines that have a server name and IP address.

4 NFS failover setup

4.1 Installing NFSSERVER

Install **nfs-kernel-server** and all its dependencies on both Nodes (YaST Software Management). Alternatively run the following command:

```
# zypper install nfs-kernel-server
```

4.2 Setting up partition and filesystem

1. If `/dev/sda` and `/dev/sdb` are the two virtual disks available on SPERC, use the `fdisk` utility to create a single partition on both virtual disks as `/dev/sda1` and `/dev/sdb1`. In the current setup `/dev/sda1` is used as a `STONITH_SBD` drive and `/dev/sdb1` is used as `NFS Mount Storage`.

   ```
   /dev/sda1 – STONITH (SBD Cluster Storage Resource)

   /dev/sdb1 – NFS Mount
   ```

2. To make these drives usable create a partition of full size on these drives. The `/dev/sda1` drive is created for `STONITH` and `/dev/sdb1` for `NFS`.
3. Once the drives are partitioned, run the following command to format the drives with EXT3 filesystem.

   ```
   sles-ha:~ # mkfs.ext3 /dev/sda1

   sles-ha:~ # mkfs.ext3 /dev/sdb1
   ```

4.3 Setting up stonith_sbd

1. Configure Stonith_sbd accurately using the following command, as it is the fencing mechanism used in SLES-HA. The wwn refers to `/dev/sda1` that is outlined in the *Setting up partition and filesystem* section.

   ```
   sles-ha:~# ls -l /dev/disk/by-id/ | grep wwn.*sd.1
   ```

   The command lists all the available disks by their WWN IDs.

   > Ensure that you use only the wwn-<id> handle for configuring stonith_sbd. The /dev/sda1 handle is not persistent and using the handle might cause sbd unavailability after a reboot.

2. Run the following command:

   ```
   sles-ha:~# sbd –d /dev/disk/by-id/wwn-<Id of STONITH disk>-part1 -4 20 -1
   10 create
   ```

   After running the command the following response is displayed:

```
/dev/disk/by-id/wwn-<id>
```

3. Run following command to obtain details on the parameters applied, :

   ```
   sles-ha:~# sbd –d /dev/disk/by-id/wwn-<id of STONITH disk>-part1 dump
   ```

   The parameters for the Stonith-sbd to be configured are listed.

4. Verify that the contents of `/etc/sysconfig/sbd` are updated and that an entry for sbd device from the previously performed commands is reflected. Run the following command to can verify the contents:

   ```
   sles-ha:~# cat /etc/sysconfig/sbd

   SBD_DEVICE=”/dev/disk/by-id/wwn-<id>-part1”

   SBD_OPTS=”-W”
   ```

5. Copy the `/etc/sysconfig/sbd` file to the rest of nodes. In the current example, copy it to Node 2 using the following command:

   ```
   sles-ha:~# scp /etc/sysconfig/sbd root@sles-ha-1:/etc/sysconfig
   ```

6. Run the following command to allocate sbd resource to Node 1:

   ```
   sles-ha:~# sbd –d /dev/disk/by-id/wwn-<id>-part1 allocate sles-ha
   ```

7. Run the following command to allocate sbd resource to Node 2:

   ```
   sles-ha:~# sbd –d /dev/disk/by-id/wwn-<id>-part1 allocate sles-ha-1
   ```

8. Run the following command to confirm that the resource has been successfully allocated to both the nodes:

   ```
   sles-ha:~# sbd –d /dev/disk/by-id/wwn-<id>-part1 list

   0  sles-ha           clear

   1  sles-ha-1   clear
   ```

9. Run the following command to restart the `OpenAIS/corosync` daemon on both nodes:

   ```
   sles-ha:~# rcopenais restart

   sles-ha-1:~# rcopenais restart
   ```

10. Run the following command to verify that both the nodes are able to communicate to each other through sbd:

    ```
    sles-ha:# sbd -d /dev/disk/by-id/wwn-<id>-part1 message sles-ha-1 test
    ```

