XC™ Series Software Initial Installation and Configuration Guide (CLE 6.0.UP01) S-2559 Rev A

Contents

1 About XC™ Series Software Initial Installation and Configuration Guide (CLE 6.0.UP01) S-2559 Rev A

The *XC™ Series Software Initial Installation and Configuration Guide* (S-2559) provides information and instructions to perform an initial installation of System Management Workstation (SMW) and Cray Linux Environment (CLE) software. The software consists of the SMW base operating system, which is SUSE® Linux Enterprise Server version 12 (SLES® 12), the Cray SMW software, and the CLE software.

Release CLE 6.0.UP01 / SMW 8.0.UP01

XC™ Series Software Initial Installation and Configuration Guide (S-2559) supports Cray software release CLE 6.0.UP01 / SMW 8.0.UP01 for Cray XC™ Series systems, released on 20 June 2016. It combines initial installation and configuration procedures for both SMW and CLE, and it supersedes *System Management Workstation (SMW) Software Installation Guide* (S-2480) and *CLE Installation and Configuration Guide* (S-2444).

This publication was republished as *XC™ Series Software Initial Installation and Configuration Guide (CLE 6.0.UP01) S-2559 Rev A* on 26 August 2016 to include a procedure for changing the time zone on cabinet and blade controllers.

Basic procedure to change the time zone on cabinet and blade controllers:

1. Set the SMW time or time zone using existing yast2 tools.
2. Run the following commands.

```
smw# cp /etc/localtime /opt/tftpboot/localtime
smw# xtccreboot -c all
smw# xtalive -a l1sysd -l l1 s0 # Repeat until all are alive
smw# xtccreboot -b all
smw# xtalive s0 # Repeat until all are alive.
```

That procedure has been incorporated in this publication as follows:

- For a fresh install, the procedure is distributed among these topics.
  - *Install the SLES12 Base Linux Distribution on the SMW* on page 41: Sets the SMW time zone (the `yast2` command in step 7).
  - *Prepare and Update the Global Config Set* on page 89: Puts the SMW time zone setting in a directory accessible by the cabinet and blade controllers (the `cp` command in step 18).
  - *Bootstrap Hardware Discovery* on page 95: Reboots the cabinet and blade controllers to apply the changed SMW time zone setting (the `xtccreboot` and `xtalive` commands in steps 6 and 7).
- For customization of a preinstalled system, the procedure is contained in this topic.
  - *Change the Time Zone* on page 211: Has the entire procedure in one place (in steps 3, 6, 7c, and 7d), as well as some additional verification steps.

Audience and Scope

XC™ Series Software Initial Installation and Configuration Guide is intended for system installers, administrators, and anyone who installs and configures software on a Cray XC™ Series system. It assumes some familiarity with standard Linux and open source tools (e.g., zypper/yum for RPMs, Ansible, YAML/JSON configuration data).

Command Prompt Conventions

hostname in command prompts

Hostnames in command prompts indicate where the command must be run.

`hostname#`	Run the command on this hostname.
`smw#`	Run the command on the SMW.
`boot#`	Run the command on the boot node.
`sdb#`	Run the command on the SDB node.
`login#`	Run the command on any login node.
`smw1#` `smw2#`	For a system configured with the SMW failover feature there are two SMWs—one in an active role and the other in a passive role. The SMW that is active at the start of a procedure is *smw1*. The SMW that is passive is *smw2*.
`smwactive#` `smwpassive#`	In some scenarios, the active SMW is *smw1* at the start of a procedure—then the procedure requires a failover to the other SMW. In this case, the documentation will continue to refer to the formerly active SMW as *smw1*, even though *smw2* is now the active SMW. If further clarification is needed in a procedure, the active SMW will be called *smwactive* and the passive SMW will be called *smwpassive*.

account name in command prompts

The account that must run the command is also indicated in the prompt.

`smw#`
`boot#`
`sdb#`
`login#`
`hostname#`
	The `root` or super-user account always has the `#` character at the end of the prompt.

`crayadm@smw>` `crayadm@login>`	Any non-`root` account is indicated with *`account@hostname`*.
`user@hostname>`	A user account that is neither `root` nor `crayadm` is referred to as *user*.

command prompt inside chroot

If the `chroot` command is used, the prompt changes to indicate that it is inside a chroot'd environment on the hostname.

```
smw# chroot /path/to/chroot
chroot-smw#
```

directory path in command prompt

Sometimes the current path can be so long that including it in the prompt does not add clarity to the command example. Most of the time, the command can be executed from any directory. When it matters which directory the command is invoked within, the `cd` command is used to change into the directory, and the directory is referenced with a period (`.`) to indicate the current directory.

For example, here are actual prompts as they appear on the system:

```
smw:~ # cd /etc
smw:/etc# cd /var/tmp
smw:/var/tmp# ls ./file
smw:/var/tmp# su - crayadm
crayadm@smw:~> cd /usr/bin
crayadm@smw:/usr/bin> ./command
```

And here are the same prompts as they would appear in examples in this publication:

```
smw# cd /etc
smw# cd /var/tmp
smw# ls ./file
smw# su - crayadm
crayadm@smw> cd /usr/bin
crayadm@smw> ./command
```

Typographic Conventions

`Monospace`	Indicates program code, reserved words, library functions, command-line prompts, screen output, file/path names, and other software constructs.
`Monospaced Bold`	Indicates commands that must be entered on a command line or in response to an interactive prompt.
`Oblique` or *`Italics`*	Indicates user-supplied values in commands or syntax definitions.
Proportional Bold	Indicates a graphical user interface window or element and key strokes (e.g., **Enter**, **Alt-Ctrl-F**).
\ (backslash)	At the end of a command line, indicates the Linux® shell line continuation character (lines joined by a backslash are parsed as a single line). Do not type anything after the backslash or the continuation feature will not work correctly.

Feedback

Your feedback is important to us. Visit the Cray Publications Portal at *http://pubs.cray.com* and make comments online using the **Contact Us** button in the upper-right corner, or email comments to pubs@cray.com.

Trademarks

1.1 Related Publications

This publication supersedes *System Management Workstation (SMW) Software Installation Guide* (S-2480) and *CLE Installation and Configuration Guide* (S-2444).

Although this publication is all that is necessary for installing SMW and CLE software, the following publications contain additional information that may be helpful. The release errata and readme files are available on CrayPort, and the rest of these publications (and other Cray publications) can be found at *http://pubs.cray.com*.

- *SMW Release Errata* (includes notice of any day-1 patches) and the *SMW README*, which are provided with the SMW release software
- *CLE Release Errata* and the *CLE README*, which are provided with the CLE release software
- *XC™ Series Configurator User Guide*
- *XC™ Series System Administration Guide*
- *XC™ Series Lustre® Administration Guide*
- *XC™ Series Power Management Administration Guide*
- *XC™ Series System Environment Data Collections (SEDC) Guide*
- (*XC™ Series DataWarp™ Installation and Administration Guide*), which supersedes *DataWarp Installation Guide* (S-2547)
- *Cray Compiling Environment Release Overview and Installation Guide*
- *Cray eLogin Installation Guide*
- *Configure Cray SEC Software* (Simple Event Correlator)
- *XC™ Series Aries™ Network Resiliency Guide*
- *XC™ Series DVS Administration Guide*
- For a system that will be configured for SMW high availability (HA):
  - *XC™ Series SMW HA Installation Guide*
  - *XC™ Series SMW HA Administration Guide*

1.2 Distribution Media

The Cray CLE 6.0.UP01 / SMW 8.0.UP01 release distribution media consists of one DVD and several other pieces of media which may be on DVDs or as ISO files.

Configuration worksheets for CLE config sets and the global config set are also included in this distribution, so that sites can begin entering site-specific configuration data in them before and during the installation process.

For an initial installation of the SMW/CLE software, first install the DVD labeled Cray-SMWbase12-201511021655, which contains the SMW base operating system, SUSE Linux Enterprise Server version 12 (SLES 12). This table shows all installation media included with this release.

bootable SMW SLES12 media	• `Cray-SMWbase12-201511021655.iso` DVD
SMW release	• `smw-8.0.96-201605131428.iso`
CLE release	• `cle-6.0.96-201605131428.iso`
SLES release	• `SLE-12-Server-DVD-x86_64-GM-DVD1.iso` • `SLE-12-SDK-DVD-x86_64-GM-DVD1.iso` • `SLE-12-WE-DVD-x86_64-GM-DVD1.iso` • `SLE-12-Module-Legacy-DVD-x86_64-GMC-CD1.iso` • `SLE-12-Module-Public-Cloud-DVD-x86_64-GMC2-DVD1.iso`
SLE update	• `sleupdate-sle12sp0-201601271436.iso`
CentOS software	• `CentOS-6.5-x86_64-bin-DVD1.iso`
CLE and global configuration worksheets	• `cle-0513-worksheets.tar` • `global-0513-worksheets.tar`
SLE HA software	• `SLE-12-HA-DVD-x86_64-GM-CD1.iso`
SMW HA release	• `smwha-sleha12sp0-12.0.48-201605180227.iso`

2 Introduction to Initial Installation and Configuration of Cray XC Software

This guide provides information and instructions to perform an initial installation of System Management Workstation (SMW) and Cray Linux Environment (CLE) software release packages on a Cray XC Series system. It also provides information about how to customize a preinstalled system.

SMW HA only: For a system with two SMWs that will be configured for SMW high availability (HA), the process to install the first SMW is the same as for a system with a stand-alone SMW, with a few minor differences that are noted in this guide. However, installing the second SMW uses a completely different process. Do not use this guide for the second SMW. For more information, see *SMW HA Installation Guide for XC Series Systems*.

With this release, Cray has changed the way software is installed, configured, and managed on XC Series systems. The changes that most affect installation and configuration are summarized here.

The new Cray Management System (CMS)

- uses a common installation process for SMW and CLE (which is why there is now a single installation guide for XC systems—this one—instead of separate guides for SMW and CLE)
- leverages standard Linux and common open source tools (e.g., zypper/yum for RPMs, Ansible, YAML/JSON configuration data)
- keeps software images and configuration separate until boot
  - prescriptive image creation using recipes
  - centralized configuration
  - configuration applied at boot time or after configuration adjusted

The core elements of this new management system are:

IMPS Image Management and Provisioning System (IMPS) is responsible for creating and distributing repository content and for prescriptive image creation. Note that although filepaths for configuration data and tools include `imps`, this is an artifact of an early implementation that grouped both image and configuration management under IMPS. IMPS is now image management only.

CMF Configuration Management Framework (CMF) comprises the configuration data (stored in config sets on the SMW), tools to manage and distribute that data (e.g., the configurator and the IMPS Distribution System (IDS)), and software to apply the configuration data to the running image (Ansible plays).

NIMS Node Image Mapping Service (NIMS) is responsible for keeping track of which images get booted on which nodes, what additional kernel parameters to pass to nodes at boot time, and which load file to use within a boot image.

What else is new?

- New base operating system for the SMW/CLE: SUSE® Linux Enterprise Server version 12 (SLES® 12) for x86_64
- New base operating system for HSS (Hardware Supervisory System) controllers: OpenSUSE 13.2 for 32 bit

- New modular installer

Cray Scalable Services. Cray Scalable Services is an essential part of CMS that is used to both distribute and aggregate information. Within Cray Scalable Services, nodes are designated as SoA (server of authority), tier1, tier2, or tier3. A node can be a member of only one of these groups. Tier1 node are clients of the SoA and servers for tier2 nodes. Tier2 nodes are clients of tier1 nodes and servers for tier3 nodes. Tier3 nodes are clients of tier2 nodes. The SMW is the designated SoA in Cray XC systems. The boot and SDB nodes are designated tier1 nodes, and they must have direct network connectivity to the SMW via Ethernet. Typically, tier2 nodes are service nodes or repurposed compute nodes that have no other duties beyond being part of the Scalable Services. Most nodes are tier3 nodes.

Distributed from SMW to rest of system	• config set data, which is shared using a 9P file system and DIOD (distributed I/O daemon) • zypper and yum software repositories on SMW, which can by used from any node with the Live Update feature
Distributed from boot node to rest of system	• PE (Programming Environment) • diags (online diagnostics) • Netroot
Aggregated to SMW from rest of system	• Lightweight Logging Manager (LLM) logging

Much of the software remains unchanged, for the most part, such as Application-level Placement Scheduler (ALPS), Node Health Checker (NHC), Resource Utilization Reporting (RUR), and so forth.

This guide includes instructions for installing the following software: the SMW base operating system, the Cray SMW software, CLE software, SLES security updates, Programming Environment (PE), and direct-attached Lustre (DAL), if needed. The following visual guide provides an overview of the process.

Figure 1. Visual Guide to Initial Installation and Configuration of Cray XC Series Software

2.1 About Config Set Caching

Config sets are defined and reside on the Server of Authority, which on XC systems is the SMW. Config set content is made available to all nodes in the system by means of Cray Scalable Services (described in the introduction to this publication).

To make that export both quick and reliable, the cray-cfgset-cache service was created. It caches config sets locally on nodes (compressed for a smaller footprint). On the SMW, it does the following:

- notices changes to config sets on the SMW
- refreshes the local caches dynamically
- detects failures and retries automatically

The cray-cfgset-cache service ensures that config set content gets refreshed on all nodes whenever config sets are created or updated on the SMW. It is triggered when `cray-ansible` is run on a node with the `start`, `restart`, or `link` commands.

ATTENTION: If the cray-cfgset-cache service is stopped, config set content in node-local memory will not get refreshed when `cray-ansible` is run. If that happens, nodes will continue to use the most recent compressed copy of the config set data created before the service was stopped.

What Gets Cached

The cray-cfgset-cache service does not copy an entire config set to node-local memory. Instead, it uses the config set on the SMW to create these two files in the root of the config set:

- a compressed copy of the config set using SquashFS tools, (typically < 3 MB)
- a checksum of the compressed copy of the config set

The compressed copy is made available (effectively copied) to node-local RAM, and the checksum is used to know when the config set in node-local memory no longer matches the config set on the SMW. Even though Scalable Services makes the entire config set directory structure on the SMW available to the rest of the system, only the compressed copy and its associated checksum are used by nodes. They are the key to the performance, scalability, and reliability improvements provided by config set caching.

When `cray-ansible` is run on a node, the node will do the following:

1. Check to see if the cached node-local version of the compressed config set is out of date.
2. If it is stale, replace it with a newer version available on the SMW and start using that newer version.

Add Kernel Watch Descriptors to Improve Config Set Caching Performance

The cray-cfgset-caching service operates on kernel watch descriptors to automatically generate the config set compressed copy and checksum files. The performance of this service depends on the number of config sets created and the number of directories within each config set. Service startup performance in particular is affected by large numbers of cfgsets and the availability of kernel watch descriptors. Additional watch descriptors may be required to provide coverage for large numbers of config sets.

To determine how many watch descriptors are in use:

```
smw# cd /var/opt/cray/imps/config/sets
smw# find . -type d | wc
```

To increase the total number of available watch descriptors:

```
smw# sysctl fs.inotify.max_user_watches=524288
```

3 Prepare for a Fresh Install

WARNING: When a fresh install is performed on a system, disks are wiped clean. To prevent loss of necessary data, before beginning any installation procedures, consider what configuration files, log files, or other files should be preserved, and save them in a location unaffected by the installation.

In preparation for a fresh install, do the following:

- Back up site files (configuration files, log files, etc.) to prevent loss of necessary data (see *Back Up Site Data* on page 232).
- Extract the configuration worksheets in preparation for entering site data. They are provided in the CLE 6.0.UP01 release directory available on CrayPort, `/cray/css/release/package/release/CLE/6.0.UP01`.
  - `cle-0513-worksheets.tar`
  - `global-0513-worksheets.tar`
- Read the *SMW Release Errata* and the *SMW README* provided with the SMW release package for any additional installation-related requirements, corrections to this installation guide, and other relevant information about the release package.
- Read the *CLE Release Errata* and the *CLE README* provided with the CLE release package for any additional installation-related requirements, corrections to this installation guide, and other relevant information about the release package.
- Read the Field Notices (FN) related to kernel security fixes to identify any changes to this release package. Apply any needed changes before installing the new software.
- Collect information about the XC system: SMW, system hardware, and service node roles and networks (see *Information to Collect Before Installation* on page 15).
- Verify that the network connections are in place (see *Network Connections* on page 17).
- Know which configuration values are site-specific and which are defaults (see *Configuration Values* on page 19).
- Be familiar with the default passwords used during the installation process (see *Passwords* on page 20).

3.1 Information to Collect Before Installation

SMW Information

This information will be needed to update the global config set during configuration.

- Network base IP address for SMW eth0
- Netmask for SMW eth0

- Gateway IP address for SMW eth0
- List of IP addresses to use as DNS server
- List of domains to use in the DNS search path for hosts attached to SMW eth0 network
- List of NTP servers
- Hostname of the SMW: both the short name and the fully qualified domain name (FQDN)
- IP address of SMW eth0

Hardware Information

When `xtdiscover` is used to discover XC system hardware, it will prompt for this information.

- Maximum cabinet size in the X dimension
- Maximum cabinet size in the Y dimension
- Network topology class (0 or 2 for Cray XC30 systems, 0 for Cray XC30-AC systems)
- Primary boot node (and alternate boot node if enabling boot node failover)
- Primary SDB node (and alternate SDB node if enabling SDB node failover)

Service Node Roles

The XC system being installed and configured must have service nodes designated to function in some or all of the following roles. A node may have more than one role (e.g., boot and tier1). The system at this site may not require all of these roles.

boot
SDB
login
tier1 (boot node and SDB node)
tier2 (see notes*)
LNet router to external Lustre server
realm-specific IP (RSIP)
DataWarp-managed nodes with SSD hardware
DataWarp API gateway nodes
nodes providing a role for workload management (WLM)
DVS-to-external-filesystem
Direct-attached Lustre (DAL) MGS, MDS, or OSS
compute node repurposed to be a service node

Tier2 Node FAQ

Q. How many tier2 nodes are needed?	**A.** At least one server must be provided, and for resiliency, two nodes placed on different blades. The recommended ratio of clients to tier2 nodes (servers) is 400 to 1.
Q. Will adding more tier2 nodes help performance?	**A.** More tier2 nodes does not always yield additional performance and are subject to diminishing returns.

Q. What kind of node can be used as tier2?	**A.** Use these: ● OPTIMAL: dedicated repurposed compute nodes (RCN) ● dedicated service nodes ● nodes with uniform light to moderate load ● nodes with relatively homogeneous single core speeds to reduce resource contention disparity during periods of partial availability NEVER use these: ● nodes with slower individual CPU cores, such as KNL (Knights Landing) compute nodes as RCN ● direct-attached Lustre (DAL) servers ● RSIP (realm-specific IP) servers ● login nodes
Q. Can a tier2 node have more than one role?	**A.** Small test deployment systems (TDS) may use tier2 nodes that have additional roles, but generally, it is better for tier2 nodes to be dedicated.
Q. Where should tier2 nodes be placed?	**A.** Distribute nodes throughout the system (on different blades) for resiliency in the event of hardware failure.

Service Node Network Information

For each service node with a network interface, either Ethernet or InfiniBand, collect this information.

For each network defined:	● unique identifier for the network (management, login, lnet) ● description or notes about the network (e.g. "Network to external Lustre") ● network base IP address ● netmask ● gateway IP address
For each network interface added to a host	● unique identifier for each interface (primary_ethernet, eth0, eth1, eth2, eth3, ib0, ib1, etc.) on this host ● device name for the interface (eth0, ib1, etc.) ● description or notes about the nterface (e.g., "Ethernet connecting boot node to SMW") ● any hostname aliases by which this node should be known ● name of the network to which this interface belongs (see list of networks defined above) ● IPv4 network address for the interface

3.2 Network Connections

The following network connections are required.

- A stand-alone SMW with a single quad-ethernet card has these private network connections:

  `eth0` - To the customer network
  `eth1` - To the hardware supervisory system (HSS) network
  `eth2` - Used for SMW HA (failover) heartbeat 1 network
  `eth3` - To the boot node

- An SMW configured for SMW failover (SMW HA) has a second quad-ethernet card with these connections:

  `eth4` - Used for SMW HA heartbeat 2 network
  `eth5` - Used for SMW HA distributed replicated block device (DRBD)
  `eth6` - Reserved for future use
  `eth7` - Reserved for future use

Things to note about network connections:

- Ethernet port assignments are valid only after the SMW software installation completes.
- The SMW must have a Fibre Channel or serial attached SCSI (SAS) connection to the boot RAID.
- A boot node must have a Fibre Channel or SAS connection to the boot RAID. If boot node failover is enabled or there are multiple logical CLE partitions, then each boot node should have such a connection to the boot RAID.
- A service database (SDB) node must have a Fibre Channel or SAS connection to the boot RAID. If SDB node failover is enabled or there are multiple logical CLE partitions, then each SDB node should have such a connection to the boot RAID..

  IMPORTANT: The SMW must be disconnected from the boot RAID before the initial installation of the SLES software.

  IMPORTANT: Ensure that the Fibre Channel optic cable connectors or SAS cable connectors have protective covers when disconnected from the SMW, boot node, SDB node, or boot RAID.

3.3 SMW Internal Disk Requirements

Internal SMW disks are used for the boot disk (with optional RAID1 mirroring between two boot drives) and the power management disk (PMDISK).

The PMDISK requires a minimum of 500 GB. This may be a fresh disk or a repurposed disk on an existing SMW. The PMDISK will be allocated to `/var/lib/pgsql` in an ext4 file system.

The boot disk (or pair of boot disks in RAID1 configuration) requires a minimum of 160 GB, but may be larger. If a RAID1 mirror is enabled, the drives in the RAID1 configuration must be the same size. The boot disk has 4 GB allocated to `/boot` in an ext3 file system, 32 GB for swap, and the rest of the disk for the `/` file system in a btrfs file system.

Table 1. SMW Internal Disk Requirements

Mount Point	FS Type	Disk	Size	Description
`/boot`	ext3	boot	4 GB	Booting information
`swap`	swap	boot	32 GB	SMW swap

Mount Point	FS Type	Disk	Size	Description
/	btrfs	boot	200+ GB	root file system of SMW with btrfs subvolumes
/var/lib/pgsql	ext4	power management	1000+ GB	Power Management disk

3.4 Configuration Values

The following IP addresses are set by default and are not site dependent.

Table 2. Default IP Addresses

IP Address	Description
10.1.0.1	Primary boot RAID controller
10.1.0.2	Secondary boot RAID controller
10.1.0.15	Storage RAID controller
10.1.1.1	SMW, `eth1` - HSS network
10.2.1.1	(SMW HA only) SMW, `eth2` - SMW HA heartbeat 1
10.3.1.1	SMW, `eth3` - admin network
10.3.1.254	boot node
10.4.1.1	(SMW HA only) SMW, `eth4` - SMW HA heartbeat 2
10.5.1.1	(SMW HA only) SMW, eth5 - SMW HA DRBD
127.0.0.1	localhost (loopback)

The following configuration values are site dependent. Record the actual values for the installation site in the third column.

Table 3. Site-dependent Configuration Values

Description	Example Value	Actual Value
SMW hostname	smw	
Domain	cray.com	
Aliases	cray-smw smw1	
Customer network IP address	192.168.78.68	
Customer network netmask	255.255.255.0	
Default gateway	192.168.78.1	
Domain names to search	us.cray.com mw.cray.com	

Description	Example Value	Actual Value
Nameserver IP address	10.0.73.30 10.0.17.16	
iDRAC hostname	cray-drac	
iDRAC IP address	192.168.78.69	
iDRAC Subnet Mask	255.255.255.0	
iDRAC Default GW	192.168.78.1	
Timezone	US/Central	
NTP servers	ntphost1 ntphost2	
X dimension	1-64	
Y dimension	1-32	
Topology Class	0, 2 (see note below)	

NOTE: Regardless of the number of cabinets in the system, Cray XC Series air-cooled systems must be set to topology class 0. Cray XC Series liquid-cooled systems can be topology class 0 or 2.

3.5 Passwords

The following default account names and passwords are used throughout the initial software installation process. Cray recommends changing these default passwords during the installation and configuration process at appropriate times before the SMW or network CLE nodes are connected to networks that are external to the XC system.

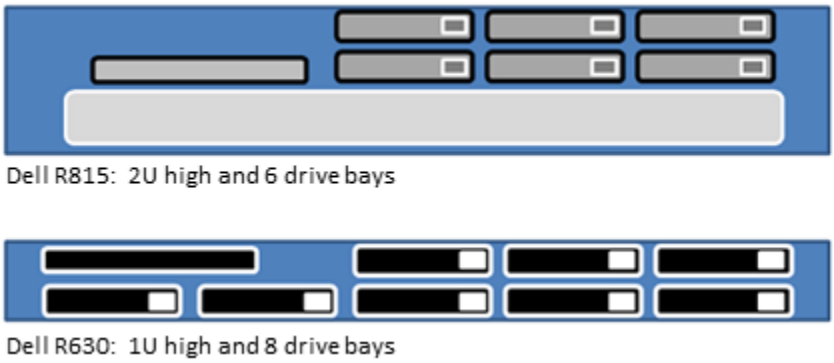
Table 4. Default System Passwords

Account Name	Password
`root`	`initial0`
`crayadm`	`crayadm`
`mysql`	None; a password must be created
`root` (iDRAC)	`initial0`

4 Install the Base Operating System on the SMW

The base operating system must be installed on the SMW before the Cray SMW and CLE software release packages can be installed. Cray provides two rack-mount SMW models: the Dell PowerEdge™ R815 Rack Server and the Dell PowerEdge™ R630 Rack Server. Earlier deskside SMW hardware is not supported. The figure below shows an easy way to distinguish between the two rack-mount models when viewing them from the front.

Figure 2. Distinguishing Features of Dell R815 and R630 Servers



Continue the installation process with *Prepare to Install the Base Linux Distribution* on page 21.

4.1 Prepare to Install the Base Linux Distribution

About this task

A full initial installation begins with installing the base operating system. This procedure provides initial steps that are common to installing the base OS both Dell R815 and R630 SMW models.

Procedure

1. Disconnect the SMW connection to the boot RAID.

   Disconnect the data cables and place protective covers on the fibre optic cable connectors (if present).

2. Connect the SMW keyboard, monitor, and mouse.

   Connect a keyboard, monitor, and mouse to the USB and monitor connectors on the SMW, if not already connected.

NOTE: Once the iDRAC has been configured, the keyboard, monitor, and mouse can be connected to the iDRAC for remote console activities instead of being directly connected to the SMW console.

3. To continue preparing to install the base OS, do one of the following, depending on the SMW model.
   - For a Dell R630 SMW, first configure the SMW RAID, then configure the BIOS and iDRAC:
     1. *Configure the Dell R630 SMW RAID Virtual Disks* on page 28
     2. *R630 SMW: Change the BIOS and iDRAC Settings* on page 31
   - For a Dell R815 SMW, just configure the BIOS and iDRAC: *R815 SMW: Change the BIOS and iDRAC Settings* on page 22.

   As a result of following the procedure for changing the BIOS and iDRAC settings, all SMW internal disks that are not to receive the base operating system should be physically ejected from SMW internal disk drive bays.

4.1.1 R815 SMW: Change the BIOS and iDRAC Settings

Prerequisites

This procedure assumes that the SMW is disconnected from the boot RAID and connected to a keyboard, monitor, and mouse.

About this task

This procedure changes the system setup for a Dell R815 SMW: the network connections, remote power control, and the remote console. Depending on the server model and version of BIOS configuration utility, there could be minor differences in the steps to configure the system. For more information, refer to the documentation for the Dell server used at this site. Because Cray ships systems with most of the installation and configuration completed, some of the steps may have been done already.

For a Dell R630 SMW, see *R630 SMW: Change the BIOS and iDRAC Settings* on page 31.

Procedure

1. Remove SMW non-boot internal drives.

   Eject all the internal disk drives from the SMW except for the primary boot disk in slot 0 and the secondary boot disk in slot 1.

2. Power up the SMW. When the BIOS power-on self-test (POST) process begins, **quickly press the F2 key** after the following messages appear in the upper-right of the screen.

   ```
              F2 = System Setup
            F10 = System Services
          F11 = BIOS Boot Manager
                   F12 = PXE Boot
   ```

   When the **F2** keypress is recognized, the **F2 = System Setup** line changes to **Entering System Setup**.

   After the POST process completes and all disk and network controllers have been initialized, the BIOS **System Setup** menu appears.

3. Change system time.

   The system time should be in UTC, not in the local timezone.

a. Select **System Time** in the **System Setup** menu.

   The hours will be highlighted in blue.

b. Set the correct time.

   1. Press the space key to change hours.

   2. Use the right-arrow key to select minutes, then change minutes with the space key.

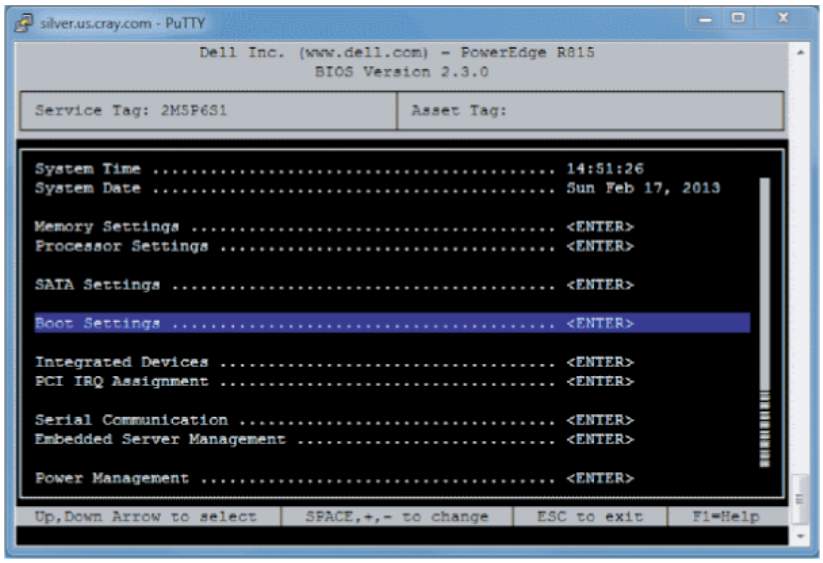
   3. Use the right-arrow key to select seconds, then change seconds with the space key.

c. Press **Esc** when the correct time is set.

4. Change boot settings.

a. Select **Boot Settings** in the **System Setup** menu, then press **Enter**.

Figure 3. Dell R815 SMW Boot Settings Menu

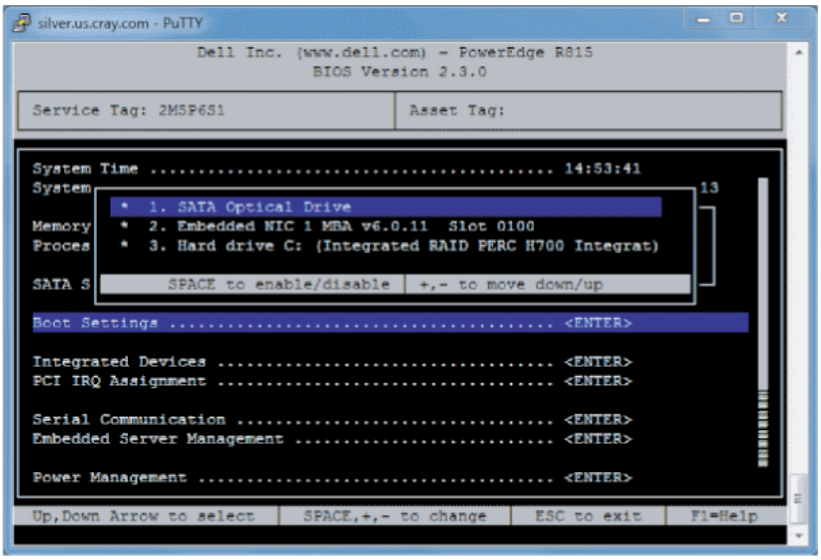


A pop-up menu with the following list appears:

```
Boot Mode ..................................... BIOS
Boot Sequence ................................. <ENTER>
USB Flash Drive Emulation Type................. <ENTER>
Boot Sequence Retry ........................... <Disabled>
```

b. Select **Boot Sequence**, then press **Enter**.

Figure 4. Dell R815 SMW Boot Sequence Settings



c. Change the order of items in the **Boot Sequence** list so that the optical (DVD) drive appears first, then the hard drive. If **Embedded NIC** appears in the list, it should end up below the optical drive and hard drive in the list.

d. Disable embedded NIC.

   Select `Embedded NIC`, then press **Enter** to disable it.

e. Press **Esc** to exit the **Boot Sequence** menu.

f. Press **Esc** again to exit the **Boot Settings** menu.

5. Change serial communication.

a. Select **Serial Communication** in the **System Setup** menu, then press **Enter**.

