Setting Up Prime Central High Availability in a VMware or KVM Environment

Revised: Month Day, Year,

Installation Overview

The Prime Central RHCS local redundancy HA configuration in a VMware and KVM environment has the following characteristics and requirements:

- Prime Central and the embedded database are installed in a dual-node cluster. Figure 2-1 shows a basic dual-node, local redundancy cluster.
- RHCS must be installed on both cluster nodes. Each node has the platform to run both Prime Central and the database services. RHCS manages the local HA setup.
- RHCS requires a disk resource that is mountable from both nodes.
- The Prime Central installer places Prime Central and the database on the node where you ran the installation. After installation, you can relocate them as needed.
- Prime Central and the database services must be placed on the same server.
- Prime Central and the embedded database are always mounted with external shared storage.
- Prime Central does not recognize RHCS. RHCS continuously obtains the cluster status by running a set of scripts. If a problem occurs, RHCS unmounts and then remounts the appropriate server and database. Therefore, every node in the HA setup must be able to mount the storage.

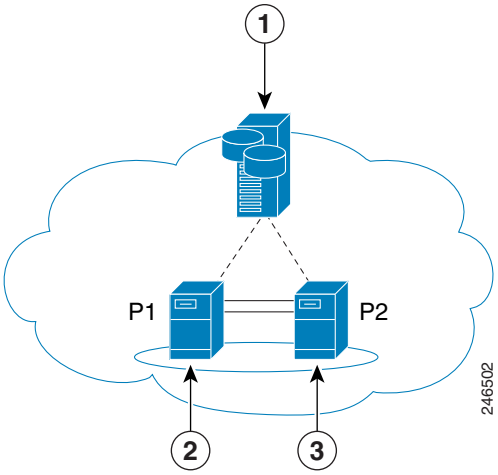
Note the following:

- The Prime Central and oracle home directories must be created manually under the mounted storage. This ensures that the operating system (OS) user created on both servers has a home directory available, even if the storage is moved to another node. These directories must have relevant permissions for the network user and oracle user.
- Each service has its own virtual IP address (virtual IP). This means Prime Central clients treat a failover or switchover like a local service restart.
- Only one instance of the Prime Central files exists, and it is located on the shared storage. Duplicate user and home directories are created on each node as placeholders. If a switchover occurs, the storage unmounts from one node and mounts on the other.

- A local redundancy configuration requires a *fencing* hardware unit for cutting off a node from the shared storage. Fencing ensures data integrity and prevents a "split-brain scenario" where servers are disconnected from each other, and each assumes the other server failed. If a failure occurs, the cutoff can be performed by:
  - Powering off the node with a remote power switch
  - Disabling a switch port fiber channel
  - Revoking a host's SCSI 3 reservations

  If a problem with the cluster node occurs, RHCS invokes the fencing device with the peer and waits for the success signal.

Figure 2-1 Prime Central Dual-Node, Local Redundancy Cluster



1	External storage	**3**	Local cluster node 2
2	Local cluster node 1		

RHCS Components and Functionality

RHCS is included with the Red Hat Enterprise Linux 6.5, 6.7, or 6.8 (RHEL 6.5, 6.7, or 6.8)Advanced Program and has the following components:

- Cluster infrastructure—Basic functions that enable a group of computers (nodes) to work together as a cluster. The RHCS cluster infrastructure performs cluster management, lock management, fencing, and cluster configuration management.
- High availability service management—Failover of services from one cluster node to another when a node becomes inoperative.
- Cluster administration tools—Configuration and management tools for setting up, configuring, and managing a Red Hat cluster, including cluster infrastructure components, high availability service management components, and storage.

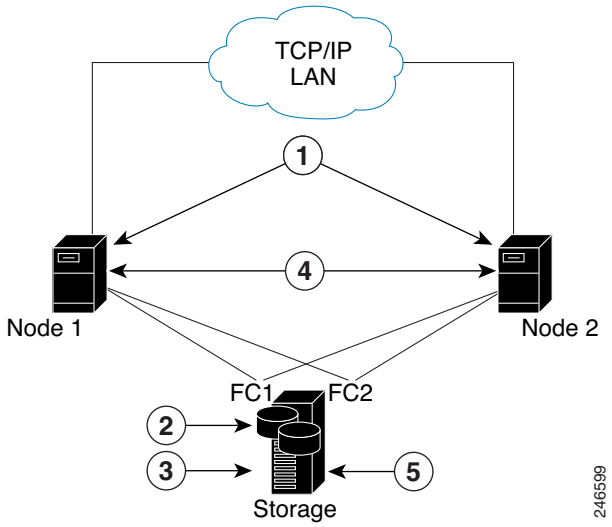
RHCS HA Local Redundancy Requirements

Table 2-1 Prime Central RHCS HA Local Redundancy Requirements

Area	Requirement
OS	Red Hat Enterprise Linux 6.5, 6.7, or 6.8 with the Red Hat Clustering Suite.
Oracle	12cR1 Enterprise Edition. **Note** Oracle 12cR1 EE is included in the Prime Central embedded database installation.
Hardware	RHEL 6.5, 6.7, or 6.8 certified platform with fencing capabilities.
Network	• Cluster nodes must be able to communicate with each other using multicast. • Each network switch and the associated networking equipment in a Red Hat cluster must be configured to enable multicast addresses and support IGMP. Without multicast and IGMP, not all nodes can participate in a cluster, causing the cluster to fail. • Network timing must be configured. **Note** During the RHCS HA installation, you will be asked to confirm that NTP timing is configured.
Storage and file system	RHCS requires shared Fibre Channel, Fibre Channel over Ethernet (FCoE), or Internet Small Computer System Interface (iSCSI) storage that presents an ext4 file system and is accessible from all cluster nodes.
Disk space	5 GB under /tmp.
Miscellaneous	• The same subnet must be assigned to all nodes, including all services (virtual IP addresses in the same subnet). • All nodes must have RHEL 6.5, 6.7, or 6.8 installed. • All systems must be homogeneous with the same configuration versions and packages. • Shared storage must not be auto-mounted, because RHCS performs the mounting. Use one partition for each cluster service. For a single shared disk, use a single partition for each service on the same disk. In other words, the shared storage must not appear in /etc/fstab. • All shared storage units must be configured with a label, which RHCS uses to mount and unmount storage. • Virtual IP addresses must be assigned for each service. IP addresses assigned to services should not be attached to any server. RHCS will manage the IP addresses; that is, RHCS will add and remove the IP addresses from the server that is running the service. • Fencing devices must be deployed. • Multicast communication must not be blocked by a firewall.

We strongly recommend that you use a hardware installation designed to avoid a single point of failure. See Figure 2-2.

Figure 2-2 Local Redundancy Hardware Installation Designed to Avoid a Single Point of Failure

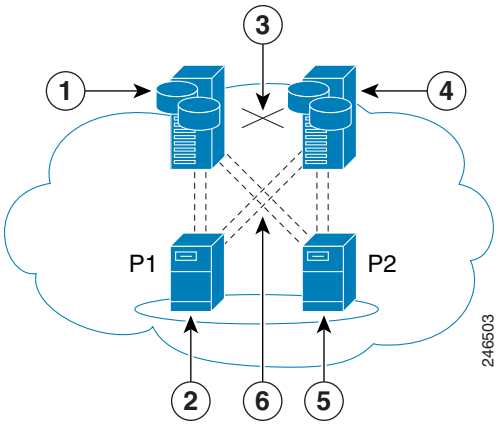


1	Dual NICs	**4**	NIC bonding (active/backup mode)
2	Disk mirroring	**5**	Each NIC connects to a separate switch
3	Redundant RAID controllers		

Configuring Hardware for Prime Central RHCS Local Redundancy HA

Figure 2-3 shows the recommended hardware configuration for Prime Central RHCS local redundancy HA.

Figure 2-3 Prime Central Dual-Node Cluster for Local Redundancy HA



1	Prime Central external storage: one Prime Central volume	**4**	Oracle external storage: • One to three data volumes • One archive volume • Zero to three redo log volumes
2	Prime Central server: • Two internal disks • One OS • One mirror	**5**	Oracle database server: • Two internal disks • One OS • One mirror
3	Dual Gigabit Ethernet crossover connections	**6**	Dual connections from each server to each external disk storage unit

Configure the external storage so all disks and logical unit numbers (LUNs) are accessible from both servers in the cluster. The disk and LUN configuration depends on the storage type:

- If you are using JBOD disks, provide enough physical disks to create the volumes listed in Table 2-2 to satisfy the Oracle performance requirements.
- If you are using storage that supports hardware RAID, divide the physical disks into LUNs so that the volumes listed in Table 2-2 can be created and configured to satisfy the Oracle performance requirements and can be protected with RAID5, RAID1, or RAID10. The Oracle volumes can be created on a single LUN.

Table 2-2 Volume Sizes

Volume	Minimum Size	Comments
Prime Central	50 GB	—
Oracle application + data files	10 GB	—
Oracle redo logs	12.8 GB	—
Oracle archives	20 GB	Contact your Cisco account representative for information.
Oracle additional data files (if used)	—	Based on Prime Central alarm history.
Oracle backup	50 GB	—

Configuring the OS for Prime Central Local Redundancy HA Managed by RHCS

1. Install the OS and all recommended patches on both servers in the cluster. Installations on both servers must be identical.
2. Verify that access is available to all external disks.
3. Create the internal disk partitions listed in Table 2-3. Placing the individual directories in separate partitions is recommended, but not required.
4. Complete the internal disk partitions for both servers.
5. Keep the nodes synchronized:

```
# echo server tick.redhat.com$'\n'restrict tick.redhat.com mask 255.255.255.255
nomodify notrap noquery >> /etc/ntp.conf
# chkconfig ntpd on
# service ntpd start
```

Table 2-3 Disk Partitions

Partition	Space Required (MB)
swap	Standard amount of space, as per the system configuration
/tmp	Standard amount of space + 5120
/	Standard amount of space + 6144
/var	Standard amount of space + 1024 for HA utilities
/usr/local/bin	Standard amount of space + 200 for cluster utilities
/etc	Standard amount of space + 200 for the cluster configuration

Installing the Red Hat Cluster Service

Using the procedures in the Red Hat user documentation, install RHEL 5.8/6.5 with the RHCS. When you set up the RHCS disk groups and volumes, keep the following in mind:

- All of the shared storage should have an ext3 file system installed and a label set.
- Shared storage must be accessible from all cluster nodes.

Verifying the Prime Central RHCS HA Installation

Table 2-4 Local Redundancy Verification Tests

Description	Procedure	Expected Results
Local Cluster Hardware Failure		
Name: Cluster node hardware failure. Purpose: Test the local site failover (including fence test) due to node failure.	1. Aggressively power off the active node that runs both services (Prime Central and DB). 2. Verify that both services are relocated to the redundant node.	Within several minutes, the redundant cluster node identifies that the active node is not available and fences it, evicting it from the cluster and relocating all the services to the only remaining node.
Manual Cluster Administration		
Name: Manual service stop. Purpose: Verify that the service can be stopped manually.	1. Enter **clusvcadm -d** *service-name*. 2. Verify that the service is not running and no errors appear in the cluster log (/var/log/messages for both cluster nodes).	The stopped service is no longer running.
Name: Manual service start. Purpose: Verify that the service can be started manually.	1. Enter **clusvcadm -e** *service-name*. 2. Verify that the service is running and no errors exist in the cluster log (/var/log/messages on both cluster nodes).	The service is running.
Name: Manual service restart. Purpose: Verify that the service can be restarted manually.	1. Enter **clusvcadm -R** *service-name*. 2. Verify that the service is running and no errors exist in the cluster log (/var/log/messages on both cluster nodes).	The service is running.
Name: Manual service relocation. Purpose: Verify that the service can be relocated manually.	1. Enter **clusvcadm -r** *service-name*. 2. Verify that the service is not running on the current node and is running on the standby node. 3. Verify that no errors appear in the cluster log (/var/log/messages on both cluster nodes). The service is stopped on the active node and then started on the redundant node. 4. Test both the Prime Central and Oracle services.	The service is stopped on the active node and started on the redundant node.

