Red Hat HPC Solution 5.3
Installation Guide

Creating, managing and using high performance computing clusters running Red Hat® Enterprise Linux.
Edition 5

Mark Black    Kailash Sethuraman    Daniel Riek
Creating, managing and using high performance computing clusters running Red Hat® Enterprise Linux.
Edition 5

Mark Black
Platform Computing Inc
mblack@platform.com

Kailash Sethuraman
Platform Computing Inc
hsaliak@platform.com

Daniel Riek
Red Hat
riek@redhat.com
Legal Notice

Copyright © 2008 Red Hat.

This document is licensed by Red Hat under the Creative Commons Attribution-ShareAlike 3.0 Unported License. If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, MetaMatrix, Fedora, the Infinity Logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux® is the registered trademark of Linus Torvalds in the United States and other countries.

Java® is a registered trademark of Oracle and/or its affiliates.

XFS® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack® Word Mark and OpenStack Logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

The Red Hat HPC Solution product is no longer available from Red Hat. Please contact customer support to evaluate renewal options. Refer to the following KnowledgeBase article for more details: The Red Hat HPC Solution is a fully integrated software stack that enables the creation, management and usage of a high performance computing cluster running Red Hat Enterprise Linux.
# Table of Contents

**Preface** ................................................................. 3
   1. Document Conventions ............................................. 3
      1.1. Typographic Conventions ........................................ 3
      1.2. Pull-quote Conventions .......................................... 4
      1.3. Notes and Warnings ............................................ 5

**Chapter 1. What is the Red Hat HPC Solution** .............................................. 6

**Chapter 2. Installation Prerequisites** ......................................................... 7

**Chapter 3. Installation Procedure** .............................................................. 8
   3.1. Recommended Network Topology ....................................... 8
   3.2. Starting the Install ................................................ 8
   3.3. Updating an Existing Installation ................................... 10

**Chapter 4. Updating the Installer Node and the Compute Node Repository** ............. 13

**Chapter 5. Installing Additional Red Hat HPC Kits** ........................................ 15

**Chapter 6. Viewing Available Red Hat HPC Kits** ............................................ 16

**Chapter 7. Verifying the Red Hat HPC install** ................................................. 17

**Chapter 8. Adding Nodes to the Cluster** ....................................................... 19

**Chapter 9. Managing Node Groups** ............................................................... 23
   9.1. Adding RPM Packages in RHEL to Node Groups .................... 23
   9.2. Adding RPM Packages not in RHEL to Node Groups ............... 24
   9.3. Adding Kit Components to Node Groups ............................ 25

**Chapter 10. Synchronizing Files in the Cluster** ............................................ 27

**Chapter 11. Note on ABI Stability** ............................................................... 29

**Chapter 12. Known Issues** ........................................................................... 30

**Revision History** ....................................................................................... 31
Preface

1. Document Conventions

This manual uses several conventions to highlight certain words and phrases and draw attention to specific pieces of information.

In PDF and paper editions, this manual uses typefaces drawn from the Liberation Fonts set. The Liberation Fonts set is also used in HTML editions if the set is installed on your system. If not, alternative but equivalent typefaces are displayed. Note: Red Hat Enterprise Linux 5 and later include the Liberation Fonts set by default.

1.1. Typographic Conventions

Four typographic conventions are used to call attention to specific words and phrases. These conventions, and the circumstances they apply to, are as follows.

Mono-spaced Bold

Used to highlight system input, including shell commands, file names and paths. Also used to highlight keys and key combinations. For example:

To see the contents of the file my_next_bestselling_novel in your current working directory, enter the `cat my_next_bestselling_novel` command at the shell prompt and press **Enter** to execute the command.

The above includes a file name, a shell command and a key, all presented in mono-spaced bold and all distinguishable thanks to context.

Key combinations can be distinguished from an individual key by the plus sign that connects each part of a key combination. For example:

Press **Enter** to execute the command.

Press **Ctrl+Alt+F2** to switch to a virtual terminal.

The first example highlights a particular key to press. The second example highlights a key combination: a set of three keys pressed simultaneously.

If source code is discussed, class names, methods, functions, variable names and returned values mentioned within a paragraph will be presented as above, in mono-spaced bold. For example:

File-related classes include **filesystem** for file systems, **file** for files, and **dir** for directories. Each class has its own associated set of permissions.

