



Red Hat Reference Architecture Series

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

Volume 1: NFS Web Content

Version 1.1

September 2008





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

Volume 1: NFS Web Content

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 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

<u>1</u>	<u>Executive Summary.....</u>	<u>5</u>
1.1	Introduction.....	5
1.2	Audience.....	5
1.3	Reference Documentation.....	5
1.1	Document Conventions.....	6
1.4	Acronyms.....	6
<u>2</u>	<u>Hardware Configuration.....</u>	<u>7</u>
2.1	Environment	7
2.2	Multicast.....	8
<u>3</u>	<u>OS Installation.....</u>	<u>8</u>
3.1	Installation Numbers.....	9
3.2	Additional Package Groups.....	16
<u>4</u>	<u>First Boot.....</u>	<u>17</u>
4.1	Initial Settings.....	18
4.1.1	Firewall.....	18
4.1.2	SELinux.....	19
4.2	Software Updates (RHN Configuration).....	20
<u>5</u>	<u>OS Customization.....</u>	<u>29</u>
5.1	Secure Shell.....	29
5.2	ACPI.....	31
5.3	Firewall (iptables) Rules.....	31
5.3.1	Modifying.....	32
5.3.2	Saving.....	34
5.4	SELinux.....	35
5.4.1	Booleans.....	37
5.4.2	Labeling.....	38
5.4.3	GUI.....	39
5.5	Public and Private Networks.....	40
5.6	Network Interface Bonding.....	41
5.7	/etc/hosts.....	44



6	Conga	45
6.1	Installing ricci	46
6.2	Installing luci	48
7	Clustering	50
7.1	Cluster Creation	52
7.2	Configuring Cluster Members	57
7.3	Fencing	58
7.4	Failover Domains	59
7.5	Cluster Resources	61
7.5.1	Script	61
7.5.2	IP Address	62
7.5.3	NFS Mount	64
7.6	Web Service (httpd)	67
7.6.1	Service Creation	69
7.6.2	httpd Configuration Directives	73
7.6.3	Testing	74
8	SELinux Policy Adjustments	75
8.1	AVC Denials	75
8.2	audit2allow	77
9	Diagnostics	80
9.1	clustat	80
9.2	Logs	80
10	Conclusions & Next Steps	80
Appendices		81
Appendix A: Using Local Web Content		81
Appendix B: Configuration Files		81
	cluster.conf	81
	iptables	83
	Network Interfaces	84
Appendix C: RHN		84
	Manual Configuration	84
	Modifying Subscriptions	91
Appendix D: Issue Tracking		96
Appendix E: Procedure Checklist		100



1 Executive Summary

This paper details the deployment of a highly available web service on a Red Hat Enterprise Linux 5 cluster. This volume will focus on the configuration of a 2-node cluster and the management of a web server using the Red Hat Cluster Suite. Subsequent volumes will include shared storage, Global File System, qdisk heuristics, command line cluster creation and maintenance.

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 high availability aspect of any cluster service indicates that it is configured in a manner such that the failure of any one cluster member, or even a subsystem failure within a member, will not prevent the continued availability of the service itself. We 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 Red Hat Enterprise Linux 5.2 documents were used explicitly for the operating system installation and cluster creation procedures.

- Red Hat Enterprise Linux Installation Guide
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
- 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
- Cluster Project Frequently Asked Questions (FAQ)
<http://sources.redhat.com/cluster/faq.html>