Table 2-4 Local Redundancy Verification Tests (continued)

Description	Procedure	Expected Results
Ordered Cluster Node Startups		
Name: Node startup in existing cluster. Purpose: Verify that a cluster node starts up and rejoins a cluster after it is restarted.	1. Restart one of the cluster nodes. 2. Verify that the node joins the cluster after the reboot. 3. Relocate one of the services to the rebooted node and verify that it is running. 4. Check the log for errors.	The rebooted node joins the cluster and runs the services.
Name: Simultaneous node startup. Purpose: Verify that the cluster is set up correctly when both nodes start simultaneously.	1. Start both nodes from the power off state. Verify that both nodes appear in the cluster after they are up, with both services running on the cluster. 2. Check the log for errors.	Both cluster nodes join the cluster; both services are running.
Name: Single-node startup. Purpose: Test the cluster functionality when only one is node running.	1. Power down both nodes and then start one of them. 2. The running node fences the other node and runs the services. The fenced node joins the cluster to create the dual-node cluster. 3. Check the log for errors.	Both cluster nodes join the cluster; both services are running.
Local Cluster Service Failure		
Name: Service failure. Purpose: Test the service startup after a failure occurs.	1. Simulate a service failure by stopping its processes or shutting down the Oracle listener. 2. Verify that the service restarts on the same node where it was running. 3. Check the log for errors. 4. Test both the Prime Central and Oracle services.	The service is restarted on the same node.

Table 2-4 Local Redundancy Verification Tests (continued)

Description	Procedure	Expected Results
Local Cluster Hardware Failure		
Name: Stop node with fencing off. Purpose: Verify that the node requires manual fencing after the other node, including its fencing agent, is removed.	1. Disconnect the fencing agent from one of the nodes and then power it off. 2. Observe the other node behavior. 3. Check the log for errors and the request for manual fencing.	The fence_ack_manual required notification appears in the log files. The cluster is running with one node and all services running on it.
Name: Single-node cluster. Purpose: Verify that the cluster can function when the other node does not exist or has no power.	1. Power down both nodes. 2. Disconnect the fencing agent from one of the nodes. 3. Start the other node. It attempts to fence the other node, but fails with the regular fencing agent. Manual fencing is required. 4. Acknowledge the manual fencing.	The cluster does not start the services (and does not show in the **clustat** command) before acknowledging that manual fencing is performed.

Installing Prime Central and an Embedded Database in a Local Redundancy HA Configuration

Installing Prime Central in an RHCS local HA configuration is a three-part process:

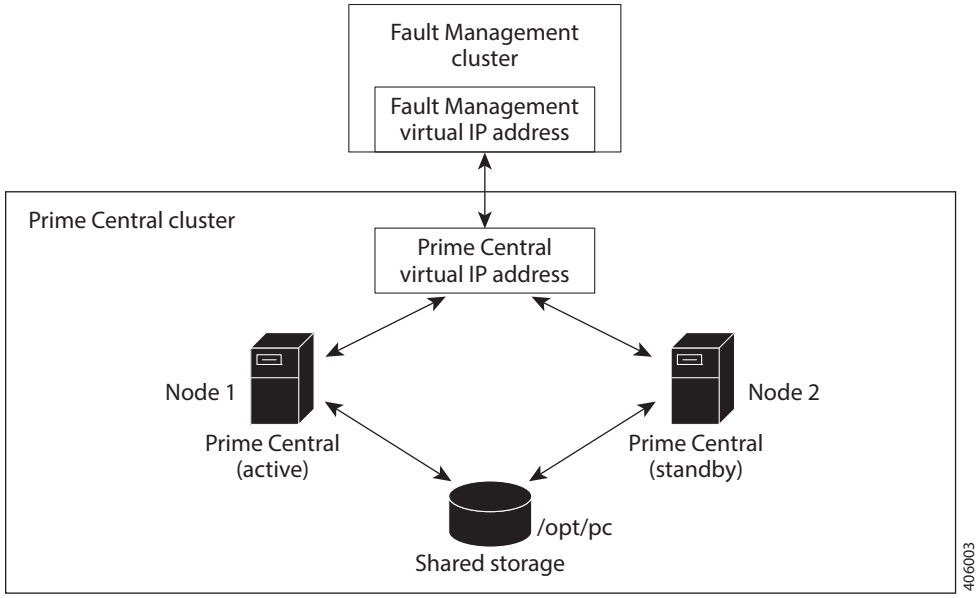
1. Install RHEL 6.5, 6.7, or 6.8 on both nodes.
2. Use multipath shared storage and install Prime Central on node 1.
3. Configure and enable clustering so that Prime Central can relocate between nodes.

The examples provided use the following hostnames and IP addresses; yours will be different:

- Node 1—prime-pc-node1.cisco.com (192.168.1.110)
- Node 2—prime-pc-node2.cisco.com (192.168.1.120)
- Virtual IP address—prime-pc-service.cisco.com (192.168.1.130)
- Gateway—192.168.1.1
- Domain Name System (DNS)—192.168.1.2
- luci—prime-pc-luci.cisco.com (192.168.1.140)

Figure 2-4 shows an example of a Prime Central cluster in an HA configuration.

Figure 2-4 Prime Central Cluster in an HA Configuration



Before You Begin

- Ensure that both PC and FM are installed on different VMs.
- Verify that your system meets all the hardware and software requirements in "Installation Requirements" in the *Cisco Prime Central 2.0.0 Quick Start Guide*.
- Set up two nodes that have:
  - Static IP addresses and hostnames that are registered correctly in the DNS.
  - The same root password, which cannot contain a percent sign (%, ^, $, *).
- Set up one virtual IP address and hostname that are registered correctly in the DNS. In this section, the virtual IP address is 192.168.1.130.
- Set up shared storage that is compatible with RHEL device-mapper (DM) multipath and cluster fencing.
- Install RHEL 6.5, 6.7, or 6.8 on both nodes.
- If the default folder location and names are changed, look for the section "Require Manual Definition" to make corresponding changes in the below mentioned files:
  - /root/ha-stuff/pc/PrimeCentral.sh
  - /root/ha-stuff/pc/UninstallPrimeCentral.sh
  - /usr/local/bin/pc.sh
  - /usr/local/bin/createUserGroup.sh

Prime Central HA Setup on RHEL 6.5 or RHEL 6.7 or 6.8

Step 1 Create three VMs with the configuration specified in the *Cisco Prime Central 2.0 Quick Start Guide*.

In this setup:

- Two VMs act as cluster nodes.
- A third VM runs luci, the web-based high-availability administration application. By running luci on a third system, you can still manage the cluster from another system if either the primary or standby node goes down.
- Shared network storage is required.

Step 2 Install RHEL 6.5 or 6.7 or 6.8 on all three VMs with the Desktop option selected.

Ensure that:

- All three VMs can ping each other via IP address and hostname.
- Both the active and standby cluster nodes have a 200 GB shared repository, which is located in the /dev/sdb directory.

Step 3 Disable the firewall on all three VMs:

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

Step 4 On both cluster nodes, switch the network daemons:

```
# service NetworkManager stop
# chkconfig NetworkManager off
# chkconfig network on
```

Step 5 On both cluster nodes, make sure that the nscd RPM package is not installed.

a. Log in as the root user.

b. Check nscd RPM is installed or not:

```
#rpm -qv nscd
```

c. Enter the following command to uninstall:

```
#yum remove nscd
```

Step 6 Disable Security-Enhanced Linux (SELinux) on all three VMs:

```
# vi /etc/selinux/config
SELINUX=disabled
```

Step 7 Reboot the VMs.

Adding Clustering to the Installed Red Hat Server (Prime Central)

To add clustering to the newly installed Red Hat server, complete Steps 1 through 6 of the following procedure on both cluster nodes and the VM running luci. Then complete Steps 7 through 12 on just the cluster nodes.

Note Below steps are specific for RHEL 6.5 and this procedure is also supported for RHEL 6.7 and 6.8. You have to change the folder names and .iso file names accordingly for RHEL 6.7 and 6.8.

Step 1 Create local directories named /rhel and /cdrom-6.5.

Step 2 Copy the .iso file that was used for the virtual machine (VM) RHCS installation to the /rhel directory.Mount the /rhel .iso file to /cdrom:

```
# cd /rhel
# mount -t iso9660 -o loop /rhel/rhel-server-6.5-x86_64-dvd.iso /cdrom-6.5
```

Note To permanently mount the drive, update the /etc/fstab file. See http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/4/html/Introduction_To_System_Administration/s2-storage-mount-fstab.html.

Step 3 Create a file named /etc/yum.repos.d/local.repo. Use UNIX format and be sure there are no spaces before lines.

Step 4 Save the newly created file in local.repo, as follows:

```
[local]
name=Red Hat Enterprise Linux $releasever - $basearch - Local
baseurl=file:///cdrom-6.5/Server
enabled=1
gpgcheck=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release

[HighAvailability]
name=Red Hat Enterprise Linux $releasever - $basearch - HighAvailability
baseurl=file:///cdrom-6.5/HighAvailability
enabled=1
gpgcheck=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release

[ResilientStorage]
name=Red Hat Enterprise Linux $releasever - $basearch - ResilientStorage
baseurl=file:///cdrom-6.5/ResilientStorage
enabled=1
gpgcheck=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
```

Step 5 Add the information for both cluster nodes and the VM running luci to the /etc/hosts file; for example:

```
192.168.1.110     prime-pc-node1.cisco.com     prime-pc-node1
192.168.1.120     prime-pc-node2.cisco.com     prime-pc-node2
192.168.1.140     prime-pc-luci.cisco.com     prime-pc-luci
```

Step 6 Generate a Secure Shell (SSH) key for the root user:

```
# chmod 755 ~
# ssh-keygen -t rsa -N "" -b 2047 -f ~/.ssh/id_rsa
# chmod 600 ~/.ssh/id_rsa
```

Step 7 (On the first node only) Share the node's public key with the other node so that dynamically creating a secure shell between the nodes does not prompt for a password:

```
# rsync -av ~/.ssh/id_rsa.pub ~/.ssh/authorized_keys
# ssh root@prime-ha-node2.cisco.com "cat ~/.ssh/id_rsa.pub" >> ~/.ssh/authorized_keys
# rsync -av ~/.ssh/authorized_keys root@prime-ha-node2.cisco.com:/root/.ssh/
```

Step 8 Verify that the .ssh directory has 700 permission and the .ssh/id_rsa file has 600 permission:

```
# chmod 700 ~/.ssh
# chmod 600 ~/.ssh/id_rsa
```

Step 9 Verify that your SSH is working without an authentication or password prompt:

Caution The Prime Central service will not start if SSH prompts for authentication or a password. Be sure to complete all of the following substeps.

a. On node prime-ha-node1.cisco.com, enter:

```
# ssh root@prime-ha-node2.cisco.com
# exit
# ssh root@prime-ha-node2
# exit
# ssh root@192.168.1.120
# exit
```

b. On node prime-ha-node2.cisco.com, enter:

```
# ssh root@prime-ha-node1.cisco.com
# exit
# ssh root@prime-ha-node1
# exit
# ssh root@192.168.1.110
# exit
```

c. If you are prompted for a password, check the permissions of all folders and files that you modified in the preceding steps.

d. If you are prompted to continue connecting, enter **yes**. (The prompt should appear only the first time you use SSH to connect to the node.)