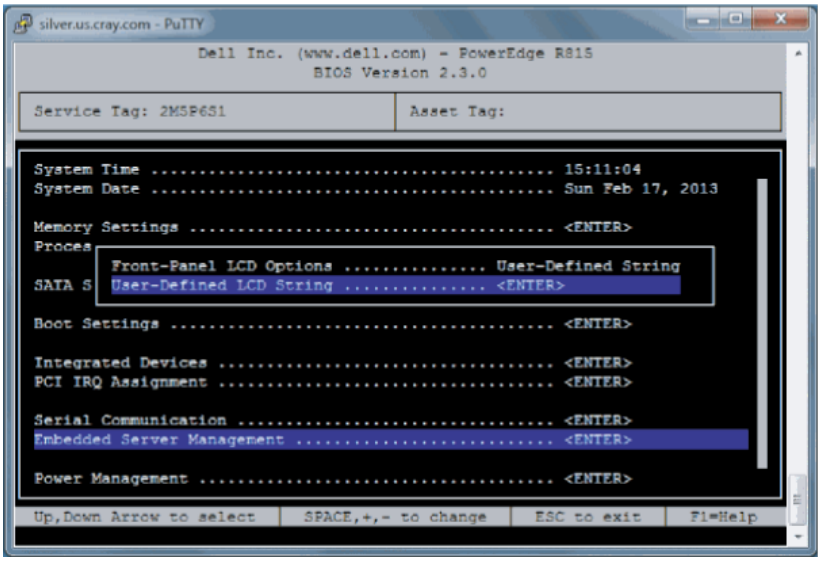
b. Confirm these settings in the **Serial Communication** menu.

- **Serial Communication** is set to **On with Console Redirection via COM2**
- **Serial Port Address** is set to **Serial Device1=COM2, Serial Device2=COM1**
- **External Serial Connector** is set to **Serial Device2**
- **Failsafe Baud Rate** is set to **115200**

c. Press **Esc** to exit the **Serial Communication** menu.

6. Select **Embedded Server Management** in the **System Setup** menu, then press **Enter**.

Figure 5. Dell R815 SMW Embedded Server Management Settings



a. Set **Front-Panel LCD Options** to **User-Defined LCD String** in the **Embedded Server Management** menu. Use the space key to cycle through the choices, then use the down-arrow key.

b. Set **User-Defined LCD String** to the login hostname (e.g., cray-drac), then press **Enter**.

c. Press **Esc** to exit the **Embedded Server Management** menu.

7. Insert base operating system DVD into SMW.

Insert the base operating system DVD labeled Cray-SMWbase12-201511021655 into the DVD drive. (The DVD drive on the front of the SMW may be hidden by a removable decorative bezel.)

8. Save BIOS changes and exit.

a. Press **Esc** to exit the BIOS **System Setup** menu.

A menu with a list of exit options appears.

```
Save changes and exit
Discard changes and exit
Return to Setup
```

b. Ensure that **Save changes and exit** is selected, then press **Enter**.

The SMW resets automatically.

9. Enter BIOS boot manager.

a. When the BIOS POST process begins again, **quickly press the F11 key** within 5 seconds of when the following messages appear in the upper-right of the screen.

```
          F2 = System Setup
        F10 = System Services
      F11 = BIOS Boot Manager
             F12 = PXE Boot
```

When the **F11** keypress is recognized, the **F11 = BIOS Boot Manager** line changes to **Entering BIOS Boot Manager**.

10. Change the integrated Dell Remote Access Controller (iDRAC) settings.

Watch the screen carefully as text scrolls until the **iDRAC6 Configuration Utility 1.57** line is visible. When the line **Press <Ctrl-E> for Remote Access Setup within 5 sec...** displays, press **Ctrl-E** within 5 seconds.

```
0 5 0 ATA WDC WD5000BPVT-0 1A01 465 GB
LSI Corporation MPT2 boot ROM successfully installed!
iDRAC6 Configuration Utility 1.57
Copyright 2010 Dell Inc. All Rights Reserved
iDRAC6 Firmware Revision version: 1.54.15
Primary Backplane Firmware Revision 1.07
------------------------------
IPv6 Settings
------------------------------
IPv6 Stack : Disabled
Address 1 : ::
Default Gateway : ::
------------------------------
IPv4 Settings
------------------------------
IPv4 Stack : Enabled
IP Address : 172. 31. 73.142
Subnet mask : 255.255.255. 0
Default Gateway : 172. 31. 73. 1
Press <Ctrl-E> for Remote Access Setup within 5 sec...
```

The **iDRAC6 Configuration Utility** menu appears.

11. Set **iDRAC LAN** to **ON**.

12. Configure the iDRAC LAN.

Select **LAN Parameters**, then press **Enter**.

a. Configure iDRAC6 name.

Use the arrow key to scroll down and select **iDRAC6 Name**, then press **Enter**. Enter a value for **Current DNS iDRAC6 Name** (e.g., smw-drac), then press **Enter**.

b. Configure domain name.

Use the arrow key to scroll down and select **Domain Name**, then press **Enter**. Enter a value for **Current Domain Name** (e.g., us.cray.com), then press **Enter**.

c. Configure hostname string.

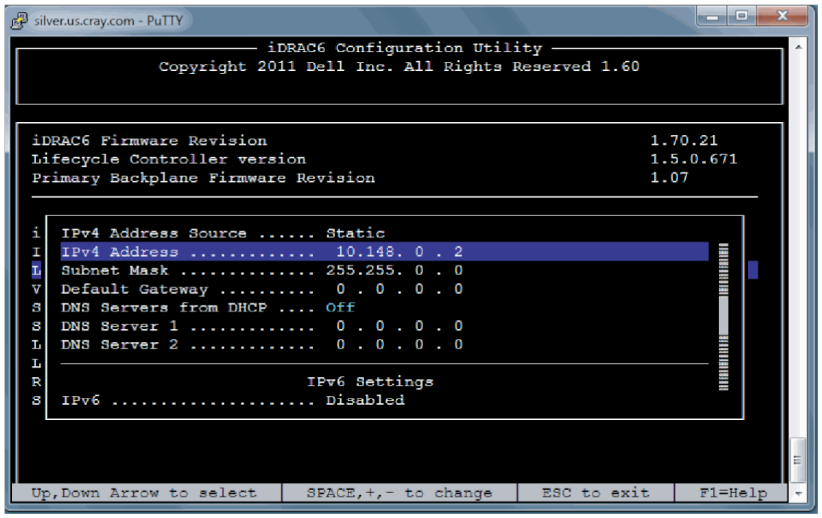
Use the arrow key to scroll down and select **Host Name String**, then press **Enter**. Enter a value for **Current Host Name String** (e.g., smw-drac), then press **Enter**.

d. Configure IPv4 settings.

Use the arrow key to scroll down into the **IPv4 Settings** group and confirm that the **IPv4 Address Source** is set to **static**. Then enter values for the following:

IPv4 Address (the SMW DRAC IP address)
Subnet Mask (the SMW iDRAC subnet mask)
Default Gateway (the SMW iDRAC default gateway)
DNS Server 1 (the first site DNS server)
DNS Server 2 (the second site DNS server)

Figure 6. Dell R815 SMW DRAC IPv4 Parameter Settings



e. Configure IPv6 settings.

   Use the arrow key to scroll down into the **IPv6 Settings** group and ensure that **IPv6** is disabled.

f. Press **Esc** to exit **LAN Parameters** and return to the **iDRAC6 Configuration Utility** menu.

13. Configure iDRAC virtual media.

a. Select **Domain Name**, then press **Enter**.

b. Select **Virtual Media Configuration**, then press **Enter**.

c. Select the **Virtual Media** line and press the space key until it indicates **Detached**.

d. Press **Esc** to exit the **Virtual Media Configuration** menu.

14. Set the password for the iDRAC LAN root account.

Using the arrow keys, select **LAN User Configuration**, then press **Enter**. The following configuration is for both SSH and web browser access to the iDRAC.

a. Select **Account User Name** and enter the account name "root."

b. Select **Enter Password** and enter the intended password.

c. Select **Confirm Password** and enter the intended password again.

d. Press **Esc** to return to the **iDRAC6 Configuration Utility** menu.

15. Exit the iDRAC configuration utility.

a. Press **Esc** to exit the **iDRAC6 Configuration Utility** menu.

b. Select **Save Changes and Exit**.

The **BIOS Boot Manager** menu appears.

16. Choose to boot from SATA Optical Drive.

Using the arrow keys, select the **SATA Optical Drive** entry, then press **Enter**.

Dell R815 SMW system setup (changing default BIOS and iDRAC settings) is now complete. This completes the preparation for installing the base operating system. The next step in the process is *Install the SLES12 Base Linux Distribution on the SMW* on page 41.

4.1.2 Configure the Dell R630 SMW RAID Virtual Disks

Prerequisites

This procedure assumes that the SMW is disconnected from the boot RAID and connected to a keyboard, monitor, and mouse.

About this task

Before installing and configuring SMW software, the base operating system needs to be installed on the SMW. And before the base operating system can be installed, the internal disk drives of the SMW must be configured as RAID virtual disks, as described in this procedure, and the default system setup for the R630 SMW node must be configured, as described in *R630 SMW: Change the BIOS and iDRAC Settings* on page 31.

A Dell R630 SMW has five physical disks. The SMW node must be reconfigured so that the internal Dell PERC RAID controller treats four of these disks as RAID 5 with a hot spare and the fifth disk as non-RAID. This procedure describes how to do that. Because Cray ships systems with most of the installation and configuration completed, some of the steps may be needed only if changes are made to the configuration.

This procedure includes detailed steps for the DELL R630 server using the PERC H330 Mini BIOS Configuration Utility `4.03-0010`. Depending on the server model and version of RAID configuration utility, there could be minor differences in the steps to configure the system. For more information, refer to the documentation for the DELL PERC controller or server RAID controller software.

Procedure

1. Connect a keyboard, monitor, and mouse to the front panel USB and monitor connectors on the SMW, if not already connected.

2. Ensure all SMW internal disk drives are inserted into the SMW drive slots.

3. Power up the SMW. As the SMW node reboots, watch for the Power Edge Expandable RAID Controller section and be ready to press **Ctrl-R** when prompted.

   Cray recommends using the RAID configuration utility (via **Ctrl-R**) to configure the RAID virtual disks instead of the **System Setup Device Settings** menu.

   TIP: In the RAID configuration utility:

   - Use the up-arrow or down-arrow key to highlight an item in a list.
   - Press the **Enter** key to select an item.
   - Press the **F2** key to display a menu of options for an item.
   - Use the right-arrow, left-arrow, or **Tab** key to switch between the **Yes** and **No** buttons in a confirmation window.

4. Clear existing/default disk configuration, if necessary.

   If any disk groups are currently defined:

a. Select **Disk Group 0**, then press **F2**.

b. Select **Delete Disk Group**, then press **Enter**.

c. Select **Yes** in the pop-up confirmation window to confirm the changes.

5. Switch disk controller from HBA-Mode to RAID-Mode.

Some SMW hardware might be configured for HBA-Mode. If it is, then change it to RAID-Mode using the following substeps. If it is not, then skip these substeps.

a. Switch disk controller from HBA-Mode to RAID-Mode.

1. Press **Ctrl-N** (multiple times) to move to the **Ctrl Mgmt** tab.

2. Press **Tab** (multiple times) to get to **Personality Mode**.

3. Press **Enter** to see choice between **RAID-Mode** and **HBA-Mode**.

4. Use the up-arrow or down-arrow key to select **RAID-Mode**, then press **Enter**.

5. Press **Tab** (multiple times) to get to **Apply**, then press **Enter**. This message appears: "The operation has been performed successfully. Reboot the system for the change to take effect."

6. Press **Enter**.

b. Exit RAID configuration utility.

1. Press **Esc** to exit the RAID configuration utility.

2. Select **OK** to confirm, then press **Enter**.

c. Reboot the SMW.

Press **Ctrl-Alt-Delete** at the prompt to reboot. The server will restart the boot process. Be prepared to press **Ctrl-R** when prompted.

d. Enter RAID configuration utility.

As the SMW node reboots, enter the RAID controller configuration utility by pressing **Ctrl-R** when prompted. This will return to the point prior to switching from HBA-Mode to RAID-Mode.

6. Configure most internal disks to appear as `/dev/sda` in a RAID-5 virtual disk.

This step configures most internal disks to appear as /dev/sda in a RAID-5 virtual disk. The R630 has four identical 500-GB drives and one 1-TB drive. The 1-TB drive will be excluded from this RAID-5 configuration. Instead, that drive will be used to hold the postgresql database with Power Management data.

a. Select **No Configuration Present**, then press the **F2** key.

b. Select **Convert to RAID capable**, then press **Enter**. The **Convert Non-RAID Disks to RAID capable** screen appears.

c. Create virtual disk /sda

7. Convert non-RAID disks to RAID-capable.

a. Press **Enter** to check the box for a physical disk, which selects it for this RAID-5 disk group. This action also advances the selection to the next disk. In this manner, select all four of the identical 500-GB drives but exclude the 1-TB drive (leave it unselected).

b. Press **Tab** to move to **OK**, then press **Enter**.

8. Verify the virtual disk changes.

To verify the virtual disk changes, compare settings with those shown in the figure.

9. Create virtual disk sda.

a. Use up-arrow to return to the **No Configuration Present!** item.

b. Press **F2** to see a pop-up menu.

c. Press **Enter** to choose **Create New VD**.

The **Convert Non - RAID Disks to RAID capable** screen appears. The only disk left on this screen should be the large (1-TB) disk which was excluded earlier. It should not be added to the RAID capable set of disks, so continue to exclude it.

d. Press **Tab** to move from the list of disks to **Cancel**, then press **Enter**.

This cancels the conversion of non-RAID disks to RAID capable. The **Create New VD** screen appears.

10. Create new virtual disk (VD).

a. Press **Enter** to switch from **RAID-0** to other options.

b. Use down-arrow to select **RAID-5**, then press **Enter**.

c. Press **Tab** to move to the **Physical Disks** area.

d. Press **Enter** to select each disk except one.

One disk should not be selected so that it can become the hot spare (configured in the next step).

e. Press **Tab** to move to **VD Name**.

f. Select name sda.

g. Press **Tab** to move to **Advanced**, then press **Enter**.

The **Create Virtual Disk-Advanced** screen appears.

11. Configure one disk as the hot spare.

a. Press **Tab** multiple times to move to **Initialize**, then press **Enter** to select it.

A pop-up window with the following text appears: "Initialization will destroy data on the virtual disk. Are you sure you want to continue?"

b. Press **Tab** or arrow keys to move to **OK**, then press **Enter** to confirm initialization.

c. Press **Tab** to move to **Configure HotSpare**, then press **Enter** to select it.

d. Press **Tab** or arrow keys to move to **OK** on the **Create Virtual Disk-Advanced** screen, then press **Enter**.

e. Press **Tab** or arrow keys to move to **OK** on the **Create New VD** screen, then press **Enter**.

A pop-up window with the following text appears: "Virtual disk is successfully created and initialized."

f. Press **Tab** or arrow keys to move to **OK**, then press **Enter**.

A pop-up window with the following text appears: "Dedicated Hotspare for Disk Group 0."

g. Select the disk to be the hot spare, then press **Enter**.

h. Press **Tab** or arrow keys to move to **OK**, then press **Enter**.

A pop-up window with the following text appears: "Initialization complete on VD 0."

i. Press **Tab** or arrow keys to move to **OK**, then press **Enter**.

The RAID will initialize in the background.

12. Exit RAID configuration utility.

Exit the RAID configuration utility, reboot, and then begin installing the base operating system.

a. Press the **Esc** key to exit the RAID configuration utility.

b. Select **OK**, then press **Enter** to confirm.

13. Reboot the system.

A message appears that prompts to reboot.

ATTENTION: Only the disk drives configured to be the RAID-5 virtual disk sda should be inserted into the SMW internal drive bays when installing SLES12.

a. Eject the 1-TB disk (which was not added to the RAID-5 virtual disk sda) from the SMW.

This will be re-inserted when SLES12 installation is complete.

b. Press **Ctrl-Alt-Delete**.

The server will restart the boot process and will not interrupt RAID initialization. During the system reboot, be prepared to press **F2** when prompted, to change the system setup.

RAID configuration is now complete. The next step in preparing to install the base operating system is to configure the system setup for the R630 SMW node, as described in *R630 SMW: Change the BIOS and iDRAC Settings* on page 31.

4.1.3 R630 SMW: Change the BIOS and iDRAC Settings

Prerequisites

This procedure assumes that the internal disk drives of the SMW have just been configured as RAID virtual disks and the system is rebooting. If the system is not rebooting, press **Ctrl-Alt-Delete** to reboot.

About this task

This procedure describes how to change the system setup for the SMW: the network connections, remote power control, and the remote console. This procedure includes detailed steps for the Dell R630 server. Depending on the server model and version of BIOS configuration utility, there could be minor differences in the steps to configure the system. For more information, refer to the documentation for the Dell server used at this site. Because Cray ships systems with most of the installation and configuration completed, some of the steps may have been done already.

For a Dell R815 server, see *R815 SMW: Change the BIOS and iDRAC Settings* on page 22.

Procedure

Watch as the system reboots and the BIOS power-on self-test (POST) process begins. Be prepared to press **F2**, when prompted, to change the system setup.

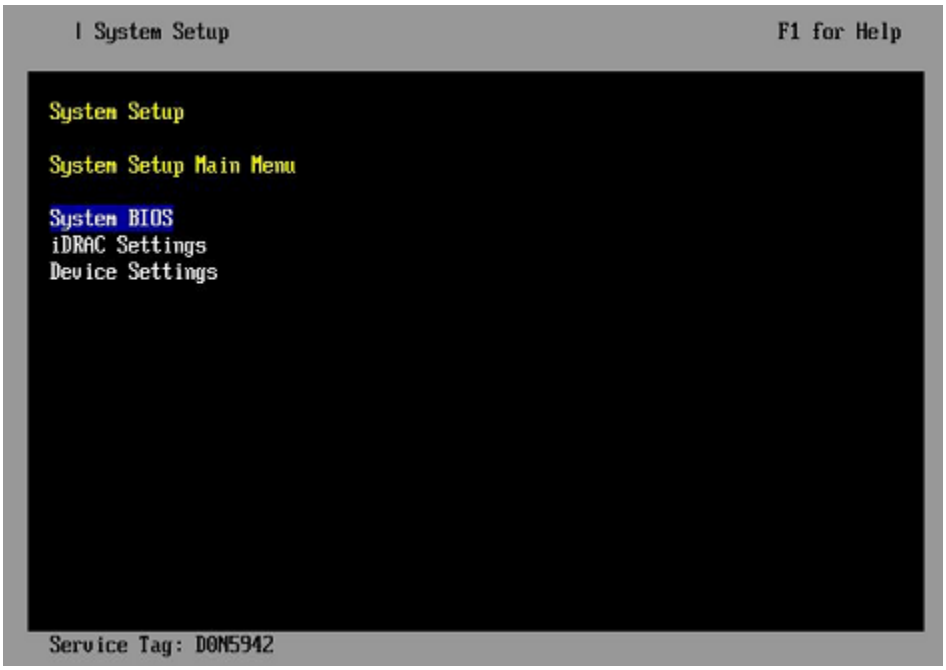
1. Press the **F2** key immediately after the following messages appear in the upper-left of the screen:

```
F2 = System Setup
F10 = System Services
F11 = BIOS Boot Manager
F12 = PXE Boot
```

When the **F2** keypress is recognized, the **F2 = System Setup** line changes color from white-on-black to white-on-blue.

After the POST process completes and all disk and network controllers have been initialized, the Dell **System Setup** screen appears. The following submenus are available on the **System Setup Main Menu** and will be used in subsequent steps: **System BIOS**, **iDRAC Settings**, and **Device Settings**.

Figure 7. Dell R630 System Setup Main Menu



TIP: In system setup screens,

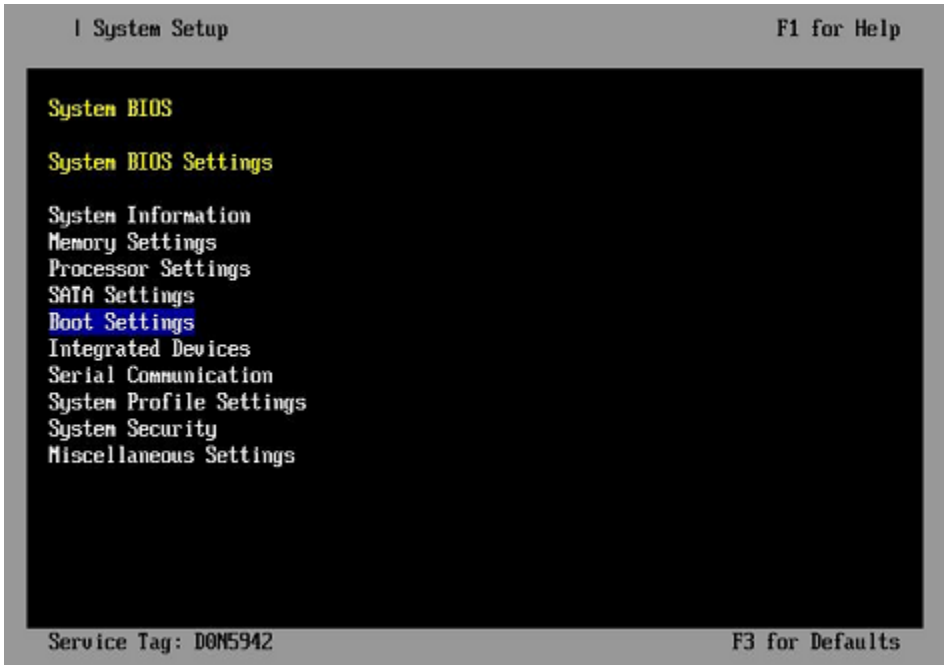
- Use the **Tab** key to move to different areas on the screen.
- Use the up-arrow and down-arrow keys to highlight or select an item in a list, then press the **Enter** key to enter or apply the item.
- Press the **Esc** key to exit a submenu and return to the previous screen.

2. Change the BIOS settings.

a. Select **System BIOS** on the **System Setup Main Menu**, then press **Enter**.

The **System BIOS Settings** screen appears.

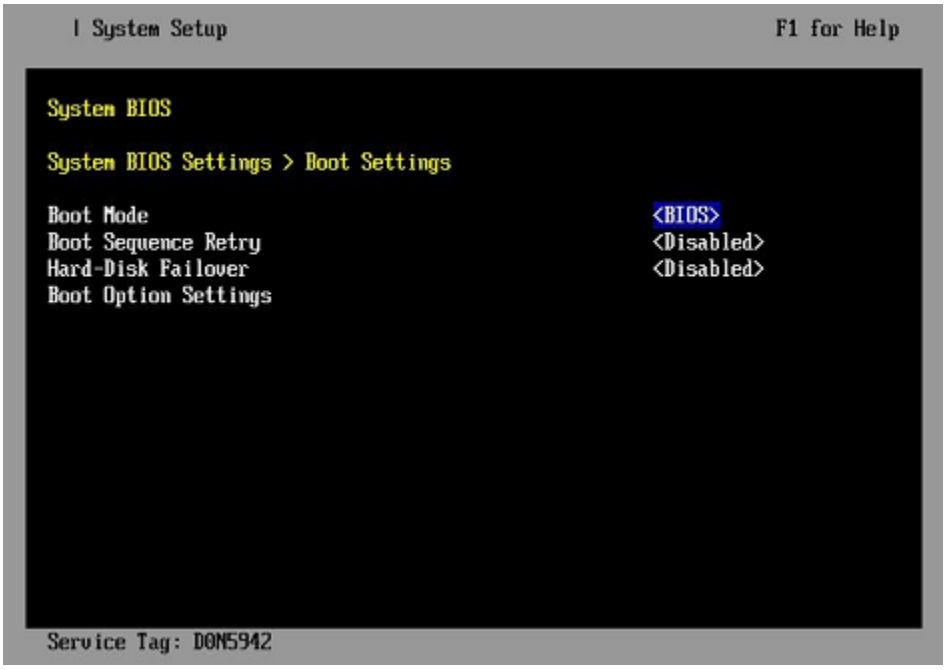
Figure 8. Dell R630 System BIOS Settings Screen



b. Change Boot Settings.

1. Select **Boot Settings** on the **System BIOS Settings** screen, then press **Enter**. The **Boot Settings** screen appears.

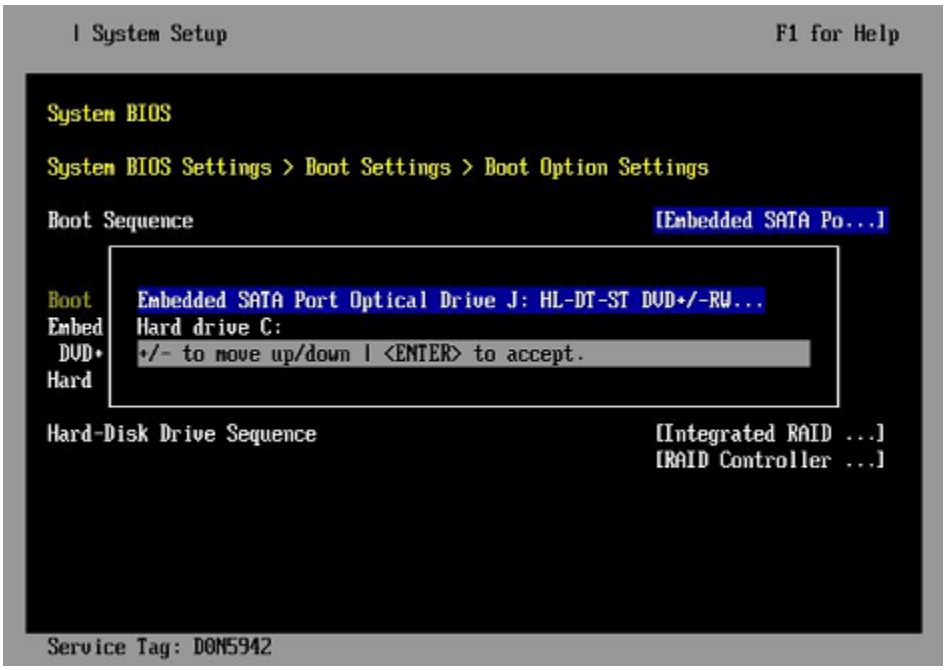
   Figure 9. Dell R630 Boot Settings Screen



2. Ensure that **Boot Mode** is **BIOS** and not **UEFI**.
3. Select **Boot Option Settings**, then press **Enter**.

4. Select **Boot Sequence** on the **Boot Option Settings** screen, then press **Enter** to view a pop-up window with the boot sequence.

   Figure 10. Dell R630 BIOS Boot Sequence



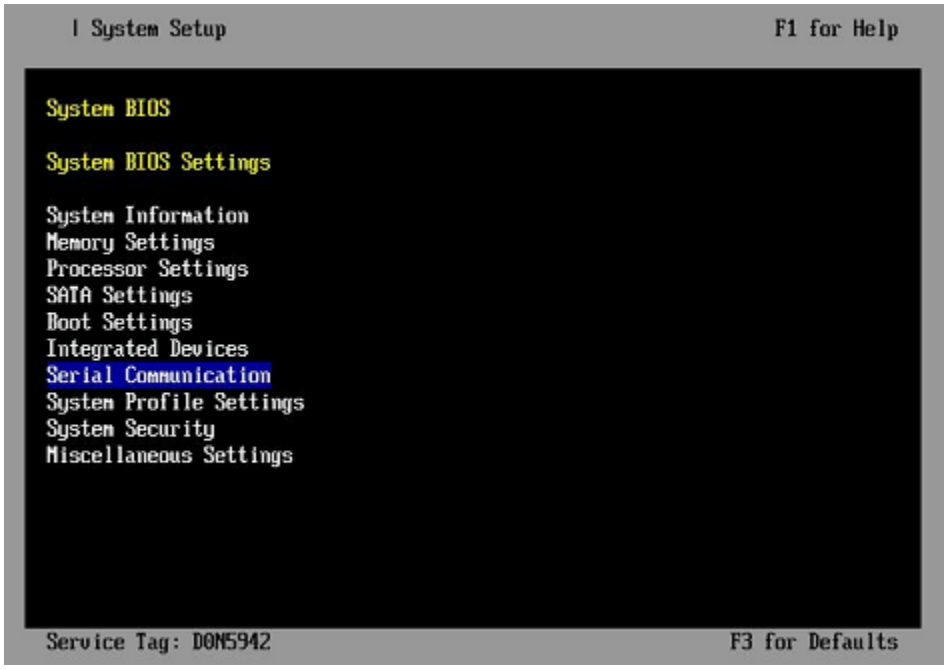
5. Change the boot order in the pop-up window so that the optical drive appears first, then the hard drive. If **Integrated NIC** appears in the list, it should end up below the optical drive and hard drive in the list.

   TIP: Use the up-arrow or down-arrow key to highlight or select an item, then use the **+** and **-** keys to move the item up or down.

6. Select **OK**, then press **Enter** to accept the change.
7. Click the box next to **Hard drive C:** under the **Boot Option/Enable/Disable** section to enable it. Do the same for the optical drive, if necessary.
8. Select **integrated NIC**, then press **Enter** to disable it.
9. Press **Esc** to exit **Boot Option Settings**.
10. Press **Esc** to exit **Boot Settings** and return to the **System BIOS Settings** screen.

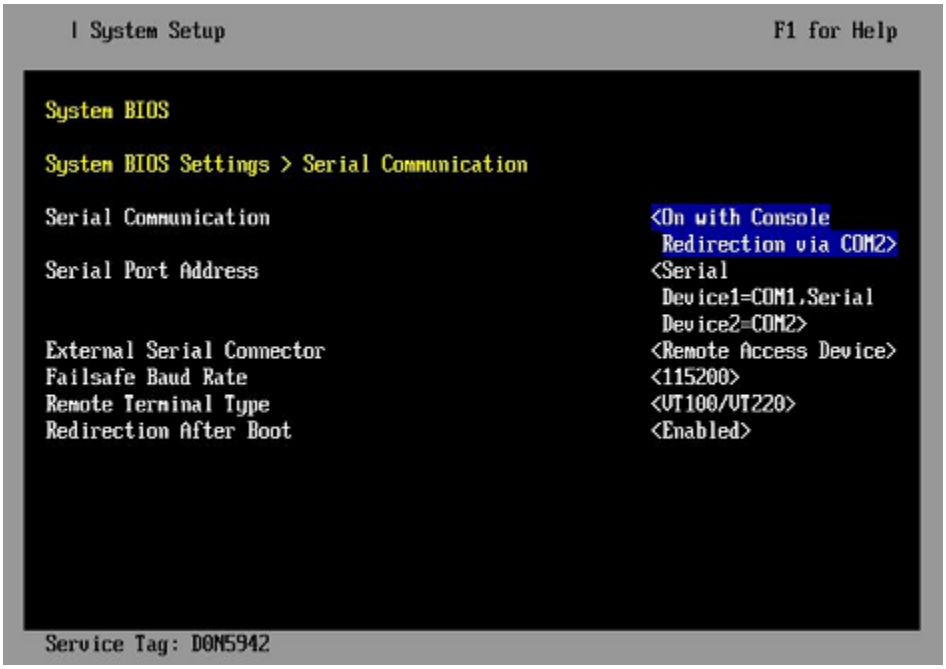
c. Change Serial Communication Settings.

Figure 11. Dell R630 System BIOS Settings: Serial Communication



1. Select **Serial Communication** on the **System BIOS Settings** screen. The **Serial Communication** screen appears.

   Figure 12. Dell R630 Serial Communication Screen



2. Select **Serial Communication** on the **Serial Communication** screen, then press **Enter**. A pop-up window displays the available options.
3. Select **On with Console Redirection via COM2** in the pop-up window, then press **Enter** to accept the change.

4. Select **Serial Port Address**, then select **Serial Device1=COM1, Serial Device2=COM2**, then press **Enter**.
5. Select **External Serial Connector**, then press **Enter**. A pop-up window displays the available options.
6. Select **Remote Access Device** in the pop-up window, then press **Enter** to return to the previous screen.
7. Select **Failsafe Baud Rate**, then press **Enter**. A pop-up window displays the available options.
8. Select **115200** in the pop-up window, then press **Enter** to return to the previous screen.
9. Press the **Esc** key to exit the **Serial Communication** screen.
10. Press **Esc** to exit the **System BIOS Settings** screen. A "Settings have changed" message appears.
11. Select **Yes** to save changes. A "Settings saved successfully" message appears.
12. Select **Ok**.

3. Change the iDRAC (Integrated Dell Remote Access Controller) settings.

   Select **iDRAC Settings** on the **System Setup Main Menu**, then press **Enter**.

   The **iDRAC Settings** screen appears.

   Figure 13. Dell R630 iDRAC Settings Screen

4. Change the iDRAC network.

   a. Select **Network** to display a long list of network settings.

   b. Change the DNS DRAC name.

      Use the arrow key to scroll down to **DNS DRAC Name**, then enter an iDRAC hostname that is similar to the SMW node hostname (e.g., cray-drac).

   c. Change the static DNS domain name.

Use the arrow key to scroll down to **Static DNS Domain Name**, then enter the DNS domain name and press **Enter**.

d. Change the IPv4 settings.

Use the arrow key to scroll down to the **IPV4 SETTINGS** list.

1. Ensure that IPv4 is enabled.
   a. If necessary, select **Enable IPV4**, then press **Enter**.
   b. Select **<Enabled>** in the pop-up window, then press **Enter** to return to the previous screen.
2. Ensure that DHCP is disabled.
   a. If necessary, select **Enable DHCP**, then press **Enter**.
   b. Select **<Disabled>** in the pop-up window, then press **Enter** to return to the previous screen.
3. Change the IP address.
   a. Select **Static IP Address**.
   b. Enter the IP address of the iDRAC interface (`ipmi0`) for the SMW, then press **Enter**.
4. Change the gateway.
   a. Select **Static Gateway**.
   b. Enter the appropriate value for the gateway of the network to which the iDRAC is connected, then press **Enter**.
5. Change the subnet mask.
   a. Select **Subnet Mask**.
   b. Enter the subnet mask for the network to which the iDRAC is connected (such as `255.255.255.0`), then press **Enter**.
6. Change the DNS server settings.
   a. Select **Static Preferred DNS Server**, enter the IP address of the primary DNS server, then press **Enter**.
   b. Select **Alternate DNS Server**, enter the IP address of the alternate DNS server, then press **Enter**.

e. Change the IPMI settings.

