Installation Guide

Scyld ClusterWare Release 7.4.1-741g0000

October 31, 2017

Installation Guide: Scyld ClusterWare Release 7.4.1-741g0000; October 31, 2017

Revised Edition
Published October 31, 2017

Table of Contents

Preface

Congratulations on purchasing *Scyld ClusterWare*, the most scalable and configurable Linux Cluster Software on the market. This guide describes how to install Scyld ClusterWare using Penguin's installation repository. You should read this document in its entirety, and should perform any necessary backups of the system before installing this software. You should pay particular attention to keeping a copy of any local configuration files.

The Scyld ClusterWare documentation set consists of:

- The *Installation Guide* containing detailed information for installing and configuring your cluster.
- The *Release Notes* containing release-specific details, potentially including information about installing or updating the latest version of Scyld ClusterWare.
- The *Administrator's Guide* and *User's Guide* describing how to configure, use, maintain, and update the cluster.
- The *Programmer's Guide* and *Reference Guide* describing the commands, architecture, and programming interface for the system.

These product guides are available in two formats, HTML and PDF. You can browse the documentation on the Penguin Computing Support Portal at http://www.penguincomputing.com/support/documentation.

Once you have completed the Scyld ClusterWare installation, you can view the PDF documentation in `/usr/share/doc/PDF/scyld-doc/`, or launch your Mozilla browser and go to the default page, file:///usr/share/doc/HTML/index.html. In the *Featured Links* section, click on the *ClusterWare Documentation* link.

Note: If your reseller pre-installed *Scyld ClusterWare* on your cluster, you may skip these installation instructions and visit the *User's Guide* and *Reference Guide* for helpful insights about how to use Scyld ClusterWare.

Feedback

We welcome any reports on errors or difficulties that you may find. We also would like your suggestions on improving this document. Please direct all comments and problems to support@penguincomputing.com.

When writing your email, please be as specific as possible, especially with errors in the text. Please include the chapter and section information. Also, please mention in which version of the manual you found the error. This version is *Scyld ClusterWare, Revised Edition*, published October 31, 2017.

Notes

1. http://www.penguincomputing.com/support/documentation
2. file:///usr/share/doc/HTML/index.html

Chapter 1. Scyld ClusterWare System Overview

System Components and Layout

Scyld ClusterWare streamlines the processes of configuring, running, and maintaining a Linux cluster using a group of commodity off-the-shelf (COTS) computers connected through a private network.

The front-end "master node" in the cluster is configured with a full Linux installation, distributing computing tasks to the other "compute nodes" in the cluster. Nodes communicate across a private network and share a common process execution space with common, cluster-wide process ID values.

A compute node is commonly diskless, as its kernel image is downloaded from the master node at node startup time using the Preboot eXecution Environment (*PXE*), and libraries and executable binaries are transparently transferred from the master node as needed. A compute node may access data files on locally attached storage or across NFS from an NFS server managed by the master node or some other accessible server.

In order for the master node to communicate with an outside network, it needs two network interface controllers (*NIC*s): one for the private internal cluster network, and the other for the outside network. It is suggested that the master node be connected to an outside network so multiple users can access the cluster from remote locations.

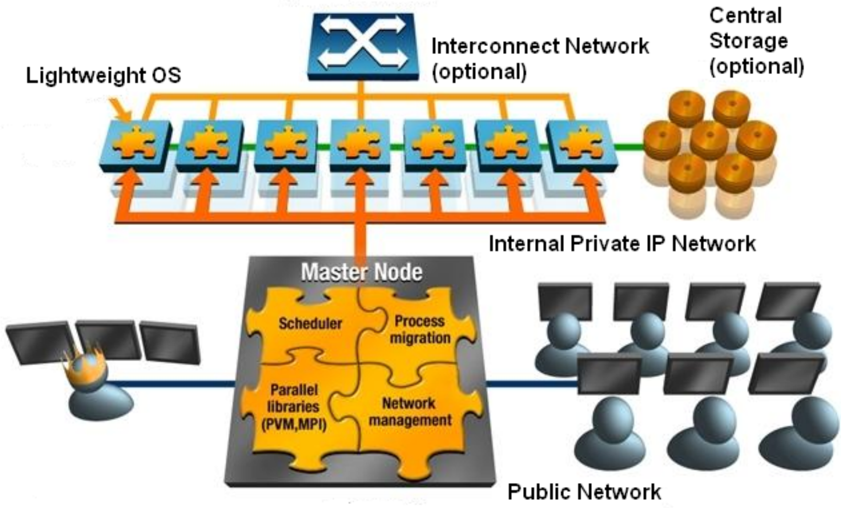


Figure 1-1. Cluster Configuration

Recommended Components

Hardware selection for a ClusterWare system is based on the price/performance ratio. Scyld recommends the components listed below:

Processors

64-bit Intel® or AMD ™ x86_64 architecture **required**, single-core or multi-core

Architecture

1 or multiple sockets per motherboard

Physical Memory

4096 MBytes (4 GBytes) or more preferred, minimum 2048 MBytes (2 GBytes)

Operating System

Red Hat Enterprise Linux 7 (RHEL7) or CentOS7 **required**

The Scyld ClusterWare *Release Notes* state the specific version and update of Red Hat or CentOS required to support the ClusterWare release you are installing.

Network Interface Controllers (NIC)

Gigabit Ethernet (Fast Ethernet at a minimum) PCI-X or PCI-Express adapters (with existing Linux driver support) in each node for the internal private IP network.

The master node typically employs an additional NIC for connecting the cluster to the external network. This NIC should be selected based on the network infrastructure (e.g., Fast Ethernet if the external network you are connecting the cluster to is Fast Ethernet).

Network Switch

The master node private network NIC and all compute nodes should be connected to a non-blocking Gigabit Ethernet switch for the internal private network. At a minimum, the network switch should match the speed of the network cards.

The switch is a critical component for correct operation and performance of the cluster. In particular, the switch must be able to handle all network traffic over the private interconnect, including cluster management traffic, process migration, library transfer, and storage traffic. It must also properly handle DHCP and PXE.

> **Tip:** It is sometimes confusing to identify which NIC is connected to the private network. Take care to connect the master node to the private switch through the NIC with the same or higher speed than the NICs in the compute nodes.

Disk Drives

For the master node, we recommend using either Serial ATA (SATA) or SCSI disks in a RAID 1 (mirrored) configuration. The operating system on the master node requires approximately 3 GB of disk space. We recommend configuring the compute nodes without local disks (disk-less).

If local disks are required on the compute nodes, we recommend using them for storing data that can be easily re-created, such as scratch storage or local copies of globally-available data.

In the default configuration, `/home` on the master node is exported to the compute nodes; other file systems may be exported as well. After installing Scyld ClusterWare, see the file `/etc/beowulf/fstab` for the full list of default mounts for compute nodes. If you expect heavy file system traffic, we recommend that you provide a second pair of disks in a RAID 1 (mirrored) configuration for these exported file systems. Otherwise, it is possible for accesses to the exported file systems to interfere with the master node accessing its system files, thus affecting the master node's ability to launch new processes and manage the cluster.

Optional Hardware Components

Gigabit Ethernet with a non-blocking switch serves most users. However, some applications benefit from a lower-latency interconnect.

Infiniband is an industry standard interconnect providing low-latency messaging, IP, and storage support. Infiniband can be configured as a single universal fabric serving all of the cluster's interconnect needs.

More information about Infiniband may be found at the Infiniband Trade Association web site at http://www.infinibandta.org. Scyld supports Infiniband as a supplemental messaging interconnect in addition to Ethernet for cluster control communications.

Assembling the Cluster

The full Scyld ClusterWare Cluster Virtualization Software and the underlying Linux operating system are installed only on the master node.

Most recent hardware supports network boot (PXE boot), which Scyld requires for booting the compute nodes.

Software Components

The following are integral components of *Scyld ClusterWare*:

- *beostatus*: A graphic utility for monitoring the status of a Scyld cluster.
- *Scyld ClusterWare*: Allows processes to be started on compute nodes in the cluster and tracked in the process table on the master node. *Scyld ClusterWare* also provides process migration mechanisms to help in creating remote processes, and removes the need for most binaries on the remote nodes.
- *MPICH2*, *MVAPICH2*, and *OpenMPI*: Message Passing Interfaces, customized to work with Scyld ClusterWare.

For more detailed information on these software components, see the *Administrator's Guide* and the *User's Guide*.

Notes

1. http://www.infinibandta.org

Chapter 2. Quick Start Installation

Introduction

Scyld ClusterWare is supported on Red Hat Enterprise Linux 7 (RHEL7) and CentOS7. This document describes installing on Red Hat, though installing on CentOS will be identical, except where explicitly noted. Scyld ClusterWare is installed on the master node after installing a RHEL7 or CentOS7 base distribution. You must configure your network interface and network security settings to support Scyld ClusterWare.

The compute nodes join the cluster without any explicit installation. Having obtained a boot image via PXE, the nodes are converted to a Scyld-developed network boot system and seamlessly appear as part of a virtual parallel computer.

This chapter introduces you to the Scyld ClusterWare installation procedures, highlights the important steps in the Red Hat installation that require special attention, and then steps you through the installation process. Installation is done using the **/usr/bin/yum** command, installing from a repository of rpms, typically across a network connection. See Chapter 3 for more detailed instructions. Refer to the Red Hat documentation for information on installing RHEL7.

Network Interface Configuration

> **Tip:** To begin, you must know which interface is connected to the public network and which is connected to the private network. Typically, the public interface is eth0 and the private interface is eth1.

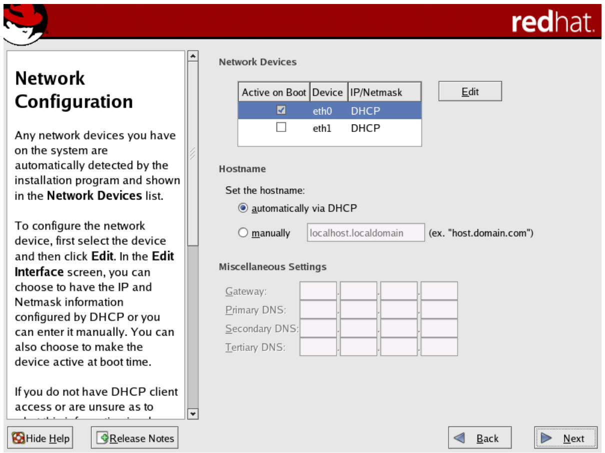
It is important to properly configure the network interfaces to support Scyld ClusterWare. The Network Configuration screen is presented during the RHEL7 installation; it can be accessed post-installation via the **Applications -> System Settings -> Network** menu options.

Cluster Public Network Interface

For the public network interface (typically eth0), the following settings are typical, but can vary depending on your local needs:

- DHCP is the default, and is recommended for the public interface.
- If your external network is set up to use static IP addresses, then you must configure the public network interface manually. Select and edit this interface, setting the IP address and netmask as provided by your Network Administrator.
- If you use a static IP address, the subnet must be different from that chosen for the private interface. You must set the hostname manually and also provide gateway and primary DNS IP addresses.

> **Tip:** When configuring the network security settings (see the Section called *Network Security Settings*), Scyld recommends setting a firewall for the public interface.



