



**Red Hat Reference Architecture Series**

# **Deploying a Highly Available Web Server on Red Hat® Enterprise Linux® 5**

**Volume 2: GFS2 and Shared Storage**

Version 1.0

November 2008





## Deploying a Highly Available Web Server on Red Hat® Enterprise Linux® 5

### Volume 2: GFS2 and Shared Storage

Copyright © 2008 by Red Hat, Inc.

1801 Varsity Drive  
Raleigh NC 27606-2072 USA  
Phone: +1 919 754 3700  
Phone: 888 733 4281  
Fax: +1 919 754 3701  
PO Box 13588  
Research Triangle Park NC 27709 USA

"Red Hat," Red Hat Linux, the Red Hat "Shadowman" logo, and the products listed are trademarks or registered trademarks of Red Hat, Inc. in the United States and other countries. Linux is a registered trademark of Linus Torvalds.

All other trademarks referenced herein are the property of their respective owners.

© 2008 by Red Hat, Inc. This material may be distributed only subject to the terms and conditions set forth in the Open Publication License, V1.0 or later (the latest version is presently available at <http://www.opencontent.org/openpub/>).

The information contained herein is subject to change without notice. Red Hat, Inc. shall not be liable for technical or editorial errors or omissions contained herein.

Distribution of substantively modified versions of this document is prohibited without the explicit permission of the copyright holder.

Distribution of the work or derivative of the work in any standard (paper) book form for commercial purposes is prohibited unless prior permission is obtained from the copyright holder.

The GPG fingerprint of the [security@redhat.com](mailto:security@redhat.com) key is:  
CA 20 86 86 2B D6 9D FC 65 F6 EC C4 21 91 80 CD DB 42 A6 0E



# Table of Contents

1 Executive Summary.....	5
1.1 Introduction.....	5
1.2 Audience.....	5
1.3 Reference Documentation.....	5
1.1 Document Conventions.....	6
1.4 Acronyms.....	7
2 Hardware Configuration.....	7
2.1 Environment .....	8
2.2 Multicast.....	9
3 OS Installation.....	9
3.1 Installation Numbers.....	10
3.2 Additional Package Groups.....	16
4 First Boot.....	17
4.1 Initial Settings.....	18
4.1.1 Firewall (iptables).....	18
4.1.2 SELinux.....	19
4.2 Software Updates (RHN Configuration).....	20
5 OS Customization.....	29
5.1 Secure Shell.....	29
5.2 NTP.....	31
5.3 ACPI.....	32
5.4 Firewall (iptables) Rules.....	32
5.4.1 Modifying.....	32
5.4.2 Saving.....	34
5.5 SELinux.....	35
5.5.1 Booleans.....	37
5.5.2 Labeling.....	38
5.5.3 GUI.....	39
5.6 Public and Private Networks.....	39
5.7 Network Interface Bonding.....	40
6 Shared Storage.....	44
6.1 Device Mapper / Multipath.....	44
6.2 CLVM.....	49
6.2.1 Logical Volumes.....	49



6.2.2 Configuration.....	58
6.3 /etc/hosts/.....	58
7 Conga.....	59
7.1 Installing ricci.....	60
7.2 Installing luci.....	62
8 Clustering.....	63
8.1 Cluster Creation.....	65
8.1.1 Considering Quorum Disks.....	69
8.2 Configuring Cluster Members.....	75
8.3 Fencing.....	76
8.4 Failover Domains.....	78
8.5 Cluster Resources.....	79
8.5.1 Script.....	79
8.5.2 IP Address.....	80
8.6 Web Service (httpd).....	83
8.6.1 Service Creation.....	84
8.6.2 httpd Configuration Directives.....	88
8.6.3 Testing.....	88
9 SELinux Policy Adjustments.....	89
9.1 AVC Denials .....	90
9.2 audit2allow.....	92
10 Diagnostics.....	94
10.1 clustat.....	94
10.2 Logs.....	94
11 Conclusions & Next Steps.....	95
Appendix A: Using Local Web Content.....	95
Appendix B: Configuration Files.....	95
Cluster.....	95
Firewall.....	97
Multipathing.....	98
Network Interfaces.....	100
Appendix C: RHN.....	100
Manual Configuration.....	100
Modifying Subscriptions.....	107
Appendix D: Issue Tracking.....	112
Appendix E: Procedure Checklist.....	113



# 1 Executive Summary

This paper details the deployment of a highly available web service on a Red Hat Enterprise Linux 5 cluster. While volume 1 of this document involved Red Hat Cluster Suite (RHCS) serving NFS based web content, volume 2 will focus on the configuration of a 2-node RHCS cluster with shared storage and quorum device, a single image global file system (GFS2), cluster logical volume management (CLVM), and the implementation of a highly available web service.

## 1.1 Introduction

A cluster is essentially a group of two or more computers working together which, from an end user's perspective, appear as one server. Clustering can be used to enable storage clustering, balance load among cluster members, parallel processing, and high-availability. The "highly available" aspect of any cluster service indicates that it is configured in a manner such that the failure of any one cluster member, or a subsystem failure within a member, will not prevent the continued availability of the service itself. This document will illustrate the procedure for creating and maintaining an Apache-based HTTP server, with firewall and security enhanced OS, providing availability across clustered nodes.

## 1.2 Audience

Although this document does not require extensive Linux expertise, it is expected that the end user possess some knowledge and/or experience at networking and basic system administration skills.

## 1.3 Reference Documentation

This document does not intend to reproduce existing documentation to an extent where it would then be required that the documents be kept in synch should either change over time. To that end, the following list includes the Red Hat Enterprise Linux 5.2 documents that were used explicitly for the operating system installation and cluster creation procedures as well as other documents and articles that were helpful in assembling the information.

- **Red Hat Enterprise Linux Installation Guide**

[http://www.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/5.2/html/Installation\\_Guide/index.html](http://www.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5.2/html/Installation_Guide/index.html)

- **Red Hat Cluster Suite Overview**

[http://www.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/5.2/pdf/Cluster\\_Suite\\_Overview/Cluster\\_Suite\\_Overview.pdf](http://www.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5.2/pdf/Cluster_Suite_Overview/Cluster_Suite_Overview.pdf)

- **LVM Administrator's Guide**

[http://www.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/5.2/html/Cluster\\_Logical\\_Volume\\_Manager/index.html](http://www.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5.2/html/Cluster_Logical_Volume_Manager/index.html)



ml

- **Configuring and Managing a Red Hat Cluster**  
[http://www.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/5.2/html/Cluster\\_Administration/index.html](http://www.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5.2/html/Cluster_Administration/index.html)
- **Global File System**  
[http://www.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/5.2/html/Global\\_File\\_System/index.html](http://www.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5.2/html/Global_File_System/index.html)
- **Using Device-Mapper Multipath**  
[http://www.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/5.2/html/DM\\_Multipath/index.html](http://www.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5.2/html/DM_Multipath/index.html)
- **Cluster Project Frequently Asked Questions (FAQ)**  
<http://sources.redhat.com/cluster/faq.html>
- **Red Hat Knowledgebase Article 13315 (multipathed qdisk)**  
[http://kbase.redhat.com/faq/FAQ\\_46\\_13315.shtm](http://kbase.redhat.com/faq/FAQ_46_13315.shtm)

Although they may be referenced where necessary for common Red Hat Enterprise Linux installation and Red Hat Cluster Suite configuration instructions, any procedures relevant to the configuration and management of a highly available web server using shared storage will be included in this document with their specific details.

## 1.1 Document Conventions

As in most procedural reference documents, certain paragraphs in this manual are represented in different fonts, typefaces, sizes, and color. This highlighting is helpful in determining command line user input from text file content. Information represented in this manner include the following:

- command names  
Linux commands like `iptables` or `yum` will be differentiated by font.
- user input  
All commands demonstrated in this document are assumed run as root. User entered commands and their respective output are displayed as seen below.

```
# echo "This is an example of command line input and output, some of which  
may be highlighted."
```

```
This is an example of command line input and output, some of which may be  
highlighted.
```

- file content  
Listing the content or partial content of a text file will be displayed as seen below.

```
# This is the appearance of commented text contained within a file
```



## 1.4 Acronyms

Common acronyms used within this document are listed below.

ACPI	Advanced Configuration and Power Interface
AVC	Access Vector Cache
CLVM	Cluster Logical Volume Manager
CSSD	Cluster Services Synchronization Daemon
DLM	Distributed Lock Manager
DRAC	Dell Remote Access Controller
EULA	End User License Agreement
GFS	Global File System
GNBD	Global Network Block Device
HA	High-Availability
HBA	Host Bus Adapter
HTTP	Hypertext Transfer Protocol
HTTPD	Hypertext Transfer Protocol Daemon
ILO	Integrated Lights Out
IP	Internet Protocol
IPMI	Intelligent Platform Management: Interface
LUN	Logical Unit Number
NTP	Network Time Protocol
OS	Operating System
RAID	Redundant Arrays of Independent Disks
RHCS	Red Hat Cluster Suite
RHEL	Red Hat Enterprise Linux
RHN	Red Hat Network
SAN	Storage Area Network

## 2 Hardware Configuration

Referencing the *Configuration Basics* section in *Configuring and Managing a Red Hat Cluster* provides the necessary information for physically connecting the hardware (servers, switches, interconnects, etc.) for cluster use prior to OS installation.

There should be at least two Network Interface Cards (NIC), whether embedded or added to each server. One NIC will be configured with an external IP address while the other will be



configured as an interconnect between cluster members using a local switch. Clusters are very dependent on a constant heartbeat between nodes which are maintained across the local interconnect. It is highly recommended that a private network be used to avoid outside factors such as high network traffic or network hardware failures.

Please reference the *Configuring and Managing a Red Hat Cluster* guide as it illustrates the required cluster connectivity in detail.

**Note:** Refer to *Appendix E* in this document for a complete checklist of this procedure.

## 2.1 Environment

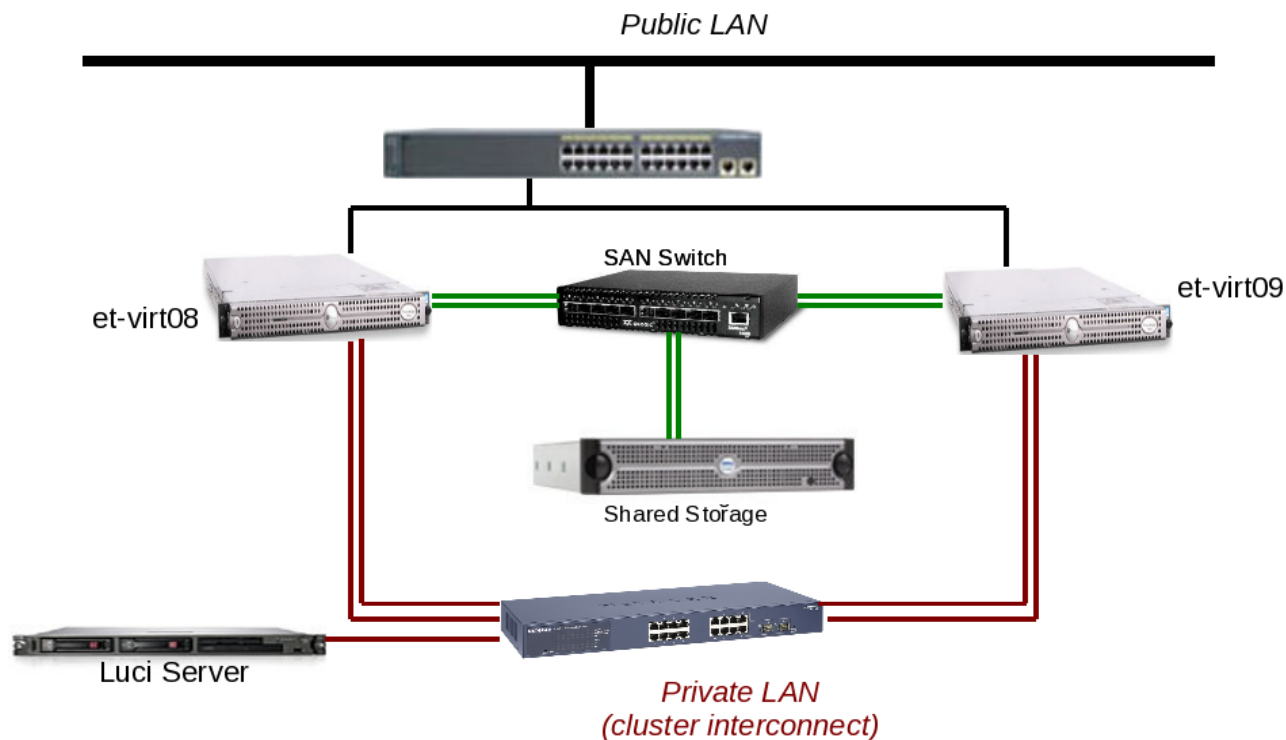
This section provides information about the specific hardware and software used to build a highly available web server.

The following table describes the primary components of the test environment.

System (et-virt08)	Dell PowerEdge 1850 RHEL 5.2 (2.6.18-92.1.6.el5) (4) Intel <sup>(R)</sup> Xeon <sup>(TM)</sup> CPU 2.80GHz 4 GB RAM gigabit ethernet (2) QLA2312 2G FC HBA (rev 02)
System (et-virt09)	Dell PowerEdge 1850 RHEL 5.2 (2.6.18-92.1.6.el5) (4) Intel <sup>(R)</sup> Xeon <sup>(TM)</sup> CPU 2.80GHz 4 GB RAM gigabit ethernet (2) QLA2312 2G FC HBA (rev 02)
Storage	(2) EMC CLARiiON AX100
SAN Switch	QLogic SANbox 1400
Network Switch	Catalyst 2960G Switch
Private Interconnect	NetGear ProSafe 16-port Gigabit Smart Switch

The diagram below illustrates the hardware component connectivity.





## 2.2 Multicast

By default, the newer cluster infrastructure with openais (Red Hat Enterprise Linux 5, etc.) uses multicast. This allows the configuration of a cluster with nodes running on different network subnets. There are some Cisco switches that do not support IP multicast in their default configuration. Since openais uses multicast for cluster communications, multicast should be enabled at the switch(es) to facilitate the use of cluster software. Before making any changes to Cisco switches, it is recommended to contact Cisco Support to ensure the changes will have no negative consequences on the network.

Reference the following URL for more information regarding Cisco switches and multicast: [http://www.openais.org/doku.php?id=faq:cisco\\_switches](http://www.openais.org/doku.php?id=faq:cisco_switches)

## 3 OS Installation

Reference the Red Hat Enterprise Linux Installation Guide for the specific details regarding the acquisition and installation of Red Hat Enterprise Linux. The guide will include information specific to the platform on which the installation will take place (x86, AMD64, Intel® 64 and Itanium) so be sure to read the appropriate section for your platform.

Once the platform specific information has been understood and the hardware configuration has been performed to accommodate a cluster, install Red Hat Enterprise Linux 5.2 on the server(s) using the preferred method. The installation methods include:

- CD-ROM or DVD

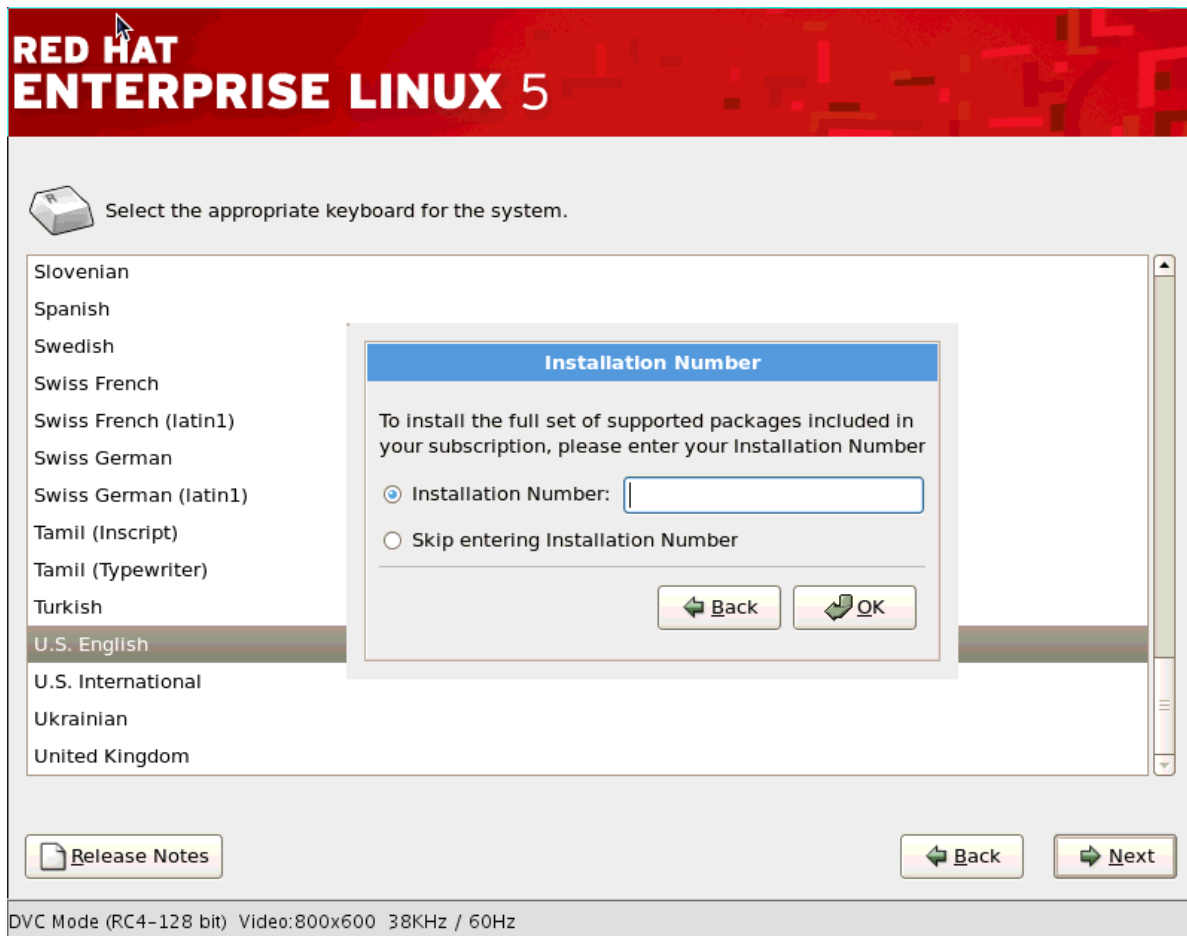


- Network
- FTP or HTTP
- NFS
- Hard Drive

Regardless of the method chosen, the install process will guide the user through the procedures unique to each method and then begin the OS installation. The Red Hat Enterprise Linux Installation Guide will provide details regarding each of the screens that will be presented during the installation process.

### 3.1 Installation Numbers

After some language and keyboard prompts, the user will be presented with an option to use an installation number to select a predetermined group of packages assembled for specific purposes.



For this effort, an installation number associated with the Red Hat Enterprise Linux 5 Advanced Platform configuration was used, which includes:

- Red Hat Cluster Suite
- Global File System



- Virtualization

The use of installation numbers has obvious advantages as it guarantees that all the required packages for each package group, as well as any resulting dependencies, are handled by the installer and ready for configuration after the OS installation. It also ensures that the resulting OS installation will be supported by Red Hat Network (RHN) for server registration, configures RHN entitlements and automatically subscribes the server to the appropriate channels for subsequent OS updates. Installation numbers can be obtained from Red Hat Customer Service, within a new subscription activation email or via Red Hat's Subscription Management web page.

Reference the Red Hat Enterprise Linux 5 Installation Number FAQ for answers to the common questions regarding the use of installation numbers.

To view your installation numbers on the Subscription Management page for your RHN account, log into RHN where you will be placed in the page entitled *Your RHN*.



Select the Red Hat Customer Center box on the left side of the page to view the Customer Center page.

redhat.com - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://www.redhat.com/wapps/support/protected/overview.html

redhat.com

United States ( [change](#) ) Downloads | Fedora | Red Hat Network [Cart](#) | [Your Account](#) | [Logout](#)

redhat Search

Home Solutions Services & Products Partners Developers Training **Support** Store

Knowledgebase Documentation Downloads Offerings Support Policy Support Process Customer Center

## Customer Center

Hello **Steve**. Your account number is           . Edit your [Red Hat account](#).

Overview **Subscriptions** Renewals

### Your Support Overview

Company: **blah**

Subscriptions		Systems		Tickets	
Active Subscriptions	81	Total Systems	0 <a href="#">View</a>	Open Tickets	0
Expired Subscriptions	0	Inactive Systems	0 <a href="#">View</a>	Tickets Awaiting Your Response	0
Subscriptions Due to Expire	1	Out of Date Systems	0 <a href="#">View</a>	You do not currently have any web support entitlements.	
<a href="#">Manage Subscriptions</a>		Unentitled Systems	0 <a href="#">View</a>		
<a href="#">Manage Renewals</a>		<a href="#">View all your systems at Red Hat Network</a>			

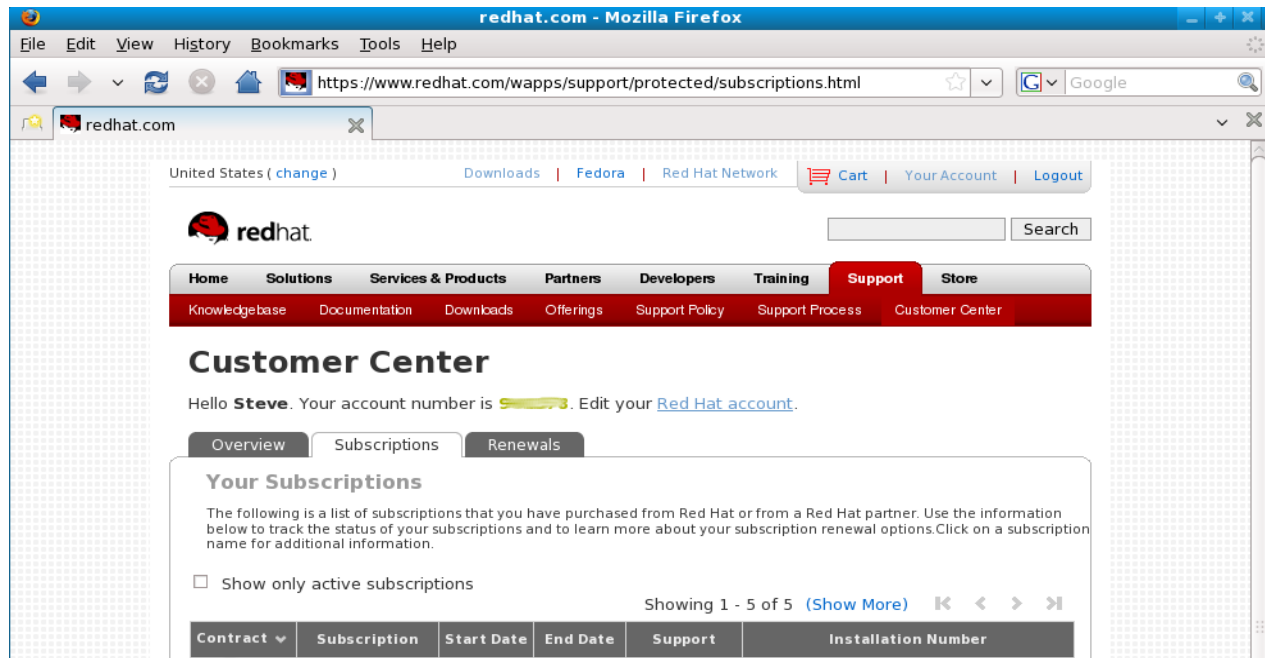
#### More Support Options

- [Customer Service FAQ](#)
- [Transitioning from Enterprise Linux 3 or 4 to Enterprise Linux 5](#)
- [Global support services](#)
- [Global support hours and numbers](#)

Copyright © 2008 Red Hat, Inc. All rights reserved.  
[Privacy Policy](#) : [Terms of Use](#) : [Patent Promise](#) : [Company](#) : [Contact](#)

Done www.redhat.com

Select the gray *Subscriptions* tab. The page displayed will list the installation numbers available to you.

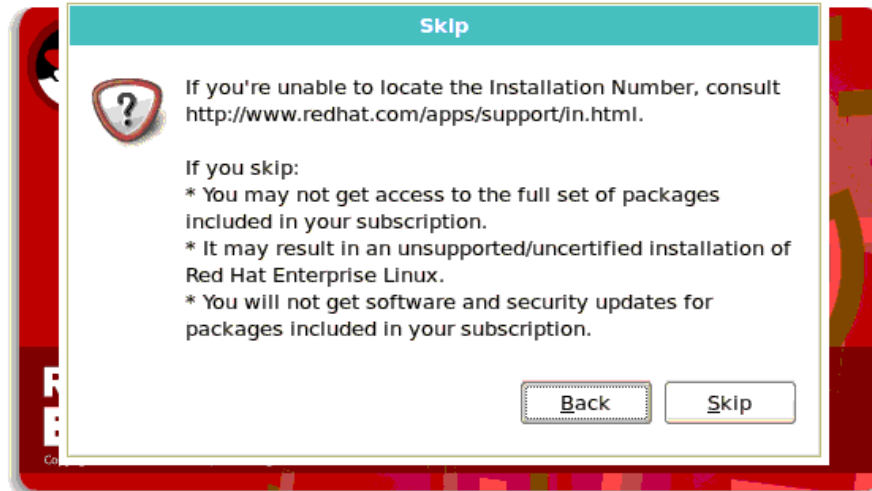



It is not necessary to use an installation number. The opportunity to select individual packages for installation will be provided at a later time. See the *Configuring and Managing a Red Hat Cluster* guide or the *Red Hat Cluster Suite* section of this document for instructions on manually installing and configuring the Red Hat Cluster software. The RHN server registrations, entitlements and subscriptions will need to be established before the clustering software can be installed from RHN.

If the user opts to choose packages manually, a pop-up message will inform them of the advantages of using an installation number and the caveats of not (seen below). If so desired, click 'Skip' to proceed.



# RED HAT ENTERPRISE LINUX 5



 Release Notes

 Back

 Next

Note that while an installation number is not required, using one that includes more package groups than are required will do no harm. For instance, users requiring cluster functionality are in no manner inconvenienced by the inclusion of the Virtualization package group. The presence of the additional software would have no effect on the clustering functionality.

The Red Hat Enterprise Linux installer will continue with the procedure, offering hard drive selection and partitioning layout options.



# RED HAT ENTERPRISE LINUX 5

Installation requires partitioning of your hard drive. By default, a partitioning layout is chosen which is reasonable for most users. You can either choose to use this or create your own.

Remove linux partitions on selected drives and create default layout. ▾

Select the drive(s) to use for this installation.

sda 70002 MB MAXTOR ATLAS10K5\_73SCA

+ Advanced storage configuration

Review and modify partitioning layout

[Release Notes](#)

[Back](#)

[Next](#)

For this project, the internal hard drive was formatted for a fresh installation and the default layout (volumes, sizes, etc.) were used.