Proportional Bold

This denotes words or phrases encountered on a system, including application names; dialog box text; labeled buttons; check-box and radio button labels; menu titles and sub-menu titles. For example:

Choose **System → Preferences → Mouse** from the main menu bar to launch **Mouse Preferences**. In the **Buttons** tab, select the Left-handed mouse check box and click **Close** to switch the primary mouse button from the left to the right (making the mouse suitable for use in the left hand).

To insert a special character into a gedit file, choose **Applications → Accessories →**
Character Map from the main menu bar. Next, choose Search → Find… from the Character Map menu bar, type the name of the character in the Search field and click Next. The character you sought will be highlighted in the Character Table. Double-click this highlighted character to place it in the Text to copy field and then click the Copy button. Now switch back to your document and choose Edit → Paste from the gedit menu bar.

The above text includes application names; system-wide menu names and items; application-specific menu names; and buttons and text found within a GUI interface, all presented in proportional bold and all distinguishable by context.

**Mono-spaced Bold Italic** or **Proportional Bold Italic**

Whether mono-spaced bold or proportional bold, the addition of italics indicates replaceable or variable text. Italics denotes text you do not input literally or displayed text that changes depending on circumstance. For example:

To connect to a remote machine using ssh, type `ssh username@domain.name` at a shell prompt. If the remote machine is `example.com` and your username on that machine is `john`, type `ssh john@example.com`.

The `mount -o remount file-system` command remounts the named file system. For example, to remount the `/home` file system, the command is `mount -o remount /home`.

To see the version of a currently installed package, use the `rpm -q package` command. It will return a result as follows: `package-version-release`.

Note the words in bold italics above — username, domain.name, file-system, package, version and release. Each word is a placeholder, either for text you enter when issuing a command or for text displayed by the system.

Aside from standard usage for presenting the title of a work, italics denotes the first use of a new and important term. For example:

Publican is a *DocBook* publishing system.

### 1.2. Pull-quote Conventions

Terminal output and source code listings are set off visually from the surrounding text.

Output sent to a terminal is set in **mono-spaced roman** and presented thus:

<table>
<thead>
<tr>
<th>books</th>
<th>Desktop</th>
<th>documentation</th>
<th>drafts</th>
<th>mss</th>
<th>photos</th>
<th>stuff</th>
<th>svn</th>
</tr>
</thead>
<tbody>
<tr>
<td>books_tests</td>
<td>Desktop1</td>
<td>downloads</td>
<td>images</td>
<td>notes</td>
<td>scripts</td>
<td>svgs</td>
<td></td>
</tr>
</tbody>
</table>

Source-code listings are also set in **mono-spaced roman** but add syntax highlighting as follows:
static int kvm_vm_ioctl_deassign_device(struct kvm *kvm, struct kvm_assigned_pci_dev *assigned_dev)
{
    int r = 0;
    struct kvm_assigned_dev_kernel *match;

    mutex_lock(&kvm->lock);

    match = kvm_find_assigned_dev(&kvm->arch.assigned_dev_head, assigned_dev->assigned_dev_id);
    if (!match) {
        printk(KERN_INFO "%s: device hasn't been assigned before, "
               "so cannot be deassigned\n", __func__);
        r = -EINVAL;
        goto out;
    }

    kvm_deassign_device(kvm, match);
    kvm_free_assigned_device(kvm, match);

out:
    mutex_unlock(&kvm->lock);
    return r;
}

1.3. Notes and Warnings
Finally, we use three visual styles to draw attention to information that might otherwise be overlooked.

**Note**

Notes are tips, shortcuts or alternative approaches to the task at hand. Ignoring a note should have no negative consequences, but you might miss out on a trick that makes your life easier.

**Important**

Important boxes detail things that are easily missed: configuration changes that only apply to the current session, or services that need restarting before an update will apply. Ignoring a box labeled 'Important' will not cause data loss but may cause irritation and frustration.

**Warning**

Warnings should not be ignored. Ignoring warnings will most likely cause data loss.
Chapter 1. What is the Red Hat HPC Solution

The Red Hat HPC Solution is a fully integrated software stack that enables the creation, management and usage of a high performance computing cluster running Red Hat Enterprise Linux.

The cluster management tools provided with the Red Hat HPC Solution are based on Platform OCS from Platform Computing Corporation.

For more information about Platform OCS, visit http://www.platform.com/Products/platform-open-cluster-stack
Chapter 2. Installation Prerequisites

Installing Red Hat HPC Solution (Red Hat HPC) requires one system to be designated as an installer node. This installer node is responsible for installing the rest of the nodes in the cluster.