11. Run the following command on separate terminals of both nodes to verify that the communication succeeds:

```
sles-ha-1: # tail -f /var/log/messages
```

12. Run the following command to send message from `sles-ha-1` to `sles-ha`. Press CRTL + C to exit from the previous command.

```
sles-ha-1:# sbd -d /dev/disk/by-id/wwn-<id>-part1 message sles-ha test
```

13. Once message communication is tested and successful, configure stonith_sbd as a resource using 'the `crm` command line utility.

```
sles-ha:# crm configure

crm(live)configure# property stonith-enabled="true"

crm(live)configure# property stonith-timeout="70s"

crm(live)configure# primitive stonith_sbd stonith:external/sbd params
sbd_device="/dev/disk/by-id/<id>-part1"

crm(live)configure# commit

crm(live)configure# quit
```

You can modify the cluster policy setting as shown below:

```
sles-ha:# crm configure

crm(live)configure# property no-quorum-policy="ignore"

crm(live)configure# rsc_defaults resource-stickiness="100"

crm(live)configure# commit

crm(live)configure# quit
```

4.4 Adding NFS cluster resource

1. Create mount folders on both nodes (sles-ha and sles-ha-1).

```
sles-ha:# mkdir /nfs

sles-ha:# mkdir /nfs/part2


sles-ha-1:# mkdir /nfs

sles-ha-1:# mkdir /nfs/part2
```

The virtual disk mentioned in this section is the second virtual disk to be created. The details mentioned in this section are specific to the `/dev/sdb1` virtual disk created and formatted in the previous sections. The `/dev/disk/by-id/wwn-<id>` is the qualifier for the second drive instead of `/dev/sdb1`.

This drive being created is not a Quorum Drive.
Ensure that you do not manually mount this ext3 partition Mounting the partition manually corrupts the file system.

2. Add the following contents to `/etc/exports` on both `sles-ha` and `sles-ha-1` using a text file editor program such as **vi**.

```
sles-ha:~ # vi /etc/exports
```

```
/nfs/part2 *(fsid=1,rw,no_root_squash,mountpoint)
```

Ensure that there no a space between “*” and “(“ in the `/etc/exports` file.

3. Configure the NFSSERVER to be started or stopped by the cluster.

```
sles-ha:~ # crm configure
```

```
crm(live)configure# primitive lsb_nfsserver lsb:nfsserver op monitor
interval="15s" timeout=”15s”
```

4. Configure a File system service

```
crm(live)configure# primitive p_fs_part2 ocf:heartbeat:Filesystem params
device=/dev/disk/by-id/wwn-<id of data volume>-part1 directory=/nfs/part2
fstype=ext3 op monitor interval="10s"
```

5. Configure a Virtual IP address. The IP address is different from the IP address is binded to the Ethernet ports. This IP address moves between both the nodes. Specify the netmask according to your network. For the current example, us 192.168.1.100 as floating IP address for NFS Mount available to be mounted.

```
crm(live)configure# primitive p_ip_nfs ocf:heartbeat:IPaddr2 params
ip=192.168.1.100 cidr_netmask=24 op monitor interval="30s"
```

6. Create a group and add the resources part of the same group. Ensure that stonith_sbd is not part of this group.

```
crm(live)configure# group g_nfs p_fs_part2
```

```
crm(live)configure# edit g_nfs
```

A text editor is displayed. Modify the group g_nfs as mentioned below:

```
group g_nfs p_ip_nfs p_fs_part2 lsb_nfsserver
```

```
crm(live)configure# commit
```

```
crm(live)configure# quit
```