Once the hard drive and partitioning preferences have been identified, the user will be given configuration opportunities for the boot loader, network interface, timezone and root password. Refer to the Red Hat Enterprise Linux Installation Guide for specific instructions. Remember that one NIC must be configured with a public (external) IP address.



## 3.2 Additional Package Groups

Once the basic packages have been identified for installation, whether by installation number or manual selection, the option to further customize the package content is presented.

The screenshot shows the 'RED HAT ENTERPRISE LINUX 5' installation window. The title bar is red with the text 'RED HAT ENTERPRISE LINUX 5' in white. Below the title bar, the text reads: 'The default installation of Red Hat Enterprise Linux Server includes a set of software applicable for general internet usage. What additional tasks would you like your system to include support for?'. Below this text is a list of package groups with checkboxes: 'Clustering' (checked), 'Software Development' (unchecked), 'Storage Clustering' (checked), 'Virtualization' (checked), and 'Web server' (checked). The 'Web server' option is highlighted with a blue background. Below the list, the text reads: 'You can further customize the software selection now, or after install via the software management application.' Below this text are two radio buttons: 'Customize later' (selected) and 'Customize now' (unselected). At the bottom left is a button labeled 'Release Notes' with a document icon. At the bottom right are two buttons: 'Back' with a left arrow and 'Next' with a right arrow.

Note that in the above example, because an installation number was used, options for:

- Virtualization
- Clustering
- Storage Clustering

are included in this window and are automatically preselected for install, leaving the user the option to add Software Development and/or Web Server functionality to the OS.

As seen below, when the same installation procedure is executed without using the Advanced Platform installation number, the three package groups listed above are not present in package customization window.





# RED HAT ENTERPRISE LINUX 5


The default installation of Red Hat Enterprise Linux Server includes a set of software applicable for general internet usage. What additional tasks would you like your system to include support for?

Software Development

Web server

You can further customize the software selection now, or after install via the software management application.

Customize later     Customize now

 [Release Notes](#)

 Back

 Next

Regardless of the installation method chosen, select the check box next to the Web Server option to provide the necessary software for the intended web service.

This window also provides the opportunity to manually select the packages for installation. If the user has opted not to use an installation number to specify component packages, selecting the Customize Now button will allow the user to hand select all the packages for install according to their preference. See the *Package Group Selection* section of the *Configuring and Managing a Red Hat Cluster* guide for details.

At this point, the installation will check and resolve all selected package dependencies and install the specified packages. When completed, the user is prompted to reboot to continue initial configurations.

## 4 First Boot

After the freshly installed OS reboots and the user has accepted the End User License Agreement (EULA), additional configuration windows are presented. These options are presented only once after a fresh OS installation but can be configured easily afterward if the user does not possess all the necessary information to configure them at this time. Among the first time configurations are initial settings for security features such as Linux Firewall (aka: iptables) and Security Enhanced Linux (SELinux).



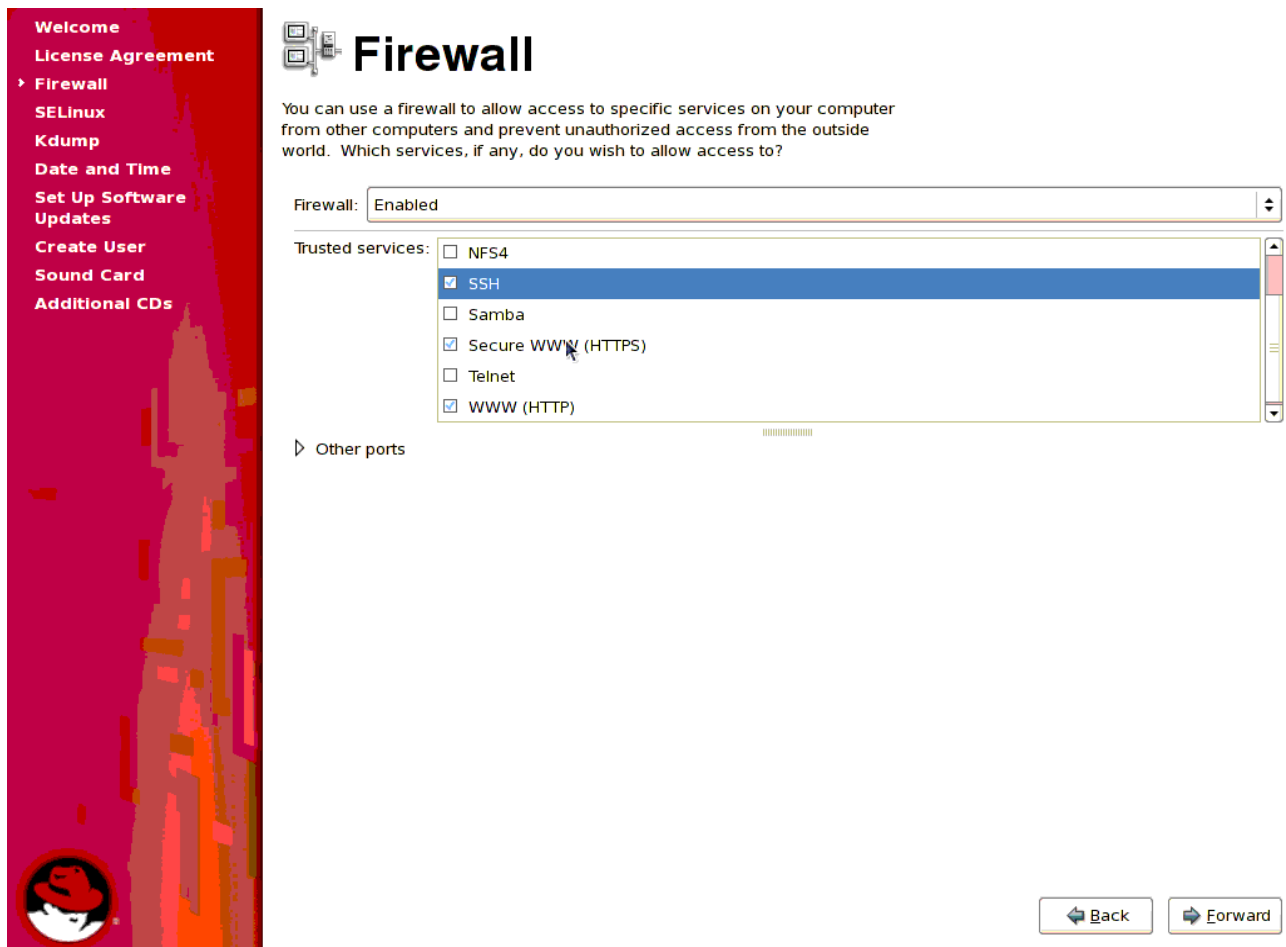
## 4.1 Initial Settings

### 4.1.1 Firewall (iptables)

It is recommended that you configure a firewall for any Red Hat Enterprise Linux server with an Internet connection. The role of the firewall is to exist between the server and the network to:

- prevent viruses from entering the server
- prevent unauthorized user access
- specify which services are accessible by end users

The firewall configuration window allows the user to enable or disable iptables. It also provides a list of installed and available, trusted services. In the example below, note the SSH service is preselected. SSH is the preferred method of communication among members in a secure cluster.



Leave the firewall enabled unless security is not an area of concern.

The HTTP protocol is used by Apache and by other web servers. Selecting **WWW (HTTP)** will



include the HTTP daemon (httpd) package automatically.

If the SSL version of HTTP (HTTPS) is required, select the **Secure WWW (HTTPS)** check box.

After having selected the trusted service(s) and clicking the Forward button, the user is prompted to confirm that they wish to alter the default security level of the server and override the existing firewall configuration. Click Yes and proceed.

## 4.1.2 SELinux

The user will now have the option to set the initial setting for SELinux.



SELinux is an implementation of *mandatory access control* in the Linux kernel. It can be set to any one of three modes with regard to the SELinux security policy:

- Enforcing (policy is enforced)
- Permissive (prints warnings instead of enforcing)
- Disabled (no policy)

By default, SELinux is enabled (aka: 'Enforcing') at installation and can be left as is for now.



Later it will be temporarily set to Permissive to help determine the necessary adjustments to SELinux policy.

Continue with the initial server configurations including kdump (enable or disable) and setting system date & time.

## 4.2 Software Updates (RHN Configuration)

The next step in the initial setup procedures is configuring the host to receive software updates from RHN. If the user has already arranged an account on RHN, then associating the newly installed server with that account now is the simplest method, especially if an installation number was used during install.

Note that it is not required that the newly installed system be registered with RHN at this time. It can be registered later using the procedures outlined in *Appendix C* of this document.

After configuring the system date & time, the option to setup software updates is displayed.

**Welcome**  
License Agreement  
Firewall  
SELinux  
Kdump  
Date and Time  
Set Up Software Updates  
Create User  
Sound Card  
Additional CDs

### Set Up Software Updates

This assistant will guide you through connecting your system to Red Hat Network (RHN) for software updates, such as:

- Your Red Hat Network or Red Hat Network Satellite login
- A name for your system's Red Hat Network profile
- The address to your Red Hat Network Satellite (optional)

If you do not have a Red Hat Login, this assistant will allow you to create one.

[Why Should I Connect to RHN? ...](#)

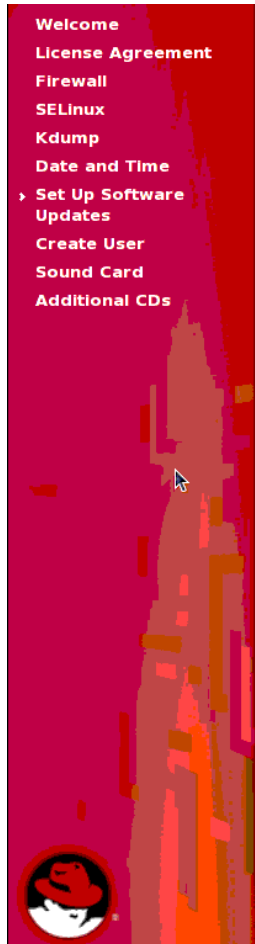
Would you like to register your system at this time?  
**(Strongly recommended.)**

Yes, I'd like to register now.

No, I prefer to register at a later time.

[Back](#) [Forward](#)

By selecting Yes, the option to choose whether to receive the updates from RHN or from a separately maintained satellite server.



## Choose Server

You may connect your system to **Red Hat Network** (<https://rhn.redhat.com/>) or to a **Red Hat Network Satellite** or **Red Hat Network Proxy** in order to receive software updates.

I'd like to receive updates from **Red Hat Network**. (I don't have access to a Red Hat Network Satellite or Proxy.)

I have access to a **Red Hat Network Satellite** or **Red Hat Network Proxy**. I'd like to receive software updates from the Satellite or Proxy below:

Red Hat Network Location:

Example: <https://satellite.example.com>

[Advanced Network Configuration ...](#)

[← Back](#)

[Forward →](#)

See the RHN Architectural Overview for details on how it is available in two different deployment models (hosted, satellite), based on your server management needs and size of deployment. A satellite server keeps all RHN functionality on your network, providing greater functionality and customization. The satellite server connects with Red Hat over the internet to acquire new content and updates.

For this effort, the cluster members were configured to receive updates from RHN directly. By choosing so, the user is then given the option to enter any proxy information that may be necessary for internal server to access the external network.



- Welcome
- License Agreement
- Firewall
- SELinux
- Kdump
- Date and Time
- Set Up Software Updates
- Create User
- Sound Card
- Additional CDs



## Choose Server

You may connect your system to **Red Hat Network** (<https://rhn.redhat.com/>) or to a **Red Hat Network Satellite** or **Red Hat Network Proxy** in order to receive software updates.

- I'd like to receive updates from **Red Hat Network**. (I don't have access to a Red Hat Network Satellite or Proxy.)
- I have access to a **Red Hat Network Satellite** or **Red Hat Network Proxy**. I'd like to receive software updates from the Satellite or Proxy below:

**Advanced Network Configuration**

**HTTP Proxy**

I would like to connect to Red Hat Network via an HTTP proxy.

Proxy Location:

**Example:** squid.example.com:3128

Use Authentication with HTTP Proxy:

Proxy Username:

Proxy Password:

Close

Advanced Network Configuration ...

Back

Forward

When prompted in the next screen, enter your RHN credentials.




# Red Hat Login

Please enter your account information for **Red Hat Network** (<http://rhn.redhat.com/>)

Login:

Password:

 Tip: Forgot your login or password? Look it up at <https://www.redhat.com/wapps/sso/rhn/lostPassword.html>

[Create a New Login](#)

[← Back](#)

[→ Forward](#)

This will direct the user to the Profile Creation window where the server name should already be present in the System Name field.



Welcome  
License Agreement  
Firewall  
SELinux  
Kdump  
Date and Time  
Set Up Software Updates  
Create User  
Sound Card  
Additional CDs

## Create Profile

**System Name**

You'll want to choose a name for this system so you'll be able to identify it in the Red Hat Network interface.

System Name:

**Profile Data**

You'll need to send us a profile of what packages and hardware are installed on your system so we can determine what updates are available.

Send hardware profile

Send package profile

Choose whether or not to send a snapshot of the server hardware and/or package profiles.





Once completed, the Subscription Review window will be displayed listing the software channel subscriptions applied to the server.

Welcome  
License Agreement  
Firewall  
SELinux  
Kdump  
Date and Time  
Set Up Software Updates  
Create User  
Sound Card  
Additional CDs

## Review Subscription

Please review the subscription details below:

**Software channel subscriptions:**

This system will receive updates from the following Red Hat Network software channels:

- rhel-x86\_64-server-5
- rhel-x86\_64-server-vt-5
- rhel-x86\_64-server-cluster-5
- rhel-x86\_64-server-cluster-storage-5
- rhn-tools-rhel-x86\_64-server-5

Warning: If an installed product on this system is not listed above, you will not receive updates or support for that product. If you would like to receive updates for that product, please visit <http://rhn.redhat.com/> and subscribe this system to the appropriate software channels to get updates for that product. See Kbase article 6227 for more details. ([http://kbase.redhat.com/faq/FAQ\\_58\\_6227.shtml](http://kbase.redhat.com/faq/FAQ_58_6227.shtml))

**RHN service level:**

Depending on what RHN modules are associated with a system, you'll enjoy different benefits of Red Hat Network. The following are the RHN modules associated with this system:


- Management module: automatic updates, systems grouping, systems permissions, system package profiling
- Virtualization Platform module: software updates for an unlimited number virtual guests of this system, access to additional software channels for guests of this system.

Back Forward

When the same procedure is executed without using an installation number, the Review Subscription window will not show the additional channels to support the package groups included by using an installation number.



- Welcome
- License Agreement
- Firewall
- SELinux
- Kdump
- Date and Time
- Set Up Software Updates
- Create User
- Sound Card
- Additional CDs



## Review Subscription

Please review the subscription details below:

### Software channel subscriptions:

This system will receive updates from the following Red Hat Network software channels:

- rhel-x86\_64-server-5

Warning: If an installed product on this system is not listed above, you will not receive updates or support for that product. If you would like to receive updates for that product, please visit <http://rhn.redhat.com/> and subscribe this system to the appropriate software channels to get updates for that product. See Kbase article 6227 for more details. ([http://kbase.redhat.com/faq/FAQ\\_58\\_6227.shtml](http://kbase.redhat.com/faq/FAQ_58_6227.shtml))

### RHN service level:

Depending on what RHN modules are associated with a system, you'll enjoy different benefits of Red Hat Network. The following are the RHN modules associated with this system:

- Management module: automatic updates, systems grouping, systems permissions, system package profiling

← Back

Forward →

If the user opts to not register the server with RHN for updates at this time, they will be informed that the Updates Setup procedure is complete and that the server is not setup for software updates.



Welcome  
License Agreement  
Firewall  
SELinux  
Kdump  
Date and Time  
Set Up Software Updates  
Create User  
Sound Card  
Additional CDs

## Finish Updates Setup

Your system is not setup for software updates.

You won't be able to receive software updates, including security updates, for this system.

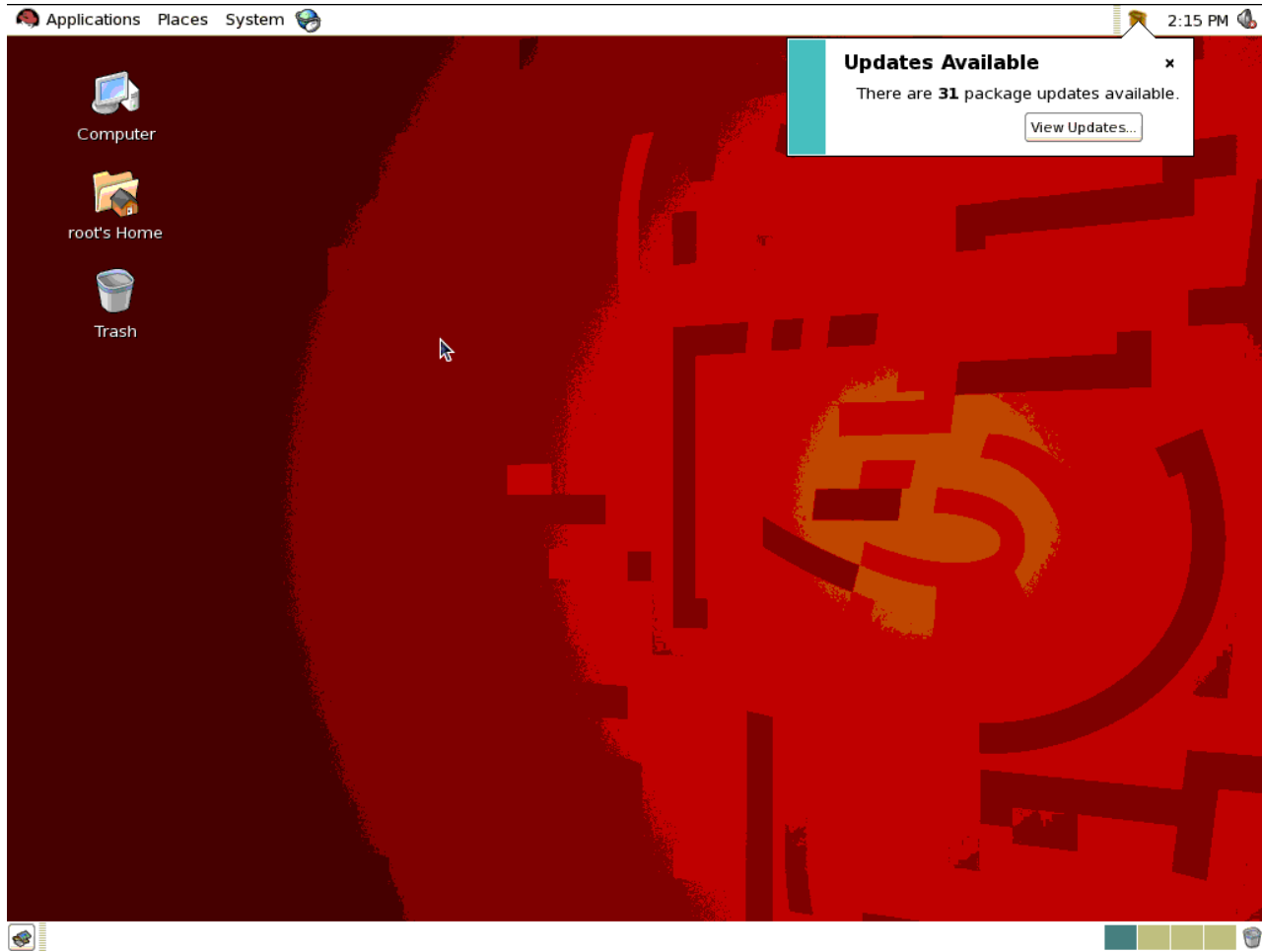
To keep your system updated, secure, and supported, please connect this system to RHN at your earliest convenience. You may access this software updates setup tool at any time by running Software Updater in the Applications > System Tools menu.

[Back](#) [Forward](#)

Once the RHN configuration procedure is either completed or skipped, the user will be provided configuration windows for Create User, Sound Card detection and configuration, and the option to install any additional CD content. The initial OS configuration procedures are now completed and the login screen is presented at console.



Upon first root login, provided the public network interface has been configured, the system will immediately check for system updates.





## 5 OS Customization

This section describes alterations to the default OS settings required to accommodate a Red Hat cluster.

### 5.1 Secure Shell

OpenSSH (ssh, scp, sftp, etc.) is a secure replacement for the unsecured binaries such as telnet, rsh, rcp, ftp, etc. The two primary security benefits provided by ssh are strong encryption, making communication difficult to intercept, and digital signature keys, hindering the ability to impersonate legitimate traffic from a trusted machine. Each host must create its own security key and share it amongst the other trusted cluster members in order to create an environment not requiring passwords or pass phrases.

The first step in configuring ssh is generating keys on the host and producing an *authorized\_keys* file that contains the public keys of trusted systems.

```
# ssh-keygen -t dsa
Generating public/private dsa key pair.
Enter file in which to save the key (/root/.ssh/id_dsa):
/root/.ssh/id_dsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase): <Enter your passphrase>
Enter same passphrase again: <Enter your passphrase>
Your identification has been saved in /root/.ssh/id_dsa.
Your public key has been saved in /root/.ssh/id_dsa.pub.
The key fingerprint is:
19:84:51:11:0f:e1:e7:87:9a:44:40:9c:1b:d1:4c:e2 root@et-virt08
```

The newly generated public key is then appended to the *authorized\_keys* files on the other cluster member.

```
# ssh-copy-id -i ~/.ssh/id_dsa.pub et-virt09
21
root@et-virt09's password: <Enter passwd>
Now try logging into the machine, with "ssh 'et-virt09'", and check in:

    .ssh/authorized_keys

to make sure we haven't added extra keys that you weren't expecting.

# ssh et-virt09
Last login: Thu Sep 11 13:19:32 2008 from et-virt08.lab.bos.redhat
```

Under some conditions, such as when a new trusted system is configured and an existing system has collected keys from the other trusted systems, it may be quicker to copy the collected keys to the new system. In such case, securely copy the *authorized\_keys* file to the new node.



```
# scp et-virt08:~/ .ssh/authorized_keys .
The authenticity of host 'et-virt08 (10.16.41.100)' can't be established.
RSA key fingerprint is 11:70:79:26:ee:81:25:17:1b:5f:27:c2:e2:42:ae:2b.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'et-virt08' (RSA) to the list of known hosts.
root@et-virt08's password: <Enter password>
authorized_keys
100% 600 0.6KB/s 00:00
```

ssh-agent is a mechanism to remember authenticated identities throughout an entire session. Used in conjunction with ssh-add, identities can be authenticated and stored or the active identities can be listed. In the example below, the agent is started. Listing the remembered phrases initially shows none. One is added, shown, then used to log into another node without having to type a password or pass phrase.

```
# exec ssh-agent $SHELL

# ssh-add -L
The agent has no identities.

# ssh-add
Enter passphrase for /root/.ssh/id_dsa: <Enter passphrase>
Identity added: /root/.ssh/id_dsa (/root/.ssh/id_dsa)

# ssh-add -L
ssh-dss
AAAAB3NzaC1kc3MAAACBAIuWk5ToS8eAjKOHGR847Km6yZRV2lXbkAQL3r5ilVIjei83v7PteAgEm/2M
8XdHwI+nzzLansvnrk4l0VdLlt/177srzwJVxXXlioybIvbzy+h/Id5A3JuQVzf+GBcw6AwwbH3t0ks3
cI+TCleernK8SkSVNVWdznzCJUOoWOLHAAAAFQDaq/pkOJUUsA80zqtAnN7aGHt6q3QAAAIAXmX0+rK2
lNvmXqazJTaa0oKg1ifTKMN7b9CmAbv0+6SomeODgvU35RcOAYDYd9kCX4OHgnMztTHOBKj6CICQyby
YwFxrZ64YfZ+9RxxXuNGbTEdM4WE9V2YX68UqZ66S2YcfTyk8Swr7Rz5U12DQHhR5iM7E70DQTjEU0Bd
3AAAAIBbnUxpVFMb6HiZn9XDUH9mZGRN+SWgiXrYMVMnY5Xd9LFYhb5YPKtTsjm+07SSGHH37ZiVdcHH
4gXTWs6l9t9DSGWSJ5zNd0N5sP6Rh5iCuZRfapf6TCmpCyAV+fcvh+bFKBiwosyx67SYZ+XpCs2j8Yz
q9PAASVvCJld/8M5Yg== /root/.ssh/id_dsa

# ssh -l root et-virt09.lab.bos.redhat.com
Last login: Tue Jul 22 15:14:21 2008 from et-virt08.lab.bos.redhat.com

# exit
logout

Connection to et0-virt09.lab.bos.redhat.com closed.
```

The user can now log into the other node without having to type a password or phrase although the initial login will require accepting the key fingerprint for that host.

```
# ssh -l root et-virt09
The authenticity of host 'et-virt09 (10.16.41.77)' can't be established.
RSA key fingerprint is c9:5f:b5:4d:36:64:f8:60:88:5f:01:84:99:76:f4:c3.
Are you sure you want to continue connecting (yes/no)? yes
```



```
Warning: Permanently added 'et-virt09,10.16.41.77' (RSA) to the list of known
hosts.
Last login: Tue Jul 22 15:14:21 2008 from et-virt08

# exit
logout
Connection to et-virt09 closed.
```

Each subsequent login will no longer require any user input. Repeat this procedure as needed on other cluster members and ensure that each node can log into and execute remote shell commands on any other node. e.g., from node et-virt08 ...

```
# ssh et-virt09 date
Wed Aug 27 09:45:45 EDT 2008
```

## 5.2 NTP

The synchronization of system clocks in a cluster becomes infinitely more important when storage is shared among members. System times can be synchronized to a network time server via the Network Time Protocol (NTP) by using the `ntpd` time daemon. This daemon can be started by using the command:

```
# service ntpd start
Starting ntpd: [ OK ]
```

To ensure `ntpd` starts automatically after each boot:

```
# chkconfig ntpd on
```

To verify that `ntpd` is enabled and at what runtime init levels:

```
# chkconfig --list | grep ntpd
ntpd          0:off  1:off  2:on   3:on   4:on   5:on   6:off
```

NTP daemon configuration parameters are set in the file `/etc/ntp.conf`. The default `ntp.conf` file shipped with Red Hat Enterprise Linux details the various configuration options for `ntpd`. The `server` entries in this file inform the daemon with which NTP server(s) it should synchronize time.

```
server <NTP server name or IP address>
```

This is typically set to an internal NTP server (within any corporate firewall) but there are public NTP servers that can be used as well. It is not as important that the system times across members be correct as it is that they be the same.

`ntpstat` will report the synchronization state and approximate time accuracy of the NTP daemon running on the local machine.

```
# ntpstat
```



```
synchronized to NTP server (66.187.233.4) at stratum 2
time correct to within 37 ms
polling server every 1024 s
```

## 5.3 ACPI

Please reference the *Configuring ACPI For Use with Integrated Fence Devices* section in *Configuring and Managing a Red Hat Cluster*. As described there, disabling ACPI Soft-Off allows an integrated fence device to shut down a server immediately rather than attempting a clean shutdown. Soft-Off allows some components to remain powered so the system can be roused from input from the keyboard, clock, modem, LAN, or USB device and subsequently takes longer to fully shutdown.