Prior to installing Red Hat HPC, confirm that the designated machine has Red Hat Enterprise Linux 5.3 installed and meets the following requirements:

- Root partition with at least 40 GBytes free.
- Disable SELinux.
- One or more network interfaces use a statically defined IP addresses. Connect these to the networks where the machines are provisioned.
- Red Hat Enterprise Linux Version 5.3 installation media
- A valid subscription to Red Hat Network is required including an entitlement to Red Hat HPC Channel
- The firewall (iptables) must be configured to permit the services needed for installation on all networks used to provision nodes (HTTP, HTTPS, TFTP, DNS, NTP, BOOTPS, etc). A script is provided to appropriately configure the firewall.
- Red Hat HPC creates a private DNS zone for all machines under its control. The name of this zone must NOT be the same as any other DNS zone within the organization where the cluster is installed.
Chapter 3. Installation Procedure

Verify that the installer node meets the prerequisites.

Register on Red Hat Network and subscribe to the appropriate channels.

3.1. Recommended Network Topology

In its default configuration, the Red Hat HPC Solution treats one Network interface of the installer node as a public interface on which it imposes a standard firewall policy, while other interfaces are treated as trusted, private interfaces to the cluster nodes. While this can be easily adopted to the customer's preferences, it is the recommended network topology for an installation of the Red Hat HPC Solution. It provides clear separation of the public network from the private cluster-internal network(s). In that topology, the installer node acts as a gateway and firewall, protecting the cluster nodes. This allows a relaxed set of firewall and security settings within the private cluster network, while still maintaining secure operations.

Please consider the installation notes below, when planning your network topology.

For improved security, Red Hat recommends enabling the firewall on the external interfaces of the installer node and maintaining a clean separation between the public networks and the private cluster network. Also customers are advised that optional monitoring tools like Nagios®, Cacti®, or ntop disclose details of the network topology and are only accessible to authorized users over a secure connection. Red Hat recommends to use the encrypted https protocol rather than plain http connections for these services.

3.2. Starting the Install

Log into the machine as root and install the Red Hat HPC bootstrap RPM:

```
# yum install ocs mod_ssl
```

After installing the ocs RPM, source the OCS environment:
Run the installation script:

```
# /opt/kusu/sbin/ocs-setup
```

The script detects your network settings and provide a summary per NIC:

<table>
<thead>
<tr>
<th>Device</th>
<th>IP</th>
</tr>
</thead>
<tbody>
<tr>
<td>eth0</td>
<td>172.25.243.44</td>
</tr>
<tr>
<td>Network</td>
<td>172.25.243.0</td>
</tr>
<tr>
<td>Subnet</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>mac</td>
<td>00:0C:29:C4:61:06</td>
</tr>
<tr>
<td>Gateway</td>
<td>172.25.243.2</td>
</tr>
<tr>
<td>dhcp</td>
<td>False</td>
</tr>
<tr>
<td>boot</td>
<td>1</td>
</tr>
</tbody>
</table>

Note

Red Hat HPC cannot provision over DHCP configured NICs, only statically configured NICs. The OCS installer asks if you want to provision on all networks, and if not which ones to provision on.

Red Hat HPC creates a separate DNS zone for the nodes it installs. The tool prompts for this zone.

Warning

Do not use the same DNS zone as any other in your organization. Using an existing zone causes DNS name resolution problems.

Note

The Red Hat HPC Solution tries to generate IP addresses for the individual Compute Nodes by incrementing from the Installer Node’s IP address in the private cluster network. The Installer Node therefore has a low IP address in that network and a free range following that IP address, or the user must adjust the Starting IP for provisioned Compute Nodes using the “netedit” tool.

The Red Hat HPC Solution stores a copy of the OS media and installation images. The OCS installer prompts for the location of the directory to store the operating system. The default is /depot. A symbolic link to /depot is created if another location is used.

The OCS installer builds a local repository using the OS media. This repository is by OCS when provisioning compute nodes.

The OCS installer asks for the physical DVD or CDs (in the optical drive physically connected to the installer host), a directory containing the contents of the OS media, or an ISO file providing the media.
Note

if using the “file(system)” option to provide the OS media (via a subdirectory) to the OCS installer, answer “N” to the prompt for additional disks when prompted.

After the OS media is successfully imported (approximately 5-10 minutes when importing from a physical optical drive) and the local OCS repository created, a sequence of scripts runs to configure the OCS cluster for the installation.