Network Configuration

Any network devices you have on the system are automatically detected by the installation program and shown in the **Network Devices** list.

To configure the network device, first select the device and then click **Edit**. In the **Edit Interface** screen, you can choose to have the IP and Netmask information configured by DHCP or you can enter it manually. You can also choose to make the device active at boot time.

If you do not have DHCP client access or are unsure as to

Network Devices

Active on Boot	Device	IP/Netmask
☑	eth0	DHCP
□	eth1	DHCP

Hostname

Set the hostname:

● automatically via DHCP

○ manually localhost.localdomain (ex. "host.domain.com")

Miscellaneous Settings

Figure 2-1. Public Network Interface Configuration

Cluster Private Network Interface

Caution

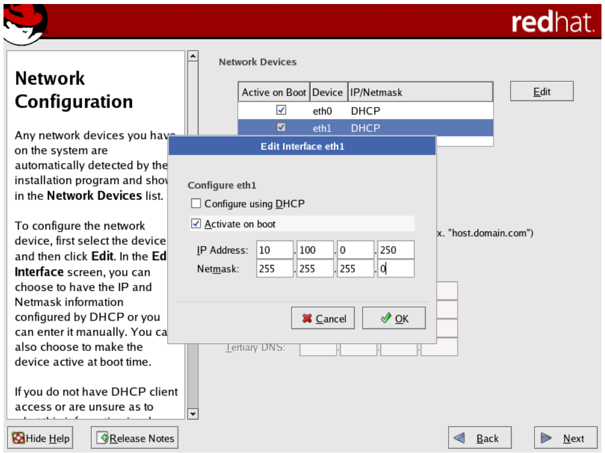
For the private network interface (typically eth1), DHCP is shown as default, but this option cannot be used. You must configure the network interface manually and assign a static IP address and netmask.

Caution

The cluster will not run correctly unless the private network interface is trusted. You can set this interface as a "trusted device" when configuring the network security settings post-installation; see the Section called *Trusted Devices*.

For the cluster private interface (typically eth1), the following settings are required for correct operation of Scyld ClusterWare:

- Do not configure this interface using DHCP. You must select this interface in the Network Configuration screen and edit it manually in the Edit Interface dialog (see Figure 2-2).
- Set this interface to "activate on boot" to initialize the specific network device at boot-time.
- Specify a static IP address. We recommend using a non-routable address (such as 192.168.x.x, 172.16.x.x to 172.30.x.x, or 10.x.x.x).
- If the public subnet is non-routable, then use a different non-routable range for the private subnet (e.g., if the public subnet is 192.168.x.x, then use 172.16.x.x to 172.30.x.x or 10.x.x.x for the private subnet).
- Once you have specified the IP address, set the subnet mask based on this address. The subnet mask must accommodate a range large enough to contain all of your compute nodes.



Network Configuration

Any network devices you hav
on the system are
automatically detected by the
installation program and show
in the **Network Devices** list.

To configure the network
device, first select the device
and then click **Edit**. In the **Ed**
Interface screen, you can
choose to have the IP and
Netmask information
configured by DHCP or you
can enter it manually. You ca
also choose to make the
device active at boot time.

If you do not have DHCP client
access or are unsure as to

Network Devices

Active on Boot	Device	IP/Netmask
☑	eth0	DHCP
☑	eth1	DHCP

Edit Interface eth1

Configure eth1

☐ Configure using DHCP

☑ Activate on boot

IP Address: 10 . 100 . 0 . 250

Netmask: 255 . 255 . 255 . 0

x. "host.domain.com")

Tertiary DNS:

Figure 2-2. Private Network Interface Configuration

Tip: You must first select the private interface in the Network Configuration screen, then click **Edit** to open the Edit Interface dialog box.

Tip: Although you can edit the private interface manually during the Red Hat installation, making this interface a "trusted device" must be done post-installation.

Network Security Settings

Caution

The security features provided with this system do not guarantee a completely secure system.

The Firewall Configuration screen presented during the RHEL7 installation applies to the public network interface and should be set according to your local standards.

The RHEL7 installer allows you to select some, but not all, of the security settings needed to support Scyld ClusterWare. The remaining security settings must be made post-installation; see the Section called *Trusted Devices*

Scyld has the following recommendations for configuring the firewall:

- Set a firewall for the public network interface (typically eth0).
- If you chose to install a firewall, you must make the private network interface (typically eth1) a "trusted device" to allow all traffic to pass to the internal private cluster network; otherwise, the cluster will not run correctly. This setting must be made post-installation.
- The Red Hat installer configures the firewall with most services disabled. If you plan to use SSH to connect to the master node, be sure to select SSH from the list of services in the Firewall Configuration screen to allow SSH traffic to pass through the firewall.

Red Hat RHEL7 or CentOS7 Installation

Scyld ClusterWare depends on the prior installation of certain RHEL7 or CentOS7 packages from the base distribution. Ideally, each Scyld ClusterWare rpm names every dependency, which means that when you use **/usr/bin/yum** to install Scyld ClusterWare, **yum** attempts to gracefully install those dependencies if the base distribution yum repository (or repositories) are accessible and the dependencies are found. If a dependency cannot be installed, then the Scyld installation will fail with an error message that describes what rpm(s) or file(s) are needed.

Caution

Check the Scyld ClusterWare *Release Notes* for your release to determine whether you must update your Red Hat or CentOS base installation. If you are not familiar with the **yum** command, see the Section called *Updating Red Hat or CentOs Installation* in Chapter 3 for details on the update procedures.

Scyld ClusterWare Installation

Scyld ClusterWare is installed using the Penguin Yum repository http://updates.penguincomputing.com/clusterware/. Each Scyld ClusterWare release is continuously tested with the latest patches from Red Hat and CentOS7. Before installing or updating your master node, be sure to visit the Support Portal to determine if any patches should be excluded due to incompatibility with ClusterWare. Such incompatibilities should be rare. Then, update RHEL7 or CentOS7 on your master node before proceeding (excluding incompatible packages if necessary) with installing or updating your Scyld ClusterWare

Configure Yum To Support ClusterWare

The Yum repo configuration file for Scyld ClusterWare must be downloaded from the Penguin Computing Support Portal and properly configured:

1. Login to the Support Portal at http://www.penguincomputing.com/support
2. Click on *Download your Yum repo file* to download this `clusterware.repo` file and place the it in the `/etc/yum.repos.d/` directory.
3. Set the permissions:

   ```
   [root@scyld ~]# chmod 644 /etc/yum.repos.d/clusterware.repo
   ```

With this setup complete, your master node is ready to retrieve Scyld ClusterWare installations and updates.

Install ClusterWare

You can use Yum to install ClusterWare and all updates up to and including the latest ClusterWare release, assuming you have updated your RHEL7 or CentOS7 base distribution as prescribed in the ClusterWare *Release Notes*.

1. Verify the version you are running with the following:

   ```
   [root@scyld ~]# cat /etc/redhat-release
   ```

This should return a string similar to “Red Hat Enterprise Linux Server release 7.4“ or “CentOS Linux release 7.4.1708 (Core)“.

2. Install the Scyld ClusterWare script that simplifies installing (and later updating) software, then execute that script:

   ```
   [root@scyld ~]# yum install install-scyld
   [root@scyld ~]# install-scyld
   ```

3. Configure the network for Scyld ClusterWare: edit `/etc/beowulf/config` to specify the cluster interface, the maximum number of compute nodes, and the beginning IP address of the first compute node. See the remainder of this guide and the *Administrator’s Guide* for details.
4. Reboot your system.
5. To verify that ClusterWare was installed successfully, do the following:

   ```
   [root@scyld ~]# uname -r
   ```

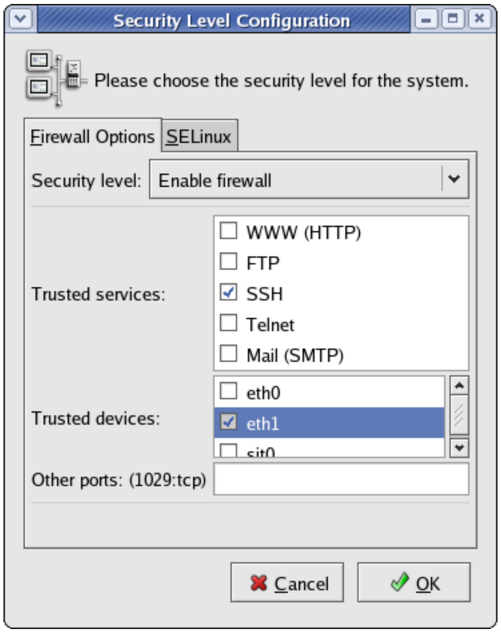
   The result should match the specific ClusterWare kernel version noted in the *Release Notes*.

Trusted Devices

If you chose to install a firewall, you must make the private network interface (typically eth1) a "trusted device" to enable all traffic on this interface to pass through the firewall; otherwise, the cluster will not run properly. This must be done post-installation.

1. After you have installed Red Hat and Scyld ClusterWare, reboot the system and log in as "root".
2. Access the security settings through the Red Hat **Applications -> System Settings -> Security Level** menu options.
3. In the Security Level Configuration dialog box, make sure the private interface is checked in the "trusted devices" list, then click **OK**.

Tip: If you plan to use SSH to connect to the master node, be sure that SSH is checked in the "trusted services" list.



Please choose the security level for the system.

Security level: Enable firewall

Trusted services:
□ WWW (HTTP)
□ FTP
☑ SSH
□ Telnet
□ Mail (SMTP)

Trusted devices:
□ eth0
☑ eth1
□ sit0

Other ports: (1029:tcp)

Figure 2-3. Security Settings Post-Installation

You are now ready to boot and configure the compute nodes, as described in the next section.

Compute Nodes

In a Scyld cluster, the master node controls booting, provisioning, and operation of the compute nodes. You do not need to explicitly install Scyld ClusterWare on the compute nodes.

Scyld requires configuring your compute nodes to boot via PXE and using the auto-activate node options, so that each node automatically joins the cluster as it powers on. Nodes do not need to be added manually.

1. If you are not already logged in as root, log into the master node using the root username and password.
2. Use the command **bpstat -U** in a terminal window on the master node to view a continuously updated table of node status information.
3. Set the BIOS on each compute node to boot via PXE. Using the auto-activate option with PXE booting allows each node to automatically boot and join the cluster as it powers on.
4. Node numbers are initially assigned in order of connection with the master node. Boot the compute nodes by powering them up in the order you want them to be numbered, typically one-by-one from the top of a rack downwards (or from the bottom up). You can reorder nodes later as desired; see the *Administrator's Guide*.
5. The nodes transition through the boot phases. As the nodes join the cluster and are ready for use, they will be shown as "Up" by the **bpstat -U** command.

The cluster is now fully operational with disk-less compute nodes. See Chapter 4 for more about **bpstat** and node states.

Notes

1. http://updates.penguincomputing.com/clusterware/

2. http://www.penguincomputing.com/support

Chapter 3. Detailed Installation Instructions

This chapter provides detailed instructions for installing Scyld ClusterWare. This software installation is intended for the first computer ("node") of the cluster, which functions as the "master node" to control and monitor other nodes and distribute jobs.

Scyld ClusterWare is installed on the master node that is running with a base distribution of RHEL7 or CentOS7.