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 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"  
This is an example of command line input and output  
#
```

- **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
CSSD	Cluster Services Synchronization Daemon
DLM	Distributed Lock Manager
DRAC	Dell Remote Access Controller
GNBD	Global Network Block Device
HA	High Availability
ILO	Integrated Lights Out
IPMI	Intelligent Platform Management: Interface
EULA	End User License Agreement
GFS	Global File System



HTTP	Hypertext Transfer Protocol
HTTPD	Hypertext Transfer Protocol Daemon
IP	Internet Protocol
RHCS	Red Hat Cluster Suite
RHEL	Red Hat Enterprise Linux
RHN	Red Hat Network
OS	Operating System

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.

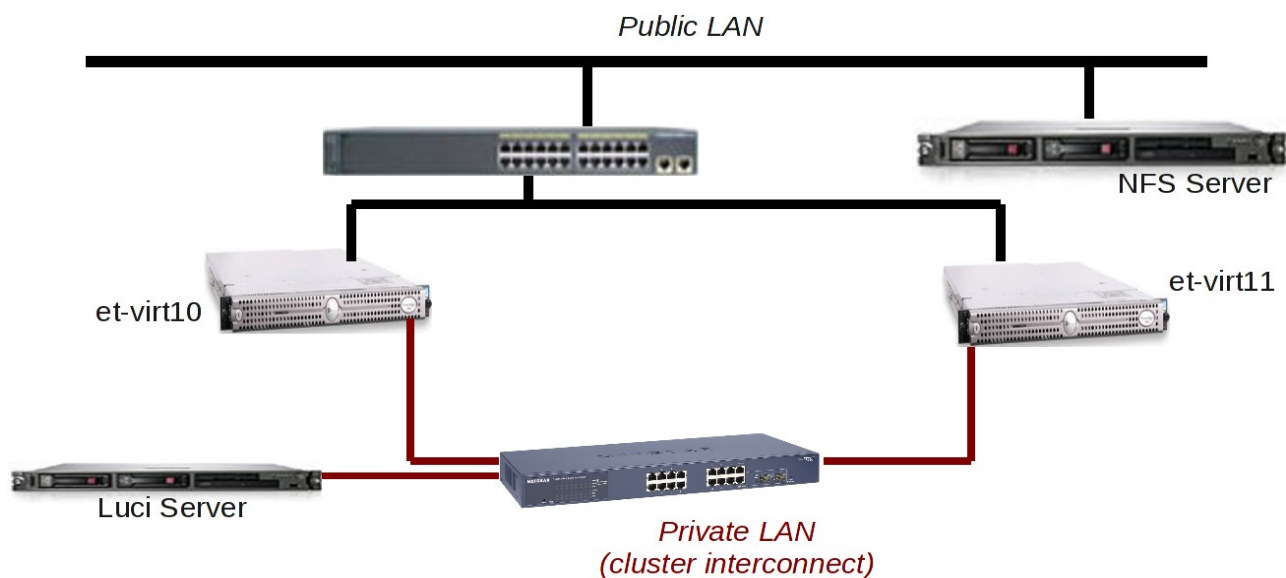
The following table describes the primary components of the test environment.

System (et-virt10)	Dell PowerEdge 1850 RHEL 5.2 (2.6.18-92.1.10.el5) (4) Intel ^(R) Xeon ^(TM) CPU 2.80GHz 4 GB RAM gigabit ethernet
System (et-virt11)	Dell PowerEdge 1850 RHEL 5.2 (2.6.18-92.1.10.el5) (4) Intel ^(R) Xeon ^(TM) CPU 2.80GHz 4 GB RAM gigabit ethernet



NFS Server	HP DL360 G3 RHEL 5.2 (2.6.18-92.el5) (2) Intel(R) Xeon(TM) CPU 3.20GHz 8 GB RAM gigabit ethernet 2G FC HBA
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

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*.



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

Your RHN

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



The screenshot shows the Red Hat Customer Center interface. The browser window title is "redhat.com - Mozilla Firefox". The address bar shows "https://www.redhat.com/wapps/support/protected/overview.html". The page header includes navigation links for Home, Solutions, Services & Products, Partners, Developers, Training, Support (highlighted), and Store. Below the header is a search bar and a navigation menu with links for Knowledgebase, Documentation, Downloads, Offerings, Support Policy, Support Process, and Customer Center.

Customer Center

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

Overview | Subscriptions | Renewals

Your Support Overview

Company: **blah**

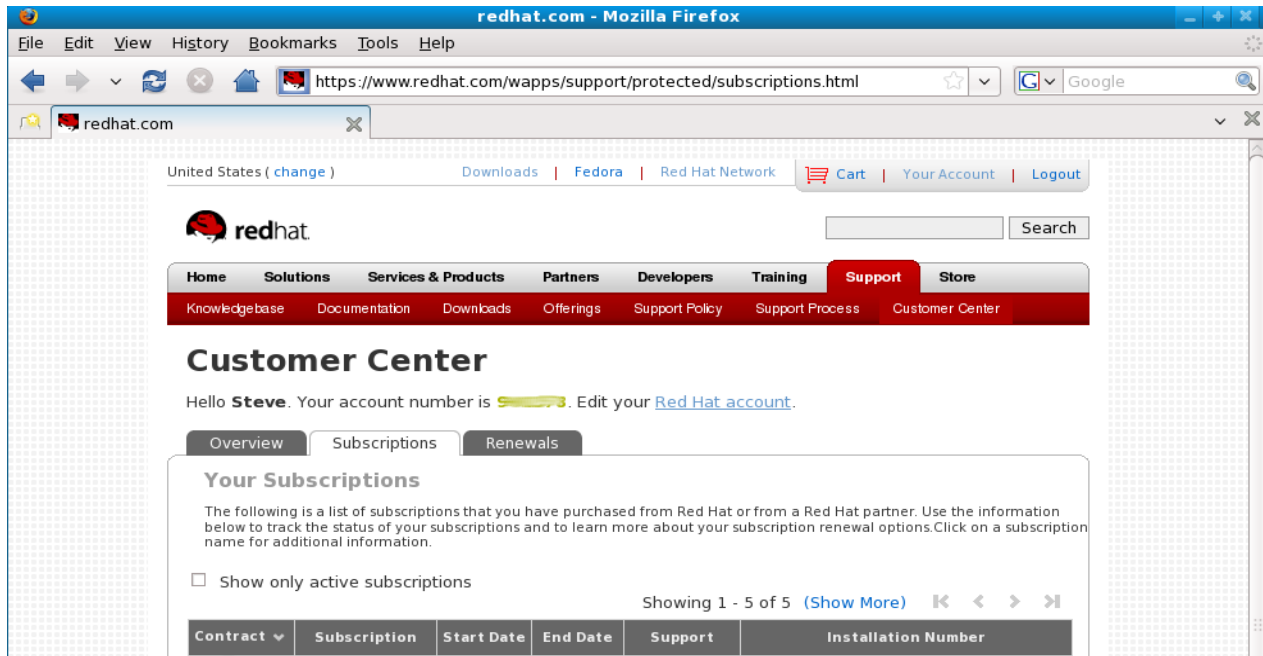
Subscriptions		Systems		Tickets	
Active Subscriptions	81	Total Systems	0 View	Open Tickets	0
Expired Subscriptions	0	Inactive Systems	0 View	Tickets Awaiting Your Response	0
Subscriptions Due to Expire	1	Out of Date Systems	0 View	You do not currently have any web support entitlements.	
Manage Subscriptions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

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](#)

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

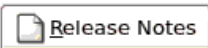
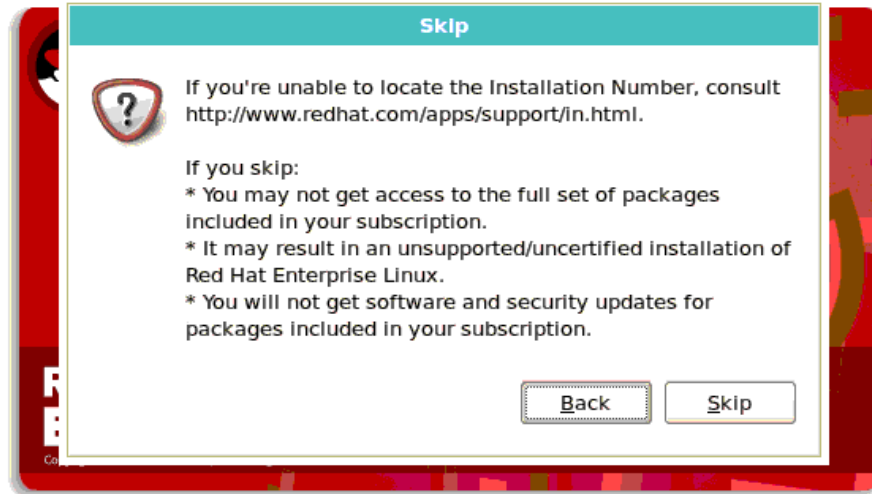


It is not necessary to use an installation number. The opportunity to select individual packages for installation will be provided at a later time. See the [Configuring and Managing a Red Hat Cluster](#) guide or the *Red Hat Cluster Suite* section of this document for instructions on manually installing 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



Note that while it is not required that users enter an installation number, using one that specifies 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.

**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?

- Clustering
- Software Development
- Storage Clustering
- Virtualization
- 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](#)

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).

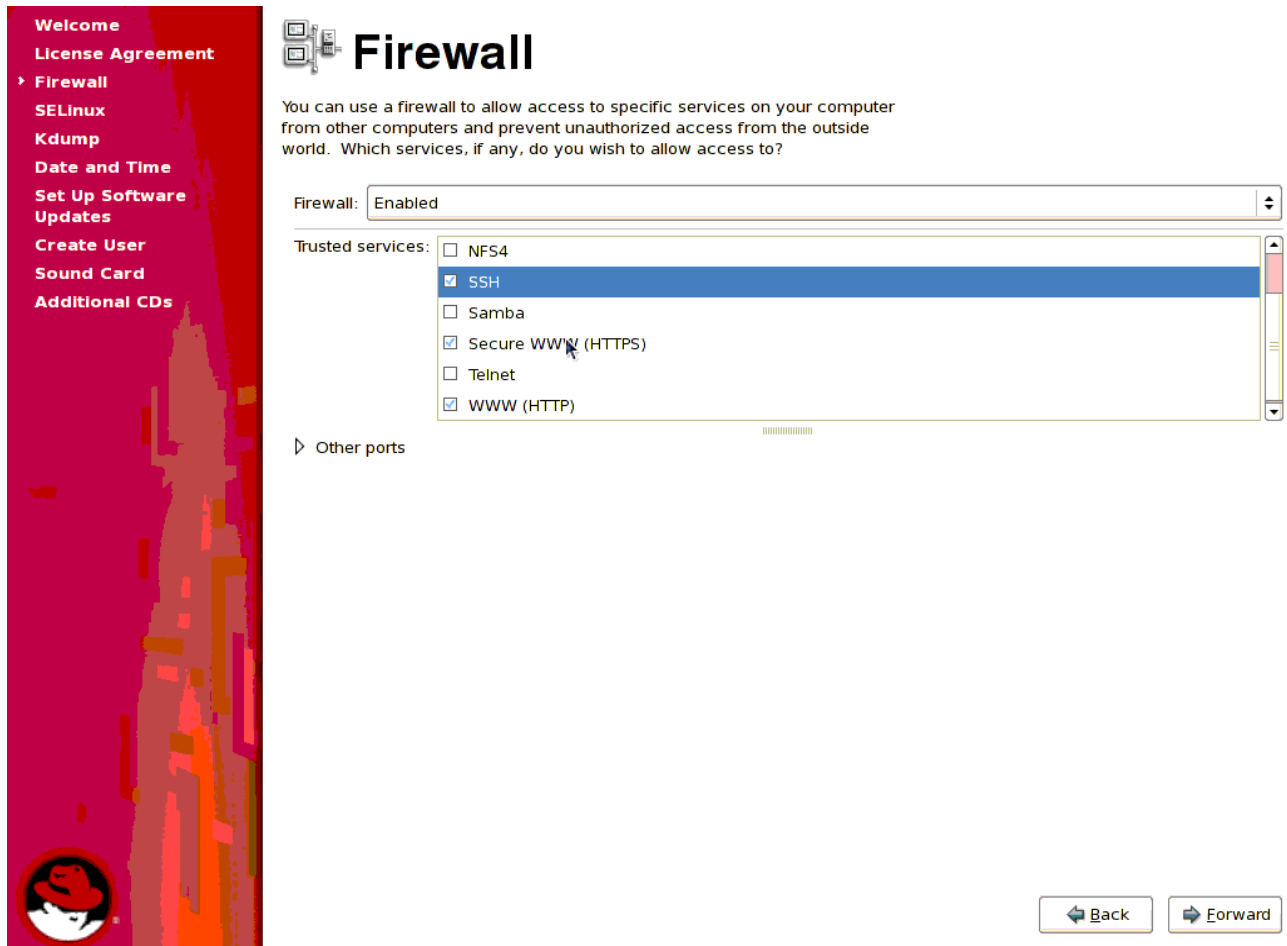
4.1 Initial Settings

4.1.1 Firewall

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.



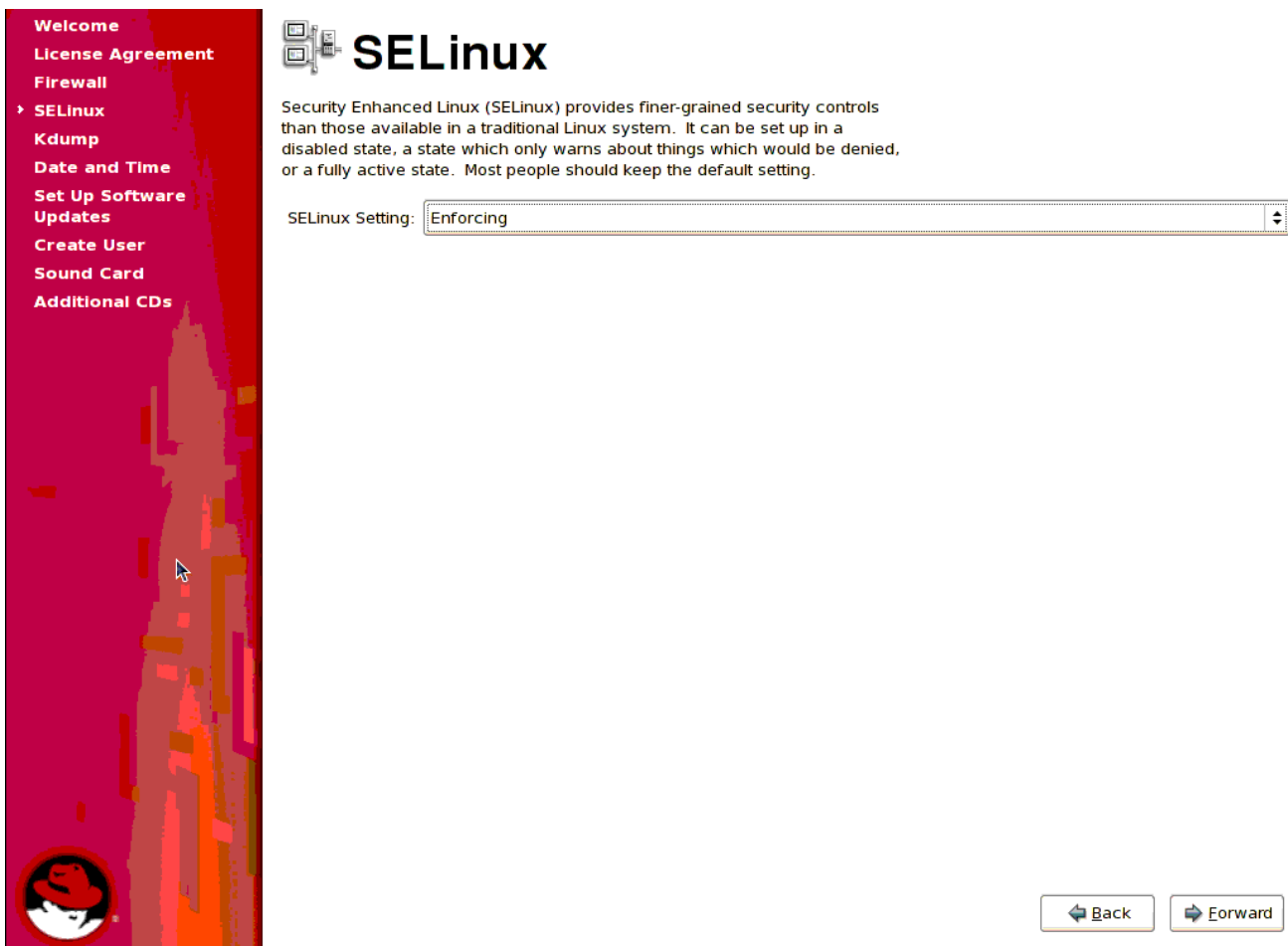
The HTTP protocol is used by Apache and by other web servers. Selecting **WWW (HTTP)** will include the httpd package automatically.

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 should be left as is.

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? ...](#)

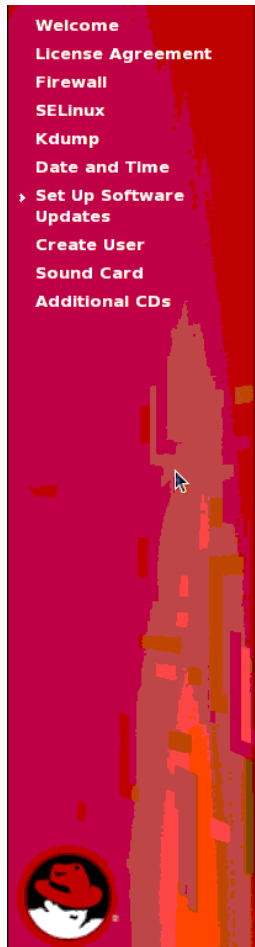
Would you like to register your system at this time?
(Strongly recommended.)

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.

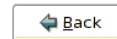
I'd like to receive updates from **Red Hat Network**. (I don't have access to a Red Hat Network Satellite or Proxy.)

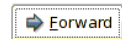
I have access to a **Red Hat Network Satellite** or **Red Hat Network Proxy**. I'd like to receive software updates from the Satellite or Proxy below:

Red Hat Network Location:

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

[Advanced Network Configuration ...](#)





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)

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)

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.



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.

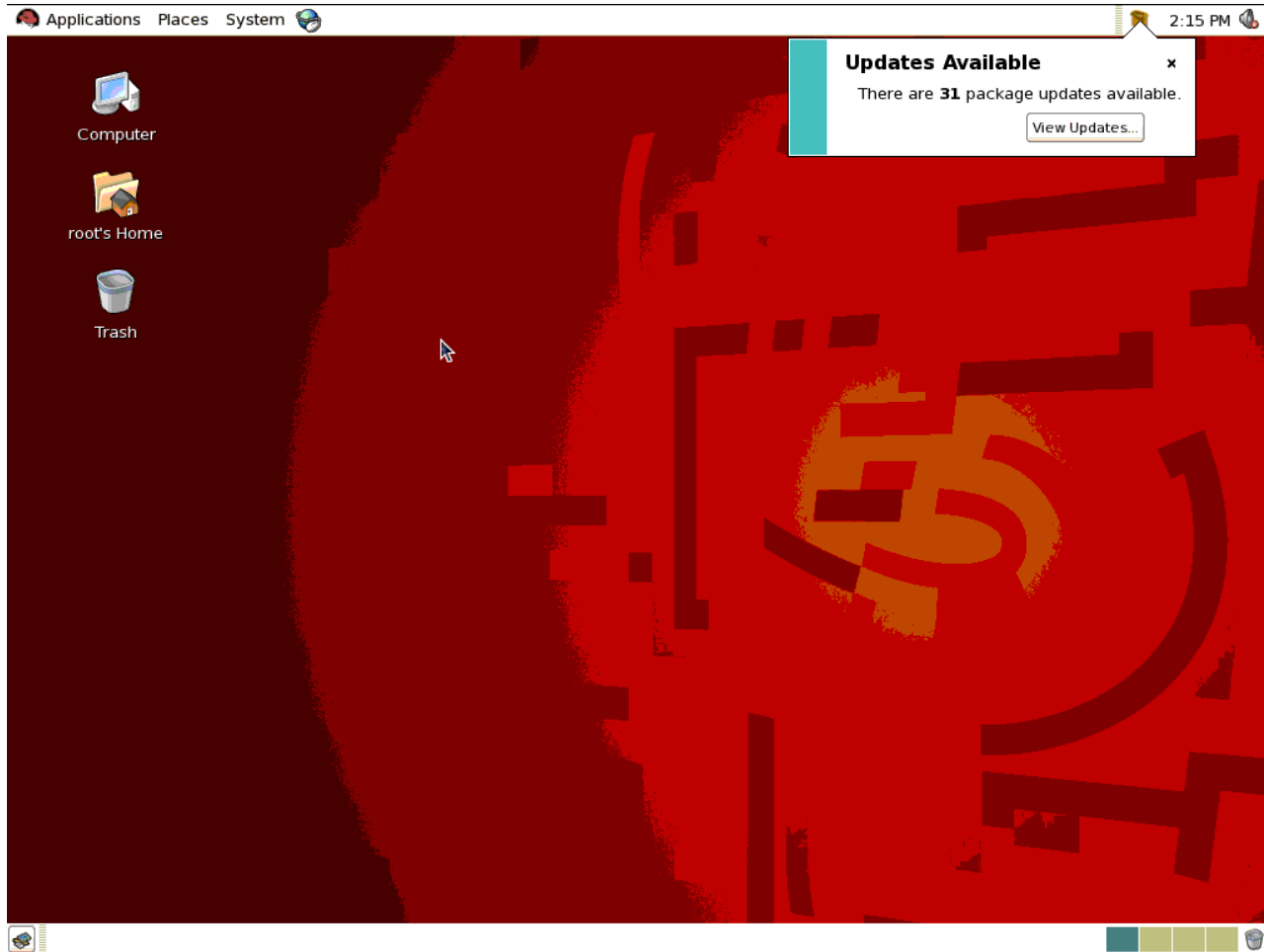
To keep your system updated, secure, and supported, please connect this system to RHN at your earliest convenience. You may access this software updates setup tool at any time by running Software Updater in the Applications > System Tools menu.

[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-virt10
#
```