Step 10 Verify that the virtual IP address is accessible from outside the cluster's subnet:

```
# ip addr add 192.168.1.130 dev eth0
```

Step 11 On a computer outside the cluster's subnet, ping the virtual IP address:

```
# ping 192.168.1.130
# ip addr del 192.168.1.130 dev eth0
```

If you do not get a valid response, determine which part of the OS or network setup is blocking.

Adding Shared Partitions to Prime Central

To add shared partitions, complete the following steps in parallel on both nodes, except where noted:

Note The examples provided use device mapping names such as mpatha and mpathap1; yours may be different.

Step 1 Install and Set up multipath. Execute the below commands in sequence:

```
# yum install device-mapper-multipath
```

```
# mpathconf --enable --user_friendly_names y
# modprobe dm-multipath
# service multipathd start
# chkconfig multipathd on
# vi /etc/multipath.conf
    -- Comment out 'blacklist' section
         -- For example:
         -- #blacklist {
         # devnode "*"
         #}
         -- Append below lines at the end of file
        devices {
               device {
                   vendor "DGC"
                   product ".*"
                   product_blacklist "LUNZ"
                   path_checker "directio"
                   path_grouping_policy "group_by_prio"
                   path_selector "round-robin 0"
                   failback "immediate"
                   }
              }
# service multipathd restart
# multipath -v2
# multipath -ll
# service multipathd reload
# modprobe dm-multipath
# service multipath restart
```

Step 2 Check for available disks:

```
cd /dev/mapper/
ls -la
total 0
drwxr-xr-x  2 root root    160 Jan  6 15:19 .
drwxr-xr-x 18 root root   3820 Jan  6 15:20 ..
crw-rw----  1 root root 10, 58 Jan  6 15:19 control
lrwxrwxrwx  1 root root      7 Jan  6 15:19 mpatha -> ../dm-3
lrwxrwxrwx  1 root root      7 Jan  6 15:19 vg_primepcnode2-lv_home -> ../dm-2
lrwxrwxrwx  1 root root      7 Jan  6 15:19 vg_primepcnode2-lv_root -> ../dm-0
lrwxrwxrwx  1 root root      7 Jan  6 15:19 vg_primepcnode2-lv_swap -> ../dm-1
```

In the output, note *mpath2*, which is the multipath virtual device or disk that you will use later as shared storage.

Note If you previously set up a partition on the disk, you might see output such as *mpath2p*. You must delete that partition before proceeding to the next step.

Step 3 (On the first node only) Create a shared partition:

```
# fdisk mpatha
Command (m for help): p
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
```

```
Partition number (1-4): 1
First cylinder (1-19581, default 1): <Enter>
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-19581, default 19581): <Enter>
Command (m for help): w
```

Step 4 Reboot both nodes.

Step 5 Check for new partitions:

```
# cd /dev/mapper/
# ls -la
total 0
drwxr-xr-x  2 root root    160 Jan  6 15:19 .
drwxr-xr-x 18 root root   3820 Jan  6 15:20 ..
crw-rw----  1 root root 10, 58 Jan  6 15:19 control
lrwxrwxrwx  1 root root      7 Jan  6 15:19 mpatha -> ../dm-2
lrwxrwxrwx  1 root root      7 Jan  6 15:19 mpathap1 -> ../dm-3
lrwxrwxrwx  1 root root      7 Jan  6 15:19 vg_primepcnode2-lv_home -> ../dm-4
lrwxrwxrwx  1 root root      7 Jan  6 15:19 vg_primepcnode2-lv_root -> ../dm-0
lrwxrwxrwx  1 root root      7 Jan  6 15:19 vg_primepcnode2-lv_swap -> ../dm-1
```

Step 6 (On the first node only) Format the new shared partition:

```
# mkfs.ext4 /dev/mapper/mpathap1
```

Step 7 Create target locations on both nodes:

```
# mkdir -p /opt/pc
```

Step 8 Verify that both nodes can mount and unmount the shared storage:

a. On the first node, mount the shared storage and save a file that contains only the value *1* to the shared storage. The test.txt file should exist in the list of contents of /opt/pc:

```
# mount /dev/mapper/mpathap1 /opt/pc
# vi /opt/pc/test.txt
1
:wq
# ls -la /opt/pc
# umount /opt/pc
```

b. On the second node, mount the shared storage and verify that the test.txt file exists and contains the value *1*:

```
# mount /dev/mapper/mpathap1 /opt/pc
# vi /opt/pc/test.txt
:q
# umount /opt/pc
```

If you cannot mount or unmount the shared storage, or if the test.txt file does not exist when you mount it to the second node, your multipath is not set up correctly.

Accessing and Distributing the Prime Central .tar File

Step 1 Insert the Cisco Prime Central 2.0.0 USB, navigate to the High Availability/RHCS KVM VMWare ESXi Local HA/Prime Central, and locate the primecentral_v2.0.0_ha_vm.tar.gz file.

Step 2 Use SSH to connect to the first node.

Step 3 Copy the primecentral_v2.0.0_ha_vm.tar.gz file to the first node.

Step 4 Back up the following directories on both nodes:

- /root/ha-stuff/pc
- /usr/local/bin

Step 5 Distribute the file on both nodes:

```
# tar -zxf primecentral_v2.0.0_ha_vm.tar.gz -C / --owner root --no-same-owner
```

Installing Prime Central in an HA Setup

Step 1 Mount the shared partitions on first node:

```
# mount /dev/mapper/mpath2p1 /opt/pc
```

Step 2 Navigate to the Base Application folder and copy primecentral*.bin and all Oracle zip files to the **/root/ha-stuff/pc** folder:

```
# chmod 755 /root/ha-stuff/pc/*
# chmod 755 /usr/local/bin/*
```

Step 3 Add a virtual IP cluster service address for the Prime Central service:

```
# ip addr add 192.168.1.130 dev eth0
```

Make sure that virtual IP address is reachable.

Step 4 Update the install.properties file (located at /root/ha-stuff/pc) and verify that all required properties have values. Refer to the comments at the top of the install.properties file for details.

Step 5 Install Prime Central:

```
# cd /root/ha-stuff/pc
# ./PrimeCentral.sh 192.168.1.130 first-node-root-password second-node-IP-address
```

Step 6 In another terminal window, check the installation process:

```
# tail -f /tmp/primecentral_install.log
```

Step 7 After the installation completes, start Prime Central:

```
# /usr/local/bin/pc.sh start
```

Step 8 Verify that Prime Central is running correctly; then, stop it:

```
# /usr/local/bin/pc.sh stop
```

Step 9 Remove the virtual IP addresses:

```
# ip addr del <Virtual-IP>  dev eth0
```

Step 10 Unmount the shared partitions:

```
# umount /opt/pc
```

Setting Up the Prime Central Cluster Service

To set up the Prime Central cluster service, complete the following steps in parallel on all the nodes including luci, except where noted:

Step 1 Install the ricci RPMs:

```
# yum -y install ricci
```

Step 2 Start the ricci daemon and configure it to start on boot:

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

Step 3 Set the ricci user password:

```
# passwd ricci
```

You will need to enter this password later.

At this point, both cluster nodes should be running the ricci servers and be ready to be managed by the cluster web user interface (luci).

Installing the Cluster Web User Interface (luci)

Complete the following procedure to install and configure luci on the third VM:.

Step 1 Install the luci RPMs:

```
# yum -y install luci
```

Step 2 Start the luci daemon and configure it to start on boot:

```
# chkconfig luci on
# service luci start

The following output is displayed:
Adding following auto-detected host IDs (IP addresses/domain names), corresponding to
`prime-ha-luci.cisco.com' 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://prime-ha-luci.cisco.com:8084 (or equivalent) to access
luci
```

Step 3 Launch the URL listed in the last line of the system output (https://prime-ha-luci.cisco.com:8084, in this example) and log in as the root user when prompted.

Step 4 Verify that the RHEL DVD is mounted on both cluster nodes.

Step 5 Select **Manage Clusters > Create** and then specify a name for the cluster (for example, PrimePCCluster).

Step 6 Enter the name (fully-qualified domain name or name specified in the /etc/hosts directory) and password (the password for the user *ricci*) for the first cluster node.

Step 7 Click **Add Another Node** and enter the same information described in Step 6 for the second cluster node.

Step 8 Do the following, and then click **Create Cluster**:

- **Use the Same Password for All Nodes**: Leave this check box unchecked.
- **Download Packages**: Select this radio button.
- **Reboot Nodes Before Joining Cluster**: Select this check box.
- **Enable Shared Storage Support**: Leave this unchecked.

Assuming that the nodes can be contacted, luci will set up each cluster node and add each node to the cluster.

After each node has been set up, the High Availability Management screen opens.

Step 9 Create a failover domain:

a. Select the **Failover Domains** tab and then click **Add**.

b. Do the following:

- **Name**: Enter a name for the failover domain.
- **Prioritized**: Select this check box.
- **Restricted**: Select this check box.
- **No Failback**: Select this check box.
- **Member**: Select the check box for each node.
- **Priority**. In the Priority column, enter *1* for node1 and *2* for node2.

c. Click **Create**.

Step 10 Configure fence devices appropriate for the hardware you have, adding a fence device and instance for each node.

The settings you need to configure depend on your particular hardware and software configuration. Refer to the Cluster Administration Guide for help with configuring fence devices.

To configure VMware fencing, complete the following steps:

a. Retrieve the UUID for prime-pc-node1.cisco.com:

```
fence_vmware_soap -v -z -a VCenter-IP-Address -l VCenter-Username -p VCenter-Password
--action list  | grep VM-Name
```

This returns: *VM-Name*,UUID

Keep in mind that *VM-Name* is the name given to the VM during VM creation, not its hostname.

b. Repeat the previous step for prime-pc-node2.cisco.com.

c. In luci, select the **Fence Devices** tab and then click **Add**.

d. Do the following:

- **Fence type**: Select VMWare Fencing (SOAP Interface).
- **Name**: Enter a descriptive name for the fence device.
- **IP Address or Hostname**: Enter the appropriate VCenter IP address or hostname.
- **IP Port** (optional): Leave blank.

- **Login**: Enter the VCenter username.
- **Password**: Enter the VCenter password.

Leave the rest of the fields as is or blank.

e. Click **Submit**.

f. Select the **Nodes** tab and then select your first node by clicking its name.

g. At the bottom of the node's window, click **Add Fence Method**.

h. Enter a name and then click **Submit**. Now select **Add Fence Instance** that appears inside the method box and fill it out as described below:

i. Select the fencing device you configured in Step 10d and do the following:

- **VM Name**: Leave blank.
- **VM UUID**: Enter the UUID you retrieved in Step 10a (for example, 4209838d-1e91-104c-9f5c-1181ad87ddd8).
- **Use SSL**: Select this check box. Fencing will not work unless you do so.

j. Repeat Steps 10f through 10i for your second node, as well.