It is assumed that you are familiar with the concepts outlined in the previous chapters, and that you have correctly assembled the hardware for your cluster. If this is not the case, please refer to the previous chapters to acquaint yourself with Scyld ClusterWare, and then verify that your hardware configuration is set up properly.

Red Hat Installation Specifics

During a RHEL7 installation, you have the option to configure various aspects of the installation to support Scyld ClusterWare. Important points include the following:

- *Disk partitioning* — Scyld recommends letting the installer automatically partition the disk; refer to the Red Hat documentation if you plan to manually partition instead.
- *Network interface configuration* — To support your Scyld cluster, you need to configure one interface dedicated to the external public network (typically eth0) and one to your internal private cluster network (typically eth1). Detailed instructions are provided in the section on Network Interface Configuration later in this chapter.
- *Network security settings* — You can configure some of your firewall settings during a RHEL7 installation. Other settings needed to support a Scyld cluster must be made post-installation. Detailed instructions are provided in the sections on Network Security Settings and Trusted Devices later in this chapter.
- *Package group selection* — Scyld recommends installing all Red Hat packages. See the Section called *Package Group Selection* later in this chapter.

The following sections provide instructions and/or recommendations for specific portions of the RHEL7 installation that are relevant to an optimal Scyld ClusterWare installation. This guide does not cover all steps in the RHEL7 installation; you should refer to the Red Hat documentation for more complete information.

Network Interface Configuration

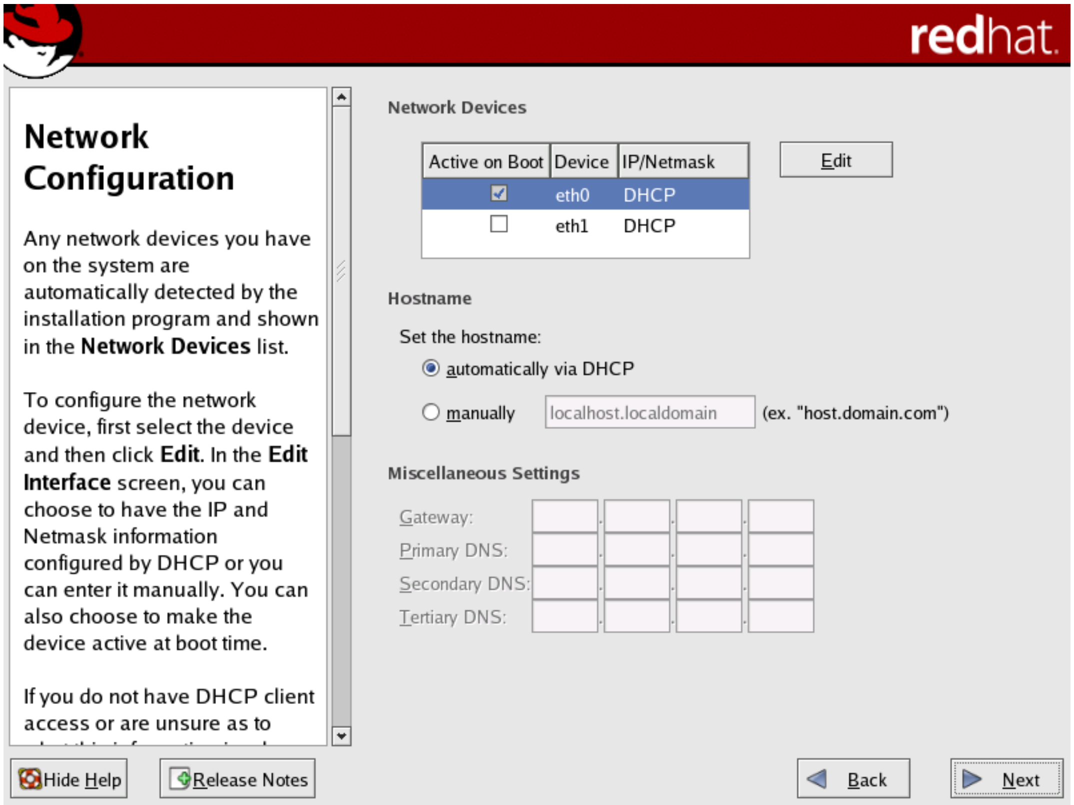
> **Tip:** To begin, you must know which interface is connected to the public network and which is connected to the private network. Typically, the public interface is eth0 and the private interface is eth1.

A typical Scyld cluster has one interface dedicated to the external public network (typically eth0) and one dedicated to your internal private cluster network (typically eth1). It is important to properly to configure both of these interfaces to support your Scyld ClusterWare installation.

The network interface configuration screen will be presented to you during a RHEL7 installation. For an existing Red Hat installation, you can access the network configuration screens through the Red Hat **Applications -> System Settings -> Network** menu options.

Cluster Public Network Interface

DHCP is selected by default for all network devices, as shown below in the Red Hat Network Configuration Screen. For the public network interface (typically eth0), this option is recommended.



Network Configuration

Any network devices you have on the system are automatically detected by the installation program and shown in the **Network Devices** list.

To configure the network device, first select the device and then click **Edit**. In the **Edit Interface** screen, you can choose to have the IP and Netmask information configured by DHCP or you can enter it manually. You can also choose to make the device active at boot time.

If you do not have DHCP client access or are unsure as to

Network Devices

Active on Boot	Device	IP/Netmask
☑	eth0	DHCP
☐	eth1	DHCP

Hostname

Set the hostname:

● automatically via DHCP

○ manually localhost.localdomain (ex. "host.domain.com")

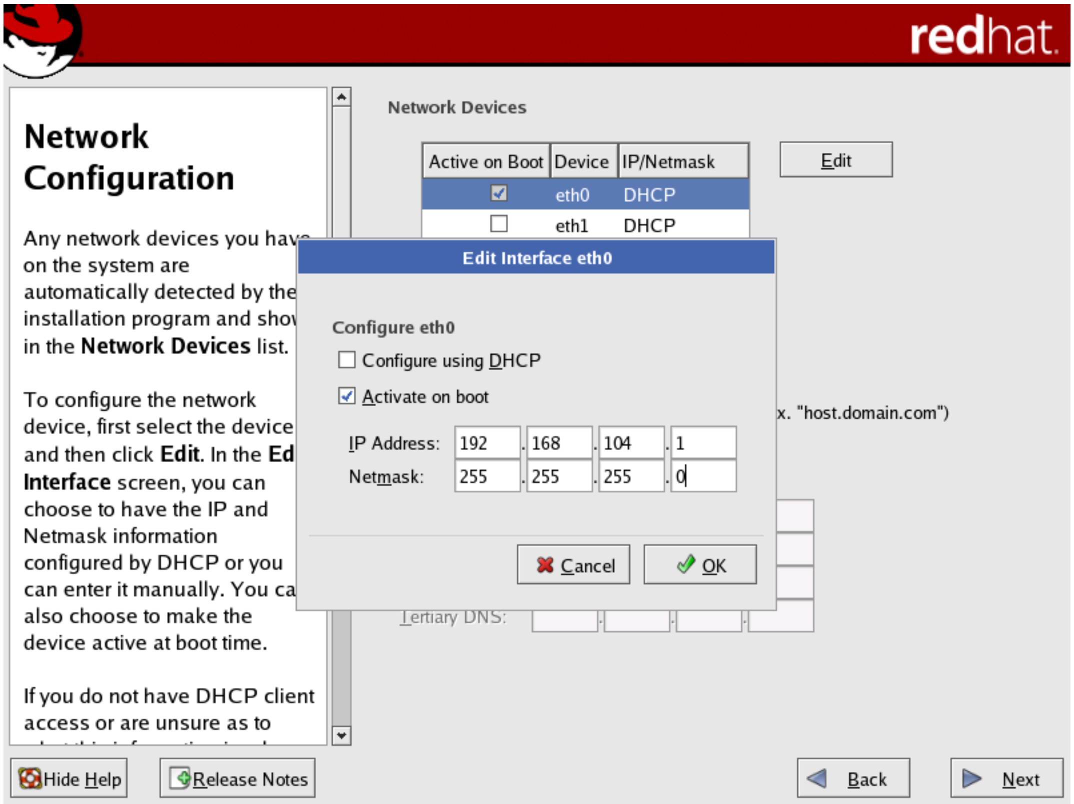
Miscellaneous Settings

Gateway: . . .
Primary DNS: . . .
Secondary DNS: . . .
Tertiary DNS: . . .

Figure 3-1. Public Network Interface (DHCP Default is Recommended)

However, if your external network is set up to use static IP addresses, then follow these steps to manually configure the interface:

1. In the Network Configuration screen, select the public network interface (typically eth0) in the Network Devices list, then click **Edit** to open the Edit Interface dialog box.



Network Configuration

Any network devices you hav on the system are automatically detected by the installation program and sho in the **Network Devices** list.

To configure the network device, first select the device and then click **Edit**. In the **Ed Interface** screen, you can choose to have the IP and Netmask information configured by DHCP or you can enter it manually. You ca also choose to make the device active at boot time.

If you do not have DHCP client access or are unsure as to

Network Devices

Active on Boot	Device	IP/Netmask
☑	eth0	DHCP
□	eth1	DHCP

Edit Interface eth0

Configure eth0

□ Configure using DHCP

☑ Activate on boot

IP Address: 192 . 168 . 104 . 1

Netmask: 255 . 255 . 255 . 0

x. "host.domain.com")

Tertiary DNS:

Figure 3-2. Public Network Interface (Manual Configuration is Optional)

2. In the Edit Interface dialog box:

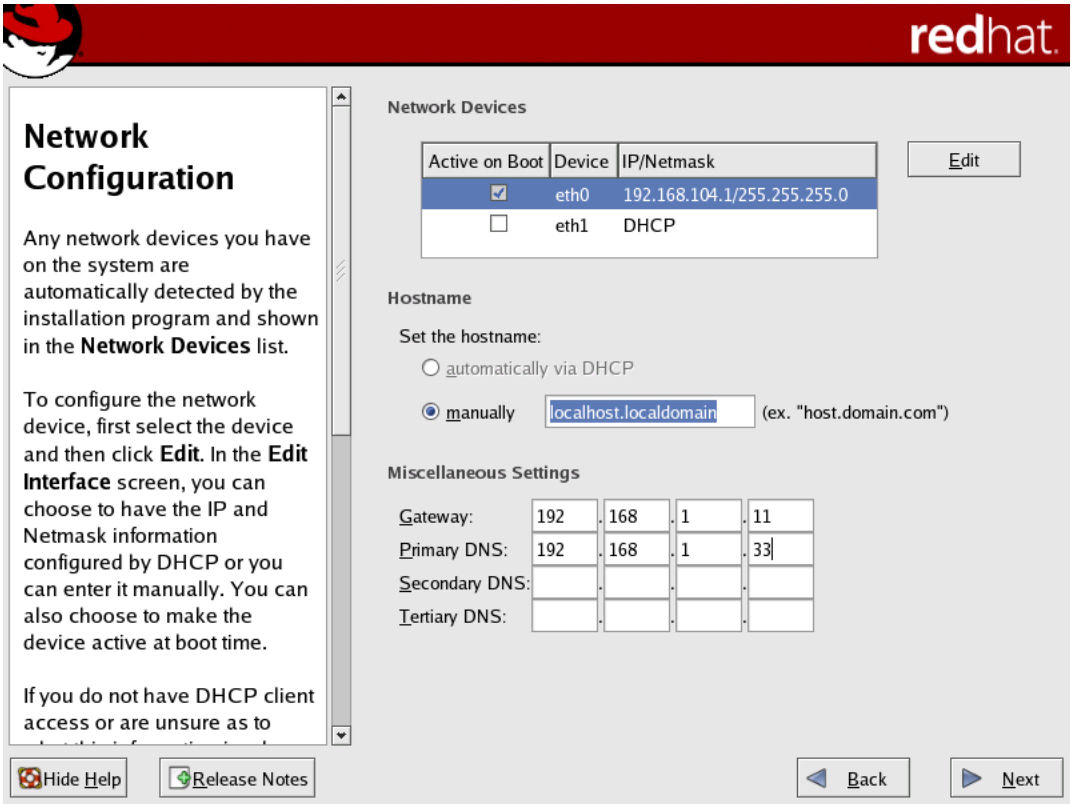
   a. Select the *Activate on boot* checkbox to initialize the specific network device at boot-time.

   b. Specify the IP address and netmask provided by your network administrator.