If a cluster member is configured to be fenced by an integrated fence device, disable ACPI Soft-Off for that node. Otherwise, if ACPI Soft-Off is enabled, an integrated fence device can take four or more seconds to turn off a node (refer to note that follows). In addition, if ACPI Soft-Off is enabled and a node panics or freezes during shutdown, an integrated fence device may not be able to turn off the node. Under those circumstances, fencing is delayed or unsuccessful. Consequently, when a node is fenced with an integrated fence device and ACPI Soft-Off is enabled, a cluster recovers slowly or requires administrative intervention to recover.

```
# chkconfig acpid off
```

## 5.4 Firewall (iptables) Rules

Specific IP ports will need to be identified to the firewall in order to enable the communication between clustered servers. The *Enabling IP Ports on Cluster Nodes* section of *Configuring and Managing a Red Hat Cluster* lists the IP port numbers, their respective protocols, the components to which the port numbers are assigned, and references to the `iptables` syntax needed to define the specific firewall rules.

### 5.4.1 Modifying

To accommodate RHCS communication requirements, the IP ports for these services will be enabled.

- `openais` (Linux HA, application failover)
- `rgmanager` (cluster resources)
- `ricci` (remote configuration interface agent)
- `gnbd`
- `dlm`
- `ccsd`

Execute the following commands to instruct the firewall to accept traffic for these services on their corresponding port numbers. The examples below enable the IP ports for the processes listed above.





opeanais [5404,5405]:

```
# iptables -I RH-Firewall-1-INPUT -m state --state NEW -m multiport -p udp
--dport 5404,5405 -j ACCEPT
```

rgmanager [41966, 41967, 41968, 41969]:

```
# iptables -I RH-Firewall-1-INPUT -m state --state NEW -m multiport -p tcp
--dports 41966,41967,41968,41969 -j ACCEPT
```

ricci [11111]:

```
# iptables -I RH-Firewall-1-INPUT -m state --state NEW -m multiport -p tcp
--dports 11111 -j ACCEPT
```

gnbd [14567]:

```
# iptables -I RH-Firewall-1-INPUT -m state --state NEW -m multiport -p tcp
--dports 14567 -j ACCEPT
```

dln [21064]:

```
# iptables -I RH-Firewall-1-INPUT -m state --state NEW -m multiport -p tcp
--dports 21064 -j ACCEPT
```

cssd [50006, 50007, 50008, 50009]:

```
# iptables -I RH-Firewall-1-INPUT -m state --state NEW -m multiport -p tcp
--dports 50006,50008,50009 -j ACCEPT
```

```
# iptables -I RH-Firewall-1-INPUT -m state --state NEW -m multiport -p udp
--dports 50007 -j ACCEPT
```

Verify the new rules have been added to the *RH-Firewall-1-INPUT* chain using `iptables -L`

...

```
# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination
RH-Firewall-1-INPUT  all  --  anywhere              anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination
RH-Firewall-1-INPUT  all  --  anywhere              anywhere

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source                destination
ACCEPT    udp  --  anywhere              anywhere                state NEW multiport
dports 50007
ACCEPT    tcp  --  anywhere              anywhere                state NEW multiport
dports 50006,50008,50009
```



```
ACCEPT      tcp -- anywhere anywhere state NEW multiport
dports 21064
ACCEPT      tcp -- anywhere anywhere state NEW multiport
dports 14567
ACCEPT      tcp -- anywhere anywhere state NEW multiport
dports vce
ACCEPT      tcp -- anywhere anywhere state NEW multiport
dports 41966,41967,41968,41969
ACCEPT      udp -- anywhere anywhere state NEW multiport
dports hpoms-dps-lstn,netsupport
ACCEPT      all -- anywhere anywhere
ACCEPT      icmp -- anywhere anywhere icmp any
ACCEPT      esp -- anywhere anywhere
ACCEPT      ah -- anywhere anywhere
ACCEPT      udp -- anywhere 224.0.0.251 udp dpt:mdns
ACCEPT      udp -- anywhere anywhere udp dpt:ipp
ACCEPT      tcp -- anywhere anywhere tcp dpt:ipp
ACCEPT      all -- anywhere anywhere state
RELATED,ESTABLISHED
ACCEPT      tcp -- anywhere anywhere state NEW tcp
dpt:ssh
ACCEPT      tcp -- anywhere anywhere state NEW tcp
dpt:https
ACCEPT      tcp -- anywhere anywhere state NEW tcp
dpt:http
REJECT      all -- anywhere anywhere reject-with icmp-
host-prohibited
```

Additional ports can be made accessible for other specific needs. For instance, to access the VNC desktop of a cluster member using `vncviewer`, execute the following on the target server to enable the port used by VNC:

```
# iptables -I RH-Firewall-1-INPUT -m state --state NEW -p tcp --destination-port
5900 -j ACCEPT
```

Likewise, to access a cluster member via VNC using a web browser:

```
# iptables -I RH-Firewall-1-INPUT -m state --state NEW -p tcp --destination-port
5800 -j ACCEPT
```

If `luci` is running on a separate server (not a member of the cluster) that has firewall rules enforced, that specific IP port (8084) will require enabling on that server:

```
# iptables -I RH-Firewall-1-INPUT -m state --state NEW -m multiport -p tcp
--dports 8084 -j ACCEPT
```

## 5.4.2 Saving

The previous `iptables` commands alter the firewall rules dynamically in memory but they are not yet stored permanently and will be lost if the system is rebooted before doing so. Once the necessary firewall rules have been applied, the configuration can be made



persistent across server reboots by instructing the iptables service to preserve the current rules in `/etc/sysconfig/iptables`.

```
# service iptables save
Saving firewall rules to /etc/sysconfig/iptables:      [ OK ]
```

Now each time the system boots, the firewall init script will use the `iptables - restore` command to apply the saved rules.

To save the firewall rules to an external file that can be parsed, used as backup or used as distribution for other systems:

```
# iptables-save > <filename>
```

Stored rules can be applied to this or other systems using the `iptables - restore` command.

```
# iptables-restore <filename>
```

Note that if `/etc/sysconfig/iptables` is distributed to other systems, iptables will have to be restarted in order to read the new rules.

```
# service iptables restart
```

## 5.5 SELinux

SELinux provides a more secure environment by making it more difficult to tamper with the system in the manner an intruder might. Consequently, SELinux can sometimes hamper the legitimate use of an application and will log audit messages when the application is accessed. Each release of Red Hat Enterprise Linux introduces new policies for SELinux, which in turn simplify the administrator's role in protecting servers using the SELinux subsystem.

When SELinux prevents an activity that it deems a conflict with the established security policy, its behavior is dependent on the modes that are determined by its configuration file, `/etc/selinux/config` (or its symbolic link, `/etc/selinux/config`). The settings contained in this file are set at the time of OS installation or later using the GUI tools, `system-config-securitylevel` or `system-config-selinux`.

The default content of the SELinux configuration file looks as follows:

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



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

The two controlling definitions are SELINUX and SELINUXTYPE. As described in the *Initial SELinux Settings* section above, the SELINUX variable can have one of three settings (enforcing, permissive, or disabled).

- *Enforcing*: This is the default mode and tells the system to run with SELinux watching all system access checks and to stop all "Denied" access. The kernel is blocking all access unless they are explicitly allowed. All denied accesses are reported in the logging system as Access Vector Cache (AVC) denials, unless policy has explicitly told the kernel to *dontaudit* the message.
- *Permissive*: The kernel will report access violations in the form of AVC denial messages but will allow the access. The kernel will continue to create properly labeled files. There are a couple of major differences in the way the kernel reports these denials.
  1. The kernel will only report the first access violation in permissive mode for a confined domain on a particular object, where as in enforcing mode, it will report each and every denied access.
  2. The user can get many additional AVC messages that would never have shown up in enforcing mode. An example being a confined domain that is not allowed to read a directory or any of the files in it. In Enforcing mode ,the directory access would be denied and one AVC message would be generated. In Permissive mode, the directory access would generate an AVC and each file read would generate another AVC.
- *Disabled*: This setting tells the init program to disable SELinux entirely and stops the creation of proper labels on the files. SELinux should only be disabled if the user do not intend to use it. Permissive mode should be used when diagnosing a problem.

The SELinux configuration file is read only at system boot so if the user desires to disable SELinux, they will have to either reboot after setting this desired mode or set the mode dynamically using `setenforce 0` to turn on permissive mode or `setenforce 1` to alter enforcing mode dynamically.

If the user needs to edit SELinux at boot, they can enter 'selinux=0' as a boot parameter on the kernel command line. This will cause init to avoid reading the SELinux configuration directory altogether. Alternatively, the system can boot in permissive mode by booting with the 'enforcing=0' parameter or in enforcing mode with the 'enforcing=1' parameter.

The SELINUXTYPE variable can be set to one of two modes (targeted, or strict).

- *Targeted*: This policy was introduced to provide additional security to some of the more commonly used daemons like httpd, dhcpd, mailman, named, portmap and many more. The purpose of the targeted policy is to increase security in the most important areas without reducing usability.
- *Strict*: This policy runs every program in a restrictive domain and is not as easy to use. By default, it denies all and permits only specified allowances.



For the purpose of configuring a web server, SELinux is intended to be run in a targeted enforcing mode. To debug initial SELinux violations, it will be set to permissive mode only during the configuration procedures in order to determine the exceptions to policy that the user will need to address before running in enforcing mode. Rather than authoring a new policy from scratch, this procedure will concentrate on the individual instances where SELinux prevents any activity required to host a web service and augment the current policy with exceptions for those specific actions.

Modifying the SELinux mode can be accomplished at the initial boot after an OS installation, by running the GUI application from the Red Hat Enterprise Linux 5 desktop (System ⇒ Administration ⇒ Security Level and Firewall), or via the command line using `setenforce` and `getenforce`.

```
# setenforce 0

# getenforce
Permissive
```

## 5.5.1 Booleans

SELinux policy is customizable based on least access required. By default, SELinux prevents certain http scripts from functioning. httpd policy is flexible and has several booleans that allow a user to manipulate the policy and run httpd with the tightest access possible. Booleans are on/off toggle settings that provide a method of modifying the SELinux policy behavior without requiring the authoring of new policy. These booleans are queried and set using the `getsebool` and `setsebool` commands or via the GUI, `system-config-selinux`. Users can configure their web services as securely as possible using the flexible HTTPD policies.

To view the current `selinux_httpd` settings on the command line (remember that the 'Web Server' package group was included at the time of installation and that the initial SELinux configuration included HTTP and HTTPS as trusted services):

```
# getsebool -a | grep httpd
allow_httpd_anon_write --> off
allow_httpd_bugzilla_script_anon_write --> off
allow_httpd_mod_auth_pam --> off
allow_httpd_nagios_script_anon_write --> off
allow_httpd_squid_script_anon_write --> off
allow_httpd_sys_script_anon_write --> off
httpd_builtin_scripting --> on
httpd_can_network_connect --> off
httpd_can_network_connect_db --> off
httpd_can_network_relay --> off
httpd_disable_trans --> off
httpd_enable_cgi --> on
httpd_enable_ftp_server --> off
httpd_enable_homedirs --> on
httpd_rotatelogds_disable_trans --> off
httpd_ssi_exec --> off
```



```
httpd_suexec_disable_trans --> off
httpd_tty_comm --> on
httpd_unified --> on
```

For more detailed information on these booleans, reference the manpage for `httpd_selinux`.

## 5.5.2 Labeling

SELinux requires files to have an extended attribute to define the file type and is very sensitive regarding file and directory labeling. Every process and object on the system has an associated label. Access could be denied to files not labeled correctly. If the labeling is correct, everything should work together smoothly.

SELinux policy governs the access various daemons have to these files. These context types are defined for `httpd`.

<code>httpd_sys_content_t</code>	Set files with <code>httpd_sys_content_t</code> for content available from all <code>httpd</code> sys scripts and the daemon
<code>httpd_sys_script_exec_t</code>	Set cgi scripts with <code>httpd_sys_script_exec_t</code> to allow access to all sys types
<code>httpd_sys_content_rw_t</code>	Set files with <code>httpd_sys_content_rw_t</code> if you want <code>httpd_sys_script_exec_t</code> scripts the ability to read/write the data while preventing other non sys scripts from accessing
<code>httpd_sys_content_ra_t</code>	Set files with <code>httpd_sys_content_ra_t</code> if you want <code>httpd_sys_script_exec_t</code> scripts to read/append to the file while preventing other non sys scripts from accessing
<code>httpd_unconfined_script_exec_t</code>	Set cgi scripts with <code>httpd_unconfined_script_exec_t</code> to allow them to run without any SELinux protection. This should only be used for a very complex <code>httpd</code> scripts, after exhausting all other options. Use of this script is preferred rather than turning off SELinux for <code>httpd</code> protection

The default location from which the web service serves its web content is `/var/www/html/`. This document demonstrates using a non default location from which to serve its web content (`/web_content`). In doing so, the user needs to inform SELinux that the files stored there need to be accessible to the web server process. This is accomplished by setting correct labeling. Apache already has permission to access files labeled `httpd_sys_content_t` so to label the `/web_content` directory accordingly, use the `semanage` utility.

```
# semanage fcontext -a -t httpd_sys_content_t '/web_content(/.*)?'
```

This tells SELinux that the `/web_content` directory and all files under it should be labeled `httpd_sys_content_t`. Other Linux tools read this data when they are labeling or relabeling files.

`semanage` is the command to change the actual labels on files on your machine. The user will need to run `restorecon` to fix the actual labels. `restorecon` uses SELinux to determine



how files under `/web_content` should be labeled and repairs the labeling as needed.

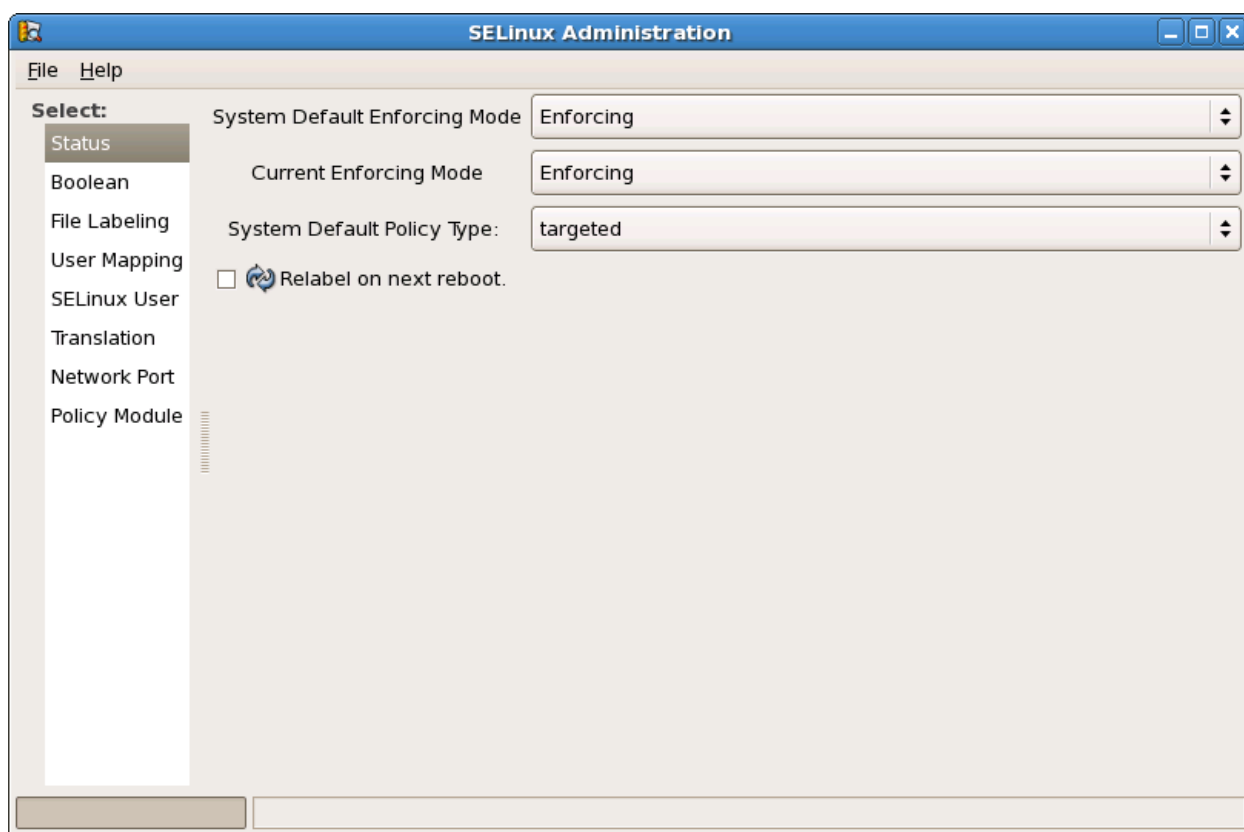
```
# restorecon -R /web_content
```

Use `matchpathcon` to view the default label for a specified path.

```
# matchpathcon /web_content
/web_content      system_u:object_r:httpd_sys_content_t
```

## 5.5.3 GUI

To view or set the boolean values or labels using the GUI interface, run `system-config-selinux` to view the SELinux Administration window ...



... and select *Boolean* or *File Labeling* from the list at left to configure the specific boolean or label values.

## 5.6 Public and Private Networks

The primary reason for requiring at least two network interfaces for clustering is to separate cluster traffic from all other network traffic. Cluster traffic is comprised of heartbeats and inter-node communication and is normally confined to a private (local) network. Clusters are immensely dependent on their inter-node communication (aka: heartbeats) for their integrity.

The user will configure one network interface to connect to the public network. Another





network interface on the system can be configured to communicate on a private LAN. The primary public interface was configured at OS installation by supplying a system name (if not named automatically via DHCP), the IP address, and subnet mask. This can also be accomplished afterward using `ifconfig` on the command line or via the network configuration GUI, `system-config-network`.

Below is a common network configuration for a system participating in a cluster.

```
# ifconfig -a
eth0      Link encap:Ethernet  HWaddr 00:13:72:4C:A2:36
          inet addr:10.16.41.77  Bcast:10.16.47.255  Mask:255.255.248.0
          inet6 addr: fe80::213:72ff:fe4c:a236/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:651474 errors:0 dropped:0 overruns:0 frame:0
          TX packets:80630 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:78508453 (74.8 MiB)  TX bytes:13453503 (12.8 MiB)
          Base address:0xbcc0 Memory:df6e0000-df700000

eth1      Link encap:Ethernet  HWaddr 00:04:23:D8:03:AC
          inet addr:10.10.1.1   Bcast:10.10.1.255  Mask:255.255.255.0
          inet6 addr: fe80::204:23ff:fed8:3ac/64 Scope:Link
          UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
          RX packets:28 errors:0 dropped:0 overruns:0 frame:0
          TX packets:28 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4994 (4.8 KiB)  TX bytes:5126 (5.0 KiB)
          Base address:0xccc0 Memory:df8e0000-df900000

.
.
.
```

Note that in the above example, `eth0` was configured for access to the public LAN while `eth1` was assigned an address on a private LAN. This may vary depending on which connections are configured when the hardware is physically cabled.

## 5.7 Network Interface Bonding

Where multiple network interfaces are available, NIC bonding can be implemented for additional availability and is the only current method to provide NIC failover. This section describes how one public network interface was configured for public use while two other interfaces were bonded to function as one interface on the private LAN.

The network card defined as the public interface was chosen at OS installation. Display all of the ethernet interfaces available and their present configurations.

```
# ifconfig -a
eth0      Link encap:Ethernet  HWaddr 00:04:23:D8:03:AC
          inet6 addr: fe80::204:23ff:fed8:3ac/64 Scope:Link
          BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
```





```
RX bytes:4556 (4.4 KiB) TX bytes:10442 (10.1 KiB)
Base address:0xe8c0 Memory:dfdc0000-dfde0000

eth1 Link encap:Ethernet HWaddr 00:04:23:D8:03:AC
inet6 addr: fe80::204:23ff:fed8:3ac/64 Scope:Link
BROADCAST MULTICAST MTU:1500 Metric:1
RX packets:28 errors:0 dropped:0 overruns:0 frame:0
TX packets:28 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:4994 (4.8 KiB) TX bytes:5126 (5.0 KiB)
Base address:0xccc0 Memory:df8e0000-df900000

eth2 Link encap:Ethernet HWaddr 00:13:72:4C:A2:36
inet addr:10.16.41.77 Bcast:10.16.47.255 Mask:255.255.248.0
inet6 addr: fe80::213:72ff:fe4c:a236/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:651474 errors:0 dropped:0 overruns:0 frame:0
TX packets:80630 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:78508453 (74.8 MiB) TX bytes:13453503 (12.8 MiB)
Base address:0xbcc0 Memory:df6e0000-df700000

.
.
.
```

Since interface eth2 was configured for public LAN access, eth0 and eth1 will be used to configure a single bonded network interface providing a failover path for cluster communication.

To create a configuration file for the new interface, bond0, start with the config file for the first interface to be bonded. This can be used as a template for creating a configuration file for an interface file since the same hardware address that will be used for the bonded link

```
# cd /etc/sysconfig/network-scripts

# ls -l ifcfg-*
-rw-r--r-- 3 root root 185 Aug 27 11:36 ifcfg-eth0
-rw-r--r-- 3 root root 185 Aug 27 11:36 ifcfg-eth1
-rw-r--r-- 3 root root 162 Aug 6 09:12 ifcfg-eth2
-rw-r--r-- 1 root root 254 Mar 3 10:39 ifcfg-lo

# cat ifcfg-eth0
# Intel Corporation 82541GI Gigabit Ethernet Controller
DEVICE=eth0
BOOTPROTO=dhcp
HWADDR=00:04:23:D8:03:AC
ONBOOT=no

# cp ifcfg-eth0 ifcfg-bond0
```

Edit the *ifcfg-bond0* file and make the following modifications.

1. Change the DEVICE setting to 'bond0'



2. Change any entries present in the file for IPADDR, NETMASK, NETWORK or BROADCAST to those chosen for this system's local interconnect
3. Ensure BOOTPROTO setting to 'none'
4. Ensure that ONBOOT is set to 'yes'
5. Ensure USERCTL is present and set to 'no'
6. Add BONDING\_OPTS with the desired options for when *ifcfg-bond0* is executed at network start

The *ifcfg-bond0* file used for this server resembled the following.

```
DEVICE=bond0
BOOTPROTO=none
ONBOOT=yes
NETMASK=255.255.255.0
IPADDR=10.10.1.1
USERCTL=no
TYPE=Ethernet
IPV6INIT=no
PEERDNS=yes
BONDING_OPTS="mode=1 miimon=100 primary=eth0"
```

In the example above, 10.10.1.1 was used for this system's local interconnect (10.10.1.2 was used for the other cluster member) and the bonding options include specifying eth0 as the primary interface.

Next, modify the configuration files for the interfaces chosen for bonding (*ifcfg-eth0* and *ifcfg-eth1*).

1. Ensure the ONBOOT entry in each file is set to 'yes'
2. Add the following lines:
  - SLAVE=yes
  - MASTER=bond0

```
ONBOOT=yes
SLAVE=yes
MASTER=bond0
```

Edit */etc/modprobe.conf* to load bonding with the desired options when *ifcfg-bond0* is executed at network start. In this case the file is appended with the following line.

```
alias bond0 bonding mode=1 miimon=100 primary=eth0
```

Lastly, restart the network service to start the newly created configuration file for interface bond0.

```
# service network restart
```



```
Shutting down interface eth2: [ OK ]
Shutting down loopback interface: [ OK ]
Disabling IPv4 packet forwarding: net.ipv4.ip_forward = 0
[ OK ]
Bringing up loopback interface: [ OK ]
Bringing up interface bond0: [ OK ]
Bringing up interface eth2: [ OK ]
#
```

Once the networking subsystem has restarted, the bond link should be up and running.

```
# ifconfig -a
bond0 Link encap:Ethernet HWaddr 00:0E:0C:B6:5D:03
inet addr:10.10.1.1 Bcast:10.10.1.255 Mask:255.255.255.0
inet6 addr: fe80::20e:cff:feb6:5d03/64 Scope:Link
UP BROADCAST RUNNING MASTER MULTICAST MTU:1500 Metric:1
RX packets:464136 errors:0 dropped:0 overruns:0 frame:0
TX packets:854453 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:83022738 (79.1 MiB) TX bytes:144597240 (137.8 MiB)

eth0 Link encap:Ethernet HWaddr 00:0E:0C:B6:5D:03
inet6 addr: fe80::20e:cff:feb6:5d03/64 Scope:Link
UP BROADCAST RUNNING SLAVE MULTICAST MTU:1500 Metric:1
RX packets:463930 errors:0 dropped:0 overruns:0 frame:0
TX packets:854423 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:82990062 (79.1 MiB) TX bytes:144591117 (137.8 MiB)
Base address:0xe8c0 Memory:dfdc0000-dfde0000

eth1 Link encap:Ethernet HWaddr 00:0E:0C:B6:5D:03
inet6 addr: fe80::20e:cff:feb6:5d03/64 Scope:Link
UP BROADCAST RUNNING SLAVE MULTICAST MTU:1500 Metric:1
RX packets:206 errors:0 dropped:0 overruns:0 frame:0
TX packets:30 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:32676 (31.9 KiB) TX bytes:6123 (5.9 KiB)
Base address:0xccc0 Memory:df8e0000-df900000

eth2 Link encap:Ethernet HWaddr 00:13:72:4C:A5:E5
inet addr:10.16.41.71 Bcast:10.16.47.255 Mask:255.255.248.0
inet6 addr: fe80::213:72ff:fe4c:a5e5/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:2797646 errors:0 dropped:0 overruns:0 frame:0
TX packets:99864 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:384844792 (367.0 MiB) TX bytes:10274604 (9.7 MiB)
Base address:0xbcc0 Memory:df6e0000-df700000

...
```



## 6 Shared Storage

Shared storage indicates external storage accessible by every member in the cluster. For the purposed of increased availability, it is configured with multiple physical paths between the servers and the storage in an attempt to reduce any single point of cluster failure. Whether using iSCSI, fibre channel or another means of connecting to shared storage, higher availability is typically accomplished by using:

- multiple HBAs on the host
- multiple switches (if applicable) between host and storage
- multiple controllers on the storage arrays
- using RAID sets when creating LUNs on the storage arrays

In this manner, the chances of any one hardware failure resulting in a loss of cluster functionality is greatly reduced.

Since there are many vendors of storage available, this document assume that LUNs have been created on the storage and presented to the hosts for device discovery.

For this demonstration, three RAID-5 LUNs (one 200gb and two 40gb) were created and presented to all HBAs on each host, for a total of 240gb of web content space.

A single 1gb, RAID-1/0 LUN was also presented to all HBAs to serve as a cluster quorum device. Considerations for quorum devices will be examined later in the *Cluster Creation* section of this document.