Note We recommend that you check the bottom of this window for each node and verify that all of the required daemons are running before you test fencing.

To configure KVM fencing, complete the following steps:

a. Retrieve the UUID for prime-pc-node1.cisco.com by executing the below command on the KVM host:

```
virsh dominfo prime-pc-node1.cisco.com
```

b. Repeat the previous step for prime-pc-node2.cisco.com.

c. In luci, select the **Fence Devices** tab, and then click **Add**.

d. From the **Fence Type** drop-down list, choose fence_virt(Multicast Mode).

e. In the **Name** field, enter a descriptive name.

Note Leave the other fields as is or blank.

f. Click **Submit**.

g. Select the **Nodes** tab, and then select your primary node by clicking its name.

h. Click **Add Fence** method that is available at the bottom of the node window.

i. Enter a name, and then click **Submit**. Select Add Fence Instance that appears inside the method box and enter the details as specified below:

- Select the fencing device that you configured in step d, and then in the **Domain name** field, enter the domain name. For example, you can enter 4209838d-1e91-104c-9f5c-1181ad87ddd8.

Note Leave the **Delay** field as blank.

j. Click **Submit**.

Setting Up Shared Resources

You will next create the cluster service (named PrimePCService in this example). To begin, open the High Availability Management page (luci) in a browser. From there, you will:

- Identify the virtual IP address for this cluster
- Identify the PrimeService service and add the virtual IP address to this service

At several points, you will run commands from the cluster nodes (ricci) to verify that the configuration you set in luci is working.

Identifying the Cluster Service's Virtual IP Address

From luci, do the following to identify the cluster service's virtual IP address:

Step 1 Select the cluster name (for example, **PrimePCCluster**).

Step 2 Select the **Resources** tab and then click **Add**.

Step 3 Select **IP Address**.

Step 4 Enter the following information:

- **IP Address**: Specify a valid IP address, which will be used to access the service from a web browser.
- **Monitor Link**: Select this check box.

Step 5 Click **Submit**.

Creating the Prime Service

From luci, with the cluster still selected, add a new service and associate it with the IP address.

Step 1 Select the **Service Groups** tab and then click **Add**.

Step 2 Enter the following information for the new service group:

- **Service name**: Name of the service group (for example, **PrimePCService**)
- **Automatically start this service**: Select this check box.
- **Failover Domain**: Select the PCFailOver you created while installing luci.
- **Recovery Policy**: Select **Relocate**.

Step 3 Click **Add Resource**.

Step 4 Select the IP address you added earlier and then click **Submit**.

Step 5 Wait 5 minutes and then reboot both cluster nodes.

After the VMs come back online, the **PrimePCService** service should up and running.

Adding Mount Points and Setting Up File Systems

In the following procedure, you will create mount points on the two nodes. After the mount points have been set up, you can then configure the resources necessary to mount file systems to the proper locations.

Step 1 Create a file system resource:

a. From luci, select the **Resources** tab and then click **Add**.

b. From the Add Resource to Cluster window, select **Filesystem** from the dropdown list and then do the following:

- **Name**: Enter the name of the file system resource.
- **Filesystem** type: Select the ext4 option.
- **Mount point**: Enter the mount point path—/opt/pc
- Device, FS label, or UUID: Enter the appropriate path—in this example, /dev/mapper/mpathap1
- Leave the **Mount options** and **Filesystem ID (optional)** field blank.
- Select the **Force Unmount** check box.
- Select the **Reboot host node if unmount fails** check box.

c. Click **Submit**.

Step 2 Add the file system resource to the **PrimePCService** service group:

a. From luci, select the **Service Groups** tab.

b. In the Name column, select the **PrimePCService** service group name.

c. At the bottom of the page, click **Add Resource**.

d. Click **Select a Resource Type**.

e. The filesystem resources you added should appear in the list.

f. From the list, click the filesystem entries you added in the previous step.

g. Click **Submit**.

Step 3 Log in to the node currently running the PrimePCService service.

Step 4 Verify that the logical volume ext4 filesystems are mounted with read/write privileges on the node running the PCService service:

```
# cat /etc/mtab
```

Step 5 **Verify** that the logical volume ext4 filesystems are not mounted on the node which is not running the PCService service:

```
# cat /etc/mtab
```

Adding Scripts to the Prime Central Cluster Service

Step 1 Add the pc.sh script resource:

a. From luci, select the **Resources** tab and then click **Add**.

b. From the Add Resource to Cluster window, select **Script** from the dropdown list and then enter the following information:

- **Name**: Name of the script
- **Full Path to Script File:** Directory in which the script will reside—/usr/local/bin/pc.sh

Step 2 Add script resources to the **PrimePCService** service group:

a. From luci, select the **Service Groups** tab.

b. In the Name column, select the **PrimePCService** service group name.

c. At the bottom of the page, click **Add Resource**.

d. Click **Select a Resource Type**.

The script resource you added should appear in the list.

e. Click the script entries from the list you added in the previous step.

f. Click **Submit**.

Verifying the PrimeService Service Can Be Reached

From each of the cluster nodes, check the status of the PrimePCService services you created. Then, from any machine on the network, verify that you can reach the cluster from the IP address you assigned.

Step 1 From a shell on each of the cluster nodes, verify that the service group is running by entering the following commands (in bold):

```
# clustat

The output is similar to the following:
Cluster Status for PrimeCluster @ Fri Nov  8 15:09:42 2013
Member Status: Quorate

Member Name                  ID    Status
```

```
------ ----                  ----  ------
prime-pc-node1.cisco.com     1     Online, rgmanager
prime-pc-node2.cisco.com     2     Online, Local, rgmanager

Service Name                 Owner (Last)                 State
------- ----                 ------ ------                ------
service:PrimePCService       prime-pc-node1.cisco.com     started
```

Step 2 From a shell anywhere that can reach the cluster over the network, verify that you can ping the IP address associated with the IP address resource you created previously (see Creating the Prime Service, page 2-21):

```
ping 192.168.1.130
```

Note To end the ping, enter **Ctrl + C**.

Step 3 Relocate the service:

```
# clusvcadm -r PrimePCService
```

Step 4 Verify that the service has been relocated successfully to the other node.

Checking the Cluster Services

On both nodes, check the status of the cluster:

```
# clustat
```

The output is similar to the following:

```
Cluster Status for PrimeCluster @ Fri Nov  8 15:09:42 2013
Member Status: Quorate

Member Name                  ID    Status
------ ----                  ----  ------
prime-pc-node1.cisco.com     1     Online, rgmanager
prime-pc-node2.cisco.com     2     Online, Local, rgmanager

Service Name                 Owner (Last)                 State
------- ----                 ------ ------                ------
service:PrimePCService       prime-pc-node1.cisco.com     started
```

Next Steps

Complete the following steps on both nodes, except where noted:

Step 1 Verify that the required ports are open. For a list of ports that Prime Central requires, see "Prime Central Protocols and Ports" in the *Cisco Prime Central 2.0 Quick Start Guide*.

Step 2 Enable the firewall:

```
# service iptables start
# chkconfig iptables on
# service ip6tables start
# chkconfig ip6tables on
```

Installing Prime Central Fault Management in a Local Redundancy HA Configuration

Installing the Prime Central Fault Management component in a dual-node, RHCS HA configuration is a three-part process:

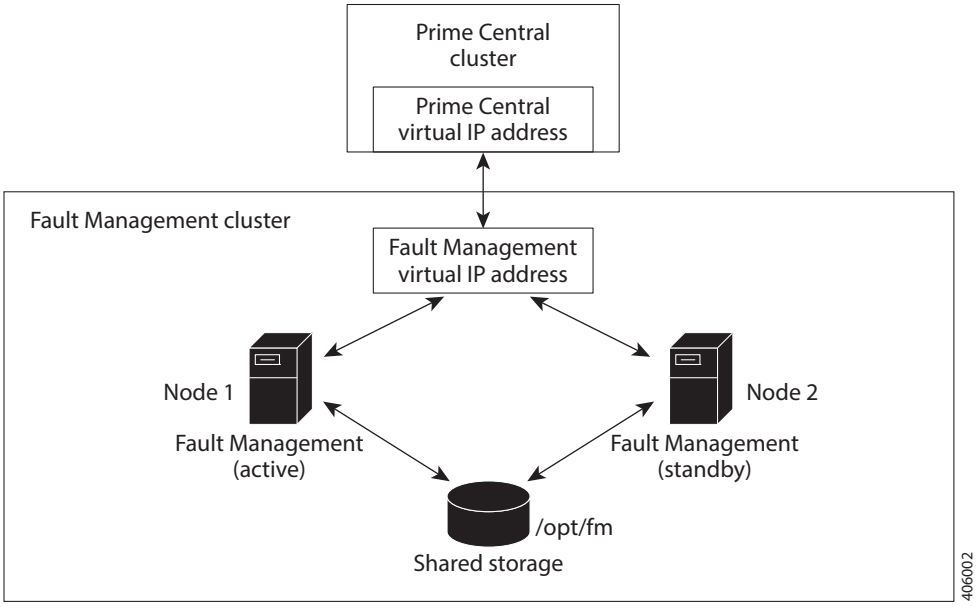
1. Install RHEL 6.5, 6.7, or 6.8 on both nodes.
2. Use multipath shared storage and install Prime Central on node 1.
3. Use multipath shared storage that contains the virtual machine image.

The examples provided use the following hostnames and IP addresses; yours will be different:

- Node 1—prime-fm-node1.cisco.com (192.168.1.150)
- Node 2—prime-fm-node2.cisco.com (192.168.1.160)
- Virtual IP address—prime-fm.cisco.com (192.168.1.170)
- Gateway—192.168.1.1
- DNS—192.168.1.2
- luci—prime-ha-luci.cisco.com (192.168.1.140)

Figure 2-5 shows an example of a Fault Management cluster in an HA configuration.

Figure 2-5 Fault Management Cluster in an HA Configuration



Before You Begin

- Ensure that both PC and FM are installed on different VMs.
- Verify that your system meets all the hardware and software requirements in "Installation Requirements" in the *Cisco Prime Central 2.0 Quick Start Guide*.
- If you changed the default installation folder (/opt/fm/primeusr/faultmgmt), make the equivalent changes in the following files (look for the section titled "Require manual definition" in each file):
  - /usr/local/bin/fm.sh

Fault Management HA Setup on RHEL 6.5 or RHEL 6.7 or 6.8

Step 1 Create two VMs with the configuration specified in the *Cisco Prime Central 2.0 Quick Start Guide*.

In this setup:

- Two VMs act as cluster nodes.
- Create two VMs—one with the virtual IP address (virtual hostname) and the other with the node2 IP address (node2 hostname). After the fault management installation is complete, you will then change the virtual IP address to node 1's IP address (node1 hostname) and then configure the cluster services.
- Shared network storage is required.

Step 2 Install RHEL 6.5 or 6.7 or 6.8 on the cluster nodes with the Desktop option selected.

Ensure that:

- The cluster nodes can ping each other via IP address and hostname.

- Both the active and standby cluster nodes have a 200 GB shared repository, which is located in the /dev/sdb directory.

