

HPSA and EP over Redhat Cluster Suite

Installation guide

Release v.5.1



Legal Notices

Warranty.

Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

Restricted Rights Legend.

Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Copyright Notices.

©Copyright 2001-2011 Hewlett-Packard Development Company, L.P., all rights reserved.

No part of this document may be copied, reproduced, or translated to another language without the prior written consent of Hewlett-Packard Company. The information contained in this material is subject to change without notice.

Trademark Notices.

Java™ is a U.S. trademark of Sun Microsystems, Inc.

Linux is a U.S. registered trademark of Linus Torvalds

Microsoft® is a U.S. registered trademark of Microsoft Corporation.

Oracle® is a registered U.S. trademark of Oracle Corporation, Redwood City, California.

UNIX® is a registered trademark of the Open Group.

Windows® and MS Windows® are U.S. registered trademarks of Microsoft Corporation.

All other product names are the property of their respective trademark or service mark holders and are hereby acknowledged.

Document id:

Table of Contents

1 Introduction	7
1.1 Definitions	8
1.1.1 Acronyms	8
2 Prerequisites	9
2.1 Installing HPSA and EP for Cluster	9
3 Hardware example	10
3.1 Linux Server (for each node)	10
3.2 SAN Storage	10
4 Redhat Cluster Suite installation	11
4.1 Server update	11
4.2 Firewall	11
4.3 Network	11
4.4 SELinux	13
4.5 SSH Trust	13
4.6 Cluster package installation	14
4.7 Quorum disk configuration	14
4.8 Cluster services	14
4.9 Web gui interface	14
5 Redhat Cluster Suite configuration	16
5.1 Login luci	16
5.2 Cluster creation	16
5.3 Quorum	18
5.4 Fencing device	20
6 Configuring HPSA cluster services	22
6.1 HPSA prerequisites	22
6.2 HPSA script resource	22
6.3 HPSA Failover domain	24
6.4 HPSA service	24
6.5 IP Virtual	25
7 Configuring EP over Redhat Cluster Suite	27
7.1 Solution Container	27
7.2 SOSA	30
7.3 ECP	36
7.4 LockManager	40
8 Miscellaneous	45
8.1 Configure a SCSI target	45
8.2 Create SCSI target using Openfiler	47
8.2.1 Configure the Openfiler	47
8.2.2 Create SCSI target	51

In This Guide

This document is a guide for the process of installation HPSA and Extension Pack over Redhat Cluster Suite.

Audience

The audience for this guide is the Solutions Integrator (SI). The SI has a combination of some or all of the following capabilities:

Understands and has a solid working knowledge of:

- UNIX® commands

Understands networking concepts and language

Is able to program in Java™ and XML

Understands security issues

Understands the customer's problem domain

References

Manual Organization

This guide contains the following chapters:

Chapter 1, "Introduction"

Chapter 2, "Prerequisites"

Chapter 3, "Hardware example"

Chapter 4, "Redhat Cluster Suite installation"

Chapter 5, "Redhat Cluster Suite configuration"

Chapter 6, "Configuring HPSA cluster services"

Chapter 7, "Configuring EP over Redhat Cluster Suite"

Chapter 8, "Miscellaneous"

Conventions

The following typographical conventions are used in this guide.

Font	What the Font Represents	Example
<i>Italic</i>	Book or manual titles, and man page names	Refer to the <i>HP Service Activator — Workflows and the Workflow Manager</i> and the <i>Javadocs</i> man page for more information.
	Provides emphasis	You <i>must</i> follow these steps.
	Specifies a variable that you must supply when entering a command	Run the command: InventoryBuilder <sourceFiles>
	Parameters to a method	The <i>assigned_criteria</i> parameter returns an ACSE response.
Bold	New terms	The distinguishing attribute of this class...
Computer	Text and items on the computer screen	The system replies: <code>Press Enter</code>
	Command names	Use the <code>InventoryBuilder</code> command ...
	Method names	The <code>get_all_replies()</code> method does the following...
	File and directory names	Edit the file <code>\$ACTIVATOR_ETC/config/mwfm.xml</code>
	Process names	Check to see if <code>mwfm</code> is running.
	Window/dialog box names	In the <code>Test and Track</code> dialog...
	XML tag references	Use the <code><DBTable></code> tag to...
Computer Bold	Text that you must type	At the prompt, type: <code>ls -l</code>
Keycap	Keyboard keys	Press Return .
[Button]	Buttons on the user interface	Click [Delete]. Click the [Apply] button.
Menu Items	A menu name followed by a colon (:) means that you select the menu, then the	Select <code>Locate:Objects->by Comment</code> .

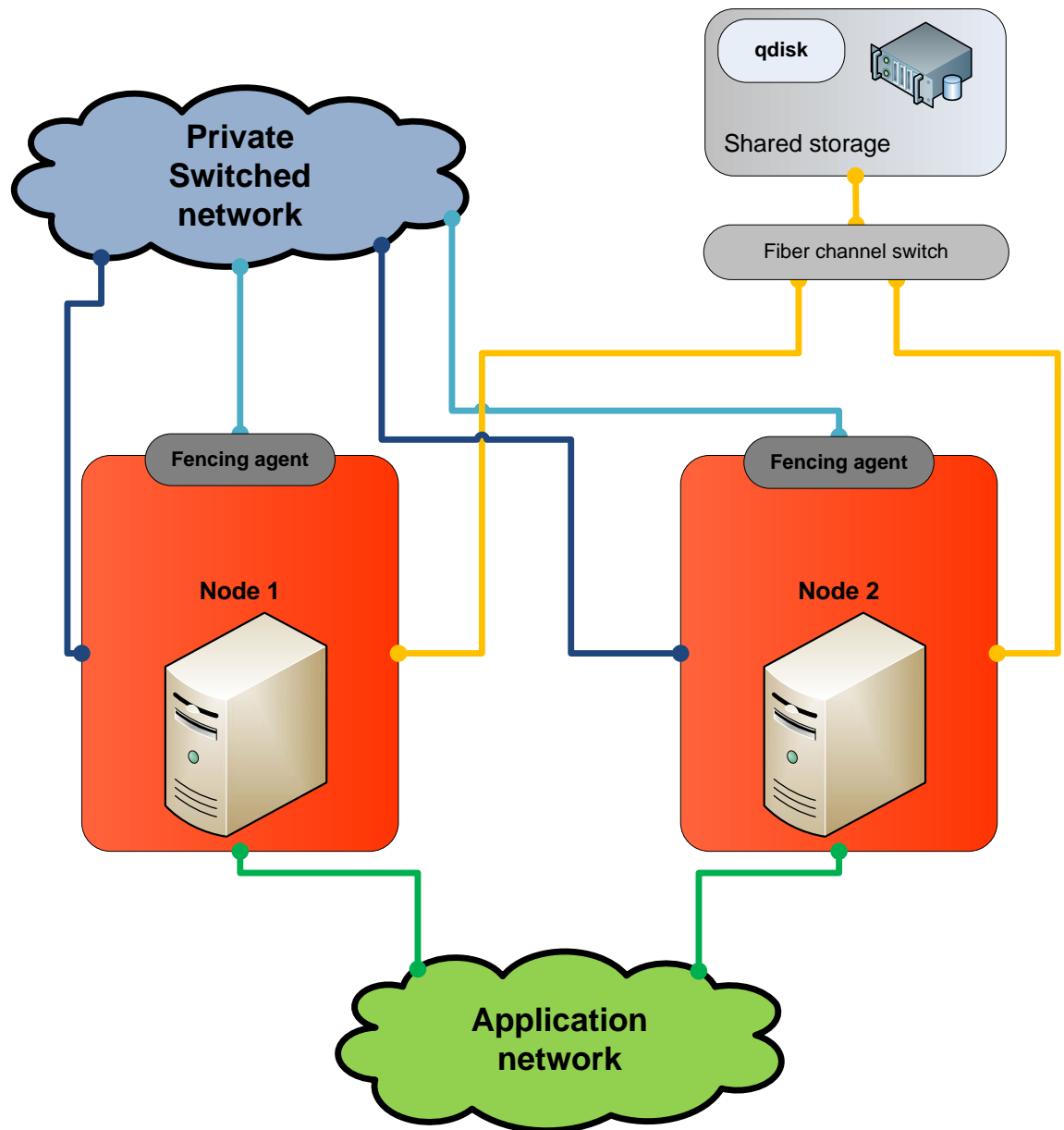
Font	What the Font Represents	Example
	item. When the item is followed by an arrow (->), a cascading menu follows	

1 Introduction

The HPSA and Extension Pack software can be installed over Redhat Server. When we use this OS it's also possible to configure the HPSA and EP into a RH Cluster suite environment.