   When you have completed these settings, click **OK** to return to the Network Configuration screen.

3. In the *Set the hostname* area of the Network Configuration screen, select the **manually** radio button and provide a host name.

4. In the *Miscellaneous Settings* area of the screen, enter the gateway and primary DNS IP addresses provided by your Network Administrator.



Network Configuration

Any network devices you have on the system are automatically detected by the installation program and shown in the **Network Devices** list.

To configure the network device, first select the device and then click **Edit**. In the **Edit Interface** screen, you can choose to have the IP and Netmask information configured by DHCP or you can enter it manually. You can also choose to make the device active at boot time.

If you do not have DHCP client access or are unsure as to

Network Devices

Active on Boot	Device	IP/Netmask
☑	eth0	192.168.104.1/255.255.255.0
□	eth1	DHCP

Hostname

Set the hostname:

○ automatically via DHCP

● manually localhost.localdomain (ex. "host.domain.com")

Miscellaneous Settings

Gateway:	192	168	1	11
Primary DNS:	192	168	1	33
Secondary DNS:				
Tertiary DNS:				

Figure 3-3. Public Network Interface (Miscellaneous Settings for Manual Configuration)

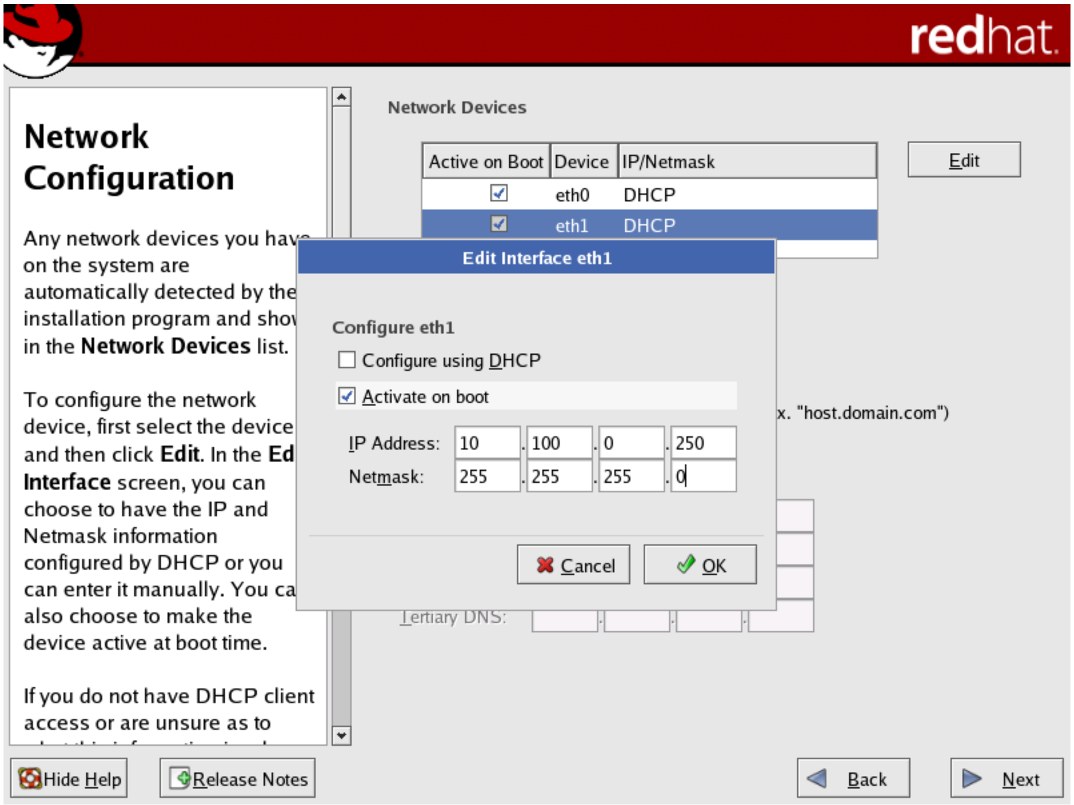
Cluster Private Network Interface

Caution

For the private network interface (typically eth1), DHCP is shown as default, but this option cannot be used. The configuration tool **Beonetconf** requires a static IP address for the private interface. Therefore, you must configure the network interface manually and assign a static IP address and netmask.

The cluster will not run correctly unless the private network interface is trusted. You can set this interface as a "trusted device" when configuring the network security settings post-installation; see the Section called *Trusted Devices*.

1. In the Network Configuration screen, select the private network interface (typically eth1) in the Network Devices list, then click **Edit** to open the Edit Interface dialog box.



Network Configuration

Any network devices you hav on the system are automatically detected by the installation program and sho in the **Network Devices** list.

To configure the network device, first select the device and then click **Edit**. In the **Ed Interface** screen, you can choose to have the IP and Netmask information configured by DHCP or you can enter it manually. You ca also choose to make the device active at boot time.

If you do not have DHCP client access or are unsure as to

Network Devices

Active on Boot	Device	IP/Netmask
☑	eth0	DHCP
☑	eth1	DHCP

x. "host.domain.com")

Tertiary DNS:

Edit Interface eth1

Configure eth1

☐ Configure using DHCP

☑ Activate on boot

IP Address: 10 . 100 . 0 . 250

Netmask: 255 . 255 . 255 . 0

Figure 3-4. Private Network Interface (Manual Configuration Required)

2. In the Edit Interface dialog box:

   a. Select the *Activate on boot* checkbox to initialize the specific network device at boot-time.

   b. Specify a static IP address. We recommend using a non-routable address (such as 192.168.x.x, 172.16.x.x to 172.30.x.x, or 10.x.x.x).

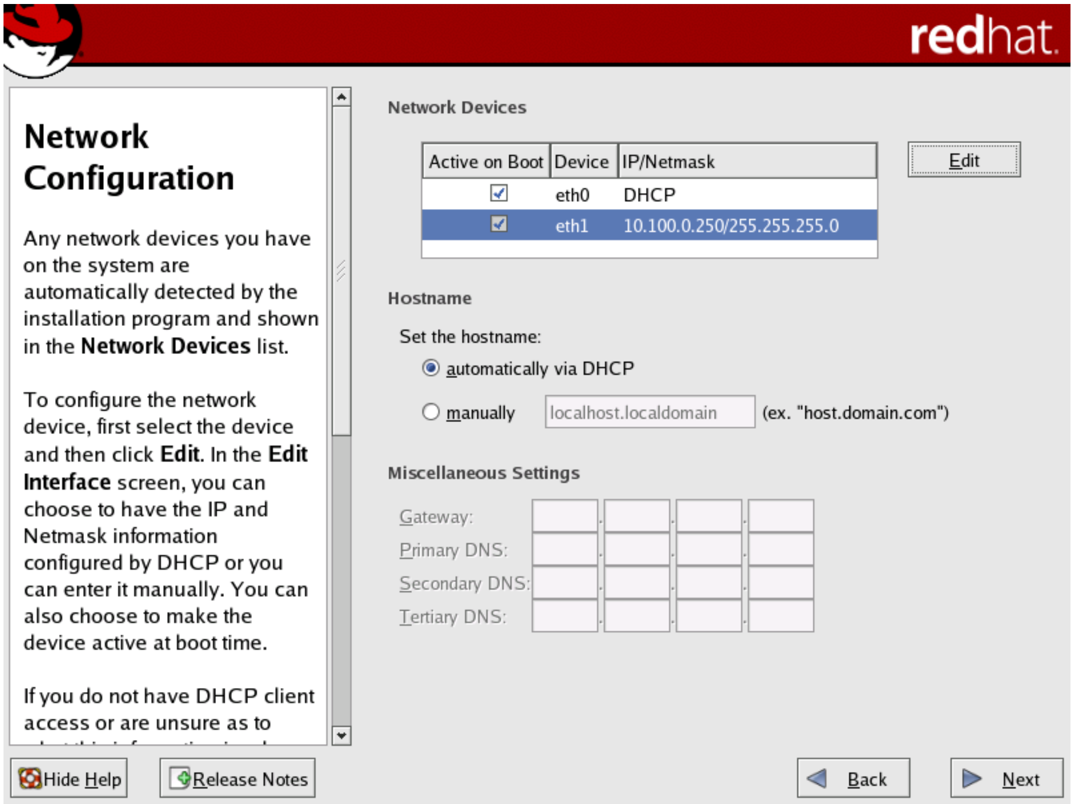
   c. If the public subnet is non-routable, then use a different non-routable range for the private subnet (e.g., if the public subnet is 192.168.x.x, then use 172.16.x.x to 172.30.x.x or 10.x.x.x for the private subnet).

   d. Once you have specified the IP address, set the subnet mask based on this address. The subnet mask must accommodate a range large enough to contain all of your compute nodes.

   When you have completed these settings, click **OK** to return to the Network Configuration screen.

3. In the *Set the hostname* area of the Network Configuration screen, you have the option to set the hostname automatically via the DHCP server or to provide one manually; this can be done according to your local standards.

The following figure illustrates a completed typical configuration for both the public and private network interfaces.



Network Configuration

Any network devices you have on the system are automatically detected by the installation program and shown in the **Network Devices** list.

To configure the network device, first select the device and then click **Edit**. In the **Edit Interface** screen, you can choose to have the IP and Netmask information configured by DHCP or you can enter it manually. You can also choose to make the device active at boot time.