Step 3 Disable the firewall on the VMs:

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

Step 4 On both cluster nodes, switch the network daemons:

```
# service NetworkManager stop
# chkconfig NetworkManager off
# chkconfig network on
```

Step 5 Disable Security-Enhanced Linux (SELinux) on the VMs:

```
# vi /etc/selinux/config
SELINUX=disabled
```

Step 6 Reboot the VMs.

Adding Clustering to the Installed Red Hat Server (Fault Management)

To install RHEL 6.5, complete the following steps in parallel on both nodes, except where noted:

Note Below steps are specific for RHEL 6.5 and this procedure is also supported for RHEL 6.7 and 6.8. You have to change the folder names and iso file names accordingly for RHEL 6.7 and 6.8.

Step 1 Create local directories named /rhel and /cdrom-6.5.

Step 2 Copy the .iso file that was used for the node installation to the /rhel directory.

Step 3 Mount the /rhel .iso file to /cdrom-6.5:

```
# cd /rhel
# mount -t iso9660 -o loop /rhel/rhel-server-6.5-x86_64-dvd.iso /cdrom-6.5
```

Note To permanently mount the drive, update the /etc/fstab file. See http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/4/html/Introduction_To_System_Administration/s2-storage-mount-fstab.html.

Step 4 Create a file named /etc/yum.repos.d/local.repo. Use UNIX format and be sure there are no spaces before lines.

Step 5 Save the newly created file in local.repo, as follows:

```
[local]
name=Red Hat Enterprise Linux $releasever - $basearch - Local
baseurl=file:///cdrom-6.5/Server
enabled=1
gpgcheck=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
```

```
[HighAvailability]
name=Red Hat Enterprise Linux $releasever - $basearch - HighAvailability
baseurl=file:///cdrom-6.5/HighAvailability
enabled=1
gpgcheck=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
[ResilientStorage]
name=Red Hat Enterprise Linux $releasever - $basearch - ResilientStorage
baseurl=file:///cdrom-6.5/ResilientStorage
enabled=1
gpgcheck=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
```

Step 6 Keep the nodes synchronized:

```
# echo server tick.redhat.com$'\n'restrict tick.redhat.com mask 255.255.255.255 nomodify
notrap noquery >> /etc/ntp.conf
# chkconfig ntpd on
# service ntpd start
```

Step 7 Edit the /etc/hosts file to add the node information; for example:

```
192.168.1.140     prime-ha-luci.cisco.com         prime-ha-luci
192.168.1.150     prime-fm-node1.cisco.com        prime-fm-node1
192.168.1.160     prime-fm-node2.cisco.com        prime-fm-node2
192.168.1.170     prime-fm.cisco.com              prime-fm
```

Step 8 Generate an SSH key for the root user:

```
# chmod 755 ~
# ssh-keygen -t rsa -N "" -b 2047 -f ~/.ssh/id_rsa
# chmod 600 ~/.ssh/id_rsa
```

Step 9 (On the first node only) Share the node's public key with the other node so that dynamically creating a secure shell between the nodes does not prompt for a password:

```
# rsync -av ~/.ssh/id_rsa.pub ~/.ssh/authorized_keys
# ssh root@prime-fm-node2.cisco.com "cat ~/.ssh/id_rsa.pub" >> ~/.ssh/authorized_keys
# rsync -av ~/.ssh/authorized_keys root@prime-fm-node2.cisco.com:/root/.ssh/
```

Step 10 Verify that the .ssh directory has 700 permission and the .ssh/id_rsa file has 600 permission:

```
# chmod 700 ~/.ssh
# chmod 600 ~/.ssh/id_rsa
```

Step 11 Verify that your SSH is working without an authentication or password prompt:

Caution The Fault Management service will not start if SSH prompts for authentication or a password. Be sure to complete all of the following substeps.

a. On node fm-ha-node1.cisco.com, enter:

```
# ssh root@prime-fm-node2.cisco.com
# exit
# ssh root@prime-fm-node2
# exit
# ssh root@192.168.1.150
# exit
```

b. On node fm-ha-node2.cisco.com, enter:

```
# ssh root@prime-fm.cisco.com
# exit
```

```
# ssh root@prime-fm
# exit
# ssh root@192.168.1.160
# exit
```

c. If you are prompted for a password, check the permissions of all folders and files that you modified in the preceding steps.

d. If you are prompted to continue connecting, enter **yes**. (The prompt should appear only the first time you use SSH to connect to the node.)

Configuring Multipath

To configure multipath, complete the following steps in parallel on both nodes, except where noted:

Note The examples provided use device mapping names such as mpatha and mpathap1; yours may be different.

Step 1 Install multipath:

```
# yum install device-mapper-multipath
```

Step 2 Set up multipath. Execute the below commands in sequence:

```
# mpathconf --enable --user_friendly_names y
# modprobe dm-multipath
# service multipathd start
# chkconfig multipathd on
# vi /etc/multipath.conf
        -- Comment out 'blacklist' section
        -- For example:
        -- #blacklist {
        # devnode "*"
        #}
        -- Append below lines at the end of file
            devices {
            device {
                vendor "DGC"
                product ".*"
                product_blacklist "LUNZ"
                path_checker "directio"
                path_grouping_policy "group_by_prio"
                path_selector "round-robin 0"
                failback "immediate"
                }
             }
# service multipathd restart
# multipath -v2
# multipath -ll
# service multipathd reload
# modprobe dm-multipath
# service multipath restart
```

Step 3 Check for available disks. The names of the multipath disks must be identical on both nodes:

```
cd /dev/mapper/
ls -la
total 0
drwxr-xr-x  2 root root    140 Jan  8 15:18 .
```

```
drwxr-xr-x 17 root root   3760 Jan  8 15:18 ..
crw-rw----  1 root root 10, 58 Jan  8 12:44 control
lrwxrwxrwx  1 root root      7 Jan  8 15:18 mpatha -> ../dm-3
lrwxrwxrwx  1 root root      7 Jan  8 12:44 vg_primefmnode2-lv_home -> ../dm-2
lrwxrwxrwx  1 root root      7 Jan  8 12:44 vg_primefmnode2-lv_root -> ../dm-0
lrwxrwxrwx  1 root root      7 Jan  8 12:44 vg_primefmnode2-lv_swap -> ../dm-1
```

In the output, note *mpatha*, which is the multipath virtual device or disk that you will use later as shared storage.

Adding Shared Partitions

To add shared partitions, complete the following steps in parallel on both nodes, except where noted:

Note The examples provided use device mapping names such as mpatha and mpathap1; yours may be different.

Step 1 (On the first node only) Create a 100-GB, shared partition:

```
# cd /dev/mapper
# fdisk mpatha
Command (m for help): p
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-19581, default 1): <Enter>
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-19581, default 19581): <Enter>
Command (m for help): w
```

Step 2 Reboot both nodes.

Step 3 Check for new partitions:

```
# cd /dev/mapper/
# ls -la
total 0
drwxr-xr-x  2 root root    160 Jan  8 15:27 .
drwxr-xr-x 17 root root   3800 Jan  8 15:27 ..
crw-rw----  1 root root 10, 58 Jan  8 15:27 control
lrwxrwxrwx  1 root root      7 Jan  8 15:27 mpatha -> ../dm-2
lrwxrwxrwx  1 root root      7 Jan  8 15:27 mpathap1 -> ../dm-3
lrwxrwxrwx  1 root root      7 Jan  8 15:27 vg_primefmnode2-lv_home -> ../dm-4
lrwxrwxrwx  1 root root      7 Jan  8 15:27 vg_primefmnode2-lv_root -> ../dm-0
lrwxrwxrwx  1 root root      7 Jan  8 15:27 vg_primefmnode2-lv_swap -> ../dm-1
```

Step 4 (On the first node only) Format the new shared partition:

```
# mkfs.ext4 /dev/mapper/mpathap1
```

Step 5 Create target locations on both nodes:

```
# mkdir -p /opt/fm
```

Step 6 Verify that both nodes can mount and unmount the shared storage:

a. On the first node, mount the shared storage and save a file that contains only the value *1* to the shared storage. The test.txt file should exist in the list of contents of /opt/fm:

```
# mount /dev/mapper/mpathap1 /opt/fm
# vi /opt/fm/test.txt
1
:wq
# ls -la /opt/fm
# umount /opt/fm
```

b. On the second node, mount the shared storage and verify that the test.txt file exists and contains the value *1*:

```
# mount /dev/mapper/mpathap1 /opt/fm
# vi /opt/fm/test.txt
:q
# umount /opt/fm
```

If you cannot mount or unmount the shared storage, or if the test.txt file does not exist when you mount it to the second node, your multipath is not set up correctly.

Installing Prime Central Fault Management

Step 1 Verify that both fault management VMs can ping each other via the hostname and IP address. Also verify that the both VMs can ping the PC cluster nodes and luci node via the hostname and IP address.

Step 2 Verify that both VMs can ssh to each other without using the configured password.

Step 3 Insert the Cisco Prime Central 2.0.0 USB, navigate to the High Availability/RHCS VMWare ESXi Local HA/Fault Management folder, and locate the primefm_v2.0.0_ha_node.tar.gz file.

Step 4 Use SSH to connect to the virtual IP node.

Step 5 Copy the primefm_v2.0.0_ha_node.tar.gz file to the virtual IP node.

Step 6 Back up the /root/ha-stuff/fm and /usr/local/bin directories.

Step 7 Mount the partition to the /opt/fm directory on the virtual IP node:

```
# mount /dev/mapper/mpathcp1 /opt/fm
```

Step 8 Distribute the file on both nodes:

```
# tar -zxf primefm_v2.0.0_ha_node.tar.gz -C / --owner root --no-same-owner
# cd /root/ha-stuff/fm
```

Step 9 Navigate to the top-level Fault Management folder and copy the FM2.0.0Build.tar.gz file to the /root/ha-stuff/fm directory:

```
# tar -zxf FM2.0.0Build.tar.gz
# chmod 755 /root/ha-stuff/fm/*
# chmod -R 755 /root/ha-stuff/fm/Disk*
# chmod 755 /usr/local/bin/*
```

Step 10 Specify all necessary properties in the fm_install.properties file.

Step 11 Change HOSTNAME in /etc/sysconfig/network to match the value of PRIMEFM_SERVER_HOSTNAME in fm_install.properties file.

Note The hostname must be a Fully Qualified Distinguished Name (FQDN).
`For example: prime-fm-service.cisco.com`

Step 12 Change active hostname to the host name used in step 11 (the virtual FQDN hostname) using the command:

```
hostname
For example: run: hostname prime-fm-service.cisco.com
```

Step 13 Install Fault Management:

./InstallFaultManagement.sh *<install-location> <FM-virtual-IP-address> <root-password> <second-node-IP-address>*

```
For example:
./InstallFaultManagement.sh /opt/fm/primeusr/faultmgmt 192.168.1.170 cisco123
192.168.1.160
```

Step 14 After the installation is complete, make sure that FM can be started properly.

```
/usr/local/bin/fm.sh status
```

Step 15 Freeze the Prime Central cluster and then restart the integration layer so that Prime Central displays Fault Management's status as UP.

Step 16 Stop Fault Management:

```
# /usr/local/bin/fm.sh stop
```

Step 17 Change the IP address from virtual IP (hostname) to node 1's IP address (hostname).
Revert to use the original hostname, if there are any changes in /etc/sysconfig/network file (in Step 11).