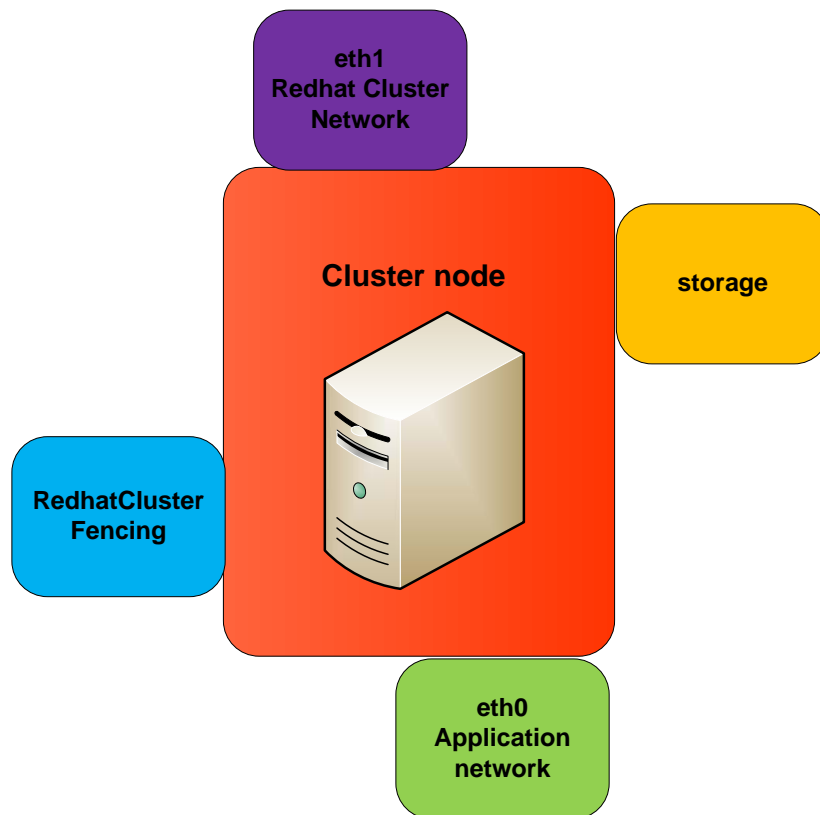
In the next picture is showed the typical Redhat Cluster Suite configuration. The usual elements required to mount the Redhat Cluster Suite are:

1. At least two RH Server (2 nodes)
2. Private switched, application and storage network
3. Shared storage



A single cluster node needs:

1. Cluster network interface
2. Fencing
3. Application network
4. Shared storage



1.1 Definitions

1.1.1 Acronims

WFM: Work Flow Manager

HPSA: HP Service Activator

EP: Extension Pack

SC: Solution Container

RH: Redhat

RHCL: Rehat Cluster Suite

2 Prerequisites

Before installing the Redhat Cluster Suite and HPSA with Extension Pack the user must ensure that the follow requisites are accomplished:

- Software requisites
 - Redhat 5.6 Server (at least two servers)
 - HPSA and/or EP installed in each node with the last hotfix
- Hardware requisites
 - The hardware requisites includes the ones specified for the HPSA
 - SCSI target storage over fiber channel network
 - Private network for Redhat cluster network and Redhat cluster fence

2.1 Installing HPSA and EP for Cluster

The HPSA and EP software need to be installed in all Redhat Cluster nodes. In each cluster the HPSA and EP will share the same database scheme. Then, it's important to create database scheme in the first installation:

1. HPSA: selecting the checkbox "create tables" only in the first node installation
2. EP: answering yes to "create database scheme" only in the first node installation

3 Hardware example

This example shows the hardware required to create Redhat Cluster Suite with shared storage for HPSA and/or EP.

3.1 Linux Server (for each node)

- Xeon X5677 CPU
- 8 GB RAM
- 15,000 RPM hard disk
- 4 Gigabit network adapters
- 2 fiber channel

3.2 SAN Storage

HP Storage Works 4400 EVA (Enterprise Virtual Array)

4 Redhat Cluster Suite installation

4.1 Server update

First, it highly recommended update the system executing next command in all RH servers.

(All commands provided in this guide are required to execute as root.)

```
# yum update
```

Four options will be displayed and answer yes.

4.2 Firewall

It's also highly recommended to disable the firewalls installed into the servers. Execute next commands in all RH servers:

```
# chkconfig iptables off
# chkconfig ip6tables off
# service ip6tables stop
# service iptables stop
```

If the firewalls are not disabled it's mandatory to add the rules required to allow any communications between all RH servers.

4.3 Network

The RH cluster use different networks for different purposes. There're 3 main networks to be configured into the network interfaces:

1. Application network
2. Storage network
3. Cluster network

Depends of the network configuration the ip address and mask can be different. To configure the network interfaces the files `/etc/sysconfig/network-scripts/ifcfg-*` needs to be modified.

Application network file example `/etc/sysconfig/network-scripts/ifcfg-eth1`

```
DEVICE=eth1
BOOTPROTO=static
ONBOOT=yes
HWADDR=00:0c:29:ec:4b:de
IPADDR=10.77.88.154
NETMASK=255.255.255.0
```

Storage network file example `/etc/sysconfig/network-scripts/ifcfg-eth4`

```
DEVICE=eth4
BOOTPROTO=static
```

```
ONBOOT=yes
HWADDR=00:0c:29:ec:4b:fc
IPADDR=10.77.66.154
NETMASK=255.255.255.0
```

Cluster network should be configured in “bond” mode with at least two network interfaces. Follow next steps to configure bond interface:

Modify the file `/etc/modprobe.conf` adding next two lines

```
alias bond0 bonding
options bond0 mode=1 miimon=100
```

Create the file `/etc/sysconfig/network-scripts/ifcfg-bond0` with next content.

```
DEVICE=bond0
IPADDR=10.77.77.14
NETMASK=255.255.255.0
NETWORK=10.77.77.0
BROADCAST=10.77.77.255
BOOTPROTO=none
TYPE=Ethernet
ONBOOT=yes
```

Modify the two interface network files to link to the bond interface. In this case, `/etc/sysconfig/network-scripts/ifcfg-eth2` and `/etc/sysconfig/network-scripts/ifcfg-eth3`

```
DEVICE=eth2
HWADDR=00:0c:29:ec:4b:e8
MASTER=bond0
SLAVE=yes
BOOTPROTO=none
TYPE=Ethernet
ONBOOT=no
```

```
DEVICE=eth3
HWADDR=00:0c:29:ec:4b:f2
MASTER=bond0
SLAVE=yes
BOOTPROTO=none
TYPE=Ethernet
ONBOOT=no
```

It's important to add requires entries to the `/etc/hosts` file. For example;

```
10.77.77.11 rhc11
10.77.77.12 rhc12

10.77.88.151 rhc11-appl
10.77.88.152 rhc12-appl

10.77.66.157 openfiler-storage
```

```
10.77.66.11 rhcl1-storage
10.77.66.12 rhcl2-storage
```

4.4 SELinux

It's mandatory to disable SELINUX. Edit the file `/etc/selinux/config` and change the value to disabled

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - SELinux is fully disabled.
SELINUX=disabled
# SELINUXTYPE= type of policy in use. Possible values are:
#     targeted - Only targeted network daemons are protected.
#     strict - Full SELinux protection.
SELINUXTYPE=targeted

# SETLOCALDEFS= Check local definition changes
SETLOCALDEFS=0
```

4.5 SSH Trust

It's recommended to configure the ssh connection in trust mode between the servers. Then, it's possible to connect, copy files, execute commands,... without password.

Execute in all RH servers:

```
# ssh-keygen -t dsa
Generating public/private dsa key pair.
Enter file in which to save the key (/root/.ssh/id_dsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_dsa.no.
Your public key has been saved in /root/.ssh/id_dsa.no.pub.
The key fingerprint is:
```

Create a file named "authorized_keys2" who includes the content of all id_rsa.pub generated into all servers.

Copy this file to the `/root/.ssh` directory in all RH servers (we're supposing the HOME directory is `/root`). Create next symbolic link in all RH servers:

```
# ln -s /root/.ssh/authorized_keys2 /root/.ssh/authorized_keys
```

Edit the file `/etc/ssh/sshd_config` and check if next parameters are enable in all RH servers. If not, enable and restart sshd process.

```
RSAAuthentication yes
PubkeyAuthentication yes
AuthorizedKeysFile      .ssh/authorized_keys
```

4.6 Cluster package installation

Now, install the RH Cluster suite into all RH servers:

```
# yum groupinstall -y "Cluster Storage" "Clustering"
```

4.7 Quorum disk configuration

It's required to install a quorum disk for each cluster. In this case, we'll suppose we're installing the cluster named clusterhpsa. Before configure the quorum disk it's required to configure the SCSI target. See the "[configure SCSI target](#)" of this guide.

Once, there's configured the SCSI quorum disk execute this command only in one RH server. It's very important to define properly the label of the quorumdisk because it'll be used later into the cluster configuration. For convection, the same name of the cluster is used, in this case "clusterhpsa"

```
# mkqdisk -c /dev/iscsi/qdisk/part -l clusterhpsa
```

4.8 Cluster services

Start the service qdiskd and configure the service to be started with the server. Execute next command in all servers:

```
# service qdiskd start  
# chkconfig qdiskd on
```

Start the service ricci and configure the service to be started with the server. Execute next command in all servers:

```
# service ricci start  
# chkconfig ricci on
```

4.9 Web gui interface

Finally, it's necessary to install the luci web interface to manage the cluster. It's highly recommended to install this service into a RH server different from any RH cluster node. Execute next commands as root only into the server who will contain the luci service.