If you do not have DHCP client access or are unsure as to

Network Devices

Active on Boot	Device	IP/Netmask
☑	eth0	DHCP
☑	eth1	10.100.0.250/255.255.255.0

Hostname

Set the hostname:

● automatically via DHCP

○ manually localhost.localdomain (ex. "host.domain.com")

Miscellaneous Settings

Gateway:

Primary DNS:

Secondary DNS:

Tertiary DNS:

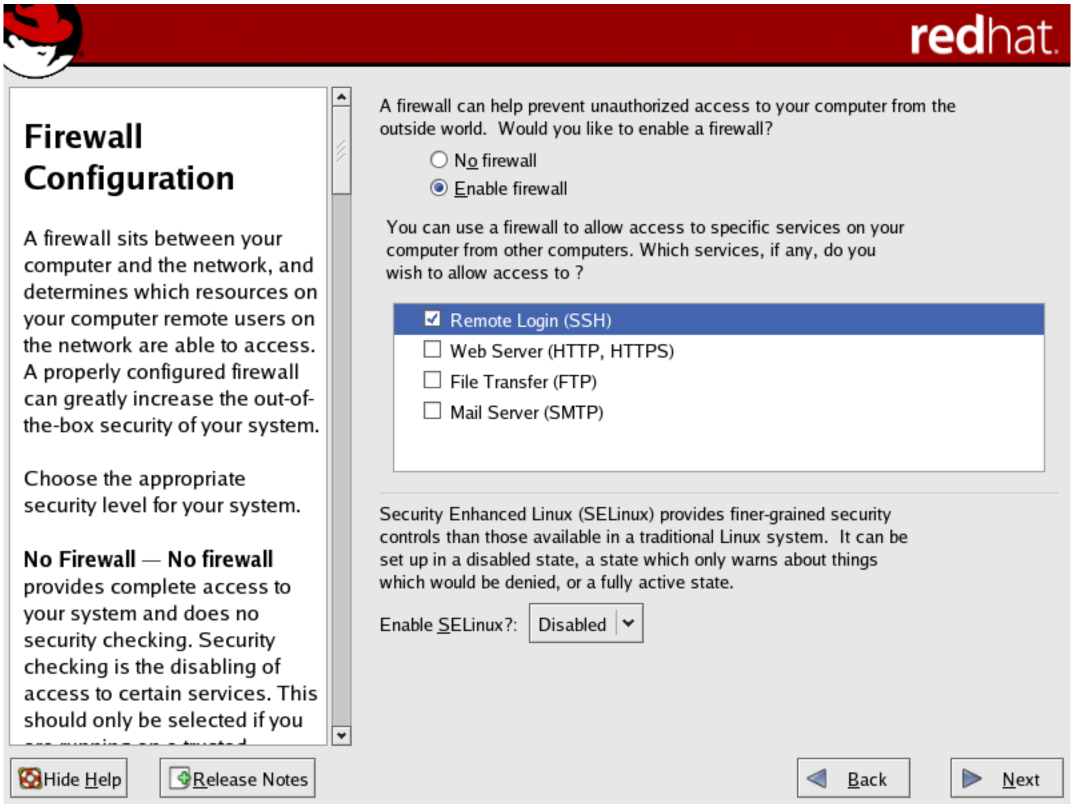
Figure 3-5. Public and Private Network Interfaces (Typical Configuration Completed)

Network Security Settings

> **Caution**
>
> The security features provided with this system do not guarantee a completely secure system.

The Firewall Configuration screen presented during the RHEL7 installation applies to the public network interface and should be set according to your local standards. This screen allows you to customize several aspects of the firewall that protects your cluster from possible network security violations.

The RHEL7 installer allows you to select some, but not all, of the security settings needed to support Scyld ClusterWare. The remaining security settings must be made post-installation; see the Section called *Trusted Devices*.



Firewall Configuration

A firewall sits between your computer and the network, and determines which resources on your computer remote users on the network are able to access. A properly configured firewall can greatly increase the out-of-the-box security of your system.

Choose the appropriate security level for your system.

No Firewall — No firewall provides complete access to your system and does no security checking. Security checking is the disabling of access to certain services. This should only be selected if you are running on a trusted

A firewall can help prevent unauthorized access to your computer from the outside world. Would you like to enable a firewall?

○ No firewall
● Enable firewall

You can use a firewall to allow access to specific services on your computer from other computers. Which services, if any, do you wish to allow access to ?

☑ Remote Login (SSH)
□ Web Server (HTTP, HTTPS)
□ File Transfer (FTP)
□ Mail Server (SMTP)

Security Enhanced Linux (SELinux) provides finer-grained security controls than those available in a traditional Linux system. It can be set up in a disabled state, a state which only warns about things which would be denied, or a fully active state.

Enable SELinux?: Disabled

Figure 3-6. Security Settings During Installation

Scyld recommends setting a firewall for the public network interface (typically eth0). You can configure the following security settings during the Red Hat install:

1. Select from the following firewall options:

   a. *No Firewall* — Allows all connections to your system and does no security checking. This option is not recommended unless you plan to configure your firewall after the installation.

   b. *Enable Firewall* — Blocks any connections to your system that are not defaults or explicitly defined by you. By default, connections are allowed in response to outbound requests, such as DNS replies or DHCP requests.

2. Select services for which you want to allow possible connections. You can select any combination of the services listed.

   Tip: If you plan to use SSH to connect to the master node, be sure that SSH is checked in the *Trusted Services* list.

3. Set the **Enable SELinux?** dropdown to "Disabled".

If you chose to install a firewall, you must make the private network interface (typically eth1) a "trusted device" to enable all traffic on this interface to pass through the firewall. See the Section called *Trusted Devices*.

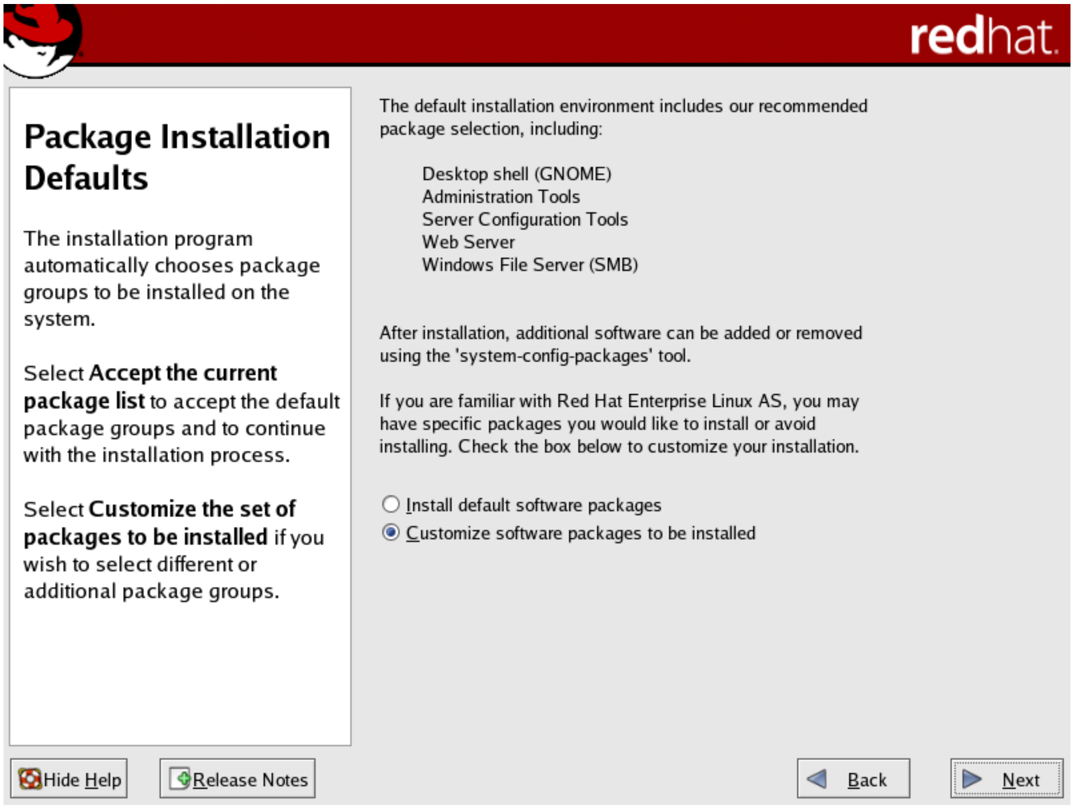
Package Group Selection

Caution

Scyld ClusterWare depends on certain Red Hat packages, and the Scyld installation may fail if the necessary Red Hat packages are not installed. Therefore, Scyld recommends that you install all Red Hat packages.

The Red Hat package selection screens enable you to select the particular software packages that you wish to install.

1. In the Package Installation Defaults screen, select the **Customize...** option.



Package Installation Defaults

The installation program automatically chooses package groups to be installed on the system.

Select **Accept the current package list** to accept the default package groups and to continue with the installation process.

Select **Customize the set of packages to be installed** if you wish to select different or additional package groups.

The default installation environment includes our recommended package selection, including:

Desktop shell (GNOME)
Administration Tools
Server Configuration Tools
Web Server
Windows File Server (SMB)

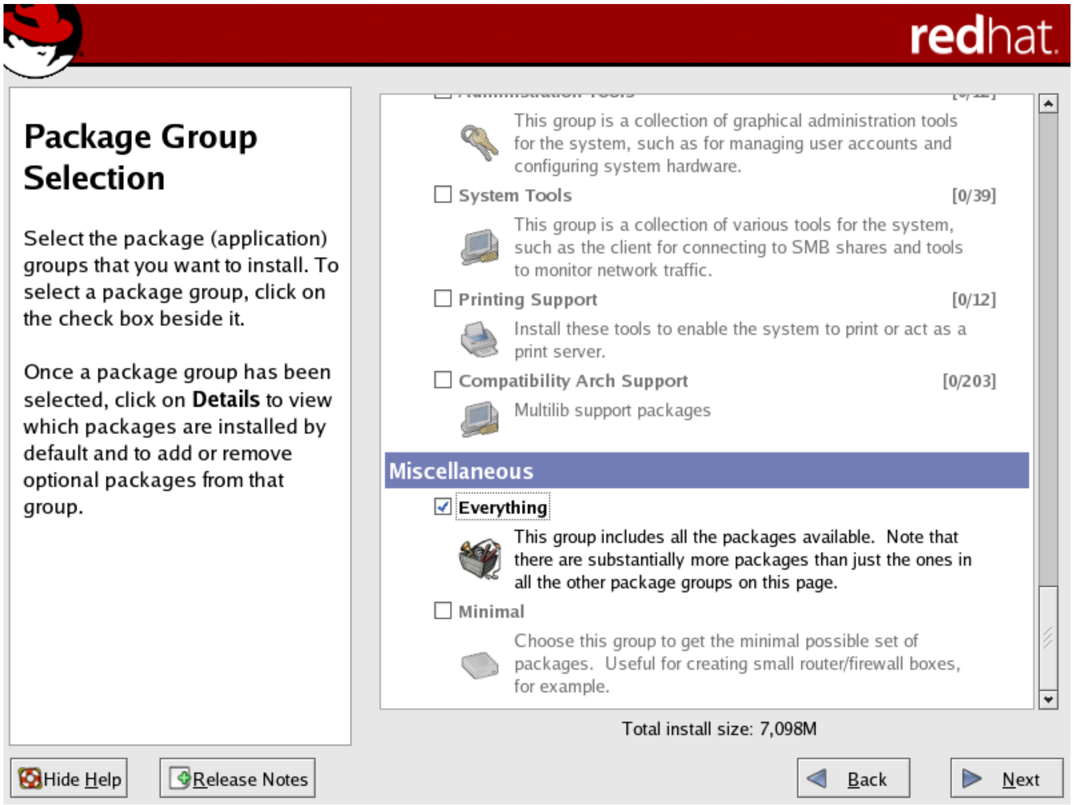
After installation, additional software can be added or removed using the 'system-config-packages' tool.