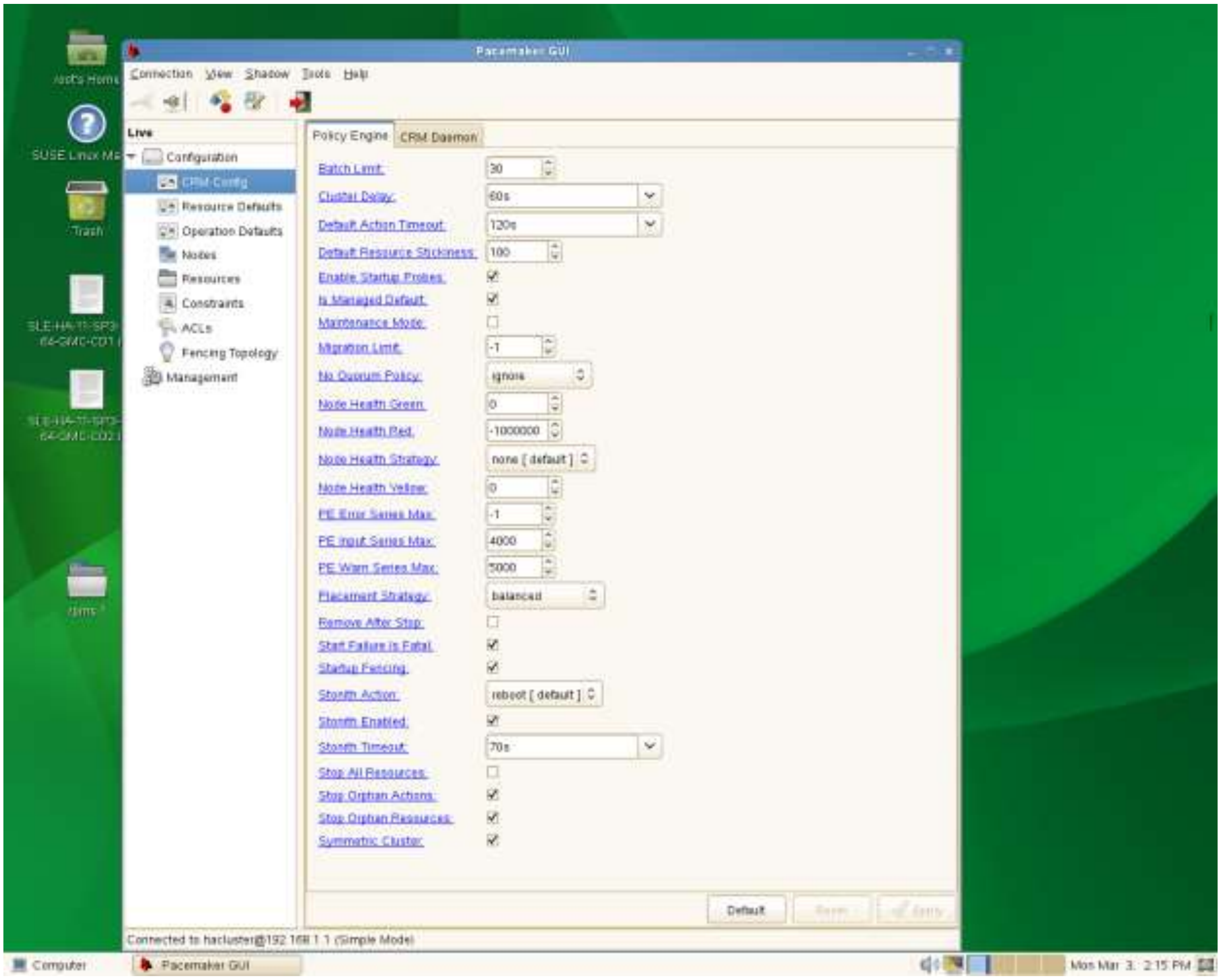
7. Verify that the resources are added and the parameters are set to the value as specified below:

```
sles-ha-2:~ # crm configure show
node sles-ha
node sles-ha-1
primitive lsb_nfsserver lsb:nfsserver \
   op monitor interval="15s" timeout="15s" \
   meta target-role="Started"
primitive p_fs_part2 ocf:heartbeat:Filesystem \
    params device="/dev/disk/by-id/wwn-0x6b8ca3a0edb7d4001a96364e5ea4dbf2-
part1"   directory="/nfs/part2" fstype="ext3" \
   op monitor interval="10s" \
   meta target-role="Started"
       primitive p_ip_nfs ocf:heartbeat:IPaddr2 \
   params ip="192.168.1.100" cidr_netmask="24" \
   op monitor interval="30s"
       primitive stonith_sbd stonith:external/sbd \
   params sbd_device="/dev/disk/by-id/wwn-
0x6b8ca3a0edb7d4001a96362a5c843538-part1"
       group g_nfs p_ip_nfs p_fs_part2 lsb_nfsserver \
   meta target-role="Started"
       property $id="cib-bootstrap-options" \
   no-quorum-policy="ignore" \
   placement-strategy="balanced" \
   dc-version="1.1.9-2db99f1" \
   cluster-infrastructure="classic openais (with plugin)" \
   expected-quorum-votes="2" \
   stonith-timeout="70s" \
   default-action-timeout="120s" \
   default-resource-stickiness="100" \
   last-lrm-refresh="1393026384"
       rsc_defaults $id="rsc-options" \
   resource-stickiness="100" \
   migration-threshold="3"
      op_defaults $id="op-options" \
   timeout="600" \
   record-pending="false"
```