The default firewall rules for a RHEL installation blocks the ports needed to provision nodes. The script provided configures the firewall to allow these ports. When the script runs, it opens the ports necessary for provisioning the nodes. It also configures Network Address Translation (NAT) on the installer node, so that the provisioned nodes can access the non-provisioning networks connected to the installer on other interfaces.

To run the script to configure the firewall as root, run:

```
# /opt/kusu/bin/kusurc /opt/kusu/etc/S02KusuIptables.rc.py
```

Once the installation has completed the following message appears:

```
Congratulations! The base kit is installed and configured to provision on:
Network 1.2.3.4 on interface  ethX
```

The installer node is ready to begin installing other nodes in the cluster.

Prior to installing the compute nodes it is best to add all the desired kits, and customize the node groups. If the kits are added after the Compute Nodes have been installed it is necessary to run the following command to get Nagios® and Cacti® to display the nodes in their respective web interfaces:

```
# addhost  -u
```

This causes re-generation of many of the application configuration files.

### 3.3. Updating an Existing Installation

Updating a existing Red Hat HPC cluster is a two step process. The installer node contains a Red Hat repository for RHEL 5. This repository must first be updated prior to updating the kits or running a `yum update` on the master installer. If the master installer contains packages that are newer than the packages in the Kusu repository, there can be dependency problems when installing some kits. See chapter 4 on how to update the repository.

Once the repository is updated the kits on the installer can be updated. Before the base kit can be updated, the existing addon kits in the RHHPC system must be removed. This is required as some of the older kits are not guaranteed to be compatible with RHHPC 5.3. The updated versions of the addon kits for RHHPC 5.3 must be installed.

To update kits please follow the instructions below.

**Procedure 3.1. Removing Incompatible_Kits Prior to Updating_the Base Kit**

1. Remove the kit components from the nodegroup. Run `ngedit` and select the installer node group
to edit. Go to the component screen. De-select the components of the kits you wish to upgrade. Continue and apply the changes.

2. Run the above step for all nodegroups.

3. Remove the kit associations from the repository.

   ```shell
   # repoman -e -kkitname -rreponame
   ```

   Optionally, to list repositories and associated kits, the following command can be used:

   ```shell
   # repoman -l
   ```

4. Update repository after removing kit associations:

   ```shell
   # repoman -u -rreponame
   ```

5. Remove older kits from the system.

   ```shell
   # kitops -e -kkitname
   ```

   Optionally use this command to list installed kits.

   ```shell
   # kitops -l
   ```

The base kit must be updated prior to reinstalling the other kits. The steps below outline how to update the base kit on the installer.

**Procedure 3.2. Updating the Base Kit Prior to Other Kits**

1. Ensure that the installer node can connect to Red Hat Network (RHN).

2. Update the `ocs` package:

   ```shell
   # yum update ocs
   ```

3. Source the environment:

   ```shell
   # source /etc/profile.d/kusuenv.sh
   ```

4. Run the OCS upgrade script. This will update the base kit from RHN, and rebuild the repository for installing nodes.

   ```shell
   # ocs-setup -u
   ```

The base kit is now updated. If desired the other kits can be updated. Use the following procedure to update the kits:

**Procedure 3.3. Updating Other Kits**

1. Update the kit downloaders by running the following command for the downloader you wish to upgrade

   ```shell
   #yum update ocs-kit-kitname
   ```

2. Follow the instructions in chapter 5 for installing kits
Warning

There is a known issue in upgrading the Cacti kit from Red Hat Enterprise Linux HPC Solution 5.2 to 5.3. The cacti user must be removed prior to adding the new cacti kit. Use `userdel cacti` to remove the user.

Note

There is a known issue whereby the `ocs-setup -u` command does not proceed and fails with the message `OCS setup script does not seem to have run in this machine, cannot upgrade`. To work around this, perform the following:

```
# touch /var/lock/subsys/ocs-setup
```
Chapter 4. Updating the Installer Node and the Compute Node Repository

Red Hat HPC manages updates to the installer nodes differently from all other nodes in the cluster. The RPM packages and updates to the Operating System Repository for all nodes provisioned by the installer (and that includes compute nodes and diskless nodes) are managed independently from updating the installer node. To update the installed packages on the installer node, use the following command:

```
# yum update
```

The `yum` tool downloads all of the required updates for the operating system and installs them on the installer node. Since updating installer nodes and compute nodes is separate you can choose to update the installer node – and either choose to update the compute nodes or not update the compute nodes.