```
# service luci start  
# chkconfig luci on  
# service luci stop  
# luci_admin init  
  (here a password will be configured to be able to connect luci  
  interface)  
# service luci restart
```

After that, it's possible to start configuring the cluster on luci.

Login into lucci <https://127.0.0.1:8084/>

Account details

Login Name
Login names are case sensitive, make sure the caps lock key is not enabled.

Password
Case sensitive, make sure caps lock is not enabled.

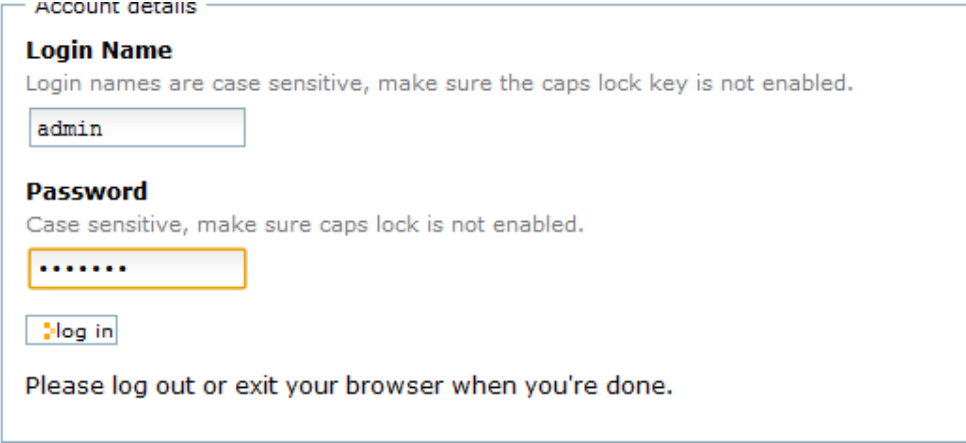
Please log out or exit your browser when you're done.

5 Redhat Cluster Suite configuration

5.1 Login luci

Once the cluster installation is finished is required to configure the cluster.

First, login into luci web interface <https://luciserver:8084/> The user is admin and the password is the same configured into the luci_admin init command.



ACCOUNT details

Login Name
Login names are case sensitive, make sure the caps lock key is not enabled.

Password
Case sensitive, make sure caps lock is not enabled.

Please log out or exit your browser when you're done.

5.2 Cluster creation

Create a cluster adding all nodes. Before click on "submit", click on "View SSL cert fingerprints" to check in the connection with all nodes are ok.

homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure

Create a new cluster

Cluster Name

Node Hostname	Root Password	Key ID
<input type="text" value="rhcl1"/>	<input type="password" value="....."/>	
<input type="text" value="rhcl2"/>	<input type="password" value="....."/>	

Download packages
 Use locally installed packages.

Enable Shared Storage Support
 Reboot nodes before joining cluster
 Check if node passwords are identical.

After create the cluster all nodes will join the cluster.


homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure
- clusterhpsa


clusterhpsa

Please be patient - this cluster's configuration is being modified.



Creating node "rhcl1" for cluster "clusterhpsa"
Node still being created


Installed Reboot Configure Join



[Stop waiting for this job to complete](#)

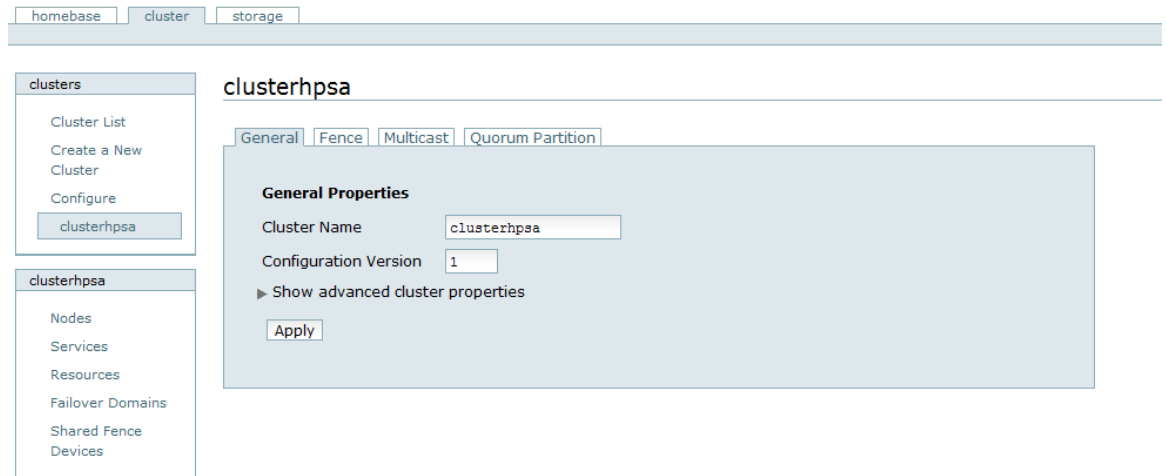
Creating node "rhcl2" for cluster "clusterhpsa"
Node still being created

Install Reboot Configure Join



[Stop waiting for this job to complete](#)

After a while the cluster is created with the nodes selected.



5.3 Quorum

Once the cluster is created, it's required to configure the "Quorum Partition". The values selected will be:

- Interval: 1
- Votes: 1
- TKO: 10
- Minimum score: 1
- Label: the same value defined in mkqdisk. For convection, same as cluster name. In this case, clusterhpsa.
- Heuristic program: ping -c2 t1 ip_storage_server , with interval 1 score 1

homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure
- clusterhpsa

clusterhpsa

- Nodes
- Services
- Resources
- Failover Domains
- Shared Fence
- Devices

clusterhpsa

General Fence Multicast Quorum Partition

Quorum Partition Configuration

Do not use a Quorum Partition
 Use a Quorum Partition

Interval

Votes

TKO

Minimum Score

Label
 Device (deprecated)

Heuristics

Path to Program	Interval	Score
<input type="text" value="ping -c2 -t1 10.77.66.157"/>	<input type="text" value="1"/>	<input type="text" value="1"/>

Once the quorum disk is configured, click on “Cluster list” and check that the value of “Total Cluster Votes” is number of nodes + 1. In this case, we’ve configure 2 nodes and then the value has to be 3. In the value is equal to the number of nodes, restart the cluster and check again. If after restart the cluster the quorum disk keeps equal to the number of nodes, check the steps of this guide and /var/log/messages log file to see the error.

homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure

Choose a cluster to administer

Cluster Name: clusterhpsa Restart this cluster

- **Status:** Quorate
- **Total Cluster Votes:** 3
- **Minimum Required Quorum:** 2

<p>Nodes</p> <ul style="list-style-type: none"> rhc1 rhc2 	<p>Services</p>
--	------------------------

5.4 Fencing device

Finally, it's required to configure the "Fencing device" to be used in this cluster. In this guide, we're going to use "SCSI fencing" but it possible this fencing device needs to be different depending of the hardware environment.

The screenshot shows the Red Hat Cluster Manager web interface. At the top, there are navigation tabs for 'homebase', 'cluster', and 'storage'. Below these, there are two main panels. The left panel, titled 'clusters', contains a 'clusterhpsa2' sub-panel with a list of actions: 'Cluster List', 'Create a New Cluster', 'Configure', 'Nodes', 'Services', 'Resources', 'Failover Domains', 'Shared Fence Devices', 'Add a Fence Device' (highlighted), and 'Configure a Fence Device'. The right panel, titled 'clusterhpsa2', shows the 'Add a Sharable Fence Device' configuration page. It features a 'Fencing Type' dropdown menu set to 'SCSI Fencing', a 'Fence Type' label 'SCSI Reservation Fencing', and a 'Name' input field containing 'fen_scsi_1'. At the bottom of this panel is a button labeled 'Add this shared fence device'.

After add the fending device, each node needs to be assigned to this fencing. For each node, assign the fence device using the node

name.

clusters

- Cluster List
- Create a New Cluster
- Configure

clusterhpsa

- Nodes
- Add a Node
- Configure
- rhcl1
- rhcl2
- rhcl3
- rhcl4
- Services
- Resources
- Failover Domains
- Shared Fence Devices

clusterhpsa

Node Name: rhcl1

Choose a Task...

Status: Cluster member
Show recent log activity for this node

Cluster daemons running on this node

Daemon	Currently running	Enabled at start-up
cman	yes	<input checked="" type="checkbox"/>
rgmanager	yes	<input checked="" type="checkbox"/>

[Update node daemon properties](#)

Services on this Node

- No cluster services are currently running here

Failover Domain Membership

- This node has no failover domain membership

Main Fencing Method

[Add a fence device to this level](#)

[Update main fence properties](#)

Backup Fencing Method

[Add a fence device to this level](#)

[Update backup fence properties](#)

Main Fencing Method

Fence Type SCSI Reservation Fencing

Name

Node name

[Remove this device](#)
[Add an instance](#)

[Add a fence device to this level](#)

[Update main fence properties](#)

Backup Fencing Method

[Add a fence device to this level](#)

[Update backup fence properties](#)

6 Configuring HPSA cluster services

6.1 HPSA prerequisites

First change the parameter "start_watch_dog_process" of keep_alive module to false into the file /etc/opt/OV/ServiceActivator/config/mwfm.xml and stop HPSA:

```
# /etc/init.d/activator stop
```

The persistence needs to be based in data base, this means the transaction module into the file /etc/opt/OV/ServiceActivator/config/mwfm.xml needs to be configured as DBTransactionModule. In case the transaction module is based on file, if a HPSA instance goes down the workflows inside of this server cannot be recovered until the same instance is started again.

```
<Module>
  <Name>transaction_manager</Name>
  <Class-
Name>com.hp.ov.activator.mwfm.engine.module.DBTransactionModule</Class-
Name>
</Module>
```

6.2 HPSA script resource

Create the file /opt/OV/ServiceActivator/bin/hpsa_cluster in all nodes with the next content:

```
#!/bin/bash
#Service script for HPSA under RH Cluster
# Global variables
HPSA_USER=root
OVERRIDE_FILE=/var/tmp/hpsa_override
REC_LIST=""

function override () {
    if [ -f $OVERRIDE_FILE ]
    then
        exit 0
    fi
}

function start () {
    status && return 0 || echo ""
    su - $HPSA_USER -c "sh /etc/init.d/activator start"
    status
}

function stop () {
    su - $HPSA_USER -c "sh /etc/init.d/activator stop"
    c=1
    while [ $c -le 60 ]
    do
        status && (( c++ )) || return 0
        echo "sleeping 1"
        sleep 1
    done
}
```

```
done
if [ -f /opt/HP/jboss/jboss.pid ]
then
    JBOSS_PPID=`cat /opt/HP/jboss/jboss.pid`
    jboss_pid=`ps -ef | grep $JBOSS_PPID | grep java | grep run.sh |
/bin/awk '{print $2}'`
    kill -9 $jboss_pid
fi
status && return 1 || return 0
}

function status () {
    if [ -f /opt/HP/jboss/jboss.pid ]
    then
        JBOSS_PPID=`cat /opt/HP/jboss/jboss.pid`
        jboss_pid=`ps -ef | grep $JBOSS_PPID | grep java | grep run.sh |
/bin/awk '{print $2}'`
        if [ "" = "$jboss_pid" ]
        then
            return 1
        fi
        ps -afu $HPSA_USER | grep -v grep | grep $jboss_pid
    else
        return 1
    fi
    return $?
}

function notify () {
    if [ "" = "$REC_LIST" ]
    then
        return 0
    fi
    mail -s "$1 hpsa on `hostname`" $REC_LIST < /dev/null
}

override
case "$1" in
start) start
    notify $1
    ;;
stop) stop
    notify $1
    ;;
status) status
    ;;
*) echo "Usage: $0 start|stop|status"
    ;;
esac
```

Add resource the hpsa script resource. The path to be used is
/opt/OV/ServiceActivator/bin/hpsa_cluster

The screenshot shows the 'clusterhpsa' configuration page with the 'Add a Resource' section. The 'Name' field is set to 'hpsa' and the 'Full path to script file' is set to '/bin/hpsa_cluster.sh'. A 'Submit' button is visible at the bottom of the configuration area.

clusters
Cluster List
Create a New Cluster
Configure

clusterhpsa
Nodes
Services
Resources
Add a Resource
Configure a Resource
Failover Domains
Shared Fence Devices

clusterhpsa
Add a Resource

Script Resource Configuration

Name:

Full path to script file:

6.3 HPSA Failover domain

Add a failover domain for each node. In each failover only one node needs to be configured. Check the "prioritized" and "Restrict failover to this domain's members" check box.

The screenshot shows the 'clusterhpsa' configuration page with the 'Add a Failover Domain' section. The 'Failover Domain Name' is 'failoverrh11'. The 'Prioritized' and 'Restrict failover to this domain's members' checkboxes are checked. The 'Do not fail back services in this domain' checkbox is unchecked. The 'Failover domain membership' table lists nodes rhc1 through rhc4, with rhc1 selected as a member with a priority of 1.

clusters
Cluster List
Create a New Cluster
Configure

clusterhpsa
Nodes
Services
Resources
Failover Domains
Add a Failover Domain
Configure a Failover Domain
Shared Fence Devices

clusterhpsa
Add a Failover Domain

Failover Domain Name:

Prioritized:

Restrict failover to this domain's members:

Do not fail back services in this domain:

Failover domain membership

Node	Member	Priority
rhc1	<input checked="" type="checkbox"/>	<input type="text" value="1"/>
rhc2	<input type="checkbox"/>	<input type="text" value="1"/>
rhc3	<input type="checkbox"/>	<input type="text" value="1"/>
rhc4	<input type="checkbox"/>	<input type="text" value="1"/>

6.4 HPSA service

For each hpsa instance to be used create a service, using the failover of the node and setting the recovery policy to "restart". Also, enable the "Automatically start this services" checkbox. Finally, assign hpsa resource.

homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure

clusterhpsa

- Nodes
- Services
- Add a Service
- Configure a Service
- Resources
- Failover Domains
- Shared Fence Devices

clusterhpsa

Add a Service

Service name

Automatically start this service

Enable NFS lock workarounds

Run exclusive

Failover Domain

Recovery policy

Maximum number of restart failures before relocating

Length of time in seconds after which to forget a restart

Script Resource Configuration

Name

Full path to script file

This resource is an independent subtree

This resource is non-critical

Maximum number of restart failures before giving up (applies only for non-critical resources)

Restart expire time (applies only for non-critical resources)

6.5 IP Virtual

Sometimes is required to create a IP Virtual to define a single entry point. For these cases, a IP resource needs to be created with the IP value.

homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure

clusterhpsa

- Nodes
- Services
- Resources
- Add a Resource
- Configure a Resource
- Failover Domains
- Shared Fence Devices

clusterhpsa

Add a Resource

IP Address Resource Configuration

IP address

Monitor link

Create a failover domain that includes all hpsa cluster nodes. It's possible to define where we want to locate the IP Virtual by default. Setting the parameter "Prioritized" to true and defining the priority for each node.

The screenshot shows the 'Add a Failover Domain' configuration page. The 'Failover Domain Name' is 'failoverhpsa'. The 'Prioritized' checkbox is checked. The 'Restrict failover to this domain's members' checkbox is checked. The 'Do not fail back services in this domain' checkbox is unchecked. The 'Failover domain membership' table lists nodes rhcd1, rhcd2, rhcd3, and rhcd4, all of which are members with priorities 1, 2, 3, and 4 respectively. A 'Submit' button is at the bottom.

Node	Member	Priority
rhcd1	<input checked="" type="checkbox"/>	1
rhcd2	<input checked="" type="checkbox"/>	2
rhcd3	<input checked="" type="checkbox"/>	3
rhcd4	<input checked="" type="checkbox"/>	4

Create the service hpsaipvirtual adding the IP resource and hpsa script as as child of IP resource.

The screenshot shows the 'Add a Service' configuration page. The service name is 'hpsaipvirtual'. The 'Automatically start this service' checkbox is checked. The 'Enable NFS lock workarounds' checkbox is unchecked. The 'Run exclusive' checkbox is unchecked. The 'Failover Domain' is 'failoverhpsa'. The 'Recovery policy' is 'Relocate'. The 'Maximum number of restart failures before relocating' is 0. The 'Length of time in seconds after which to forget a restart' is 0. The 'IP Address Resource Configuration' section shows the IP address '10.77.88.161', 'Monitor link' checked, and checkboxes for 'This resource is an independent subtree' and 'This resource is non-critical'. The 'Script Resource Configuration' section shows the name 'hpsa' and the full path to script file '/opt/OV/ServiceActivi...'. There are buttons for 'Add a child' and 'Delete this resource'.

7 Configuring EP over Redhat Cluster Suite

7.1 Solution Container

Edit the file `/opt/OV/ServiceActivator/bin/run_sc.sh` and write next content:

```
#!/usr/bin/ksh

export JBOSS_CLASSPATH=/etc/opt/OV/ServiceActivator/config/sosa
HPPIA=/opt/OV/ServiceActivator
JBOSS_HOME=/opt/HP/jboss

# get the activator environment variables that we need
. $HPPIA/bin/setenv

PATH=/sbin:/usr/sbin:/usr/bin:/bin:/etc:$JAVA_HOME/bin
export PATH

cd $JBOSS_HOME/bin
#nohup ./run.sh -b 0.0.0.0 > /dev/null 2>&1 &

./run.sh -c diagnostic -b 0.0.0.0 > /dev/null 2>&1 &
```

Create the file `/opt/OV/ServiceActivator/bin/sc_cluster` in all nodes with the next content:

```
#!/bin/bash
#Service script for SC under RH Cluster
# Global variables
SC_USER=root
HOMEDIR=/root
OVERRIDE_FILE=/var/tmp/sc_override
REC_LIST=""

function override () {
    if [ -f $OVERRIDE_FILE ]
    then
        exit 0
    fi
}

function start () {
    status && return 0 || echo ""
    su - $SC_USER -c "sh /opt/HP/jboss/bin/run_sc.sh"
    sleep 2
    status
}

function stop () {
    sc_pid=`ps -afu $SC_USER | grep -v grep | grep java | grep run.sh |
grep diagnostic | /bin/awk '{print $2}'`
    status && kill -9 $sc_pid || return 0
    status && return 1 || return 0
}
```

```
function status () {
    ps -afu $SC_USER | grep -v grep | grep java | grep run.sh | grep
diagnostic
    return $?
}

function notify () {
    if [ "" = "$REC_LIST" ]
    then
        return 0
    fi
    mail -s "$1 SC on `hostname`" $REC_LIST < /dev/null
}

override
case "$1" in
start)  start
        notify $1
        ;;
stop)   stop
        notify $1
        ;;
status) status
        ;;
*) echo "Usage: $0 start|stop|status"
    ;;
esac
```

Add the solution container script resource. The path to be used is
/opt/OV/ServiceActivator/bin/sc_cluster

The screenshot shows the Red Hat Cluster Manager web interface. At the top, there are tabs for 'homebase', 'cluster', and 'storage'. Below these, there are two main panels. The left panel, titled 'clusters', contains a menu with options: 'Cluster List', 'Create a New Cluster', and 'Configure'. The right panel, titled 'clusterhpsa', contains a sub-menu with options: 'Nodes', 'Services', 'Resources', 'Add a Resource' (highlighted), 'Configure a Resource', 'Failover Domains', 'Shared Fence', and 'Devices'. The 'Add a Resource' page is displayed, showing the 'Script Resource Configuration' section. It has two input fields: 'Name' with the value 'sc' and 'Full path to script file' with the value '/opt/HP/jboss/bin/sc_'. A 'Submit' button is located below the input fields.

Create the IP virtual resource for Solution Container

homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure

clusterhpsa

- Nodes
- Services
- Resources
- Add a Resource
- Configure a Resource
- Failover Domains
- Shared Fence Devices

clusterhpsa

Add a Resource

IP Address Resource Configuration

IP address

Monitor link

Create the failover domain for solution container. Depends where to locate the SC by default the priority values needs to be configured properly.

homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure

clusterhpsa

- Nodes
- Services
- Resources
- Failover Domains
- Add a Failover Domain
- Configure a Failover Domain

clusterhpsa

Failover Domain Form

Failover Domain Name

Prioritized

Restrict failover to this domain's members

Do not fail back services in this domain

Failover domain membership

Node	Member	Priority
rhcl1	<input checked="" type="checkbox"/>	<input type="text" value="2"/>
rhcl2	<input checked="" type="checkbox"/>	<input type="text" value="1"/>

Create the service Solution Container setting the failover Solution Container, recovery policy to relocate, automatic restart and add the IP virtual and Solution Container resource as tree.

Add a Service

Service name	<input type="text" value="sc"/>
Automatically start this service	<input checked="" type="checkbox"/>
Enable NFS lock workarounds	<input type="checkbox"/>
Run exclusive	<input type="checkbox"/>
Failover Domain	<input type="text" value="failoversc"/>
Recovery policy	<input type="text" value="Relocate"/>
Maximum number of restart failures before relocating	<input type="text" value="0"/>
Length of time in seconds after which to forget a restart	<input type="text" value="0"/>

IP Address Resource Configuration	
IP address	<input type="text" value="10.77.88.163"/>
Monitor link	<input checked="" type="checkbox"/>
This resource is an independent subtree	<input type="checkbox"/>
This resource is non-critical	<input type="checkbox"/>
Maximum number of restart failures before giving up (applies only for non-critical resources)	<input type="text"/>
Restart expire time (applies only for non-critical resources)	<input type="text"/>
<input type="button" value="Add a child"/>	<input type="button" value="Delete this resource"/>
▼ Hide Children	
Script Resource Configuration	
Name	<input type="text" value="sc"/>
Full path to script file	<input type="text" value="/opt/HP/jboss/bin/sc_"/>
This resource is an independent subtree	<input type="checkbox"/>
This resource is non-critical	<input type="checkbox"/>

7.2 SOSA

Edit the file

`/opt/HP/jboss/server/agnostic/deploy/hpovact.sar/activator.war/properties/sosa3.properties` in all nodes and change the parameter to the IP virtual or `/etc/hosts` name.

```
sosamanager.service.host = sosa
```

It's mandatory create at least next SCSI target (see the "[configure SCSI target](#)" of this guide), one for SOSA configuration directory, one for SOSA persistence directory and one for history. It's possible to configure more than one persistence directory to improve the performance in case each SCSI target goes to different physical hard disk.

In each server execute next commands:

```
# mv /opt/OV/ServiceActivator/EP/SOSA/conf
/opt/OV/ServiceActivator/EP/SOSA/conf.install
# mkdir /opt/OV/ServiceActivator/EP/SOSA/conf
```

In one server mount the SCSI target and copy the conf content.

```
# mount /dev/iscsi/sosalconf/par /opt/OV/ServiceActivator/EP/SOSA/conf
# cp -r /opt/OV/ServiceActivator/EP/SOSA/conf.install/*
/opt/OV/ServiceActivator/EP/SOSA/conf
# umount /dev/iscsi/sosalconf/par
```

If the paths for file persistence is used /opt/OV/ServiceActivator/EP/SOSA/filepersistence and for history /opt/OV/ServiceActivator/EP/SOSA/filehistory it's not required to modify the sosa_conf.xml.

Add configuration, persistence and history GFS resource.

1. /opt/OV/ServiceActivator/EP/SOSA/conf
2. /opt/OV/ServiceActivator/EP/SOSA/filepersistence
3. /opt/OV/ServiceActivator/EP/SOSA/filepersistence

The screenshot shows a web interface for managing cluster resources. At the top, there are tabs for 'homebase', 'cluster', and 'storage'. The main content area is titled 'clusterhpsa' and 'Add a Resource'. Below this, there is a section for 'File System Resource Configuration' with the following fields:

- Name: sosalconf
- File system type: ext3
- Mount point: /opt/OV/ServiceActivator/EP/SOSA/conf
- Device: /dev/iscsi/sosalconf,par
- Options: (empty)
- File system ID (optional): (empty)
- Force unmount:
- Reboot host node if unmount fails:
- Check file system before mounting:

At the bottom of the form is a 'Submit' button. On the left side, there is a navigation menu with the following items:

- clusters
 - Cluster List
 - Create a New Cluster
 - Configure
- clusterhpsa
 - Nodes
 - Services
 - Resources
 - Add a Resource
 - Configure a Resource
 - Failover Domains
 - Shared Fence Devices

Add a Resource

File System Resource Configuration

Name	<input type="text" value="sosalpers1"/>
File system type	<input type="text" value="ext3"/>
Mount point	<input type="text" value="/opt/OV/ServiceActivat"/>
Device	<input type="text" value="/dev/iscsi/sosalpers1"/>
Options	<input type="text"/>
File system ID (optional)	<input type="text"/>
Force unmount	<input checked="" type="checkbox"/>
Reboot host node if unmount fails	<input type="checkbox"/>
Check file system before mounting	<input type="checkbox"/>

Add a Resource

File System Resource Configuration

Name	<input type="text" value="sosalhist"/>
File system type	<input type="text" value="ext3"/>
Mount point	<input type="text" value="/opt/OV/ServiceActivat"/>
Device	<input type="text" value="/dev/iscsi/sosalhist1"/>
Options	<input type="text"/>
File system ID (optional)	<input type="text"/>
Force unmount	<input checked="" type="checkbox"/>
Reboot host node if unmount fails	<input type="checkbox"/>
Check file system before mounting	<input type="checkbox"/>

Create the file `/opt/OV/ServiceActivator/EP/SOSA/bin/sosa_cluster.sh` in all nodes with the next content