Change the IPMI settings to enable the Serial Over LAN (SOL) console.

1. Use the arrow key to scroll down to the **IPMI SETTINGS** list.
2. Ensure that **Enable IPMI over LAN** is selected.

   TIP: Use the left-arrow or right-arrow to switch between two settings.
3. Ensure that **Channel Privilege Level Limit** is set to **Administrator**.

f. Exit Network screen.

Press the **Esc** key to exit the **Network** screen and return to the **iDRAC Settings** screen.

5. Change hostname in iDRAC LCD display.

Change front panel security to show the hostname in LCD display.

a. Use the arrow key to scroll down and highlight **Front Panel Security** on the **iDRAC Settings** screen, then press **Enter**.

b. Select **Set LCD message**, then press **Enter**.

c. Select **User-Defined String**, then press **Enter**.

d. Select **User-Defined String**, then enter the SMW hostname and press **Enter**.

e. Press the **Esc** key to exit the **Front Panel Security** screen.

6. (Optional) Change the iDRAC **System Location** fields.

Change the **System Location** configuration on the **iDRAC Settings** screen to set any of these fields: **Data Center Name**, **Aisle Name**, **Rack Name**, and **Rack Slot**.

7. Set the password for the iDRAC root account.

a. Use the arrow key to highlight **User Configuration** on the **iDRAC Settings** screen, then press **Enter**.

b. Confirm that User Name is root. Select **User Name**, then enter the "root" user name.

c. Select **Change Password**, then enter a new password.

d. Reenter the new password in the next pop-up window to confirm it (the default password is "calvin").

e. Press the **Esc** key to exit the **User Configuration** screen.

8. Exit iDRAC settings.

a. Press the **Esc** key to exit the **iDRAC Settings** screen.

A "Settings have changed" message appears.

b. Select **Yes**, then press **Enter** to save the changes.

A "Success" message appears.

c. Select **Ok**, then press **Enter**.

The main screen (**System Setup Main Menu**) appears.

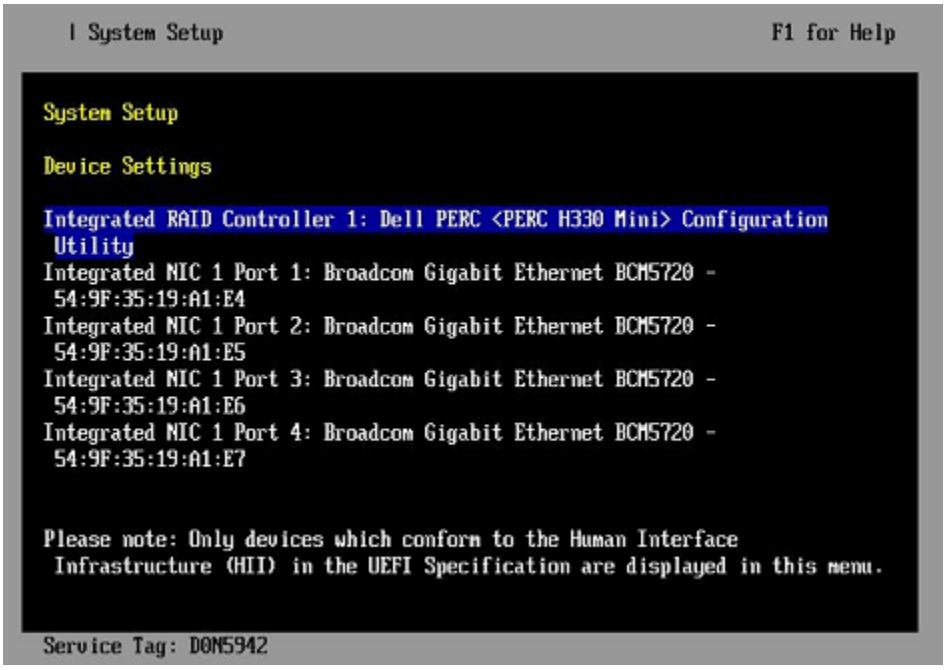
9. Change device settings.

These steps disable an integrated NIC device by changing the setting for the integrated NIC on a port from **PXE** to **None**.

a. Change Integrated NIC 1 Port 1

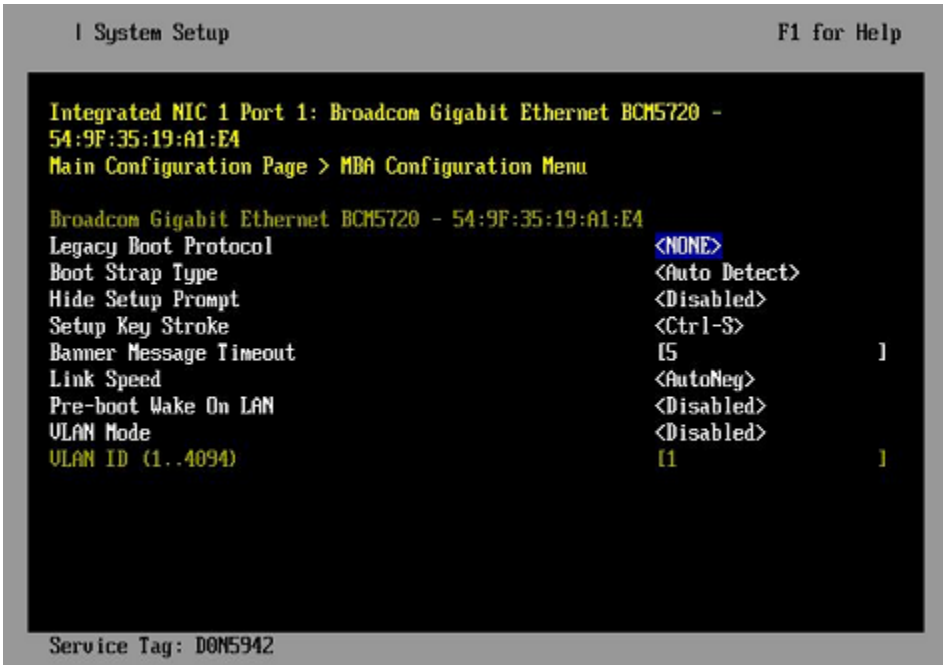
1. Select **Device Settings** on the **System Setup Main Menu**, then press **Enter**. The **Device Settings** screen appears.

Figure 14. Dell R630 Device Settings Screen



2. Select **Integrated NIC 1 Port 1: ...** on the **Device Settings** screen, then press **Enter**.
3. Select **MBA Configuration Menu** on the **Main Configuration Page** screen, then press **Enter**.

Figure 15. Dell R630 BIOS MBA Configuration Settings



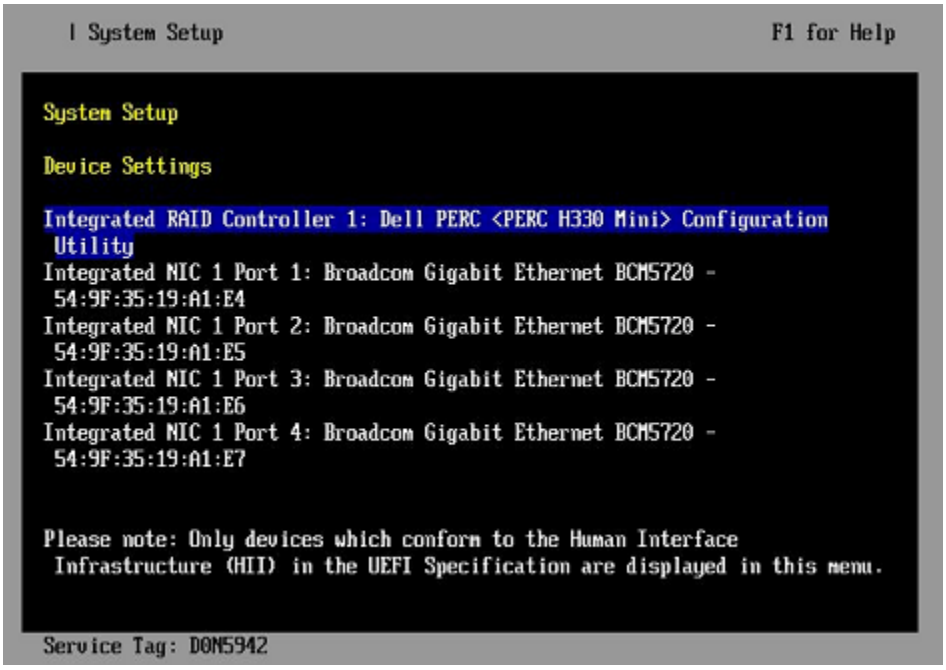
4. Select **Legacy Boot Protocol** on the **MBA Configuration Menu** screen, use the right-arrow or left-arrow key to highlight **None**, then press **Enter**.
5. Press the **Esc** key to exit the **MBA Configuration Menu** screen.

6. Press **Esc** to exit the **Main Configuration Page** screen. A "Warning Saving Changes" message appears.
7. Select **Yes**, then press **Enter** to save the changes. A "Success" message appears.
8. Select **OK**, then press **Enter**. The **Device Settings** screen appears.
9. Press **Esc** to exit the **Device Settings** screen. A "Settings have changed" message appears.
10. Select **Yes**, then press **Enter** to save the changes. A "Settings saved successfully" message appears.
11. Select **OK**, then press **Enter**. The main screen (**System Setup Main Menu**) appears.

b. Change Integrated NIC 1 Port 2

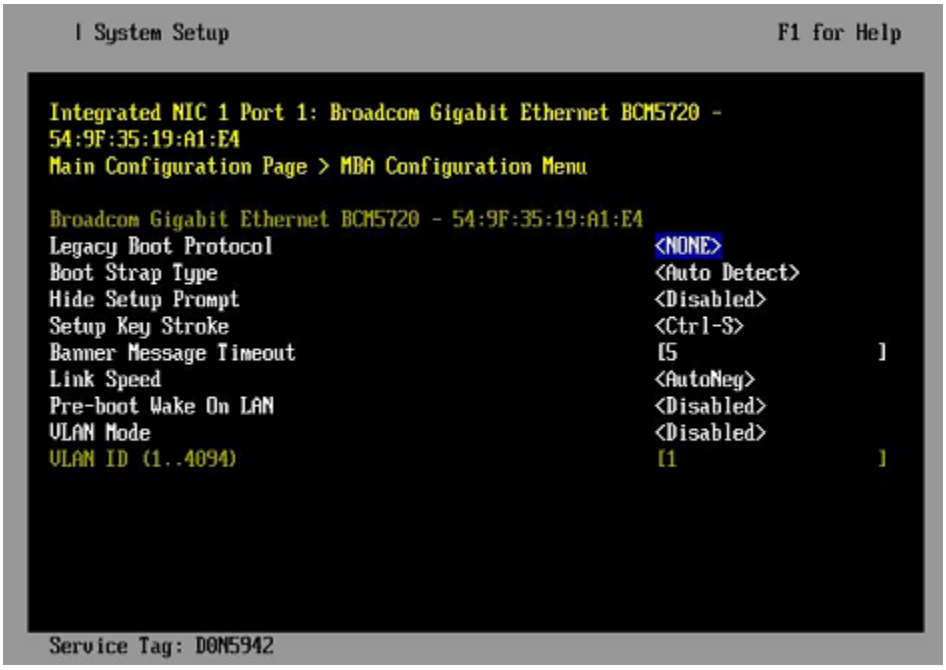
1. Select **Device Settings** on the **System Setup Main Menu**, then press **Enter**. The **Device Settings** screen appears.

Figure 16. Dell R630 Device Settings Screen



2. Select **Integrated NIC 1 Port 2: ...** on the **Device Settings** screen, then press **Enter**.
3. Select **MBA Configuration Menu** on the **Main Configuration Page** screen, then press **Enter**.

Figure 17. Dell R630 BIOS MBA Configuration Settings



4. Select **Legacy Boot Protocol** on the **MBA Configuration Menu** screen, use the right-arrow or left-arrow key to highlight **None**, then press **Enter**.
5. Press the **Esc** key to exit the **MBA Configuration Menu** screen.
6. Press **Esc** to exit the **Main Configuration Page** screen. A "Warning Saving Changes" message appears.
7. Select **Yes**, then press **Enter** to save the changes. A "Success" message appears.
8. Select **OK**, then press **Enter**. The **Device Settings** screen appears.
9. Press **Esc** to exit the **Device Settings** screen. A "Settings have changed" message appears.
10. Select **Yes**, then press **Enter** to save the changes. A "Settings saved successfully" message appears.
11. Select **OK**, then press **Enter**. The main screen (**System Setup Main Menu**) appears.

SMW system setup (changing default BIOS and iDRAC settings) is now complete. This completes the preparation for installing the base operating system. The next step in the process is *Install the SLES12 Base Linux Distribution on the SMW* on page 41.

4.2 Install the SLES12 Base Linux Distribution on the SMW

Prerequisites

This procedure assumes the following:

- The BIOS and iDRAC settings have just been changed on the SMW and it is restarting the boot process.

- All SMW internal disks that are not to receive the operating system are physically ejected from SMW internal disk drive bays.
- All connections to the boot RAID are unplugged so that no disk devices from the boot RAID will inadvertently lose existing data or receive the operating system.

About this task

This procedure describes the base operating system installation process. It provides detailed instructions for installing SLES12 on the SMW (both Dell R815 and R630 models); configuring the SMW; and performing final steps: reconnect cables, reinsert drives, and reboot the SMW.

Procedure

SLES12 SOFTWARE PACKAGE INSTALLATION

1. Select one of the **Cray SMW Initial Install** options.

   Within 10 to 15 seconds after this **SUSE Linux Enterprise Server** boot menu displays, use the arrow key to scroll down and select one of the install options, then press **Enter**.

```
- Boot from Hard Disk
- Cray SMW Initial Install with software RAID1
- Cray SMW Initial Install without software RAID
- Rescue System
- Check Installation Media
- Firmware Test
- Memory Test
```

   Select the option that is best for the SMW model:

For a Dell R815 SMW	Select **Cray SMW Initial Install with software RAID1**, a mirrored boot disk option, which creates a software RAID1 mirror on the first two drives. This option is best for a Dell R815 because the R815 should use two disk drives to become the software RAID1 mirror.
For a Dell R630 SMW	Select **Cray SMW Initial Install without software RAID**, a non-mirrored boot disk option, for servers with a single disk or virtual disk. This option is best for a Dell R630 because the R630 should have the internal RAID controller configured to present four disk drives as a virtual disk.

   WARNING: If the selection is not made in time, the system will boot from the default selection, which is **Boot from Hard Disk**. If that happens, shut down the SMW, then start the power-up sequence again.

   Note: The upper left corner of the installation screen has a date/time stamp for when the bootable SLES12 DVD was created.

   As the base installation progresses, the following phases appear on the screen:

```
Starting ... Loading Linux kernel
Initializing
Preparing System for Automated Installation
Initializing the Installation Environment
System Probing
Installation Settings
```

2. Review installation settings while the installation pauses on the **Installation Settings** screen.

3. Confirm the language for the SMW.

   English (US) is the primary language by default. To change the primary language:

   a. Select the **Language** heading in the **Installation Settings** screen.

      The **Languages** window opens.

   b. Select a language (or multiple languages) from the drop-down menu, then select **Accept** at the bottom of the window.

4. Begin automated install.

   a. On the **Installation Settings** screen, select **Install**.

      The **Confirm Installation** pop-up window appears.

   b. Select **Install**.

   The installation of software packages runs for approximately 20 minutes. The process automatically reboots the SMW from the hard disk, and the installation process continues with system configuration.

   SYSTEM CONFIGURATION

5. Log in to SMW as root.

   When the login screen is displayed with the crayadm account as the account which will be logged in:

   a. Select **Not listed?**, then enter `root` for the username.

   b. Either press **Enter** or select **Sign In**.

   c. Enter the password for root.

6. Change default passwords on the SMW by executing the following commands.

   The SMW contains its own `/etc/passwd` file that is separate from the password file for the rest of the CLE system.

   ```
   smw# passwd root
   ```

   ```
   smw# passwd crayadm
   ```

   ```
   smw# passwd mysql
   ```

7. Change the SMW local time zone, if needed.

   The default timezone on the SMW is **America/Chicago**. To change it:

   a. Execute this command:

      ```
      smw# yast2 timezone
      ```

      `yast2` opens a new window for changing the timezone, then a pop-up window appears with this message: "file `/etc/ntp.conf` has been changed manually. YaST might lose some of the changes."

   b. Select the **Do not show this message anymore** checkbox, then select **Continue**.

c. Choose the timezone either by selecting a region on the map or by using the drop-down menus for **Region** and **Time Zone**.

d. Select **Other Settings** if the time is incorrect, then select the **Manually** radio button and enter **Current Time** and **Current Date**. Select **Accept** when done.

e. Select **OK** when done with timezone settings.

8. Configure the SMW firewall.

The SuSE firewall settings may need to be adjusted to match site firewall policy and to customize for site IP addresses. These steps enable and configure the firewall.

TIP: It is not necessary to shut down the system before performing this task.

a. Save the SuSE firewall configuration.

Before modifying the SuSE firewall settings, make a copy of the configuration file.

```
smw# cp -p /etc/sysconfig/SuSEfirewall2 /etc/sysconfig/SuSEfirewall2.orig
```

b. Check current firewall settings.

Check current firewall settings and change to support any site requirements. During the process of configuring Cray SMW and CLE software, some of the firewall settings may be adjusted. SSH access is one of the protocols permitted through the firewall from the external network to the SMW.

```
smw# iptables -L
smw# vi /etc/sysconfig/SuSEfirewall2
```

c. Start the firewall immediately.

Invoke the modified configuration.

```
smw# systemctl start SuSEfirewall2_init.service
smw# systemctl start SuSEfirewall2.service
```

d. Ensure that the firewall will start at next boot.

Execute the following commands to start the firewall at boot time.

```
smw# systemctl enable SuSEfirewall2_init.service
smw# systemctl enable SuSEfirewall2.service
```

e. Verify firewall changes.

Verify the changes to the `iptables`.

```
smw# iptables -nvL
```

9. Configure LAN on the SMW.

Set network configuration for eth0 and the hostname for the SMW.

a. Execute this command:

```
smw# yast2 lan
```

The **Network Settings** screen appears with the **Overview** tab highlighted.

b. Select the **eth0** line on the **Overview** tab, then select **Edit**.

The **Network Card Setup** screen appears with the **Address** tab highlighted.

c. Select **Statically Assigned IP address** on the **Address** tab and enter values for IP address, subnet mask, and hostname (including the domain name). Then select **Next**.

d. Select the **Hostname/DNS** tab on the **Network Settings** screen.

   1. For the **Hostname and Domain Name** area, enter Hostname and Domain Name.
   2. For the **Name Servers and Domain Search List**, enter Name Server 1, Name Server 2, Name Server 3, and Domain Search.

e. Select the **Routing** tab on the **Network Settings** screen, then enter the Default IPv4 Gateway (for the network connected to eth0) and set Device to eth0 using the dropdown menu.

f. Click **OK** after all of the **Network Settings** have been prepared.

FINAL STEPS

10. Reconnect boot RAID disk cables.

Remove the protective covers from the Fibre Channel or SAS cable connectors, clean the ends of the cable connectors, and reconnect the data cables that connect the SMW to the boot RAID.

11. Reinsert SMW non-boot internal drives.

Reinsert all of the SMW internal disk drives that were removed earlier.

> **TIP:** It is not necessary to turn off the power for the SMW before inserting these drives—the operating system can be in a booted state.

12. Eject the Cray-SMWbase12 DVD.

If the base operating system DVD (Cray-SMWbase12-201511021655) is still in the DVD drive, eject it.

```
smw# eject
```

13. Reboot the SMW.

Reboot the SMW to allow the SMW to discover the drives properly.

```
smw# reboot
```

If the SMW was configured with RAID1, then it may still be synchronizing the data between the two disks in the RAID1 mirror. The resync can take about 30 minutes when SLES12 is freshly installed. If the SMW is rebooted at this point in the process, that resync will be interrupted. However, that is not a problem because as soon as the SMW is up again, the resync process will continue.

(R815 SMW only) To check the status of any RAID1 resync activities on an R815 SMW, look at `/proc/mdstat`.

In this example, the resync of md127 finishes in 24.3 minutes.

```
smw# cat /proc/mdstat
Personalities : [raid1]
md125 : active raid1 sdc2[1] sda2[0]
      33559424 blocks super 1.0 [2/2] [UU]
      bitmap: 0/1 pages [0KB], 65536KB chunk

md126 : active raid1 sda1[0] sdc1[1]
      4200384 blocks super 1.0 [2/2] [UU]
```

```
      bitmap: 0/1 pages [0KB], 65536KB chunk

md127 : active raid1 sda3[0] sdc3[1]
      206437248 blocks super 1.0 [2/2] [UU]
      [======>.............]  resync = 33.7% (69700352/206437248)
finish=24.3min speed=93748K/sec
      bitmap: 2/2 pages [8KB], 65536KB chunk

unused devices: <none>
```

The next step in the installation and configuration of the SMW base operating system is *Configure Boot RAID Devices* on page 46.

> **NOTE:** (SMW HA only): The boot RAID configuration procedures are only done on the first SMW. Skip the boot RAID procedures for the second SMW.

4.3 Configure Boot RAID Devices

In typical system installations, the RAID provides the storage for file systems used by the SMW, boot node, and SDB node. These file systems are prepared from LVM volumes in LVM volume groups using the physical volumes that are created on the RAID LUNs (logical unit numbers) or volumes.

In typical system installations, RAID units also provide user and scratch space and can be configured to support a variety of file systems. For more information about configuring RAID, see *Lustre Administration Guide for CLE* (S-0010), which is provided with the CLE release package.

Boot RAID Configuration Procedures

Cray provides support for system boot RAID from NetApp, Inc.

> **NOTE:** Cray ships systems with much of this software installed and configured. Performing all of the steps in these boot RAID procedures may not be necessary unless the configuration needs to be changed.

1. Configure the boot RAID for a NetApp, Inc.Storage System using the following procedures. To configure the boot RAID for a NetApp, Inc. Storage System, first install the SANtricity Storage Manager Utility. Then use that utility to perform the next procedure, which creates/configures volumes and assigns them to LUNs.
   a. *Install SANtricity Storage Manager for NetApp, Inc. Devices* on page 48. SANtricity is provided as a separate package and is installed from a CD. It may already be installed on the SMW.
   b. *Configure the Boot RAID for a NetApp, Inc. Storage System*. Assumes familiarity with using the SANtricity interface.
   c. Configure remote logging of the boot RAID messages. The NetApp, Inc. storage system uses SNMP to provide boot RAID messages. Cray does not provide a procedure for this; see *NetApp, Inc. Storage System documentation* for information about how to configure remote logging.
2. Zone the SAS (Serial Attached SCSI) or FC (Fibre Channel) switch. For FC storage, there will be an FC Switch to be configured. For SAS storage, there will be a SAS Switch to be configured. Use the applicable procedure(s):
   - Zone the QLogic switch and back it up:
     1. *Zone the QLogic FC Switch* on page 52.

2. (Recommended) *Create a Backup of the QLogic Switch Configuration* on page 54.

- *Zone the Brocade FC Switch* on page 55
- *Zone the LSI SAS Switch* on page 62

3. *Reboot the SMW and Verify LUNs are Recognized* on page 65.
4. *Set Up Boot RAID Space for Direct-attached Lustre* on page 50.

Prerequisites and Assumptions for Configuring the Boot RAID

Sites that require long distance between the SMW, XC, and the boot RAID would require Fibre Channel (FC) components, while sites that have the SMW, XC, and boot RAID in the same area (within 10 meters) will typically use SAS as the interface for the boot RAID.

- The SMW has an Ethernet connection to the Hardware Supervisory System (HSS) network.
- The SMW has a Fibre Channel (FC) or Serial Attached SCSI (SAS) connection to the boot RAID or to an FC or SAS switch.
- The boot nodes have an FC or SAS connection to the boot RAID or to an FC or SAS switch.
- The SDB nodes have an FC or SAS connection to the boot RAID or to an FC or SAS switch.

4.3.1 Recommended Boot RAID LUN Values

The recommended boot RAID LUN configuration is shown in these tables for different sizes of boot RAID: 4.5 TB, 9.0 TB, and 1.5 TB.

Boot RAID with 4.5 TB Available, Non-partitioned System

For a boot RAID with 4.5 TB available, use these values for a non-partitioned system. This is the default configuration installed in the factory.

LUN	Label	Size	Segment Size
0	smw0	3000 GB	256 KB
1	boot0	1000 GB	256 KB
2	sdb0	200 GB	256 KB

Boot RAID with 4.5 TB Available, Multiple Partitions

For a boot RAID with 4.5 TB available, use these values for a system with multiple CLE partitions.

- There must be one SMW LUN for the entire system with a size of at least 1000GB.
- There must be one boot LUN for each partition with a size of at least 500GB.
- There must be one SDB LUN for each partition with a size of at least 100GB.

This table shows example values for three CLE partitions.

LUN	Label	Size	Segment Size
0	smw1	2500 GB	256 KB

LUN	Label	Size	Segment Size
1	boot1	500	256 KB
2	sdb1	100 GB	256 KB
3	boot2	500 GB	256 KB
4	sdb2	100 GB	256 KB
5	boot3	500 GB	256 KB
6	sdb3	100 GB	256 KB

Boot RAID with 9.0 TB Available, Non-partitioned System

For a boot RAID with 9.0 TB available, use these values for a non-partitioned system. Values for boot1 and sdb1 LUNs are shown also, because they can be added to volume groups for the boot node volume group and SDB node volume group, if needed. If added, they should be the same size as the boot0 and sdb0.

LUN	Label	Size	Segment Size
0	smw0	4000 GB	256 KB
1	boot0	1000	256 KB
2	sdb0	200 GB	256 KB
3	boot1	1000 GB	256 KB
4	sdb1	200 GB	256 KB

Boot RAID with 1.5 TB Available, Non-partitioned System

For a boot RAID with only 1.5 TB available, use these values for a non-partitioned system.

LUN	Label	Size	Segment Size
0	smw0	1000 GB	256 KB
1	boot0	400 GB	256 KB
2	sdb0	100 GB	256 KB

4.3.2 Install SANtricity Storage Manager for NetApp, Inc. Devices

About this task

The SANtricity Storage Manager software is generally preinstalled and the SANtricity media is shipped with the system. If the SANtricity software is installed, then the `SMclient` executable will be found in `/opt/SMgr/client`. If this Cray system does not have the software installed on the SMW, install it using this procedure.

Procedure

1. Prepare X Windows for NetApp SANtricity Storage Manager.

   The NetApp installation software will launch an X Windows application, so an X Windows server must be ready. There are many ways to prepare this: logging into SMW console as root, logging into SMW console as crayadm and then becoming root, or logging into SMW from a remote workstation with X Windows port forwarding enabled via ssh.

   - If already logged in to the SMW as crayadm, su to root and enable X Windows port forwarding:

     ```
     crayadm@smw> su -
     smw# ssh -X localhost
     ```

   - If not already logged on to the SMW, log in and enable X Windows port forwarding like this:

     ```
     user@host> ssh -X root@smw
     ```

2. Copy NetApp SANtricity Storage Manager installer to SMW.

   - If installing from the SANtricity Storage Manager CD, insert it into the SMW CD drive and mount the CD.

     ```
     smw# mount /dev/cdrom /media/cdrom
     smw# mkdir -p /tmp/netapp
     smw# cp -p /media/cdrom/SMIA-LINUXX64-11.10.0A00.0024.bin /tmp/netapp
     smw# umount /media/cdrom
     smw# eject
     ```

   - If installing from the `SMIA-LINUXX64-11.10.0A00.0024.bin` file, copy `SMIA-LINUX-10.70.A0.25.bin` to `/tmp/netapp`.

     ```
     smw# mkdir -p /tmp/netapp
     smw# cp ./SMIA-LINUXX64-11.10.0A00.0024.bin /tmp/netapp
     ```

3. Run the NetAPP SANtricity Storage Manager installer.

   ```
   smw# /tmp/netapp/SMIA-LINUXX64-11.10.0A00.0024.bin
   ```

   The **SANtricity Storage Manager Introduction** window displays. The following substeps provide guidance through the installation, but the exact steps may differ for newer versions of the NetApp software.

   a. Select **Next** in the **SANtricity Storage Manager Introduction** window.

      The **License Agreement** window displays.

   b. Select **I accept the terms of the License Agreement**, then select **Next**.

      The **Select Installation Type** window displays.

   c. Select **Typical (Full Installation)**, then select **Next**.

      The **Multi-Pathing Driver Warning** window displays.

   d. Select **OK**.

      The **Pre-Installation Summary** window displays.

   e. Select **Install**.

      The **Installing SANtricity** window displays and shows the installation progress. When the installation completes, an **Install Complete** window displays.

f. Select **Done** to acknowledge and finish.

The SANtricity client, `SMclient`, is installed in `/opt/SMgr/client`.

4. Enable crayadm to run `SMclient`.

To be able to execute `SMclient` from the `crayadm` account, change the ownership and permissions for the executable files. If this step is skipped, only the `root` account will be able to run `SMclient`.

```
smw# chown crayadm /opt/SMgr
smw# chmod 775 /opt/SMgr
smw# chmod 755 /opt/SMgr/client/SMcli /opt/SMgr/client/SMclient
smw# chown -R crayadm:crayadm /var/opt/SM
smw# chmod -R ug+w /var/opt/SM
```

4.3.3 Set Up Boot RAID Space for Direct-attached Lustre

If the system will use direct-attached Lustre (DAL), create LUNs for DAL nodes to use for the MGT, MDT, and OST disk devices. This must be done before installing CLE and DAL.

If creating LUNs on the NetApp 2700 boot RAID device or external Netapp block storage device, use the SANtricity data management software installed on the SMW to create the DAL LUNs.

4.3.4 Create Boot RAID Volume Group and Volumes for NetApp, Inc. devices

Prerequisites

The user must be logged on to the SMW as `crayadm` to perform this task.

About this task

Create the 8+2 Volume Group and 2 Global Hot Spares for a 4.5 TB Volume Group (the amount of storage for this installation may be different). Standard new boot RAID has 2 hot spares; the number of hot spares depends on the number of available drives left over after configuration of the 8+2 RAID6.

Procedure

1. Start the SANtricity Storage Manager.

```
crayadm@smw> /usr/bin/SMclient
```

The SANtricity Storage Manager window appears.

2. Select a method for adding a volume group.

If the **Select Addition Method** window appears, choose one of the following options. Otherwise, continue with the next step.

- **Automatic**. Select this option if a serial connection was not used to assign IP addresses to the storage array controllers. The SANtricity software automatically detects the available controllers, in-band, using the Fibre Channel link.
- **Manual**. Select this option if IP addresses have already been assigned to the storage array controllers.

3. Create a volume group.

   The following substeps apply only if the **Select Addition Method** window did not display or if the **Manual** option was selected.

   a. Double-click the name for the storage array to be configured.

      The **Array Management** window displays.

   b. Select the **Logical/Physical** tab.

   c. Right-click **Unconfigured Capacity** and select **Create Volume**.

      The **Create Volume** wizard displays.

   d. Select **Next** on the **Introduction (Create Volume)** window.

   e. Select the **Manual** option on the **Specify Volume Group (Create Volume)** window.

   f. Select tray 99, slots 1-10, then select **Add**.

   g. Verify that the RAID level is set to 6.

   h. Select **Calculate Capacity**.

   i. Select **Next** on the **Specify Volume Group (Create Volume)** window.

   The **Array Management** window should still be displayed after performing this step.

   Create and Configure Volumes

   After creating the first volume group, create the first volume when prompted. Configure the boot RAID with enough LUNs to support the various system management file systems (Cray recommends a minimum of three LUNs).

4. Create a volume.

   a. Enter a new volume capacity. Specify units as GB or MB.

   b. Enter a name for the volume.

   c. Select the **Customize Settings** option.

   d. Select **Next** in the **Specify Capacity/Name (Create Volume)** window.

   e. Verify the settings on the **Customize Advanced Volume Parameters (Create Volume)** window.

      These settings are used for the all of the LUNs.

      - For **Volume I/O characteristics type**, verify that **File System** is selected.
      - For **Preferred Controller Ownership**, verify that **Slot A** is selected. This places the LUN on the A Controller.

   f. Select **Next** in the **Customize Advanced Volume Parameters (Create Volume)** window.

   g. Select the **Default** mapping option in the **Specify Volume to LUN Mapping** window.

   h. For **Host** type, select **Linux** from the drop-down menu.

   i. Select **Finish** in the **Specify Volume to LUN Mapping** window.

j. Select **Yes** when prompted to create more LUNs in the **Creation Successful (Create Volume)** window, unless this is the last volume to be created. If this is the last volume, select **No** and continue with the next step (skipping the rest of these substeps).

k. Verify that **Free Capacity** is selected on **Volume Group 1 (RAID 5)** in the **Allocate Capacity (Create Volume)** window.

l. Select **Next** in the **Allocate Capacity (Create Volume)** window.

m. Repeat step 4 to create all of the volumes (applicable to this system) described in *Recommended Boot RAID LUN Values* on page 47

5. Indicate that volume creation and configuration is complete.

Select **OK** in the **Completed (Create Volume)** window.