Prior to updating the repository it is recommended that a snapshot (copy) of the repository be made. If there are any application issues with the updates the copy can be used:

```
# repoman -r rhel5_x86_64 -s
```

To update the compute nodes in a Red Hat HPC cluster use the following command:

```
# repopatch -r rhel5_x86_64
```

The `repopatch` tool downloads all of the required updates for the operating system and installs them into the repository for the compute nodes. `repopatch` displays an error if it is not properly configured. For example:

```
# repopatch -r rhel5_x86_64
Getting updates for rhel-5-x86_64. This may take awhile...
Unable to get updates. Reason: Please configure
/opt/kusu/etc/updates.conf
```

Edit the `/opt/kusu/etc/updates.conf` file adding your `username` and `password` for Red Hat Network to the `[rhel]` section of the file, for example:

```
[fedora]
url=http://download.fedora.redhat.com/pub/fedura/linux/

[rhel]
username=
pASSWORD==
url=https://rhn.redhat.com/XMLRPC
yumrhn=https://rhn.redhat.com/rpc/api
```

After configuring the `/opt/kusu/etc/updates.conf` file, repopatch downloads all of the updates from Red Hat Network and creates an update kit which is then associated with the rhel-5-x86_64 repository using `ngedit`. 

13
repopatch automatically associates the update kit with the correct repository. View the list of update kit components from ngedit on the Components screen and list the available update kits with the kitops command. For example:

Once repopatch has retrieved the updated packages and rebuilt the repository, the compute nodes can be updated. This is done by either reinstalling them using:

```
# boothost -r -n {Name of Node group}
```

or without reinstalling by using:

```
# cfmsync -u -n {Name of Node group}
```

The cfmsync command causes the compute nodes to start updating packages from the repository they installed from.

**Note**

Remember that yum is used to update the installer node directly from Red Hat Network or other yum repositories. The repopatch command updates the repositories used to provision compute nodes, and the cfmsync command is used to signal the compute nodes to update.
Chapter 5. Installing Additional Red Hat HPC Kits

Additional software tools such as Nagios® and Cacti are packaged as software kits. Software packaged as a kit is easier to install onto a Red Hat HPC Cluster. A kit contains rpms for the software, rpms for meta-data and configuration files.

**Note**

This might require the repositories to be updated as described in the previous section.

To install Cacti® onto the Red Hat HPC cluster use the following commands:

```
# yum install ocs-kit-cacti
# /opt/kusu/sbin/install-kit-cacti
```

To install Nagios® onto the Red Hat HPC cluster use the following commands:

```
# yum install ocs-kit-nagios
# /opt/kusu/sbin/install-kit-nagios
```

To see what kits are available use:

```
# yum search ocs-kit
```

The `yum` commands above download the respective kit downloaders from Red Hat Network. The kit downloaders are distinguished by the `ocs-kit-*` prefix. In the case of a download problem the kit downloaders can be safely re-run.

Included in the kit downloader RPM is an installation script that adds the kit to the Red Hat HPC cluster repository and rebuilds the cluster repository.

Every kit that is downloaded from Red Hat Network has a corresponding script used to install the kit into the cluster repository.
Chapter 6. Viewing Available Red Hat HPC Kits

Use the following command to query the kits available from Red Hat Network:

```bash
# yum list ocs-kit-\*
```

At the time of writing, the following kits are available:

### Table 6.1. Available Kits

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ocs-kit-cacti</td>
<td>A reporting tool</td>
</tr>
<tr>
<td>ocs-kit-lava</td>
<td>Open source LSF, a batch scheduling and queuing system</td>
</tr>
<tr>
<td>ocs-kit-nagios</td>
<td>A network monitoring tool</td>
</tr>
<tr>
<td>ocs-kit-ntop</td>
<td>A network monitoring tool</td>
</tr>
<tr>
<td>ocs-kit-rhel-java</td>
<td>The Java Runtime</td>
</tr>
<tr>
<td>ocs-kit-hpc</td>
<td>A collection of MPIs (MPICH 1,2, MVAPICH 1,2 and OpenMPI), math libraries (ATLAS, BLACS, SCALAPACK), and benchmarking tools.</td>
</tr>
<tr>
<td>ocs-kit-ganglia</td>
<td>Another system monitoring tool</td>
</tr>
<tr>
<td>ocs-kit-rhel-ofed</td>
<td>The OFED stack</td>
</tr>
</tbody>
</table>

Other non-Open Source kits are available from [http://my.platform.com](http://my.platform.com)

**Note**

The ntop kit has the potential to disclose a large degree of information about the local network. It therefore by default can only be reached from the installer node. In order to allow access from outside the cluster, you need to explicitly make it reachable by opening the ntop port 3001 in the firewall by executing:

```bash
# iptables -I INPUT -p tcp --dport 3001 -m state --state NEW,ESTABLISHED -j ACCEPT
# service iptables save
```

Use this with great care as it could provide an intruder with valuable information.
Chapter 7. Verifying the Red Hat HPC install

Once the installer node is successfully configured the next step is to verify that all software components are installed and working correctly. The following steps can be used to verify the Red Hat HPC Install.