```
#!/bin/bash
#Service script for SOSA under RH Cluster
# Global variables
SOSA_USER=root
```



```

OVERRIDE_FILE=/var/tmp/sosa_override
REC_LIST=""

function override () {
    if [ -f $OVERRIDE_FILE ]
    then
        exit 0
    fi
}

function start () {
    status && return 0 || echo ""
    su - $SOSA_USER -c "sh /opt/OV/ServiceActivator/EP/SOSA/bin/sosa.sh
start"
    status
}

function stop () {
    su - $SOSA_USER -c "sh /opt/OV/ServiceActivator/EP/SOSA/bin/sosa.sh
stop"
    c=1
    while [ $c -le 60 ]
    do
        status && (( c++ )) || return 0
        sleep 1
    done
    if [ -f /opt/OV/ServiceActivator/EP/SOSA/tmp/sosa.pid ]
    then
        ps -afu $ECP_USER | grep -v grep | grep `cat
/opt/OV/ServiceActivator/EP/SOSA/tmp/sosa.pid`
        kill -9 `cat /opt/OV/ServiceActivator/EP/SOSA/tmp/sosa.pid`
    fi
    status && return 1 || return 0
}

function status () {
    if [ -f /opt/OV/ServiceActivator/EP/SOSA/tmp/sosa.pid ]
    then
        ps -afu $SOSA_USER | grep -v grep | grep `cat
/opt/OV/ServiceActivator/EP/SOSA/tmp/sosa.pid`
    else
        return 1
    fi
    return $?
}

function notify () {
    if [ "" = "$REC_LIST" ]
    then
        return 0
    fi
    mail -s "$1 sosa on `hostname`" $REC_LIST < /dev/null
}

override
case "$1" in
start) start
notify $1
;;

```

```
stop)    stop
        notify $1
        ;;
status)  status
        ;;
*)      echo "Usage: $0 start|stop|status"
        ;;
esac
```

Add resource the solution container script resource. The path to be used is
/opt/OV/ServiceActivator/EP/SOSA/bin/sosa_cluster.sh

homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure

clusterhpsa

- Nodes
- Services
- Resources
- Add a Resource**
- Configure a Resource
- Failover Domains
- Shared Fence
- Devices

clusterhpsa

Add a Resource

Script Resource Configuration

Name

Full path to script file

Create the IP virtual resource for SOSA

homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure

clusterhpsa

- Nodes
- Services
- Resources
- Add a Resource**
- Configure a Resource
- Failover Domains
- Shared Fence
- Devices

clusterhpsa

Add a Resource

IP Address Resource Configuration

IP address

Monitor link

Create the failover domain for SOSA. Depends where to locate SOSA by default the priority values needs to be configured properly.

The screenshot shows a web interface for configuring a failover domain. The breadcrumb trail is 'homebase > cluster > storage'. The main heading is 'clusterhpsa' and the sub-heading is 'Failover Domain Form'. The form contains the following fields and options:

- Failover Domain Name:** A text input field containing 'failoversosa'.
- Prioritized:** A checked checkbox.
- Restrict failover to this domain's members:** A checked checkbox.
- Do not fail back services in this domain:** An unchecked checkbox.
- Failover domain membership:** A table with columns for Node, Member, and Priority.

Node	Member	Priority
rhcd1	<input checked="" type="checkbox"/>	<input type="text" value="1"/>
rhcd2	<input checked="" type="checkbox"/>	<input type="text" value="2"/>
- Submit:** A button at the bottom left of the form.

On the left side, there is a navigation menu with 'clusters' and 'clusterhpsa' sections. The 'clusters' section includes 'Cluster List', 'Create a New Cluster', and 'Configure'. The 'clusterhpsa' section includes 'Nodes', 'Services', 'Resources', 'Failover Domains', 'Add a Failover Domain', and 'Configure a...'. The 'homebase', 'cluster', and 'storage' tabs are visible at the top.

Create the service SOSA setting the failover sosa, recovery policy to relocate, automatic restart and add all resources; IP virtual, directories and sosa script resource as tree.

Add a Service

Service name	<input type="text" value="sosa"/>
Automatically start this service	<input checked="" type="checkbox"/>
Enable NFS lock workarounds	<input type="checkbox"/>
Run exclusive	<input type="checkbox"/>
Failover Domain	<input type="text" value="failoversosa"/>
Recovery policy	<input type="text" value="Relocate"/>
Maximum number of restart failures before relocating	<input type="text" value="0"/>
Length of time in seconds after which to forget a restart	<input type="text" value="0"/>

IP Address Resource Configuration	
IP address	<input type="text" value="10.77.88.165"/>
Monitor link	<input checked="" type="checkbox"/>
This resource is an independent subtree	<input type="checkbox"/>
This resource is non-critical	<input type="checkbox"/>
Maximum number of restart failures before giving up (applies only for non-critical resources)	<input type="text"/>
Restart expire time (applies only for non-critical resources)	<input type="text"/>
<input type="button" value="Add a child"/>	<input type="button" value="Delete this resource"/>
▼ Hide Children	
File System Resource Configuration	
Name	<input type="text" value="sosalconf"/>
File system type	<input type="text" value="ext3"/>
Mount point	<input type="text" value="/opt/OV/ServiceActivator"/>
Device	<input type="text" value="/dev/iscsi/sosalconf"/>

7.3 ECP

Edit the file

`/opt/HP/jboss/server/agnostic/deploy/hpovact.sar/activator.war/properties/ecp.properties` in all nodes and change the parameter to the IP virtual or etc/hosts name.

```
ecpmanager.service.host = ecp
```

Create the file `/opt/OV/ServiceActivator/EP/ECP/bin/ecp_cluster.sh` in all nodes and write next content:

```
#!/bin/bash
```

```
#Service script for ECP under RH Cluster
# Global variables
ECP_USER=root
OVERRIDE_FILE=/var/tmp/ecp_override
REC_LIST=""

function override () {
    if [ -f $OVERRIDE_FILE ]
    then
        exit 0
    fi
}

function start () {
    status && return 0 || echo ""
    su - $ECP_USER -c "sh
/opt/OV/ServiceActivator/EP/ECP/bin/StartServer.sh"
    status
}

function stop () {
    su - $ECP_USER -c "sh
/opt/OV/ServiceActivator/EP/ECP/bin/StopServer.sh"
    c=1
    while [ $c -le 60 ]
    do
        status && (( c++ )) || return 0
        sleep 1
    done
    if [ -f /opt/OV/ServiceActivator/EP/ECP/log/ecp.pid ]
    then
        ps -afu $ECP_USER | grep -v grep | grep `cat
/opt/OV/ServiceActivator/EP/ECP/log/ecp.pid`
        kill -9 `cat /opt/OV/ServiceActivator/EP/ECP/log/ecp.pid`
    fi
    status && return 1 || return 0
}

function status () {
    if [ -f /opt/OV/ServiceActivator/EP/ECP/log/ecp.pid ]
    then
        ps -afu $ECP_USER | grep -v grep | grep `cat
/opt/OV/ServiceActivator/EP/ECP/log/ecp.pid`
    else
        return 1
    fi
    return $?
}

function notify () {
    if [ "" = "$REC_LIST" ]
    then
        return 0
    fi
    mail -s "$1 ECP on `hostname`" $REC_LIST < /dev/null
}