6. Create a hot spare.

The hot spare provides a ready backup if any of the drives in the volume group fail.

a. Right-click on the last drive in the slot 12 icon on the right portion of the window and select **Hot Spare Coverage**.

b. Select the **Manually Assign Individual Drives** option.

c. Select **OK**.

d. Select **Close**.

7. Exit the tool.

The next step in the process is to zone the switches. Go to one of the following, depending on the switch type:

- *Zone the QLogic FC Switch* on page 52.
- *Zone the Brocade FC Switch* on page 55
- *Zone the LSI SAS Switch* on page 62

4.3.5 Zone the QLogic FC Switch

Prerequisites

This procedure assumes the following:

- The QLogic SANBox™ FC (Fibre Channel) switch has been configured and is on the HSS network.
- The disk device has four host ports connected to ports 0-3 for the QLogic SANbox switch, and the following connections have been made:
  - The SMW must be connected to port 10 on the SANBox.
  - The boot node must be connected to port 4 on the SANBox.
  - The SDB node must be connected to port 5 on the SANBox.

About this task

This procedure describes how to use the *QuickTools* utility to zone the LUNs on the QLogic SANBox FC switch. QuickTools is an application that is embedded in the QLogic switch and is accessible from a workstation browser with a compatible Java™ plug-in. It requires a Java browser plugin, version 1.4.2 or later.

Zoning is implemented by creating a *zone set*, adding one or more zones to the zone set, and selecting the ports to use in the zone.

> **NOTE:** If a LUN is to be shared between failover host pairs, each host must be given access to the LUN. The SMW host port should be given access to all LUNs.

Procedure

1. Start a web browser.
2. Enter the IP address of the switch.

   If the configuration has a single switch, the IP address is 10.1.0.250. The IP address of each RAID controller is preconfigured by Cray and is listed on a sticker on the back of the RAID controller.
3. Enter the login name and password when the **Add a New Fabric** window pops up and prompts for them.

   The default administrative login name is `admin`, and the default password is `password`.

   The QuickTools utility displays in the browser.
4. Select **Add Fabric**.

   If a dialog box appears stating that the request failed to connect over a secured connection, select **Yes** and continue.
5. Double-click the switch icon when the switch is located and displayed in the window.

   Information about the switch displays in the right panel.
6. Select the **Configured Zonesets** tab at the bottom of the panel.
7. Select **Zoning** and then **Edit Zoning** from the toolbar menu.

   The **Edit Zoning** window displays.
8. Create a zone set.

   a. Select the **Zone Set** button.

      The **Create a Zone Set** window displays.

   b. Create a new zone set.

      In this example, assume that the zone set is named `XT0`.
9. Create a zone.

   a. Right-click the `XT0` zone and select **Create a Zone**.

   b. Create a new zone named BOOT.

10. Define the ports in the zone.

On the right panel, select the button in front of BOOT to open a view of the domain members. Ports 0, 4, 5, and 10 are added to the BOOT zone. Define the ports in the zone to ensure that the discovery of LUNs is consistent among the SMW, the boot node, and the SDB node.

a. Using the mouse, left-click Port # 0 and drag it to the BOOT zone.

b. Using the mouse, left-click Port # 4 and drag it to the BOOT zone. This port is for the boot node.

c. Using the mouse, left-click Port # 5 and drag it to the BOOT zone. This port is for the SDB node.

d. Using the mouse, left-click Port # 10 and drag it to the BOOT zone. This port is for the SMW.

e. Select **Apply**. The error-checking window displays.

f. Select **Perform Error Check** when prompted.

g. Select **Save Zoning** after confirming that no errors were found.

11. Select **Yes** when prompted to activate a zone set, then select the appropriate `XT0` zone set.

At this point, Cray recommends creating a backup of the switch configuration (*Create a Backup of the QLogic Switch Configuration* on page 54) before closing and exiting the application. Otherwise, proceed to the next step in the boot RAID configuration process: *Reboot the SMW and Verify LUNs are Recognized* on page 65.

4.3.6 Create a Backup of the QLogic Switch Configuration

About this task

Use the QuickTools utility to create a backup of the QLogic switch configuration. To use QuickTools, a Java browser plugin, version 1.4.2 or later is required.

To start a web browser and open the QuickTools utility, complete steps 1 through 4. If the QuickTools utility is already open, skip to step 5.

Procedure

1. Start a web browser.

2. Enter the IP address of the switch.

The IP address of each RAID controller is preconfigured by Cray and is listed on a sticker on the back of the RAID controller. If the configuration has a single switch, the IP address is 10.1.0.250.

3. Enter the login name and password when the **Add a New Fabric** window pops up and prompts for them.

The default administrative login name is `admin`, and the default password is `password`.

The QuickTools utility displays in the browser.

4. Select **Add Fabric**.

If a dialog box appears stating that the request failed to connect over a secured connection, select **Yes** and continue.

The QuickTools utility opens.

5. Complete the configuration backup from within the QuickTools utility:

   a. Select **Switch** and then **Archive** from the top bar.

      A **Save** window pops up with blanks for **Save in:** and **File Name:**.

   b. Enter the directory (for example, `crayadm`) and a file name (for example, `sanbox_archive`) for saving the QLogic switch configuration.

   c. Select the **Save** button.

6. Close and exit the application.

The QLogic FC switch is now zoned and backed up. Proceed to the next step in the boot RAID configuration process: *Reboot the SMW and Verify LUNs are Recognized* on page 65.

4.3.7 Zone the Brocade FC Switch

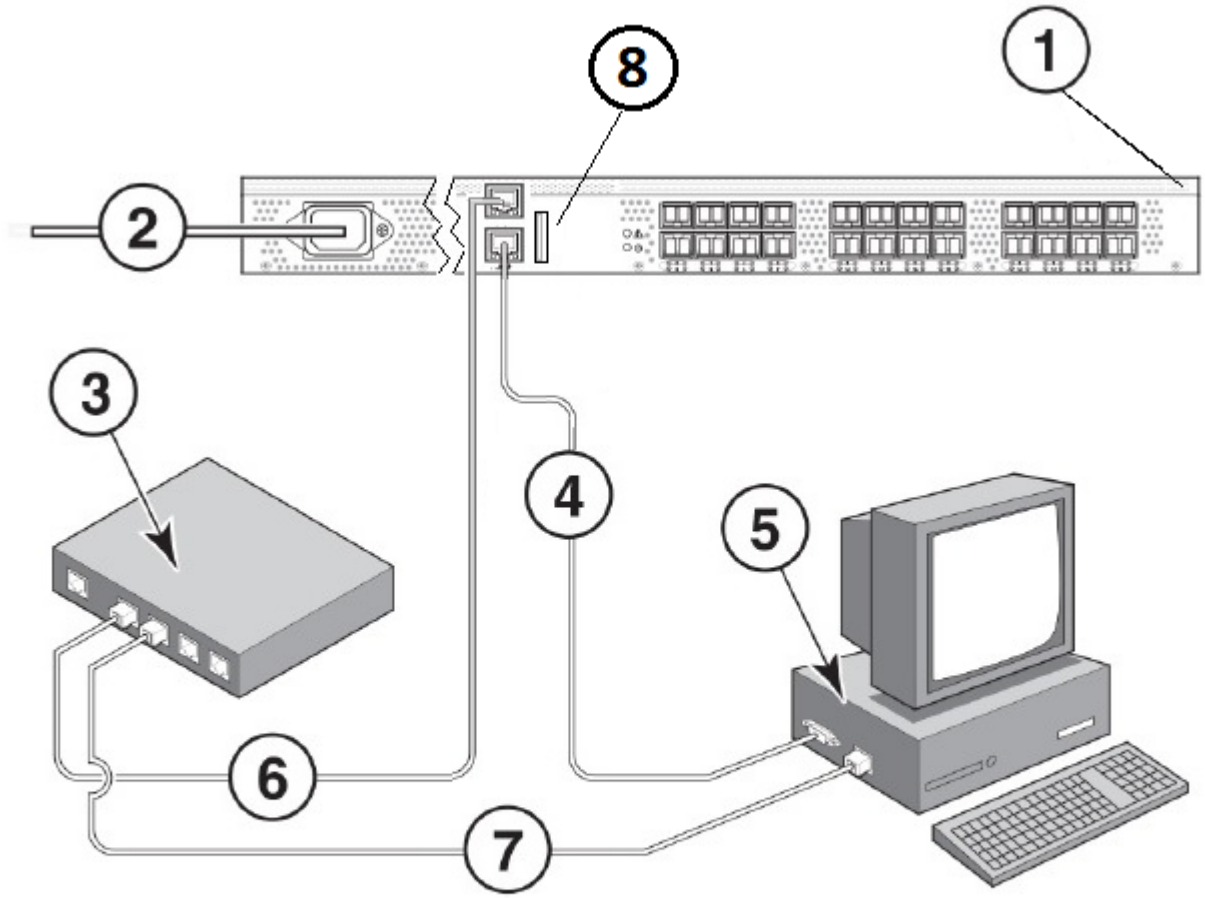
Prerequisites

This procedure assumes the following:

- The Brocade FC (Fibre Channel) switch has been configured and is on the HSS network.
- The connections shown in the figure have been made.

  NOTE: The SMW Ethernet port can be directly connected to the switch MGT port if no hub/switch (item 3 in figure) is available.

Figure 18. Brocade FC Switch Connections



1 Brocade switch	6 Ethernet cable from hub to Brocade switch
2 Power cable	7 Ethernet cable from setup computer to Ethernet hub or switch
3 Ethernet hub or switch	
4 Serial cable	8 **Brocade USB -requires special USB Drive**
5 Setup computer	

About this task

This procedure describes how to set up and use Web Tools, the embedded GUI application, to configure zoning of the Brocade 6505 Fibre Channel (FC) switch for standard boot RAID usage.

Procedure

1. Set up the GUI.

   a. Open a Firefox web browser on the SMW.

   ```
   crayadm@smw> cd
   crayadm@smw> firefox
   ```

   b. Enter the IP address of the switch (10.1.0.250) into the address bar.

A pop up window appears.

c. Select **Save File** to save the file in the default Downloads directory.

The Java plugin starts the GUI and a security warning appears, stating that the certificate is not trusted and the browser will not let the application continue.

d. Exit the browser.

e. Run `javaws -viewer` in an SMW xterm window.

```
crayadm@smw> cd
crayadm@smw> cd /Downloads
crayadm@smw> javaws -viewer
```

The **Java Control Panel** appears.

f. Add the switch to the **Exception Site List** in the **Security** tab.

Select the **Security** tab. Under **Exception Site List**, click **Edit Site List** and enter **`http://10.1.0.250`** as a trusted site, then click **Apply**.

g. Exit the `javaws -viewer` application.

h. Open `switchExplorer_installed.html` in an SMW xterm window.

Change directories to the `Downloads` directory, which is where the `switchExplorer_installed.html` file should be located (type **`ls`** to verify, if desired). Run `javaws -verbose switchExplorer_installed.html`.

```
crayadm@smw> cd /Downloads
crayadm@smw> javaws -verbose switchExplorer_installed.html
```

A **Verifying application** window appears. About three minutes later, a **Security Warning** window appears.

i. Select **I accept the risk and want to run this application**, then click **Run**.

A **Login** window appears.

j. Log in to the switch as `admin` with password `password`.

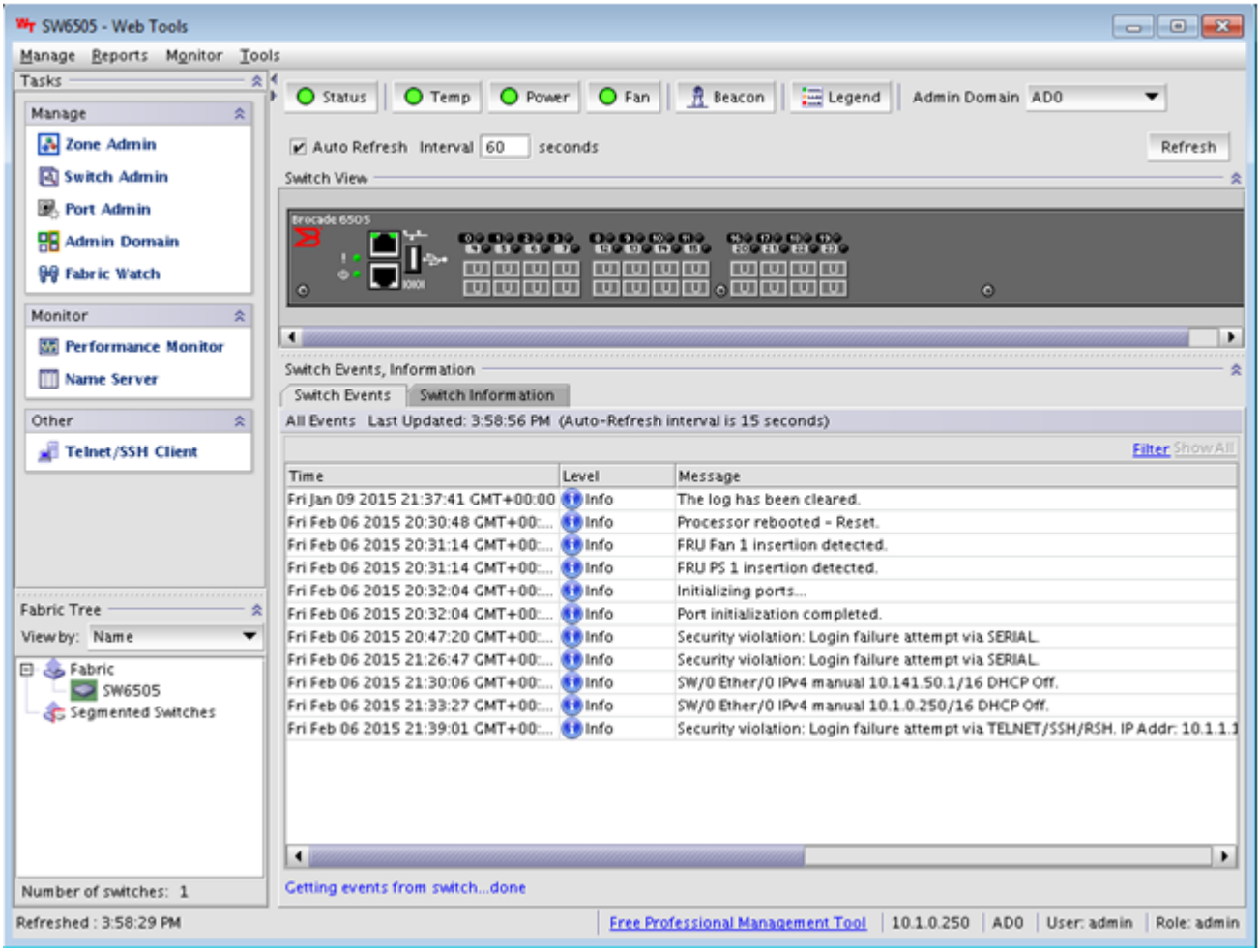
The main **Web Tools** switch window appears.

Use the Web Tools GUI to complete configuration of the Brocade FC switch.

2. Check the LED status of the tabs on the **Web Tools** window to ensure that there are no major issues.

The tab LEDs should all be green.

Figure 19. Brocade FC Switch Web Tools Window



3. Change the name of the switch.

a. Click **Switch Admin** in the **Manage** pane (upper left).

The **Switch Administration** window appears.

b. Change the name of the switch to BS#, where the # is the number of the switch being configured (e.g., BS01), then click **Apply** to save the name.

A confirmation window appears.

c. Click **Yes** to confirm, then close the **Switch Administration** window to return to the main **Web Tools** window.

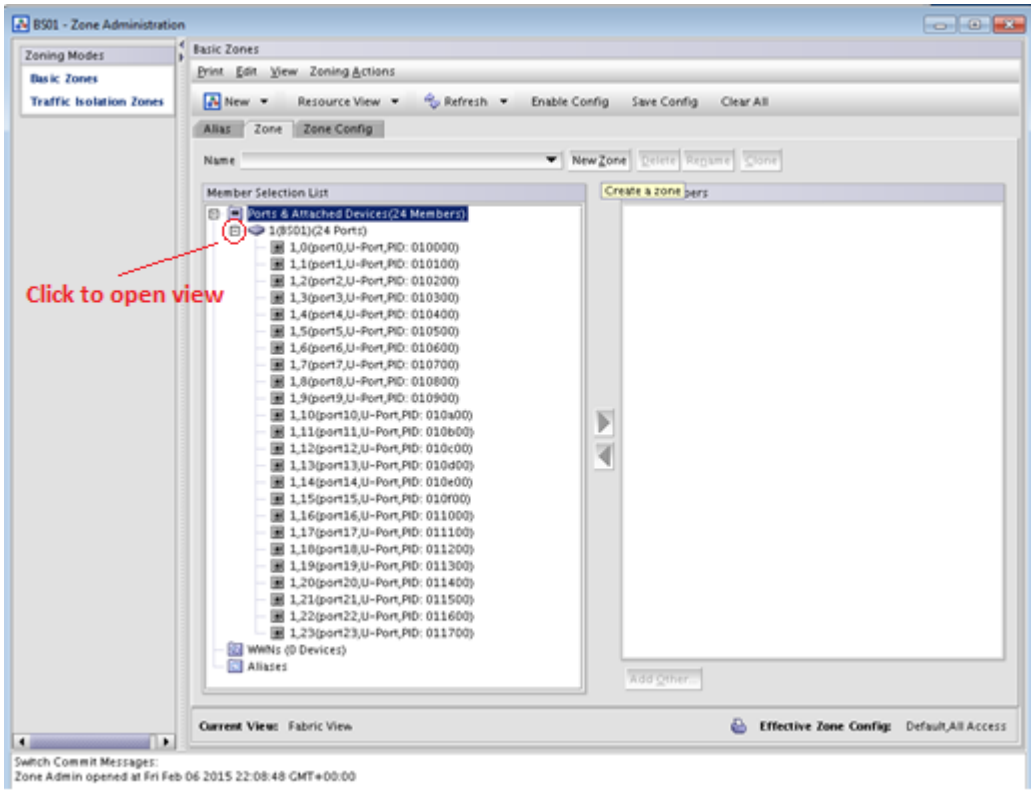
4. Set up a Boot zone for the switch.

a. Click **Zone Admin** in the **Manage** pane (upper left).

The **Zone Administration** window appears.

b. Click the **Zone** tab in the **Zone Administration** window, then click the **New Zone** button (located just below tabs, to the right).

Figure 20. Brocade FC Switch Zone Administration Window



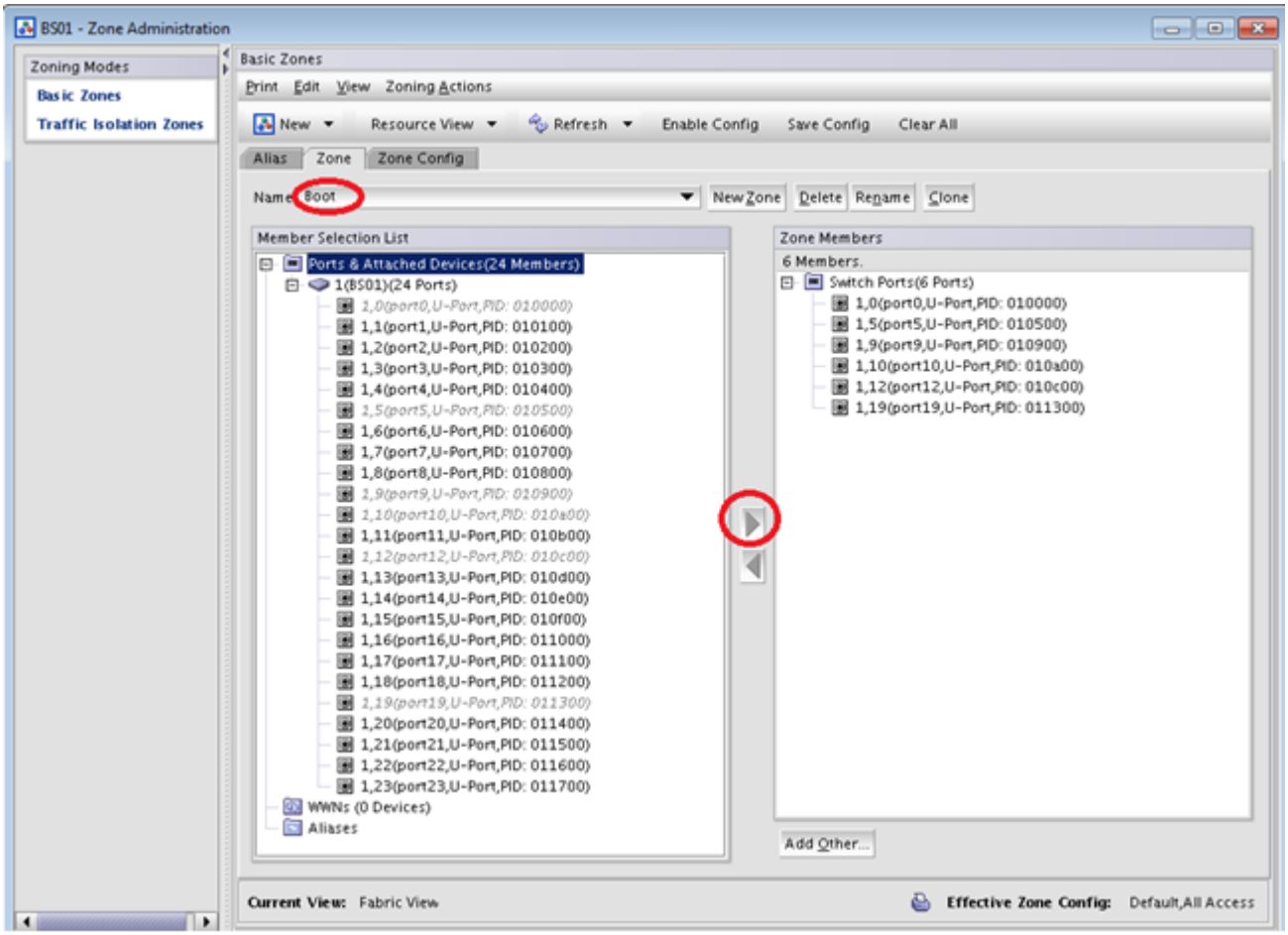
A **Create New Zone** window appears.

c. Enter the name of the new zone as "Boot."

"Boot" is displayed in the **Name** field in the **Zone** tab of the **Zone Administration** window.

d. Select port 0 in the **Member Selection List** (left pane in the **Zone** tab), then click the right arrow icon to move port 0 into **Zone Members** (right pane in the **Zone** tab). Repeat for ports 4, 5, 10, 11, 12, and 19.

Figure 21. Brocade FC Switch Boot Zone Members



Ports 0, 4, 5, 10, 11, 12, and 19 are in the **Zone Members** pane.

5. Configure the Boot zone.

a. Click the **Zone Config** tab in the **Zone Administration** window.

b. Click all of the **+** icons in the **Member Selection List** to expand all of the entries.

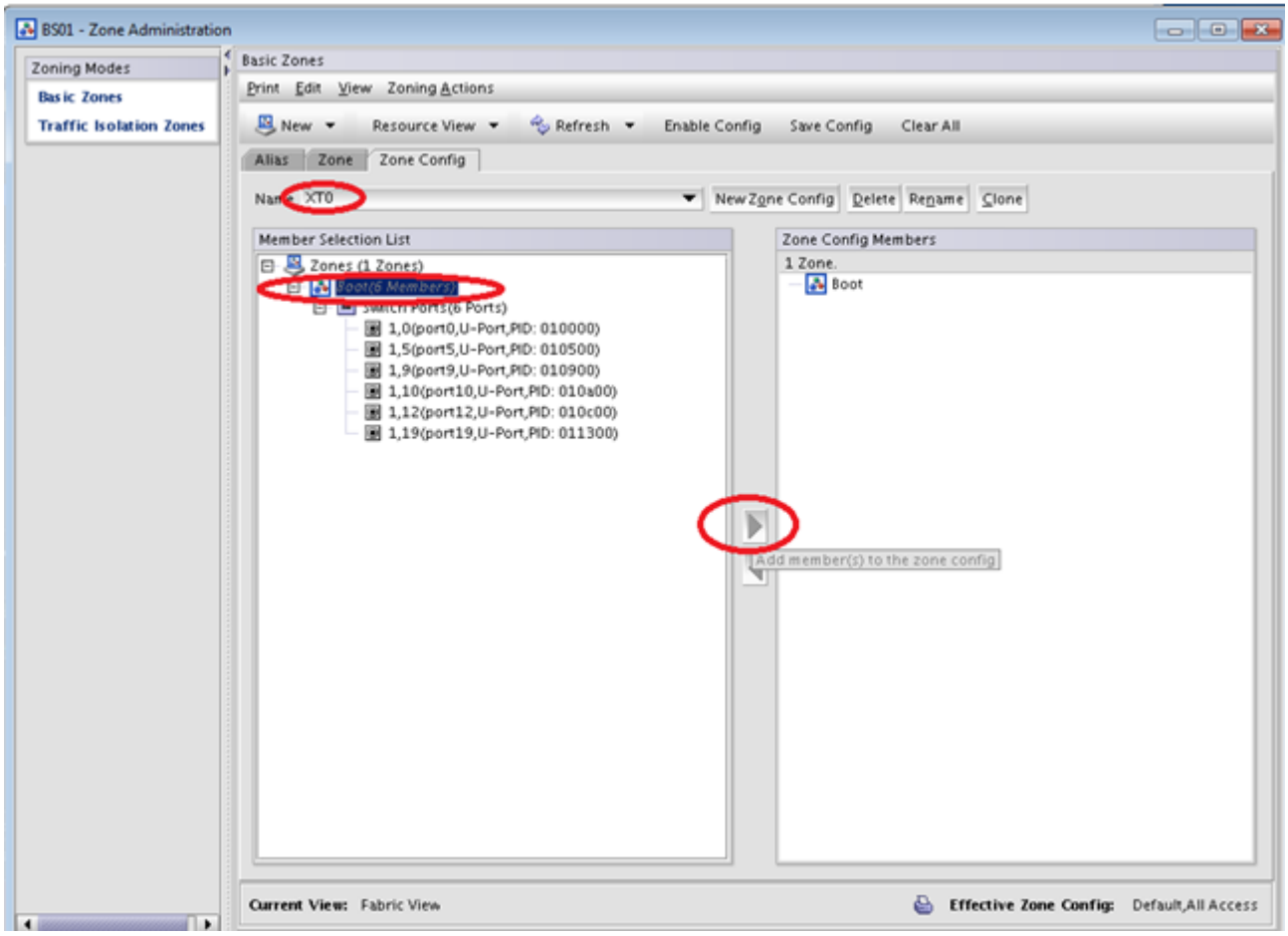
c. Click the **New Zone Config** button.

A **Create New Config** window appears.

d. Enter the name "XT0" in the **Create New Config** window, then click **OK**.

e. Select the Boot zone in the **Member Selection List** (left pane in the **Zone Config** tab), then click the right arrow icon to move the Boot zone into **Zone Config Members** (right pane in the **Zone Config** tab), which puts it in the XT0 group.

Figure 22. Brocade FC Switch XT0 Zone Config Members



f. Click the **Save Config** button (located just above tabs, to the right), then click **Yes** in the confirmation window that appears.

6. Enable the Boot zone configuration.

a. Click the **Enable Config** button (located just above tabs, to the right).

A **Choose Zone Config to be enabled** window appears.

b. Select the XT0 zone config from the menu, click **OK**, then click **Yes** in the **Enable Config XT0** confirmation window that appears.

At the bottom of the **Zone Administration** window, a status appears.

c. Click the **X** button at the top right to exit from the **Zone Administration** window when the status shows a "Commit succeeded" message, then click **Yes** in the exit confirmation window that appears.

The main **Web Tools** window appears.

7. Verify the configuration.

a. Remove power from the switch.

b. Re-apply power after 30 seconds, then wait for the Brocade FC switch to finish booting via the serial connection.

c. Enter the following from the serial connection:

```
BS01:> enable
BS01:> zoneshow

Defined configuration:
cfg:   XT0    boot
zone:  boot    1,0; 1,4; 1,5; 1,10; 1,11; 1,12; 1,19

Effective configuration:
cfg:   XT0
zone:  boot 1,0
            1,4
            1,5
            1,10
            1,11
            1,12
            1,19
```

d. Verify that the configuration matches the effective configuration.

The Brocade FC switch is now zoned. Proceed to the next step in the boot RAID configuration process: *Reboot the SMW and Verify LUNs are Recognized* on page 65.

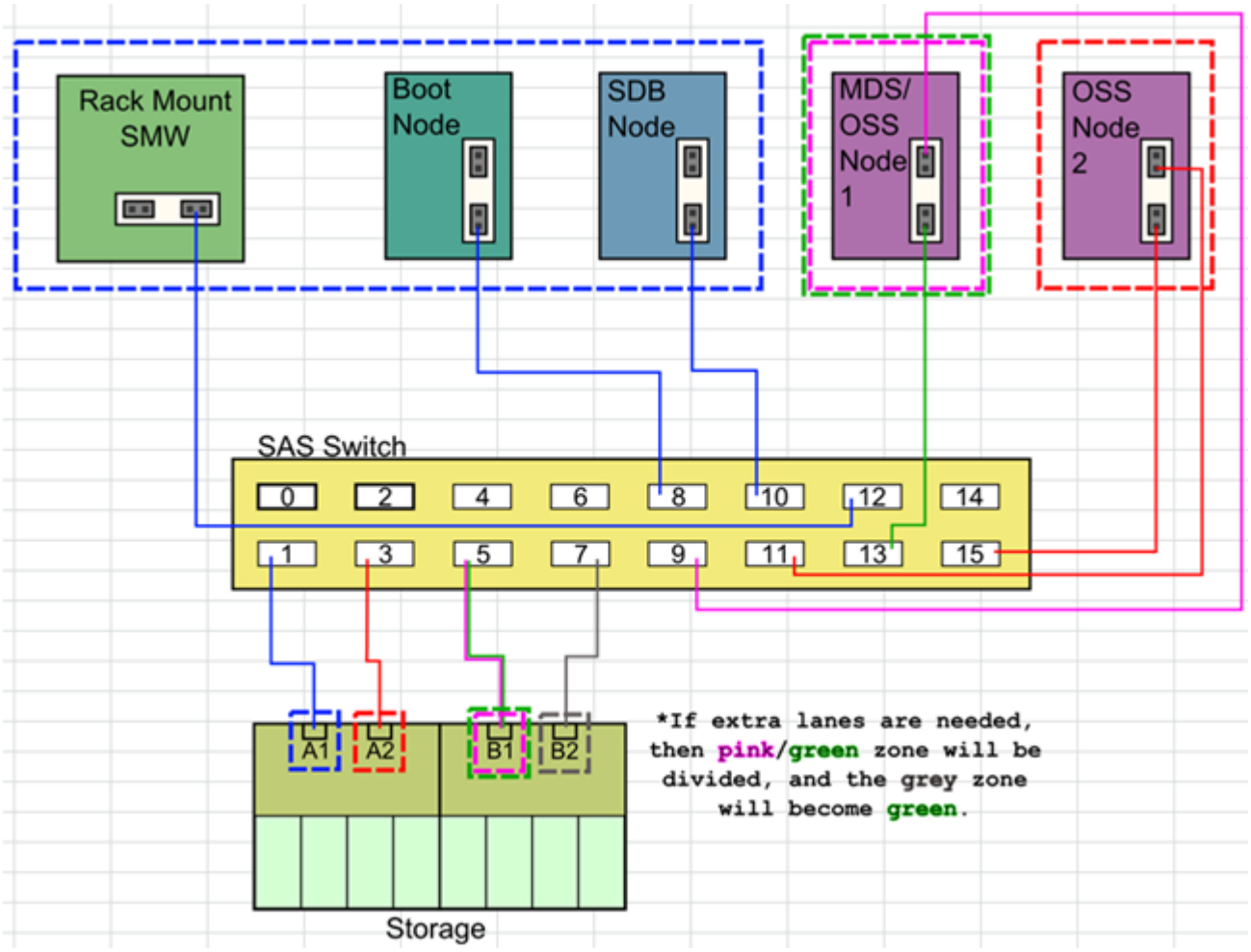
4.3.8 Zone the LSI SAS Switch

Prerequisites

This procedure assumes the following:

- The LSI 6160 SAS (Serial Attached SCSI) switch has been configured and is on the HSS network.
- The SMW is within 10 meters of the SAS switch (otherwise an FC switch is required).
- The following connections have been made (as shown in figure):
  - The SMW SAS card port 1 is connected by SAS cable to port 12 on the SAS switch.
  - The boot node is connected by SAS cable to port 8 on the SAS switch.
  - The SDB node is connected by SAS cable to port 10 on the SAS switch.

Figure 23. LSI SAS Switch Boot RAID Cable Connections



About this task

This procedure describes how to use SAS Domain Manager, the embedded GUI application, to zone the LUNs on the LSI (now Broadcom) 6160 SAS switch for standard boot RAID usage.

Procedure

1. Open the SAS Domain Manager GUI.

   a. Open a Firefox web browser on the SMW.