If you are familiar with Red Hat Enterprise Linux AS, you may have specific packages you would like to install or avoid installing. Check the box below to customize your installation.

○ Install default software packages
● Customize software packages to be installed

Figure 3-7. Customize Package Installation

2. In the Package Group Selection screen, scroll down to the *Miscellaneous* section. Select the **Everything** checkbox, then continue the installation process.



Package Group Selection

Select the package (application) groups that you want to install. To select a package group, click on the check box beside it.

Once a package group has been selected, click on **Details** to view which packages are installed by default and to add or remove optional packages from that group.

This group is a collection of graphical administration tools for the system, such as for managing user accounts and configuring system hardware.

□ System Tools [0/39]

This group is a collection of various tools for the system, such as the client for connecting to SMB shares and tools to monitor network traffic.

□ Printing Support [0/12]

Install these tools to enable the system to print or act as a print server.

□ Compatibility Arch Support [0/203]

Multilib support packages

Miscellaneous

☑ Everything

This group includes all the packages available. Note that there are substantially more packages than just the ones in all the other package groups on this page.

□ Minimal

Choose this group to get the minimal possible set of packages. Useful for creating small router/firewall boxes, for example.

Total install size: 7,098M

Figure 3-8. Install Everything

Tip: To update an existing Red Hat installation to include all packages, insert the first Red Hat CD and invoke the Red Hat update program. Check the *Everything* box in the Package Group Selection screen, then continue with the update process.

Updating Red Hat or CentOs Installation

Update RHEL7 or CentOS7 either using **yum**, or using Red Hat or CentOS distribution media. Note that Penguin continually tests ClusterWare with new patches from Red Hat and CentOS. Visit the Penguin Computing Support Portal at http://www.penguincomputing.com/support to see the most recent errata fix tested with ClusterWare, and see any cautions about updated packages which may cause problems with ClusterWare.

Updating Using Yum

Use the following command:

```
[root@scyld ~]# yum update --disablerepo=cw*
```

(**--disablerepo-cw*** is used above in case the ClusterWare repo is already installed in `/etc/yum.repos.d`, you must exclude it during the **yum update**). You can also exclude other packages using the **--exclude=$package** parameter. See the **yum** man page for instructions on using **yum**. The CentOS web site also provides an online manual for **yum** at http://www.centos.org/docs/4/html/yum/.

Updating Using Media

If you update your system via distribution media, be sure to select an "upgrade install" rather than a "full install", then follow the instructions provided with the media.

> **Tip:** The just-installed newest base distribution kernel becomes the default in `/etc/grub.conf`. However, the Scyld ClusterWare distribution includes a customized kernel that must be the kernel that is booted when running Scyld ClusterWare.

Scyld ClusterWare Installation

Scyld ClusterWare is installed using the Penguin Yum repository http://updates.penguincomputing.com/clusterware/. Each Scyld ClusterWare release is continuously tested with the latest patches from Red Hat and CentOS7. Before installing or updating your master node, be sure to visit the Support Portal to determine if any patches should be excluded due to incompatibility with ClusterWare. Such incompatibilities should be rare. Then, update RHEL7 or CentOS7 on your master node before proceeding (excluding incompatible packages if necessary) with installing or updating your Scyld ClusterWare

Configure Yum To Support ClusterWare

The Yum repo configuration file for Scyld ClusterWare must be downloaded from the Penguin Computing Support Portal and properly configured:

1. Login to the Support Portal at http://www.penguincomputing.com/support
2. Click on *Download your Yum repo file* to download this `clusterware.repo` file and place the it in the `/etc/yum.repos.d/` directory.
3. Set the permissions:

   ```
   [root@scyld ~]# chmod 644 /etc/yum.repos.d/clusterware.repo
   ```

With this setup complete, your master node is ready to retrieve Scyld ClusterWare installations and updates.

Install ClusterWare

You can use Yum to install ClusterWare and all updates up to and including the latest ClusterWare release, assuming you have updated your RHEL7 or CentOS7 base distribution as prescribed in the ClusterWare *Release Notes*.

1. Verify the version you are running with the following:

   ```
   [root@scyld ~]# cat /etc/redhat-release
   ```

   This should return a string similar to “Red Hat Enterprise Linux Server release 7.4“ or “CentOS Linux release 7.4.1708 (Core)“.
2. Install the Scyld ClusterWare script that simplifies installing (and later updating) software, then execute that script:

```
[root@scyld ~]# yum install install-scyld
[root@scyld ~]# install-scyld
```

3. Configure the network for Scyld ClusterWare: edit `/etc/beowulf/config` to specify the cluster interface, the maximum number of compute nodes, and the beginning IP address of the first compute node. See the remainder of this guide and the *Administrator's Guide* for details.
4. Reboot your system.
5. To verify that ClusterWare was installed successfully, do the following:

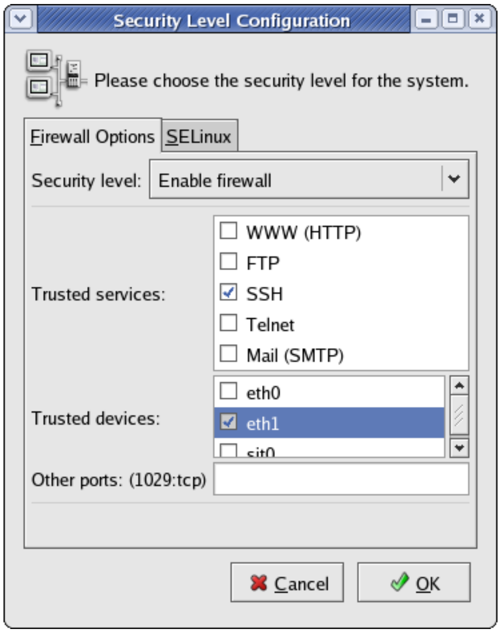
   ```
   [root@scyld ~]# uname -r
   ```
   The result should match the specific ClusterWare kernel version noted in the *Release Notes*.

Trusted Devices

If you chose to install a firewall, you must make the private network interface (typically eth1) a "trusted device" to enable all traffic on this interface to pass through the firewall; otherwise, the cluster will not run properly. This must be done post-installation.

1. After you have installed Red Hat and Scyld ClusterWare, reboot the system and log in as "root".
2. Access the security settings through the Red Hat **Applications -> System Settings -> Security Level** menu options.
3. In the Security Level Configuration dialog box, make sure the private interface is checked in the "trusted devices" list, then click **OK**.

> **Tip:** If you plan to use SSH to connect to the master node, be sure that SSH is checked in the "trusted services" list.



Please choose the security level for the system.

Security level: Enable firewall

Trusted services:

☐ WWW (HTTP)
☐ FTP
☑ SSH
☐ Telnet
☐ Mail (SMTP)

Trusted devices:

☐ eth0
☑ eth1
☐ sit0

Other ports: (1029:tcp)

Figure 3-9. Security Settings Post-Installation

You are now ready to boot your compute nodes.

Enabling Access to External License Servers

1. Enable ipforward in the /etc/beowulf/config file. The line should read as follows:

   ipforward yes

2. Restart the cluster services as "root":

```
[root@scyld ~]# systemctl restart clusterware
```

Post-Installation Configuration

Following a successful update or install of Scyld ClusterWare, you may need to make one or more configuration changes, depending upon the local requirements of your cluster. Larger cluster configurations have additional issues to consider. Accordingly, review the Release Notes sections titled *Post-Installation Configuration Issues* and *Post-Installation Configuration Issues For Large Clusters* for important detailed information.

Scyld ClusterWare Updates

You can use Yum update to update ClusterWare once you have upgraded your RHEL7 or CentOS7 base distribution. See the Section called *Updating Red Hat or CentOs Installation* for details on updating your base distribution, and the Section called *Scyld ClusterWare Installation* for how to set up the Yum repo configuration files.

To verify which distribution you are currently running, do the following:

```
[root@scyld ~]# cat /etc/redhat-release
```

Updating ClusterWare

1. It is advisable to update the base distribution prior to updating Scyld ClusterWare, taking care to exclude the base distribution's kernel-* packages to avoid potentially updating to a newer kernel than is currently available in the Scyld ClusterWare yum repos:

   ```
   [root@scyld ~]#
   yum --disablerepo=cw* --exclude=kernel-* update
   ```

2. Update the Scyld ClusterWare package that contains a useful script that simplifies updating ClusterWare, then execute that script:

   ```
   [root@scyld ~]#yum update install-scyld
   [root@scyld ~]#install-scyld -u
   ```

3. Compare `/etc/beowulf/config`, which remains untouched by the Scyld ClusterWare update, with the new `config.rpmnew` (if that file exists), and examine the differences:

   ```
   [root@scyld ~]#cd /etc/beowulf
   [root@scyld ~]#diff config config.rpmnew
   ```

   and carefully merge the `config.rpmnew` differences into `/etc/beowulf/config`. Similarly, the preexisting `/etc/beowulf/fstab` may have been saved as `fstab.rpmsave` if it was locally modified. If so, merge those local changes back into `/etc/beowulf/fstab`

4. Reboot your system.

5. To verify that ClusterWare was installed successfully, do the following:

   ```
   [root@scyld ~]# uname -r
   ```

   The result should match the ClusterWare kernel version noted in the *Release Notes*.

6. Restart the compute nodes.

Notes

1. http://www.penguincomputing.com/support
2. http://www.centos.org/docs/4/html/yum/
3. http://updates.penguincomputing.com/clusterware/
4. http://www.penguincomputing.com/support

Chapter 4. Cluster Verification Procedures

Once the master node and compute nodes have been configured and rebooted, you should run through the cluster verification to identify common software and hardware configuration problems. This chapter describes the Scyld ClusterWare tools for monitoring cluster status and running jobs across the cluster.

Cluster verification is generally required by reseller technical support when starting on a new issue. When you call your reseller for support, they will require that you have completed the cluster verification procedures outlined in this chapter, and that you capture information using the **beosi** script.

Also see the *Administrator's Guide* and the *User's Guide* for more detailed information.

Monitoring Cluster Status

You can monitor the status of the nodes in your cluster using the **bpstat** or **beostatus** commands.

bpstat

The **bpstat** command, run at a shell prompt on the master node, shows a table of status information for each node in the cluster. You do not need to be a privileged user to use this command.

Following is an example of the outputs from **bpstat** for a cluster with 10 compute nodes.

```
[root@cluster ~]# bpstat
  Node(s)      Status        Mode       User        Group
  5-9          down          ---------- root        root
  4            up            ---x--x--x any         any
  0-3          up            ---x--x--x root        root
```

Some things to keep in mind for **bpstat**:

- Ensure that each node is listed as *up*. The node count is based upon the *nodes* and *iprange* entries in the `/etc/beowulf/config` configuration file.
- Nodes that have not yet been configured are marked as *down*.
- Nodes currently booting are temporarily shown with a status of *boot*.
- An *error* status indicates a node initialization problem. Check for error messages in the log file `/var/log/beowulf/node.N` (where *N* is the node number). Typical problems are failing network connections, unpartitioned harddrives, or unavailable network file systems.