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-virt11
21
root@et-virt11's password: <Enter passwd>
Now try logging into the machine, with "ssh 'et-virt11'", and check in:

    .ssh/authorized_keys

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

# ssh et-virt11
Last login: Thu Sep 11 13:19:32 2008 from et-virt10.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-virt10:~/.ssh/authorized_keys .
The authenticity of host 'et-virt10 (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-virt10' (RSA) to the list of known hosts.
root@et-virt10'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+TCleernK8SkSVNVWdnznCJU0oWOLHAAAAFQDaq/pkOJUa80zqtAnN7aGHt6q3QAAAIA0xmX0+rK2
lNvmXqaZJTaa0oKglifTKMN7b9CmAbv0+6S0meODgvU35RcOAYDYd9kCX40HgnMztTHOBKj6CICQyby
YwFxrZ64YfZ+9RxxXUNgBTEdM4WE9V2YX68UqZ66S2YCFtyk8Swr7Rz5U12DQHhR5iM7E70DQTjEU0Bd
3AAAAIBbnUxpVFMb6HiZn9XDUH9mZGRN+SWgiXrYMVMnY5Xd9LFYhb5YpKtTs j m+07SSGHH37ZiVdcHH
4gXTWs6l9t9DSGWSJ5zNd0N5sP6Rh5iCuZRfapf6TCmpCyAV+fvcvh+bFKBiwosyxE67SYZ+XpCs2j8Yz
q9PAaSVvCJld/8M5Yg== /root/.ssh/id_dsa

# ssh -l root et-virt11.lab.bos.redhat.com
Last login: Tue Jul 22 15:14:21 2008 from et-virt10.lab.bos.redhat.com

# exit
logout

Connection to et0-virt11.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-virt11
The authenticity of host 'et-virt11 (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-virt11,10.16.41.77' (RSA) to the list of known
hosts.
Last login: Tue Jul 22 15:14:21 2008 from et-virt10

# exit
logout
Connection to et-virt11 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-virt10 ...

```
# ssh et-virt11 date
Wed Aug 27 09:45:45 EDT 2008
#
```

5.2 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 "wake" from input from the keyboard, clock, [modem](#), [LAN](#), or [USB](#) device and subsequently takes longer to shutdown completely.

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
#
```

5.3 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.

5.3.1 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
- ccscd

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.

openais [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
#
```

dlm [21064]:

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

ccscd [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 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
#
```

5.3.2 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
#
```

5.4 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
#
```

5.4.1 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`. The flexible HTTPD and NFS policies allow users to configure their web services as securely as possible.

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
#
```

To view the current `selinux_nfs` settings on the command line:



```
# getsebool -a | grep nfs
allow_ftp_use_nfs --> off
allow_nfsd_anon_write --> off
nfs_export_all_ro --> on
nfs_export_all_rw --> on
nfsd_disable_trans --> off
samba_share_nfs --> off
use_nfs_home_dirs --> off
#
```

To use a remote NFS server for the served web content, use `setsebool` to modify the `use_nfs_home_dirs` boolean.

```
# setsebool -P use_nfs_home_dirs 1
#
```

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

5.4.2 Labeling

SELinux requires files to have an extended attribute to define the file type and is very sensitive regarding file and directory labeling. Access to incorrectly labeled files can and will be prevented. 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 (`/www`). 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



/www directory accordingly, use the semanage utility.

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

This tells SELinux that the /www 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 step 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 /www should be labeled and repairs the labeling as needed.

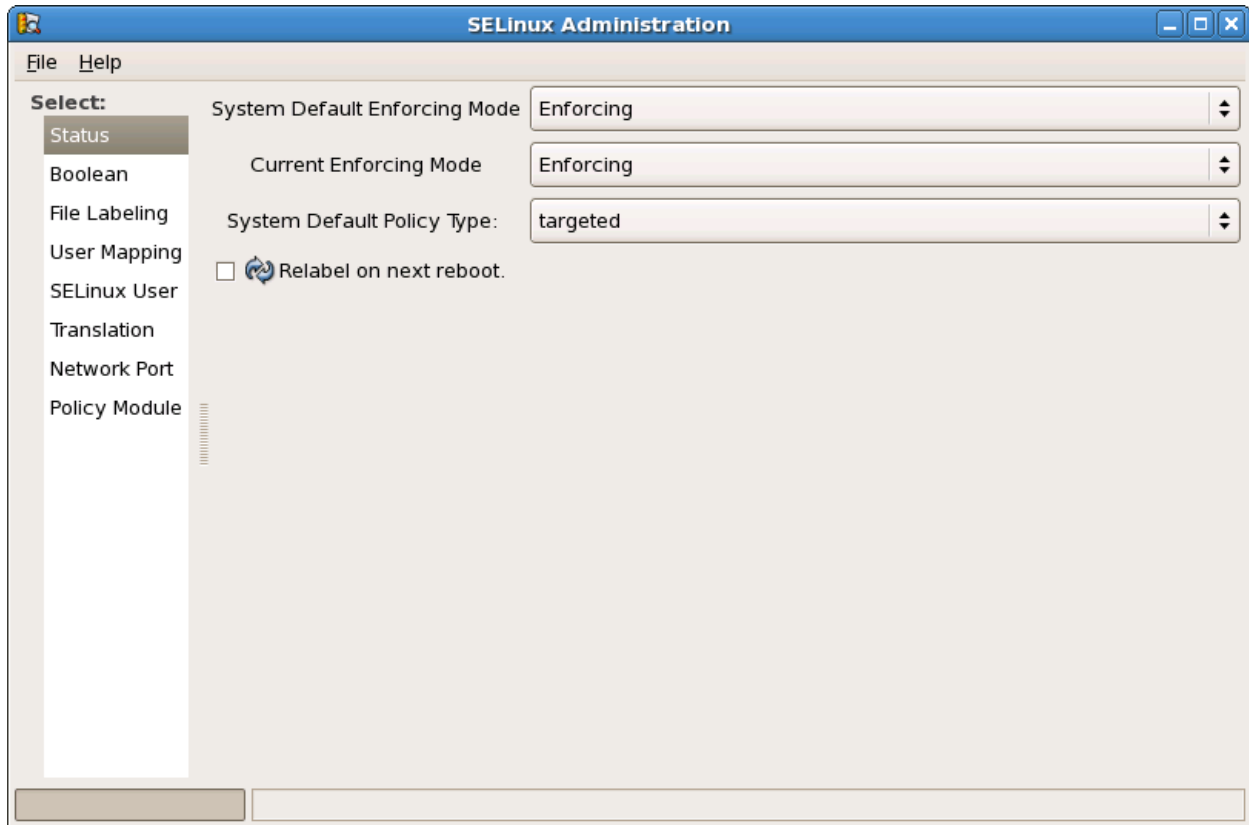
```
# restorecon -R /www  
#
```

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

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

5.4.3 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.

5.5 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:10.10.1.4 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:0xcc0 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.

5.6 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:0xcc0 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=10.10.1.4
USERCTL=no
TYPE=Ethernet
IPV6INIT=no
PEERDNS=yes
BONDING_OPTS="mode=1 miimon=100 primary=eth0"
```

In the example above, 10.10.1.4 was used for this system's local interconnect (10.10.1.3 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:
 - o SLAVE=yes
 - o 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 00:04:23:D8:03:AC
      inet addr:10.10.1.4 Bcast:10.10.1.255 Mask:255.255.255.0
      inet6 addr: fe80::204:23ff:fed8:3ac/64 Scope:Link
      UP BROADCAST RUNNING MASTER MULTICAST MTU:1500 Metric:1
      RX packets:49 errors:0 dropped:0 overruns:0 frame:0
      TX packets:99 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:0
      RX bytes:9550 (9.3 KiB) TX bytes:15568 (15.2 KiB)

eth0 Link encap:Ethernet HWaddr 00:04:23:D8:03:AC
      inet6 addr: fe80::204:23ff:fed8:3ac/64 Scope:Link
      UP BROADCAST RUNNING SLAVE MULTICAST MTU:1500 Metric:1
      RX packets:21 errors:0 dropped:0 overruns:0 frame:0
      TX packets:71 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
      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

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

.
.
.
```