   ```
   crayadm@smw> cd
   crayadm@smw> firefox
   ```

   b. Enter the IP address of the switch (10.1.0.250) into the address bar.

   c. If Firefox displays the message "JRE 1.6 or higher required," do one of the following:

   - **Select Tools > Add-ons > Plugins** from the Firefox menu bar, then click the green button marked “enable” for Java to enable Java and allow the switch GUI to work.

- Run the following command from the command line to bypass Firefox and open the GUI.

  ```
  crayadm@smw> /usr/lib64/jvm/jre/bin/javaws http://10.1.0.250/sdmgui.jnlp
  ```

  When the **Opening sdmgui.jnlp** window appears, click OK to open the file, then in the **Warning - Security** window that appears, click **Run** to run the application.

The **SAS Domain Manager GUI** login window appears.

d. Log in to the switch as `admin` with password `admin`.

The **SAS Domain Manager GUI** main window appears.

2. Create the zone groups.

a. Click the **Domain** tab in the **SAS Domain Manager GUI** main window, then click **Create Zone Group**.

Create the following zone groups and assign the ports as indicated. Note that it is the phys values that should be mapped to the zone group, not the port(s).

Zone Group Name	SAS Ports	Phys
XT-A1-Storage	1	4,5,6,7
XT-A2-Storage	3	12,13,14,15
XT-B1-Storage	5	20,21,22,23
XT-B2-Storage	7	28,29,30,31
XT-SysRaid	8 10 12 4 6	32,33,34,35 40,41,42,43 48,49,50,51 16,17,18,19 24,25,26,27
XT-MDS-OSSn1(1)	9 13 14	36,37,38,39 52,53,54,55 56,57,58,59
XT-OSSn2(1)	11 15	44,45,46,47 60,61,62,63

(1) If an external Lustre server will be used instead of an internal Lustre server (DAL), then the XT-MDS-OSSn1 and XT-OSSn2 zones are not necessary.

3. Create the zone set.

a. Click the **Domain** tab in the **SAS Domain Manager GUI** main window, then click **Create Zone Set**.

Create the zone set XT0.

b. Assign the zone groups to the zone set by clicking the empty boxes to match this layout:

Zone Group Name	1	2	3	4	5	6	7
XT-A1-Storage					X		
XT-A2-Storage							X
XT-B1-Storage						X	
XT-B2-Storage						*X*(2)	
XT-SysRaid	X						
XT-MDS-OSSn1(1)			X	*X*(2)			
XT-OSSn2(1)		X					

(1) If an external Lustre server will be used instead of an internal Lustre server (DAL), then the XT-MDS-OSSn1 and XT-OSSn2 zones are not necessary.
(2) Setting a box marked with *X* may add more SAS lanes for performance, but do not set unless instructed to.

4. Activate the zone set.

   a. Click the **Domain** tab, then click **Activate/Deactivate Zone Set**.

   b. Select XT0 from the menu and enter the Zone Password of `lynx` to activate the zone set.

   c. Click the **Views** tab under **Active zone set** to verify that the zone groups and zone set are correct and active.

The SAS switch is now zoned. Proceed to the next step in the boot RAID configuration process: *Reboot the SMW and Verify LUNs are Recognized* on page 65.

4.3.9 Reboot the SMW and Verify LUNs are Recognized

About this task

Use this procedure to make the SMW rediscover the LUNs (logical unit numbers) and zones that were created.

Procedure

1. Log on as the `root` user.

```
crayadm@smw> su - root
```

2. Reboot the SMW to ensure that the LUNs are recognized.

```
smw# reboot
```

CAUTION: Failure to reboot the SMW at this point could produce unexpected results later on.

3. When the SMW has finished rebooting, log on as the `root` user.

```
crayadm@smw> su - root
```

4. Execute the `lsscsi` command to verify that the LUNs (volumes) have been rediscovered.

```
smw# lsscsi
```

5. List the disk devices by using the `fdisk` command to verify that the LUNs (volumes) are configured according to the boot LUN configuration table in *Recommended Boot RAID LUN Values* on page 47.

```
smw# fdisk -l
```

4.4 Make a Snapshot Manually

Prerequisites

This procedure assumes that the SLES12 base operating system has been installed on the SMW and boot RAID devices have been configured, but no other software has been installed yet.

About this task

A Btrfs snapshot of the SMW should be created immediately after SLES12 has been installed and before any files or directories have been modified by Cray's installation software or the rest of the installation process. With this snapshot, it will be possible to revert to this point if an initial/fresh install is repeated.

Snapshots are usually made using the `snaputil` program, but that program has not been installed at this point in the installation process. `snaputil` will be installed to the SMW with other Cray RPMs for the SMW and will be used for all Btrfs snapshot manipulations after this point.

Note that snapshots capture content in these three directories: `/`, `/var/lib/mysql`, and `/var/opt/cray/repos`. Used in conjunction with backups of config sets, they provide enough information to be able to re-create the state of the system at a particular time.

Procedure

1. Determine the root subvolume.

   It will be the string starting with "UUID." In this example it is "UUID=ffb0b613-2033-4835-87b5-6ca8ff1bacde."

```
smw# grep " / " /etc/fstab
UUID=ffb0b613-2033-4835-87b5-6ca8ff1bacde /            btrfs
defaults        0 0
```

2. Mount the root subvolume.

   Substitute the correct subvolume string for the example string shown in this command.

```
smw# mount -o subvol=@ UUID=ffb0b613-2033-4835-87b5-6ca8ff1bacde /mnt
```

3. Create a subvolume for snapshots (if `/mnt/snapshots` does not already exist).

```
smw# btrfs sub create /mnt/snapshots
```

4. Create the snapshot (if `/mnt/snapshots/SLES12` does not already exist).

   ```
   smw# btrfs sub snap / /mnt/snapshots/SLES12
   ```

5. Unmount the snapshot.

   ```
   smw# umount /mnt
   ```

6. Make a new `/media/root-sv` directory.

   ```
   smw# mkdir -p /media/root-sv
   ```

7. Mount root subvolume under `/media/root-sv` instead of `/mnt` as was used above.

   Substitute the correct subvolume string for the example string shown in this command.

   ```
   smw# mount -o subvol=@ UUID=ffb0b613-2033-4835-87b5-6ca8ff1bacde /media/root-sv
   ```

A "SLES12" snapshot has been made. Reboot to this snapshot whenever it is necessary to restart a fresh software installation from this point.

5 Install the SMW and CLE Software

To install the SMW and CLE software, use the following procedures in the order listed.

1. *Start a Typescript File* on page 68
2. *Prepare to Bootstrap the SMW Installation* on page 69
3. *Determine the Persistent Device Name for a LUN* on page 71
4. *RAID Disk Space Requirements* on page 72
5. *Bootstrap the SMW Installation* on page 74
6. *Provision SMW Storage* on page 82
7. *Run the Installer* on page 83
8. Think you know how to boot an SMW? Don't miss the extra, crucial step in this procedure: *Set Default Snapshot and Boot the SMW* on page 85

5.1 Start a Typescript File

About this task

Sites can make as few or as many typescripts as they deem useful. Cray recommends starting a typescript file at these software installation/configuration milestones:

- just before installing a new software release
- just before configuring the newly installed software

Procedure

1. Log in as root to the SMW.
2. (First time only) Create a release directory for the typescript file.

   ```
   smw# mkdir -p /var/adm/cray/release
   ```

3. Change to the release directory.

   ```
   smw# cd /var/adm/cray/release
   ```

4. Set a variable equal to today's date.

```
smw# export TODAY=`date +%Y%m%d`
smw# echo $TODAY
```

5. Start a typescript file.

```
smw# script -af ${TODAY}.suffix
```

   where *suffix* is some way to distinguish among typescript files (e.g., `install.1` or `install.2`).

6. Change prompt to include a timestamp.

```
smw# PS1="\u@\h:\w \t# "
```

5.2 Prepare to Bootstrap the SMW Installation

Prerequisites

This procedure assumes that the base operating system has been installed on the SMW and that the boot RAID has been set up.

About this task

The installer requires several ISO files to be available for setting up and installing packages from SLE repositories. The names of these ISOs are hard-coded in the installer configuration, but the containing directory can be anywhere that makes sense.

> **IMPORTANT:** The default location for these ISO files is `/root/isos`. The `--iso-dir` argument must be specified for `SMWinstall` if this is not the correct location for the ISO files on this system.

The Cray release distribution media consists of one DVD and several other pieces of media that may be on DVDs or furnished as ISO files. These ISO files are available for download at CrayPort (*https://crayport.cray.com*). To perform an initial installation of the SMW software, the first DVD to install is labeled Cray-SMWbase12-201511021655 and contains SUSE Linux Enterprise Server version 12 (SLES 12), the SMW base operating system.

Procedure

COLLECT ISO FILES

1. Make a directory on the SMW to hold the ISO files.

```
smw# mkdir -p /root/isos
```

2. Download the SLES 12 distribution ISOs to the new directory on the SMW.
   - `SLE-12-Module-Legacy-DVD-x86_64-GMC-CD1.iso`
   - `SLE-12-Module-Public-Cloud-DVD-x86_64-GMC2-DVD1.iso`
   - `SLE-12-SDK-DVD-x86_64-GM-DVD1.iso`
   - `SLE-12-Server-DVD-x86_64-GM-DVD1.iso`
   - `SLE-12-WE-DVD-x86_64-GM-DVD1.iso`

3. Download the CentOS 6.5 distribution ISO (`CentOS-6.5-x86_64-bin-DVD1.iso`) to the new directory on the SMW.

4. Download CLE 6.0 and SMW 8.0 SLES12 ISOs to the new directory on the SMW.
   - SMW release
     - `Cray-SMWbase12-201511021655 DVD`
     - `smw-8.0.xx-201605131428.iso`
   - CLE release
     - `cle-6.0.xx-201605131428.iso`

5. Download the SLES12 security updates ISO (`sleupdate-sle12sp0-201601271436.iso`) to the new directory on the SMW.

6. Make a directory on the SMW to hold any patches that may be available on CrayPort. See the release Errata documents for information about any day-1 patches.

```
smw# mkdir -p /var/adm/cray/release/patchsets
```

7. Download patches to the new patchset directory on the SMW.

   MOUNT MEDIA

8. Mount SMW media.

   a. Confirm that this is the right SMW media.

```
smw# ls -l /root/isos/smw*iso
-rw-r--r-- 1 root root  427184128 Nov  9 10:41 smw-8.0.19-201511122043.iso
```

   b. Set environment variables for the SMW media.

      Use the release string and the date-time stamp for the SMW media as the values for SMW_RELEASE and SMW_SOFTWARE, as shown in this example.

```
smw# export SMW_RELEASE=8.0.19
smw# echo $SMW_RELEASE

smw# export SMW_SOFTWARE=201511122043
smw# echo $SMW_SOFTWARE
```

   c. Mount the SMW release media.

```
smw# mkdir -p /media/SMW
smw# mount -o loop,ro /root/isos/smw-${SMW_RELEASE}-${SMW_SOFTWARE}.iso \
/media/SMW
```

9. Mount CLE media.

   a. Confirm that this is the right CLE media.

```
smw# ls -l /root/isos/cle*iso
-rw-r--r-- 1 root root 1146388480 Nov  9 09:22 cle-6.0.19-201511122043.iso
```

   b. Set environment variables for the CLE media.

Use the release string and the date-time stamp for the CLE media as the values for CLE_RELEASE and CLE_SOFTWARE, as shown in this example.

```
smw# export CLE_RELEASE=6.0.19
smw# echo $CLE_RELEASE

smw# export CLE_SOFTWARE=201511122043
smw# echo $CLE_SOFTWARE
```

c. Mount the CLE release media.

```
smw# mkdir -p /media/CLE
smw# mount -o loop,ro /root/isos/cle-${CLE_RELEASE}-${CLE_SOFTWARE}.iso \
/media/CLE
```

10. Set an environment variable for the SLES12 security updates media.

Use the date-time stamp for the SLES12 security updates media (`sleupdate-sle12sp0-201601271436.iso`) as the value for SLE_SOFTWARE, as shown in this example. This will be used when installing SMW and CLE software and SLES12 security updates together later in the process.

```
smw# export SLE_SOFTWARE=201601271436
smw# echo $SLE_SOFTWARE
```

INSTALL CONFIGURATION FILES

11. Copy `install.cle.conf`.

The `install.cle.conf` file contains configuration that controls the installer's image building behavior. Copy it from the CLE installation media and modify it if necessary. Copy `install.cle.conf.example` from the CLE media to `/var/adm/cray/install.cle.conf`.

```
smw# cp -p /media/CLE/products/cle/install.cle.conf.example \
/var/adm/cray/install.cle.conf
```

At this point there is nothing in this file that should be changed for a fresh install. Later this will be changed for updates to CLE.

12. Unmount CLE media.

```
smw# umount /media/CLE
```

5.3 Determine the Persistent Device Name for a LUN

About this task

After initial partitioning of the boot RAID, always address the storage via its persistent `/dev/disk/by-id/` name. Do not use the short `/dev/sd`*xx* name, which cannot uniquely identify the disk between reboots.

Use this procedure to determine the persistent device name from the LUN number on the boot RAID.

Procedure

1. Use `lsscsi` to show the `/dev/sd*` device name associated with a LUN number.

   In the first column of the output, the LUN is the final number in the `[n:n:n:n]` value. In this example, LUN 15 is associated with `/dev/sdo`.

   ```
   crayadm@smw1> lsscsi
   [0:0:0:0]    disk    ATA      TOSHIBA MK1661GS ME0D  /dev/sda
   [0:0:1:0]    disk    ATA      ST91000640NS     AA03  /dev/sdb
   [0:0:2:0]    disk    ATA      TOSHIBA MK1661GS ME0D  /dev/sdc
   .
   .
   .
   [5:0:0:15]   disk    LSI      INF-01-00        0786  /dev/sdo
   [5:0:0:16]   disk    LSI      INF-01-00        0786  /dev/sdp
   [5:0:0:17]   disk    LSI      INF-01-00        0786  /dev/sdq
   [5:0:0:18]   disk    LSI      INF-01-00        0786  /dev/sdr
   ```

   If multipathing is used, this command may show more than one line (device name) for a single LUN, making it difficult to know which is the correct one. In this case, try using the `SMdevices` command to see the volume labels assigned to each LUN. That aids in the process of matching the LUN to the Linux device and ensuring that the intended function of the LUN matches the volume name assigned using the SANtricity Storage Manager software.

2. Use `ls -l` to map the `/dev/sd*` device name to the persistent device name.

   To display the persistent device name for only one LUN, use `grep`. This example displays the persistent device name for `/dev/sdo` (that is, LUN 15).

   ```
   crayadm@smw1> ls -l /dev/disk/by-id | grep sdo
   lrwxrwxrwx 1 root root 10 Sep  4 00:56 scsi-360080e500037667a000003a2519e3ff2 -
   > ../../sdo
   lrwxrwxrwx 1 root root 10 Sep  4 00:56 wwn-0x60080e500037667a000003a2519e3ff2 -
   > ../../sdo
   ```

   There are two results for LUN 15. The one with prefix "scsi" is the one to use, so the persistent device name for LUN 15 is scsi-360080e500037667a000003a2519e3ff2.

3. Record the LUNs and corresponding persistent (by-id) device names for the following devices in preparation for bootstrapping the SMW installation.
   - Disk devices on the boot RAID that can be used for boot node persistent storage
   - Disk devices on the boot RAID that can be used for SDB node persistent storage
   - Disk devices on the boot RAID that can be used for SMW persistent storage

5.4 RAID Disk Space Requirements

The SMW, the boot node, and the SDB node all use space on the boot RAID. Here are the recommended sizes for the RAID LUNs, or LVM volume groups, based on the file systems for each. This information will be needed to bootstrap the SMW installation, which is next in the installation process.

SMW File Systems

On the boot RAID, the LVM volume group for the SMW will have the file systems listed in this table in the Mount Point column. The third column shows the recommended LUN size for each file system assuming a standard 4.5 TB RAID. For sites with storage constraints or extra storage, the fourth and fifth columns show suggested LUN sizes.

> **IMPORTANT:** The volume group for the `/var/opt/cray/imps` file system on the SMW should be significantly larger than the volume group for that file system on the boot node.

Table 5. SMW RAID Requirements

Mount Point	FS Type	Recommended Size for 4.5 TB RAID	Minimum Size	Size for 9.0 TB RAID	Description
`/home`	xfs	200 GB	40 GB	200 GB	Home directories on SMW
`/var/lib/mysql`	btrfs	10 GB	10 GB	10 GB	HSS database
`/var/opt/cray/disk/1`	xfs	1000 GB	400 GB	2000 GB	logs, debug, dumps
`/var/opt/cray/imps`	btrfs	1000 GB	400 GB	1000 GB	IMPS data
`/var/opt/cray/repos`	btrfs	200 GB	100 GB	200 GB	IMPS repos

CLE File Systems

On the boot RAID, storage for the boot node and SDB node is defined in the CLE storage set. Within that storage set, storage for the boot node is in the boot node LVM volume group, and storage for the SDB node is in the SDB node LVM volume group. The file systems for those nodes are listed in the tables below in the Mount Point column. The fourth column shows the recommended LUN size for each file system assuming a standard 4.5 TB RAID. For sites with storage constraints, the fifth column shows suggested LUN sizes.

Note that for partitioned systems, the requirements for LUN size apply to the boot node and SDB node in each partition.

Expanding storage space. The LUN sizes for the `/cray_home` and `/non_volatile` file systems may need to be adjusted depending on site usage of those file systems. For example, workload managers, DataWarp, and any node that needs permanent storage can store information in `/non_volatile`, so it may need to be larger than the suggested size. If size adjustment is not made at install time, it can be made later. See *XC Series System Administration Guide* for instructions on how to expand storage in a file system, volume, or volume group.

Table 6. Boot Node RAID Requirements

Owning Node	Mount Point	FS Type	Recommended Size for 4.5 TB RAID	Minimum Size	Description
boot	`/cray_home`	xfs	50 GB	50 GB	Home directories on CLE
boot	`/var/opt/cray/imps`	btrfs	250 GB	250 GB	IMPS data for PE image roots and for Netroot

Owning Node	Mount Point	FS Type	Recommended Size for 4.5 TB RAID	Minimum Size	Description
					compute-large and login-large image roots
boot	/non_volatile	xfs	200 GB	50 GB	persistent data, including /var if necessary, for service nodes provided from boot node

Table 7. SDB Node RAID Requirements

Owning Node	Mount Point	FS Type	Recommended Size for 4.5 TB RAID	Minimum Size	Description
sdb	/var/lib/mysql	xfs	20 GB	10 GB	SDB database
sdb	/alps_shared	xfs	20 GB	10 GB	ALPS data

5.5 Bootstrap the SMW Installation

Prerequisites

The following information must be gathered before running the installer in bootstrap mode. To find the persistent devices names for these devices, see *Determine the Persistent Device Name for a LUN* on page 71. For typical file system sizes, see *RAID Disk Space Requirements* on page 72.

- Disk devices on the boot RAID that can be used for boot node persistent storage
- Disk devices on the boot RAID that can be used for SDB node persistent storage
- Disk devices on the boot RAID that can be used for SMW persistent storage
- Size of file systems to be created within LVM volumes within LVM volume groups

  NOTE: Check that these file system sizes do not exceed the total size of the volume group containing them. Adjust file system sizes, if needed.

About this task

This procedure runs `SMWinstall` in bootstrap mode which installs IMPS and Ansible on the SMW, along with some of the global configuration templates. The `SMWinstall` command also invokes the configurator to prepare the storage set configuration. The configurator initiates an interactive session to gather the necessary information, unless the storage configuration template is supplied as a command-line argument, in which case no interactive session is needed. This configuration can be updated later by running the configurator manually.

Note that the `smw_postgres_vg` command is not supported with CLE 6.0.UP01 / SMW 8.0.UP01.

Configurator navigation hints:

- To get context-sensitive command help, enter **?**.
- To add a single value, enter the data and press **Enter**.
- To add a list, press the **+** key, enter each list item on its own line, and press **Ctrl-d** when finished entering the entire list of items.
- To correct an error in a previous setting, press the **<** key to go back to the previous setting, correct it, then continue forward. Use **<** to back up several settings, if needed.

Procedure

1. Install in bootstrap mode.

   - Method 1: Provide storage configuration information interactively.

     ```
     smw# /media/SMW/SMWinstall --mode bootstrap
     ```

   - Method 2: Provide storage configuration information using an existing storage configuration template:

     ```
     smw# /media/SMW/SMWinstall --mode bootstrap --storage-config \
     /path/to/cray_bootraid_config.yaml
     ```

   If Method 1 used, continue to step 2. If Method 2 used, skip to step 12.

2. Ensure that `cray_bootraid.enabled` is set to `true` to enable the storage service.

   CONFIGURE THE CLE DEFAULT STORAGE SET (`cledefault`) VOLUME GROUPS

   The configurator now shows the settings for a `storage_set` entry named `cledefault`, within which are two `volume_groups` entries: `boot_node_vg` and `sdb_node_vg`. The full name of these settings begins with `cray_bootraid.settings.storage_sets.data.cledefault.volume_groups.` followed by *<volume group name>.<field name>*. For brevity, the next steps show only the volume group name and field name of each setting.

3. Configure the boot node volume group (`boot_node_vg`).

   a. Set the owner of the boot node volume group.

      Ensure that `boot_node_vg.owner` is set to "boot" rather than a cname. For a partitioned system, include the partition name (e.g., "boot-p2" for partition p2).

   b. Add entries for the physical volumes (disk devices) that are going to be part of the boot node LVM volume group.

      This setting is a list. To add list data, enter **+** at the prompt for `boot_node_vg.devices` to enter list entry mode. Add persistent device names such as `/dev/disk/by-id/scsi-360080e50002f7160000014905640c0c4` for each physical volume. Do not use short names like `/dev/sdn`, which may vary from node to node (SMW, boot node, and SDB node), and may vary from boot to boot of a particular node.

      Press **Enter** after each list entry, and when done adding entries, press **Ctrl-d** to exit list entry mode. Remember to press **Enter** to set the list entries.

   c. Set these values for the volumes of the boot node volume group.

Ensure that these settings for the three `boot_node_vg` volumes (`home`, `imps`, and `nvolatile`) are set to "false." These settings pertain only to compute nodes with on-board SSDs. In future releases, these settings will not be presented for this volume group.

```
boot_node_vg.volumes.home.fs_remove_data: False
boot_node_vg.volumes.home.fs_cncu_enable: False

boot_node_vg.volumes.imps.fs_remove_data: False
boot_node_vg.volumes.imps.fs_cncu_enable: False

boot_node_vg.volumes.nvolatile.fs_remove_data: False
boot_node_vg.volumes.nvolatile.fs_cncu_enable: False
```

d. For each volume of the boot node volume group, change file system size to match the recommended values in the Boot Node RAID Requirements table in *RAID Disk Space Requirements* on page 72.

The `home` volume corresponds to the `/cray_home` file system in the table, the `imps` volume corresponds to `/var/opt/cray/imps`, and the `nvolatile` volume corresponds to `/non_volatile`.

At the following prompt, enter these values to get to the `fs_size` setting for each of the file systems.

```
cray_bootraid.settings.storage_sets.data.cledefault.volume_groups.boot_node_vg.volumes
[<cr>=set 3 entries, +=add an entry, ?=help, @=less] $
```

To get to this setting	enter this at prompt
`boot_node_vg.volumes.home.fs_size`	**1d***
`boot_node_vg.volumes.imps.fs_size`	**2d***
`boot_node_vg.volumes.nvolatile.fs_size`	**3d***

Then at the prompt for that setting, enter a new file system size to change the value, as needed.

e. When done with the last volume, press **Enter** to set the `boot_node_vg` "volumes" entries.

```
cray_bootraid.settings.storage_sets.data.cledefault.volume_groups.boot_node_vg.volumes
[<cr>=set 3 entries, +=add an entry, ?=help, @=less] $ <cr>
```

4. Configure the SDB node volume group (`sdb_node_vg.owner`).

a. Set the owner of the SDB node volume group.

Ensure that `sdb_node_vg.owner` is set to "sdb" rather than a cname. For a partitioned system, include the partition name (e.g., "sdb-p2" for partition p2).

b. Add entries for the physical volumes (disk devices) that are going to be part of the SDB node LVM volume group.

This setting is a list. To add list data, enter **+** at the prompt for `sdb_node_vg.devices` to enter list entry mode. Add persistent device names such as `/dev/disk/by-id/scsi-360080e50002f7160000014925640c108` for each physical volume. Do not use short names like `/dev/sdn`, which may vary from node to node (SMW, boot node, and SDB node), and may vary from boot to boot of a particular node.

Press **Enter** after each list entry, and when done adding entries, press **Ctrl-d** to exit list entry mode. Remember to press **Enter** to set the list entries.

c. Change these values for the volumes of the SDB node volume group.

Ensure that these settings for the two `sdb_node_vg` volumes (`db` and `alps`) are set to "false." These settings pertain only to compute nodes with on-board SSDs. In future releases, these settings will not be presented for this volume group.

```
sdb_node_vg.volumes.db.fs_remove_data: False
sdb_node_vg.volumes.db.fs_cncu_enable: False

sdb_node_vg.volumes.alps.fs_remove_data: False
sdb_node_vg.volumes.alps.fs_cncu_enable: False
```

d. For each volume of the SDB node volume group, change file system size to match the recommended values in the SDB Node RAID Requirements table in *RAID Disk Space Requirements* on page 72.

The `db` volume corresponds to the `/var/lib/mysql` file system in the table, and the `alps` volume corresponds to `/alps_shared`.

At the following prompt, enter these values to get to the `fs_size` setting for each of the file systems.

```
cray_bootraid.settings.storage_sets.data.cledefault.volume_groups.sdb_node_vg.volumes
[<cr>=set 2 entries, +=add an entry, ?=help, @=less] $
```

To get to this setting	enter this at prompt
`sdb_node_vg.volumes.db.fs_size`	**1d***
`sdb_node_vg.volumes.alps.fs_size`	**2d***

Then at the prompt for that setting, enter a new file system size to change the value, as needed.

e. When done with the last volume, press **Enter** to set the `sdb_node_vg` "volumes" entries.

```
cray_bootraid.settings.storage_sets.data.cledefault.volume_groups.sdb_node_vg.volumes
[<cr>=set 2 entries, +=add an entry, ?=help, @=less] $ <cr>
```

5. (Only for systems using compute nodes with SSDs) Change the compute node volume group (`compute_node_local`), as needed.

This is the third of three predefined volume groups in the `cledefault` storage set. It is needed only for systems using compute nodes with SSDs. For all other systems, skip this step and go to step 6.

a. Set the owner of the compute node volume group.

Set `compute_node_local.owner` to "compute." For a partitioned system, include the partition name (e.g., "compute-p2" for partition p2).

b. Add entries for the physical volumes (disk devices) that are going to be part of the compute node LVM volume group.

Add one or more entries to the `compute_node_local.devices` list. Use persistent device names such as `/dev/disk/by-id/scsi-360080e50002f7160000014925640c108`. Do not use short names like `/dev/sdn`, which may vary from node to node or from boot to boot of a particular node.

c. Change these values for the volumes of the compute node volume group.

Ensure that these settings for the three `compute_node_local` volumes (`temporary`, `swap`, and `unmanaged`) are set as follows:

```
compute_node_local.volumes.temporary.fs_remove_data: True
compute_node_local.volumes.temporary.fs_cncu_enable: True
```

```
compute_node_local.volumes.swap.fs_remove_data: False
compute_node_local.volumes.swap.fs_cncu_enable: False

compute_node_local.volumes.unmanaged.fs_remove_data: False
compute_node_local.volumes.unmanaged.fs_cncu_enable: False
```

Note that these settings are set to true for the temporary volume, while the other volumes' settings are set to false. Setting `fs_remove_data` to true causes all files to be removed from this file system at boot, while setting `fs_cncu_enable` to true enables CNCU (compute node cleanup) tracking of files on this file system, so that at the end of a job reservation, files created by the job will be removed. Both behaviors are appropriate for temporary file storage on compute nodes.

d. When done with the last volume, press **Enter** to set the `compute_node_local` "volumes" entries, or add another volume, as needed for this site.

```
cray_bootraid.settings.storage_sets.data.cledefault.volume_groups.compute_node_local.volumes
[<cr>=set 3 entries, +=add an entry, ?=help, @=less] $ <cr>
```

After completing this step, skip the next step. Do not remove the compute node volume group that has just been configured.

6. (Only for systems NOT using compute nodes with SSDs) Remove the compute node volume group (`compute_node_local`).

Perform this step only if this system does not use compute nodes with on-board SSDs.

a. Skip all of the compute node volume group settings.

Use the **>** key to skip each `compute_node_local` setting (including when prompted to set the three `compute_node_local` "volumes" entries).

```
compute_node_local.owner: >
compute_node_local.devices: >
compute_node_local.volumes.temporary.fs_remove_data: >
compute_node_local.volumes.temporary.fs_cncu_enable: >
compute_node_local.volumes.swap.fs_remove_data: >
compute_node_local.volumes.swap.fs_cncu_enable: >
compute_node_local.volumes.unmanaged.fs_remove_data: >
compute_node_local.volumes.unmanaged.fs_cncu_enable: >
```

```
cray_bootraid.settings.storage_sets.data.cledefault.volume_groups.compute_node_local.volumes
[<cr>=set 3 entries, +=add an entry, ?=help, @=less] $ >
```

b. Remove the compute node volume group.

The next prompt is to set the three `cledefault` volume group entries. At this prompt, remove the third entry, `compute_node_local`.

```
cray_bootraid.settings.storage_sets.data.cledefault.volume_groups
[<cr>=set 3 entries, +=add an entry, ?=help, @=less] $ 3-
```

Trouble? If `SMWinstall` fails during this step, it is because `cfgset` failed (which was invoked by `SMWinstall` to gather this configuration information), and that failure may be due to missing information. Do not try running `SMWinstall --mode bootstrap` again.

Try one of these options:

Option	Description

Run the configurator manually	**1.** Invoke the configurator in interactive mode. This example uses the default config set name "global." Substitute the site config set name, if different.
	`smw# cfgset update -m interactive -s cray_bootraid global`
	2. In the `cledefault` storage set, remove the `compute_node_local` volume group.
	`Boot RAID Configuration Menu [default: save & exit - Q] $ 1`
	`Boot RAID Configuration Menu [default: configure - C] $ C`
	`cray_bootraid.settings.storage_sets`
	`[<cr>=set 2 entries, +=add an entry, ?=help, @=less] $ 1a3-`
	3. Run the installer in bootstrap mode again.
	`smw# /media/SMW/SMWinstall --mode bootstrap`

Run `SMWinstall` in reconfigure mode	**1.** Run `SMWinstall` in reconfigure mode, which invokes the configurator in interactive mode.
	`smw# /media/SMW/SMWinstall --mode bootstrap --reconfigure`
	2. In the `cledefault` storage set, remove the `compute_node_local` volume group. This example begins at the point where the cray_bootraid service is being configured.
	`Boot RAID Configuration Menu [default: save & exit - Q] $ 1`
	`Boot RAID Configuration Menu [default: configure - C] $ C`
	`cray_bootraid.settings.storage_sets`
	`[<cr>=set 2 entries, +=add an entry, ?=help, @=less] $ 1a3-`

7. Set the `cledefault` "volume groups" entries.

```
cray_bootraid.settings.storage_sets.data.cledefault.volume_groups
[<cr>=set N entries, +=add an entry, ?=help, @=less] $ <cr>
```

(*N* is the number of volume groups defined.)

CONFIGURE THE SMW DEFAULT STORAGE SET (`smwdefault`) VOLUME GROUPS

The configurator now shows the settings for a `storage_set` entry named `smwdefault`, within which is a `volume_groups` entry named `smw_node_vg`. The full name of these settings begins with `cray_bootraid.settings.storage_sets.data.smwdefault.volume_groups.` followed by *<volume group name>.<field name>*. For brevity, the next steps show only the volume group name and field name of each setting.

8. Configure the SMW node volume group (`smw_node_vg`).

a. Set the owner of the SMW node volume group.

   Ensure that `smw_node_vg.owner` is set to "smw" or the hostname of the SMW, such as "orion-smw."

b. Add entries for the physical volumes (disk devices) that are going to be part of the SMW node LVM volume group.

   This setting is a list. To add list data, enter **+** at the prompt for `smw_node_vg.devices` to enter list entry mode. Add persistent device names such as `/dev/disk/by-id/scsi-360080e50002f889c00000a0654e32232` for each physical volume.

Do not use short names like `/dev/sdn`, which may vary from node to node (SMW, boot node, and SDB node), and may vary from boot to boot of a particular node.

Press **Enter** after each list entry, and when done adding entries, press **Ctrl-d** to exit list entry mode. Remember to press **Enter** to set the list entries.

c. Change these values for the volumes of the SMW node volume group.

These settings for five `smw_node_vg` volumes (`home`, `db`, `log`, `imps`, and `repos`) must be set to "false." These settings pertain only to compute nodes with on-board SSDs. In future releases, these settings will not be presented for this volume group.

```
smw_node_vg.volumes.home.fs_remove_data: False
smw_node_vg.volumes.home.fs_cncu_enable: False

smw_node_vg.volumes.db.fs_remove_data: False
smw_node_vg.volumes.db.fs_cncu_enable: False

smw_node_vg.volumes.log.fs_remove_data: False
smw_node_vg.volumes.log.fs_cncu_enable: False

smw_node_vg.volumes.imps.fs_remove_data: False
smw_node_vg.volumes.imps.fs_cncu_enable: False

smw_node_vg.volumes.repos.fs_remove_data: False
smw_node_vg.volumes.repos.fs_cncu_enable: False
```

d. For each volume of the SMW node volume group, change file system size to match the recommended values in the SMW RAID Requirements table in *RAID Disk Space Requirements* on page 72.