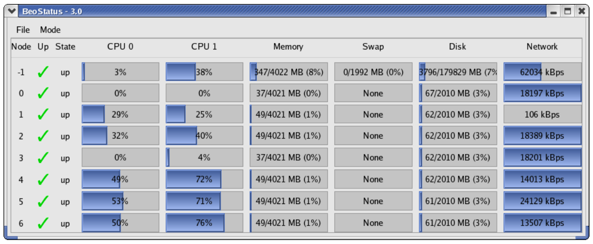
BeoStatus

The BeoStatus tool is a graphical user interface (GUI) program. You can start it by clicking the BeoStatus icon on the desktop.

Alternatively, type the command **beostatus** in a terminal window on the master node; you do not need to be a privileged user to use this command.

You can also view the status of the cluster in text mode by typing the command **beostatus -c** at a terminal window on the master node.

The default BeoStatus GUI mode (shown below) is a tabular format known as the "Classic" display. Each row corresponds to a different node, with specific state and resource usage information displayed per node.



Node	Up	State	CPU 0	CPU 1	Memory	Swap	Disk	Network
-1	✓	up	3%	38%	347/4022 MB (8%)	0/1992 MB (0%)	3796/179829 MB (7%	62034 kBps
0	✓	up	0%	0%	37/4021 MB (0%)	None	67/2010 MB (3%)	18197 kBps
1	✓	up	29%	25%	49/4021 MB (1%)	None	62/2010 MB (3%)	106 kBps
2	✓	up	32%	40%	49/4021 MB (1%)	None	62/2010 MB (3%)	18389 kBps
3	✓	up	0%	4%	37/4021 MB (0%)	None	62/2010 MB (3%)	18201 kBps
4	✓	up	49%	72%	49/4021 MB (1%)	None	62/2010 MB (3%)	14013 kBps
5	✓	up	53%	71%	49/4021 MB (1%)	None	61/2010 MB (3%)	24129 kBps
6	✓	up	50%	76%	49/4021 MB (1%)	None	61/2010 MB (3%)	13507 kBps

Figure 4-1. BeoStatus in the "Classic" Display Mode

You should sanity-check the information shown in the BeoStatus window. The configured nodes that are powered up (those with a green checkmark in the "Up" column) should show expected values in the associated usage columns. When there are no active jobs on your cluster, the CPU and Network columns should be fairly close to zero. The memory usage columns (Memory, Swap, and Disk) should be showing reasonable values.

- *Node* — The node's assigned node number, starting at zero. Node -1, if shown, is the master node. The total number of node entries shown is set by the "iprange" or "nodes" keywords in the file `/etc/beowulf/config`, rather than the number of detected nodes. The entry for an inactive node displays the last reported data in a grayed-out row.
- *Up* — A graphical representation of the node's status. A green checkmark is shown if the node is up and available. Otherwise, a red "X" is shown.
- *State* — The node's last known state. This should agree with the state reported by both the **bpstat** and **beostatus** commands.
- *CPU "X"* — The CPU loads for the node's processors; at minimum, this indicates the CPU load for the first processor in each node. Since it is possible to mix uni-processor and multi-processor machines in a Scyld cluster, the number of CPU load columns is equal to the maximum number of processors for any node in your cluster. The label "N/A" will be shown for nodes with less than the maximum number of processors.
- *Memory* — The node's current memory usage.
- *Swap* — The node's current swap space (virtual memory) usage.
- *Disk* — The node's harddrive usage. If a RAM disk is used, the maximum value shown is one-half the amount of physical memory. As the RAM disk competes with the kernel and application processes for memory, not all the RAM may be available.
- *Network* — The node's network bandwidth usage. The total amount of bandwidth available is the sum of all network interfaces for that node.

Running Jobs Across the Cluster

Jobs can be executed on a Scyld cluster using either "directed execution" with the **bpsh** command or "dynamic execution" with the **beorun** or **mpprun** commands.

Directed Execution with bpsh

In the directed execution mode, the user explicitly defines which node (or nodes) will run a particular job. This mode is invoked using the **bpsh** command, the ClusterWare shell command analogous in functionality to both the **rsh** (remote shell) and **ssh** (secure shell) commands. Following are some examples of using **bpsh**:

- This example runs **hostname** on the compute node and writes the output back to the user's screen from compute node 0:

  ```
  [user@cluster user]$ bpsh 0 /bin/hostname
    .0
  ```

- The following example runs the uptime utility on node 0, assuming it is installed in /usr/bin:

  ```
  [user@cluster user]$ bpsh 0 /usr/bin/uptime
    12:56:44 up  4:57,  5 users,  load average: 0.06, 0.09, 0.03
  ```

Dynamic Execution with beorun and mpprun

In the dynamic execution mode, Scyld decides which node is the most capable of executing the job at that moment in time. Scyld includes two parallel execution tools that dynamically select nodes, **beorun** and **mpprun**. They differ only in that **beorun** runs the job on the selected nodes concurrently, while **mpprun** runs the job sequentially on one node at a time.

The following example shows the difference in the amount of time the system uses to run a command with **beorun** vs. **mpprun**:

```
[user@cluster user]$ date;beorun -np 8 sleep 1;date
  Fri Aug 18 11:48:30 PDT 2006
  Fri Aug 18 11:48:31 PDT 2006

[user@cluster user]$  date;mpprun -np 8 sleep 1;date
  Fri Aug 18 11:48:46 PDT 2006
  Fri Aug 18 11:48:54 PDT 2006
```

Chapter 5. Troubleshooting ClusterWare

Failing PXE Network Boot

If a compute node fails to join the cluster when booted via PXE network boot, there are several places to look, as discussed below.

Rule out physical problems.

Check for disconnected Ethernet cables, malfunctioning network equipment, etc.

Check the system logs.

There are several log files:

- The master node's `/var/log/messages` file combines rsyslog output from the master node and each compute node. The master node's Scyld ClusterWare **beoserv** daemon serves as the cluster's DHCP server, and it logs the basic PXEboot interactions with each compute node. If a compute node shows no PXEboot logging, then the **beoserv** daemon is not seeing the initial PXEboot or DHCP request. Verify that the master node's private cluster network firewall is not blocking incoming requests.
- If the syslog shows a compute node is making repeated PXEboot responses without ever reaching *boot*, *error*, or *up* state, then the Scyld ClusterWare **beoclient** daemon on the compute node is unable to start up the node.

  Commonly, **beoclient** is failing to load the appropriate kernel binary module for the Ethernet interface. Ensure that `/etc/beowulf/config` specifies a *bootmodule* for the Ethernet controller hardware used by that specific compute node server, and that any *modarg* module options are valid for that particular kernel driver. Scyld ClusterWare distributes *bootmodule* entries for all Penguin Computing servers. If your compute node is not a Penguin Computing server, then verify that the necessary kernel driver is named as a *bootmodule*.

  Definitive diagnosis may require viewing the compute node's console output, either by attaching a graphical monitor to the console port, attaching a serial cable from the compute node's serial console output to another server and using **/usr/bin/minicom** to capture the output, or capturing the compute node's serial console output using the IPMI serial console functionality.
- If a compute node reaches *boot* state, then example the node's individual `/var/log/beowulf/node.N` log file, where *N* is the node number.

Check for the correct DHCP server.

If a node fails to appear initially (on power-up), or appears then subsequently disappears, then the node may be unable to find the master node's DHCP server. Another DHCP server may be answering and supplying IP addresses.

To check whether the master is seeing the compute node's DHCP requests, or whether another server is answering, use the Linux **tcpdump** utility. The following example shows a correct dialog between compute node 0 (10.10.100.100) and the master node.

```
[root@cluster ~]# tcpdump -i eth1 -c 10
Listening on eth1, link-type EN10MB (Ethernet),
  capture size 96 bytes
18:22:07.901571 IP master.bootpc > 255.255.255.255.bootps:
  BOOTP/DHCP, Request from .0, length: 548
18:22:07.902579 IP .-1.bootps > 255.255.255.255.bootpc:
  BOOTP/DHCP, Reply, length: 430
18:22:09.974536 IP master.bootpc > 255.255.255.255.bootps:
  BOOTP/DHCP, Request from .0, length: 548
18:22:09.974882 IP .-1.bootps > 255.255.255.255.bootpc:
  BOOTP/DHCP, Reply, length: 430
18:22:09.977268 arp who-has .-1 tell 10.10.100.100
18:22:09.977285 arp reply .-1 is-at 00:0c:29:3b:4e:50
18:22:09.977565 IP 10.10.100.100.2070 > .-1.tftp:  32 RRQ
  "bootimg::loader" octet tsize 0
18:22:09.978299 IP .-1.32772 > 10.10.100.100.2070:
  UDP, length 14
10 packets captured
32 packets received by filter
0 packets dropped by kernel
```

Check the network interface.

Verify that the master node's network interface is properly set up. Then check the network interface settings using **beonetconf**. Reconfigure as needed, and restart cluster services again.

Verify that ClusterWare services are running.

Check the status of ClusterWare services by entering the following command in a terminal window:

```
[root@cluster ~]# systemctl status clusterware
```

Restart ClusterWare services from the command line using:

```
[root@cluster ~]# systemctl restart clusterware
```

Check the switch configuration.

If the compute nodes fail to boot immediately on power-up but successfully boot later, the problem may lie with the configuration of a managed switch.

Some Ethernet switches delay forwarding packets for approximately one minute after link is established, attempting to verify that no network loop has been created ("spanning tree"). This delay is longer than the PXE boot timeout on some servers.

Disable the spanning tree check on the switch; the parameter is typically named "fast link enable". See the *Administrator's Guide* for more details.

Mixed Uni-Processor and SMP Cluster Nodes

The Scyld ClusterWare system architecture eliminates the problem of unintentionally running different versions of a program over the cluster's compute nodes.

The cluster nodes are required to run the same kernel version, typically with the same features and optimization enabled. Uni-processor machines can run the SMP kernel. The best choice for a mixed cluster is to run the SMP kernel. Beginning with CW4.1.1, support for uniprocessor kernels was dropped.

IP Forwarding

If IP forwarding is enabled in `/etc/beowulf/config` but is still not working, then check `/etc/sysctl.conf` to see if it is disabled.

Check for the line "net.ipv4.ip_forward = 1". If the value is set to 0 (zero) instead of 1, then IP forwarding will be disabled, even if it is enabled in `/etc/beowulf/config`.

SSH Traffic

The Red Hat installer configures the firewall with most services disabled. If SSH traffic isn't passing through the firewall, then check your firewall settings to make sure SSH is selected as a trusted service.

To do this, log in as a root user and choose the Red Hat **Applications -> System Settings -> Security Level** menu option to open the Security Level Configuration window. Then make sure that SSH is checked in the list of trusted services.

Device Driver Updates

Scyld ClusterWare releases are tested on many different machine configurations, but it is impossible to provide device drivers for hardware unknown at release time.

Most problems with unsupported hardware or device-specific problems are resolved by updating to a newer device driver. Some devices may not yet be supported under Linux. Check with your hardware vendor.

The Scyld ClusterWare architecture makes most driver updates simple. Drivers are installed and updated on the master node exactly as with a single machine installation. The new drivers are immediately available to compute nodes, although already-loaded drivers are not replaced.

There are two irregular device driver types that require special actions: disk drivers and network drivers, both of which apply to the compute nodes. In both cases, the drivers must be available to load additional drivers and programs, and are thus packaged in initial RAM disk images.