### 6.1 Device Mapper / Multipath

Device Mapper Multipath (dm-multipath) is described here since it is shipped and supported with Red Hat Enterprise Linux. Other multipath solutions may be available. Please reference the Using Device-Mapper Multipath guide for additional details on the configuration and use of dm-multipath.

If the *device-mapper-multipath* package is not installed, install the package using yum:

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

Load the dm\_multipath kernel module if it is not already loaded.

```
# modprobe dm_multipath
```

One of the primary issues with shared storage in clusters is the lack of persistent device naming, or LUN persistence. Without it, a device discovered as */dev/sdc* could easily be discovered after next boot as */dev/sdg*, effectively breaking clustering as it expects the devices to remain consistent across system reboots. LUN persistence guarantees that each device will have the same name, on each cluster member, after restarting.

If the physical configuration of the SAN has included multiple paths to storage for added throughput and availability, then we should see these multiple paths to the devices in */proc/partitions* by comparing the block sizes.



```
# cat /proc/partitions
major minor #blocks name
 8      0  71687325 sda
 8      1   104391 sda1
 8      2  71577607 sda2
 8     16   1048576 sdb
 8     32  209715200 sdc
 8     48   41943040 sdd
 8     64   41943040 sde
 8     80   9437184 sdf
 8     96  209715200 sdg
 8    112   41943040 sdh
 8    128   41943040 sdi
 8    144   9437184 sdj
 8    160   1048576 sdk
 8    176   1048576 sdl
 8    192  209715200 sdm
 8    208   41943040 sdn
 8    224   41943040 sdo
 8    240   9437184 sdp
65      0  209715200 sdq
65     16   41943040 sdr
65     32   41943040 sds
65     48   9437184 sdt
65     64   1048576 sdu
253     0  69533696 dm-0
253     1   2031616 dm-1
253     3  209715200 dm-3
253     4   41943040 dm-4
253     5   41943040 dm-5
253     7  292552704 dm-7
```

Given that the LUNs created for our use are 1gb, 40gb, and 200gb in size, it can be determined that four paths exist per device. In the above case we see four block devices (sdc, sdg, sdm, sdq) are all different paths to the same 200gb LUN presented from storage.

The `scsi_id` command can be used to verify this by comparing the WWID of a each device.

```
# for i in `awk '/sd/ {print $4}' /proc/partitions`; \
do echo "$i: `scsi_id -g -u -s /block/$i`"; done
sda: 20010b9fc080ee839
sda1:
sda2:
sdb: 360060160c914150142d0d3f571bfdc11
sdc: 360060160541415013a6c98657610dd11
sdd: 36006016054141500d0ce586cb2d8dc11
sde: 36006016054141500183f857db2d8dc11
sdf: 36006016054141501266764dc3c02dd11
sdg: 360060160541415013a6c98657610dd11
sdh: 36006016054141500d0ce586cb2d8dc11
sdi: 36006016054141500183f857db2d8dc11
sdj: 36006016054141501266764dc3c02dd11
```



```
sdk: 360060160c914150142d0d3f571bfdc11
sdl: 360060160c914150142d0d3f571bfdc11
sdm: 360060160541415013a6c98657610dd11
sdn: 36006016054141500d0ce586cb2d8dc11
sdo: 36006016054141500183f857db2d8dc11
sdp: 36006016054141501266764dc3c02dd11
sdq: 360060160541415013a6c98657610dd11
sdr: 36006016054141500d0ce586cb2d8dc11
sds: 36006016054141500183f857db2d8dc11
sdt: 36006016054141501266764dc3c02dd11
sdu: 360060160c914150142d0d3f571bfdc11
```

We can use the WWIDs to identify the four device paths to each of the LUNs presented from storage: the three LUNs targeted for web content space and one created for quorum.

Purpose	Size	Devices
web data	200gb	sdc, sdg, sdm, sdq
web data	40gb	sdd, sdh, sdn, sdr
web data	40gb	sde, sdi, sdo, sds
cluster quorum	1gb	sdb, sdk, sdl, sdu

Rather than guess at which of the four device names to use when referencing each of the LUNs, it is recommended to use the WWIDs and create dm-multipath aliases to associate with specific devices. The aliases can be used to create user friendly persistent names for purposes such as Oracle voting, CSS voting, and for ASM disks (if using ASM without ASMLib). This is accomplished by modifying the *multipaths* stanza in */etc/multipath.conf* to link the desired aliases. Examples for some common SAN storage configurations can be found in */usr/share/doc/device-mapper-multipath\*/multipath.conf.defaults*.

The *multipaths* stanza of the */etc/multipath.conf* file used in this document:

```
multipaths {
  multipath {
    wwid          360060160c914150142d0d3f571bfdc11
    alias         quorum
  }
  multipath {
    wwid          360060160541415013a6c98657610dd11
    alias         data1
  }
  multipath {
    wwid          36006016054141500183f857db2d8dc11
    alias         data2
  }
  multipath {
    wwid          36006016054141500d0ce586cb2d8dc11
    alias         data3
  }
}
```



After establishing the configuration above, perform a dry run to evaluate the configuration:

```
# multipath -v2 -d
```

Start the multipath daemon using the following commands:

```
# chkconfig multipathd on
# service multipathd start
```

DM-Multipath creates a single device that reroutes I/O to those four underlying devices according to the multipath configuration. It creates kernel block devices (*/dev/dm-\**) and corresponding block devices (with persistent names) in the */dev/mapper* directory.

Note that the */dev/dm-\** devices are for internal use only and should not be used in cluster configuration.

The default naming convention of the block devices is */dev/mapper/mpath\**. Since aliases were defined in the *multipath.conf* file, they supersede the default names, are consistent across all nodes in the cluster and should now be visible as block devices in */dev/mapper*.

```
# ls -l /dev/mapper/
total 0
drwxr-xr-x  2 root root    240 Nov 12 14:07 ./
drwxr-xr-x 16 root root   5060 Nov 13 13:47 ../
crw-----  1 root root   10, 63 Nov 12 14:07 control
brw-rw----  1 root disk 253,  3 Nov 12 14:07 data1
brw-rw----  1 root disk 253,  5 Nov 12 14:07 data2
brw-rw----  1 root disk 253,  4 Nov 12 14:07 data3
brw-rw----  1 root disk 253,  2 Nov 12 14:07 quorum
brw-rw----  1 root disk 253,  0 Nov 12 14:07 VolGroup00-LogVol100
brw-rw----  1 root disk 253,  1 Nov 12 14:07 VolGroup00-LogVol101
brw-rw----  1 root disk 253,  8 Nov 12 14:07 www_vg-www_lv
```

The `multipath` command, at its most verbose level, will list all related information when searching for any connectivity issues.

```
# multipath -v3
...
```

To force a re-reading of the partition table and update the dm-multipath devices, use `partprobe` and `multipath`.

```
# partprobe
# multipath -F
# multipath -v3
...
```

To view the status of the individual paths to each multipathed device:

```
# multipath -ll
quorum (360060160c914150142d0d3f571bfdc11) dm-2 DGC,RAID 10
```



```
[size=1.0G][features=1 queue_if_no_path][hwhandler=1 emc]
\_ round-robin 0 [prio=2][active]
  \_ 1:0:0:0 sdb 8:16 [active][ready]
  \_ 2:0:0:0 sdl 8:176 [active][ready]
\_ round-robin 0 [prio=0][enabled]
  \_ 1:0:3:0 sdk 8:160 [active][ready]
  \_ 2:0:3:0 sdu 65:64 [active][ready]
data3 (36006016054141500d0ce586cb2d8dc11) dm-4 DGC,RAID 10
[size=40G][features=1 queue_if_no_path][hwhandler=1 emc]
\_ round-robin 0 [prio=2][active]
  \_ 1:0:2:1 sdh 8:112 [active][ready]
  \_ 2:0:2:1 sdr 65:16 [active][ready]
\_ round-robin 0 [prio=0][enabled]
  \_ 1:0:1:1 sdd 8:48 [active][ready]
  \_ 2:0:1:1 sdn 8:208 [active][ready]
data2 (36006016054141500183f857db2d8dc11) dm-5 DGC,RAID 10
[size=40G][features=1 queue_if_no_path][hwhandler=1 emc]
\_ round-robin 0 [prio=2][active]
  \_ 1:0:2:2 sdi 8:128 [active][ready]
  \_ 2:0:2:2 sds 65:32 [active][ready]
\_ round-robin 0 [prio=0][enabled]
  \_ 1:0:1:2 sde 8:64 [active][ready]
  \_ 2:0:1:2 sdo 8:224 [active][ready]
data1 (360060160541415013a6c98657610dd11) dm-3 DGC,RAID 10
[size=200G][features=1 queue_if_no_path][hwhandler=1 emc]
\_ round-robin 0 [prio=2][active]
  \_ 1:0:2:0 sdg 8:96 [active][ready]
  \_ 2:0:2:0 sdq 65:0 [active][ready]
\_ round-robin 0 [prio=0][enabled]
  \_ 1:0:1:0 sdc 8:32 [active][ready]
  \_ 2:0:1:0 sdm 8:192 [active][ready]
```

Above we note that each of the devices created for our use (data1, data2, data3, quorum) each display their four ready paths, two active and two passive (enabled).

The *multipath.conf* file can also be used to set storage specific attributes. These multipath specific settings are usually obtained from the storage vendor and typically supersede the default settings. Refer to Appendix B in this document for the device settings used for the EMC CLARiiON storage array used in this document.

The information in this section is a small subset of the functionality provided by the *multipath.conf* file. For information regarding

- the configuration of device identifiers and attributes
- storage array support
- ignoring local disks when generating multipath devices
- blacklisting specific devices
- multipath / dmsetup commands and queries

and more, please reference the Using Device-Mapper Multipath guide.





## 6.2 CLVM

The Clustered Logical Volume Manager (CLVM) is a set of LVM clustering extensions that allow a cluster to manage shared storage.

The `clvmd` daemon is the primary clustering extension to LVM. This daemon runs on each cluster member and distributes LVM metadata updates in a cluster, presenting each member with the identical view of logical volumes.

Please note that shared storage for use in Red Hat Cluster Suite requires the user be running the clustered logical volume manager daemon (`clvmd`) or the High-Availability Logical Volume Management agents (HA-LVM). If you are not able to use either the `clvmd` daemon or HA-LVM for operational reasons or because you do not have the correct entitlements, you must not use single-instance LVM on shared storage as this may result in data corruption. Please contact your Red Hat service representative if you have any concerns.

### 6.2.1 Logical Volumes

For our purposes, the three storage LUNs (previously created for housing web content) will be used to create a single logical volume, accessible by the hosts. These devices will need to be initialized accordingly in order for `conga` to recognize them as candidates for CLVM use.

**Note:** The cluster quorum disk should be excluded from this procedure as this would overwrite the quorum label assigned to the device.

Initialize the three devices that will be used for storing web content as follows.

```
# pvcreate /dev/mapper/data1 /dev/mapper/data2 /dev/mapper/data3
Physical volume "/dev/mapper/data1" successfully created
Physical volume "/dev/mapper/data2" successfully created
Physical volume "/dev/mapper/data3" successfully created
```

In `luci`, click on the blue *Storage* tab at the top of the page to view a list of the systems currently under `conga` control as well as the default storage configuration preferences.

Click on one of the cluster member names to have `luci` probe that system and list all of its currently discovered devices, their sizes and SCSI identifiers.

[homebase](#)[cluster](#)[storage](#)[storage](#)[System List](#)[et-virt08-ic.lab.bos.redhat.com](#)[Hard Drives](#)[Partition Tables](#)[Volume Groups](#)

## et-virt08-ic.lab.bos.redhat.com

[View recent log activity](#)

### Hard Drives

- ▶ [sda](#) 68.36 GB, SC SI ID = 20010b91c080eda3b
- ▶ [sdb](#) 1.0 GB, SC SI ID = 360060160c914150142d0d31571b1dc11
- ▶ [sdc](#) 200.0 GB, SC SI ID = 360060160541415013a6c98657610dd11
- ▶ [sdd](#) 40.0 GB, SC SI ID = 36006016054141500d0ce586cb2d8dc11
- ▶ [sde](#) 40.0 GB, SC SI ID = 360060160541415001831857db2d8dc11
- ▶ [sdf](#) 9.0 GB, SC SI ID = 36006016054141501266764dc3c02dd11
- ▶ [sdg](#) 200.0 GB, SC SI ID = 360060160541415013a6c98657610dd11
- ▶ [sdh](#) 40.0 GB, SC SI ID = 36006016054141500d0ce586cb2d8dc11
- ▶ [sdi](#) 40.0 GB, SC SI ID = 360060160541415001831857db2d8dc11
- ▶ [sdj](#) 9.0 GB, SC SI ID = 36006016054141501266764dc3c02dd11
- ▶ [sdk](#) 1.0 GB, SC SI ID = 360060160c914150142d0d31571b1dc11
- ▶ [sdl](#) 1.0 GB, SC SI ID = 360060160c914150142d0d31571b1dc11
- ▶ [sdm](#) 200.0 GB, SC SI ID = 360060160541415013a6c98657610dd11
- ▶ [sdn](#) 40.0 GB, SC SI ID = 36006016054141500d0ce586cb2d8dc11
- ▶ [sdo](#) 40.0 GB, SC SI ID = 360060160541415001831857db2d8dc11
- ▶ [sdp](#) 9.0 GB, SC SI ID = 36006016054141501266764dc3c02dd11
- ▶ [sdq](#) 200.0 GB, SC SI ID = 360060160541415013a6c98657610dd11
- ▶ [sdr](#) 40.0 GB, SC SI ID = 36006016054141500d0ce586cb2d8dc11
- ▶ [sds](#) 40.0 GB, SC SI ID = 360060160541415001831857db2d8dc11
- ▶ [sdt](#) 9.0 GB, SC SI ID = 36006016054141501266764dc3c02dd11
- ▶ [sdu](#) 1.0 GB, SC SI ID = 360060160c914150142d0d31571b1dc11

### Partition Tables

- ▶ [sda](#)

### Volume Groups

- ▶ [VolGroup00](#)
- ▶ [www\\_vg](#)

[Reprobe Storage](#)

Click on *Volume Groups* in the left menu to display the currently defined cluster logical volume groups for that member.

Click on *New Volume Group* to display a list of available devices that have been initialized for CLVM use.



homebase cluster storage help log out

storage

et-virt08-ic.lab.bos.redhat.com

System List

et-virt08-ic.lab.bos.redhat.com

Hard Drives

Partition Tables

Volume Groups

New Volume Group

VolGroup00

Reprobe Storage

### Creating New Volume Group

Volume Group Name:

Extent Size:

Clustered:

#### Select 1 to 4 Physical Volumes

- /dev/mapper/data1 (200.0 GB - Hard Drive)
- /dev/mapper/data2 (40.0 GB - Hard Drive)
- /dev/mapper/data3 (40.0 GB - Hard Drive)
- /dev/sdab (814.03 GB - Hard Drive)

Reset Create

If no available devices are listed, then the `pvc create` command has yet to be executed to ready the desired devices.

1. Enter the desired *Volume Group Name*.
2. Set the *Extent Size* if the default value is not preferred.
3. Leave the *Clustered* option set to true.
4. Choose the devices from the list of physical volumes available.
5. Click the *Create* button.

The above example will create a volume group named `www_vg` and display a graphical view of the physical volumes that comprise the group. Once the volume group is created, CLVM will provide a common name across the systems which persists across reboots.

Verify that the volume group exists on both members using the `vgs` command.

```
# vgs
VG          #PV #LV #SN Attr   VSize  VFree
VolGroup00  1   2   0 wz--n- 68.25G  0
www_vg      3   1   0 wz--nc 279.99G 1012.00M
```

In `luci`, clicking on any one of the sections in red will display details regarding that physical volume.



storage

et-virt08-ic.lab.bos.redhat.com

System List

et-virt08-ic.lab.bos.redhat.com

Hard Drives

Partition Tables

Volume Groups

VolGroup00

www\_vg

**Volume Group www\_vg**

Graphical View (Uncheck if volumes are too small to select)

Logical Volumes:

Physical Volumes:

Click cylinders to view properties, **mapper/data1** view Volume Group's properties

**Hard Drive 'mapper/data1' - /dev/mapper/data1**

[Go to Hard Drives view](#)

Model	unknown
Vendor	unknown
Size	200.0 GB
Type	ide

**Content: Physical Volume**

Volume Group Name	www_vg
Unused	199.99 GB
Size	199.99 GB
Extent Size	4.0 MB
Attributes	a-
Format	lvm2
UUID	Fgm0G5-bwXa-qaHX-2Z8Y-ivfe-yRCJ-YMBj5m

Reprobe Storage

Reset Apply

Click on the blue bar representing Logical Volumes (presently listed as *unused space*) to display the fields necessary for creating a new logical volume on the newly created volume group.



homebase cluster storage

storage

System List

et-virt08-ic.lab.bos.redhat.com

Hard Drives

Partition Tables

Volume Groups

VolGroup00


www\_vg

et-virt08-ic.lab.bos.redhat.com


**Volume Group www\_vg**

Graphical View (Uncheck if volumes are too small to select)

**Logical Volumes:**



**Physical Volumes:**



Click cylinders to view properties, unselect all to view Volume Group's

**Unused Space - Creating New Logical Volume**

Logical Volume Name	<input type="text" value="www_lv"/>	<b>Content</b>	<input type="text" value="Empty"/>
Volume Group Name	www_vg		
Size	<input type="text" value="279.98"/> (0.00 - 279.98) GB		
Clustered	true		

Reset Create

Reprobe Storage

1. Enter a *Logical Volume* name
2. Select the GFS2 option from the *Content* pull-down menu



storage

et-virt08-ic.lab.bos.redhat.com

System List

et-virt08-ic.lab.bos.redhat.com

Hard Drives

Partition Tables

Volume Groups

VolGroup00

www\_vg

**Volume Group www\_vg**

Graphical View (Uncheck if volumes are too small to select)

Logical Volumes:

Physical Volumes:

Click cylinders to view properties, unselect all to view Volume Group's

**Unused Space - Creating New Logical Volume**

Logical Volume Name	<input type="text" value="www_lv"/>
Volume Group Name	www_vg
Size	<input type="text" value="279.98"/> (0.00 - 279.98) GB
Clustered	true

**Content**

- Empty
- Linux Extended FS
- GFS1 - Global FS v.1
- GFS2 - Global FS v.2
- Swap

Reset Create

Reprobe Storage

Depending on the option selected in the *Content* pull-down menu, additional fields will be displayed for providing the necessary information to create the file system of choice.



storage

System List

et-virt08-ic.lab.bos.redhat.com

Hard Drives

Partition Tables

Volume Groups

VolGroup00

www\_vg

et-virt08-ic.lab.bos.redhat.com

Volume Group **www\_vg**

Graphical View (Uncheck if volumes are too small to select)

Logical Volumes:

Physical Volumes:

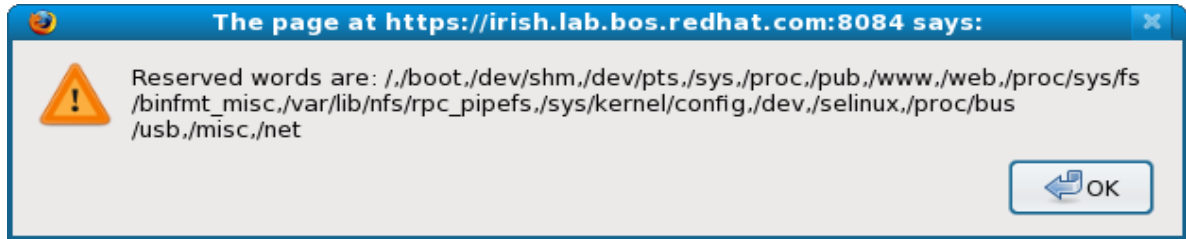
Click cylinders to view properties, unselect all to view Volume Group's

### Unused Space - Creating New Logical Volume

Logical Volume Name	<input type="text" value="www_lv"/>	Content	<input type="text" value="GFS2 - Global FS v.2"/>
Volume Group Name	www_vg	Cluster Name	webserver1
Size	<input type="text" value="279.98"/> (0.00 - 279.98) GB	Unique GFS Name	<input type="text" value="www_gfs"/>
Clustered	true	Journal Size	<input type="text" value="33554432"/>
		Mountpoint	<input type="text" value="/web_content"/>
		Mount	<input type="text" value="true"/>
		List in /etc/fstab	<input type="text" value="true"/>
		Mountable	true
		Locking Protocol	dlm
		Number of Journals	<input type="text" value="3"/> (1 - 128)
		Clustered	<input type="text" value="true"/>

Reset Create

1. Ensure that the *Cluster Name* matches the cluster on which this file system is to be used.
2. Enter a *Unique GFS Name* (typically a logical reference to its purpose or function).
3. If the user has reason the adjust the *Journal Size* from the default value, that can be entered at this time.
4. Provide the absolute path to an existing mount point where the new file system is to be mounted. The following directories names are reserved. Luci will produce an error if any of the following are entered in this field.



5. The *Mount* and *List in /etc/fstab* options will depend on whether the file system is to be a cluster resource. If so, then select 'false' for both options since the cluster service (to which the file system will be associated) will determine when and on which member it should be mounted. For our purposes, the file system will not be used as a cluster resource and will instead have an entry in */etc/fstab* so it is mounted by all members at startup. This provides faster failover of the web service since the cluster will not have to wait for the file system to migrate to another member before restarting or relocating the web service.

Leave both options set to 'true'.

6. Enter a preferred *Number of Journals*. GFS2 uses a journal for file system crash recovery. This amount should be at least one journal for each member that may mount the file system. Conga is already aware that there are two cluster members, so it defaults to two journals. A general rule is to allow for one more journal than there are cluster members. However, journal addition is one of the newer features of GFS2 so more journals can be added later if necessary.
7. Leave the *Clustered* option set to 'true'.
8. Click the *Create* button.
9. Once the procedure completes, verify that the logical volume exists on both members using `lvs`.

```
# lvs
LV          VG          Attr   LSize   Origin Snap%   Move Log Copy%  Convert
LogVol100  VolGroup00 -wi-ao 66.31G
LogVol101  VolGroup00 -wi-ao 1.94G
www_lv     www_vg      -wi-ao 279.00G
```

10. Verify that the newly created volume was successfully mounted and is accessible from any cluster member using `mount` or `df`.

```
# df -h | grep web_content
279G 109M 279G 1% /web_content
```

11. Verify that an entry has been made in the */etc/fstab* file on the member used by `luci` to ensure that volume will mount after reboot. For instance,

```
# grep web_content /etc/fstab
/dev/www_vg/www_lv /web_content gfs2 defaults 0 0

# mount | grep web_content
/dev/mapper/www_vg-wwww_lv on /web_content type gfs2
```





```
(rw,hostdata=jid=1:id=196609:first=0)
```

12. The `/etc/fstab` file on the other cluster member will have to be updated and the file system mounted there. This can be accomplished in one of two ways.

- On the other member, manually edit the `/etc/fstab` file, restart `clvmd`, and mount the filesystem.

... OR ...

- Use `luci` to connect to other node. Navigate to the logical volume just created by clicking the `storage` tab, selecting the other node name, selecting `www_vg` under Volume Groups and clicking on the blue bar representing the logical volume `www_lv`. Fill in the mount point info as shown below.

The screenshot shows the Red Hat storage management interface. At the top, there are tabs for 'homebase', 'cluster', and 'storage'. The 'storage' tab is active. On the left, there is a sidebar with a 'System List' and a list of nodes. The selected node is 'et-virt09-ic.lab.bos.redhat.com'. Below the node name, there are options for 'Hard Drives', 'Partition Tables', 'Volume Groups', and 'VolGroup00'. The 'www\_vg' volume group is selected. The main area shows the configuration for the 'www\_vg' volume group. It includes a 'Graphical View' checkbox which is checked. Below this, there are 'Logical Volumes' and 'Physical Volumes' represented by cylinders. The 'Logical Volume 'www\_lv' - /dev/www\_vg/www\_lv' is selected. The configuration details for this logical volume are shown in a table:

Logical Volume Name	www_lv
Volume Group Name	www_vg
Extent Size	4.0 MB
Size	279.0 GB
Mirrored	false
Attributes	-wi-ao
Clustered	true
Snapshots	
UUID	gfAdbK-HeuT-qh8w-VGsv-01vc-sa6g-v8fHcM

Below the table, there is a 'Content' section with a dropdown menu set to 'GFS2 - Global FS v.2'. The configuration details for the content are shown in a table:

Cluster Name	webserver1
Unique GFS Name	www_gfs
Block Size	4.0 KB
Mountpoint	/web_content
/etc/fstab Mountpoint	/web_content
Mountable	true
Locking Protocol	dlm
Clustered	true

At the bottom of the configuration panel, there are buttons for 'Remove', 'Reset', and 'Apply'. Below the configuration panel, there is a 'Reprobe Storage' button.



## 6.2.2 Configuration

There are a few modifications to make to the CLVM configuration file, `/etc/lvm/lvm.conf`, to accommodate a clustered environment.

1. If several entries in the scanned directories correspond to the same block device, all the paths are matched against any item in the list of regular expressions in the `preferred_names` setting. To have LVM scan the preferred devices only, the following entry was included in the `devices` section of the configuration file:

```
preferred_names = [ "^/dev/mapper/" ]
```

If multiple expressions are supplied, the first match is used.

2. When creating an LVM volume with multiple paths to active/passive storage arrays, the configuration file should include filters that exclude the physical disks used to comprise the underlying devices. A filter instructs LVM to use only a restricted set of devices. Otherwise, if the storage array switches from the active path to the passive path when it receives I/O, multipath will failover (and failback) any time LVM scans the passive path. If the storage array requires a command to activate the passive path, then LVM will print a warning message when this occurs.

The following setting was used in the `devices` section of the configuration file:

```
filter = [ "r|/dev/dm-.*|", "r|/dev/sd.*|", "a|/dev/mapper/.*/", "a./.*/" ]
```

3. By default, LVM will not recognize device mapper devices as physical volumes. To account for this, include this line in the `devices` section of the configuration file:

```
types = [ "device-mapper", 16 ]
```

4. All changes to logical volumes and their states are communicated using locks. LVM defaults to local file-based locking and needs to instead use built-in cluster-wide locking. LVM commands are instructed to communicate with the CLVM daemon (`clvmd`) using the following lines in the `global` section of the configuration file:

```
locking_type = 3
```

Please reference the LVM Administrator's Guide for more detailed information on using LVM in a cluster.

## 6.3 `/etc/hosts/`

The `/etc/hosts` file for each cluster member should contain an entry defining localhost. If the external hostname of the system is defined on the same line, the hostname reference should be removed.