Step 18 Reboot the VM.

Setting Up the Fault Management Cluster Service

On both cluster nodes:

Step 1 Install the ricci RPMs:

```
# yum -y install ricci
```

Step 2 Start the ricci daemon and configure it to start on boot:

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

Step 3 Set the ricci user password:

```
# passwd ricci
```

You will need to enter this password later.

At this point, both cluster nodes should be running the ricci servers and be ready to be managed by the cluster web user interface (luci).

Configuring the Cluster Web User Interface (luci)

Complete the following procedure to configure luci on the third VM for use with Prime Central Fault Management.

Step 1 Launch the URL listed in the last line of the system output (https://prime-ha-luci.cisco.com:8084, in this example) and log in as the root user when prompted.

Step 2 Verify that the RHEL DVD is mounted on both cluster nodes.

Step 3 Select **Manage Clusters > Create** and then specify a name for the cluster (for example, PrimeFMCluster).

Step 4 Enter the name (fully-qualified domain name or name specified in the /etc/hosts directory) and password (the password for the user *ricci*) for the first cluster node.

Step 5 Click **Add Another Node** and enter the same information described in Step 4 for the second cluster node.

Step 6 Do the following, and then click **Create Cluster**:

- **Use the Same Password for All Nodes**: Leave this check box unchecked.
- **Download Packages**: Select this radio button.
- **Reboot Nodes Before Joining Cluster**: Select this check box.
- **Enable Shared Storage Support**: Leave this unchecked.

Assuming that the nodes can be contacted, luci will set up each cluster node and add each node to the cluster.

After each node has been set up, the High Availability Management screen opens.

Step 7 Create a failover domain:

a. Select the **Failover Domains** tab and then click **Add**.

b. Do the following:

- **Name**: Enter a name for the failover domain.
- **Prioritized**: Select this check box.
- **Restricted**: Select this check box.
- **No Failback**: Select this check box.
- **Member**: Select the check box for each node.
- **Priority**. In the Priority column, enter *1* for node1 and *2* for node2.

c. Click **Create**.

Step 8 Configure fence devices appropriate for the hardware you have, adding a fence device and instance for each node.

The settings you need to configure depend on your particular hardware and software configuration. Refer to the Cluster Administration Guide for help with configuring fence devices.

Complete the following steps to configure VMware fencing:

a. Retrieve the UUID for prime-ha-node1.cisco.com:

```
fence_vmware_soap -v -z -a VCenter-IP-Address -l VCenter-Username -p VCenter-Password
--action list  | grep VM-Name
```

This returns: *VM-Name*,UUID

Keep in mind that *VM-Name* is the name given to the VM during VM creation, not its hostname.

b. Repeat the previous step for prime-ha-node2.cisco.com.

c. In luci, select the **Fence Devices** tab and then click **Add**.

d. Do the following:

- **Fence type**: Select VMWare Fencing (SOAP Interface).
- **Name**: Enter a descriptive name for the fence device.
- **IP Address or Hostname**: Enter the appropriate VCenter IP address or hostname.
- **IP Port** (optional): Leave blank.
- **Login**: Enter the VCenter username.
- **Password**: Enter the VCenter password.

Leave the rest of the fields as is or blank.

e. Click **Submit**.

f. Select the **Nodes** tab and then select your first node by clicking its name.

g. At the bottom of the node's window, click **Add Fence Method**.

h. Enter a name and then click **Submit**. Now select **Add Fence Instance** that appears inside the method box and fill it out as described below:

i. Select the fencing device you configured in Step 8d and do the following:

- **VM Name**: Leave blank.
- **VM UUID**: Enter the UUID you retrieved in Step 8a (for example, 4209838d-1e91-104c-9f5c-1181ad87ddd8).
- **Use SSL**: Select this check box. Fencing will not work unless you do so.

j. Repeat Steps 8f through 8i for your second node.

Note We recommend that you check the bottom of this window for each node and verify that all of the required daemons are running before you test fencing.

Step 9 To configure KVM fencing, complete the following steps:

a. Retrieve the UUID for prime-pc-node1.cisco.com by executing the below command on the KVM host:

```
virsh dominfo prime-pc-node1.cisco.com
```

b. Repeat the previous step for prime-pc-node2.cisco.com.

c. In luci, select the **Fence Devices** tab, and then click **Add**.

d. From the **Fence Type** drop-down list, choose fence_virt(Multicast Mode).

e. In the **Name** field, enter a descriptive name.

Note Leave the other fields as is or blank.

f. Click **Submit**.

g. Select the **Nodes** tab, and then select your primary node by clicking its name.

h. Click **Add Fence** method that is available at the bottom of the node window.

i. Enter a name, and then click **Submit**. Select the Add Fence Instance that appears inside the method box, and then enter the details as specified below:

– Select the fencing device that you configured in step 9d, and then in the **Domain name** field, enter the domain name. For example, you can enter 4209838d-1e91-104c-9f5c-1181ad87ddd8.

Note Leave the **Delay** field as blank.

j. Click **Submit**.

Step 10 Set up the network configuration:

a. Select the **Configure** tab.

b. Select the **Network** tab.

c. Select the **UDP Unicast (UDPU)** radio button and then click **Apply**.

Setting Up Shared Resources

You will next create the cluster service (named PrimeService in this example). To begin, open the High Availability Management page (luci) in a browser. From there, you will:

- Identify the virtual IP address for this cluster
- Identify the PrimeService service and add the virtual IP address to this service

At several points, you will run commands from the cluster nodes (ricci) to verify that the configuration you set in luci is working.

Identifying the Cluster Service's Virtual IP Address

From luci, do the following to identify the cluster service's virtual IP address:

Step 1 Select the cluster name (for example, **PrimeCluster**).

Step 2 Select the **Resources** tab and then click **Add**.

Step 3 Select **IP Address**.

Step 4 Enter the following information:

- **IP Address**: Specify a valid IP address, which will be used to access the service from a web browser.
- **Monitor Link**: Select this check box.

Step 5 Click **Submit**.

Creating the Prime Fault Management Service

From luci, with the cluster still selected, add a new service and associate it with the IP address.

Step 1 Select the **Service Groups** tab and then click **Add**.

Step 2 Enter the following information for the new service group:

- **Service name**: Name of the service group (for example, **PrimeFMService**)
- **Automatically start this service**: Select this check box.
- **Failover Domain**: Select the PCFailOver you created while installing luci.
- **Recovery Policy**: Select **Relocate**.

Step 3 Click **Add Resource**.

Step 4 Select the IP address you added earlier and then click **Submit**.

Step 5 Wait 5 minutes and then reboot both cluster nodes.

After the VMs come back online, the **PrimeFMService** service should up and running.

Adding Mount Points and Setting Up File Systems

In the following procedure, you will create mount points on the two nodes. After the mount points have been set up, you can then configure the resources necessary to mount file systems to the proper locations.

Step 1 Create a file system resource:

a. From luci, select the **Resources** tab and then click **Add**.

b. From the Add Resource to Cluster window, select **Filesystem** from the dropdown list and then do the following:

- **Name**: Enter the name of the file system resource.
- **Filesystem** type: Select the ext4 option.
- **Mount point**: Enter the mount point path—/opt/fm
- Device, FS label, or UUID: Enter the appropriate path—in this example, /dev/mapper/mpath2p1
- Leave the **Mount options** and **Filesystem ID (optional)** field blank.
- Select the **Force Unmount** check box.
- Select the **Reboot host node if unmount fails** check box.

c. Click **Submit**.

Step 2 Add the file system resource to the **PrimeService** service group:

a. From luci, select the **Service Groups** tab.

b. In the Name column, select the **PrimeFMService** service group name.

c. At the bottom of the page, click **Add Resource**.

d. Click **Select a Resource Type**.

The filesystem resources you added should appear in the list.

e. From the list, click the filesystem entries you added in the previous step.

f. Click **Submit**.

Step 3 Log in to the node currently running the PrimeFMService service.

Step 4 Verify that the logical volume ext4 filesystems are mounted with read/write privileges on the node running the PCService service:

```
# cat /etc/mtab
```

Step 5 **Verify** that the logical volume ext4 filesystems are not mounted on the node which is not running the PCService service:

```
# cat /etc/mtab
```

Adding Scripts to the Prime Central Cluster Service

Step 1 Add the fm.sh script resource:

a. From luci, select the **Resources** tab and then click **Add**.

b. From the Add Resource to Cluster window, select **Script** from the dropdown list and then enter the following information:

- **Name**: Name of the script
- **Full Path to Script File:** Directory in which the script will reside—/usr/local/bin/fm.sh

Step 2 Add script resources to the **PrimeService** service group:

a. From luci, select the **Service Groups** tab.

b. In the Name column, select the **PrimeFMService** service group name.

c. At the bottom of the page, click **Add Resource**.

d. Click **Select a Resource Type**.

The script resource you added should appear in the list.

e. Click the script entries from the list you added in the previous step.

f. Click **Submit**.

Verifying the PrimeService Service Can Be Reached

From each of the cluster nodes, check the status of the PrimeService services you created. Then, from any machine on the network, verify that you can reach the cluster from the IP address you assigned.

Step 1 From a shell on each of the cluster nodes, verify that the service group is running by entering the following commands (in bold):

```
# clustat

The output is similar to the following:
Cluster Status for PrimeFMCluster @ Thu Jan  9 14:15:02 2014
Member Status: Quorate

Member Name                 ID      Status
------ ----                 ----    ------
prime-fm-node1.cisco.com    1       Online, Local, rgmanager
prime-fm-node2.cisco.com    2       Online, rgmanager

Service Name                Owner (Last)                State
------- ----                ----- ------                -----
service:PrimeFMService      prime-fm-node1.cisco.com    started
```

Step 2 From a shell anywhere that can reach the cluster over the network, verify that you can ping the IP address associated with the IP address resource you created previously (see Creating the Prime Service, page 2-21):

```
ping <Virtual-IP>
```

Note To end the ping, enter **Ctrl + C**.

Step 3 Relocate the service:

```
# clusvcadm -r PrimeFMService
```

Step 4 Verify that the service has been relocated successfully to the other node.

Checking the Cluster Services

Step 1 Review the cluster log file in /var/log/messages.

Step 2 After the Fault Management service is running in an HA cluster, you cannot restart its components (such as Netcool/Impact, OMNIbus, and Tivoli Common Reporting [TCR]) without first freezing the cluster. After you restart the component, you can unfreeze the cluster.

To restart a Fault Management component:

a. On the active Fault Management node, enter:

```
# clusvcadm -Z Fault-Management-service-name
```

b. Use SSH to connect to the Fault Management virtual machine and enter:

```
# /usr/local/bin/fm.sh stop
# /usr/local/bin/fm.sh start
```

c. Use SSH to connect to the active Fault Management node and enter:

```
# clusvcadm -U Fault-Management-service-name
```

Troubleshooting

The following troubleshooting steps help to solve common problems in HA configuration.

Problem The HA installation fails.

Solution Check the log files to locate the problem and take the appropriate action. Log files contain detailed information about request processing and exceptions and are your best diagnostic tool for troubleshooting. See “Troubleshooting the Installation” in the *Cisco Prime Central 2.0 Quick Start Guide*.

Problem Prime Central does not start in a clustered setup.

Solution Check the /var/log/messages files for failure to either mount the shared storage or add the virtual IP address. If the shared storage failed to mount, shut down the cluster and verify that you can manually add the shared storage to a node. (Be sure to unmount it after your test.)