5.7 */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-virt10.lab.bos.redhat.com et-virt10 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.

```
10.10.1.3 et-virt10-ic.lab.bos.redhat.com et-virt10-ic
10.10.1.4 et-virt11-ic.lab.bos.redhat.com et-virt11-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.

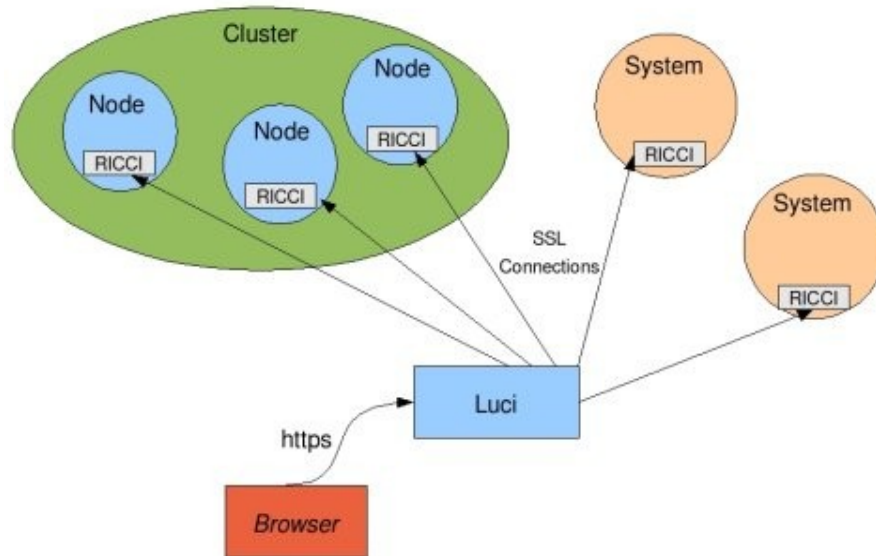
6 Conga

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.

6.1 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!
#

```

6.2 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!
#

```

7 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.



Please log in

To access this part of the site, you need to log in with your user name and password.

Account details

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

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

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

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



homebase

cluster

storage

- admin
- Add a System
- Add an Existing Cluster
- Manage Systems
- Add a User

Luci Homepage

Welcome to Luci, admin.

Select an action from the list on the left.



7.1 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.

The screenshot shows the Red Hat Cluster and Storage Systems web interface. At the top, there is a navigation bar with 'homebase', 'cluster', and 'storage' tabs. Below this is a sidebar with 'clusters' selected, containing 'Cluster List', 'Create a New Cluster', and 'Configure' options. The main content area is titled 'Create a new cluster' and contains a form with the following fields and options:

- Cluster Name:** haws
- Node Hostname:** et-virt10.lab.bos.redhat.com, et-virt11.lab.bos.redhat.com, and an empty field.
- Root Password:** Two fields with masked passwords (dots) and an empty field.
- Key ID:** Three fields with icons representing keys.
- Buttons:** 'Add a cluster node', 'View SSL cert fingerprints', and 'Submit'.
- Options:**
 - Download packages
 - Use locally installed packages.
 - Enable Shared Storage Support
 - Reboot nodes before joining cluster
 - Check if node passwords are identical.

Below the form, a 'Status messages' box displays the following information:

```
Status messages:  
■ Host et-virt10.lab.bos.redhat.com has SSL key fingerprint  
41:65:E3:5A:49:C8:1E:DE:7B:7E:26:18:11:3C:BE:A0:27:B3:A2:6B  
■ Host et-virt11.lab.bos.redhat.com has SSL key fingerprint  
43:78:0F:7D:8C:27:82:B1:9A:DD:5B:4C:5C:4B:D1:CD:C0:3F:5A:17
```

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

Note that it is possible to use the external hostnames of the servers to build a cluster (as seen above). This means that the cluster will be using the public LAN for its inter-node communications and heartbeats. It also means that the server running luci will need to be able to access the clustered systems on the same public LAN. A safer and more highly

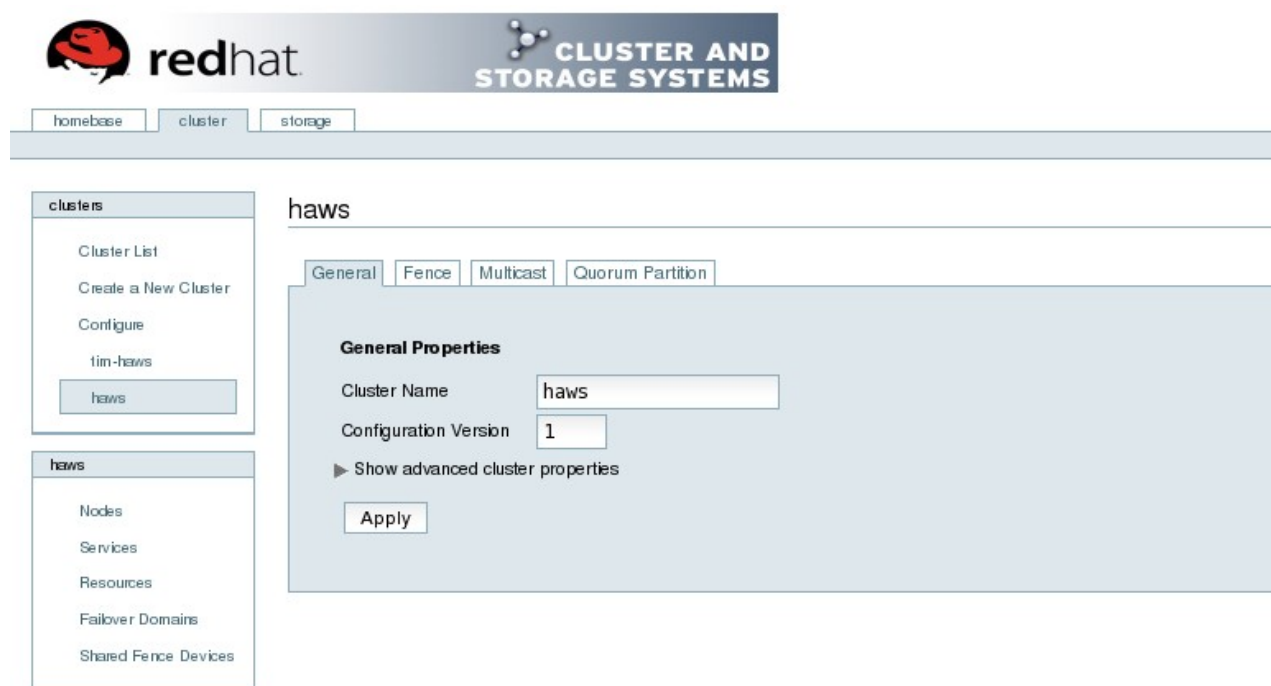


recommended configuration is to use the interconnect names (or their IP addresses) when building the cluster. This will require that the luci server also have a connection to the private LAN and will remove any possibilities of public IO traffic interfering with the cluster activities.

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.

This will then display the main configuration window for the newly created cluster. The *General* tab (shown below) 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.



clusters

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

tim-haws

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

haws

General Fence Multicast Quorum Partition

Fence Daemon Properties

Post Fail Delay

Post Join Delay

Run XVM fence daemon

XVM fence daemon key distribution

Enter a node hostname from the host cluster

Enter a node hostname from the hosted (virtual) cluster

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. While the default value is 3, a more practical setting is between 20 and 30 seconds, but can vary to user preference. For this effort, the default *Post-Join Delay* 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.



redhat.

CLUSTER AND STORAGE SYSTEMS

homebase

cluster

storage

clusters

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

tim-haws

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

tim-haws

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

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.



clusters

- Cluster List
- Create a New Cluster
- Configure
- tim-haws
- haws**

haws

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

haws

General Fence Multicast Quorum Partition

Quorum Partition Configuration

- Do not use a Quorum Partition
- Use a Quorum Partition

Interval

Votes

TKO

Minimum Score

Device

Label

Heuristics

Path to Program	Interval	Score	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
<input type="button" value="Add another heuristic"/>			

A quorum partition is not used in this two node example and thus, no information was entered or changed.

Reference the Global Cluster Properties section of [Configuring and Managing a Red Hat Cluster](#) for further quorum partition details if required.

7.2 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

haws

- Nodes
 - Add a Node
 - Configure
 - et-virt11.lab.bos.redhat.com
 - et-virt10.lab.bos.redhat.com**
- Services
- Resources
- Failover Domains
- Shared Fence Devices

haws

Node Name: **et-virt10.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"/>

Update node daemon properties

Services on this Node

- No cluster services are currently running here

Failover Domain Membership

- This node has no failover domain membership

Main Fencing Method	Backup Fencing Method
Add a fence device to this level	Add a fence device to this level
Update main fence properties	Update backup fence properties

7.3 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
Fence Type Dell Drac	
Name <input type="text" value="jet-virt10-drac.lab.b"/>	Add a fence device to this level
IP Address <input type="text" value="10.16.41.76"/>	
Login <input type="text" value="root"/>	
Password <input type="password" value="....."/>	
Password Script (optional) <input type="text"/>	
<input type="button" value="Remove this device"/>	
Add a fence device to this level	
<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 both nodes.

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.

7.4 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 will need to create a failover domain and include the members of the newly created cluster. From the cluster details window, click *Failover Domains* and then *Add a Failover Domain*.

The screenshot shows the Red Hat Cluster and Storage Systems web interface. The main header includes the Red Hat logo and the text 'CLUSTER AND STORAGE SYSTEMS'. Below the header are navigation tabs for 'homebase', 'cluster', and 'storage'. The left sidebar contains a 'clusters' menu with options like 'Cluster List', 'Create a New Cluster', and 'Configure', and a 'haws' menu with options like 'Nodes', 'Services', 'Resources', 'Failover Domains', 'Add a Failover Domain', 'Configure a Failover Domain', and 'Shared Fence Devices'. The main content area is titled 'haws' and 'Add a Failover Domain'. It contains a form with the following fields and options:

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

Node	Member	Priority
et-virt11.lab.bos.redhat.com	<input checked="" type="checkbox"/>	<input type="text" value="1"/>
et-virt10.lab.bos.redhat.com	<input checked="" type="checkbox"/>	<input type="text" value="1"/>
- Submit:** A button at the bottom of the form.

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.

7.5 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 three resource types will be defined to provide the high availability functionality of the web service.

- Script
- IP Address
- NFS Mount

7.5.1 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, 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 *Script*. 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.



clusters

- Cluster List
- Create a New Cluster
- Configure

haws

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

haws

Add a Resource

Script Resource Configuration

Name

Full path to script file

Submit

Click *Submit* to create the script resource.

7.5.2 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 the GUI interface (luci) or 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*.



clusters

- Cluster List
- Create a New Cluster
- Configure

haws

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

haws

Add a Resource

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 `/sbin/ip addr list` command. Note that interface eth3 (the public NIC) lists both the external system IP address as well as the IP resource address (10.16.40.165).

```
# /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 00:12:79:9e:98:22 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::212:79ff:fe9e:9822/64 scope link
        valid_lft forever preferred_lft forever
3: eth1: <NO-CARRIER,BROADCAST,MULTICAST,SLAVE,UP> mtu 1500 qdisc pfifo_fast
master bond0 qlen 1000
    link/ether 00:12:79:9e:98:22 brd ff:ff:ff:ff:ff:ff
4: eth2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop qlen 1000
    link/ether 00:13:72:4c:a5:a3 brd ff:ff:ff:ff:ff:ff
5: eth3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:13:72:4c:a5:a4 brd ff:ff:ff:ff:ff:ff
    inet 10.16.41.75/21 brd 10.16.47.255 scope global eth3
    inet 10.16.40.165/21 scope global secondary eth3
    inet6 fe80::213:72ff:fe4c:a5a4/64 scope link
        valid_lft forever preferred_lft forever
```



```
6: sit0: <NOARP> mtu 1480 qdisc noop
  link/sit 0.0.0.0 brd 0.0.0.0
7: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue
  link/ether 00:12:79:9e:98:22 brd ff:ff:ff:ff:ff:ff
  inet 10.10.1.3/24 brd 10.10.1.255 scope global bond0
  inet6 fe80::212:79ff:fe9e:9822/64 scope link
    valid_lft forever preferred_lft forever
```

7.5.3 NFS Mount

Because the web content to be served will be made available to the service via an NFS mount, that mount will have to be defined as a cluster resource so that it will be made available to whichever node is providing the service.

Note: *Appendix A* in this document provides instructions for maintaining local web content synchronized between nodes if an NFS server were not available.

It is assumed that the NFS export to be served is already established and the user knows the information necessary with which to access (mount) it. To view the publicly exported mounts of a given server, use the `showmount` command.