For instance, upon examining the `/etc/hosts` file after installation, the localhost entry included the external hostname as well as the localhost definition.

```
127.0.0.1 et-virt08.lab.bos.redhat.com et-virt08 localhost.localdomain localhost
```



This will generate an error when attempting to start the cluster manager (cman) service on this node. Change the localhost entry to resemble the example below.

```
127.0.0.1 localhost.localdomain localhost
```

Additionally, each `/etc/hosts` file should define the local interconnect of each cluster member. In the excerpt example below, local IP addresses are defined for the each system in the cluster.

```
10.10.1.1 et-virt08-ic.lab.bos.redhat.com et-virt08-ic
10.10.1.2 et-virt09-ic.lab.bos.redhat.com et-virt09-ic
```

Note that each local interconnect must have a unique name and for simplicity in this case, were named by appending `'-ic'` to the end of the hostname. The same definitions must exist on both nodes.

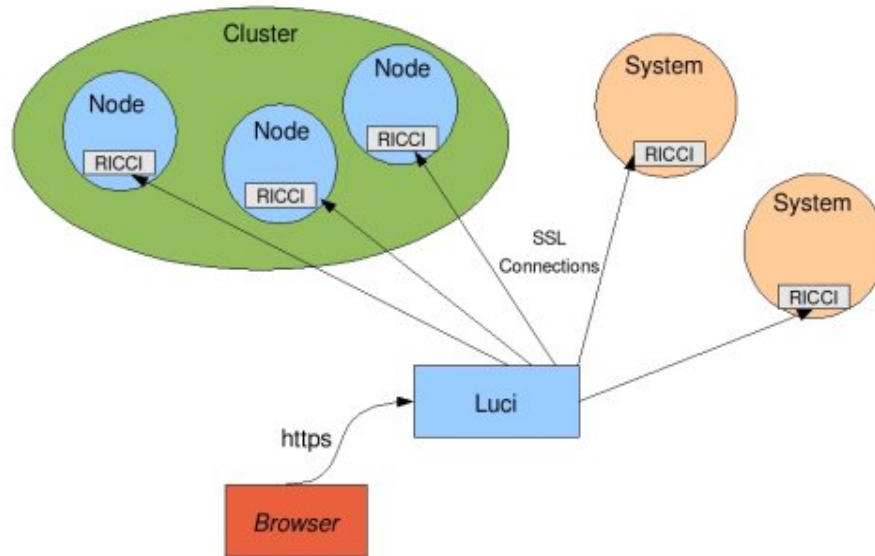
## 7 Conga

Conga was designed as an HTML graphical interface for creating and managing clusters built using Red Hat Cluster Suite software. It is built as an agent/server architecture for remote administration of systems as well as a method for managing sophisticated storage configurations.

Another cluster configuration tool (Cluster Admin GUI) can be started from the command line using the `system-config-cluster` command. While this GUI provides several convenient configuration/management tools, Conga is a more recent and comprehensive tool that provides added convenience and flexibility.

An instance of the agent component (ricci) resides on each cluster member while a single server running (luci) can communicate with multiple ricci agents installed on different systems to maintain a database of cluster specific node and user information. The luci server is accessed via a secure web page and will need to be on the same private network as the systems it will manage. One luci server can manage multiple clusters.

The diagram below illustrates Conga's architecture.



Although it is highly recommended that it reside on system external to those it manages, luci can run on a cluster member. There are obvious advantages to having a separate luci server, namely the ability to manipulate or fence cluster members without interrupting its own functionality.

## 7.1 Installing ricci

If the Clustering package group was included at the time of OS installation, then ricci is already present on the servers. If not, see the *Modifying RHN Subscriptions* section in this document for instructions on subscribing to the appropriate software update channels to support the installation of clustering software using yum. Once the user's RHN subscriptions have been established, the installation of ricci on each intended cluster member can proceed as follows.

```
# yum install ricci
Loading "rhnplugin" plugin
Loading "security" plugin
rhel-x86_64-server-cluste 100% |=====| 1.4 kB 00:00
rhel-x86_64-server-cluste 100% |=====| 1.4 kB 00:00
rhn-tools-rhel-x86_64-ser 100% |=====| 1.2 kB 00:00
rhel-x86_64-server-5      100% |=====| 1.4 kB 00:00
...
```

Note in the above output that the installer is searching multiple channels for the package(s) requested for installation. If the only channel listed in the output is rhel-x86\_64-server-5, then the required RHN subscriptions have not been setup.

As with all packages installed using yum, the installer will setup the installation process, parse the assorted installation arguments for each package, and resolve any and all dependencies. The complete list of requested packages ready for installation is then displayed along with the list of packages necessary to resolve dependencies.



```

=====
Package                Arch      Version      Repository      Size
=====
Installing:
ricci                   x86_64     0.12.0-7.el5  rhel-x86_64-server-
cluster-5 1.1 M
Installing for dependencies:
cman                    x86_64     2.0.84-2.el5  rhel-x86_64-server-5 648 k
modcluster              x86_64     0.12.0-7.el5  rhel-x86_64-server-
cluster-5 331 k
oddjob                  x86_64     0.27-9.el5    rhel-x86_64-server-5 60 k
oddjob-libs             x86_64     0.27-9.el5    rhel-x86_64-server-5 44 k
openais                 x86_64     0.80.3-15.el5 rhel-x86_64-server-5 375 k
perl-Net-Telnet         noarch     3.03-5        rhel-x86_64-server-5 56 k
perl-XML-LibXML         x86_64     1.58-5        rhel-x86_64-server-5 230 k
perl-XML-LibXML-Common x86_64     0.13-8.2.2    rhel-x86_64-server-5 16 k
perl-XML-Namespacesupport noarch     1.09-1.2.1    rhel-x86_64-server-5
15 k
perl-XML-SAX            noarch     0.14-5        rhel-x86_64-server-5 75 k

Transaction Summary
=====
Install      11 Package(s)
Update       0 Package(s)
Remove       0 Package(s)

```

The total download size is provided and the user is prompted to continue. Enter 'y' and proceed.

```

Total download size: 2.9 M
Is this ok [y/N]: y
Downloading Packages:
(1/11): oddjob-libs-0.27- 100% |=====| 44 kB 00:00
(2/11): ricci-0.12.0-7.el 100% |=====| 1.1 MB 00:01
(3/11): perl-XML-SAX-0.14 100% |=====| 75 kB 00:00
(4/11): perl-XML-Namespac 100% |=====| 15 kB 00:00
(5/11): perl-XML-LibXML-C 100% |=====| 16 kB 00:00
(6/11): perl-XML-LibXML-1 100% |=====| 230 kB 00:00
(7/11): perl-Net-Telnet-3 100% |=====| 56 kB 00:00
(8/11): modcluster-0.12.0 100% |=====| 331 kB 00:00
(9/11): cman-2.0.84-2.el5 100% |=====| 648 kB 00:00
(10/11): oddjob-0.27-9.el 100% |=====| 60 kB 00:00
(11/11): openais-0.80.3-1 100% |=====| 375 kB 00:00
warning: rpmts_HdrFromFdno: Header V3 DSA signature: NOKEY, key ID 37017186
Importing GPG key 0x37017186 "Red Hat, Inc. (release key) <security@redhat.com>"
from /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release

```

After downloading all the required packages, the user is again prompted to continue. Enter 'y' and proceed. Yum will list the installed packages as well as the additional dependencies.

```

Is this ok [y/N]: y
Running rpm_check_debug

```



```

Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing: perl-XML-LibXML-Common          ##### [ 1/11]
  Installing: openais                        ##### [ 2/11]
  Installing: perl-XML-Namespacesupport     ##### [ 3/11]
  Installing: perl-XML-SAX                  ##### [ 4/11]
  Installing: perl-XML-LibXML               ##### [ 5/11]
could not find ParserDetails.ini in /usr/lib/perl5/vendor_perl/5.8.8/XML/SAX
  Installing: perl-Net-Telnet                ##### [ 6/11]
  Installing: cman                          ##### [ 7/11]
  Installing: oddjob                        ##### [ 8/11]
  Installing: modcluster                    ##### [ 9/11]
  Installing: ricci                        ##### [10/11]
  Installing: oddjob-libs                   ##### [11/11]

Installed: ricci.x86_64 0:0.12.0-7.el5
Dependency Installed: cman.x86_64 0:2.0.84-2.el5 modcluster.x86_64
0:0.12.0-7.el5 oddjob.x86_64 0:0.27-9.el5 oddjob-libs.x86_64 0:0.27-9.el5
openais.x86_64 0:0.80.3-15.el5 perl-Net-Telnet.noarch 0:3.03-5 perl-XML-
LibXML.x86_64 0:1.58-5 perl-XML-LibXML-Common.x86_64 0:0.13-8.2.2 perl-XML-
NamespaceSupport.noarch 0:1.09-1.2.1 perl-XML-SAX.noarch 0:0.14-5
Complete!

```

## 7.2 Installing luci

The server running luci needs to reside on the same local network as the systems it will manage. For obvious reasons, although luci can be run on a cluster member it is highly recommended that it run on a non clustered server to avoid situations where the server running luci is rebooted or fenced.

If the Clustering package group was included at the time of OS installation to the server intended to run luci, then luci is already present on the server. If not, see the *Modifying RHN Subscriptions* section in this document for instructions on subscribing to the appropriate software update channel to support the installation of luci using yum.

Once the user's RHN subscriptions have been established, the installation of luci on the target https server can proceed.

```

# yum install luci
Loading "rhnplugin" plugin
Loading "security" plugin
rhel-x86_64-server-cluste 100% |=====| 1.4 kB 00:00
rhel-x86_64-server-cluste 100% |=====| 1.4 kB 00:00
rhn-tools-rhel-x86_64-ser 100% |=====| 1.2 kB 00:00
rhel-x86_64-server-5      100% |=====| 1.4 kB 00:00
...

```

The installer will setup the installation process, parse the assorted installation arguments for each package, and resolve any and all dependencies. The complete list of requested packages ready for installation is then displayed along with the list of packages necessary to



resolve dependencies.

```

=====
Package                Arch      Version      Repository      Size
=====
Installing:
  luci                  x86_64     0.12.0-7.el5  rhel-x86_64-server-
cluster-5             27 M
Installing for dependencies:
  python-imaging       x86_64     1.1.5-5.el5   rhel-x86_64-server-5 408 k
  tix                   x86_64     1:8.4.0-11.fc6 rhel-x86_64-server-5 333 k
  tkinter              x86_64     2.4.3-21.el5  rhel-x86_64-server-5 281 k

Transaction Summary
=====
Install      4 Package(s)
Update      0 Package(s)
Remove      0 Package(s)

Total download size: 28 M

```

The total download size is provided and the user is prompted to continue. Enter 'y' and proceed.

```

Is this ok [y/N]: y
Downloading Packages:
(1/4): python-imaging-1.1 100% |=====| 408 kB    00:00
(2/4): tkinter-2.4.3-21.e 100% |=====| 281 kB    00:00
(3/4): luci-0.12.0-7.el5. 100% |=====| 27 MB     00:11
(4/4): tix-8.4.0-11.fc6.x 100% |=====| 333 kB    00:00
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing: tix                    ##### [1/4]
  Installing: tkinter                ##### [2/4]
  Installing: python-imaging         ##### [3/4]
  Installing: luci                   ##### [4/4]

Installed: luci.x86_64 0:0.12.0-7.el5
Dependency Installed: python-imaging.x86_64 0:1.1.5-5.el5 tix.x86_64
1:8.4.0-11.fc6 tkinter.x86_64 0:2.4.3-21.el5
Complete!
#

```

## 8 Clustering

Now that the clustering software is present on the targeted cluster nodes and the luci server, the clustering agent and server modules can be engaged.

Start the ricci service on each server that will join the cluster.



```
# service ricci start
Starting oddjobd: [ OK ]
generating SSL certificates... done
Starting ricci: [ OK ]
```

At the remote server on which luci was installed, an administrative password must be set for luci using `luci_admin` before the service can be started.

```
# luci_admin init
Initializing the luci server

Creating the 'admin' user

Enter password: <enter password>
Confirm password: <re-enter password>

Please wait...
The admin password has been successfully set.
Generating SSL certificates...
The luci server has been successfully initialized

You must restart the luci server for changes to take effect.

Run "service luci restart" to do so
```

`luci_admin password` can be run at any time to change the luci administrative password initially set above.

Start the luci service.

```
# service luci restart
Shutting down luci: [ OK ]
Starting luci: Generating https SSL certificates... done [ OK ]

Point your web browser to https://<luci_servername>:8084 to access luci
```

The first time luci is accessed via a web browser at `https://<luci_servername>:8084`, the user will need to accept two SSL certificates before being directed to the login page.





redhat.

CLUSTER AND  
STORAGE SYSTEMS

homebase

cluster

storage

admin

Add a System

Add an Existing

Cluster

Manage Systems

Add a User

## Luci Homebase

Welcome to Luci, admin.

Select an action from the list on the left.

The Conga Cluster and Storage Management System is Copyright © 2000–2008 Red Hat, Inc.  
Distributed under the GNU GPL license.

Enter the login name and chosen password to view the Luci Homebase page.

## 8.1 Cluster Creation

In the Luci Homebase page, click on the *cluster* tab at the top of the page and then on *Create a New Cluster* from the menubar on left. In the cluster creation window, enter the preferred name for the cluster (15 char max), the host names assigned to the local interconnect of each server and their root passwords.

This window also provides options to:

- use the clustering software already present on the system or download the required packages
- enable shared storage support
- reboot the systems prior to joining the new cluster
- check to verify that system passwords are identical
- view the SSL certification fingerprints of each server









homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure

### Create a new cluster

Cluster Name

Node Hostname	Root Password	Key ID
<input type="text" value="et-virt08.lab.bos.redhat.com"/>	<input type="password" value="....."/>	 
<input type="text" value="et-virt09.lab.bos.redhat.com"/>	<input type="password" value="....."/>	 
<input type="text"/>	<input type="password"/>	 

Download packages  
 Use locally installed packages.  
 Enable Shared Storage Support  
 Reboot nodes before joining cluster  
 Check if node passwords are identical.

it is highly recommended to use the interconnect names (or their IP addresses) when building the cluster. This will require that the luci server also have a connection to the private LAN and will remove any possibilities of public IO traffic interfering with the cluster activities. This will ensure the cluster traffic is kept to the private interconnect rather than the public LAN for the cluster's inter-node communication and heartbeats.

Select the *Download packages* check box if you prefer to let conga do this for you. If all the necessary packages were installed as directed earlier in this document, this check box should not be necessary, however, it can do no harm and will ensure that all the required packages are present on each system.

Be sure to select the *Enable Shared Storage Support* check box.

Selecting the button to *View SSL cert fingerprints* is a convenient method of verifying that the secure shell configuration allows the server running luci to administer the servers to be clustered before attempting to create the cluster itself. If all is configured correctly, luci should successfully read the fingerprints of the target servers as seen in the above example. The default options will suffice for the purpose of this document.

Once the preferred options have been selected, click the *Submit* button to download (if selected) and install the cluster software packages onto each node, create the cluster configuration file, propagate the file to each cluster member, and start the cluster.




homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure
- haws
- webserver1


### webserver1

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



Creating node "et-virt09-ic.lab.bos.redhat.com" for cluster "webserver1"  
*Node still being created*

**Install** **Reboot** **Configure** **Join**




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

---

Creating node "et-virt08-ic.lab.bos.redhat.com" for cluster "webserver1"  
*Node still being created*

**Install** **Reboot** **Configure** **Join**



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

This will then display the main configuration window for the newly created cluster.

homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure

webserver1

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

### webserver1

General Fence Multicast Quorum Partition

**General Properties**

Cluster Name

Configuration Version

► Show advanced cluster properties

The *General* tab (shown above) displays cluster name and provides a method for modifying the configuration version and advanced cluster properties. The *Configuration Version* is set to 1 by default and is automatically incremented each time the cluster configuration is altered.



Click *Show advanced cluster properties* to view the list of advanced properties. Click any advanced property for online help about the property.

The *Fence* tab (shown below) will display the fence and XVM daemon properties window.

The screenshot shows the Red Hat Cluster Manager web interface. At the top, there are navigation tabs for 'homebase', 'cluster', and 'storage'. The 'cluster' tab is selected. On the left, there are two sidebars. The top sidebar is titled 'clusters' and contains links for 'Cluster List', 'Create a New Cluster', 'Configure', and a button labeled 'tim-haws'. The bottom sidebar is titled 'webserver1' and contains links for 'Nodes', 'Services', 'Resources', 'Failover Domains', and 'Shared Fence Devices'. The main content area is titled 'webserver1' and has sub-tabs for 'General', 'Fence', 'Multicast', and 'Quorum Partition'. The 'Fence' tab is active. It displays 'Fence Daemon Properties' with the following fields: 'Post Fail Delay' (input field with value 0), 'Post Join Delay' (input field with value 30), and 'Run XVM fence daemon' (checkbox, unchecked). Below this is the 'XVM fence daemon key distribution' section with two input fields: 'Enter a node hostname from the host cluster' and 'Enter a node hostname from the hosted (virtual) cluster'. There is a 'Retrieve cluster nodes' button and an 'Apply' button at the bottom.

*Post-Fail Delay* refers to the time (in seconds) the fence daemon will wait before fencing a node after the node has been deemed unreachable. The default value is 0 but can vary to accommodate network or cluster performance.

*Post-Join Delay* is the time (in seconds) the fence daemon will wait before fencing a node after the node joins the fence domain. When the cluster is quorate and the fence domain is first created, any nodes not yet in the cluster will be fenced. By default there is a delay of 3-6 seconds to allow any nodes unnecessarily flagged for fencing to join the cluster and avoid being fenced. This delay can be increased by setting `post_join_delay` in `cluster.conf`: The *Post-Join Delay* attribute was set to 30 seconds while default values were used for the other parameters. Set the *Post-Join Delay* value as preferred and click *Apply*.

The *Multicast* tab displays the multicast configuration window.



homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure

webserver1

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

webserver1

General Fence Multicast Quorum Partition

**Multicast Configuration**

Let cluster choose the multicast address

Specify the multicast address manually

Multicast address

Multicast network interface (optional)

Apply

The default option to *Let cluster choose the multicast address* is selected because Red Hat Cluster software chooses the multicast address for management communication across clustered nodes. If the user must use a specific multicast address, click *Specify the multicast address manually*, enter the address and click *Apply* for changes to take effect. Otherwise, leave the default selections alone.

The Quorum Partition tab displays the quorum partition configuration window.

### 8.1.1 Considering Quorum Disks

Although 2-node clusters are fairly common, quorum is handled in a slightly different manner than in clusters with greater than two members. Since the standard quorum algorithm ( $n/2 + 1$ , where  $n$  equals ) still equals two in this instance, quorum must still be maintained.. In this case, if one node fails the other simply takes over by itself.

The one area of concern with this configuration is that if a network disconnect occurs between the two nodes, each will think the other has failed. This is most commonly referred to as a “split brain” situation. Each node now assumes that it is the sole cluster member and will attempt to fence the other; typically via a reboot. Albeit a rare case, it is possible for this to occur. A quorum disk can be used to avoid this condition by bolstering the quorum. Additionally, one or more heuristics can be added to evaluate a specific node’s network connectivity and subsequently remove itself from the cluster if necessary. Common heuristics would include pinging a server’s own gateway router and/or the local NIC of the other cluster member. Quorum-disk parameters and heuristics depend on the complexity of the hardware configuration, site environment and any special requirements.

To understand the use of quorum disk parameters and heuristics, refer to the `qdisk(5)` man page.

Reference the *Considerations for Using Quorum Disk* and *Global Cluster Properties* sections of *Configuring and Managing a Red Hat Cluster* for further considerations regarding the use of a cluster quorum device.



Determine which device will act as the quorum disk or partition. Remember in the previous section on DM-Multipath, a 1gb LUN was created to act as the cluster quorum disk and the device WWID was used to assign the alias 'quorum' in the */etc/multipath.conf* file.

```
# ll /dev/mapper
total 0
drwxr-xr-x  2 root root    220 Sep 25 09:52 ./
drwxr-xr-x 14 root root   5220 Sep 25 09:53 ../
crw-----  1 root root  10, 63 Sep 25 09:52 control
brw-rw----  1 root disk 253,  5 Sep 25 09:52 data1
brw-rw----  1 root disk 253,  7 Sep 25 09:52 data2
brw-rw----  1 root disk 253,  6 Sep 25 09:52 data3
brw-rw----  1 root disk 253,  3 Sep 25 09:52 mpath1
brw-rw----  1 root disk 253,  4 Sep 25 09:52 mpath2
brw-rw----  1 root disk 253,  2 Sep 25 09:52 quorum
brw-rw----  1 root disk 253,  0 Sep 25 09:52 VolGroup00-LogVol100
brw-rw----  1 root disk 253,  1 Sep 25 09:52 VolGroup00-LogVol101
```

The `mkqdisk` command will create the quorum partition. Specify the device and a unique identifying label. `qdisk` was used as a label in the following example.

```
# mkqdisk -c /dev/mapper/quorum -l qdisk
mkqdisk v0.5.2
Writing new quorum disk label 'qdisk' to /dev/mapper/quorum.
WARNING: About to destroy all data on /dev/mapper/quorum; proceed [N/y] ? y
Warning: Initializing previously initialized partition
Initializing status block for node 1...
Initializing status block for node 2...
...
Initializing status block for node 16...
```

The label will then be referenced in the */etc/cluster/cluster.conf* file. The result can be checked using `mkqdisk -L` which will list all active paths to the quorum device.

```
# mkqdisk -L
mkqdisk v0.5.2
/dev/sdb:
    Magic:                eb7a62c2
    Label:                qdisk
    Created:              Wed Oct  1 13:07:40 2008
    Host:                 et-virt08.lab.bos.redhat.com
    Kernel Sector Size:  512
    Recorded Sector Size: 512

/dev/sdl:
    Magic:                eb7a62c2
    Label:                qdisk
    Created:              Wed Oct  1 13:07:40 2008
    Host:                 et-virt08.lab.bos.redhat.com
    Kernel Sector Size:  512
    Recorded Sector Size: 512
```

The label assigned is persistent after reboots so that clusters not using `dm-multipath`, or some



other means of LUN persistence, can still use a quorum device.

Now that appropriate label has been assigned to the quorum partition or disk, configure the newly labeled *qdisk* as the cluster quorum device.

1. Starting at the blue *cluster* tab at the top of the luci start page, click on the cluster name link, in this example 'webserver1'.
2. Click on the blue *Quorum Partition* tab on the cluster details information page.
3. Starting at the blue *cluster* tab at the top of the luci start page, click on the cluster name link, in this example 'webserver1'.
4. Click on the blue *Quorum Partition* tab on the cluster details information page.

The screenshot shows the RHCM web interface for cluster 'webserver1'. On the left is a navigation menu with 'clusters' and 'webserver1' sections. The 'webserver1' section is expanded, showing 'Nodes', 'Services', 'Resources', 'Failover Domains', and 'Shared Fence Devices'. The main content area is titled 'webserver1' and has tabs for 'General', 'Fence', 'Multicast', and 'Quorum Partition'. The 'Quorum Partition' tab is active, showing the 'Quorum Partition Configuration' section. It has two radio buttons: 'Do not use a Quorum Partition' (unselected) and 'Use a Quorum Partition' (selected). Below are input fields for 'Interval' (3), 'Votes' (1), 'TKO' (9), 'Minimum Score' (1), 'Device' (/dev/mapper/quorum), and 'Label' (qdisk). The 'Heuristics' section has a table with two rows of path-to-program, interval, and score. The first row is '/usr/bin/test -f /www/test.html' with interval 2 and score 1. The second row is '/bin/ping -c3 -t2 10.16.47.254' with interval 2 and score 1. There is an 'Add another heuristic' button and an 'Apply' button at the bottom.

5. Select the *Use a Quorum Partition* radio button
6. Use the table below and the following general rules for using *qdisk* with multiple paths to determine the preferred values for this page.

Interval	The number of seconds between read/write cycles (aka: heartbeats).
Votes	The number of votes the quorum daemon advertises to CMAN when it has a sufficient score.



TKO	The number of absent heartbeats a node must miss before being declared dead.
Minimum Score	The minimum score for a node to be considered alive. If omitted or set to zero, the default function $\lfloor \frac{(n+1)}{2} \rfloor$ is applied, where n is the sum of the heuristics scores. <b>The Minimum Score value must not exceed the sum of the heuristic scores</b> ; otherwise, the quorum disk cannot be available.
Device	The storage device the quorum daemon uses. The device name must be the same on all nodes.
Label	Specifies the quorum disk label created by the <code>mkqdisk</code> utility. If this field contains an entry, the label overrides the Device field. If this field is used, the quorum daemon reads <code>/proc/partitions</code> and checks for qdisk signatures on every block device found, comparing the label against the specified label. This is useful in configurations where multipathing, or LUN persistence, is not configured and the quorum device name differs among nodes.
Heuristics	<b>Path to Program:</b> [required] The program used to determine if this heuristic is alive. This can be anything that can be executed by <code>/bin/sh -c</code> . Only a return value of 0 indicates success.
	<b>Interval:</b> The frequency (in seconds) at which the heuristic is polled. The default interval for every heuristic is 2 seconds.
	<b>Score:</b> The weight of this heuristic. Exercise caution in determining scores for heuristics. The default score for each heuristic is 1.

## Quorum Timeouts

When using multiple paths to a cluster quorum device (for added availability), there are several settings that should be modified to allow the timing of each component (`multipathd`, `qdiskd`, and `cman`) sufficient time to determine if a failure has occurred before reacting to the timeout.

From Red Hat Knowledgebase article 13315, the following general ratios are recommended for each component:

- x = multipath failover time
- x \* 1.3 = qdisk failover time
- x \* 2.7 = cman failover time

Multipath failover time is controlled by the following settings in `/etc/multipath.conf`:

```
polling_interval 5
no_path_retry 3
```

These settings mean that multipath will check the path every five seconds and will retry





three times before failing it (total 20 seconds) ... or

$\text{multipath failover timeout} = \text{polling\_interval} * (\text{no\_path\_retry} + 1)$

The possible values for `no_path_retry` are 'queue' (never fail the path), 'fail' (immediately fail the path), or an integer  $n > 0$  (retry the path  $n$  times before failing it). 'fail' is not recommended for a multipathed qdisk.

qdisk failover time can be calculated using the *Interval* and *TKO* attributes, either from the *Quorum Partition* tab in the cluster details information page or from the `quorumd` tag in `/etc/cluster/cluster.conf`:

```
<quorumd interval="3" label="qdisk" min_score="1" tko="9" votes="1">
```

As described in the previous table, *Interval* is the number of seconds between read/write cycles and *TKO* determines how many cycles can fail before qdisk fails ... or

$\text{qdisk failover} = \text{interval} * \text{TKO}$

cman failover time is determined by the 'totem token' setting (in milliseconds) in `/etc/multipath.conf`:

```
<totem token="54000"/>
```

## 7. Enter any desired qdisk heuristics.

Heuristics are essentially The optional use of Quorum Disk provides even more robust protection for smaller configurations with shared storage.

Quorum Disk is a disk-based quorum daemon (`qdiskd`) that provides heuristics as a manner in which any one cluster member can determine if it is fit to be in the cluster. With heuristics, the user can determine factors that are important to the operation of the node in the event of a network problem. For example, in a four-node cluster with a 3 to 1 split, ordinarily, the three members "win" because of the three-to-one majority. Under those circumstances, the one member is fenced. With `qdiskd` however, you can set up heuristics that rely on other criteria, such as access to a critical resource (e.g. a critical network path). If the cluster requires additional methods of determining node health, then the user should configure quorum heuristics to address those needs.