Another irregular instance is where drivers must execute scripts when they load; one example is Infiniband. Contact the hardware vendor or Scyld support if you have difficulty with the script that loads the driver.

Finding Further Information

If you encounter a problem installing your Scyld cluster and find that this *Installation Guide* cannot help you, the following are sources for more information:

- See the *Release Notes* for special installation or upgrade procedures that must be taken for your particular version of ClusterWare. It is available on the master node or on the documentation CD included in the Scyld installation kit.
- See the *Administrator's Guide*, which includes descriptions of more advanced administration and setup options. It is available on the master node or on the documentation CD included in the Scyld installation kit.
- See the *Reference Guide*, a complete technical reference to Scyld ClusterWare. It is available on the master node or on the documentation CD included in the Scyld installation kit.

For the most up-to-date product documentation and other helpful information about Scyld ClusterWare, visit the Scyld Customer Support website at http://www.penguincomputing.com/support. and online documentation at http://www.penguincomputing.com/support/documentation.

Notes

1. http://www.penguincomputing.com/support
2. http://www.penguincomputing.com/support/documentation

Appendix A. Compute Node Disk Partitioning

Architectural Overview

The Scyld ClusterWare system uses a "disk-less administration" model for compute nodes. This means that the compute nodes boot and operate without the need for mounting any file system, either on a local disk or a network file system. By using this approach, the cluster system does not depend on the storage details or potential misconfiguration of the compute nodes, instead putting all configuration information and initialization control on the master.

This does not mean that the cluster cannot or does not use local disk storage or network file systems. Instead it allows the storage to be tailored to the needs of the application rather than the underlying cluster system.

The first operational issue after installing a cluster is initializing and using compute node storage. While the concept and process is similar to configuring the master machine, the "disk-less administration" model makes it much easier to change the storage layout on the compute nodes.

Operational Overview

Compute node hard disks are used for three primary purposes:

- *Swap Space* — Expands the Virtual Memory of the local machine.
- *Application File Storage* — Provides scratch space and persistent storage for application output.
- *System Caching* — Increases the size and count of executables and libraries cached by the local node.

In addition, a local disk may be used to hold a cluster file system (used when the node acts as a file server to other nodes). To make this possible, Scyld provides programs to create disk partitions, a system to automatically create and check file systems on those partitions, and a mechanism to mount file systems.

Disk Partitioning Procedures

Deciding on a partitioning schema for the compute node disks is no easier than with the master node, but it can be changed more easily.

Compute node hard disks may be remotely partitioned from the master using **beofdisk**. This command automates the partitioning process, allowing all compute node disks with a matching hard drive geometry (cylinders, heads, sectors) to be partitioned simultaneously.

If the compute node hard disks have not been previously partitioned, you can use **beofdisk** to generate default partition tables for the compute node hard disks. The default partition table allocates three partitions, as follows:

- A BeoBoot partition equal to 2 MB (currently unused)
- A swap partition equal to 2 times the node’s physical memory
- A single root partition equal to the remainder of the disk

The partition table for each disk geometry is stored in the directory `/etc/beowulf/fdisk` on the master node, with the filename specified in nomenclature that reflects the disk type, position, and geometry. Example filenames are `hda:2495:255:63`, `hdb:3322:255:63`, and `sda:2495:255:63`.

The **beofdisk** command may also be used to read an existing partition table on a compute node hard disk, as long as that disk is properly positioned in the cluster. The command captures the partition table of the first hard disk of its type and geometry (cylinder, heads, sectors) in each position on a compute node's controller (e.g., sda or hdb). The script sequentially queries the compute nodes numbered 0 through *N* - 1, where *N* is the number of nodes currently in the cluster.

Typical Partitioning

While it is not possible to predict every configuration that might be desired, the typical procedure to partition node disks is as follows:

1. From the master node, capture partition tables for the compute nodes:

   ```
   [root@cluster ~]# beofdisk -q
   ```

   With the *-q* parameter, **beofdisk** queries all compute nodes. For the first drive found with a specific geometry (cylinders, heads, sectors), it reads the partition table and records it in a file. If the compute node disk has no partition table, this command creates a default partition set and reports the activity to the console.

   If the partition table on the disk is empty or invalid, it is captured and recorded as described, but no default partition set is created. You must create a default partition using the **beofdisk -d** command; see the Section called *Default Partitioning*.

2. Based on the specific geometry of each drive, write the appropriate partition table to each drive of each compute node:

   ```
   [root@cluster ~]# beofdisk -w
   ```

   This technique is useful, for example, when you boot a single compute node with a local hard disk that is already partitioned, and you want the same partitioning applied to all compute nodes. You would boot the prototypical compute node, capture its partition table, boot the remaining compute nodes, and write that prototypical partition table to all nodes.

3. Reboot all compute nodes to make the partitioning effective.
4. If needed, update the file `/etc/beowulf/fstab` on the master node to record the mapping of the partitions on the compute node disks to the file systems.

Default Partitioning

To apply the recommended default partitioning to each disk of each compute node, follow these steps:

1. Generate default partition maps to `/etc/beowulf/fdisk`:

   ```
   [root@cluster ~]# beofdisk -d
   ```

2. Write the partition maps out to the nodes:

   ```
   [root@cluster ~]# beofdisk -w
   ```

3. You must reboot the compute nodes before the new partitions are usable.

Generalized, User-Specified Partitions

To create a unique partition table for each disk type/position/geometry triplet, follow these steps:

1. Remotely run the **fdisk** command on each compute node where the disk resides:

   ```
   [root@cluster ~]# bpsh n fdisk device
   ```

   where *n* is the node number or the first compute node with the drive geometry you want to partition, and *device* is the device you wish to partition (e.g., `/dev/sda`, `/dev/hdb`).

2. Once you have created the partition table and written it to the disk using **fdisk**, capture it and write it to all disks with the same geometry using:

   ```
   [root@cluster ~]# beofdisk -w
   ```

3. You must reboot the compute nodes before the new partitioning will be effective.

4. You must then map file systems to partitions as described later in this chapter.

Unique Partitions

To generate a unique partition for a particular disk, follow these steps:

1. Partition your disks using either default partitioning or generalized partitions as described above.

2. From the master node, remotely run the **fdisk** command on the appropriate compute node to re-create a unique partition table using:

   ```
   [root@cluster ~]# bpsh n fdisk device
   ```

   where *n* is the compute node number for which you wish to create a unique partition table and *device* is the device you wish to partition (e.g., `/dev/sda`).

3. You must then map file systems to partitions as described below.

Mapping Compute Node Partitions

If your compute node hard disks are already partitioned, edit the file `/etc/beowulf/fstab` on the master node to record the mapping of the partitions on your compute node disks to your file systems. This file contains example lines (commented out) showing the mapping of file systems to drives; read the comments in the `fstab` file for guidance.

1. Query the disks on the compute nodes to determine how they are partitioned:

   ```
   [root@cluster ~]# beofdisk -q
   ```

   This creates a partition file in `/etc/beowulf/fdisk`, with a name similar to `sda:512:128:32` and containing lines similar to the following:

   ```
   [root@cluster root]# cat sda:512:128:32
   /dev/sda1  :  start=    32,  size=  8160,   id=89,  bootable
   /dev/sda2  :  start=    8192,   size=    1048576,  Id=82
   /dev/sda3  :  start=    1056768,    size=    1040384,  Id=83
   /dev/sda4  :  start=    0, size=  0,  Id=0
   ```

2. Read the comments in `/etc/beowulf/fstab`. Add the lines to the file to use the devices named in the `sda` file:

   ```
   # This is the default setup from beofdisk
   #/dev/hda2        swap     swap   defaults     0 0
   #/dev/hda3        /        ext2   defaults     0 0
   /dev/sda1         /boot    ext23  defaults     0 0
   /dev/sda2         swap     swap   defaults     0 0
   ```

```
/dev/sda3         /scratch ext3   defaults     0 0
```

3. After saving `fstab`, you must reboot the compute nodes for the changes to take affect.

Appendix B. Changes to Configuration Files

Changes to Red Hat Configuration Files

An installation of Red Hat sets a default configuration optimized for a stand-alone server. Installing ClusterWare on a Red Hat installation changes some of these default configuration parameters to better support a cluster. The following sections describe the changes the ClusterWare installation automatically makes to the Red Hat configuration. Any of these may be reversed; however, reversing them may adversely affect the operation of the ClusterWare cluster.

1. `/etc/grub.conf` has been modified.

   After ClusterWare has been installed, the default boot becomes the newest ClusterWare kernel.

2. NFS Services default configuration has been modified.

   By default, Red Hat configures NFS to "off" for security reasons. However, most cluster applications require that at least the home directory of the master node be accessible to the compute nodes. NFS services on the master are set with the default to "on" for run levels 3, 4, and 5.

   The default out-of-box chkconfig for NFS on RHEL7 is as follows:

   ```
   [root@scyld ~]# chkconfig --list nfs
   nfs             0:off   1:off   2:off   3:off   4:off   5:off   6:off
   ```

   ClusterWare has changed the default to the following:

   ```
   [root@scyld ~]# chkconfig --list nfs
   nfs             0:off   1:off   2:off   3:on    4:on    5:on    6:off
   ```

   To get NFS to mount directories from the master to the compute nodes, the file `/etc/exports` needs one entry per line for each file system to export from the master to the compute nodes (the RHEL7 default is a blank/non-existent file). ClusterWare creates this file if it didn't already exist, and adds several new entries of the form:

   ExportedDirectoryPathname `@cluster`(*accessMode*,*syncMode*,no_root_squash)

   The export for `/home` from the master is configured with an *accessMode* of `rw` (read-write) and a *syncMode* of `sync` by default for data reliability reasons, and the non-/home directories are exported `ro` (read-only) for security reasons and `async` for performance reasons.

   See the ClusterWare *Release Notes* for details about which directories are added by Scyld.

3. `/etc/sysconfig/syslog` has been modified.

   Compute nodes will forward messages to the master node's syslogd daemon, which places them in `/var/log/messages`. In order for this to function correctly, ClusterWare modifies the `/etc/sysconfig/syslog` file by adding the `-r` option to the `SYSLOGD_OPTIONS` line:

   ```
   SYSLOGD_OPTIONS="-m 0 -r"
   ```

Possible Changes to ClusterWare Configuration Files

A clean install of ClusterWare introduces various ClusterWare configuration files that include default settings that a local sysadmin may choose to modify. A subsequent upgrade from one ClusterWare release to a newer release will avoid replacing these potentially modified files. Instead, an update installs a new version of the default file as a file of the form `CWconfigFile.rpmnew`. Therefore, after a ClusterWare upgrade, the sysadmin is encouraged to compare each such existing `CWconfigFile` with the new default version to ascertain which of the new default entries are appropriate to manually merge into the preexisting `CWconfigFile` file.

1. `/etc/beowulf/config` and `config.rpmnew`

   ClusterWare specifies additional libraries for compute nodes that may help various applications and scripts execute out-of-the-box

2. `/etc/beowulf/fstab` and `fstab.rpmnew`

   ClusterWare specifies additional `/dev` devices and NFS-mounted directories for compute nodes that may help various applications and scripts execute out-of-the-box