```
# showmount -e irish.lab.bos.redhat.com
Export list for irish.lab.bos.redhat.com:
/pub *
/www *
#
```

Test the target NFS export by mounting it manually to verify there are no issues.

```
# mount irish.lab.bos.redhat.com:/www /www
# ls /www
... <verify output> ...
# umount /www
#
```

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 *NFS Mount* to view the NFS Mount Resource Configuration window.



clusters

- Cluster List
- Create a New Cluster
- Configure

haws

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

haws

Add a Resource

NFS Mount Resource Configuration

Name	<input type="text" value="web-content"/>
Mount point	<input type="text" value="/www"/>
Host	<input type="text" value="irish.lab.bos.redhat"/>
Export path	<input type="text" value="/www"/>
NFS version	<input checked="" type="radio"/> NFS3 <input type="radio"/> NFS4
Options	<input type="text" value="ro"/>
Force unmount	<input checked="" type="checkbox"/>

To define this resource, enter:

- a logical name for the resource
- the NFS server name and export path
- the local mount point
- the NFS version on the exporting host
- NFS mount options



redhat

CLUSTER AND
STORAGE SYSTEMS

homebase

cluster

storage

clusters

- Cluster List
- Create a New Cluster
- Configure

haws

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

haws

Add a Resource

NFS Mount Resource Configuration

Name	<input type="text" value="web-content"/>
Mount point	<input type="text" value="/www"/>
Host	<input type="text" value="irish.lab.bos.redhat"/>
Export path	<input type="text" value="/www"/>
NFS version	<input checked="" type="radio"/> NFS3 <input type="radio"/> NFS4
Options	<input type="text" value="ro,soft"/>
Force unmount	<input checked="" type="checkbox"/>
<input type="button" value="Submit"/>	

In the above example, we are using `/www` as a local mount point, overriding the default Apache server location of `/var/www/html`. The read-only (`ro`) mount option was used since this service will not require write access to the web content. The soft mount option was used so no cluster member can hang if the NFS mount becomes unresponsive. These options are user chosen. Please reference the `mount(8)` manpage for details on NFS mount options.

Select the *Force unmount* check box and click *Submit* to create the NFS Mount resource.

Since the `httpd` service will be controlled by cluster services, the cluster is responsible for the starting and stopping of the `httpd` service. Given that, the `httpd` start script should not be started at system boot. Use the command below to ensure the system does not start the service at boot.

```
# chkconfig --del httpd
#
```

By the same token, the cluster will also need to determine which cluster member will mount the NFS export containing the web content. Normally an NFS mount entry would be placed in the `/etc/fstab` file to automatically mount the NFS export at boot but this is not so regarding cluster services. Do **not** add an entry mount to the `/etc/fstab` file for this NFS mount.



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

The screenshot shows the Red Hat Cluster and Storage Systems web interface. At the top, there is a navigation bar with the Red Hat logo and the text "CLUSTER AND STORAGE SYSTEMS". Below this, there are three tabs: "homebase", "cluster", and "storage". The "cluster" tab is selected. On the left side, there is a sidebar menu with two main sections: "clusters" and "haws". Under "clusters", there are links for "Cluster List", "Create a New Cluster", and "Configure". Under "haws", there are links for "Nodes", "Services", "Resources" (which is highlighted), "Add a Resource", "Configure a Resource", "Failover Domains", and "Shared Fence Devices". The main content area shows the "Resources for haws" page. It contains a table with the following data:

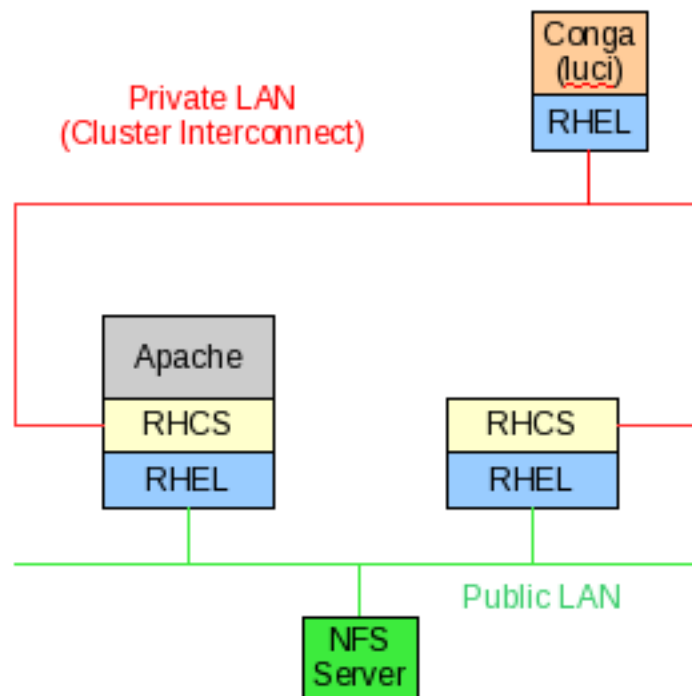
Resource Name	Type	Configure	Delete
10.16.40.165	IP Address	configure	delete
httpd	Script	configure	delete
web-content	NFS Mount	configure	delete

7.6 Web Service (*httpd*)

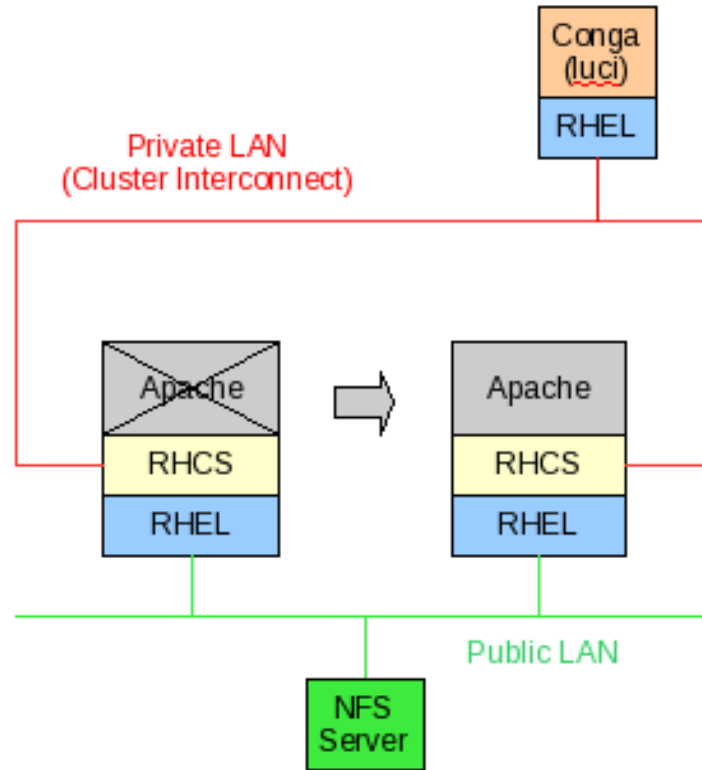
External web clients query a single IP address, in this case the IP Address resource previously created, which may be answered by any one of the cluster members at any given time.



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 for the NFS mount of the web content to be served.

```
# mkdir /www  
#
```

7.6.1 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*.



redhat



CLUSTER AND
STORAGE SYSTEMS

homebase

cluster

storage

clusters

- Cluster List
- Create a New Cluster
- Configure

haws

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

haws

Add a Service

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

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 *Recovery policy* from the pulldown menu.

Now the the three 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.



clusters

- Cluster List
- Create a New Cluster
- Configure

haws

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

haws

Add a Service

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

Add a new local resource

or

Use an existing global resource

From the *Use an existing global resource* pulldown menu, select the first of the three previously created cluster resources in the list.

Click on the *Add a resource to this domain* button again to add the next resource, and once again to add the third resource.

Now all three of the cluster resources should be listed under the newly defined service (httpd).



clusters

- Cluster List
- Create a New Cluster
- Configure

tim-haws

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

haws

Service Name httpd

Choose a Task...

Go

Service Status Running on
et-virt09.lab.bos.redhat.com

Service Composition

IP Address Resource Configuration

IP address 10.16.40.165

Monitor link

This resource is an independent subtree

Add a child

Delete this resource

Script Resource Configuration

Name httpd

Full path to script file /etc/rc.d/init.d/htt

This resource is an independent subtree

Add a child

Delete this resource

NFS Mount Resource Configuration

Name Web-Content

Mount point /www

Host irish.lab.bos.redhat

Export path /www

NFS version NFS3
 NFS4

Options hard,rw

Force unmount

This resource is an independent subtree

Add a child

Delete this resource

Automatically start this service

Run exclusive

Failover Domain HAWS

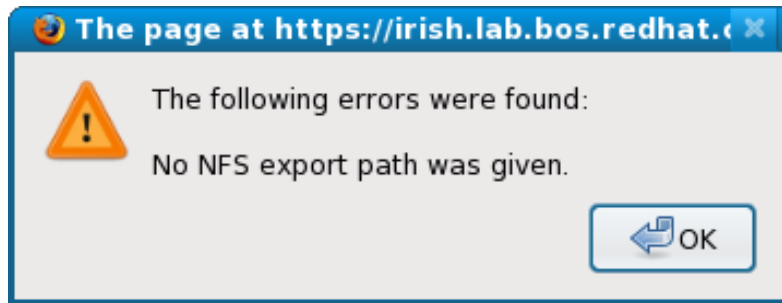
Recovery policy Relocate

Add a resource to this service

Submit



Click the *Submit* button to create the httpd service and associate the cluster resources with this service. If this action should produce the following error ...



... refer to *Appendix D* for the resolution and return to this point in the document when complete.

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.

7.6.2 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 10.16.40.165 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 10.16.40.165:80
```

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 */www* directory. Set the *DocumentRoot* directive accordingly.

```
DocumentRoot "/www"
```

Save the changes, exit the file and ensure that the same changes are on all nodes.



Use luci to restart the httpd service.

7.6.3 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 console (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, ...



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, 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.

8 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.

8.1 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 (hint: search for 'AVC').