In the above example, the first example heuristic verifies that the web content file system is accessible. The second and more common heuristic verifies successful access to the private interconnect switch for this cluster. Note that some network switches are not designed for remote management and as such would not possess an IP address to ping. Without the ability to test the nodes connection to the private network in this manner, alternatives can be applied. The next most common would be to check the link status on the network interface used for the cluster communications. If the link is up, then that member should be able to participate in cluster communications. This is easily checked using the `ethtool` command.

In this particular example however, the private interconnect was configured using



network bonding for added availability. `ethtool` can not be used to determine the status of a bonded network interface but, since we know that interfaces `eth0` and `eth1` were used to comprise the bonded interface `bond0`, `ethtool` can be used to check the status of both the underlying network links.

```
/sbin/ethtool eth0 | grep -q "Link detected: yes"
```

```
/sbin/ethtool eth1 | grep -q "Link detected: yes"
```

In this manner, each of the commands returns a zero (0) indicating successful status or one (1) if unsuccessful.

In the screen capture below, we see heuristic examples using `ethtool` to check both of the network interfaces that comprise `bond0`. Note that even though there are two heuristics, each with a score of 1, the Minimum Score is still set to 1 because as long as one of the interfaces is up, the system can participate in cluster communications.

homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure
- hevs
- webservers1

webservers1

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

webservers1

General Fence Multicast Quorum Partition

**Quorum Partition Configuration**

Do not use a Quorum Partition

Use a Quorum Partition

Interval: 3

Votes: 1

TKO: 9

Minimum Score: 1

Device: /dev/mapper/quorum

Label: qdisk

**Heuristics**

Path to Program	Interval	Score	
/sbin/ethtool eth0   grep -q "Link	5	1	
/sbin/ethtool eth1   grep -q "Link	5	1	

Add another heuristic

Apply

8. Once the preferred quorum attributes has been entered and any desired heuristic(s), and their respective scores, have been defined, click Apply to create the quorum device.



By choosing to use a quorum device, the following cman XML tag will be placed into the *cluster.conf* file indicating that this is a 2-node cluster expecting a quorum device vote:

```
<cman expected_votes="3" two_node="0"/>
```

or simply:

```
<cman expected_votes="3"/>
```

If further information regarding quorum partition details and heuristics is required, please reference:

- the *Considerations for Using Quorum Disk* and *Global Cluster Properties* sections of *Configuring and Managing a Red Hat Cluster*
- the Cluster Project FAQ
- Red Hat Knowledgebase Article ID 13315
- the `qdisk(5)` man page

## 8.2 Configuring Cluster Members

Once the initial cluster creation has completed, the user will need to configure each of the clustered nodes.

From the cluster details window, click Nodes from the menu at left. This will display a window displaying the current details of the created cluster including the system names of cluster members. Clicking on either of the system names on this page will display, and allow the modification of, the cluster configuration information page for that node.



**clusters**

- Cluster List
- Create a New Cluster
- Configure

**webserver1**

- Nodes
  - Add a Node
  - Configure
    - e1-virt09-ic.lab.bos.redhat.com
    - e1-virt08-ic.lab.bos.redhat.com**
- Services
- Resources
- Failover Domains
- Shared Fence Devices

**webserver1**

Node Name: [et-virt08-ic.lab.bos.redhat.com](#) Choose a Task... Go

Status: **Cluster member**

Show recent log activity for this node

---

**Cluster daemons running on this node**

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

---

**Services on this Node**

- No cluster services are currently running here

---

**Failover Domain Membership**

- This node has no failover domain membership

---

**Main Fencing Method** **Backup Fencing Method**

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

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

## 8.3 Fencing

Remember that for the purposes of cluster and data integrity, cluster members need to be in constant communication to coordinate all shared resource activities. Should any member of a cluster deem another member unresponsive or otherwise unreachable, it has the authority to reboot (fence) it from the cluster to prevent potential data corruption. Fencing is the component of a cluster that severs access to a resource (hard disk, etc.) from a node in the cluster should it lose contact with the rest of the cluster members. The most effective way to do this is to force the system to power down or reboot.

A node can have multiple fence methods and each fence method can have multiple fence devices.

Multiple fence methods are set up for redundancy. For example, the user may have a baseboard management fencing method for a node in the cluster such network dependent connections like IPMI, ILO, RSA, or DRAC. If this connection should fail, fencing would not occur. As a backup fence method, one can declare a second method of fencing that used a power switch or something similar to fence the node. If the first method failed to fence the node, the second fence method would then be employed.



Multiple fence devices per method are used, for example, if a node has dual power supplies and power fencing is the fence method of choice. If only one power supply were fenced, the node would not reboot given the redundant power supply. In this case, the user could configure two fence devices in one method: one for each power supply. All fence devices within a fence method must succeed in order for the method to succeed.

Click on the *Add a fence device for this level* link at the bottom of the system details page to reveal the *Use an existing Fence Device* pulldown menu.

Since the examples in this document use Dell systems, the *Dell DRAC* option was selected from the list of known devices. The DRAC information necessary to allow luci the remote access is configured in BIOS at system startup. The information configured there to allow remote console control will be needed by luci.

Once a fence device type is selected, the user is prompted for information specific to that fence type. In the example that follows, the system name, address and login information are entered for the Dell DRAC fence type.

Main Fencing Method	Backup Fencing Method
<p><b>Fence Type</b> Dell Drac</p> <p>Name <input type="text" value="v8-drac"/></p> <p>IP Address <input type="text" value="10.16.41.76"/></p> <p>Login <input type="text" value="root"/></p> <p>Password <input type="password" value="....."/></p> <p>Password <input type="password"/></p> <p>Script <input type="text"/></p> <p>(optional)</p> <p><input type="button" value="Remove this device"/></p> <p><a href="#">Add a fence device to this level</a></p>	<p><a href="#">Add a fence device to this level</a></p>
<input type="button" value="Update main fence properties"/>	<input type="button" value="Update backup fence properties"/>

Enter the information for the fence device being used. Click on *Update main fence properties* to proceed.

Be sure to configure the fence device for each node.

Reference the *Configuring Fence Devices* section in *Configuring and Managing a Red Hat Cluster* for more information regarding the various shared and non-shared fence devices available.



## 8.4 Failover Domains

A failover domain is a chosen subset of cluster members that are eligible to run a cluster service in the event of a node failure. The characteristics of a failover domain can be specified upon domain creation or later. They are:

- *Unrestricted* (although a subset of members are preferred, a cluster service assigned to this domain can run on any available member)
- *Restricted* (restricts the members that can run a specific cluster service. If none of the members in a restricted failover domain are available, the service will not be started)
- *Unordered* (the member on which the cluster service runs is chosen from the available failover domain members with no priority ordering)
- *Ordered* (specifies an ordered list of preferred members in a failover domain)

The user can create a failover domain and include the members of the newly created cluster although by default, if no failover domain exists, the service will relocate to any other node in the cluster.

From the cluster details window, click *Failover Domains* and then *Add a Failover Domain*.

The screenshot shows the Red Hat Cluster Manager interface. At the top, there are tabs for 'homebase', 'cluster', and 'storage'. Below the tabs, there are two main panels. The left panel, titled 'clusters', contains a 'Cluster List' and options to 'Create a New Cluster' and 'Configure'. The right panel, titled 'webserver1', contains an 'Add a Failover Domain' form. The form has a text input field for 'Failover Domain Name' with the value 'webserver1\_FOD'. Below this are three checkboxes: 'Prioritized', 'Restrict failover to this domain's members', and 'Do not fail back services in this domain', all of which are unchecked. Under the heading 'Failover domain membership', there is a table with three columns: 'Node', 'Member', and 'Priority'. The table contains two rows, both with the node 'et-virt08.lab.bos.redhat.com' and a checked 'Member' box. The 'Priority' column has input fields with the value '1'. At the bottom of the form is a 'Submit' button.

Node	Member	Priority
et-virt09.lab.bos.redhat.com	<input checked="" type="checkbox"/>	1
et-virt08.lab.bos.redhat.com	<input checked="" type="checkbox"/>	1

Specify a failover domain name. It is recommended to choose a name that adequately describes the domain's purpose. The remaining options are user preferences.

- The *Prioritized* check box enables setting a failover priority of the members in the failover domain
- The *Restrict failover to this domain's members* check box is for restricting failovers to



members in this failover domain (e.g. services assigned to this domain fail over only to nodes in this failover domain)

- Leave the *Do not fail back services in the domain* option unchecked

By default, failover domains are unrestricted and unordered.

Under *Failover domain membership*, select the *Member* check box for each node that is to be a member of the failover domain. If the *Prioritized* option was checked, set the preferred priority for each member of the failover domain.

Click the *Submit* button to process the option selected on the page. This will return the user to the Failover Domain Form. Clicking on the *Failover Domains* link at left should display the newly created domain.

Reference the *Configuring a Failover Domain* section in *Configuring and Managing a Red Hat Cluster* for greater detail.

## 8.5 Cluster Resources

There are many types of cluster resources that can be configured. Reference the *Adding a Cluster Service to the Cluster* section of *Configuring and Managing a Red Hat Cluster* for more information. The following two resource types will be defined to provide the high-availability functionality of the web service.

- Script
- IP Address

### 8.5.1 Script

A script resource basically references a script that will be executed. This script could be pre-existing or authored for a specific purpose. The `/etc/rc.d/init.d/httpd` script (Apache's start/stop script) used in this document is pre-existing, provided by the additional package group (Web Server) that was selected during OS installation.

Starting from the cluster details page in luci,

1. click on the *Resources* link in the menu at left
2. Click on *Add a Resource*.
3. From the *Select a Resource Type* pulldown menu, select *Script*.



homebase cluster storage

---

clusters

- Cluster List
- Create a New Cluster
- Configure

webserv1

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

webserv1

---

Add a Resource

**Script Resource Configuration**

Name

Full path to script file

4. Enter a name and the location of the httpd script. Use a logical name for the script so it can easily be recognized when viewing all cluster resources.
5. Click *Submit* to create the script resource.

## 8.5.2 IP Address

This resource address can be used by any cluster service that requires one. Once associated with a cluster service, it can be relocated by a cluster member if it deems it necessary, or manually through a GUI interface, a web interface (conga) or via command line. If any cluster member providing the service becomes unable to do so (e.g. due to hardware or software failure, network/connectivity loss, etc.), the service IP address will automatically migrate to an eligible member. Typically, a block of addresses are reserved for various cluster services.

The user must define an address that the web service will use to serve the HTML content.

Starting from the cluster details page in luci, click on the *Resources* link in the menu at left and then on *Add a Resource*. From the *Select a Resource Type* pulldown menu, select *IP address*.





homebase cluster storage

clusters

- Cluster List
- Create a New Cluster
- Configure

webserver1

Nodes

Services

Resources

Add a Resource

Configure a Resource

- httpd
- 10.16.40.165
- web-content
- Failover Domains
- Shared Fence Devices

webserver1

Configure 10.16.40.164

**IP Address Resource Configuration**

IP address

Monitor link

Enter the IP address reserved for the service and select the *Monitor Link* check box. Click *Submit* to create the IP address resource.

The IP service resource should now be listed using the `ip` command. Note that interface `eth2` (the public NIC) lists both the public IP address as well as the cluster resource IP address.

```
# /sbin/ip addr list
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
master bond0 qlen 1000
    link/ether 00:0e:0c:b6:5d:03 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::20e:cff:feb6:5d03/64 scope link
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
master bond0 qlen 1000
    link/ether 00:0e:0c:b6:5d:03 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::20e:cff:feb6:5d03/64 scope link
        valid_lft forever preferred_lft forever
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:13:72:4c:a5:e5 brd ff:ff:ff:ff:ff:ff
    inet 10.16.41.71/21 brd 10.16.47.255 scope global eth2
    inet 10.16.40.164/21 scope global secondary eth2
    inet6 fe80::213:72ff:fe4c:a5e5/64 scope link
        valid_lft forever preferred_lft forever
5: sit0: <NOARP> mtu 1480 qdisc noop
    link/sit 0.0.0.0 brd 0.0.0.0
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue
    link/ether 00:0e:0c:b6:5d:03 brd ff:ff:ff:ff:ff:ff
    inet 10.10.1.1/24 brd 10.10.1.255 scope global bond0
```



```
inet6 fe80::20e:cff:feb6:5d03/64 scope link tentative
    valid_lft forever preferred_lft forever
7: virbr0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
    inet6 fe80::200:ff:fe00:0/64 scope link
        valid_lft forever preferred_lft forever
        valid_lft forever preferred_lft forever
```

Once the resources have been defined, click on *Resources* in the blue menu at left to view the Resources page for the cluster.

homebase cluster storage

---

clusters

- Cluster List
- Create a New Cluster
- Configure

webserver1

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

### webserver1

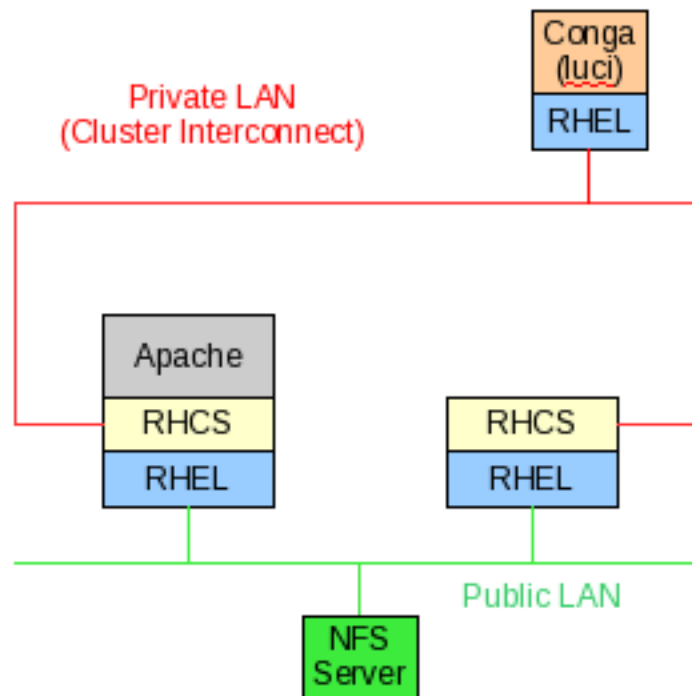
#### Resources for webserver1

Resource Name	Type	Configure	Delete
httpd	Script	<a href="#">configure</a>	<a href="#">delete</a>
10.16.40.164	IP Address	<a href="#">configure</a>	<a href="#">delete</a>



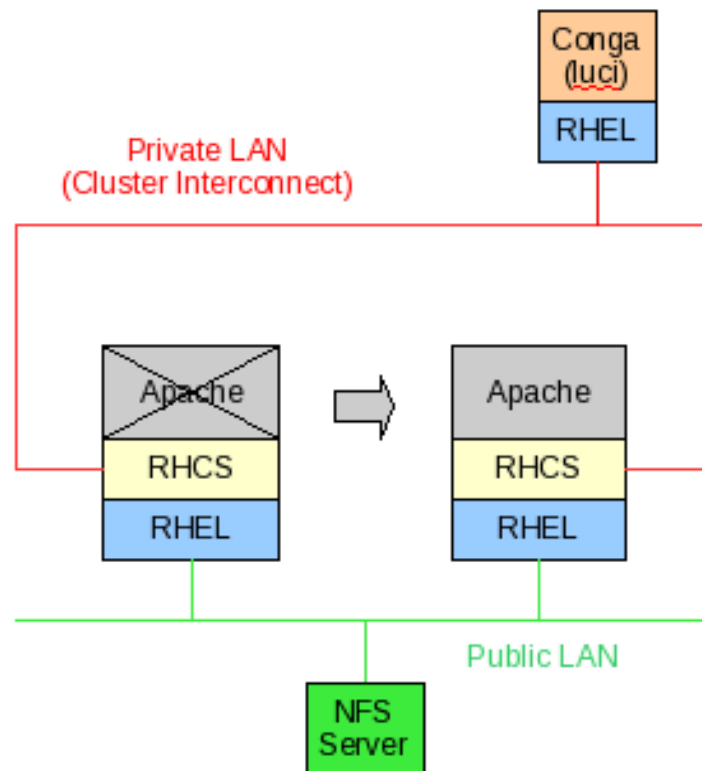
## 8.6 Web Service (httpd)

External web clients query a single IP address, in this case the IP Address resource previously created, which may be answered by any one of the cluster members at any given time.



The diagrams below illustrate the cluster components and where they reside. In the first example, the web service (Apache) is up and running on only one clustered node at any given time.

If the web service were to fail over to the other node, or if the user chose to manually migrate the service (for load balancing purposes, system maintenance, etc.), the service would appear on the other node.



On both nodes, create the directory from which the web content will be served.

```
# mkdir /web_content
```

## 8.6.1 Service Creation

Starting from the cluster details page in luci, click on the *Services* link in the menu at left and then on *Add a Service*.



homebase cluster storage

---

clusters

- Cluster List
- Create a New Cluster
- Configure

webserver1

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

### webserver1

#### Add a Service

<b>Service name</b>	<input type="text" value="httpd"/>
Automatically start this service	<input checked="" type="checkbox"/>
Run exclusive	<input type="checkbox"/>
Failover Domain	<input type="text" value="webserver1_FOD"/>
Recovery policy	<input type="text" value="Restart"/>

1. Enter a logical name for the service
2. Check the *Automatically start this service* box
3. Select the previously created failover domain from the pulldown menu
4. Select a preferred *Recovery policy* from the pulldown menu.

Now the the two previously created cluster resources can be assigned to this cluster service. Click on the *Add a resource to this domain* button to display more fields for defining each resource.



homebase cluster storage

**clusters**

- Cluster List
- Create a New Cluster
- Configure

**webserv1**

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

### webserv1

#### Add a Service

**Service name**

Automatically start this service

Run exclusive

Failover Domain

Recovery policy

**Add a new local resource**

or

**Use an existing global resource**

1. From the *Use an existing global resource* pulldown menu, select the previously created cluster script resources in the list.
2. Click on the *Add a resource to this domain* button again to add the cluster IP address resource.

Both of the cluster resources should be listed under the newly defined service (httpd).



homebase cluster storage

---

**clusters**

- Cluster List
- Create a New Cluster
- Configure

**webserver1**

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

---

### webserver1

**Service Name** httpd  
**Service Status** Running on et-virt08-ic.lab.bos.redhat.com

#### Service Composition

##### Script Resource Configuration

Name:   
Full path to script file:

This resource is an independent subtree

##### IP Address Resource Configuration

IP address:   
Monitor link:

This resource is an independent subtree

Automatically start this service   
Run exclusive   
Failover Domain:   
Recovery policy:

Click the *Submit* button to create the httpd service and associate the cluster resources with this service.

Click again on the *Services* link in the menubar at left to see the httpd service listed, green if the service is running, red if stopped. If green, stop the service using the *Choose a Task ...* pulldown menu and clicking the *Go* button.

There are some changes to make before starting the service.



## 8.6.2 httpd Configuration Directives

This section describes two server configuration directives that must be modified to configure the httpd service. The configuration directives for httpd are maintained in the `/etc/httpd/conf/httpd.conf` file.

Edit file on all nodes to bind Apache to the preferred IP address(es) and port(s). The `Listen` directive tells the httpd service on which IP address and port it should listen for HTML queries. It should be set to the shared resource IP address that was arranged for the cluster service to use and is specified to prevent Apache from using all bound IP addresses (0.0.0.0).

In the example below, IP address 10.16.40.164 was the cluster resource for this service and the default HTTP port (80) will be used. Modify the `httpd.conf` file and set the `Listen` directive accordingly.

```
Listen 10.16.40.164:80
```

The default location for the root directory for files served by httpd is `/var/www/html`. The user can override this default by modifying the `DocumentRoot` directive. It is not necessary to alter this default setting. In this document, the cluster will serve the HTML pages from the local `/web_content` directory. Set the `DocumentRoot` directive accordingly.

```
DocumentRoot "/web_content"
```

If CGI scripts are to be used and the script directory resides in a non-standard location, specify the directory that contains the CGI programs. For example:

```
ScriptAlias /cgi-bin/ "/mnt/httpdservice/cgi-bin/"
```

Save the changes, exit the file and ensure that the same changes are performed on all cluster members.

Reference the *Example of Setting Up Apache HTTP Server* appendix of *Configuring and Managing a Red Hat Cluster* for more information on Apache configuration.

Use `luci` to restart the httpd service.

## 8.6.3 Testing

In the cluster details window of `luci`, click on the `Services` link in the menubar at left to see the httpd service listed. Start the service on any node in the cluster using the *Choose a Task ...* pulldown menu and clicking the `Go` button. If the service has successfully started, its name will be displayed in green.

Remember that at this point SELinux is still in Permissive mode. Now any attempts to access the web service should generate some SELinux warnings in `/var/log/messages` resembling the following:

```
setroubleshoot: SELinux is preventing httpd (httpd_t) "read write" to socket (initrc_t). For complete SELinux messages. run sealert -l bf7a9b2c-fc8f-49fa-a5f7-0f40b6b52c4a
```





(remember to search for 'avc'). Try accessing the web page by directing your browser to the IP address used in the httpd service resource.

The following page, ...

## Red Hat Enterprise Linux Test Page

This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.

### If you are a member of the general public:

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting [www.example.com](http://www.example.com), you should send e-mail to "webmaster@example.com".

For information on Red Hat Enterprise Linux, please visit the [Red Hat, Inc. website](http://www.redhat.com). The documentation for Red Hat Enterprise Linux is [available on the Red Hat, Inc. website](http://www.redhat.com).

### If you are the website administrator:

You may now add content to the directory `/var/www/html/`. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file `/etc/httpd/conf.d/welcome.conf`.

You are free to use the image below on web sites powered by the Apache HTTP Server:



... is the default Apache start page. To prevent this page from being displayed, follow the instructions in the `/etc/httpd/conf.d/welcome.conf` file. Commenting out the active lines of the file will prevent it from being seen.

Once the `welcome.conf` file has been disabled, try using `luci` to start/stop/relocate the service. See if the `httpd` service is accessible using the same IP address when the service is relocated to another node.

Try using `luci` to restart the cluster. Do the nodes rejoin the cluster? Does the `httpd` service automatically start on one node?

Not all of these activities are expected to succeed but the various actions will be generating violation entries in the SELinux log. These violations (reported as warnings because SELinux is in permissive mode) will be used to modify SELinux policy to allow just the activities desired to support the web service.

Now the user can adjust SELinux accordingly and test to see if SELinux can run in Enforcing mode and still permit the web service functionality.

## 9 SELinux Policy Adjustments

Constant observation of the system console and audit log entries is critical to SELinux administration. All troubleshooting of SELinux violations involves monitoring the logs for SELinux AVC denials.



## 9.1 AVC Denials

SELinux violations are reported in the system logs as AVC denials. AVC is a convenient acronym when searching for denials in the log files. Each AVC denial references some type of policy breach and possibly a potential resolution.

The easiest method to determine where SELinux policies inconvenience a web service is to run SELinux in permissive mode and monitor `/var/log/messages` and `/var/log/audit/audit.log` for entries where SELinux would have prevented the action had it been in Enforcing mode.

Try accessing the web page by directing your browser to the IP address used in the httpd service resource. If the `httpd_selinux` booleans relevant to the http or https service were addressed earlier, let SELinux identify which policies will hamper the web service activities by monitoring `/var/log/messages` for entries such as:

```
setroubleshoot: SELinux is preventing httpd (httpd_t) "read write" to socket (initrc_t). For complete SELinux messages. run sealert -l bf7a9b2c-fc8f-49fa-a5f7-0f40b6b52c4a
...
setroubleshoot: SELinux is preventing modclusterd (ricci_modclusterd_t) "write" to pipe (ricci_modcluster_t). For complete SELinux messages. run sealert -l 371b2686-6b6b-45d0-a5bc-d199624f6946
...
setroubleshoot: SELinux is preventing service (ricci_modstorage_t) "read" to ./consoletype (consoletype_exec_t). For complete SELinux messages. run sealert -l 0ae4080a-5e27-4116-b095-7b5a244ba433
```

For most entries observed, the warning includes a method for viewing the details behind the alert. From the example above,

```
# sealert -l 371b2686-6b6b-45d0-a5bc-d199624f6946
```

Summary:

```
SELinux is preventing modclusterd (ricci_modclusterd_t) "write" to pipe (ricci_modcluster_t).
```

Detailed Description:

```
SELinux denied access requested by modclusterd. It is not expected that this access is required by modclusterd and this access may signal an intrusion attempt. It is also possible that the specific version or configuration of the application is causing it to require additional access.
```

Allowing Access:

```
You can generate a local policy module to allow this access - see FAQ (http://fedora.redhat.com/docs/selinux-faq-fc5/#id2961385) Or you can disable SELinux protection altogether. Disabling SELinux protection is not recommended. Please file a bug report (http://bugzilla.redhat.com/bugzilla/enter\_bug.cgi) against this package.
```

Additional Information:



```
Source Context      system_u:system_r:ricci_modclusterd_t
Target Context     system_u:system_r:ricci_modcluster_t
Target Objects     pipe [ fifo_file ]
Source             modclusterd
Source Path        /usr/sbin/modclusterd
Port               <Unknown>
Host               et-virt08.lab.bos.redhat.com
Source RPM Packages modcluster-0.12.0-7.el5
Target RPM Packages
Policy RPM         selinux-policy-2.4.6-172.el5
Selinux Enabled    True
Policy Type        targeted
MLS Enabled        True
Enforcing Mode     Enforcing
Plugin Name        catchall
Host Name          et-virt08.lab.bos.redhat.com
Platform           Linux et-virt08.lab.bos.redhat.com
                   2.6.18-92.1.6.el5 #1 SMP Fri Jun 20 02:36:06 EDT
                   2008 x86_64 x86_64
Alert Count        2
First Seen         Wed Oct 22 14:05:30 2008
Last Seen          Fri Oct 24 14:20:15 2008
Local ID           4ca5eb2c-926f-4c85-90e2-cbc2e24c15a6
Line Numbers
```

#### Raw Audit Messages

```
host=et-virt08.lab.bos.redhat.com type=AVC msg=audit(1224872415.416:5086): avc:
denied { write } for pid=10273 comm="modclusterd" path="pipe:[802592]"
dev=pipefs ino=802592 scontext=system_u:system_r:ricci_modclusterd_t:s0
tcontext=system_u:system_r:ricci_modcluster_t:s0 tclass=fifo_file
```

```
host=et-virt08.lab.bos.redhat.com type=AVC msg=audit(1224872415.416:5086): avc:
denied { write } for pid=10273 comm="modclusterd" path="pipe:[802593]"
dev=pipefs ino=802593 scontext=system_u:system_r:ricci_modclusterd_t:s0
tcontext=system_u:system_r:ricci_modcluster_t:s0 tclass=fifo_file
```

```
host=et-virt08.lab.bos.redhat.com type=SYSCALL msg=audit(1224872415.416:5086):
arch=c000003e syscall=59 success=yes exit=0 a0=952e530 a1=952e240 a2=952eb10
a3=0 items=0 ppid=10272 pid=10273 auid=4294967295 uid=0 gid=0 euid=0 suid=0
fsuid=0 egid=0 sgid=0 fsgid=0 tty=(none) ses=4294967295 comm="modclusterd"
exe="/usr/sbin/modclusterd" subj=system_u:system_r:ricci_modclusterd_t:s0
key=(null)
```

Remember to search the `sealert` output for 'avc'). For example,

```
type=AVC msg=audit(1117726540.077:9322426): avc: denied { read } for pid=10652 comm="tail"
name=audit.log dev=dm-0 ino=328749 scontext=root:staff_r:staff_t
tcontext=system_u:object_r:auditd_log_t tclass=file
```

The above example flagged a read attempt on the file `audit.log` using the `tail` command. From the AVC entry we can determine:



- The source process context was *root:staff\_r:staff\_t*
- The context of the targeted file was *system\_u:object\_r:auditd\_log\_t*.