8. Verify that the values of the cluster are set as displayed in following images. Run the following command to open the Cluster configuration GUI:

   ```
   sles-ha-1:~ # crm_gui
   ```

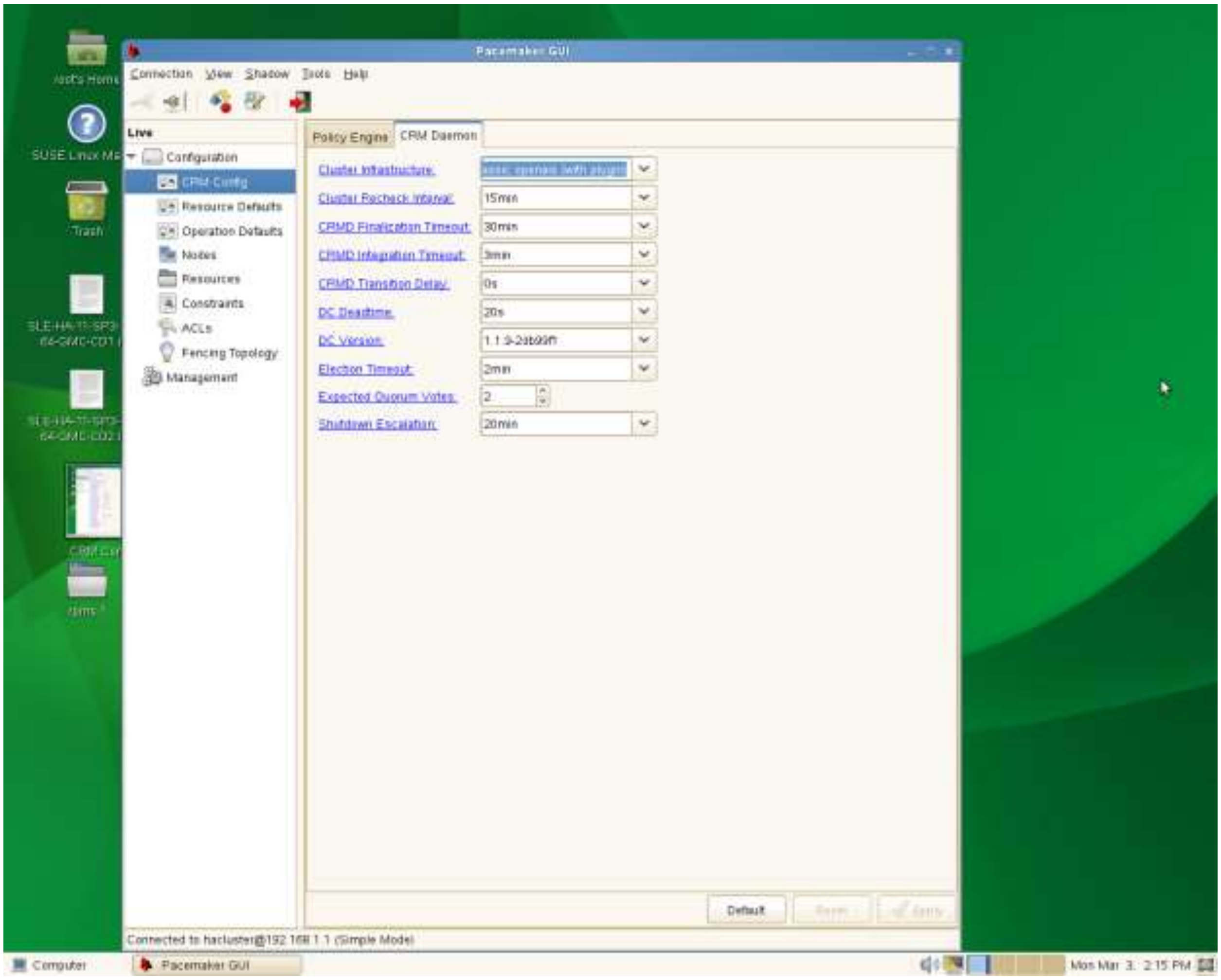
9. Specify `hacluster` as the default user name and `linux` as the default password for the cluster.