An example of the specific information to monitor is:

```
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.

- The source process context was `root:staff_r:staff_t`
- The targeted file's context 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.

Because the specific `httpd_selinux` and `nfs_selinux` booleans relevant to the https service were set earlier, let SELinux identify which policies will hamper the web service activities by monitoring `/var/log/messages` for entries such as:

```
host=et-virt10.lab.bos.redhat.com type=AVC msg=audit(1216394729.151:3582): avc: denied { read
write } for pid=30459 comm="httpd" path="/socket:[418230]" dev=sockfs ino=418230
scontext=system_u:system_r:httpd_t:s0 tcontext=system_u:system_r:initrc_t:s0
tclass=unix_stream_socket
```



```
...
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
```

```
...
Jul 18 10:07:08 et-virt10 setroubleshoot: SELinux prevented the http daemon from reading files stored
on a NFS filesystem. For complete SELinux messages. run sealert -l
37e97a83-23d1-4d6e-9db5-74b0e481fb6e
```

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

```
# sealert -l 37e97a83-23d1-4d6e-9db5-74b0e481fb6e
```

Summary:

SELinux prevented the http daemon from reading files stored on a NFS filesystem.

Detailed Description:

[SELinux is in permissive mode, the operation would have been denied but was permitted due to permissive mode.]

SELinux prevented the http daemon from reading files stored on a NFS filesystem. NFS (Network Filesystem) is a network filesystem commonly used on Unix / Linux systems. The http daemon attempted to read one or more files or directories from a mounted filesystem of this type. As NFS filesystems do not support fine-grained SELinux labeling, all files and directories in the filesystem will have the same security context. If you have not configured the http daemon to read files from a NFS filesystem this access attempt could signal an intrusion attempt.

Allowing Access:

Changing the "httpd_use_nfs" boolean to true will allow this access: "setsebool -P httpd_use_nfs=1."

The following command will allow this access:

```
setsebool -P httpd_use_nfs=1
```

Additional Information:

Source Context	system_u:system_r:httpd_t
Target Context	system_u:object_r:nfs_t
Target Objects	/www [dir]
Source	httpd
Source Path	/usr/sbin/httpd
Port	<Unknown>
Host	et-virt11.lab.bos.redhat.com
Source RPM Packages	httpd-2.2.3-11.el5_1.3
Target RPM Packages	
Policy RPM	selinux-policy-2.4.6-137.1.el5_2
Selinux Enabled	True



```
Policy Type          targeted
MLS Enabled         True
Enforcing Mode      Permissive
Plugin Name         httpd_use_nfs
Host Name           et-virt10.lab.bos.redhat.com
Platform            Linux et-virt10.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         3
First Seen          Fri Jul 18 09:55:59 2008
Last Seen           Fri Jul 18 10:16:58 2008
Local ID            37e97a83-23d1-4d6e-9db5-74b0e481fb6e
Line Numbers
```

Raw Audit Messages

```
host=et-virt10.lab.bos.redhat.com type=AVC msg=audit(1216390618.128:6): avc:
denied { getattr } for pid=7934 comm="httpd" path="/www" dev=0:12 ino=1856
scontext=system_u:system_r:httpd_t:s0 tcontext=system_u:object_r:nfs_t:s0
tclass=dir

host=et-virt10.lab.bos.redhat.com type=SYSCALL msg=audit(1216390618.128:6):
arch=c000003e syscall=4 success=yes exit=0 a0=2ae18b4f0198 a1=7fff34ed04d0
a2=7fff34ed04d0 a3=0 items=0 ppid=7933 pid=7934 auid=4294967295 uid=0 gid=0
euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=(none) ses=4294967295
comm="httpd" exe="/usr/sbin/httpd" subj=system_u:system_r:httpd_t:s0 key=(null)
```

In the above example, “httpd” was the offending component.

During the course of testing for this effort (with SELinux in targeted permissive mode), various others were observed in `/var/log/messages` and `/var/log/audit/audit.log` including:

- ifconfig
- ping
- arping
- httpd
- initrc
- netutils

8.2 audit2allow

Now the user should 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 NFS mount is successful 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 (`haws`) or service (`httpd`) links.

On the node providing the `httpd` service, the NFS mount should be present.

```
# df -h | grep www
irish.lab.bos.redhat.com: /www    64G   6.7G   54G   11% /www
#
```

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-virt11.lab.bos.redhat.com
Trying to relocate service:httpd to et-virt11.lab.bos.redhat.com...Success
service:httpd is now running on et-virt11.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



9 Diagnostics

9.1 *clustat*

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

```
# clustat
Cluster Status for haws @ Tue Aug 26 15:20:53 2008
Member Status: Quorate

Member Name                                ID    Status
-----
et-virt10.lab.bos.redhat.com                1    Online, Local, rgmanager
et-virt11.lab.bos.redhat.com                2    Online, rgmanager

Service Name                                Owner (Last)                                State
-----
service:httpd                               et-virt10.lab.bos.redhat.com               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-virt10` so the additional status of *Local* is included for that node in the output.

9.2 *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`.

10 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 necessary resources and `httpd` service, adjust the firewall and SELinux to provide system security, and successfully serve NFS based web content.

Future versions of this document will include hosting web pages from shared storage within the cluster, using the global file system (GFS), `qdisk` heuristics for 2-node clusters, command



line cluster control, as well as cluster upgrade and maintenance procedures.

Appendices

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 NFS server available, rather than acting as a mount point, the `/www` directory could contain the web content for serving. What becomes important is that both nodes have the same data since their respective `/www` 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 `/www` 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 `/www` directories in sync on both nodes. For example, from node `et-virt10` ...

```
# rsync -avz -e ssh root@et-virt11.lab.bos.redhat.com:/www/ /www/  
#
```

... will synchronize the `/www` directory contents on `et-virt10` with those on `et-virt11`. This can be configured as a cron job if preferred.

Appendix B: Configuration Files

cluster.conf

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:

- ensure that the `config_version` value in line 2 of the file has been incremented
- 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 3 to 4
```




Update complete.

```
# cman_tool version -r 4
#
```

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: 4
#
```

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

```
<?xml version="1.0"?>
<cluster alias="haws" config_version="38" name="haws">
  <fence_daemon clean_start="0" post_fail_delay="0" post_join_delay="30"/>
  <clusternodes>
    <clusternode name="et-virt11.lab.bos.redhat.com" nodeid="1" votes="1">
      <fence>
        <method name="1">
          <device name="et-virt11-drac.lab.bos.redhat.com"/>
        </method>
      </fence>
    </clusternode>
    <clusternode name="et-virt10.lab.bos.redhat.com" nodeid="2" votes="1">
      <fence>
        <method name="1">
          <device name="et-virt10-drac.lab.bos.redhat.com"/>
        </method>
      </fence>
    </clusternode>
  </clusternodes>
  <cman expected_votes="1" two_node="1"/>
  <fencedevices>
    <fencedevice agent="fence_drac" ipaddr="10.16.41.72" login="root" name="et-virt10-drac.lab.bos.redhat.com" passwd="calvin"/>
    <fencedevice agent="fence_drac" ipaddr="10.16.41.74" login="root" name="et-virt11-drac.lab.bos.redhat.com" passwd="calvin"/>
  </fencedevices>
  <rm>
    <failoverdomains>
      <failoverdomain name="HAWS" nofailback="0" ordered="1" restricted="0">
        <failoverdomainnode name="et-virt10.lab.bos.redhat.com" priority="1"/>
        <failoverdomainnode name="et-virt11.lab.bos.redhat.com" priority="1"/>
      </failoverdomain>
    </failoverdomains>
  </rm>
  <resources>
```



```
<script file="/etc/rc.d/init.d/httpd" name="httpd"/>
<ip address="10.16.40.164" monitor_link="1"/>
<netfs export="/haws" force_unmount="1" host="irish.lab.bos.redhat.com"
mountpoint="/www" name="Web-Content" nfstype="nfs" options="hard,rw"/>
</resources>
<service autostart="1" domain="HAWS" exclusive="0" name="httpd" recovery="relocate">
  <ip ref="10.16.40.164"/>
  <script ref="httpd"/>
  <netfs ref="Web-Content"/>
</service>
</rm>
</cluster>
```

iptables

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

```
# Generated by iptables-save v1.3.5 on Tue Sep 2 19:30:20 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 8084 -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 21064 -j ACCEPT
-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 11111 -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 -p udp -m state --state NEW -m multiport --dports 5404,5405 -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 Tue Sep 2 19:30:20 2008

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-virt11.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=10.10.1.4
USERCTL=no
TYPE=Ethernet
IPV6INIT=no
PEERDNS=yes
BONDING_OPTS="mode=1 miimon=100 primary=eth0"
```

Appendix C: RHN

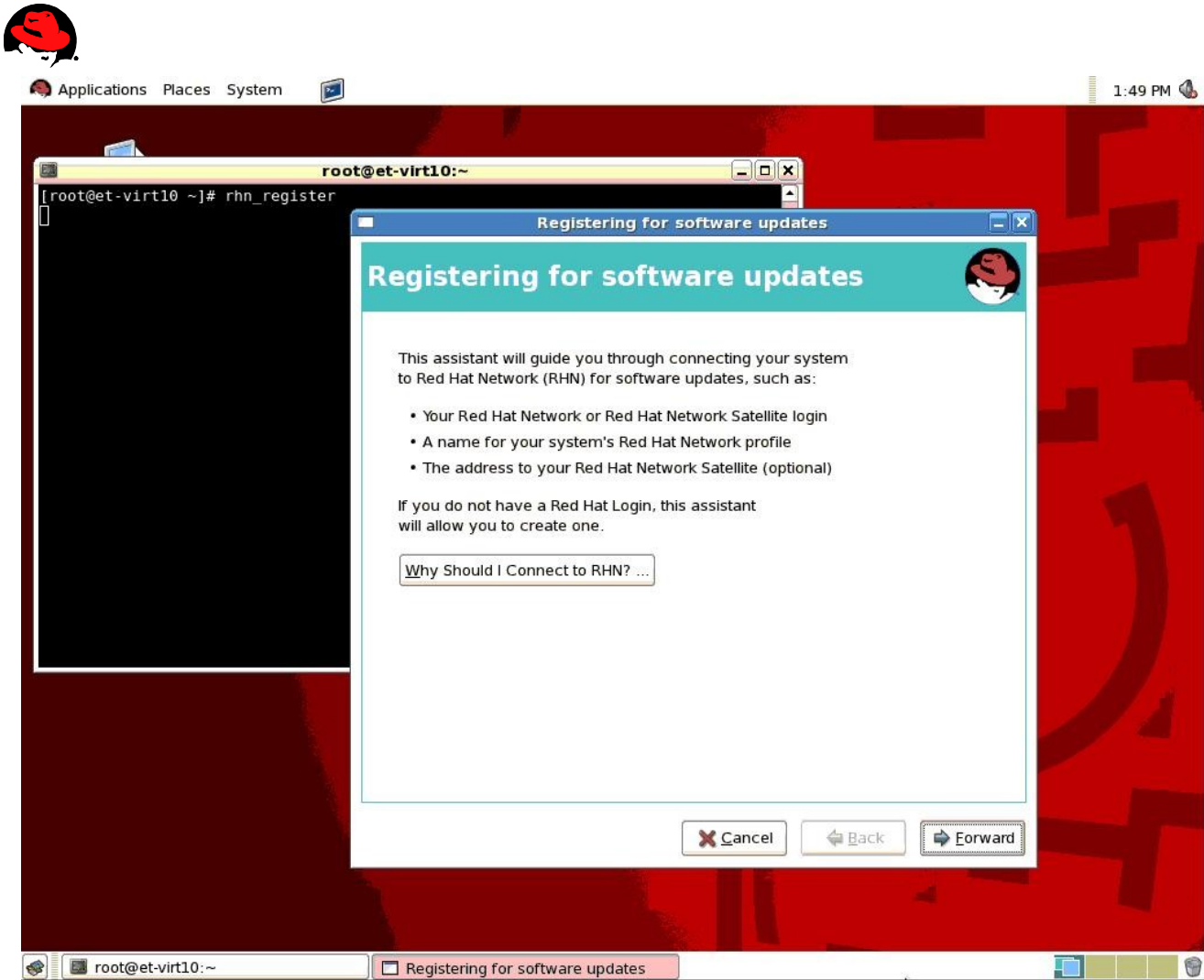
If the server was registered for RHN updates immediately after the OS installation, there is no need to perform the procedures in this section.

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.