If the virtual IP address was not added, verify that it is in the same subnet as the nodes and is not in use by any other computer in the network.

If you find that /usr/local/bin/pc.sh start failed, check /var/halogs/pc.log, which will tell you if the database or other Prime Central components failed to start.Then, to determine which component failed to start:

1. Stop the luci, ricci, rgmanager, and cman services on both nodes to shut down the cluster.
2. On the node where you originally installed Prime Central:
   a. Mount the shared storage.
   b. Add the virtual IP address.
   c. Verify that all services have stopped:

      ```
      /usr/local/bin/pc.sh stop
      ```

   d. Enter:

      ```
      # su - primeusr
      # emdbctl -start
      # itgctl start
      # portalctl start
      ```

   e. Check the output from each of the preceding commands to locate the problem.

Problem You receive the error "<err> 'fsck -p /dev/mapper/mpath2p1' failed, error=4; check /tmp/fs-vmpcfs.fsck.log.mq4986 for errors."

Solution Enter the following command and reboot when it is finished running:

```
# fsck -f /dev/mapper/mpath2p1
```

Problem You receive the error "Timeout exceeded while waiting for '/images/fm_status.sh'" in /var/log/messages.

Solution Verify that you can use SSH to connect to each node and virtual machine without an authentication or password prompt. If SSH prompts for authentication or a password, the Prime Central and Fault Management services cannot start.

Problem Your environment uses the wrong fencing device.

Solution The examples in this guide use fence_vmware for VMware Hypervisor and fence_virt for KVM Hypervisor. For information about which fencing device to use in your environment, see the *Red Hat Enterprise Linux 6 Cluster Administration: Configuring and Managing the High Availability Add-On*.

Problem The cman and rgmanager services do not start.

Solution Check the log files in /var/log/messages and /var/log/cluster. Use the following tool to verify that your multicast address is correct: http://juliandyke.wordpress.com/2010/12/03/testing-multicasting-for-oracle-11-2-0-2-grid-infrastructure/.

Problem Cannot stop the cluster.

Solution Use luci or the command line to shut down your cluster:

- luci—Select the cluster; then, from the drop-down list, choose **Stop this cluster**.
- Command line—Alternating between the two nodes, shut down the services in the reverse order in which you started them. For example, enter the **stop** command for rgmanager on node1; then, enter it on node2. Enter the **stop** command for cman on node1; then, enter it on node2.

  ```
  # service luci stop
  # service ricci stop
  # service rgmanager stop
  # service cman stop
  ```

Problem When trying to unmount the shared storage, a "device is busy" message is returned.

Solution Verify that all cluster services have stopped and that you have closed all terminal sessions that are accessing the shared storage location. To determine which user is accessing the shared storage, enter:

```
# fuser -m -v shared-storage
```

For example:

```
# fuser -m -v /opt/pc
```

Problem You do not know if the node can support virtualization.

Solution Enter:

```
# egrep '^flags.*(vmx|svm)' /proc/cpuinfo
```

If the command returns no output, the node does not support virtualization.

If the command output contains vmx or svm flags, the node supports virtualization. For example:

```
flags           : fpu vme de pse tsc msr pae mce cx8 apic mtrr pge mca cmov pat pse36
clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm constant_tsc
arch_perfmon pebs bts rep_good xtopology nonstop_tsc aperfmperf pni pclmulqdq dtes64
monitor ds_cpl vmx smx est tm2 ssse3 cx16 xtpr pdcm dca sse4_1 sse4_2 popcnt aes lahf_lm
ida arat dts tpr_shadow vnmi flexpriority ept vpid
```

Problem Cannot test the cluster.conf file.

Solution Use **rg_test** commands. For example:

- To display the resource rules that rg_test understands, enter:

  rg_test rules

- To test a configuration, enter:

  rg_test test /etc/cluster/cluster.conf

- To display the start ordering of a service, enter:

  rg_test noop /etc/cluster/cluster.conf start service *service-name*

- To display the stop ordering of a service, enter:

  rg_test noop /etc/cluster/cluster.conf stop service *service-name*

Problem When you reboot one or both nodes, the node is fenced before it can join the cluster.

Solution To start up, the node might require an additional fencing delay. Edit your cluster.conf file by increasing the value of the post_join_delay attribute:

```
<fence_daemon clean_start="0" post_fail_delay="0" post_join_delay="30"/>
```

Problem After you relocate the Prime Central service, the integration layer is shown in the Prime Central Suite Monitoring portlet > Applications tab, but its state is Down.

Solution On servers where the hardware requirements are at or below the minimum for Prime Central high availability, the integration layer requires more time to start up. Do the following:

1. On the active node where Prime Central is running, locate the /opt/pc/primecentral/esb/etc/com.cisco.prime.esb.jms.cfg file.
2. Edit the file by increasing the waitForStart attribute for the jmsvm.internalBrokerURL property. (If the line is commented, uncomment it.)

   The default waitForStart value is 10,000 milliseconds; increase it depending on the slowness of your server. For example, to increase the waitForStart value to 30 seconds, enter:

   ```
   jmsvm.internalBrokerURL=vm://internalBroker?broker.persistent=false&jms.prefetchPolicy
   .queuePrefetch=1&create=false&waitForStart=30000
   ```

Problem The Prime Central portal does not look correct.

Solution The cluster manager might have relocated the server. Clear your browser cache and refresh your screen; then, log back in to the Prime Central portal.

Problem You need to restart a Prime Central or Fault Management component in an HA environment.

Solution Prime Central contains components such as the portal, integration layer, and database. Fault Management contains components such as Netcool/Impact, OMNIbus, and TCR. If you need to perform maintenance on a specific component, you must freeze the HA cluster before you can stop the component. After you restart the component, you can unfreeze the cluster.

- To freeze the cluster, enter:

  clusvcadm -Z *service-name*

- To unfreeze the cluster, enter:

  clusvcadm -U *service-name*

Problem After adding multipath, you cannot see the multipath names when listing the /dev/mapper directory.

Solution Do the following:

1. Enter:

   ```
   # vi /etc/multipath.conf
   ```

2. Change the find_multipaths value to **no**.
3. Enter:

   ```
   # wq
   # service multipathd reload
   ```

   You should now see the multipath names.

Problem You receive the following error while mounting the storage:

```
[root@prime-central-linux4 ~]# mount /dev/mapper/mpathbp2 /images/
mount: wrong fs type, bad option, bad superblock on /dev/mapper/mpathbp2,
       missing codepage or helper program, or other error
       In some cases useful info is found in syslog - try
       dmesg | tail or so
```

Solution Enter:

```
fsck.ext3 <storage-path>
fsck.ext3 /dev/mapper/mpathbp2
```

Problem The **fmctl status** command shows that Fault Management started in KVM, but hangs in "starting" status in the cluster.

Solution Check the SSH password-less connection between the two nodes and KVM.

Problem During HA installation in a VMware and KVM environment, you may see the following warning about the 32-bit gtk2 RPM package on the console:

```
warning: %post(gtk2-2.18.9-12.el6.i686) scriptlet failed, exit status 127
```

Solution This warning is harmless and does not indicate a problem which will affect either the installation or operation of Fault Management.

Problem `-import: unknown option` is displayed during installation.

Solution This error message does not indicate an actual problem and can be ignored.

Problem Error messages are displayed during the installation of the Fault Management component.

Solution The error messages do not indicate actual problems and can be ignored.

Problem Unsuccessful installation.

Solution When an installation fails, perform the below Cleanup procedure, along with the uninstall script:

1. Run uninstall script and remove directories manually.
2. Enter:

```
cd /opt/pc
```

3. Enter:

```
rm -rf primecentral oracle
```

Upgrading to Prime Central 2.0.0 in a Local Redundancy HA Configuration

Note As part of Prime Central 2.0.0 release, Direct upgrade from 1.4.1 to 2.0.0, 1.5 to 2.0.0 and 1.5.3 to 2.0.0 are supported. For every upgrade follow the same steps mentioned below.
If you are upgrading directly from 1.4.1 to 2.0.0, as part of Prime Central 2.0.0 release, data models have been changed from PSI (Prime Service Inventory) to SSI (Sub System Inventory) internally. Before performing the upgrade of Prime Central application from Release 1.4.1 to Release 2.0.0, it is therefore recommended to check whether there is any inconsistent PSI data. To detect any potential inconsistencies with PSI, contact the Cisco support.

Step 1 Freeze the cluster.

Step 2 Insert the Cisco Prime Central 2.0.0 USB, navigate to the High Availability/RHCS Bare Metal Local HA/Prime Central folder, and locate the primecentral_v2.0.0_ha_vm.tar.gz file.

Step 3 Use SSH to connect to the first node.

Step 4 Copy the primecentral_v2.0.0_ha_vm.tar.gz file to the first node.

Step 5 Back up the following directories on both nodes:

- /root/ha-stuff/pc
- /usr/local/bin

Step 6 Distribute the file:

```
# tar -zxf primecentral_v2.0.0_ha_vm.tar.gz -C / --owner root --no-same-owner
```

Step 7 Navigate to the Base Application folder and copy primecentral_v2.0.0.bin and all available .zip files to the **/root/ha-stuff/pc** directory:

```
# chmod 755 /usr/local/bin/*
# chmod 755 /root/ha-stuff/pc/*
```

Step 8 Perform the following on the first node:

a. Mount the shared partitions:

```
# mount /dev/mapper/mpathap1 /opt/pc
```

b. Add a virtual IP cluster service address for the Prime Central service:

```
# ip addr add 192.168.1.130 dev eth0
```

c. Update the install.properties file and verify if all required properties have values. Review the comments at the top of the install.properties file for details.

Note To install Prime Central silently, you must edit the /root/ha-stuff/pc/install.properties file. See 'Sample install.properties Files' in the *Cisco Prime Central 2.0.0 Quick Start Guide*.

d. Install Prime Central:

```
# cd /root/ha-stuff/pc
# ./PrimeCentral.sh 192.168.1.130 node's-root-password second-node-IP-address
```

Note Run the PrimeCentral.sh script by adding the preceding command-line parameters. If you do not add the command-line parameters, you are prompted for the required data.

e. In another terminal window, check the upgrade process:

```
# tail -f /tmp/primecentral_install.log
```

f. After the upgrade succeeds, start Prime Central:

```
# /usr/local/bin/pc.sh start
```

g. Verify if Prime Central is running correctly. If so, stop it:

```
# /usr/local/bin/pc.sh stop
```

h. Remove virtual IP addresses:

```
# ip addr del 192.168.1.130 dev eth0
```

i. Unmount shared partitions:

```
# umount /opt/pc
```

j. Unfreeze the cluster.

When upgrade completes, log files will be available in ~/upgrade/1.5.3.0-2.0.0.0/upgrade.log.

Upgrading to Prime Central Fault Management 2.0 in a Local Redundancy HA Configuration

Note As part of Prime Central Fault Management 2.0.0 release, Direct upgrade from 1.4.1 to 2.0.0, 1.5 to 2.0.0 and 1.5.3 to 2.0.0 are supported. For every upgrade follow the same steps mentioned below.

Step 1 Freeze the cluster.

Step 2 Insert the Cisco Prime Central 2.0 USB.

Step 3 Navigate to the High Availability/RHCS VMWare ESXi Local HA/Fault Management folder, and locate the primefm_v2.0_ha_node.tar.gz file.