From this it is determined that the *staff\_t* domain has no read access to the *auditd\_log\_t* file type. This is as it should be, if we had used a `newrole` command to transition to the *sysadm\_r* role, we would be running `tail` in the *sysadm\_t* domain and access would have been granted.

## 9.2 audit2allow

The user should now be able to use the `audit2allow` command to augment the SELinux policy to ignore the specific messages observed in `/var/log/audit/audit.log`. `audit2allow` is a perl script that reads logged denials and produces matching rules that can be added to SELinux policy to thereafter allow the operations that were denied. It is not intended as an automatic policy generator, but as an aid to developing policy. For obvious reasons, modifying the system security in this manner requires that the administrator be sure that the applications that are granted exception to policy, and specifically the actions they perform, are legitimate and authorized. The *audit.log* file can be used as input to `audit2allow` as demonstrated below.

```
# cat /var/log/audit/audit.log | audit2allow -M local
***** IMPORTANT *****
To make this policy package active, execute:

semodule -i local.pp
```

This will generate two new files in the present working directory:

- `local.te`
- `local.pp`

While `local.pp` contains binary data, `local.te` can be examined to view the changes to policy that will be applied.

```
# cat local.te

module local 1.0;

require {
    type ifconfig_t;
    type ping_t;
    type httpd_t;
    type rdisc_t;
    type initrc_t;
    type netutils_t;
    class unix_stream_socket { read write };
    class file { read getattr };
}

#===== httpd_t =====
allow httpd_t initrc_t:unix_stream_socket { read write };
```



```
#===== ifconfig_t =====
allow ifconfig_t initrc_t:unix_stream_socket { read write };

#===== netutils_t =====
allow netutils_t initrc_t:unix_stream_socket { read write };

#===== ping_t =====
allow ping_t initrc_t:unix_stream_socket { read write };

#===== rdisc_t =====
allow rdisc_t initrc_t:unix_stream_socket { read write };
```

The changes are applied using the SELinux policy management tool, `semodule`, with the `-i` switch specifying which module to install.

```
# semodule -i local.pp
```

The resulting SELinux policy should be the same on both nodes to ensure cluster integrity. This can be accomplished by using `semodule` to load the same `local.pp` file onto each cluster node. If for any reason, the contents of the `local.te` file differ across cluster members (some may have additional rules), it is best to load the file containing the additional policy rules on all members.

Save the `local.te` and `local.pp` files aside for later comparison if necessary.

Now that SELinux has been adjusted to ignore the specific component exceptions to policy, set SELinux to targeted enforcing mode on all cluster members ...

```
# setenforce 1
```

Verify that the served web pages are accessible once SELinux is enforced. Then the default settings can be set accordingly in `/etc/selinux/config` on all cluster members to make the changes persistent across reboots.

Verify that the `httpd` service can start and the web content is accessible on either node and by using `luci` to:

- restart the cluster
- relocate the service
- reboot the node running the service
- fence the node running the service

using the pulldown menu next to the cluster (`webserver1`) or service (`httpd`) links.

Once running, the `httpd` service can be started, stopped or relocated to another cluster member using `luci` or via the `clusvcadm` command.

```
# clusvcadm -r httpd -m et-virt09.lab.bos.redhat.com
Trying to relocate service:httpd to et-virt09.lab.bos.redhat.com...Success
service:httpd is now running on et-virt09.lab.bos.redhat.com
```

See the manpage for `clusvcadm` for details.



Try fencing the node running the httpd service to verify that the service stops on that node, relocates to a surviving node, and that the fenced node rejoins the cluster after rebooting.

If any of these activities does not function as expected, or if any SELinux

## 10 Diagnostics

### 10.1 clustat

At any given time, the clustat command can be used to view the overall status of the cluster.

```
# clustat
Cluster Status for webserver1 @ Thu Nov 13 13:27:55 2008
Member Status: Quorate

Member Name                               ID    Status
-----
et-virt09-ic.lab.bos.redhat.com           1    Online, Local, rgmanager
et-virt08-ic.lab.bos.redhat.com           2    Online, rgmanager
/dev/sdb                                   0    Online, Quorum Disk

Service Name                               Owner (Last)                               State
-----
service:httpd                             et-virt08-ic.lab.bos.redhat.co started
```

This output is very handy for a quick glance to see what nodes are up and where the cluster services are located. In the example above, the command was executed on node et-virt0 so the additional status of *Local* is included for that node in the output.

### 10.2 Logs

All system information regarding OS status and events are logged to the `/var/log/messages` file. The `dmesg` command is also helpful in listing the most recent system bootup messages.

SELinux and httpd specific information are also logged to `/var/log/messages`. Using `'tail -f'` on this file can be helpful to see more specific complaints if luci should report a node or service not behaving as expected. It is also quite useful when running with SELinux in permissive mode to see what access violations would have been prevented if SELinux had been in enforcing mode.

SELinux also logs security audit information in `/var/log/audit/audit.log`.

CLVM message output passes through a logging module with independent choices of logging levels (defined in set in the `/etc/lvm/lvm.conf` file ) for stdout and stderr, syslog, log file, and external log function.



# 11 Conclusions & Next Steps

The goal of this volume was to demonstrate the creation of a highly available web server using RHCS services on a 2-node cluster. The examples provided demonstrate how to perform an OS installation, create the cluster, create the CLVM volumes and GFS2 file system, create the necessary resources and httpd service, adjust the firewall and SELinux to provide system security, and successfully serve cluster shared web content from shared storage.

## Appendix A: Using Local Web Content

The web pages served by the httpd service can be kept locally on both cluster nodes rather than served from shared storage or NFS export. For instance, if there were no shared storage available, rather than acting as a mount point for the GFS2 volume, the `/web_content` directory could contain the web content on each individual member. In this case, it is important that both nodes possess the same data since their respective `/web_content` directories are not shared. The locally stored content will have to be synced across both nodes. This way if the httpd service fails over to another node, the `/web_content` directory on that node will then serve the same web content.

Once the secure shell access between nodes has been configured, use `rsync` to keep the `/web_content` directories in sync on both nodes. For example, from node `et-virt08` ...

```
# rsync -avz -e ssh root@et-virt09.lab.bos.redhat.com:/web_content/ \
/web_content
```

... will synchronize the `/web_content` directory contents on `et-virt08` with those on `et-virt09`. This can be configured as a cron job if preferred.

## Appendix B: Configuration Files

### Cluster

The Cluster Configuration System (CSS) attempts to manage the `/etc/cluster/cluster.conf` file and maintain all the nodes in sync. If you make changes to the `cluster.conf` file, CSS and CMAN must be made aware that changes have been made so other nodes are updated accordingly. Else, the changes are likely to be overwritten with an older version of the `cluster.conf` file from a different node. The cluster configuration GUIs propagate changes to `cluster.conf` to all cluster members. The `system-config-cluster` GUI provides a button labeled *Send to Cluster*.

If the `cluster.conf` file has been modified by hand, then the user will need to:

1. ensure that the `config_version` value in line 2 of the file has been incremented
2. propagate the new version to the rest of the cluster





In the example below, the *cluster.conf* file has been modified by hand and the *config\_version* value has been changed from 3 to 4. To propagate the new version to the rest of the cluster, run the following on the node where the edits were done.

```
# ccs_tool update /etc/cluster/cluster.conf
Config file updated from version 44 to 45

Update complete.

# cman_tool version -r 45
```

To verify the changes have been propagated, the version number update can be viewed on any node at any time using *cman\_tool*.

```
# cman_tool status | grep -i "Config version"
Config Version: 45
```

Below is the *cluster.conf* file used in this document.

```
<?xml version="1.0"?>
<cluster alias="webserver1" config_version="45" name="webserver1">
  <fence_daemon clean_start="0" post_fail_delay="0" post_join_delay="30"/>
  <clusternodes>
    <clusternode name="et-virt09-ic.lab.bos.redhat.com" nodeid="1" votes="1">
      <fence>
        <method name="1">
          <device name="v9-drac"/>
        </method>
      </fence>
    </clusternode>
    <clusternode name="et-virt08-ic.lab.bos.redhat.com" nodeid="2" votes="1">
      <fence>
        <method name="1">
          <device name="v8-drac"/>
        </method>
      </fence>
    </clusternode>
  </clusternodes>
  <cman expected_votes="3" two_node="0"/>
  <totem token="54000"/>
  <fencedevices>
    <fencedevice agent="fence_drac" ipaddr="10.16.41.72" login="root" name="v8-drac"
passwd="calvin"/>
    <fencedevice agent="fence_drac" ipaddr="10.16.41.74" login="root" name="v9-drac"
passwd="calvin"/>
  </fencedevices>
  <rm>
    <failoverdomains>
      <failoverdomain name="webserver1_FOD" nofailback="0" ordered="0" restricted="0">
        <failoverdomainnode name="et-virt09-ic.lab.bos.redhat.com" priority="1"/>
      </failoverdomain>
    </failoverdomains>
  </rm>
</cluster>
```





```
        <failoverdomainnode name="et-virt08-ic.lab.bos.redhat.com" priority="1"/>
    </failoverdomain>
</failoverdomains>
<resources>
    <script file="/etc/rc.d/init.d/httpd" name="httpd"/>
    <ip address="10.16.40.164" monitor_link="1"/>
</resources>
<service autostart="1" domain="webserver1_FOD" exclusive="0" name="httpd"
recovery="restart">
    <script ref="httpd"/>
    <ip ref="10.16.40.164"/>
</service>
</rm>
<quorumd device="/dev/mapper/quorum" interval="3" label="qdisk" min_score="1" tko="9"
votes="1">
    <heuristic interval="5" program="/sbin/ethtool eth0 | grep -q &quot;Link detected:
yes&quot;" score="1"/>
    <heuristic interval="5" program="/sbin/ethtool eth1 | grep -q &quot;Link detected:
yes&quot;" score="1"/>
</quorumd>
</cluster>
```

## Firewall

The `/etc/sysconfig/iptables` file used during the course of testing.

```
# Generated by iptables-save v1.3.5 on Thu Jul 31 14:42:27 2008
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [685802:71309812]
:RH-Firewall-1-INPUT - [0:0]
-A INPUT -j RH-Firewall-1-INPUT
-A FORWARD -j RH-Firewall-1-INPUT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m multiport --dports 14567 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m multiport --dports 8084 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m multiport --dports 21064 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m multiport --dports 11111 -j ACCEPT
-A RH-Firewall-1-INPUT -p udp -m state --state NEW -m multiport --dports 5404,5405 -j ACCEPT
-A RH-Firewall-1-INPUT -p udp -m state --state NEW -m multiport --dports 50007 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m multiport --dports 50006,50008,50009 -j
ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m multiport --dports 41966,41967,41968,41969
-j ACCEPT
-A RH-Firewall-1-INPUT -i lo -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -m icmp --icmp-type any -j ACCEPT
-A RH-Firewall-1-INPUT -p esp -j ACCEPT
```



```
-A RH-Firewall-1-INPUT -p ah -j ACCEPT
-A RH-Firewall-1-INPUT -d 224.0.0.251 -p udp -m udp --dport 5353 -j ACCEPT
-A RH-Firewall-1-INPUT -p udp -m udp --dport 631 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 631 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 443 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 80 -j ACCEPT
-A RH-Firewall-1-INPUT -j REJECT --reject-with icmp-host-prohibited
COMMIT
# Completed on Thu Jul 31 14:42:27 2008
```

## Multipathing

The `/etc/multipath.conf` file used during the course of testing.

```
# This is a basic configuration file with some examples, for device mapper
# multipath.
# For a complete list of the default configuration values, see
# /usr/share/doc/device-mapper-multipath-0.4.7/multipath.conf.defaults
# For a list of configuration options with descriptions, see
# /usr/share/doc/device-mapper-multipath-0.4.7/multipath.conf.annotated

##
## Here is an example of how to configure some standard options.
##
#
defaults {
    udev_dir          /dev
    polling_interval  5
    selector          "round-robin 0"
    path_grouping_policy multibus
    getuid_callout    "/sbin/scsi_id -g -u -s /block/%n"
    prio_callout      /bin/true
    path_checker      readsector0
    rr_min_io         100
    max_fds           8192
    rr_weight         priorities
    failback          immediate
    no_path_retry     3
    user_friendly_names yes
}
##
## The wwid line in the following blacklist section is shown as an example
## of how to blacklist devices by wwid. The 2 devnode lines are the
## compiled in default blacklist. If you want to blacklist entire types
## of devices, such as all scsi devices, you should use a devnode line.
```



```
## However, if you want to blacklist specific devices, you should use
## a wwid line. Since there is no guarantee that a specific device will
## not change names on reboot (from /dev/sda to /dev/sdb for example)
## devnode lines are not recommended for blacklisting specific devices.
##
blacklist {
    wwid 3600805f30008a840b194839bdb6e000f
}
# devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
# devnode "^hd[a-z]"
#}
multipaths {
    multipath {
        wwid          360060160c914150142d0d3f571bfdc11
        alias         quorum
    }
    multipath {
        wwid          360060160541415013a6c98657610dd11
        alias         data1
    }
    multipath {
        wwid          36006016054141500183f857db2d8dc11
        alias         data2
    }
    multipath {
        wwid          36006016054141500d0ce586cb2d8dc11
        alias         data3
    }
}
devices {
    ## Device attributes for EMC CLARiiON
    device {
        vendor "DGC"
        product "*"
        path_grouping_policy group_by_prio
        getuid_callout "/sbin/scsi_id -g -u -s /block/%n"
        prio_callout "/sbin/mpath_prio_emc /dev/%n"
        hardware_handler "1 emc"
        features "1 queue_if_no_path"
        no_path_retry 300
        path_selector "round-robin 0"
        path_checker emc_clariion
        failback immediate
    }
}
```



## Network Interfaces

Below find examples of the network interface configuration files used.

```
/etc/sysconfig/network-scripts/ifcfg-eth2    # Public NIC
```

```
# Intel Corporation 82541GI Gigabit Ethernet Controller
DEVICE=eth2
BOOTPROTO=dhcp
HWADDR=00:13:72:4C:A2:36
ONBOOT=yes
DHCP_HOSTNAME=et-virt09.lab.bos.redhat.com
```

```
/etc/sysconfig/network-scripts/ifcfg-bond0    # Bonded cluster interconnect
```

```
DEVICE=bond0
BOOTPROTO=none
ONBOOT=yes
NETMASK=255.255.255.0
IPADDR=10.10.1.1
USERCTL=no
TYPE=Ethernet
IPV6INIT=no
PEERDNS=yes
BONDING_OPTS="mode=1 miimon=100 primary=eth0"
```

## Appendix C: RHN

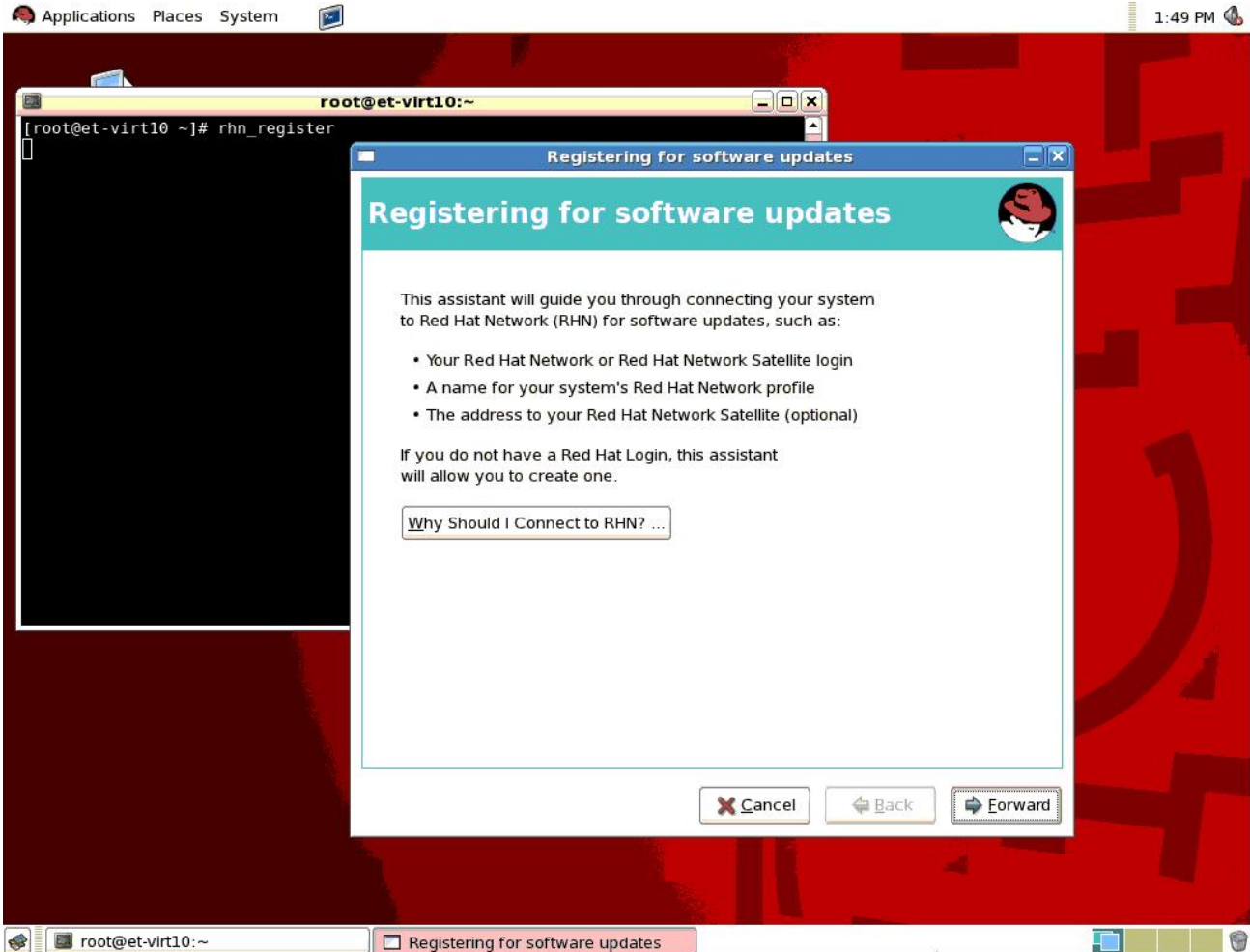
If the server was registered for RHN updates immediately after the OS installation, there is no need to perform the procedures in this section.

### Manual Configuration

If the user opted to not register the server for software updates after the OS installation procedures, the same can still be accomplished. To do so, RHN must first know something about the server by executing the RHN registration utility found under the Applications Menu (System Tools ⇒ Software Updater) or start the application using the `rhn_register` command.

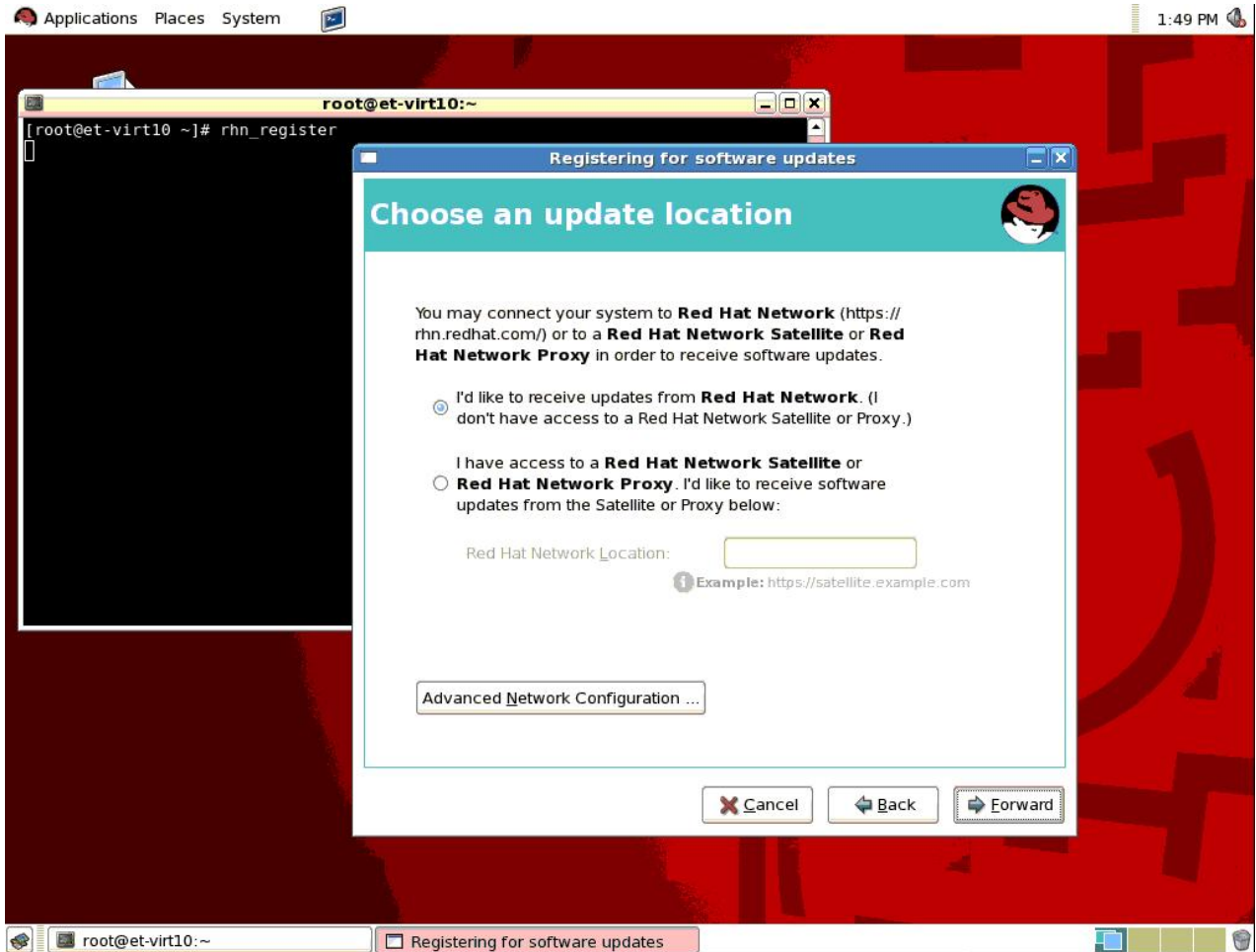
```
# rhn_register
```

which will begin the same procedure as observed when registering after OS installation.



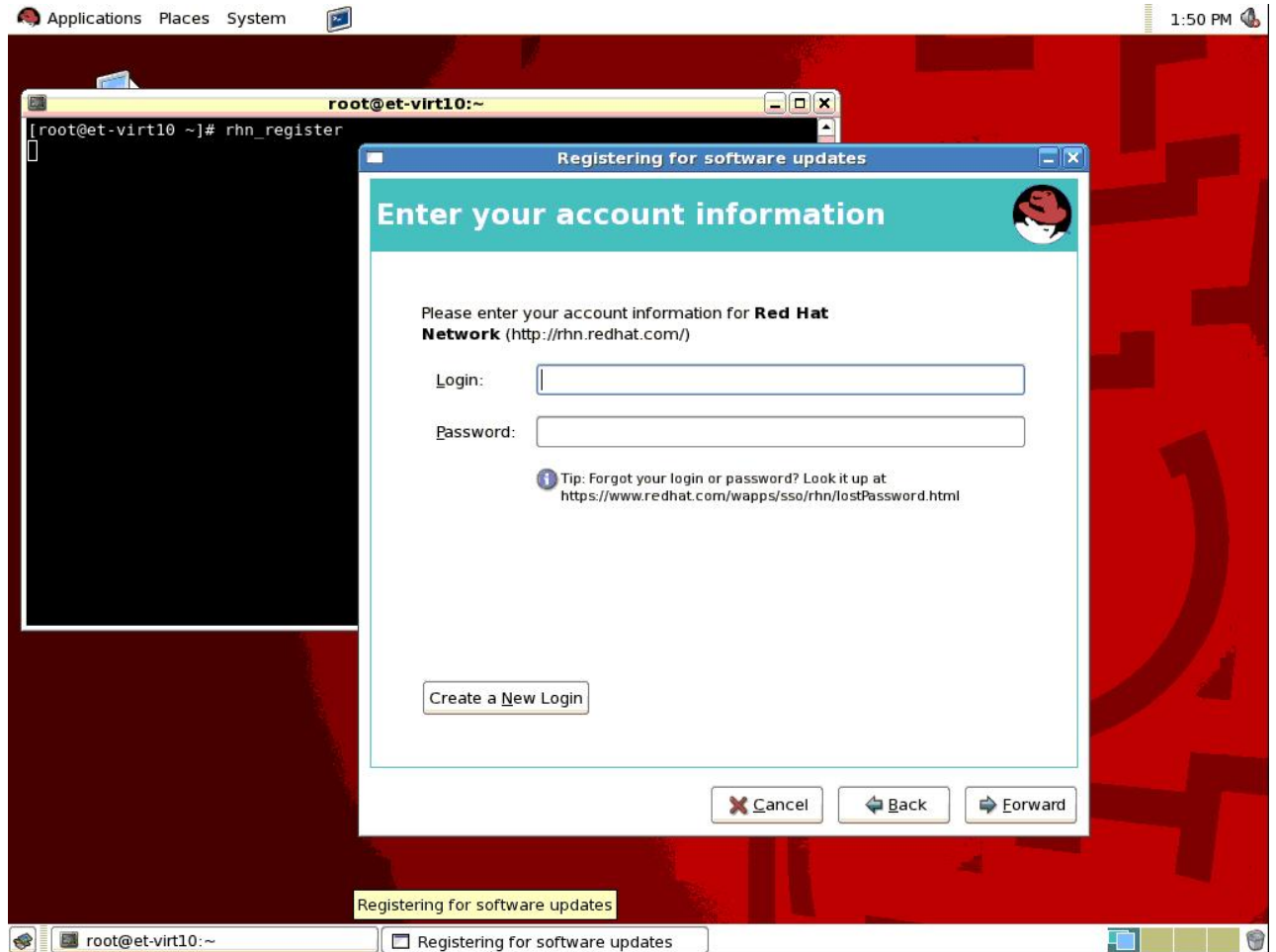
See the *Software Updates (RHN Configuration)* section in this document for information regarding satellite servers.