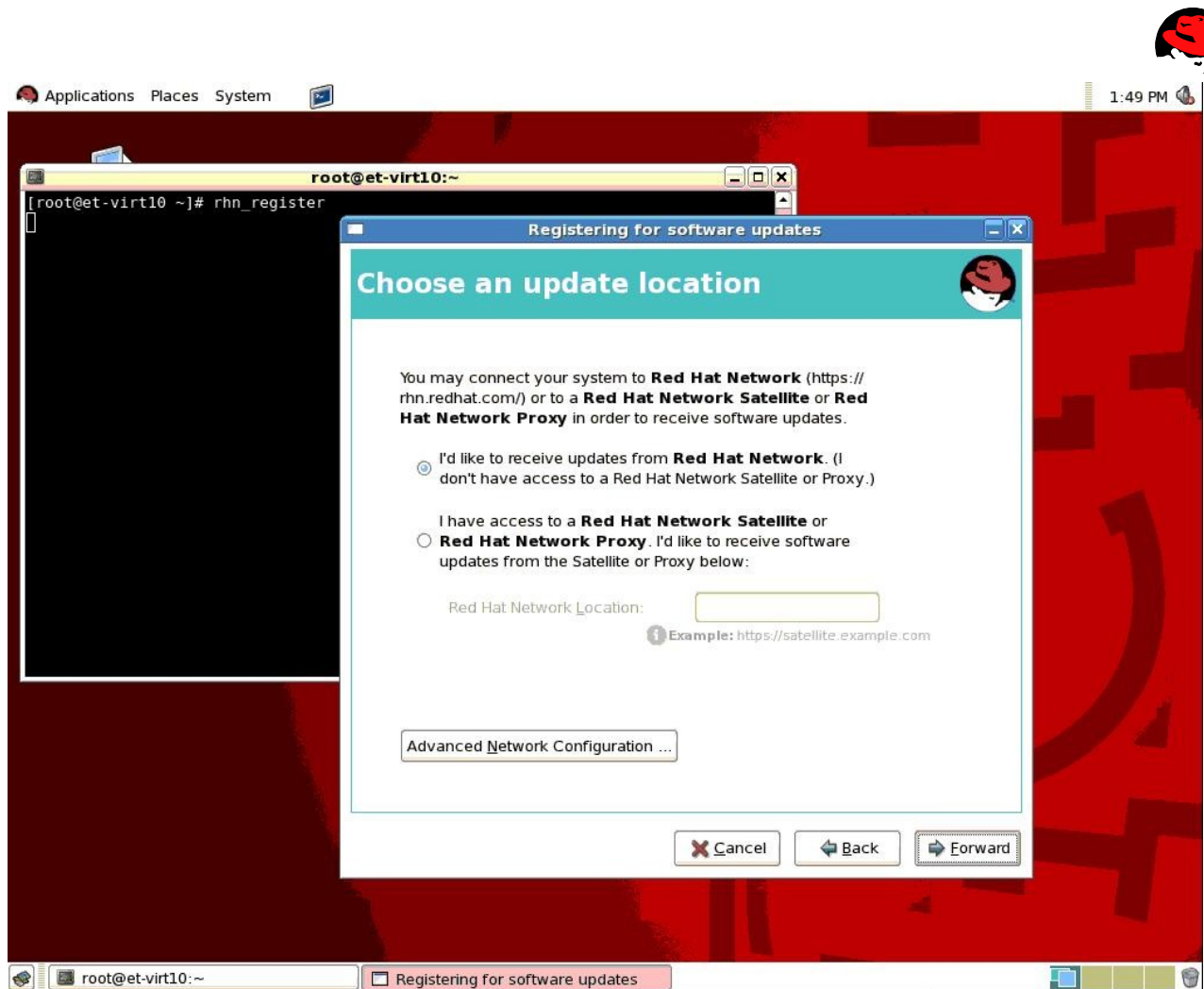
```
# rhn_register
```

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



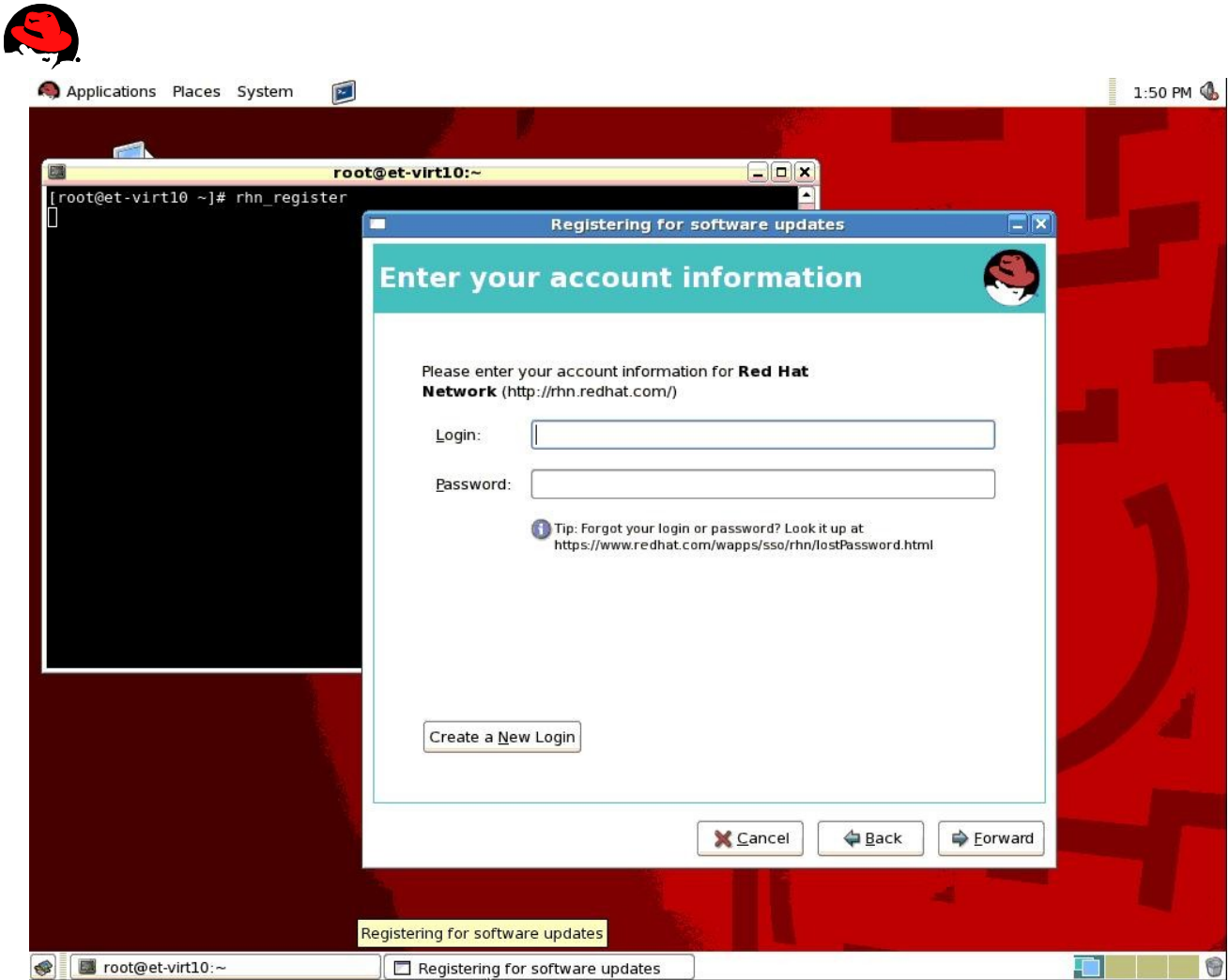
See the *Software Updates (RHN Configuration)* section in this document for information regarding satellite servers.

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.

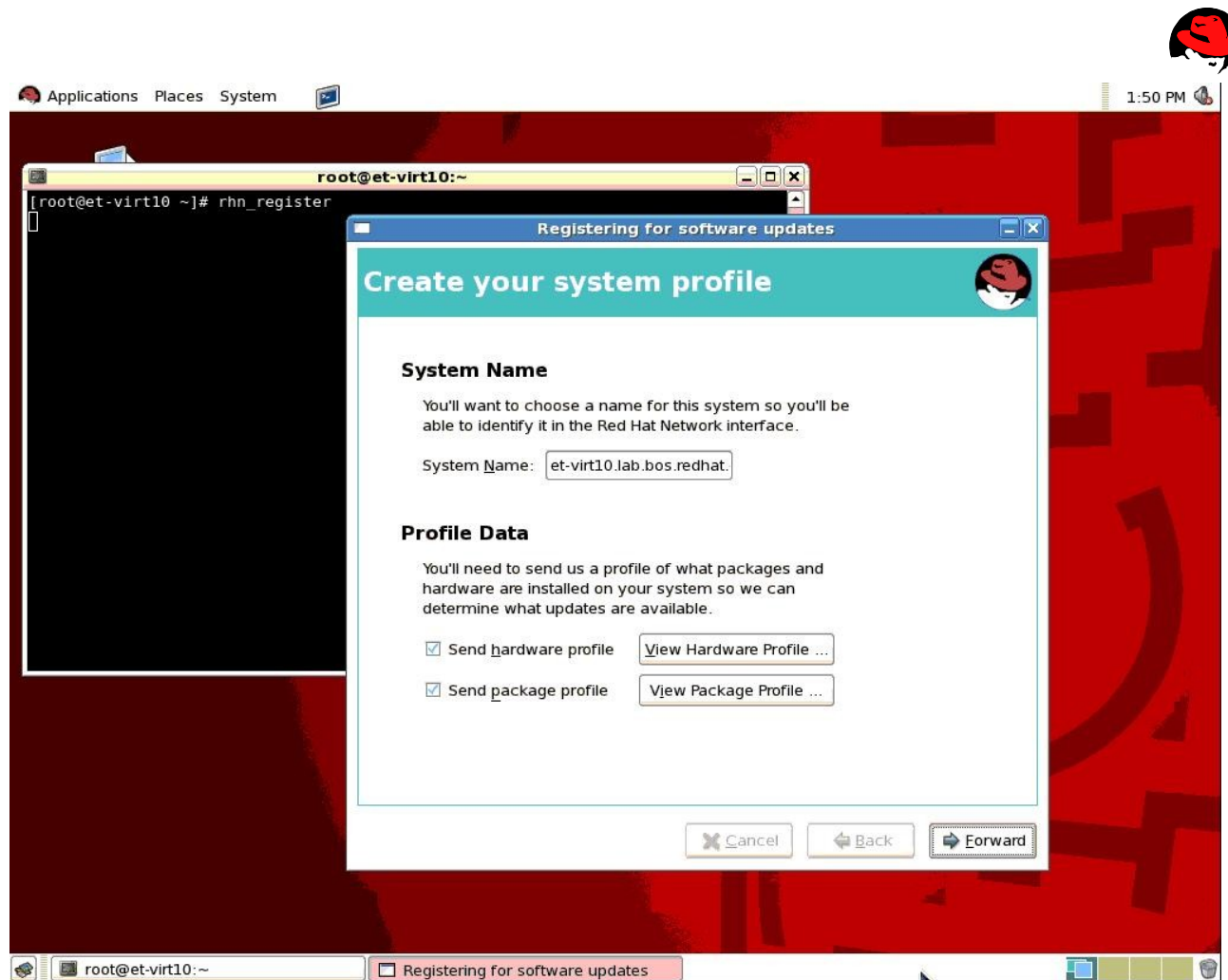


Enter any necessary proxy information if applicable.