The `home` volume corresponds to the `/home` file system in the table, the `db` volume corresponds to `/var/lib/mysql`, the `log` volume corresponds to `/var/opt/cray/disk/1`, the `imps` volume corresponds to `/var/opt/cray/imps`, and the `repos` volume corresponds to `/var/opt/cray/repos`.

At the following prompt, enter these values to get to the `fs_size` setting for each of the file systems.

```
cray_bootraid.settings.storage_sets.data.cledefault.volume_groups.sdb_node_vg.volumes
[<cr>=set 2 entries, +=add an entry, ?=help, @=less] $
```

To get to this setting	enter this at prompt
`smw_node_vg.volumes.home.fs_size`	**1d***
`smw_node_vg.volumes.db.fs_size`	**2d***
`smw_node_vg.volumes.log.fs_size`	**33d***
`smw_node_vg.volumes.imps.fs_size`	**4d***
`smw_node_vg.volumes.repos.fs_size`	**5d***

Then at the prompt for that setting, enter a new file system size to change the value, as needed.

e. When done with the last volume, press **Enter** to set those `smw_node_vg` "volumes" entries.

```
cray_bootraid.settings.storage_sets.data.smwdefault.volume_groups.smw_node_vg.volumes
[<cr>=set 5 entries, +=add an entry, ?=help, @=less] $ <cr>
```

9. If there is a Postgres volume group (`smw_postgres_vg`), remove it from the `smwdefault` storage set.

The use of `smw_postgres_vg` is not supported with CLE 6.0.UP01.

a. Skip all of the Postgres volume group settings.

   Use the **>** key to skip each `smw_postgres_vg` setting (including when prompted to set the `smw_postgres_vg` "volumes" entries).

```
smw_postgres_vg.owner: >
smw_postgres_vg.devices: >
smw_postgres_vg.volumes.pgsql.fs_remove_data: >
csmw_postgres_vg.volumes.pgsql.fs_cncu_enable: >
```

```
cray_bootraid.settings.storage_sets.data.smwdefault.volume_groups.smw_postgres_vg.volumes
[<cr>=set 1 entries, +=add an entry, ?=help, @=less] $ >
```

b. Remove the Postgres volume group.

   The next prompt is to set the `smwdefault` volume group entries. At this prompt, remove the second entry, `smw_postgres_vg`.

```
cray_bootraid.settings.storage_sets.data.smwdefault.volume_groups
[<cr>=set 2 entries, +=add an entry, ?=help, @=less] $ 2-
```

10. Set the `smwdefault` "volume groups" entries.

```
cray_bootraid.settings.storage_sets.data.smwdefault.volume_groups
[<cr>=set 1 entries, +=add an entry, ?=help, @=less] $ <cr>
```

11. Set the boot RAID "storage sets" entries.

Review the storage sets. Press **Enter** (<cr>) to set the `cledefault` and `smwdefault` storage sets, unless this system has partitions. If configuring a partitioned system, enter **+** to add another CLE storage set. A separate storage set is needed for each partition.

```
cray_bootraid.settings.storage_sets
[<cr>=set 2 entries, +=add an entry, ?=help, @=less] $
```

12. (SMW HA only) Copy the storage configuration template.

If this is the primary/first SMW installed of an SMW HA pair, save the storage configuration template to another system not on this SMW for fast and consistent system bootstrapping when installing the secondary SMW.

```
smw# scp -p /var/opt/cray/imps/config/sets/global/config/\
cray_bootraid_config.yaml user@host:~/
```

13. Display `cray_bootraid` information.

```
smw# . /opt/modules/default/etc/modules.sh
smw# module use /opt/cray/ari/modulefiles
smw# module load imps
smw# cfgset search -s cray_bootraid -l basic global
smw# cfgset search -s cray_bootraid -l advanced global
```

14. Remove existing volume groups, as needed.

If doing a fresh install onto a system, and there is a desire to reuse the storage in any existing LVM volume groups for SMW, boot node, and SDB node, then run these commands to remove the volume groups with storage to be reused.

a. Use `cfgset search` to find the names of all of the volume groups defined in the storage configuration template.

```
smw# cfgset search -s cray_bootraid global |awk -F'.' '{print $7}' |sort -u
boot_node_vg
boottest1vg
sdb_node_vg
sdbtest1vg
smw_node_vg
smwtest1vg
```

b. Display the volume groups that exist.

```
smw# vgdisplay
```

Alternative (more concise):

```
smw# vgs
```

c. Remove the volume groups with storage to be reused (in this example, the test volume groups).

```
smw# vgremove -f smwtest1vg
smw# vgremove -f boottest1vg
smw# vgremove -f sdbtest1vg
```

The system is now ready for the provisioning of boot RAID LVM volumes.

5.6 Provision SMW Storage

About this task

The provision-storage mode of `SMWinstall` can be run at any time. It uses the boot RAID configuration template (cray_bootraid_config.yaml) to provision persistent storage on the boot RAID by creating LVM volume groups and LVM volumes. This is a non-interactive procedure if bootstrap mode has already been completed, which uses the configurator to gather the necessary site-specific configuration information.

Procedure

1. Provision storage for the default SMW storage set.

   Use this command only if using an SMW storage set called "smwdefault," which is the default.

   ```
   smw# /media/SMW/SMWinstall --mode=provision-storage
   ```

   Trouble? If errors are reported, try modifying the boot RAID configuration template using one of these methods, and then run the installer in provision-storage mode again. Note that both methods will ask ALL storage configuration questions, and the defaults will be prefilled with existing data.

   - Method 1: Modify using the configurator, then run installer again.

     ```
     smw# cfgset update -s cray_bootraid -m interactive global

     smw# /media/SMW/SMWinstall --mode=provision-storage
     ```

- Method 2: Modify manually, then run installer again.

```
smw# vi /var/opt/cray/imps/config/sets/global/config/\
cray_bootraid_config.yaml

smw# /media/SMW/SMWinstall --mode=provision-storage
```

2. View the new volumes.

```
smw# lvs
```

When the provision-storage installer mode completes successfully, the system is ready for the installation of SMW and CLE software.

5.7 Run the Installer

Prerequisites

This procedure assumes the following:

- A "target" snapshot was made earlier in the process.
- All of the SLES12 ISOs are in `/root/isos`.

About this task

This procedure installs SMW and CLE software together to ensure that there is a matched set of software and configuration.

Procedure

1. Set variable for snapshot name.

   Setting a variable here enables better command substitution in later commands dealing with snapshots. The SMW will boot from this snapshot.

   (SMW HA only) This is especially important for SMW HA systems because it makes it easier to use the exact same snapshot name for both SMWs. Using different snapshots results in HSS database (MySQL) inconsistencies, which causes problems at failover.

```
smw# ls -lst /root/isos
smw# export SNAPSHOT=smw-${SMW_RELEASE}_cle-${CLE_RELEASE}.${TODAY}
smw# echo $SNAPSHOT
```

2. Install SMW and CLE software and security updates together.

   It is possible to install both SMW media and CLE media with a single command to create a unified "release" that is tagged as a snapshot on the SMW system. Run the `SMWinstall` program and tell it where the CLE media is. This invocation installs into the "target" snapshot made earlier. It assumes all of the SLES12 ISOs are in `/root/isos`.

   IMPORTANT: The SLE media must be specified before the CLE media on the command line so that SUSE security updates are installed before the CLE software is installed.

```
smw# /media/SMW/SMWinstall \
--plus-media=/root/isos/sleupdate-sle12sp0-${SLE_SOFTWARE}.iso \
--plus-media=/root/isos/cle-${CLE_RELEASE}-${CLE_SOFTWARE}.iso \
--target=${SNAPSHOT}
```

It will take about 25 minutes to run a combined installation of SMW, CLE, and security updates for the first time on the SMW. Here is example output of a completed installation:

```
-----------------------------------------------------
The Cray software CLE 6.0UP01 (6.0.90) has been installed.
-----------------------------------------------------
To set this to be the default system at the next boot:
# /media/SMW/snaputil default SMW-8.0UP01_CLE-6.0UP01.20160503
To chroot to this snapshot and look around:
# /media/SMW/snaputil chroot SMW-8.0UP01_CLE-6.0UP01.20160503
To delete this SMW release and start over:
# /media/SMW/snaputil delete SMW-8.0UP01_CLE-6.0UP01.20160503
```

Each of the first two options suggested in the output ensure that the SMW is booted from the correct snapshot, which is essential to a successful reboot in the next part of the installation/configuration process.

Logs will be in `/var/adm/cray/logs/install` for each invocation of `SMWinstall`.

3. Check new snapshot software versions.

   When `SMWinstall` completes, check the snapshot details for the expected SMW and CLE release versions.

```
smw# /media/SMW/snaputil show ${SNAPSHOT}
active_maps     :
    p0:/var/opt/cray/imps/config/sets/global/nims/maps/p0
boot menu       : False
booted          : True
btrfs_object_id : 365
cle_version     : 6.0.96
created         : 2016-05-23 11:28:01
default         : True
initrd          : initrd-3.12.51-52.39-default
kernel          : vmlinuz-3.12.51-52.39-default
name            : smw-8.0.96_cle-6.0.96.20160523
parent*          : SLES12.May23
path            : /media/root-sv/snapshots/smw-8.0.96_cle-6.0.96.20160523
read-only       : False
smw_version     : 8.0.96
smwha_version   : None
storage_set     : smwdefault
subvolumes      :
    /var/lib/mysql:smw-8.0.96_cle-6.0.96.20160523
    /var/opt/cray/repos:smw-8.0.96_cle-6.0.96.20160523
total size      : 27658.24 MB
unshared size   : 1505.28 MB
updated         : 2016-05-23 11:51:01.188064
```

NOTE: If this is a subsequent fresh install instead of the very first fresh install, the "parent" entry (indicated by an asterisk in the preceding example), will look like this instead:

```
parent          : SLES12
```

The SMW is now ready to reboot, which starts with setting the default snapshot to boot from. Trying to boot the SMW without first setting the default snapshot will result in an unbootable SMW.

5.8 Set Default Snapshot and Boot the SMW

Prerequisites

This procedure assumes that the snapshot variable has been set and the SMW and CLE software has been installed.

About this task

When the `SMWinstall` command was invoked in the previous procedure, it provided several suggested `snaputil` commands that ensure that the snapshot target is set as the default snapshot for the next boot. This procedure uses one of the commands to ensure that the correct snapshot is used to boot the SMW.

Procedure

1. Set the release snapshot as the default.

   IMPORTANT: Do not skip this step. If the SMW is rebooted without first setting the default snapshot, the SMW becomes unbootable.

   ```
   smw# /media/SMW/snaputil default ${SNAPSHOT}
   ```

2. Reboot the SMW to switch to the new release.

   ```
   smw# reboot
   ```

6 Configure the SMW for the CLE System Hardware

In this part of the installation and configuration process it is necessary to initialize the power management database, discover hardware, create a config set, and boot the system.

6.1 Start a Typescript File

About this task

Sites can make as few or as many typescripts as they deem useful. Cray recommends starting a typescript file at these software installation/configuration milestones:

- just before installing a new software release
- just before configuring the newly installed software

Procedure

1. Log in as root to the SMW.
2. (First time only) Create a release directory for the typescript file.

   ```
   smw# mkdir -p /var/adm/cray/release
   ```

3. Change to the release directory.

   ```
   smw# cd /var/adm/cray/release
   ```

4. Set a variable equal to today's date.

   ```
   smw# export TODAY=`date +%Y%m%d`
   smw# echo $TODAY
   ```

5. Start a typescript file.

   ```
   smw# script -af ${TODAY}.suffix
   ```

   where *suffix* is some way to distinguish among typescript files (e.g., `install.1` or `install.2`).

6. Change prompt to include a timestamp.

   ```
   smw# PS1="\u@\h:\w \t# "
   ```

6.2 Set or Change the HSS Data Store (MariaDB) Root Password

About this task

The method for setting or changing the HSS data store (database) root password has changed with the release of CLE 6.0. By default, MariaDB is installed with no password set up for the root account. Cray strongly recommends adding a password as part of the fresh install procedure.

Old The HSS database was implemented with MySQL. After initial installation, its root password was changed from the initial default empty string to a user-defined value by the `SMWconfig` script, which was run after `SMWinstall` and the initial discovery of the system.

New The HSS database is now implemented with MariaDB, a MySQL work-alike database with identically named commands. As before, the initial default root password is the empty string; however, the `SMWconfig` script is no longer used to set it after installation. The administrator must use the following procedure to set the root password to a user-defined value.

Once the MariaDB root password has been set, it must be placed in `/root/.my.cnf`, a file readable only by root that has the format shown in step 2. This file serves as the mechanism by which the installer and the `snaputil` command obtain the root password when they access MariaDB as root. If the file does not exist or it has no `password=` line, the system will attempt to access MariaDB using the default empty-string password, which will fail once the password has been changed. The first time the root password is set to a user-defined value, `/root/.my.cnf` must be created. Afterwards, whenever the MariaDB root password is changed, `/root/.my.cnf` must be updated to match.

IMPORTANT: For an SMW HA system, record the new MySQL root password. It will need to be changed on the second SMW later (by editing `/root/.my.cnf`). After the SMW HA cluster has been configured, the MySQL root password needs to be reset with `mysqladmin` on only one SMW, because the MySQL database is shared between both SMWs in the HA cluster.

Procedure

1. Set or change the MariaDB root password.

```
smw# mysqladmin -uroot password -p
```

Do one of the following at the prompt:

- To **set** the root password for fresh installs or after the database has been reinitialized, press **Enter** to enter an empty string, the default initial password.

```
Enter password: <cr>
```

- To **change** the root password, enter the existing password.

```
Enter password: existing_password
```

At these prompts, enter the new root password, and then enter it again.

```
New password:
Confirm new password:
```

2. Ensure that the root password in the `/root/.my.cnf` file matches the new root password.

   If this file does not yet exist, create it and add the lines shown in the example, substituting the new password for the placeholder *MariaDB-password*.

   ```
   smw# vi /root/.my.cnf
   [client]
   user=root
   password=MariaDB-password
   ```

   Ensure that only root can see or write to this file.

   ```
   smw# chmod 600 /root/.my.cnf
   ```

6.3 Set Variable for Snapshot Name and Make a Snapshot Using snaputil

Prerequisites

This procedure assumes that the SMW has been booted to a release snapshot.

About this task

This procedure uses `snaputil` to make a snapshot. `snaputil` is a Python utility delivered with the cray-install-support RPM, which is installed by default on the SMW. It enables administrators and installers to create and manage snapshots as an easy way to stage upgrades and roll back or switch between SMW/HSS and CLE releases.

Note that snapshots capture content in these three directories: `/`, `/var/lib/mysql`, and `/var/opt/cray/repos`. Used in conjunction with backups of config sets, they provide enough information to be able to re-create the state of the system at a particular time.

Sites can make as few or as many snapshots as they deem useful. Cray recommends making a snapshot of the system at the software installation/configuration milestones shown in step 3.

Procedure

1. List the available snapshots on the system.

   ```
   smw# snaputil list
   ```

2. Set the `SNAPSHOT` environment variable using the currently booted snapshot name.

   Setting a variable for the snapshot name enables better command substitution in later commands dealing with snapshots. The SMW will boot from this snapshot. (SMW HA only) This is especially important for SMW HA systems because it makes it easier to use the exact same snapshot name for both SMWs. Using different snapshots results in HSS database (MySQL) inconsistencies, which causes problems at failover.

   ```
   smw# export SNAPSHOT=$(snaputil list |grep ^cur| awk '{print $2}')
   smw# echo $SNAPSHOT
   ```

3. Create a new snapshot.

```
smw# snaputil create ${SNAPSHOT}.suffix
```

Replace *suffix* with a unique suffix to distinguish among snapshots. Here are suggested suffixes:

Suffix	Associated Milestone
pre-config	after installing a new software release
post-config	after updating the global and CLE config sets
post-boot	after booting the system
post-customize	after customizing a preinstalled system

.

6.4 Update install.cle.conf for Software Updates

Prerequisites

This procedure assumes that the installer will not be run again at this point in the installation and configuration process.

About this task

The `/var/adm/cray/install.cle.conf` file contains configuration that controls the image building behavior of the installer. Changing this file now will make later updates of CLE software easier.

Procedure

1. Edit the configuration file.

```
smw# vi /var/adm/cray/install.cle.conf
```

2. (For all systems) Change `build_images` to `yes` to enable the CLE installer to build IMPS images as part of the install process. The remaining options determine what to do if `build_images` is set to `yes`.

```
build_images: yes
```

3. (For partitioned systems only) Uncomment the map_partition line and specify the system partitions.

```
map_partition:['p1', 'p2']
```

6.5 Prepare and Update the Global Config Set

Prerequisites

This procedure assumes that the SMW and CLE software has been installed so that the global config set is present.

About this task

The global config set must be updated with site-specific information about several services. This procedure describes how to add site configuration data to the configuration worksheets for each service in the global config set, update the config set with the edited configuration worksheets, and then run Ansible plays on the SMW to effect the changes.

When editing configuration worksheets, a general rule is to uncomment all settings that are marked level=basic. All settings that remain commented are considered unconfigured. Cray recommends not modifying settings that are already uncommented in the original worksheet; those preconfigured settings are needed for proper configuration of the system. For more information about editing configuration worksheets and updating config sets, see *XC™ Series Configurator User Guide*.

> **NOTE:** (SMW HA only) For SMW HA systems, the following procedures are done only on the first SMW because the config sets are shared between both SMWs in the HA cluster. In contrast, Ansible plays must be run on each SMW.

Procedure

1. Save a copy of original global worksheets.

   Copy the original configuration worksheets into a new directory to preserve them in case they are needed later for comparison.

   ```
   smw# ls -l /var/opt/cray/imps/config/sets/global/worksheets

   smw# cp -a /var/opt/cray/imps/config/sets/global/worksheets \
   /var/opt/cray/imps/config/sets/global/worksheets.orig
   ```

2. Make a work area for global worksheets.

   a. Copy the global configuration worksheets to a new work area for editing.

      The worksheets should not be edited in their original location for two reasons: (1) the configurator will not permit updating a config set from worksheets within that config set, and (2) edits would be overwritten when the config set is updated.

      ```
      smw# cp -a /var/opt/cray/imps/config/sets/global/worksheets \
      /var/adm/cray/release/global_worksheet_workarea
      ```

   b. Change to the work area directory to simplify the editing commands in the following steps.

      ```
      smw# cd /var/adm/cray/release/global_worksheet_workarea
      ```

   UPDATE WORKSHEETS FOR GLOBAL SERVICES

3. Update cray_firewall.

a. Edit `cray_firewall_worksheet.yaml`.

b. Uncomment `cray_firewall.enabled` and set it to `true`.

4. Update cray_global_net.

a. Edit `cray_global_net_worksheet.yaml`.

b. Uncomment `cray_global_net.enabled` and ensure that it is set to `true`.

c. Search in the file for `'networks' DATA`, then uncomment all of the lines below it that begin with `cray_global_net.settings.networks` so that those settings will be applied and marked as configured. They define four networks: "admin," "SMW failover," "HSS," and "management."

NOTE: Do NOT uncomment the similar lines under this heading, because they are examples only and are not configured for these four networks.

```
# ** EXAMPLE 'networks' VALUE (with current defaults) **
```

d. Enter SMW-specific or site-specific values for these management network items.

```
cray_global_net.settings.networks.data.management.ipv4_network:
cray_global_net.settings.networks.data.management.ipv4_netmask:
cray_global_net.settings.networks.data.management.ipv4_gateway:
cray_global_net.settings.networks.data.management.dns_servers:
cray_global_net.settings.networks.data.management.dns_search:
cray_global_net.settings.networks.data.management.ntp_servers:
```

Notes:

- The DNS information for `dns_servers` and `dns_search` was entered during the SLES12 installation, so those values can be found in `/etc/resolv.conf`.
- If entering a value for a setting that currently is set to `''` (empty string), remove the quotes before entering the new value. For example, `ipv4_network: ''` becomes `ipv4_network: 10.1.0.0`.
- If entering one or more values for a list setting that is currently set to `[]` (empty list), remove the brackets and add each entry on a separate line, beginning with "- " (a dash and a space). For example, the `dns_servers`, `dns_search`, and `ntp_servers` settings are lists that can have multiple entries, and they should look like this:

```
cray_global_net.settings.networks.data.management.dns_servers:
- 172.31.84.40
- 172.30.84.40
- 172.28.84.40
```

- Do NOT change or remove the null value in lines like this that appear at the beginning of each set of network definitions. This line sets the key for that network definition, which in this example is "management."

```
cray_global_net.settings.networks.data.name.management: null
```

e. Set the management network external firewall to true.

```
cray_global_net.settings.networks.data.management.fw_external: true
```

f. Search in the file for `'hosts' DATA`, then uncomment all of the lines that begin with `cray_global_net.settings.hosts` so that those settings will be applied and marked as configured.

They define a host called "primary_smw" and two interfaces for it: one that connects to the customer management network and one that connects to admin nodes, such as the boot and SDB nodes.

g. Enter SMW-specific or site-specific values for these items.

Note that if the IP address of the SMW changes, then the host ID, which is based on the IP address, needs to change also.

See step 4d for information about how to change empty string and empty list values.

```
cray_global_net.settings.hosts.data.primary_smw.aliases:
cray_global_net.settings.hosts.data.primary_smw.hostid:
cray_global_net.settings.hosts.data.primary_smw.hostname:
cray_global_net.settings.hosts.data.primary_smw.interfaces.customer_ethernet.
ipv4_address:
```

h. Set the `unmanaged_interface` field of the `customer_ethernet` and `admin_interface` interface settings to `true`.

This applies to both stand-alone SMWs and SMW HA systems. In the case of an SMW that is or will be configured for an SMW HA system, this prevents Ansible from managing eth0 and eth3 before the SMW HA cluster has been configured.

```
cray_global_net.settings.hosts.data.primary_smw.interfaces.customer_ethernet.unmanaged_interface:
 true
...
cray_global_net.settings.hosts.data.primary_smw.interfaces.admin_interface.unmanaged_interface:
true
```

5. Update cray_ipforward.

a. Edit `cray_ipforward_worksheet.yaml`.

b. Uncomment `cray_ipforward.enabled` and ensure that it is set to `true`.

6. Update cray_logging.

a. Edit `cray_logging_worksheet.yaml`.

b. Uncomment `cray_logging.enabled` and ensure that it is set to `true`.

c. Uncomment `cray_logging.settings.global_options.data.raid` and change its value if the boot RAID has a non-standard IP address.

d. Uncomment `cray_logging.settings.site_loghost.data.name` and change its value if the site has a site_loghost.

7. Update cray_multipath.

a. Edit `cray_multipath_worksheet.yaml`.

b. Uncomment `cray_multipath.enabled` and ensure that it is set to `false`.

Multipath should be enabled only when there is hardware cabling or zoning for multiple paths to the storage.

8. Update cray_time.

a. Edit `cray_time_worksheet.yaml`.

b. Uncomment `cray_time.enabled` and ensure that it is set to `true`.

c. Uncomment `cray_time.settings.service.data.timezone` and change its value, as needed.

d. Uncomment `cray_time.settings.ntp.data.servers` and add NTP servers to it.

The `ntp_servers` setting is a list that can have multiple values, each of which must go on a separate line, like this.

```
cray_time.settings.ntp.data.servers:
- ntpserver1
- ntpserver2
```

UPLOAD WORKSHEETS AND UPDATE/VALIDATE GLOBAL CONFIG SET

9. Upload modified worksheets into global config set.

Note that the full filepath must be specified in this `cfgset` command, and it must be enclosed in single quotes (to prevent the shell trying to expand the file glob).

```
smw# cfgset update -w \
'/var/adm/cray/release/global_worksheet_workarea/*_worksheet.yaml' global
```

10. Update the global config set.

Using the configurator in interactive mode to update the global config set is a good way to check whether all required settings and basic settings have been configured for services that are enabled. If they have, then all enabled services will show OK status in the Service Configuration List Menu. If configuration of a basic setting was missed, then the menu will show how many unconfigured settings there are for each service. Set or change any settings from this menu, as needed.

```
smw# cfgset update -m interactive global
```

When the configurator session completes, it displays a message indicating the file name of the changelog file for this configuration session. The changelog is written to a file in the `/var/opt/cray/imps/config/sets/global/changelog` directory.

11. Validate the global config set.

```
smw# cfgset validate global
```

APPLY CONFIGURATION CHANGES ON THE SMW

12. Run Ansible plays on the SMW.

After the global config set has been updated, reapply any Ansible plays that consume global config set data.

> **NOTE:** (SMW HA only) Both SMWs require this command. The procedure to install and configure the second SMW includes this command.

```
smw# /etc/init.d/cray-ansible start
```

Logs from running Ansible plays, such as cray-ansible, are stored on the SMW in `/var/opt/cray/log/ansible`.

CHECK TIME SETTINGS

13. Check for external NTP servers.

Check that external NTP servers have been set as desired in the global config set for cray_time.

NOTE: (SMW HA only) Both SMWs require this command. The procedure to install and configure the second SMW includes this command.

```
smw# grep server /etc/ntp.conf
server ntpserver1 minpoll 4 iburst
server ntpserver2 minpoll 4 iburst
```

14. Put the SMW time zone setting where the cabinet and blade controllers can access it.

This SMW time zone setting will be applied to the cabinet and blade controllers when they are rebooted in *Bootstrap Hardware Discovery* on page 95.

```
smw# cp /etc/localtime /opt/tftpboot/localtime
```

6.6 Prepare the CLE Configuration Worksheets

About this task

The Cray XC system stores configuration information used to boot and customize the CLE system in the p0 config set, or if the system is partitioned, in config set p1 for partition p1 and config set p2 for partition p2, and so forth. This procedure prepares the CLE configuration worksheets, which are later edited to include site-specific configuration data.

NOTE: (SMW HA only) For SMW HA systems, config set operations need to be performed on only one SMW because the config sets are shared between both SMWs in the SMW HA pair.

Procedure

1. Obtain configuration worksheets for CLE from one of these sources.

- Find them in the CLE 6.0.UP01 release directory available on CrayPort (`cle-0513-worksheets.tar`) and extract them to `/var/opt/cray/imps/config/sets/p0_example/worksheets`.
- Generate them by creating a CLE config set using prepare mode and the no-scripts option.

```
smw# cfgset create -m prepare -t cle --no-scripts p0_example
```

2. Save a copy of original worksheets.

Copy the original CLE configuration worksheets into a new directory to preserve them in case they are needed for comparison later.

```
smw# ls -l /var/opt/cray/imps/config/sets/p0_example/worksheets

smw# cp -a /var/opt/cray/imps/config/sets/p0_example/worksheets \
/var/opt/cray/imps/config/sets/p0_example/worksheets.orig
```

3. Make a work area for CLE worksheets.

Copy the CLE configuration worksheets to a new work area for editing. The worksheets should not be edited in their original location for two reasons: (1) the configurator will not permit updating a config set from worksheets within that config set, and (2) edits would be overwritten when the config set is updated.

REMEMBER: For partitioned systems, each partition generally has its own config set and associated configuration worksheets. Copy the CLE configuration worksheets to a separate work area for each partition.

```
smw# cp -a /var/opt/cray/imps/config/sets/p0_example/worksheets \
/var/adm/cray/release/p0_worksheet_workarea
```

These worksheets can be edited while the installation/configuration process continues with discovering hardware, flashing firmware, and other hardware or HSS software activities.

- To edit the worksheets, see *Update CLE Configuration Worksheets* on page 104, but do not proceed to the task that creates the new CLE config set from the worksheets until hardware discovery and associated procedures are complete.
- To continue with hardware discovery, proceed to *Bootstrap Hardware Discovery* on page 95.

6.7 Bootstrap Hardware Discovery

Prerequisites

This procedure assumes that the following information has been gathered. Enter this information in response to system prompts when performing this procedure.

Information needed	Default value
network type (Aries=a)	a
maximum X cabinet size (columns)	There is no default value. Find the X and Y cabinet sizes and the network topology class from *Site-dependent Configuration Values* in *Configuration Values* on page 19.
maximum Y cabinet size (rows)	No default value. See above.
network topology class	0 or 2 for Cray XC Series liquid-cooled systems, 0 for Cray XC Series air-cooled systems (XC30-AC, XC40-AC)
boot node name	c0-0c0s0n1
sdb node name	c0-0c0s1n1

About this task

This procedure uses the `xtdiscover --bootstrap` command to collect some basic information that will be used to bootstrap the hardware discovery process. If boot node failover or SDB node failover will be enabled, then when `xtdiscover` asks for the boot node or the SDB node, instead of entering a single node, enter a pair of nodes with a comma between them, for example "c0-0c0s0n1,c0-2c0s0n1." For more detailed information, see the `xtdiscover(8)` man page.

NOTE: (SMW HA only) Hardware discovery is done only on the first SMW. Do not repeat hardware discovery on the second SMW.

Trouble? If a step in this procedure fails because of a hardware issue, such as a cabinet failing to power up, resolve that issue and then go back to the last successful step in the procedure and continue from there. Do not skip steps or continue out of order.

Procedure

1. Run `xtdiscover` in bootstrap mode.

   ```
   smw# xtdiscover --bootstrap
   ```

   The system prompts the user to enter the information gathered as a prerequisite to this procedure. Prior to powering on the cabinets, the system prompts the user to disable any blades that should not be powered on.

   ```
   xtdiscover is about to power on the cabinets.
   *** IF YOU NEED TO DISABLE COMPONENTS TO AVOID THEM
   *** BEING POWERED ON, PLEASE DO SO NOW USING 'xtcli disable'

   Please enter 'c' to continue, or 'a' or 'q' to abort [c]:
   ```

2. Disable any blades that should not be powered on.

   If there are any blades or other components to be disabled, open a separate window and disable them (as `crayadm`) there. In this command, replace *`cname`* with the cname of the component to be disabled.

   ```
   crayadm@smw> xtcli disable cname
   ```

3. Return to the `xtdiscover --bootstrap` window and enter **`c`** to continue the hardware discovery bootstrap.

   ```
   Please enter 'c' to continue, or 'a' or 'q' to abort [c]: c
   ```

   The `xtdiscover` command proceeds without further prompts.

   Trouble? If the `xtdiscover` command fails with the message, `The following cabinets were not detected by heartbeat`, power cycle the cabinet controller and retry the `xtdiscover --bootstrap` command.

4. Power down the system.

   ```
   smw# xtcli power down s0
   Turning off power to cabinet and waiting for confirmation...

              Component|       Flags:    Result
   ------------------------------------------------------------------
                   c0-0|    noflags|:    Success
               c0-0c0s0|    noflags|:    Success
               c0-0c0s1|    noflags|:    Success
               c0-0c0s2|    noflags|:    Success
               c0-0c0s3|    noflags|:    Success
   ```

5. Reboot the cabinet controllers (CC), then verify that all CCs are up.

   a. Reboot the cabinet controllers.

      ```
      smw# xtccreboot -c all
      xtccreboot: reboot sent to specified CCs
      smw# sleep 180
      ```

b. Are all cabinet controllers up now? Repeat this command until all of the cabinet controllers report in.

```
smw# xtalive -a l1sysd -l l1 s0
The expected responses were received.
```

6. Reboot the blade controllers (BC), then verify that all BCs are up.

   This step ensures that any time zone changes on the SMW are applied to the BCs.

   a. Reboot the blade controllers.

   ```
   smw# xtccreboot -b all

   smw# sleep 180
   ```

   Repeat the `xtalive` command until all blade controllers are alive.

   b. Are all blade controllers up now? Repeat this command until all of the BCs report in.

   ```
   smw# xtalive s0
   ```

7. Power up the system.

```
smw# xtcli power up s0
Turning on power to cabinet and waiting for confirmation...

           Component|       Flags:    Result
--------------------------------------------------------------------
                c0-0|    noflags|:    Success
             c0-0c0s0|    noflags|:    Success
             c0-0c0s1|    noflags|:    Success
             c0-0c0s2|    noflags|:    Success
             c0-0c0s3|    noflags|:    Success
```

Note that at this point the `xtcli status` output shows that all nodes are "off" because they have not yet been bounced.

The bootstrap process is now complete. The next task is to discover the Cray system hardware.

6.8 Discover Hardware and HSN Routing, Prepare STONITH

Prerequisites

This procedure assumes that the `xtdiscover --bootstrap` command has been run successfully.

About this task

About Hardware Discovery. This procedure uses `xtdiscover` to detect the Cray system hardware components on the system. The `xtdiscover` command confirms some basic information (entered earlier with `xtdiscover --bootstrap`) for the hardware discovery process, warns that changes will be made, and then confirms whether to abort or continue. Finally, this command creates entries in the system database to describe the hardware. To display the configuration, use the `xtcli` command after running `xtdiscover`. For more detailed information, see the `xtdiscover(8)` man page.

About STONITH. This procedure prepares STONITH (shoot the other node in the head), a Linux service that automatically powers down a node that is not working correctly. If either boot node failover or SDB node failover will be used, then STONITH needs to be set on the primary blade.