For this effort, the cluster members were configured to receive updates from RHN directly. By choosing so, the user is then given the option to enter any proxy information that may be necessary for internal server to access the external network.

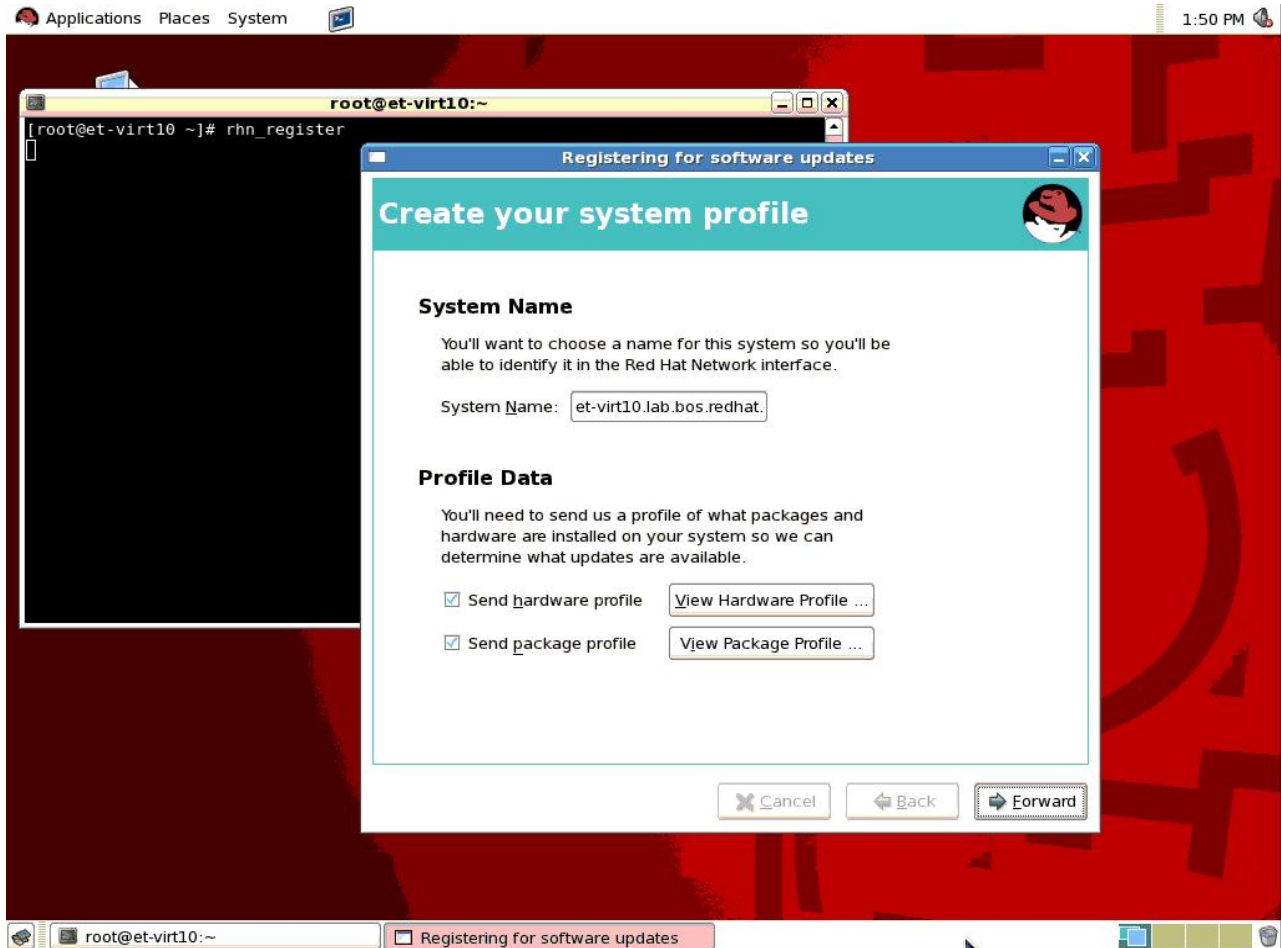


Enter any necessary proxy information if applicable.

When prompted in the next screen, enter your RHN credentials.



This will direct the user to the Profile Creation window where the server name should already be present in the System Name field.

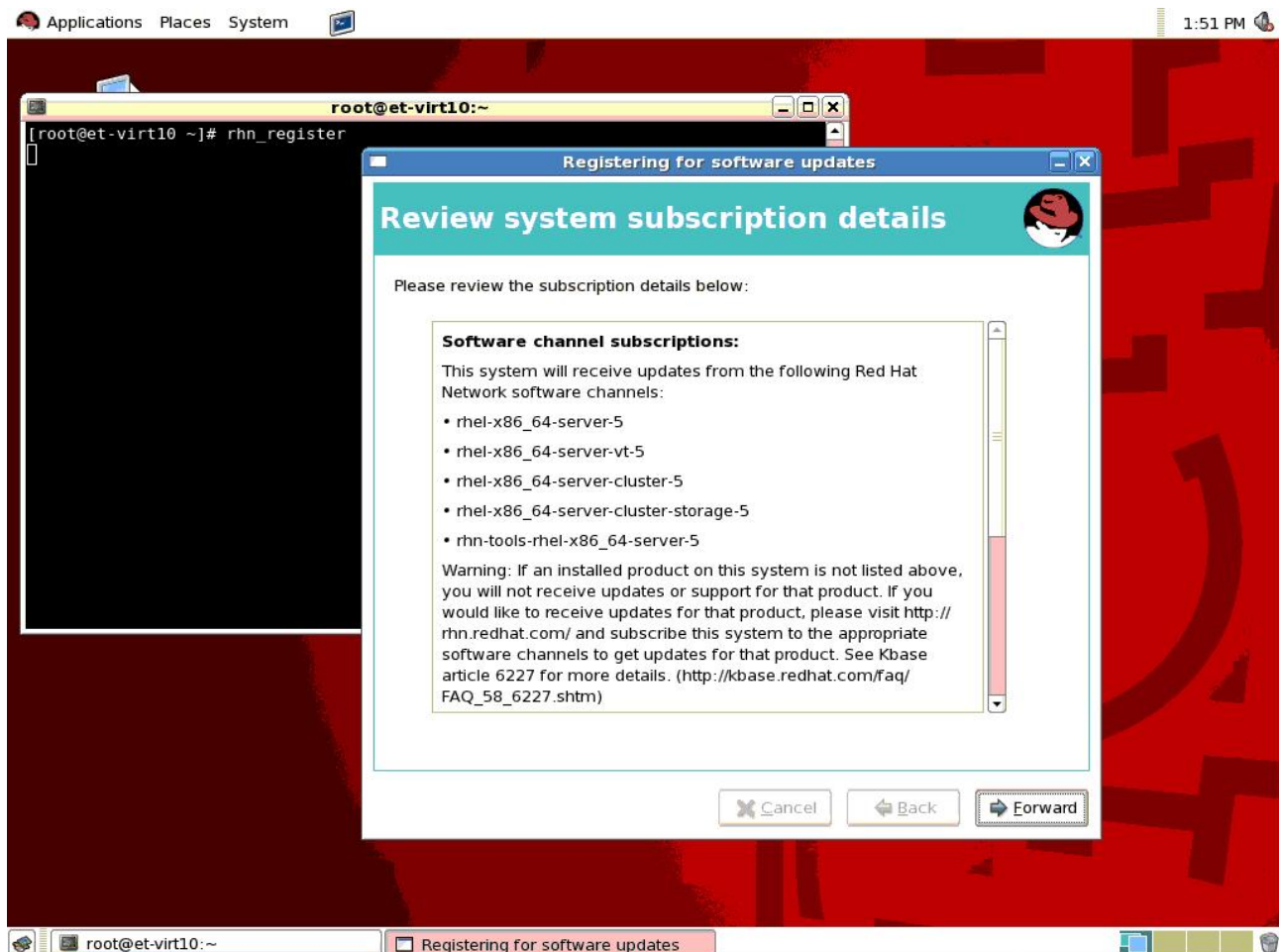


Choose whether or not to send a snapshot of the server hardware and/or package profiles and proceed.





Once completed, the Subscription Detail Review window will be displayed listing the software channel subscriptions applied to the server.



Remember that in this example, an installation number was used during the OS installation. As a result, the example above lists the channel subscriptions that will provide the necessary updates for the OS as well as for the package groups associated with the Advanced Platform configuration (Clustering, Cluster Storage, Virtualization). Above please note that the subscription to `rhel-x86_64-server-5` rhel should be present on the most basic of Red Hat Enterprise Linux installations. For the purpose of web serving from a Red Hat cluster, verify that that the following channels are subscribed:

- `rhn-tools-rhel-x86_64-server-5`
- `rhel-x86_64-server-cluster-5`
- `rhel-x86_64-server-cluster-storage-5`

Another method of checking server subscription channels and their contents is via the command line using `yum`. Below reference an example of this check on a server that has used the Advanced Platform installation number.



```
# yum grouplist
Loading "rhnplugin" plugin
Loading "security" plugin
Setting up Group Process
rhel-x86_64-server-5      100% |=====| 1.4 kB    00:00
rhel-x86_64-server-cluste 100% |=====| 1.4 kB    00:00
rhel-x86_64-server-cluste 100% |=====| 1.4 kB    00:00
rhel-x86_64-server-vt-5   100% |=====| 1.4 kB    00:00
rhn-tools-rhel-x86_64-ser 100% |=====| 1.2 kB    00:00
Installed Groups:
  Cluster Storage
  Office/Productivity
  Editors
  System Tools
  Text-based Internet
  Virtualization
  Legacy Network Server
  GNOME Desktop Environment
  Network Servers
  Games and Entertainment
  Legacy Software Development
  Clustering
  X Window System
  Graphics
  Web Server
  Printing Support
  Mail Server
  Server Configuration Tools
  Sound and Video
  Administration Tools
  Graphical Internet
Available Groups:
  Engineering and Scientific
  MySQL Database
  Development Libraries
  GNOME Software Development
  X Software Development
  DNS Name Server
  Authoring and Publishing
  FTP Server
  Java Development
  Windows File Server
  KDE Software Development
  KDE (K Desktop Environment)
  PostgreSQL Database
  News Server
  Development Tools
Done
```

Now compare this output to the same check on a server that has not used an installation number. Note that it checks only one channel as opposed to the above example that included update support for the additional package groups.



```
# yum grouplist
Loading "rhnplugin" plugin
Loading "security" plugin
Setting up Group Process
rhel-x86_64-server-5      100% |=====| 1.4 kB    00:00
Installed Groups:
  Office/Productivity
  Editors
  System Tools
  Text-based Internet
  Legacy Network Server
  GNOME Desktop Environment
  Network Servers
  Games and Entertainment
  Legacy Software Development
  X Window System
  Graphics
  Printing Support
  Mail Server
  Server Configuration Tools
  Sound and Video
  Administration Tools
  Graphical Internet
Available Groups:
  Engineering and Scientific
  MySQL Database
  Development Libraries
  GNOME Software Development
  X Software Development
  DNS Name Server
  Authoring and Publishing
  FTP Server
  Java Development
  Windows File Server
  Web Server
  KDE Software Development
  KDE (K Desktop Environment)
  PostgreSQL Database
  News Server
  Development Tools
Done
```

If the three channels listed above are not subscribed, then we will need to add them using the RHN Subscription Management page for your account in the next section, *Modifying RHN Subscriptions*. If the required channels are already subscribed, no further RHN related configurations are necessary.

## ***Modifying Subscriptions***

The appropriate installation number will include the Clustering and Cluster Storage package groups but the default OS installation does not. If they were not automatically included at the



time of OS installation, the Red Hat Cluster software can be installed manually. This will require manual modification of the RHN subscriptions for each server in the cluster.

Log into RHN where you will be directed to the page entitled *Your RHN* as seen below.

RED HAT NETWORK

LOGGED IN: SIGN OUT

Your RHN Systems Errata Channels Schedule Help

Systems Search NO SYSTEMS SELECTED Manage Clear

Your RHN Your Account Your Preferences Locale Preferences

**DOWNLOAD SOFTWARE**

**Red Hat Customer Center**  
For Subscription Management & Customer Support

**Your RHN Legend**

- OK
- Critical
- Warning
- Unknown
- Locked
- Kickstarting
- Pending Actions
- Failed Actions
- Completed Actions
- Security

**Tasks**

- Search for: Packages | Systems
- Register Systems

**Inactive Systems**

**No inactive systems.**

All of your systems are actively checking into RHN at this time. You can view a list of all of your systems at [Systems > All](#).

**Most Critical Systems**

**No critical systems.**

None of your systems are in a critical state.

**Recently Scheduled Actions**

**No recently scheduled actions.**

You have scheduled no actions within the past thirty days. You may view a list of past completed actions at [Schedule > Completed Actions](#) and a list of past failed actions at [Schedule > Failed Actions](#).

**Relevant Security Errata**

**No relevant security errata.**

There are no security errata that apply to your systems. You can view a list of **all** errata for the software your



Select 'Systems' in the red toolbar at the top of the page.

Select 'Systems' in the gray menubar on the left side of the page to view a list of the systems managed by the RHN account.

The screenshot shows the Red Hat Network (RHN) interface. At the top, there is a red navigation bar with 'Your RHN', 'Systems', 'Errata', 'Channels', 'Schedule', and 'Help'. Below this is a search bar with 'Systems' selected and a search button. The main content area is titled 'Systems' and shows a list of systems. The list has columns for 'Updates', 'Errata', 'Packages', 'System', 'Base Channel', and 'Entitlement'. Five systems are listed, each with a checkbox in the 'Updates' column that is checked. The systems are: et-virt08.lab.bos.redhat.com, et-virt09.lab.bos.redhat.com, milo.lab.bos.redhat.com, monet.lab.bos.redhat.com, and renoir.lab.bos.redhat.com. All systems are running Red Hat Enterprise Linux (v. 5 for 64-bit x86\_64). The 'Entitlement' column shows 'Management, Virtualization Platform' for the first two and 'Management' for the others. At the bottom of the list, there are buttons for 'Update List', 'Select All', and 'Unselect All'. The page also features a sidebar with navigation options like 'Overview', 'Systems', 'All', 'Virtual Systems', etc., and a 'Red Hat Customer Center' banner.

Updates	Errata	Packages	System	Base Channel	Entitlement
<input checked="" type="checkbox"/>	0	0	<a href="http://et-virt08.lab.bos.redhat.com">et-virt08.lab.bos.redhat.com</a>	Red Hat Enterprise Linux (v. 5 for 64-bit x86_64)	Management, Virtualization Platform
<input checked="" type="checkbox"/>	0	0	<a href="http://et-virt09.lab.bos.redhat.com">et-virt09.lab.bos.redhat.com</a>	Red Hat Enterprise Linux (v. 5 for 64-bit x86_64)	Management, Virtualization Platform
<input checked="" type="checkbox"/>	0	0	<a href="http://milo.lab.bos.redhat.com">milo.lab.bos.redhat.com</a>	Red Hat Enterprise Linux (v. 5 for 64-bit x86_64)	Management
<input checked="" type="checkbox"/>	0	0	<a href="http://monet.lab.bos.redhat.com">monet.lab.bos.redhat.com</a>	Red Hat Enterprise Linux (v. 5 for 64-bit x86_64)	Management, Virtualization Platform
<input checked="" type="checkbox"/>	0	0	<a href="http://renoir.lab.bos.redhat.com">renoir.lab.bos.redhat.com</a>	Red Hat Enterprise Linux (v. 5 for 64-bit x86_64)	Management

Select the system name to view the subscription and entitlement details for that server.



Your RHN Systems Errata Channels Schedule Help

Systems [ ] Search NO SYSTEMS SELECTED [ MANAGE | CLEAR ]

et-virt08.lab.bos.redhat.com delete system

Overview Systems All Virtual Systems Out of Date Unentitled Inactive Recently Registered System Groups System Set Manager Advanced Search Stored Profiles Custom System Info

DOWNLOAD SOFTWARE

Red Hat Customer Center For Subscription Management & Customer Support

System Status

**Critical Updates Available (update now)**

- Critical: firefox security update
- Critical: firefox security update
- Critical: firefox security update

System Info

Hostname: et-virt08.lab.bos.redhat.com

IP Address: 10.16.41.71

Kernel: 2.6.18-92.1.6.el5

RHN System ID: 1013156553

Lock Status: System is **unlocked** (Lock system)

System Events

Checked In: 11/13/08 1:07:58 PM EST

Registered: 7/9/08 2:23:00 PM EDT

Last Booted: 11/12/08 2:05:33 PM EST (Schedule System Reboot)

System Properties (Edit These Properties)

Entitlements: [Management]

Notifications: Daily Summary Errata Email

Auto Errata Update: No

System Name: et-virt08.lab.bos.redhat.com

Description: Initial Registration Parameters: OS: redhat-release Release: 5Server CPU Arch: x86\_64-redhat-linux

Location: (none)

Subscribed Channels (Alter Channel Subscriptions)

- Red Hat Enterprise Linux (v. 5 for 64-bit x86\_64)

Note the *Subscribed Channels* and *Entitlements* sections. Note that a system not using any installation numbers during OS install has one entitlement entry (Management) and one associated channel (in this case, Red Hat Enterprise Linux 5 for x86\_64) to facilitate the software updates.

Click the the 'Alter Channel Subscriptions' link to view all the available subscription channels.



### Software Channel Subscriptions

This system is subscribed to the base channel, listed at top, and to the checked channels beneath, if any. Disabled checkboxes indicate channels that can't be manually subscribed or unsubscribed from.

#### Release Channels for Red Hat Enterprise Linux 5 for x86\_64

- RHEL FasTrack (v. 5 for 64-bit x86\_64) [Info](#) 509 open entitlements
- RHEL Optional Productivity Apps (v. 5 for 64-bit x86\_64) [Info](#) 508 open entitlements
- RHEL Supplementary (v. 5 for 64-bit x86\_64) [Info](#) 502 open entitlements
- RHEL Virtualization (v. 5 for 64-bit x86\_64) [Info](#) 484 open entitlements
- Red Hat Network Tools for RHEL Server (v.5 64-bit x86\_64) [Info](#) 1241 open entitlements

#### BETA Channels for Red Hat Enterprise Linux 5 for x86\_64

- RHEL Optional Productivity Apps (v. 5 for 64-bit x86\_64) Beta [Info](#) 510 open entitlements
- RHEL Supplementary (v. 5 for 64-bit x86\_64) Beta [Info](#) 510 open entitlements
- RHEL Virtualization (v. 5 for 64-bit x86\_64) Beta [Info](#) 510 open entitlements
- Red Hat Enterprise Linux (v. 5 for 64-bit x86\_64) Beta [Info](#) 506 open entitlements

#### Additional Services Channels for Red Hat Enterprise Linux 5 for x86\_64

- Alfresco Enterprise 2.0 (for RHEL Server v.5 x86\_64) [Info](#) 249 open entitlements
- Amanda Enterprise Backup Server 2.6 (for RHEL Server v.5 x86\_64) [Info](#) 250 open entitlements
- CentricCRM 4.1 (for RHEL Server v.5 x86\_64) [Info](#) 250 open entitlements
- Compiere Enterprise 2.6 (for RHEL Server v.5 x86\_64) [Info](#) 250 open entitlements
- EnterpriseDB Advanced Server 8.1 (for RHEL Server v.5 x86\_64) [Info](#) 250 open entitlements
- GroundWork Monitor for RHX (for RHEL Server v.5 x86\_64) [Info](#) 250 open entitlements
- JBEAP (v 4.3.0) for 5Server x86\_64 [Info](#) 208 open entitlements
- JBoss Enterprise Application Platform (v 4) for 5Server x86\_64 [Info](#) 208 open entitlements
- JasperServer Pro RHX Ed. 1.2 (for RHEL Server v.5 x86\_64) [Info](#) 250 open entitlements
- MRG Grid v. 1 (for RHEL 5 Server 64-bit x86\_64) [Info](#) 248 open entitlements
- MRG Management v. 1 (for RHEL 5 Server 64-bit x86\_64) [Info](#) 248 open entitlements
- MRG Messaging Base v. 1 (for RHEL 5 Server 64-bit x86\_64) [Info](#) 248 open entitlements
- MRG Messaging v. 1 (for RHEL 5 Server 64-bit x86\_64) [Info](#) 248 open entitlements
- MRG Realtime v. 1 (for RHEL 5 Server 64-bit x86\_64) [Info](#) 248 open entitlements
- MySQL Enterprise 5 (for RHEL Server v.5 x86\_64) [Info](#) 250 open entitlements
- OpenFire Enterprise 3.3 (for RHEL Server v.5 x86\_64) [Info](#) 250 open entitlements
- Pentaho Reporting Pack for RHX 1.2 (for RHEL Server v.5 x86\_64) [Info](#) 250 open entitlements
- RHEL Cluster-Storage (v. 5 for 64-bit x86\_64) [Info](#) 491 open entitlements
- RHEL Clustering (v. 5 for 64-bit x86\_64) [Info](#) 501 open entitlements
- RHEL Hardware Certification (v. 5 for 64-bit x86\_64) [Info](#) 510 open entitlements
- Red Hat Application Stack v2 (for v. 5 AMD64/EM64T) [Info](#) 250 open entitlements
- Red Hat Certificate System 7.3 (for RHEL 5 for 64-bit x86\_64) [Info](#) 24996 open entitlements
- Red Hat Directory Server 8 (for RHEL 5 for 64-bit x86\_64) [Info](#) 246 open entitlements
- Red Hat Directory Server 8 (for RHEL 5 for 64-bit x86\_64) Beta [Info](#) 249 open entitlements

Recall that for the purpose of web serving from a Red Hat cluster, the following channels are required:

- Red Hat Network Tools for Red Hat Enterprise Linux Server (rhn-tools-rhel-x86\_64-server-5)
- Red Hat Enterprise Linux Cluster-Storage (rhel-x86\_64-server-cluster-5)
- Red Hat Enterprise Linux Clustering (rhel-x86\_64-server-cluster-storage-5)

Select the check boxes next to each of the three package groups (as seen in the example above) and proceed by clicking the 'Change Subscriptions' button at the page bottom which will return the user to detail overview of the node. Note that the Subscribed Channels section now includes the desired subscriptions.



Your RHN Systems Errata Channels Schedule Help

Systems [ ] Search [ ] NO SYSTEMS SELECTED [ MANAGE | CLEAR ]

Overview Systems et-virt08.lab.bos.redhat.com delete system

Details Software Virtualization Groups Events

Overview Properties Hardware Notes

**System Status**

**Critical Updates Available (update now)**

- Critical: firefox security update
- Critical: firefox security update
- Critical: firefox security update

**System info**

**Hostname:** et-virt08.lab.bos.redhat.com

**IP Address:** 10.16.41.71

**Kernel:** 2.6.18-92.1.6.el5

**RHN System ID:** 1013156553

**Lock Status:** System is **unlocked** (Lock system)

**System Events**

**Checked In:** 11/13/08 1:07:58 PM EST

**Registered:** 7/9/08 2:23:00 PM EDT

**Last Booted:** 11/12/08 2:05:33 PM EST (Schedule System Reboot)

**System Properties (Edit These Properties)**

**Entitlements:** [Virtualization Platform] [Management]

**Notifications:** Daily Summary Errata Email

**Auto Errata Update:** No

**System Name:** et-virt08.lab.bos.redhat.com

**Description:** Initial Registration Parameters: OS: rhel-release Release: 5Server CPU Arch: x86\_64-redhat-linux

**Location:** (none)

**Subscribed Channels (Alter Channel Subscriptions)**

- Red Hat Enterprise Linux (v. 5 for 64-bit x86\_64)
  - RHEL Clustering (v. 5 for 64-bit x86\_64)
  - RHEL Virtualization (v. 5 for 64-bit x86\_64)
  - Red Hat Network Tools for RHEL Server (v.5 64-bit x86\_64)
  - RHEL Cluster-Storage (v. 5 for 64-bit x86\_64)

**DOWNLOAD SOFTWARE**

**Red Hat Customer Center**

For Subscription Management & Customer Support

Ensure that the same procedure is performed for all servers intended to participate in the cluster.

Now the user will have the appropriate channel support for the cluster members when installing the Red Hat Cluster software.

## Appendix D: Issue Tracking

The following issues, and any applicable workaround(s), observed during this testing are listed below.

- ◆ “The ricci agent for this node is unresponsive. Node-specific information is not available at this time.”

This error message is observed in luci when attempting to communicate with the ricci agent on cluster members. Upon closer examination, an underlying cman error was observed in the luci progress window during the cluster creation. That error is described below.

- ◆ “cman not started: Local host name resolves to 127.0.0.1; fix /etc/hosts before starting cluster. /usr/sbin/cman\_tool: aisexec daemon didn't start”

This error message was caused by a bad entry in the */etc/hosts* file.

```
127.0.0.1 et-virt08.lab.bos.redhat.com et-virt08 localhost.localdomain localhost
```





Remove the hostname references from this line as described in the */etc/hosts* section of this document.

- ◆ **Adding/removing qdisk to existing cluster fails to update cman entry in cluster.conf**

Red Hat Bugzilla #467464

When adding or removing a quorum device to an existing cluster, conga fails to update the *cman* XML tag in the cluster.conf file. Workaround is to manually edit the cluster.conf file

- ◆ **Leaked file descriptors**

Red Hat Bugzilla #461943

Code starting *ifconfig\_t*, *netutils\_t*, *ping\_t*, and *rdisk\_t* is leaking a file descriptor causing SELinux to continuously flag them in console unless instructed to not audit them (*dontaudit*). Workaround is each domain needs to *dontaudit* these messages or simply ignore them in console.

## Appendix E: Procedure Checklist

The following checklist should be used to verify that each of the steps outlined in this document have occurred and in the correct sequence.

	Hardware Configuration & Cabling
	Ensure Multicast Enabled
	Installation of OS (installation numbers)
	Configure Public NIC (during OS install)
	Include WEB Server Package (during OS install)
	RHN Registration & Subscriptions
	Reboot after OS Install
	Enable Firewall
	Enable SELinux
	Configure Secure Shell
	Disable ACPI
	Define Firewall Rules
	Configure SELinux Booleans & Labeling



	SELinux in Permissive Mode
	Configure Private Networks (NIC Bonding)
	Configure DM Multipathing
	Configure CLVM Volumes (Shared Storage)
	Configure /etc/hosts files (local LANs, luci server)
	Install/Enable/Start ricci
	Install/Enable/Start luci
	Create Cluster
	Configure Quorum Device
	Configure Fence Devices
	Configure Failover Domain
	Configure httpd Script Resource
	Configure IP Address Resource
	Create httpd Service
	Edit HTTP Directives
	Test HTTP Functionality
	Audit SELinux Policy
	SELinux In Enforcing Mode