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



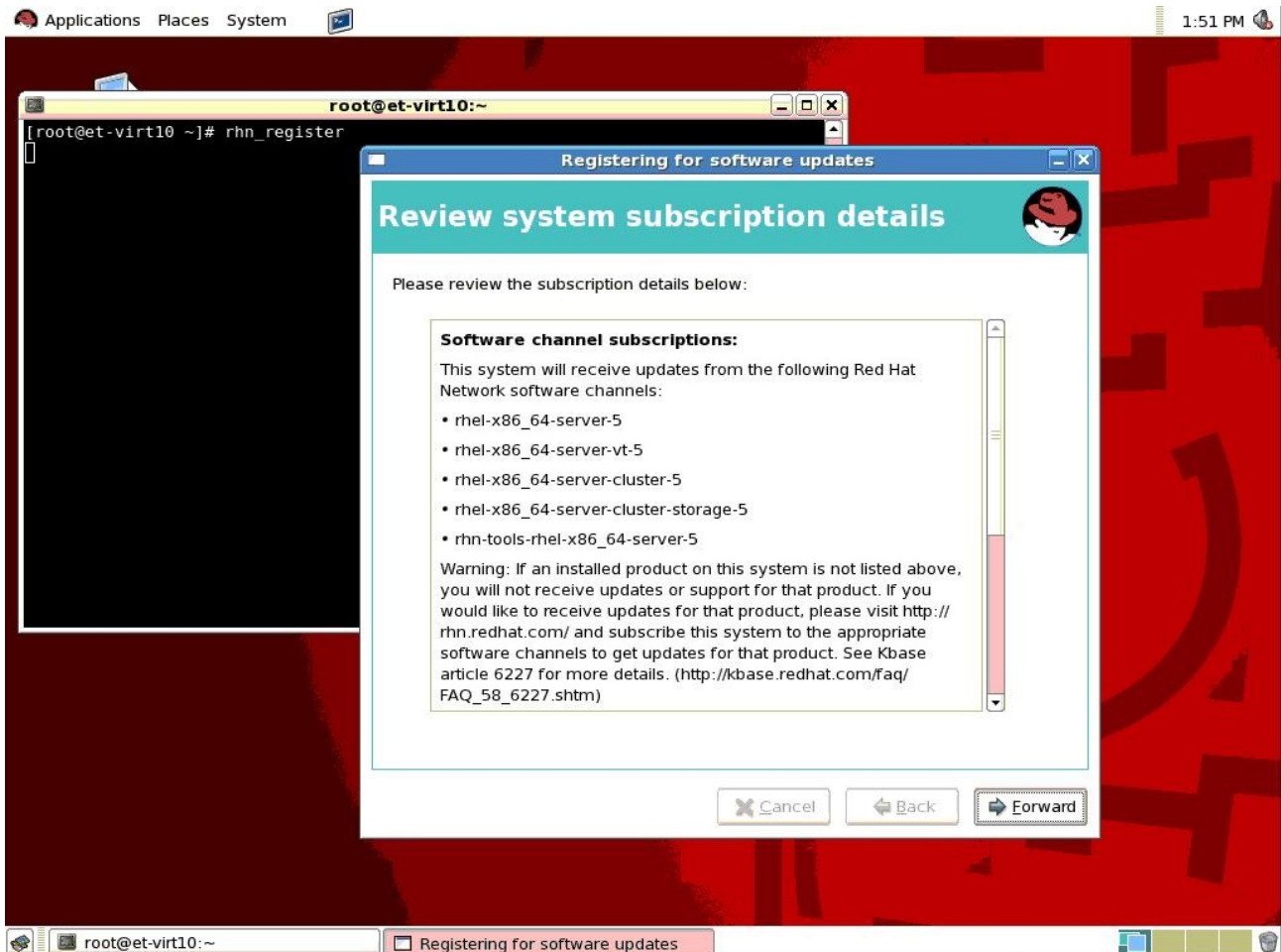
This will direct the user to the Profile Creation window where the server name should already be present in the System Name field.



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

Your RHN

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 menu bar 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 navigation bar with 'Systems' selected. Below the navigation bar, there is a search bar and a 'NO SYSTEMS SELECTED' message. The main content area displays a list of systems managed by the RHN account. The list is filtered by system name, showing 5 systems. Each system row includes a checkbox for selection, a status indicator (OK), and columns for Updates, Errata, Packages, System, Base Channel, and Entitlement. The systems listed 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. The page also includes a 'Download Software' button and a 'Red Hat Customer Center' link.

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



et-virt10.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-virt10.lab.bos.redhat.com
IP Address:	10.16.41.75
Kernel:	2.6.18-92.el5xen
RHN System ID:	1013606085
Lock Status:	System is unlocked (Lock system)

System Events

Checked In:	8/20/08 1:12:47 PM EDT
Registered:	8/20/08 11:11:26 AM EDT
Last Booted:	8/19/08 12:11:00 PM EDT (Schedule System Reboot)

System Properties (Edit These Properties)

Entitlements:	[Management]
Notifications:	Daily Summary Errata Email
Auto Errata Update:	No
System Name:	et-virt10.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

DOWNLOAD SOFTWARE



Red Hat Customer Center

For Subscription Management & Customer Support

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 (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.



et-virt10.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-virt10.lab.bos.redhat.com
IP Address:	10.16.41.75
Kernel:	unknown
RHN System ID:	1013606955
Lock Status:	System is unlocked (Lock system)

System Events

Checked In:	8/20/08 2:29:02 PM EDT
Registered:	8/20/08 2:28:57 PM EDT
Last Booted:	12/31/69 7:00:00 PM EST (Schedule System Reboot)

System Properties (Edit These Properties)

Entitlements:	[Management]
Notifications:	Daily Summary Errata Email
Auto Errata Update:	No
System Name:	et-virt10.lab.bos.redhat.com
Description:	Initial Registration Parameters: OS: redhat-release Release: 5Server CPU Arch: x86_64-redhat-linux
Location:	Room Rack Building

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)
 - 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-virt10.lab.bos.redhat.com et-virt10 localhost.localdomain localhost
```

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

- ◆ **Luci reports incorrect service status with SELinux enforcing**

- ◆ Red Hat Bugzilla #461769

This bug prevents luci from reading the accurate status (running or stopped) of a cluster service when SELinux is in Enforcing mode.

Use `c_lustat` to see the true status and location of cluster services. Fixed in `selinux-policy-2.4.6-154 RPM`

- ◆ **“No NFS export path was given”**

Red Hat Bugzilla #444381

This is caused by a bug in luci’s writing to the `cluster.conf` file. A specific attribute (for NFS Mount resources) named ‘export’ is incorrectly written to the resource definition as ‘exportpath’. When the file contains this incorrect attribute name, luci can not locate the export path for the NFS resource.

To resolve this issue, remove the NFS Mount resource and create the `httpd` service using the `Script` and `IP Address` resources.

Now see the the `cluster.conf` file example below.

```
<?xml version="1.0"?>
<cluster alias="haws" config_version="10" name="haws">
  <fence_daemon clean_start="0" post_fail_delay="0" post_join_delay="30"/>
  <clusternodes>
    <clusternode name="et-virt11.lab.bos.redhat.com" nodeid="1" votes="1">
      <fence>
        <method name="1">
          <device name="et-virt11-drac.lab.bos.redhat.com"/>
        </method>
      </fence>
    </clusternode>
    <clusternode name="et-virt10.lab.bos.redhat.com" nodeid="2" votes="1">
      <fence>
```




```
        <method name="1">
            <device name="et-virt10-drac.lab.bos.redhat.com"/>
        </method>
    </fence>
</clusternode>
</clusternodes>
<cman expected_votes="1" two_node="1"/>
<fencedevices>
    <fencedevice agent="fence_drac" ipaddr="10.16.41.76" login="root" name="et-virt10-
drac.lab.bos.redhat.com" passwd="calvin"/>
    <fencedevice agent="fence_drac" ipaddr="10.16.41.78" login="root" name="et-virt11-
drac.lab.bos.redhat.com" passwd="calvin"/>
</fencedevices>
<rm>
<failoverdomains>
    <failoverdomain name="haws_fo_domain" nofailback="0" ordered="0" restricted="0">
        <failoverdomainnode name="et-virt11.lab.bos.redhat.com" priority="1"/>
        <failoverdomainnode name="et-virt10.lab.bos.redhat.com" priority="1"/>
    </failoverdomain>
</failoverdomains>
<resources>
    <ip address="10.16.40.165" monitor_link="1"/>
    <script file="/etc/rc.d/init.d/httpd" name="httpd"/>
    <netfs exportpath="/www" force_unmount="1" host="irish.lab.bos.redhat.com"
mountpoint="/www" name="web-content" nfstype="nfs" options="ro,soft"/>
</resources>
    <service autostart="1" domain="haws_fo_domain" exclusive="0" name="httpd"
recovery="relocate">
        <ip ref="10.16.40.165"/>
        <script ref="httpd"/>
    </service>
</rm>
</cluster>
```

In the last sections of the file, note the section where the cluster resources are defined between the `<resources>` and `</resources>` tags.

Because the service was created using the Script and IP Address resources, the cluster service (and more importantly, the cluster resources that the service uses) was defined between the `<service ...>` and `</service>` tags. The three cluster resources are defined, ...

```
    <ip address="10.16.40.165" monitor_link="1"/>
    <script file="/etc/rc.d/init.d/httpd" name="httpd"/>
    <netfs exportpath="/www" force_unmount="1"
host="irish.lab.bos.redhat.com" mountpoint="/www" name="web-content" nfstype="nfs"
options="ro,soft"/>
```



... two of which are already associated with the httpd service.

```
<service autostart="1" domain="haws_fo_domain" exclusive="0" name="httpd"
recovery="relocate">
  <ip ref="10.16.40.165"/>
  <script ref="httpd"/>
</service>
```

There are three changes to make in this file.

1. The *exportpath* attribute in the NFS Mount definition (starting with *<netfs ...>*) needs to be changed to *export*. It should look like below.

```
<netfs export="/www" force_unmount="1" host="irish.lab.bos.redhat.com"
mountpoint="/www" name="web-content" nfstype="nfs" options="ro,soft"/>
```

2. Manually insert one line into the */etc/cluster/cluster.conf* file adding the missing cluster resource (NFS Mount) to the httpd service. The httpd service definition stanza above now looks like this.

```
<service autostart="1" domain="haws_fo_domain" exclusive="0" name="httpd"
recovery="relocate">
  <ip ref="10.16.40.165"/>
  <script ref="httpd"/>
  <netfs ref="web-content"/>
</service>
```

3. Increment the *config_version* value in line 2. In the example above, the value would be changed from 10 to 11.

Save the changes and exit the editor. On the node where the file was modified, propagate the new *cluster.conf* file to the other node(s) using the *css_tool* and *cman_tool* commands.

```
# ccs_tool update /etc/cluster/cluster.conf
Config file updated from version 10 to 11

Update complete.

# cman_tool version -r 11
#
```

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: 11
#
```



Any changes to the `cluster.conf` file will require the restart of the cluster to implement the changes.

Return to `luci` and click on the blue `cluster` tab at the top. Restart the cluster using the pulldown menu to the right of the cluster name (`haws`). Once restarted

Click on the `httpd` service name to view the service details and verify that the NFS Mount Resource Configuration details are listed after the IP Address and Script resources. Due to the bug itself, the Export path field will be empty but the real contents of that field are in the `cluster.conf` file. As long as the bug exists, any use of `luci` to write the service data will overwrite the changes made manually.

Return to the *Service Creation* section in this document.

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 <code>/etc/hosts</code> files (local LANs & <code>luci</code> server)
	Install/Enable <code>ricci</code>
	Install/Enable <code>luci</code>
	Create Cluster



	Configure Failover Domain
	Configure Fence Devices
	Configure httpd Script Resource
	Configure IP Address Resource
	Configure NFS Mount Resource
	Create httpd Service
	Edit Web Server Directives
	Test HTTP Functionality to Generate SELinux Logging
	Add SELinux Policy Rules (audit2allow)
	SELinux In Enforcing Mode