override
case "$1" in
```

```
start)  start
        notify $1
        ;;
stop)   stop
        notify $1
        ;;
status) status
        ;;
*)     echo "Usage: $0 start|stop|status"
        ;;
esac
```

Add the ECP script resource. The path to be used is
/opt/OV/ServiceActivator/EP/ECP/bin/ecp_cluster.sh

The screenshot shows the Red Hat Cluster Manager web interface. At the top, there are tabs for 'homebase', 'cluster', and 'storage'. Below these, there are two main sections: 'clusters' and 'clusterhpsa'. The 'clusters' section has links for 'Cluster List', 'Create a New Cluster', and 'Configure'. The 'clusterhpsa' section has links for 'Nodes', 'Services', 'Resources', 'Add a Resource', 'Configure a Resource', 'Failover Domains', and 'Shared Fence Devices'. The 'Add a Resource' link is highlighted. To the right, the 'clusterhpsa' page is open, showing the 'Add a Resource' configuration. The 'Script Resource Configuration' section has two input fields: 'Name' with the value 'ecp' and 'Full path to script file' with the value '/bin/ecp_cluster.sh'. A 'Submit' button is located below the input fields.

Create the IP virtual resource for ECP

homebase | cluster | storage

clusters

- Cluster List
- Create a New Cluster
- Configure

clusterhpsa

- Nodes
- Services
- Resources
- Add a Resource
- Configure a Resource
- Failover Domains
- Shared Fence Devices

clusterhpsa

Add a Resource

IP Address Resource Configuration

IP address

Monitor link

Create the failover domain for ECP. Depends where to locate the ECP by default the priority values needs to be configured properly.

homebase | cluster | storage

clusters

- Cluster List
- Create a New Cluster
- Configure

clusterhpsa

- Nodes
- Services
- Resources
- Failover Domains
- Add a Failover Domain
- Configure a

clusterhpsa

Failover Domain Form

Failover Domain Name

Prioritized

Restrict failover to this domain's members

Do not fail back services in this domain

Failover domain membership

Node	Member	Priority
rhcl1	<input checked="" type="checkbox"/>	<input type="text" value="1"/>
rhcl2	<input checked="" type="checkbox"/>	<input type="text" value="2"/>

Create the service ECP setting the failover ECP, recovery policy to relocate, automatic restart and add the IP virtual and ECP resource as tree.

Add a Service

Service name	<input type="text" value="ecp"/>
Automatically start this service	<input checked="" type="checkbox"/>
Enable NFS lock workarounds	<input type="checkbox"/>
Run exclusive	<input type="checkbox"/>
Failover Domain	<input type="text" value="failoverecp"/>
Recovery policy	<input type="text" value="Relocate"/>
Maximum number of restart failures before relocating	<input type="text" value="0"/>
Length of time in seconds after which to forget a restart	<input type="text" value="0"/>

IP Address Resource Configuration

IP address	<input type="text" value="10.77.88.162"/>
Monitor link	<input checked="" type="checkbox"/>
This resource is an independent subtree	<input type="checkbox"/>
This resource is non-critical	<input type="checkbox"/>
Maximum number of restart failures before giving up (applies only for non-critical resources)	<input type="text"/>
Restart expire time (applies only for non-critical resources)	<input type="text"/>

▼ Hide Children

Script Resource Configuration

Name	<input type="text" value="ecp"/>
Full path to script file	<input type="text" value="/opt/OV/ServiceActivator/EP/lockmanager_cluster.sh"/>
This resource is an independent subtree	<input type="checkbox"/>
This resource is non-critical	<input type="checkbox"/>
Maximum number of restart failures before giving up	<input type="text"/>

7.4 LockManager

Edit the file `/opt/HP/jboss/server/diagnostic/deploy/hpovact.sar/activator.war/properties/LockManager-Web.properties` in all nodes and change the parameter to the IP virtual or etc/hosts name.

```
lockmanager.service.host = lockmanager
```

Create the file `/opt/OV/ServiceActivator/EP/LockManager/bin/lockmanager_cluster.sh` in all nodes and write next content

```
#!/bin/bash  
#Service script for LOCKMANAGER under RH Cluster
```



```
# Global variables
LOCKMANAGER_USER=root
OVERRIDE_FILE=/var/tmp/lockmanager_override
REC_LIST=""

function override () {
    if [ -f $OVERRIDE_FILE ]
    then
        exit 0
    fi
}

function start () {
    status && return 0 || echo ""
    su - $LOCKMANAGER_USER -c "sh
/opt/OV/ServiceActivator/EP/LockManager/bin/StartServer.sh"
    status
}

function stop () {
    su - $LOCKMANAGER_USER -c "sh
/opt/OV/ServiceActivator/EP/LockManager/bin/StopServer.sh"
    c=1
    while [ $c -le 60 ]
    do
        status && (( c++ )) || return 0
        sleep 1
    done
    if [ -f /opt/OV/ServiceActivator/EP/LockManager/tmp/lckmgr.pid ]
    then
        ps -afu $LOCKMANAGER_USER | grep -v grep | grep `cat
/opt/OV/ServiceActivator/EP/LockManager/tmp/lckmgr.pid`
        kill -9 `cat
/opt/OV/ServiceActivator/EP/LockManager/tmp/lckmgr.pid`
    fi
    status && return 1 || return 0
}

function status () {
    if [ -f /opt/OV/ServiceActivator/EP/LockManager/tmp/lckmgr.pid ]
    then
        ps -afu $LOCKMANAGER_USER | grep -v grep | grep `cat
/opt/OV/ServiceActivator/EP/LockManager/tmp/lckmgr.pid`
    else
        return 1
    fi
    return $?
}

function notify () {
    if [ "" = "$REC_LIST" ]
    then
        return 0
    fi
    mail -s "$1 LOCKMANAGER on `hostname`" $REC_LIST < /dev/null
}

override
case "$1" in
```

```
start) start
    notify $1
    ;;
stop) stop
    notify $1
    ;;
status) status
    ;;
*) echo "Usage: $0 start|stop|status"
    ;;
esac
```

Add the LockManager script resource. The path to be used is
/opt/OV/ServiceActivator/EP/LockManager/bin/lockmanager_cluster.sh

The screenshot shows the Red Hat Cluster Manager web interface. At the top, there are tabs for 'homebase', 'cluster', and 'storage'. Below these, there are two main panels. The left panel, titled 'clusters', contains a 'Cluster List' and options to 'Create a New Cluster' and 'Configure'. The right panel, titled 'clusterhpsa', contains an 'Add a Resource' section. Under 'Script Resource Configuration', there are two input fields: 'Name' with the value 'lockmanager' and 'Full path to script file' with the value '/opt/ockmanager_cluster.sh'. A 'Submit' button is located below the input fields.

Create the IP virtual resource for LockManager

homebase | cluster | storage

clusters

- Cluster List
- Create a New Cluster
- Configure

clusterhpsa

- Nodes
- Services
- Resources
- Add a Resource
- Configure a Resource
- Failover Domains
- Shared Fence Devices

clusterhpsa

Add a Resource

IP Address Resource Configuration

IP address

Monitor link

Create the failover domain for Lockmanager. Depends where to locate the LockManager by default the priority values needs to be configured properly.

homebase | cluster | storage

clusters

- Cluster List
- Create a New Cluster
- Configure

clusterhpsa

- Nodes
- Services
- Resources
- Failover Domains
- Add a Failover Domain
- Configure a Resource

clusterhpsa

Failover Domain Form

Failover Domain Name

Prioritized

Restrict failover to this domain's members

Do not fail back services in this domain

Failover domain membership

Node	Member	Priority
rhcd1	<input checked="" type="checkbox"/>	<input type="text" value="2"/>
rhcd2	<input checked="" type="checkbox"/>	<input type="text" value="1"/>

Create the service LockManager setting the failover LockManager, recovery policy to relocate, automatic restart and add the IP virtual and LockManager resource as tree.

Add a Service

Service name	<input type="text" value="lockmanager"/>
Automatically start this service	<input checked="" type="checkbox"/>
Enable NFS lock workarounds	<input type="checkbox"/>
Run exclusive	<input type="checkbox"/>
Failover Domain	<input type="text" value="failoverlock"/>
Recovery policy	<input type="text" value="Relocate"/>
Maximum number of restart failures before relocating	<input type="text" value="0"/>
Length of time in seconds after which to forget a restart	<input type="text" value="0"/>

IP Address Resource Configuration

IP address	<input type="text" value="10.77.88.164"/>
Monitor link	<input checked="" type="checkbox"/>
This resource is an independent subtree	<input type="checkbox"/>
This resource is non-critical	<input type="checkbox"/>
Maximum number of restart failures before giving up (applies only for non-critical resources)	<input type="text"/>
Restart expire time (applies only for non-critical resources)	<input type="text"/>

▼ Hide Children

Script Resource Configuration

Name	<input type="text" value="lockmanager"/>
Full path to script file	<input type="text" value="/opt/OV/ServiceActivi"/>

This resource is an independent subtree

8 Miscellaneous

8.1 Configure a SCSI target

Check if the `iscsi-initiator-utils` and `isns-utils` are installed. If not, execute next command:

```
# yum install -y iscsi-initiator-utils isns-utils
```

Start the services and enable to be started when the server starts

```
# service iscsid start
# chkconfig iscsid on
# chkconfig iscsi on
```

Find the iSCSI targets using the ip address of the storage server:

```
# iscsiadm -m discovery -t sendtargets -p 10.77.66.157
```

Check and login to the iSCSI target that we're going to configure:

```
# iscsiadm -m node -T iqn.2006-01.com.openfiler:scsi.qdisk -p
10.77.66.157
# iscsiadm -m node -T iqn.2006-01.com.openfiler:scsi.qdisk -p
10.77.66.157 --login
```

Enable automatic login when the `iscsid` services starts:

```
# iscsiadm -m node -T iqn.2006-01.com.openfiler:scsi.qdisk -p
10.77.66.157 --op update -n node.startup -v automatic
```

When the `iscsid` services starts create a device into the directory `/dev`. The `iscsid` doesn't guarantee to create this device with the same name in every start. Then, we're going to configure a script to guarantee the device name will be always the same. This step has to be made only with the first SCSI target.

Edit the file `/etc/udev/rules.d/55-openiscsi.rules` and write next content:

```
KERNEL=="sd*", BUS=="scsi", PROGRAM="/etc/udev/scripts/iscsidev.sh
%b",SYMLINK+="iscsi/%c/part%n"
```

Create the directory `/etc/udev/scripts`

```
# mkdir -p /etc/udev/scripts
```

Create the file `/etc/udev/scripts/iscsidev.sh` in all nodes and write next content:

```
#!/bin/sh
```

```
# FILE: /etc/udev/scripts/iscsidev.sh

BUS=${1}
HOST=${BUS%%:*}

[ -e /sys/class/iscsi_host ] || exit 1

file="/sys/class/iscsi_host/host${HOST}/device/session*/iscsi_session*/targetname"

target_name=$(cat ${file})

# This is not an open-scsi drive
if [ -z "${target_name}" ]; then
    exit 1
fi

# Check if QNAP drive
check_qnap_target_name=${target_name%%:*}
if [ $check_qnap_target_name = "iqn.2004-04.com.qnap" ]; then
    target_name=`echo "${target_name%.*}"`
fi

echo "${target_name###*}."
```

Changes file permission to 755

```
# chmod 755 /etc/udev/scripts/iscsidev.sh
```

Restart ISCSI server

```
# service iscsi restart
```

Check the new device created.