Step 4 Use SSH to connect to the virtual IP node.

Step 5 Copy the primefm_v2.0_ha_node.tar.gz file to the virtual IP node.

Step 6 Back up the /root/ha-stuff/fm and /usr/local/bin directories.

Step 7 Mount the partition to the /opt/fm directory on the virtual IP node:

```
# mount /dev/mapper/mpathcp1 /opt/fm
```

Step 8 Distribute the file:

```
# tar -zxf primefm_v2.0_ha_node.tar.gz -C / --owner root --no-same-owner
# cd /root/ha-stuff/fm
```

Step 9 Navigate to the top-level Fault Management folder and copy the FM2.0Build.tar.gz file to the /root/ha-stuff/fm directory:

```
# tar -zxf FM2.0Build.tar.gz
# chmod 755 /root/ha-stuff/fm/*
# chmod -R 755 /root/ha-stuff/fm/Disk*
# chmod 755 /usr/local/bin/*
```

Step 10 Specify all of the necessary properties in the fm_install.properties file.
Change HOSTNAME in /etc/sysconfig/network to match PRIMEFM_SERVER_HOSTNAME. For example:

```
prime-fm-service.cisco.com
```

Step 11 Reboot the VM.

Step 12 Mount the shared partitions:

```
# mount /dev/mapper/mpathap1 /opt/fm
```

Step 13 Start Fault Management:

```
# /usr/local/bin/fm.sh start
```

Step 14 Upgrade Fault Management:

```
# ./InstallFaultManagement.sh <install-location> <FM-virtual-IP-address> <root-password>
<second-node-IP-address>
```

For example:

```
./InstallFaultManagement.sh /opt/fm/primeusr/faultmgmt 192.168.1.170 cisco123
192.168.1.160
```

Step 15 After the upgrade is complete, make sure that FM can be started properly.

```
# /usr/local/bin/fm.sh status
```

Step 16 Freeze the Prime Central cluster and then restart the integration layer so that Prime Central displays Fault Management's status as UP.

Step 17 Stop and start Fault Management:

```
# /usr/local/bin/fm.sh stop
# /usr/local/bin/fm.sh start
```

Step 18 Change the IP address from virtual IP (hostname) to node 1's IP address (hostname). Reverse the hostname, if there are any changes in /etc/sysconfig/network file (in Step 10).

Step 19 Reboot the VM.

Upgrading RHEL Operating System

The Operating System (OS) upgrade procedure supports Prime Central 1.5.1 customers to perform upgrade of Operating System from RHEL 5.8 to 6.5 and inline upgrade from RHEL 6.5 to 6.7. For more information, refer the Cisco Prime Central RHEL Operating System Upgrade guide.

Rollback Procedure for Prime Central

After upgrading to Prime Central 2.0.0, you may find the need to revert to the previous version. To do so, complete the following procedure.

Note By default the primeusr home folder is /opt/pc/primecentral. If your primeusr home folder is different, specify that folder instead.

Step 1 Login as the root user and make sure Cluster is frozen:

```
# clustat
```

Step 2 Confirm that the /opt/pc/primecentral_2.0.0_backup folder was created during upgrade from Prime Central 1.5.3 to Prime Central 2.0.0.

a. If the folder was created, proceed to Step 2.

b. If the folder was not created, this indicates that there was a disk space issue and the upgrade was not started. You can stop here.

Step 3 Change ownership of the /opt/primecentral_1.5.3.0_backup folder:

```
chown -R primeusr:ncoadmin /opt/pc/primecentral_1.5.3.0_backup
```

Step 4 Stop all Prime Central processes:

```
su - primeusr
itgctl stop
portalctl stop
exit
```

For distributed server environment (portal and IL are in different servers):

- Execute above portalctl stop command on Portal server as primeusr.
- Execute above itgctl stop command on IL server as primeusr.

Step 5 Move the primecentral folder to the tmp folder:

```
mv /opt/pc/primecentral/ /tmp/primecentral
```

Step 6 Rename the /opt/primecentral_1.5.3.0_backup folder:

```
mv /opt/pc/primecentral_1.5.3.0_backup/ /opt/pc/primecentral
```

Step 7 Restore the /var/.com.zerog.registry.xml file with the backup file: **/var/.com.zerog.registry.xml_backup_for_2.0.0** that was created during Prime Central upgrade process. Execute the below commands:

```
mv /var/.com.zerog.registry.xml /var/.com.zerog.registry.xml_backup_for_2.0
mv /var/.com.zerog.registry.xml_backup_for_1.5.3 /var/.com.zerog.registry.xml
```

Step 8 To find the exact time to restore database, back to the timestamp before upgrade.

Login to the database as sysdba and execute the below query to get the RMAN backup history:

```
SELECT status, to_char(START_TIME,'mm/dd/yy hh24:mi') start_time,
to_char(END_TIME,'mm/dd/yy hh24:mi') end_time
FROM V$RMAN_BACKUP_JOB_DETAILS
ORDER BY end_time DESC;
```

Step 9 Change the permissions of ssh key files as below:

For primeusr user:

```
su - primeusr
cd ~/local/prime_secured
chmod 600 id_dsa id_dsa.pub authorized_keys
```

For oracle user:

```
su - [oracle_user]
cd ~/prime_secured
chmod 600 id_dsa id_dsa.pub authorized_keys
```

Step 10 Restore database (from the backup taken before upgrading to Prime Central 2.0.0)

If it is embedded database:

```
su - primeusr
emdbctl --restore
```

Note Restore time should be the time of the full backup taken just before the start of upgrade and should be taken from the result of the query executed in Step 7.

Database restore log location: /opt/pc/oracle/prime_logs/restore***_***.log

If it is External database:

- Login/ssh to external database as root
- Restore full database to the time of backup taken (manually) just before the start of PC upgrade. Use oracle commands:

  Database restore log location: /opt/pc/oracle/prime_logs/restore***_***.log.

Step 11 Start all Prime Central processes:

```
su - primeusr
itgctl start
portalctl start
exit
```

For distributed server environment (portal and IL are in different servers):

- Execute above portalctl start command on Portal server as primeusr.
- Execute above itgctl start command on IL server as primeusr.

Step 12 Execute this step only if it is distributed server environment (portal and IL are in different servers):

Repeat the above steps 1 to 6 in IL server.

Rollback Procedure for Fault Management

After upgrading to Prime Central Fault Management 2.0.0, you may find the need to revert to the previous version. To do so, complete the following procedure.

Note By default the primeusr home folder is /opt/fm/primeusr. If your primeusr home folder is different, specify that folder

Step 1 Login as the root user and make sure Cluster is frozen:

```
# clustat
```

Step 2 Confirm that the faultmgmt_ 1.5.3.0_backup folder was created.

- If the folder was created, proceed to Step 2.
- If the folder was not created, this indicates that there was a disk space issue and the upgrade was not started. You can stop here.

Step 3 Stop all Fault Management processes:

```
su - primeusr
fmctl stop
exit
```

(As the root user) pkill nco_pad

Step 4 Move the faultmgmt folder to the tmp folder:

```
su - primeusr
mv ~/faultmgmt /tmp/faultmgmt
```

Step 5 Move the faultmgmt_ 1.5.3.0_backup folder to the faultmgmt folder:

```
su - primeusr
mv ~/faultmgmt_1.5.3.0_backup/faultmgmt ~/
```

Step 6 Change ownership of the faultmgmt folder. For example:

```
chown primeusr:ncoadmin -R //opt/fm/primeusr/faultmgmt
```

Step 7 Restore debackup obtained before the upgrade:

```
cd .acsi_primeusr/bin
setenv
./de_restoredb -bfile <backupfile_full_path>
```

Step 8 Open the .cshrc file:

```
su - primeusr
vi ~/.cshrc
```

Find the following line and change jre 1.8 to jre 1.7:

```
setenv JAVA_HOME "$PRIMEFMHOME/utils/${OSTYPE}/jre1.8/"
```

Step 9 As the root user, start the nco_pad process. For example:

```
cd /opt/fm/primeusr/faultmgmt/omnibus/bin
./nco_pad
```

Step 10 Perform the below mentioned steps:

1. Login to the Prime Central portlet.
2. Remove Fault Management from the Suite Monitoring portlet.
3. Logout from the Prime Central portlet.

   As a root user, execute the below commands:

```
su - primeusr
itgctl stop
itgctl start
```

Step 11 Reintegrate Fault Management with Prime Central:

```
su - primeusr
fmctl integrate
```

Step 12 Restore the /var/.com.zerog.registry.xml file with the backup file: **/var/.com.zerog.registry.xml_backup_for_1.5.3** that was created during Prime Central Fault Management upgrade process. Execute below commands:

```
mv /var/.com.zerog.registry.xml /var/.com.zerog.registry.xml_backup_for_2.0
mv /var/.com.zerog.registry.xml_backup_for_1.5.3 /var/.com.zerog.registry.xml
```

Uninstalling Prime Central Fault Management

Note If you are also uninstalling Prime Central, you must uninstall the Fault Management component first.

Step 1 From the Prime Central portal, choose **Administration > System > Suite Monitoring > Applications** tab, and remove Fault Management.

Step 2 Use SSH to connect to the Prime Central active node and do the following:

a. Freeze the Prime Central cluster:

```
clusvcadm -Z service-name
```

b. Restart the integration layer:

```
# su - primeusr
# itgctl stop
# itgctl start
```

c. Unfreeze the Prime Central cluster:

```
clusvcadm -U service-name
```

Step 3 From the luci web interface, stop the Fault Management Service.

Step 4 Mount and add the virtual IP address to the node that was used for the Prime Central installation:

```
# ip addr add 192.168.1.170 dev eth0
# mount /dev/mapper/mpathcp1 /opt/fm
```

Step 5 Use SSH to connect to the Prime Central Fault Management virtual machine and do the following:

a. Navigate to the /var/adm/cisco/uninstall/Uninstall_Prime_Central_Fault_Management folder.

The uninstall folder contains the installvariables.properties file.

b. Uninstall Prime Central Fault Management:

```
./Uninstall_Prime_Central_Fault_Management -i silent
```

The uninstallation log files are available at /var/adm/cisco/uninstall/PrimeFM-uninstall.log-*time-stamp*.

Step 6 Unmount and remove the virtual IP address:

```
# ip addr del 192.168.1.170 dev eth0
# unmount /opt/fm
```

Uninstalling Prime Central

Step 1 Stop and disable the Prime Central cluster service on both nodes:

```
# service ricci stop
# service rgmanager stop
# service cman stop
```

Step 2 Mount and add the virtual IP address to the node that was used for the Prime Central installation:

```
# ip addr add 192.168.1.130 dev eth0
# mount /dev/mapper/mpath2p1 /opt/pc
```

Step 3 Uninstall the application:

a. As the root user, log into the Prime Central server. (If you logged in previously as a nonroot user, enter the **su - command** to become the root user.)

b. Uninstall Prime Central:

```
# cd /root/ha-stuff/pc
# ./UninstallPrimeCentral.sh 192.168.1.120
```

In the preceding command, 192.168.1.120 is the IP address of the second node.

The uninstallation log files are available at /var/adm/cisco/uninstall/UNINSTALL_LOG_*time-stamp*.

Step 4 Unmount and remove the virtual IP address:

```
# ip addr del 192.168.1.130 dev eth0
# umount /opt/pc
```