> **IMPORTANT:** The primary boot node and primary SDB node should not be on the same blade. Likewise the secondary boot node and secondary SDB node should not be on the same blade. Four different blades should be used if there are two boot nodes and two SDB nodes.

Trouble? If a step in this procedure fails because of a hardware issue, such as a cabinet failing to power up, resolve that issue and then go back to the last successful step in the procedure and continue from there. Do not skip steps or continue out of order.

Procedure

DISCOVER CRAY SYSTEM HARDWARE

1. Log on to the SMW as `crayadm` and `su` to `root`.

```
crayadm@smw> su - root
smw#
```

2. Run the `xtdiscover` command.

```
smw# xtdiscover
```

3. Bounce the system (as `crayadm`) in a separate window.

```
crayadm@smw> /opt/cray/hss/default/etc/xtdiscover-bounce-cmd
```

4. After the `xtbounce` command from the previous step has finished, return to the `xtdiscover` window and enter "c" to continue the hardware discovery.

```
After bounce completes, enter 'c' to complete discovery
or 'q' or 'a' to abort [c]: c
```

5. Commit the results of `xtbounce` to the database.

   When asked whether to commit the `xtdiscover` results to the database, enter **`y`**.

PREPARE STONITH FOR BOOT NODE AND SDB NODE FAILOVER

6. Set STONITH for primary boot node's blade.

   For a primary boot node of c0-0c0s0n1, the blade is c0-0c0s0.

```
smw# xtdaemonconfig cX-XcXsX stonith=true
```

7. For sites using SDB failover, set STONITH for primary SDB node's blade.

   For a primary SDB node of c0-0c2s0n1, the blade is c0-0c2s0.

```
smw# xtdaemonconfig cX-XcXsX stonith=true
```

DISCOVER HSN ROUTING CONFIGURATION

8. Discover the routing configuration of the high-speed network (HSN).

   After `xtdiscover` finishes, run the `rtr` command as crayadm to determine the exact configuration of the HSN.

```
smw# su - crayadm
crayadm@smw> PS1="\u@\h:\w \t> "
crayadm@smw> rtr --discover
```

6.9 Update Firmware

Prerequisites

This procedure assumes that Cray hardware discovery has been completed successfully.

About this task

This procedure first checks whether the firmware of these components (controllers) needs to be updated, then updates the firmware only if there are Revision Mismatches.

all cabinet-level components

cc_mc (CC Microcontroller)
cc_bios (CC Tolapai BIOS)
cc_fpga (CC FPGA)
chia_fpga (CHIA FPGA)

all blade-level components

cbb_mc (CBB BC Microcontroller)
ibb_mc (IBB BC Microcontroller)
anc_mc (ANC BC Microcontroller)
bc_bios (BC Tolapai BIOS)
lod_fpga (LOD FPGA)
node_bios (Node BIOS)
loc_fpga (LOC FPGA)
qloc_fpga (QLOC FPGA)

Procedure

1. Check firmware.

   Check whether any firmware needs to be updated on the various controllers.

```
crayadm@smw> xtzap -r -v s0
```

   If the firmware on any controllers is out of date, the output looks like this, and the firmware needs to be updated (reflashed).

```
Individual Revision Mismatches:

Type       ID                Expected   Installed
---------- ----------------- ---------- ----------------------------------------
cc_bios    c0-0              0013       0012
bc_bios    c0-0c0s0          0013       0012
bc_bios    c0-0c0s1          0013       0012
```

```
bc_bios    c0-0c0s2          0013       0012
bc_bios    c0-0c0s3          0013       0012
```

2. Update firmware, if not all components are current.

   CAUTION: The `xtzap` command is normally intended for use by Cray Service personnel only. Improper use of this restricted command can cause serious damage to the computer system.

   To reflash the firmware, run `xtzap` with one or more of these options:

   `xtzap -a` Use to update all components.

   `xtzap -blade` Use when only blade types need to be updated.

   `xtzap -t` Use when only a single type needs to be updated. On larger systems, this can be a significant time savings.

   - If the output of checking the firmware shows that only a specific type needs to be updated, then use the `-t` option with that type. This example uses the node_bios type (a blade-level component).

     ```
     crayadm@smw> xtzap -t node_bios s0
     ```

   - If the output of checking the firmware shows that only blade component types need to be updated, then use the `-b` option.

     ```
     crayadm@smw> xtzap -b s0
     ```

   - If the output of checking the firmware shows that only cabinet component types need to be updated, then use the `-c` option.

     ```
     crayadm@smw> xtzap -c s0
     ```

   - If the output of checking the firmware shows that both blade- and cabinet-level component types need to be updated, or if unsure what needs to be updated, then use the `-a` option.

     ```
     crayadm@smw> xtzap -a s0
     ```

3. Perform `xtbounce --linktune`, if not all components are current.

   Force `xtbounce` to do a `linktune` on the full system before checking firmware again.

   ```
   crayadm@smw> xtbounce --linktune=all s0
   ```

4. Confirm that all components with out-of-date firmware have been updated.

   Check firmware again after updating and linktuning those components.

   ```
   crayadm@smw> xtzap -r -v s0
   ```

6.10 (Optional) Configure Partitions

About this task

This procedure describes how to divide the CLE system into "logical machines" or partitions. By definition, p0 is the entire system, and p1 through p31 are smaller partitions. Each partition must have its own set of boot, sdb, and other service nodes and compute nodes to boot the partition. See the xtcli_part(8) man page for more details.

> **NOTE:** (SMW HA only) For a partitioned SMW HA system, only the first SMW requires this procedure, because the hardware configuration is stored in a shared MariaDB (formerly MySQL) database.

To add a partition, specify the boot node, SDB node, and the components that will be members of the partition. As an example, the following steps show how to add these two partitions to an unpartitioned system (p0).

```
partition: p1
boot node: c0-0c0s0n1
sdb node: c0-0c0s1n1
members:
c0-0c0s0,c0-0c0s1,c0-0c0s4,c0-0c0s5,c0-0c0s6,c0-0c0s7,c0-0c0s8,c0-0c0s9,c0-0c0s10
,c0-0c0s11,c0-0c0s12,c0-0c0s15

partition: p2
boot node: c0-0c0s3n1
sdb node: c0-0c0s3n1
members: c0-0c0s2,c0-0c0s3,c0-0c0s13,c0-0c0s14
```

Procedure

1. Deactivate p0.

   ```
   smw# xtcli part_cfg deactivate p0
   ```

2. Add a partition.

   Note that `-b` identifies the boot node, `-d` identifies the SDB node, and `-m` identifies all members of the partition.

   ```
   smw# xtcli part_cfg add p1 -i /raw0 -b c0-0c0s0n1 -d c0-0c0s1n1 \
   -m c0-0c0s0,c0-0c0s1,c0-0c0s4,c0-0c0s5,c0-0c0s6,c0-0c0s7,\
   c0-0c0s8,c0-0c0s9,c0-0c0s10,c0-0c0s11,c0-0c0s12,c0-0c0s15
   ```

3. Activate the new partition.

   ```
   smw# xtcli part_cfg activate p1
   ```

4. Add and activate a second partition.

   ```
   smw# xtcli part_cfg add p2 -i /raw0 -b c0-0c0s3n1 -d c0-0c0s3n1 \
   -m c0-0c0s2,c0-0c0s3,c0-0c0s13,c0-0c0s14

   smw# xtcli part_cfg activate p2
   ```

6.11 Repurpose Compute Nodes

When a compute node is configured for a non-compute role, that node is a *repurposed compute node*. Compute nodes can be repurposed to become service nodes for use as tier2 servers (recommended) or in other capacities. Compute nodes should not be repurposed as service nodes for services that require external connectivity, such as dynamically shared libraries (DSL).

> **NOTE:** (SMW HA only) For SMW HA systems, perform this step only on the first SMW. This procedure is not required on the second SMW.

Use the `xtcli mark_node` command to repurpose a node in a compute blade. In this example, two compute nodes are being repurposed as service nodes and marked accordingly in the shared database.

```
crayadm@smw> xtcli mark_node service c0-0c0s2n0,c0-0c0s2n1
```

Note that service nodes can be repurposed as compute nodes as well. In that case, the command would be **`xtcli mark_node compute`**.

6.12 Finish Configuring the SMW for the CLE System Hardware

Prerequisites

This procedure assumes that Cray hardware has been discovered and component firmware has been updated (if needed).

About this task

This procedure contains the final steps of configuring the SMW for the CLE system hardware. Note that a full system is referred to as "s0" here. The term "p0" could have been used, because in this context, the two terms are interchangeable. In contrast, commands that operate on config sets use only the term "p0" when referring to a full system. In the config set context, the terms are not interchangeable.

Procedure

1. Check status on all components.

   full system For a full, unpartitioned system:

   ```
   crayadm@smw> xtcli status s0
   ```

   partitioned For a partitioned system, with partitions p1 and p2:

   ```
   crayadm@smw> xtcli status p1
   ```

   ```
   crayadm@smw> xtcli status p2
   ```

2. Check routing configuration of the system.

   full system For a full, unpartitioned system:

```
crayadm@smw> rtr -R s0
```

partitioned For a partitioned system, with partitions p1 and p2:

```
crayadm@smw> rtr -R p1
```

```
crayadm@smw> rtr -R p2
```

Note that the `rtr -R` command produces no output unless there is a routing problem.

3. Examine the hardware inventory and verify that all nodes are visible to the SMW.

full system For a full, unpartitioned system:

```
crayadm@smw> xthwinv s0 > xthwinv.out
```

```
crayadm@smw> xthwinv -x s0 > xthwinv.xml
```

partitioned For a partitioned system, with partitions p1 and p2:

```
crayadm@smw> xthwinv p1 > xthwinv.p1.out
crayadm@smw> xthwinv -x p1 > xthwinv.p1.xml
```

```
crayadm@smw> xthwinv p2 > xthwinv.p2.out
crayadm@smw> xthwinv -x p2 > xthwinv.p2.xml
```

4. Check microcontroller information.

Execute the `xtmcinfo -u` command to retrieve microcontroller information from cabinet control processors and blade control processors. Ensure that all blade controllers have output and show similar uptime values.

```
crayadm@smw> xtmcinfo -u s0
```

5. Exit from crayadm back to root account.

```
crayadm@smw> exit
smw#
```

7 Configure CLE

The CLE config set stores configuration information used to boot and customize the CLE system in the p0 config set for a full, unpartitioned system, or config set p1 for partition p1 and config set p2 for partition p2. This part of the installation and configuration process has procedures to update CLE configuration worksheets, create a new CLE config set from those worksheets, update the config set to set passwords that must be encrypted and to check that all basic and required configuration settings have been set, followed by the reconfiguration of a few services, config set validation, and making a snapshot.

1. *Update CLE Configuration Worksheets* on page 104
2. *Create New CLE Config Set from Worksheets* on page 146
3. *Update CLE Config Set* on page 146
4. Perform post-configuration activities.
   a. *Check CLE Hostnames in /etc/hosts File* on page 148
   b. *Update /etc/motd for Nodes* on page 148
   c. *Copy Files for External Lustre Fine-grained Routing* on page 149
   d. *Configure Files for Cray Simple Sync Service* on page 149
      - *About Simple Sync* on page 150
   e. *Display and Capture all Config Set Information* on page 154
   f. *Validate Config Sets* on page 155
   g. *Set Variable for Snapshot Name and Make a Snapshot Using snaputil* on page 88 (post-configuration)

      NOTE: (SMW HA only) For SMW HA systems, the following procedures are done only on the first SMW because the config sets are shared between both SMWs in the HA cluster. In contrast, Ansible plays must be run on each SMW.

7.1 Update CLE Configuration Worksheets

Prerequisites

This procedure assumes that the *Prepare the CLE Configuration Worksheets* on page 94 procedure has been performed, resulting in a set of CLE configuration worksheets that reside in a work area ready to be edited with site-specific configuration information.

About this task

The Cray XC system stores configuration information used to boot and customize the CLE system in the p0 config set, or if the system is partitioned, in config set p1 for partition p1 and config set p2 for partition p2, and so forth.

Use these procedures to edit the CLE configuration worksheets to include site-specific configuration data. Afterwards, these worksheets will be uploaded to the config set to create or update it.

When editing configuration worksheets, a general rule is to uncomment all settings that are marked level=basic. All settings that remain commented are considered unconfigured. Cray recommends not modifying settings that are already uncommented in the original worksheet; those preconfigured settings are needed for proper configuration of the system. For more information about editing configuration worksheets and updating config sets, see *XC™ Series Configurator User Guide*.

Change to the work area directory to simplify the editing commands in the following procedures.

```
smw# cd /var/adm/cray/release/p0_worksheet_workarea
smw# vi <service>_worksheet.yaml
```

REMEMBER: For partitioned systems, each partition generally has its own config set and associated configuration worksheets. Assuming a work area directory was created for each partition, change to that directory and update worksheets there for each partition.

NOTE: (SMW HA only) For SMW HA systems, the following procedures are done only on the first SMW because the config sets are shared between both SMWs in the HA cluster. In contrast, Ansible plays must be run on each SMW.

7.1.1 Update cray_alps Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray ALPS (Application Level Placement Scheduler) service configuration worksheet to add site-specific data. ALPS is a mechanism for placing, launching, and managing applications on Cray system compute nodes.

Procedure

1. Edit `cray_alps_worksheet.yaml`.

   ```
   smw# vi cray_alps_worksheet.yaml
   ```

2. Uncomment `cray_alps.enabled` and set it to `true`.

3. Uncomment `cray_alps.settings.common.data.xthostname` and set it to the name of this Cray system.

4. Uncomment `cray_alps.settings.common.data.alps_nodes` and add extra service nodes, if applicable.

   If there are service nodes (other than login nodes and the ALPS master node (the SDB node)) that need to run ALPS commands, add them to the list of `alps_nodes`. This list should include the workload manager (WLM) server and MOM (machine-oriented miniserver) nodes.

5. (Optional) If DRC (dynamic RDMA credentials) is to be used in a large system, uncomment `cray_alps.settings.apshed.data.pDomainMax` and set it to 256.

   If the maximum number of user protection domains is not increased from its default value of 10 to something like 256, DRC might exhaust all of the domains, which could cause problems.

6. Uncomment `cray_alps.settings.apsys.data.prologPath` and `cray_alps.settings.apsys.data.epilogPath`, even if they are assigned a null value.

7. (Optional) If RUR (resource utilization reporting) is to be used at this site, set the `prologPath` and `epilogPath` settings (from the previous step) to these paths.

   ```
   cray_alps.settings.apsys.data.prologPath: /opt/cray/rur/default/bin/rur_prologue.py
   cray_alps.settings.apsys.data.epilogPath: /opt/cray/rur/default/bin/rur_epilogue.py
   ```

   Also, ensure that the `cray_rur` service is enabled. See *Update cray_rur Worksheet* on page 138.

7.1.2 Update cray_auth Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray authentication domains service configuration worksheet to add site-specific data. Authentication domains (e.g., LDAP, Active Directory) are mechanisms for identifying and authenticating system users.

Procedure

1. Edit `cray_auth_worksheet.yaml`.

   ```
   smw# vi cray_auth_worksheet.yaml
   ```

2. Uncomment `cray_auth.enabled` and set it to `true`.

3. Configure these settings to connect to LDAP servers.

   In the worksheet, copy the four lines below `# ** EXAMPLE 'domain' VALUE (with current defaults) **` and paste them below `# NOTE: Place additional 'domain' setting entries here, if desired.`

   ```
   # ** EXAMPLE 'domain' VALUE  (with current defaults) **
   # cray_auth.settings.domain.data.reference.sample_key_a: null <-- setting a multival key
   # cray_auth.settings.domain.data.sample_key_a.servers: []
   # cray_auth.settings.domain.data.sample_key_a.schema: rfc2307bis
   # cray_auth.settings.domain.data.sample_key_a.aux_settings: []

   # ** 'domain' FIELD SPECIFICATION -- MULTIVAL KEY FIELD **
   ```

   Uncomment the lines, replace `sample_key_a` in all lines with some unique authentication domain identifier, and remove the `<-- setting a multival key` text at the end of the first line (note that the `null` value is

required; do not remove or change it). Finally, add values in accordance with site requirements. For settings that are lists, add each list element on a separate line prefixed by a hyphen and space (-).

```
# NOTE: Place additional 'domain' setting entries here, if desired.
# cray_auth.settings.domain.data.reference.<domain_name>: null
# cray_auth.settings.domain.data.<domain_name>.servers: []
# cray_auth.settings.domain.data.<domain_name>.schema: rfc2307bis
# cray_auth.settings.domain.data.<domain_name>.aux_settings: []
#************************ END Service Setting: domain ************************
```

4. Set the access policy.

   Uncomment `cray_auth.settings.access.data.policy`, if necessary, and set it to the list shown here. Note that the initial - character at the beginning of each list element is part of the YAML syntax; the access policy data, which begins with either a - or +, starts after that character.

   ```
   cray_auth.settings.access.data.policy:
   - +:root:LOCAL
   - +:crayadm:LOCAL
   ```

5. Uncomment `cray_auth.settings.access.data.config_computes` and set in accordance with site requirements.

6. Uncomment `cray_auth.settings.access.data.config_id_service`.

   Add the cnames of any nodes that might require this access, such as the DAL (direct-attached Lustre) nodes, if any. Add each node on a separate line prefixed by the - character followed by a space.

   ```
   cray_auth.settings.access.data.config_id_service:
   - c0-0c0s2n2
   - c0-0c1s2n2
   ```

7.1.3 Update cray_batchlimit Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray batchlimitd service configuration worksheet.

Procedure

1. Edit `cray_batchlimit_worksheet.yaml`.

2. Uncomment `cray_batchlimit.enabled` and set it to `false`.

   This disables this service. When a service is disabled, the values of all other settings in the service are ignored when configuration is applied. This worksheet contains additional configuration settings that are level=advanced, but this service and its advanced settings are not needed for a fresh install.

7.1.4 Update cray_ccm Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray CCM (cluster compatibility mode) service configuration worksheet. CCM enables users to run independent software vendor (ISV) applications without modification.

Procedure

1. Edit `cray_ccm_worksheet.yaml`.

   ```
   smw# vi cray_ccm_worksheet.yaml
   ```

2. Uncomment `cray_ccm.enabled` and set it to `false`.

   This service should be disabled until a workload manager (WLM) is installed.

   This disables this service. When a service is disabled, the values of all other settings in the service are ignored when configuration is applied. This worksheet contains additional configuration settings that are level=advanced, but this service and its advanced settings are not needed for a fresh install.

7.1.5 Update cray_cnat Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray CNAT (Compute Node Administrative Tool) service configuration worksheet to add site-specific data. CNAT is a mechanism for submitting and monitoring the execution of batch scripts; it requires a workload manager (WLM) to function.

Procedure

1. Edit `cray_cnat_worksheet.yaml`.

   ```
   smw# vi cray_cnat_worksheet.yaml
   ```

2. Uncomment `cray_cnat.enabled` and set it to `false`.

   This disables this service. When a service is disabled, the values of all other settings in the service are ignored when configuration is applied.

3. Uncomment `cray_cnat.settings.common.data.wlm`.

   When a WLM is installed later, the value of this setting will be set appropriately for this system and this service will be enabled automatically.

7.1.6 Update cray_drc Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray DRC (dynamic RDMA credentials) service configuration worksheet to add site-specific data. DRC configures dynamic RDMA (remote direct memory access) credentials, which are secure network credentials that can be shared between user applications to achieve intercommunication between applications running in different job reservations.

This service is disabled by default. For additional information about Cray DRC to help decide whether to enable it and know what configuration parameters are available, see *About Configuring Cray Dynamic RDMA Credentials (DRC)* on page 110. To enable and use this service, follow these steps.

NOTICE: Do not use DRC with VMDH (virtual memory domain handle).

Note that the database directory for DRC must be persistent `/var` to have persistent credentials and support node restart features. The server cname (step 3) and directory path (step 5) specified in this worksheet will be used in `cray_persistent_data_worksheet.yaml` to set up a client mount for persistent storage.

Procedure

1. Edit `cray_drc_worksheet.yaml`.

2. Uncomment `cray_drc.enabled` and set it to true.

3. Uncomment `cray_drc.settings.server.data.server_cname` and set it to the cname of the service node that should be running the DRC server.

   This cname will also be used as the value for the clients setting when preparing a persistent mount for DRC in the Cray persistent data worksheet (`cray_persistent_data_worksheet.yaml`).

4. Uncomment `cray_drc.settings.server.data.cookie_provider` and set it to one of these values:

   - `apmgr` if using an ALPS (Application Level Placement Scheduler) workload manager (WLM)
   - `ncmd` if using the native Slurm WLM

5. Uncomment `cray_drc.settings.server.data.database_directory` and set it to `/var/opt/cray/rdma-credentials`.

   This needs to be persistent `/var` so that the database can have persistent credentials and support node restart features. This is the same path name that will be used as the mount point when setting up a persistent

mount for DRC in `cray_persistent_data_worksheet.yaml` (see *Update cray_persistent_data Worksheet* on page 136).

6. Go back and uncomment the following settings, and set them in accordance with site preferences.

   Cray recommends configuring these settings so that diagnostic information is available if needed. Using persistent `/var` for the logging directories is best; however that depends on available storage space.

   ```
   cray_drc.settings.client.data.logging_directory
   cray_drc.settings.client.data.logging_filename
   cray_drc.settings.server.data.logging_directory
   cray_drc.settings.server.data.logging_filename
   ```

If this system uses ALPS, Cray recommends increasing the maximum number of user protection domains when DRC is in use, especially for large systems. That parameter is set in the Cray ALPS service with the `pDomainMax` field. See *Update cray_alps Worksheet* on page 105.

7.1.6.1 About Configuring Cray Dynamic RDMA Credentials (DRC)

Dynamic RDMA Credentials (DRC) is a new XC system service that enables shared network access between different user applications. DRC enables user applications to request managed network credentials, which can be shared with other users, groups, or jobs. Access to a credential is governed by the application and DRC to provide authorized and protected sharing of network access between applications. DRC extends the existing protection domain functionality provided by ALPS without exposing application data to unauthorized applications. DRC can also be used with other batch systems, such as Slurm, without any loss of functionality.

Trouble? Do not use DRC with VMDH (virtual memory domain handle). DRC does not use VMDH or limit its use; however, in a MAMU (multiple application multiple user) scenario, the use of VMDH by an application that is also using DRC could cause problems for other applications using VMDH on the same node, resulting in the failure of one or more of those processes.

When configuring Cray DRC, using the default values of the following settings will be sufficient for most cases. There are two required settings that must be configured with site-specific information however: server_cname and cookie_provider, which are both DRC server settings. Those must be assigned non-null values to complete the configuration process.

DRC Client (DRCC) Settings

None of the DRCC settings are required.

socket_location

Location of the DRCC UNIX domain socket. This location should allow read-write access for any user, because libDRC must be able to write to the socket to make any necessary requests.

Default value: `/tmp/drcc.sock`

logging_directory

Storage location for DRCC logs. This can be located anywhere convenient, as long as the directory is:

- (required) writeable by root
- (recommended) persistent between reboots so that the log file can be retrieved in a node-down event

Default value: `/tmp`

logging_filename	Name of the log file for DRCC. This name can be anything except a null value. Default value: `drcc.log`
logging_level	Verbosity of the DRCC logger. Possible values in order of increasing verbosity: `critical`, `error`, `warning`, `info`, and `debug`. Default value: `error`
requests_log_level	Verbosity of the python-requests logger. Possible values in order of increasing verbosity: `critical`, `error`, `warning`, `info`, and `debug`. Default value: `error`
llm_log_enabled	If enabled, DRCC will log messages to the lightweight log management (LLM) service. Default value: `true`
llm_log_level	Verbosity of DRCC log messages to the LLM service. Possible values in order of increasing verbosity: `critical`, `error`, `warning`, `info`, and `debug`. Default value: `error`

DRC Server (DRCS) Settings

Two of the DRCS settings are required: server_cname and cookie_provider.

server_cname (REQUIRED)	The cname of the node where DRCS will reside (e.g., c0-0c1s4n0). DRCS can reside on a login node or any unspecialized service node, but NOT on any boot or SDB nodes. This cname is also used as the value for the clients setting when preparing a persistent mount for DRC in `cray_persistent_data_worksheet.yaml` (see the database_directory setting below). Because this is a required field and no default is provided, a value must be entered.
logging_directory	Storage location for DRCS logs. This can be located anywhere convenient as long as the directory is: • (required) writeable by root • (recommended) persistent between reboots so that the log file can be retrieved in a node-down event Default value: `/tmp`
logging_filename	Name of the log file for DRCS. This name can be anything except a null value. Default value: `drcc.log`
logging_level	Verbosity of the DRCS logger. Possible values in order of increasing verbosity: `critical`, `error`, `warning`, `info`, and `debug`. Default value: `error`
port	TCP port on which the DRC server will listen to requests. Do not assign this port to any other TCP service. Default value: `4000`
use_ssl	Should the DRCS server use SSL? This additional layer of security is not necessary but is recommended. Default value: `true`

rpc_uri
Remote procedure call (RPC) URI used by both client and server to correctly address DRCS services.
Default value: `json-rpc`

werkzeug_log_level
Verbosity of the python-werkzeug logger. Possible values in order of increasing verbosity: `critical`, `error`, `warning`, `info`, and `debug`.
Default value: `error`

jsonrpc_log_level
Verbosity of the python-jsonrpc logger. Possible values in order of increasing verbosity: `critical`, `error`, `warning`, `info`, and `debug`.
Default value: `error`

requests_log_level
Verbosity of the python-requests logger. Possible values in order of increasing verbosity: `critical`, `error`, `warning`, `info`, and `debug`.
Default value: `error`

authorized_uids
List of UIDs that are allowed to interface directly with DRCS through DRCC, DRCCLI, and DRCJEDi (DRC job expiration director). If DRCC, DRCCLI, or DRCJEDi is run under a UID that is not in this list, any request made by that user will be rejected.
Default value: `['0']`

admin_uids
List of UIDs that are allowed to run DRCCLI. At present, this is limited to the values in the authorized_uids list.
Default value: `['0']`

cookie_provider (REQUIRED)
A string that indicates which workload manager binary DRCS should contact for cookies. Possible values:

- **`apmgr`** For systems running ALPS (Application Level Placement Service)
- **`ncmd`** For systems running Slurm (the native workload manager)

Because this is a required field and no default is provided, a value must be entered.

llm_log_enabled
If enabled, DRCS will log messages to the lightweight log management (LLM) service.
Default value: `true`

llm_log_level
Verbosity of DRCS log messages to the LLM service. Possible values in order of increasing verbosity: `critical`, `error`, `warning`, `info`, and `debug`.
Default value: `error`

database_directory
Storage location for the credential database. This needs to be persistent `/var/` so that the database can support node restart features. This is the same path name used in `cray_persistent_data_worksheet.yaml` to set up a persistent mount for DRC (see *Update cray_persistent_data Worksheet* on page 136).
Default value: `/var/opt/cray/rdma-credentials`

database_filename
Name of the credential database file.
Default value: `drc.db`

7.1.7 Update cray_dvs Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray DVS (Data Virtualization Service) service configuration worksheet to add site-specific data. DVS is a distributed network service that projects local file systems resident on I/O nodes or remote file servers to compute and service nodes within the Cray system.

Procedure

1. Edit `cray_dvs_worksheet.yaml`.

   ```
   smw# vi cray_dvs_worksheet.yaml
   ```

2. Uncomment `cray_dvs.enabled` and set it to `true`.

The configuration in this procedure is sufficient for a fresh install; for information about configuring DVS for other purposes, see *XC™ Series DVS Administration Guide*.

DVS uses the LNet (Lustre networking) networking layer, so ensure that cray_lnet is enabled as well. See *Update cray_lnet Worksheet* on page 117.

7.1.8 Update cray_dws Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

The Cray DataWarp Service (DWS) provides access to SSD (solid state device) storage for high bandwidth application I/O. This procedure disables DWS during a fresh install of SMW/CLE. For information about how to install and configure DataWarp, see *DataWarp Installation and Administration Guide*.

Procedure

1. Edit `cray_dws_worksheet.yaml`.

2. Uncomment `cray_dws.enabled` and set it to `false`.

7.1.9 Update Cray eLogin Service Worksheets

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure temporarily disables these three eLogin (external login) services during the fresh install of SMW/CLE. They will be enabled later during the installation and configuration process for eLogin.

Cray eLogin LNet (Lustre networking) service	LNet is needed by any system that has external login nodes that mount Lustre file systems.
Cray eLogin Networking service	Defines the number of eLogin nodes connected to the Cray system and their key network attributes.
Cray eswrap service	`eswrap` wraps several XT, ALPS (Application Level Placement Scheduler), and WLM (workload manager) commands on eLogin nodes and executes them on the Cray login gateway.

Procedure

1. Edit `cray_elogin_lnet_worksheet.yaml`.
2. Uncomment `cray_elogin_lnet.enabled` and set it to `false`.
3. Edit `cray_elogin_networking_worksheet.yaml`.
4. Uncomment `cray_elogin_networking.enabled` and set it to `false`.
5. Edit `cray_eswrap_worksheet.yaml`.
6. Uncomment `cray_eswrap.enabled` and set it to `false`.

7.1.10 Update cray_firewall Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures the inherit and enable settings in the Cray firewall service configuration worksheet. The firewall is a mechanism for restricting packet traffic from various networks. This service has a global template as well as a CLE template, and therefore it can be configured to inherit settings from the global config set.

Procedure

1. Edit `cray_firewall_worksheet.yaml`.
2. Uncomment `cray_firewall.inherit` and ensure that it is set to `false`.

   This means that firewall settings in the CLE config set will be used instead of firewall settings in the global config set.
3. Uncomment `cray_firewall.enabled` and set it to `true`.

7.1.11 Update cray_image_binding Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

The image binding service is a mechanism for mixing image content between booted IMPS (Image Management and Provisioning System) images and IMPS images that are projected onto a running system. This is a common scenario with the Cray Programming Environment (PE), which is installed into an IMPS image, pushed to the CLE boot node, then projected to compute nodes using DVS (Data Virtualization Service).

Procedure

1. Edit `cray_image_binding_worksheet.yaml`.
2. Uncomment `cray_image_binding.enabled` and set it to `true`.

   IMPORTANT: Do not enable any bind mount profiles now. Enabling them must wait until the associated image root has been pushed to the boot node. If the system has not been previously booted, failure to push an image root prior to enabling its profile may prevent the system from properly booting.

7.1.12 Update cray_ipforward Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures the inherit and enable settings in the Cray IP Forwarding service configuration worksheet. The IP Forwarding service enables IP forwarding between service nodes and the SMW. This service has a global template as well as a CLE template, and therefore it can be configured to inherit settings from the global config set.

Procedure

1. Edit `cray_ipforward_worksheet.yaml`.
2. Uncomment `cray_ipforward.inherit` and set it to `true`.

   This means that IP forwarding settings in the global config set will be used instead of IP forwarding settings in the CLE config set.
3. Uncomment `cray_ipforward.enabled` and ensure that it is set to `true`.

7.1.13 Update cray_liveupdates Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure enables/disables the Cray live updates service configuration worksheet. The live updates service enables package manager (e.g., zypper, yum) actions (e.g., install, search, upgrade) on CLE nodes using repositories shared from the SMW to those nodes.

Procedure

1. Edit `cray_liveupdate_worksheet.yaml`.

   ```
   smw# vi cray_liveupdate_worksheet.yaml
   ```
2. Uncomment `cray_liveupdate.enabled` and set it to `true`.

7.1.14 Update cray_lmt Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray LMT (Lustre Monitoring Tool) service configuration worksheet. LMT monitors Lustre servers (not clients) using the Cerebro monitoring system to collect and store server statistics.

Procedure

1. Edit `cray_lmt_worksheet.yaml`.

```
smw# vi cray_lmt_worksheet.yaml
```

2. Uncomment `cray_lmt.enabled` and set it to `false`.

   This disables this service. When a service is disabled, the values of all other settings in the service are ignored when configuration is applied. This worksheet contains additional configuration settings that are level=advanced, but this service and its advanced settings are not needed for a fresh install. If DAL (Direct Attached Lustre) is enabled and this site wishes to use LMT for DAL, enable this service later.

7.1.15 Update cray_lnet Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray LNet (Lustre networking) service configuration worksheet to add site-specific data. LNet is the networking layer used by Lustre and DVS.

Procedure

1. Edit `cray_lnet_worksheet.yaml`.

2. Uncomment `cray_lnet.enabled` and do one of the following:

   - Set it to `true` if this system has external Lustre or DAL (direct-attached Lustre) or will use DVS to mount external file systems..
   - Set it to `false` otherwise.

   For systems with external Lustre, continue to the next step. Otherwise, Cray LNet configuration is complete for a fresh install.

   THE REMAINING STEPS ARE ONLY FOR SYSTEMS WITH EXTERNAL LUSTRE *****

3. Configure these settings.

   These settings are commonly configured with site-specific data when the system has external Lustre. Uncomment and set them as appropriate for this site.

   `cray_lnet.settings.ko2iblnd.data.peer_credits`
   `cray_lnet.settings.ko2iblnd.data.concurrent_sends`
   `cray_lnet.settings.local_lnet.data.lnet_name` (set to something like gni, gni1, gni2, gni3)
   `cray_lnet.settings.local_lnet.data.ip_wildcard` (change from default on a partitioned system or any system that changes the HSN address range)

4. Configure the following group of settings if this system uses flat routes to an external Lustre server. Repeat this step for each external Lustre file system.