Procedure 7.1. Verifying the HPC Install

1. Start the web browser (Firefox). The cluster homepage is displayed.
2. Use the `dmesg` command to check for hardware issues.
3. Check all network interfaces to see if they are configured and up.
   ```
   # ifconfig -a | more
   ```
4. Verify the routing table is correct.
   ```
   # route
   ```

Make sure the following system services are running:

<table>
<thead>
<tr>
<th>Service</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Server</td>
<td>service httpd status</td>
</tr>
<tr>
<td>DHCP</td>
<td>service dhcpd status</td>
</tr>
<tr>
<td>DNS</td>
<td>service named status</td>
</tr>
<tr>
<td>Xinetd</td>
<td>service xinetd status</td>
</tr>
<tr>
<td>MySQL</td>
<td>service mysqld status</td>
</tr>
<tr>
<td>NFS</td>
<td>service nfs status</td>
</tr>
</tbody>
</table>

5. Run some basic Red Hat HPC commands.
   - List the installed repositories
     ```
     # repoman -l
     ```
   - List the installed kits
     ```
     # kitops -l
     ```
   - Run the Node Group Editor
     ```
     # ngedit
     ```
   - Run the Add Host tool
     ```
     # addhost
     ```

6. Check that Cacti is installed (optional; Cacti is only available if the Cacti kit has been installed)
   From the web browser enter the following URL:
   ```
   http://localhost/cacti
   ```
   Login to Cacti with username: `admin`, password: `admin`
7. Check that Nagios is installed (optional; Nagios is only available if the Nagios kit was installed)
   From the web browser enter the following URL:
   http://localhost/nagios
   Login to Nagios with username: admin, password: admin
Chapter 8. Adding Nodes to the Cluster

The **addhost** tool adds nodes to a Red Hat HPC cluster.

**addhost** listens on a network interface for nodes that are PXE booting and adds them to a specified node group.

Node groups are templates that define common characteristics such as network, partitioning, operating system and kits for all nodes in a node group.

Open a terminal window or login to the installer node as root to add nodes.

**Procedure 8.1. Adding Nodes to the Cluster**

1. Run **addhost**

   ```
   # addhost
   ```

2. Select the node group for the new nodes. Normally compute nodes are added to the compute-rhel node group:

3. Select the network interface to listen on for new PXE booted node
4. Indicate the rack number where the nodes are located

5. **addhost** waits for the nodes to boot
6. Boot the nodes you want to add to the cluster. Wait a few seconds between powering up nodes so that the machines are named sequentially in the order they are started.

7. When a node is successfully detected by `addhost` a line appears in the **installing node status** window.
8. Exit `addhost` when Red Hat HPC has detected all nodes. The Installing node status screen does not update to indicate that the node has installed.
Chapter 9. Managing Node Groups

Red Hat HPC cluster management is built around the concept of node groups. Node groups are a powerful template mechanism that allows the cluster administrator to define common shared characteristics among a group of nodes. Red Hat HPC ships with a default set of node groups for installer nodes, packaged installed compute nodes, diskless compute nodes and imaged compute nodes. The default node groups can be modified or new node groups can be created from the default node groups. All of the nodes in a node group share the following:

- Node Name format
- Operating System Repository
- Kernel parameters
- Kits and components
- Network Configuration and available networks
- Additional RPM packages
- Custom scripts (for automated configuration of tools)
- Partitioning

A typical HPC cluster is created from a single installer node and many compute nodes. Normally compute nodes are exactly the same as each other with a few exceptions, like the node name or other host specific configuration files. A node group for compute nodes makes it easy to configure and manage 1 or 100 nodes all from the same node group. The ngedit command is a graphical TUI (Text User Interface) run by the cluster administrator to create, delete and modify node groups. The ngedit tool modifies cluster information in the Red Hat HPC database and also automatically calls other tools and plugins to perform actions or update configuration. For example, modifying the set of packages associated with a node group in ngedit automatically calls cfm (configuration file manager) to synchronize all of the nodes in the cluster using yum to add and remove the new packages, while modifying the partitioning on the node group notifies the administrator that a re-install must be performed on the nodes in the node group in order to change the partitioning. The Red Hat HPC database keeps track of the node group state, thus several changes can be made to a node group simultaneously and the physical nodes in the group can be updated immediately or at a future time using the cfmsync command.