```
# ls /dev/iscsi/qdisk/part
```

In case the device is not going to be used as Quorum disk next command need to be executed to make the ext3 format.

```
# mkfs.ext3 -b 4096 /dev/iscsi/sosalconf/part
```

8.2 Create SCSI target using Openfiler

It's possible to simulate a SAN using the linux distribution named Openfiler. In this guide, we'll provide the steps to configure SCSI targets. The Openfiler can be useful for test environment but never has to be used into a production environment.

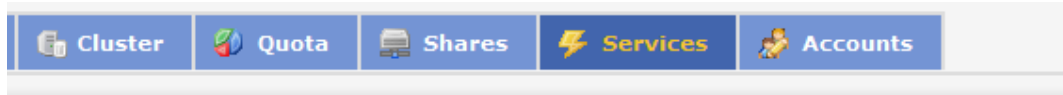
Download the iso image to install the Openfiler linux, in this guide the version "openfileresa-2.99.1-x86_64-disc1.iso" has been used. It's mandatory at least to configure two hard disks, one for the OS and another one for shared storage.

8.2.1 Configure the Openfiler

Once Openfiler is installed a web interface is published on https://ip_address:446 the default username is "openfiler" and the default password is "password".

Before to configure the first SCSI is required to next steps:

1. Enable and start iSCSI target services



Manage Services

Service	Boot Status	Modify Boot	Current Status	Start / Stop
CIFS Server	Disabled	Enable	Stopped	Start
NFS Server	Disabled	Enable	Stopped	Start
RSync Server	Disabled	Enable	Stopped	Start
HTTP/Dav Server	Disabled	Enable	Running	Stop
LDAP Container	Disabled	Enable	Stopped	Start
FTP Server	Disabled	Enable	Stopped	Start
iSCSI Target	Enabled	Disable	Running	Stop
UPS Manager	Disabled	Enable	Stopped	Start
UPS Monitor	Disabled	Enable	Stopped	Start
iSCSI Initiator	Disabled	Enable	Stopped	Start
ACPI Daemon	Enabled	Disable	Running	Stop
SCST Target	Disabled	Enable	Stopped	Start
FC Target	Disabled	Enable	Stopped	Start
Cluster Manager	Disabled	Enable	Stopped	Start

2. Add access to the nodes or networks will use the iSCSI targets (system tab).

Network Access Configuration

Delete	Name	Network/Host	Netmask	Type
<input type="checkbox"/>	storageNetwork	10.77.66.0	255.255.255.0	Share
<input type="checkbox"/>	clusterNetwork	10.77.77.0	255.255.255.0	Share
<input type="checkbox"/>	applNetwork	10.77.88.0	255.255.255.0	Share
New	<input style="width: 80%;" type="text"/>	<input style="width: 80%;" type="text"/>	<input style="width: 80%;" type="text" value="0.0.0.0"/> ▼	Share ▼

3. Create a new volume group. Ideally, this server needs at least to have two hard disks, one for OS and the other one for the new volume group.

Create a partition in /dev/sdb

💡 You can use ranges within the following extents:

Mode	Starting cylinder	Ending cylinder	Space
Primary	1	522	4.00 GB

Mode	Partition Type	Starting cylinder	Ending cylinder	Size	Create	Reset
Primary ▼	Physical volume ▼	<input style="width: 80%;" type="text" value="1"/>	<input style="width: 80%;" type="text" value="522"/>	<input style="width: 80%;" type="text" value="4 GB"/>	<input type="button" value="Create"/>	Reset

 **Volumes**  **Cluster**  **Quota**  **Shares**  **Services**  **Accounts**

Volume Group Management

Volume Group Name	Size	Allocated	Free	Members	Add physical storage
-------------------	------	-----------	------	---------	----------------------

Create a new volume group

 Valid characters for volume group name: **A-Z a-z 0-9 _ + -**

Volume group name (no spaces)

Select physical volumes to add

<input checked="" type="checkbox"/>	/dev/sdb1	3.81 GB
-------------------------------------	-----------	---------

8.2.2 Create SCSI target

1. Add new volume that will be associated to the SCSI target.

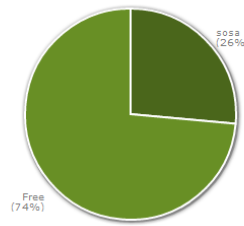


2. Set the name and the space. The type filesystem/volumen type needs to be "block(iSCSI,FC,etc)".

Create a volume in "scsi"

Volume Name (*no spaces*. Valid characters [a-z,A-Z,0-9]):	<input type="text" value="sosa"/>
Volume Description:	<input type="text" value="sosa binaries"/>
Required Space (MB):	<input type="text" value="1024"/>
Filesystem / Volume type:	<input type="text" value="block (iSCSI,FC,etc)"/> ▼

Volumes in volume group "scsi" (3872 MB)



Volume name	Volume description	Volume size	File system type	File system size	FS used space	FS free space	Delete	Properties	Snapshots
sosa	sosa binaries	1024 MB	iSCSI	Not applicable	Not applicable	Not applicable	Delete	Edit	Create
0 MB allocated to snapshots									
2848 MB of free space left									

3. Create a new iSCSI target



Add new iSCSI Target

Target IQN	Add
<input type="text" value="iqn.2006-01.com.openfiler:sosa.conf"/>	<input type="button" value="Add"/>

4. Make the LUN Mapping to the generated selecting the appropriate volume

Target Configuration LUN Mapping Network ACL CHAP Authentication

LUNs mapped to target: iqn.2006-01.com.openfiler:sosa2.conf



No LUNs mapped to this target

Map New LUN to Target: "iqn.2006-01.com.openfiler:sosa2.conf"

Name	LUN Path	R/W Mode	SCSI Serial No.	SCSI Id.	Transfer Mode	Map LUN
sosa binaries	/dev/scsi/sosa	write-thru	QZZXDH-uBPs-cB5b	QZZXDH-uBPs-cB5b	blockio	Map
quorum qdisk	/dev/scsi/qdisk	write-thru	6dkC0L-89eI-6tnY	6dkC0L-89eI-6tnY	blockio	Map
quorum disk1	/dev/site1/qdisk1	write-thru	AqeJdI-sEnB-1Opb	AqeJdI-sEnB-1Opb	blockio	Map
quorum disk2	/dev/site2/qdisk2	write-thru	O6iGkk-SJYW-Or5m	O6iGkk-SJYW-Or5m	blockio	Map
sosa1 persistence file1	/dev/site1/sosa1persistence1	write-thru	Bvw0Om-jK0t-DxTq	Bvw0Om-jK0t-DxTq	blockio	Map
sosa1 configuration	/dev/site1/sosa1conf	write-thru	BDun3J-38nd-oXWZ	BDun3J-38nd-oXWZ	blockio	Map
sosa2 persistence1	/dev/site2/sosa2persistence1	write-thru	pLW0Jd-GUOZ-5ozL	pLW0Jd-GUOZ-5ozL	blockio	Map
sosa2 configuration	/dev/site2/sosa2conf	write-thru	U2yukN-5hV0-mwdT	U2yukN-5hV0-mwdT	blockio	Map

5. Grant access to the client node/networks.

Status System Volumes Cluster Quota Shares Services Accounts

Target Configuration LUN Mapping Network ACL CHAP Authentication

iSCSI host access configuration for target "iqn.2006-01.com.openfiler:tsn.612782bc69c7"

Name	Network/Host	Netmask	Access
rhcl3	10.77.77.13	255.255.255.255	Allow
rhcl3-appl	10.77.88.153	255.255.255.255	Allow
rhcl4-appl	10.77.88.154	255.255.255.255	Allow
rhcl4	10.77.77.14	255.255.255.255	Allow

Update