Batch Limit: 30
Cluster Delay: 60s
Default Action Timeout: 120s
Default Resource Stickiness: 100
Enable Startup Probes: ☑
Is Managed Default: ☑
Maintenance Mode: □
Migration Limit: -1
No Quorum Policy: ignore
Node Health Green: 0
Node Health Red: -1000000
Node Health Strategy: none [default]
Node Health Yellow: 0
PE Error Series Max: -1
PE Input Series Max: 4000
PE Warn Series Max: 5000
Placement Strategy: balanced
Remove After Stop: □
Start Failure is Fatal: ☑
Startup Fencing: ☑
Stonith Action: reboot [default]
Stonith Enabled: ☑
Stonith Timeout: 70s
Stop All Resources: □
Stop Orphan Actions: ☑
Stop Orphan Resources: ☑
Symmetric Cluster: ☑

Connected to hacluster@192.168.1.1 (Simple Mode)

Figure 6 CRM configuration



Cluster Infrastructure: assic openais (with plugin
Cluster Recheck Interval: 15min
CRMD Finalization Timeout: 30min
CRMD Integration Timeout: 3min
CRMD Transition Delay: 0s
DC Deadtime: 20s
DC Version: 1.1.9-2db99f1
Election Timeout: 2min
Expected Quorum Votes: 2
Shutdown Escalation: 20min

Connected to hacluster@192.168.1.1 (Simple Mode)

Figure 7 CRM configuration

10. Verify that the cluster is configured as expected, by running the following command.

```
sles-ha-1:# crm_mon

Last updated: Wed Feb 26 13:27:04 2014

Last change: Mon Feb 24 11:23:55 2014 by root via cibadmin on sles-ha

Stack: classic openais (with plugin)

Current DC: sles-ha - partition with quorum

Version: 1.1.9-2db99f1

2 Nodes configured, 2 expected votes

4 Resources configured.
```

```
Online: [ sles-ha sles-ha-1 ]

Resource Group: g_nfs

p_ip_nfs   (ocf::heartbeat:IPaddr2):       Started sles-ha

p_fs_part2 (ocf::heartbeat:Filesystem):    Started sles-ha

lsb_nfsserver     (lsb:nfsserver):         Started sles-ha

stonith_sbd     (stonith:external/sbd): Started sles-ha-1
```

4.5 Mounting NFS on a remote client

On the remote system, run the following command to mount the exported NFS partition.

```
mount -t nfs 192.168.1.100:/nfs/part2 /srv/nfs/part2
```

In the example in this document a workstation is used to mount the NFS partition and the IO mounts locally management station.

5 References

- LSI Syncro CS SLES Cluster Setup Reference Guide
- SUSE Linux Enterprise High Availability Extension 11 SP3 SLEHA 11 SP3 High Availability Guide.[https://www.suse.com/documentation/sle_ha/singlehtml/book_sleha/book_sleha.html]