9.1. Adding RPM Packages in RHEL to Node Groups

Open a Terminal and run the node group editor as root.

```
# ngedit
```

Select the compute-rhel node group and move through the Text User Interface screens by pressing F8 or by choosing next on the screen. Stop at the Optional Packages screen.
Additional RPM packages are added by selecting the package in the tree list. Pressing the space bar expands or contracts the list to display the available packages.

Packages are sorted alphabetically by default. The list of packages can be sorted by Red Hat groups, just choose **Toggle View** to re-sort the packages.

Select the additional packages using the **spacebar**. When a package is selected an asterisk displays beside the package name.

Package dependencies are automatically handled by **yum**. If any selected package requires other packages they are automatically included when the package is installed on the cluster nodes.

**ngedit** automatically calls **cfm** to synchronize the nodes and install new packages but, by design, does not automatically remove packages from nodes in the cluster. If required **pdsh** and **rpm** can be used to completely remove packages from the RPM database on each node in the cluster.

### 9.2. Adding RPM Packages not in RHEL to Node Groups

Red Hat HPC maintains a repository containing all of the RPM packages that ship with Red Hat Enterprise Linux. This repository is sufficient for most customers. RPM packages that are not in Red Hat Enterprise Linux can also be added to a Red Hat HPC repository by placing the RPM packages into the appropriate **contrib** directory under **/depot**. For example:

**Procedure 9.1. Adding RPM Packages not in RHEL to Node Groups**

1. Start with the RPMs that are not in Red Hat Enterprise Linux or in a Red Hat HPC Kit
2. Create the appropriate subdirectories in **/depot/contrib**:

   ```bash
   # mkdir -p /depot/contrib/rhel/5/x86_64
   # cp foo.rpm /depot/contrib/rhel/5/x86_64/foo.rpm
   ```
3. Rebuild the Red Hat HPC repository with **repoman**:

   ```bash
   # repoman -u -r rhel5_x86_64
   ```
4. It takes some time to rebuild the repository and associated images.
5. Run `ngedit` and navigate to the Optional Packages screen.
6. Select the new package by navigating within the package tree and using the spacebar to select.
7. Continue through the `ngedit` screens and either allow `ngedit` to synchronize the nodes immediately or perform the node synchronization manually with `cfmsync -p` at a later time. Example: selecting a RPM package that is not included in Red Hat Enterprise Linux

![Screenshot of `ngedit` interface](image.png)

Contributions can be added to more than one Red Hat HPC repository, the directory structure is: `/depot/contrib/<os_name>/<version>/<architecture>`

### 9.3. Adding Kit Components to Node Groups

Adding kit components to nodes in a node group is very similar to adding additional RPM packages.

1. Open a Terminal and run `ngedit`
2. Press F8 (or choose Next) and proceed to Components screen
3. Enable components on a per-node group basis.

Each Red Hat HPC kit installs an application or a set of applications. The kit also contains components which are meta-RPM packages designed for installing and configuring applications within the cluster. By enabling the appropriate components, it is easy to configure all nodes in a node group.

For example, the Cacti kit contains two components, `component-cacti` and `component-cacti-monitored-node`. `component-cacti` installs and configures Cacti, sets up the web pages and connection to the database. This component is normally installed on the cluster installer node or any other node (or set of nodes) designated as the management node.

The other component in the Cacti kit, `component-cacti-monitored-node` contains the Cacti agent code that runs on compute nodes in the cluster.

Most Red Hat HPC Kits come configured with automatic node group association and component selection. In the case of the Cacti kit, all nodes within the `compute-rhel` node group have the `component-cacti-monitored-node` component enabled. This means these nodes are monitored by Cacti by default. The component does not need to be explicitly enabled as the Cacti kit does this.
As another example, the Platform Lava kit automatically associates the Lava master with the installer node group and the Lava compute nodes with the `compute-rhel` node group. Installing the Lava kit automatically sets up and creates a usable Lava cluster without needing any additional configuration.
Chapter 10. Synchronizing Files in the Cluster

HPC clusters are built from individual compute nodes and all of these nodes must have copies of common system files such as `/etc/passwd`, `/etc/shadow`, `/etc/group` and others.

Red Hat HPC contains a file synchronization service called CFM (Configuration File Manager).

CFM runs on each compute node in the cluster and when new files are available on the installer node a message is sent to all of the nodes notifying them that files are available. Each compute node connects to the installer node and copies the new files using the HTTP protocol. All files to be synchronized by CFM are located in the directory tree `/etc/cfm/<node group>` as can be seen in the following screenshot:

In the screenshot above `/etc/cfm` directory contains several node group directories such as `compute-diskless` and `compute-rhel`. In each of those directories is a directory tree where the `/etc/cfm/<node group>` directory represents the root of the tree. The `/etc/cfm/compute-rhel/etc` directory contains several files or symbolic links to system files.

Creating symbolic links for the files in CFM allows the compute nodes to be automatically synchronized with system files on the installer node. `/etc/passwd` and `/etc/shadow` are two examples where symlinks are used.

Adding files to `cfm` is simple. Create all of the directories and subdirectories for the file then place the file in the appropriate location.

Existing files can also have a `<filename>.append` file. The contents of a `<filename>.append` file are automatically appended to the existing `<filename>` file on all nodes in the node group.

Use the `cfmsync` command to notify all of the nodes in all node groups or nodes in a single node group. For example:

```
# cfmsync -f -n compute-rhel
```

Synchronizes all files in the compute-rhel node group.
# cfmsync -f

Synchronizes all files in all node groups

For more information on `cfmsync` view the man pages.
Chapter 11. Note on ABI Stability

Red Hat's commitment to provide binary runtime compatibility as described at http://www.redhat.com/security/updates/errata/, does not to the full extent apply to the Red Hat HPC Solution cluster middleware.

Red Hat HPC Solution, as an add-on to Red Hat Enterprise Linux, closely tracks the upstream projects, in order to provide a maximum level of enablement in this fast moving area. As a consequence, Red Hat and Platform Computing, as an exception from the general practice in Red Hat Enterprise Linux, can only preserve API/ABI compatibility across minor releases to the degree, the upstream projects do.

For this reason, applications build on-top of the HPC Solution stack, might require recompilation or even source-level code changes when moving from one minor release of Red Hat Enterprise Linux to a newer one.

This is not generally required for the underlying Enterprise Linux software stack with exception of the OFED packages specified in the Red Hat Enterprise Linux release notes at http://www.redhat.com/docs/manuals/enterprise/.
Chapter 12. Known Issues

Summary: **ocs-setup -u** fails to update the system, with the message **OCS setup script does not seem to have run in this machine, cannot upgrade.**

Details: RHHPC 5.1 used to utilize a lockfile mechanism to control if the system had been installed. This has been moved to the database where the state is stored. However, when upgrading from OCS 5.1, this file still needs to be tested. If this file is removed, then **ocs-setup -u** will not correctly trigger.

Work around: Run the following command before re-running **ocs-setup -u**:

```
#touch /var/lock/subsys/ocs-setup
```

Summary: After updating the system then removing and installing the updated cacti kit, the graphs do not display properly.

Details: The cacti user's home directory was not created properly with RHHPC 5.2's cacti kit. This has a knock on effect when updating the cacti kit because the rpms do not recreate the user, if the user already exists.

Work around: Run the following command prior to running the updated install-kit-cacti kit installer script:

```
# userdel cacti
```

Summary: The ganglia user can sometimes not be created when installing ganglia, causing the services to fail.

Details: An interaction with the other addon kits can sometimes cause the ganglia user to be not created.

Symptoms: Running gmond and gmetad fail, user ganglia does not exist.

Workaround: Run the following commands to create the **ganglia** user and permission the directories correctly:

```
#useradd -d /var/lib/ganglia -s /sbin/nologin ganglia
#cd /var/lib/ganglia/
#chown ganglia:ganglia rrds
#service gmond restart
#service gmetad restart
```
# Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Author</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5-9.400</td>
<td>2013-10-31</td>
<td>Rüdiger Landmann</td>
<td>Rebuild with publican 4.0.0</td>
</tr>
<tr>
<td>5-9</td>
<td>2012-07-18</td>
<td>Anthony Towns</td>
<td>Rebuild for Publican 3.0</td>
</tr>
<tr>
<td>5-8</td>
<td>Mon Aug 16 2010</td>
<td>Michael Hideo</td>
<td>Migrating to automated publishing system</td>
</tr>
</tbody>
</table>