In the worksheet, copy the six lines below `# ** EXAMPLE 'flat_routes' VALUE (with current defaults) **` and paste them below `# NOTE: Place additional 'flat_routes' setting entries here, if desired.`

```
# ** EXAMPLE 'flat_routes' VALUE (with current defaults) **
# cray_lnet.settings.flat_routes.data.dest_lnet.sample_key_a: null <-- setting a multival key
# cray_lnet.settings.flat_routes.data.sample_key_a.dest_lnet_ip_wildcard: ''
# cray_lnet.settings.flat_routes.data.sample_key_a.routers: []
# cray_lnet.settings.flat_routes.data.sample_key_a.src_lnet: ''
# cray_lnet.settings.flat_routes.data.sample_key_a.ko2iblnd_peer_credits: 126
# cray_lnet.settings.flat_routes.data.sample_key_a.ko2iblnd_concurrent_sends: 63
#
# ** 'flat_routes' FIELD SPECIFICATION -- MULTIVAL KEY FIELD **
```

Uncomment the lines, replace `sample_key_a` with `o2ib` in all lines, and remove the `<-- setting a multival key` text at the end of the first line (note that the `null` value is required; do not remove or change it). Finally, modify the values as appropriate for this site.

```
# NOTE: Place additional 'flat_routes' setting entries here, if desired.
cray_lnet.settings.flat_routes.data.dest_lnet.o2ib: null
cray_lnet.settings.flat_routes.data.o2ib.dest_lnet_ip_wildcard: 10.149.*.*
cray_lnet.settings.flat_routes.data.o2ib.routers:
- c0-0c1s3n1
- c0-0c1s3n2
cray_lnet.settings.flat_routes.data.o2ib.src_lnet: gni0
cray_lnet.settings.flat_routes.data.o2ib.ko2iblnd_peer_credits: 126
cray_lnet.settings.flat_routes.data.o2ib.ko2iblnd_concurrent_sends: 63
#************* END Service Setting: flat_routes *************
```

5. Configure the following group of settings if this system uses fine-grained routing (FGR) to an external Lustre server. Repeat this step for each external Lustre file system.

   In the worksheet, copy the six lines below `# ** EXAMPLE 'fgr_routes' VALUE (with current defaults) **` and paste them below `# NOTE: Place additional 'fgr_routes' setting entries here, if desired.`

   ```
   # ** EXAMPLE 'fgr_routes' VALUE  (with current defaults) **
   # cray_lnet.settings.fgr_routes.data.dest_name.sample_key_a: null <-- setting a multival key
   # cray_lnet.settings.fgr_routes.data.sample_key_a.routers: []
   # cray_lnet.settings.fgr_routes.data.sample_key_a.ip2nets_file: ''
   # cray_lnet.settings.fgr_routes.data.sample_key_a.routes_file: ''
   # cray_lnet.settings.fgr_routes.data.sample_key_a.ko2iblnd_peer_credits: 126
   # cray_lnet.settings.fgr_routes.data.sample_key_a.ko2iblnd_concurrent_sends: 63
   #
   # ** 'fgr_routes' FIELD SPECIFICATION -- MULTIVAL KEY FIELD **
   ```

   Uncomment the lines, replace `sample_key_a` with `sonexion` in all lines, and remove the `<-- setting a multival key` text at the end of the first line (note that the `null` value is required; do not remove or change it). Finally, modify the values as appropriate for this site.

   ```
   # NOTE: Place additional 'fgr_routes' setting entries here, if desired.
   cray_lnet.settings.fgr_routes.data.dest_name.sonexion: null
   cray_lnet.settings.fgr_routes.data.sonexion.routers:
   - c0-0c1s0n1
   - c0-0c1s0n2
   - c0-0c1s2n1
   - c0-0c1s2n2
   cray_lnet.settings.fgr_routes.data.sonexion.ip2nets_file: 'ip2nets.conf'
   cray_lnet.settings.fgr_routes.data.sonexion.routes_file: 'routes.conf'
   cray_lnet.settings.fgr_routes.data.sonexion.ko2iblnd_peer_credits: 126
   ```

```
cray_lnet.settings.fgr_routes.data.sonexion.ko2iblnd_concurrent_sends: 63
#************* END Service Setting: fgr_routes *************
```

Remember to generate the two configuration files (`ip2nets.conf` and `routes.conf`) using an external tool, such as `clcvt`, and then place them in `/var/opt/cray/imps/config/sets/p0/files/roles/lnet` (for p0 config set).

There may be additional settings that should be set for sites with external Lustre servers. Seek advice from the site Lustre server administrator.

7.1.16 Update cray_local_users Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Local Users service configuration worksheet to add site-specific data. This service defines local user accounts such as root.

Procedure

1. Edit `cray_local_users_worksheet.yaml`.

   ```
   smw# vi cray_local_users_worksheet.yaml
   ```

2. Uncomment `cray_local_users.enabled` and set it to `true`.

3. If using local home directories, uncomment `cray_local_users.settings.directories.data.home` to configure the home directory location.

   (Most sites mount an external home file system instead.)

Do not set any other values at this point. In particular, do not set the crayadm and root accounts "crypt" setting, which must be an encrypted string. Later in this process, all of the configuration worksheets will be imported into the new CLE config set, and the config set will be updated. During the update, the configurator will prompt for the crayadm and root "crypt" settings. Because they are encrypted, the configurator will ask for the password, ask a second time to verify that they match, and then put an encrypted form of that password into the config set. Attempting to place an encrypted string into this worksheet manually is prone to error and could result in being unable to access these accounts.

7.1.17 Update cray_logging Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures the inherit and enable settings in the Cray Logging service configuration worksheet. This service has a global template as well as a CLE template, and therefore it can be configured to inherit settings from the global config set.

Procedure

1. Edit `cray_logging_worksheet.yaml`.

   ```
   smw# vi cray_logging_worksheet.yaml
   ```

2. Uncomment `cray_logging.inherit` and set it to `true`.

   This means that logging settings in the global config set will be used instead of logging settings in the CLE config set. If `cray_logging.inherit` is set to `false`, then other settings may need to be changed.

7.1.18 Update cray_login Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Login service configuration worksheet to add site-specific data. The Login service provides information and listings for login nodes, which are used by users to access the Cray system.

Procedure

1. Edit `cray_login_worksheet.yaml`.

   ```
   smw# vi cray_login_worksheet.yaml
   ```

2. Uncomment `cray_login.enabled` and set it to `true`.

3. Enter the cnames of the login nodes on this system.

   Uncomment `cray_login.settings.login_nodes.data.members` and set it to the list of login nodes (by cname). Add each node on a separate line prefixed by the - character followed by a space.

   ```
   cray_login.settings.login_nodes.data.members:
   - c0-0c1s1n2
   - c0-0c0s0n2
   ```

4. Uncomment `cray_login.settings.login_nodes.data.login_prohibited_after_boot` and do one of the following:

- Set it to `false` to have `/etc/nologin` removed automatically on each node as it completes its own boot.
- Set it to `true` to require a system administrator to remove `/etc/nologin` on each node by running a command like `sdb# pcmd -r -n ALL_SERVICE "rm /etc/nologin"` after all of the CLE nodes have been booted and the system is ready for users to log in.

7.1.19 Update cray_lustre_client Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Lustre client configuration worksheet to add site-specific data.

Procedure

1. Edit `cray_lustre_client_worksheet.yaml`.

   ```
   smw# vi cray_lustre_client_worksheet.yaml
   ```

2. Uncomment `cray_lustre_client.enabled` and do one of the following:

   - Set it to `true` for systems that are a Lustre client of either an external Lustre server or direct-attached Lustre (DAL). Proceed to the next step.
   - Set it to `false` otherwise. Skip the next step.

3. Configure a client mount for each Lustre file system that will be mounted.

   For example, to mount the two file systems that were described in *Update cray_lnet Worksheet* on page 117, these settings need to be added.

   In the worksheet, copy the seven lines below `# ** EXAMPLE 'client_mounts' VALUE (with current defaults) **` and paste them twice (this example has two client mounts) below `# NOTE: Place additional 'client_mounts' setting entries here, if desired.`

   ```
   # NOTE: Place additional 'client_mounts' setting entries here, if desired.
   # cray_lustre_client.settings.client_mounts.data.fs_name.sample_key_a: null <-- setting a
   multival key
   # cray_lustre_client.settings.client_mounts.data.sample_key_a.lustre_fs_name: ''
   # cray_lustre_client.settings.client_mounts.data.sample_key_a.mount_point: ''
   # cray_lustre_client.settings.client_mounts.data.sample_key_a.mgs_lnet_nids: []
   # cray_lustre_client.settings.client_mounts.data.sample_key_a.mount_options: rw,flock,lazystatfs
   # cray_lustre_client.settings.client_mounts.data.sample_key_a.mount_at_boot: true
   # cray_lustre_client.settings.client_mounts.data.sample_key_a.mount_locations:
   # - login
   # - compute
   # - elogin
   #************************ END Service Setting: client_mounts ************************
   ```

Uncomment the lines, replace `sample_key_a` with `snx11023` in the first set of lines and with `esfprod` in the second set, then remove the `<-- setting a multival key` text at the end of the first line in each set (note that the `null` value is required; do not remove or change it). Finally, modify the values as shown here.

```
# NOTE: Place additional 'client_mount' setting entries here, if desired.
cray_lustre_client.settings.client_mounts.data.fs_name.snx11023: null
cray_lustre_client.settings.client_mounts.data.snx11023.lustre_fs_name: ''
cray_lustre_client.settings.client_mounts.data.snx11023.mount_point: '/lus/snxs1'
cray_lustre_client.settings.client_mounts.data.snx11023.mgs_lnet_nids:
- 10.149.4.3@o2ib
- 10.149.4.4@o2ib
cray_lustre_client.settings.client_mounts.data.snx11023.mount_options: rw,flock,lazystatfs
cray_lustre_client.settings.client_mounts.data.snx11023.mount_at_boot: true
cray_lustre_client.settings.client_mounts.data.snx11023.mount_locations:
- login
- compute
- elogin

cray_lustre_client.settings.client_mounts.data.fs_name.esfprod: null
cray_lustre_client.settings.client_mounts.data.esfprod.lustre_fs_name: ''
cray_lustre_client.settings.client_mounts.data.esfprod.mount_point: '/lus/esfs'
cray_lustre_client.settings.client_mounts.data.esfprod.mgs_lnet_nids:
- 10.150.10.1@o2ib8000
- 10.150.10.2@o2ib8000
cray_lustre_client.settings.client_mounts.data.esfprod.mount_options: rw,flock,lazystatfs
cray_lustre_client.settings.client_mounts.data.esfprod.mount_at_boot: true
cray_lustre_client.settings.client_mounts.data.esfprod.mount_locations:
- login
- compute
- elogin
#************ END Service Setting: client_mounts *************
```

7.1.20 Update cray_lustre_server Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Lustre server service worksheet to add site-specific data. This service enables configuration of Lustre-server-related kernel module parameters.

Procedure

1. Edit `cray_lustre_server_worksheet.yaml`.

   ```
   smw# vi cray_lustre_server_worksheet.yaml
   ```

2. Uncomment `cray_lustre_server.enabled` and do one of the following:
   - Set it to `true` for systems that have DAL (Direct Attached Lustre). Proceed to the next step.
   - Set it to `false` otherwise. Skip the remaining steps.

3. (Only for systems with DAL) Uncomment `cray_lustre_server.settings.lustre_servers.data.mgs` and set it to the service node (cname) that will be the Lustre Management Server (MGS).

```
cray_lustre_server.settings.lustre_servers.data.mgs: c0-0c0s2n1
```

4. (Only for systems with DAL) Uncomment `cray_lustre_server.settings.lustre_servers.data.mds` and set it to the list of service nodes (cnames) that will be the Lustre MetaData Server(s) (MDS).

```
cray_lustre_server.settings.lustre_servers.data.mds:
- c0-0c0s2n2
- c0-0c1s2n2
```

5. (Only for systems with DAL) Uncomment `cray_lustre_server.settings.lustre_servers.data.oss` and set it to the list of service nodes (cnames) that will be the Lustre Object Storage Server(s) (OSS).

```
cray_lustre_server.settings.lustre_servers.data.oss:
- c0-0c0s2n2
- c0-0c1s2n2
- c0-0c2s2n2
- c0-0c2s3n1
```

This worksheet contains additional settings that tune the Lustre kernel modules. Seek advice from the site Lustre server administrator before changing them.

7.1.21 Update cray_multipath Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures the inherit and enable settings in the Cray multipath service configuration worksheet. If the hardware connections to the boot RAID are present to support multipath, the multipath service is a mechanism for configuring redundant paths for failover or performance reasons. This service has a global template as well as a CLE template, and therefore the CLE template can be configured to inherit settings from the global config set or configured independently. For most multipath configurations, Cray recommends managing multipath settings from the cray_multipath template in the global config set.

Procedure

1. Edit `cray_multipath_worksheet.yaml`.

2. Uncomment `cray_multipath.inherit` and set it to `true`.

This means that multipath settings in the global config set will be used instead of multipath settings in this CLE config set.

If this site chooses to configure multipath in one or more CLE config sets instead, then set `cray_multipath.inherit` to `false` in step 2 and continue to step 3.

3. Uncomment `cray_multipath.enabled`.

   Set it to `true` if the hardware connections to the boot RAID are present to support multipath, otherwise set it to `false`. If enabling this service, check other settings as well.

7.1.22 Update cray_munge Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray MUNGE service configuration worksheet. MUNGE is an authentication service that creates and validates credentials. It is required by the DataWarp service (cray_dws) and Slurm, a workload manager.

Procedure

1. Edit `cray_munge_worksheet.yaml`.

2. Uncomment `cray_munge.enabled` and do one of the following:

   - Set it to `true` only if this site wishes to enable the DataWarp service or Slurm while doing a fresh install of SMW/CLE software.
   - Set it to `false` otherwise.

   If the MUNGE service was disabled in this step, it can be enabled later when configuring DataWarp or Slurm.

7.1.23 Update cray_netroot_preload Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure enables the Cray Netroot Preload service using its configuration worksheet. Netroot Preload is a mechanism for populating a Cray system node's root file system early in the boot process to reduce load on the DVS (Data Virtualization Service) servers providing the data and thereby reduce boot times for Netroot nodes. Netroot Preload also improves post-boot performance—how much improvement depends on the workloads. More

information about Netroot is provided later in the installation process; it is not necessary to know about Netroot before enabling Netroot Preload (steps 1 and 2). This service is needed if Netroot is used, and does no harm if Netroot is not used.

Cray provides two default "load" settings that define target nodes and files to be preloaded to them. If this site wishes to define a custom load as well (optional), continue to step 3 (see also *About Configuring Netroot Preload* on page 125).

Procedure

1. Edit `cray_netroot_preload_worksheet.yaml`.
2. Uncomment `cray_netroot_preload.enabled`. Keep it set to `true`, which is the default.

   Continue to step 3 to define a custom load (optional).
3. Define a custom load.

   In the worksheet, copy the four lines below `# ** EXAMPLE 'load' VALUE (with current defaults) **` and paste them below `# NOTE: Place additional 'load' setting entries here, if desired.`

   ```
   # NOTE: Place additional 'load' setting entries here, if desired.
   # cray_netroot_preload.settings.load.data.label.sample_key_a: null <-- setting a multival key
   # cray_netroot_preload.settings.load.data.sample_key_a.targets: []
   # cray_netroot_preload.settings.load.data.sample_key_a.content_lists: []
   # cray_netroot_preload.settings.load.data.sample_key_a.size_limit: 0
   #************************ END Service Setting: load ************************
   ```

   Uncomment the lines, replace `sample_key_a` with the label for this load (e.g., `my_load`) in all lines, and remove the `<-- setting a multival key` text at the end of the first line (note that the `null` value is required; do not remove or change it). Finally, add site-specific values. Add each list element on a separate line prefixed by the – character followed by a space.

   ```
   # NOTE: Place additional 'domain' setting entries here, if desired.
   cray_netroot_preload.settings.load.data.label.my_load: null
   cray_netroot_preload.settings.load.data.my_load.targets:
   - <site-defined node group>
   cray_netroot_preload.settings.load.data.my_load.content_lists:
   - relative/path/to/oft-requested/files
   cray_netroot_preload.settings.load.data.my_load.size_limit: 0
   #************* END Service Setting: domain *************
   ```

7.1.23.1 About Configuring Netroot Preload

Netroot is a feature that enables nodes booted with a minimal, local in-memory file system to execute within the context of a larger, full-featured root file system. Netroot uses the Data Virtualization Service (DVS) to access the remote root content. While DVS has data and attribute caching features that minimize the impact of most remote references, files that are referenced frequently may still incur an undesirable performance penalty.

The Netroot Preload feature mitigates that performance penalty by copying specified remote files from the Netroot to a node-local in-memory file system early in the node boot process. All future references to those files will be serviced by the local file system rather than requiring remote data and/or metadata DVS operations. This improves system and application performance. However, as a consequence, the amount of memory available on the node is reduced by the cumulative size of all files copied into its memory.

Netroot Preload can be enabled, disabled, and customized using the configurator on the SMW or by editing the configuration worksheets on the SMW.

Netroot Preload Configuration Settings

Netroot Preload configuration consists of defining one or more "loads," or sets of data to be preloaded on specified nodes. The load setting has the following fields:

label A convenient, descriptive label for a particular load.

targets A list of node groups that reference the nodes that will be preloaded with files on their local file systems. Must provide at least one node group.

content_lists Content lists are relative paths to files within the config set. These files contain file paths that are copied into the node-local memory by Netroot Preload. For example, content list `dist/compute-preload.cray` within config set p0 has these contents:

```
smw# cat p0/dist/compute-preload.cray
/etc/ld.so.cache
/opt/cray/rca/*/bin/rca-helper
/lib64/libc-*.so
/lib64/ld-*.so
/opt/cray/rca/*/lib*/librca.so.*
[...]
```

Pattern matching is supported.

size_limit The memory-consumption limit (in MB) set for this load, which limits how much can be copied to any node. As the files are copied via Netroot, Netroot Preload checks the sizes and amount of data copied so far. When it reaches the size limit, it stops, and any remaining files are not copied. Setting this to zero (0) indicates no limit.

Default Loads

Cray provides two default loads: the 'compute' load, which targets all compute nodes in the system, and the 'login' load, which targets all internal login nodes in the system. The compute load has a single content list specified: `dist/compute-preload.cray`. This file contains paths that are commonly referenced during the node boot and initialization process. Similarly, the login load specifies this content list as the only entry in its content_lists setting: dist/login-preload.cray. Note that each of these is a relative path. The full path would be `/var/opt/cray/imps/config/sets/p0/dist/login-preload.cray` for the login content list entry. If a site disables or modifies these default settings, the time it takes to boot and initialize nodes may increase.

Custom Loads

Sites may define their own loads as well. This enables sites to optimize for specific workloads. For targets, sites can use existing node groups or define their own (see *Update cray_node_groups Worksheet* on page 133). To aid in determining which file paths to add to load content lists, use the DVS request log, introduced in CLE 6.0.UP01, which is how the Cray default content lists were created. The `/proc/fs/dvs/request_log` file contains a log of all DVS requests initiated since the log was enabled. File paths that are referenced often make good candidates. See "DVS Can Log Requests Sent to Servers" in *DVS Administration Guide* (S-0005) for additional information about this request log.

The Netroot Preload Log File and a Note about Symlinks

Netroot Preload creates a log file on affected nodes at `/var/opt/cray/log/netroot_preload.log`. This log file contains details on the files preloaded, which, if any, files were not found in the Netroot, and the size of the files preloaded on the node. Any failures will also be logged to the console file on the SMW.

Note that any symlinks included in a load content list may not be copied from Netroot to the node-local RAM file system (i.e., "promoted" in the log file), which might look confusing. For example, suppose a site content list contains `/etc/alternatives/unzip`, which is a symlink to `/usr/bin/unzip-plain`. While both the link and its target are present in Netroot, neither of them appear in the node-local file system, despite the log saying `Promoted '/new_root/merge/etc/alternatives/unzip'`. This is expected and correct behavior. A site that is concerned about possible confusion for administrators can decide to exclude symlinks from content lists or simply list the target of the symlink in a content list to ensure that it is present in the node-local file system.

7.1.24 Update cray_net Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Networking service configuration worksheet to add site-specific data. The Cray Networking service defines all network information for CLE nodes, so there are many site-specific settings that need to be configured.

> **REMEMBER:** For partitioned systems, each partition generally has its own config set and associated configuration worksheets. Follow this procedure to make changes to the `cray_net_worksheet.yaml` for each partition. Some steps call out what settings should be different for different partitions.

Procedure

1. Edit `cray_net_worksheet.yaml`.
2. Uncomment `cray_net.enabled` and set it to `true`.
3. Uncomment these two settings for the HSN (high speed network).

   If this is a partitioned system, then enter different values for these settings. Partitions p1 and p2 will not have the same ipv4 network, but will have similar ipv4_netmask (though different from the full machine).

   ```
   # ** 'networks' DATA **
   cray_net.settings.networks.data.hsn.ipv4_network: 10.128.0.0
   cray_net.settings.networks.data.hsn.ipv4_netmask: 255.252.0.0
   ```

   DEFINE NETWORKS ----------

   IMPORTANT:

   - Add values for the `dns_servers` and `dns_search` settings to the login network only, not to any other network.

- DO NOT add a value for the `ntp_servers` setting for any network used for CLE nodes, because CLE nodes must source their time/NTP settings from the SMW rather than try to contact NTP servers on the login network.

4. Configure a login network and add the information for the "Customer network" to which the login nodes connect.

Scroll down to the predefined network settings below the `# ** 'networks' DATA **` line and find the login network definition. Modify the values as needed for this site's internal systems. Note that in the first line, the `null` value is required; do not remove or change it.

```
# ** 'networks' DATA **
...
cray_net.settings.networks.data.name.login: null
cray_net.settings.networks.data.login.description: Customer network
cray_net.settings.networks.data.login.ipv4_network: 172.30.48.0
cray_net.settings.networks.data.login.ipv4_netmask: 255.255.240.0
cray_net.settings.networks.data.login.ipv4_broadcast: ''
cray_net.settings.networks.data.login.ipv4_gateway: 172.30.48.1
cray_net.settings.networks.data.login.dns_servers:
- 172.30.84.40
- 172.31.84.40
- 172.28.84.40
cray_net.settings.networks.data.login.dns_search:
- us.cray.com
- americas.cray.com
- cray.com
cray_net.settings.networks.data.login.ntp_servers: []
cray_net.settings.networks.data.login.fw_external: false
```

IMPORTANT: If the login network should be treated as an external network for the firewall, then set `cray_net.settings.networks.data.login.fw_external` (the last line) to true.

5. (Only for systems with an external Lustre server) Configure a network for each external Lustre server.

In the worksheet, copy the ten lines below `# ** EXAMPLE 'networks' VALUE (with current defaults) **` and paste one set for each external Lustre server below the line `# NOTE: Place additional 'networks' setting entries here, if desired.`

```
# ** EXAMPLE 'networks' VALUE  (with current defaults) **
# cray_net.settings.networks.data.name.sample_key_a: null <-- setting a multival key
# cray_net.settings.networks.data.sample_key_a.description: ''
# cray_net.settings.networks.data.sample_key_a.ipv4_network: ''
# cray_net.settings.networks.data.sample_key_a.ipv4_netmask: ''
# cray_net.settings.networks.data.sample_key_a.ipv4_broadcast: ''
# cray_net.settings.networks.data.sample_key_a.ipv4_gateway: ''
# cray_net.settings.networks.data.sample_key_a.dns_servers: []
# cray_net.settings.networks.data.sample_key_a.dns_search: []
# cray_net.settings.networks.data.sample_key_a.ntp_servers: []
# cray_net.settings.networks.data.sample_key_a.fw_external: false

# ** 'networks' FIELD SPECIFICATION -- MULTIVAL KEY FIELD **
```

Uncomment the lines, replace `sample_key_a` with lnet in all lines, and remove the `<-- setting a multival key` text at the end of the first line (note that the null value is required; do not remove or change it). Finally, modify the values as needed for this site. The following example shows three different 'networks' entries.

```
# NOTE: Place additional 'networks' setting entries here, if desired.
cray_net.settings.networks.data.name.lnet: null
cray_net.settings.networks.data.lnet.description: LNet network to CLFS
```

```
cray_net.settings.networks.data.lnet.ipv4_network: 10.150.0.0
cray_net.settings.networks.data.lnet.ipv4_netmask: 255.255.0.0
cray_net.settings.networks.data.lnet.ipv4_broadcast: ''
cray_net.settings.networks.data.lnet.ipv4_gateway: ''
cray_net.settings.networks.data.lnet.dns_servers: []
cray_net.settings.networks.data.lnet.dns_search: []
cray_net.settings.networks.data.lnet.ntp_servers: []
cray_net.settings.networks.data.lnet.fw_external: false

cray_net.settings.networks.data.name.lnet1: null
cray_net.settings.networks.data.lnet1.description: LNet network to CLFS2
cray_net.settings.networks.data.lnet1.ipv4_network: 10.151.0.0
cray_net.settings.networks.data.lnet1.ipv4_netmask: 255.255.0.0
cray_net.settings.networks.data.lnet1.ipv4_broadcast: ''
cray_net.settings.networks.data.lnet1.ipv4_gateway: ''
cray_net.settings.networks.data.lnet1.dns_servers: []
cray_net.settings.networks.data.lnet1.dns_search: []
cray_net.settings.networks.data.lnet1.ntp_servers: []
cray_net.settings.networks.data.lnet1.fw_external: false

cray_net.settings.networks.data.name.lnet2: null
cray_net.settings.networks.data.lnet2.description: LNet network to Sonexion
cray_net.settings.networks.data.lnet2.ipv4_network: 10.149.0.0
cray_net.settings.networks.data.lnet2.ipv4_netmask: 255.255.0.0
cray_net.settings.networks.data.lnet2.ipv4_broadcast: ''
cray_net.settings.networks.data.lnet2.ipv4_gateway: ''
cray_net.settings.networks.data.lnet2.dns_servers: []
cray_net.settings.networks.data.lnet2.dns_search: []
cray_net.settings.networks.data.lnet2.ntp_servers: []
cray_net.settings.networks.data.lnet2.fw_external: false
#************* END Service Setting: networks *************
```

DEFINE HOSTS ----------

6. Configure a host as the boot node.

   Cray has defined a default `bootnode` host, which is located under the `# ** 'hosts' DATA **` line.

   > **IMPORTANT:** Never set `cray_net.settings.hosts.data.bootnode.aliases` to "boot" because that is a hostname alias that belongs to the virtual IP address for the boot node in support of the boot node failover feature.

   a. Uncomment `cray_net.settings.hosts.data.bootnode.roles` and the line following to set the boot node role.

   ```
   cray_net.settings.hosts.data.bootnode.roles:
   - boot
   ```

   b. Uncomment `cray_net.settings.hosts.data.bootnode.hostid` and set it to be the cname of the boot node.

   c. Uncomment `cray_net.settings.hosts.data.bootnode.hostname` and set it to the hostname of the boot node.

      Choose a name that includes the machine name and "boot" such as "boot-orion," or if this is a partitioned system, then identify the boot node as "boot-p1," "boot-p2," and so forth. Do not set the hostname to "boot" because that name is reserved for the virtual IP address of the boot node, regardless of whether it is the full system or a partitioned system.

d. Uncomment `cray_net.settings.hosts.data.bootnode.interfaces.primary_ethernet.ipv4_address` and set it to one of the following.

   This is on the "admin" network to the SMW, so this setting should be set to:

   - 10.3.1.254 for a full system (p0).
   - 10.3.1.254 for p1, 10.3.1.252 for p2, and so forth for partitioned systems.

e. Uncomment `cray_net.settings.hosts.data.bootnode.interfaces.hsn_boot_alias.ipv4_address` and set it to one of the following.

   This is on the HSN and is the "'virtual IP address" for the virtual interface ipgif0:1, so the setting should be set to:

   - 10.131.255.254 for a full system (p0).
   - The highest address possible for a partition's HSN, for partitioned systems. For example, if p1 HSN `ipv4_address`=10.128.0.0 with `ipv4_netmask` 255.255.0.0, then set `ipv4_address`=10.128.255.254 for p1. If p2 HSN `ipvr_address`=10.129.0.0 with `ipv4_netmask` 255.255.0.0, then set `ipv4_address`=10.129.255.254 for p2.

7. Configure a host as the SDB node.

Cray has defined a default `sdbnode` host, which is located under the `# ** 'hosts' DATA **` line.

a. Uncomment `cray_net.settings.hosts.data.sdbnode.roles` and the line following to set the SDB node role.

```
cray_net.settings.hosts.data.sdbnode.roles:
- sdb
```

b. Uncomment `cray_net.settings.hosts.data.sdbnode.hostid` and set it to the cname of the SDB node.

c. Uncomment `cray_net.settings.hosts.data.sdbnode.hostname` and set it to "sdb."

d. Uncomment `cray_net.settings.hosts.data.sdbnode.interfaces.primary_ethernet.ipv4_address` and set it to one of the following.

   This is on the "admin" network to the SMW, so this setting should be set to:

   - 10.3.1.253 for a full system (p0).
   - 10.3.1.253 for p1, 10.3.1.251 for p2, and so forth for partitioned systems.

e. Uncomment `cray_net.settings.hosts.data.sdbnode.interfaces.hsn_sdb_alias.ipv4_address` and set it to one of the following.

   This is on the HSN and is the "'virtual IP address" for the virtual interface ipgif0:1, so the setting should be set to:

   - 10.131.255.253 for a full system (p0).
   - The highest address possible for a partition's HSN, for partitioned systems. For example, if p1 HSN `ipv4_address`=10.128.0.0 with `ipv4_netmask` 255.255.0.0, then set `ipv4_address`=10.128.255.253 for p1. If p2 HSN `ipvr_address`=10.129.0.0 with `ipv4_netmask` 255.255.0.0, then set `ipv4_address`=10.129.255.253 for p2.

8. Configure a host as the login node.

a. Uncomment `cray_net.settings.hosts.data.login_node.aliases` and set it as follows:

- If this site wishes the login node to have a hostname alias of "login," set this setting to "login."
- If this site has more than one login node, the first one could have aliases of "login" and "login1," and the others would be set to "login2," "login3," and so forth.

b. Uncomment `cray_net.settings.hosts.data.login_node.roles`, but do not set it to any value.

This should not be set for any nodes except the boot node and SDB node.

c. Uncomment `cray_net.settings.hosts.data.login_node.hostid` and set it to the cname of the login node.

If this system has more than one login node, set this to be the first login node.

d. Uncomment `cray_net.settings.hosts.data.login_node.hostname` and set it to the hostname.

This could be the machine name, for systems that have only one login node. For example, on a machine known as orion, this would be "orion." For systems with more than one login node, the hostname could be "orion1" for the first one, "orion2" for the second one, and so forth.

e. Uncomment `cray_net.settings.hosts.data.login_node.interfaces.login_ethernet.ipv4_address` and set it to the IP address of the login node's eth0 interface on the "login" network.

9. Configure additional hosts, as needed.

If this system has additional service nodes that need to have hostname or hostname alias or network interface settings, then add a section like this for each of the hosts. The first example shows the host configuration of a DVS node with the hostname set to "dvs1," a hostname alias of "dvs," and one Ethernet interface connected to the "login" network.

```
cray_net.settings.hosts.data.common_name.dvs_node: null
cray_net.settings.hosts.data.dvs_node.description: DVS node
cray_net.settings.hosts.data.dvs_node.aliases:
- dvs
cray_net.settings.hosts.data.dvs_node.roles: []
cray_net.settings.hosts.data.dvs_node.hostid: 'c0-0c0s0n2'
cray_net.settings.hosts.data.dvs_node.host_type: ''
cray_net.settings.hosts.data.dvs_node.hostname: 'dvs1'
cray_net.settings.hosts.data.dvs_node.standby_node: false

cray_net.settings.hosts.data.dvs_node.interfaces.common_name.eth0: null
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.name: eth0
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.description: Ethernet
    connecting the DVS node to the customer network.
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.aliases: []
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.network: login
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.ipv4_address: '172.30.50.128'
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.mac: ''
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.startmode: auto
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.bootproto: static
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.mtu: ''
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.extra_attributes: []
#cray_net.settings.hosts.data.dvs_node.interfaces.eth0.module: ''
#cray_net.settings.hosts.data.dvs_node.interfaces.eth0.params: ''
#cray_net.settings.hosts.data.dvs_node.interfaces.eth0.unmanaged_interface: false
```

The second example shows the host configuration for an LNet router node that has two different InfiniBand interfaces (ib0 and ib2) to connect to two different networks.

```
cray_net.settings.hosts.data.common_name.clfs_lnet_1: null
cray_net.settings.hosts.data.clfs_lnet_1.description: CLFS router 1 node
cray_net.settings.hosts.data.clfs_lnet_1.aliases: []
```

```
cray_net.settings.hosts.data.clfs_lnet_1.roles: []
cray_net.settings.hosts.data.clfs_lnet_1.hostid: 'c0-0c1s0n1'
cray_net.settings.hosts.data.clfs_lnet_1.host_type: ''
cray_net.settings.hosts.data.clfs_lnet_1.hostname: lnet1
cray_net.settings.hosts.data.clfs_lnet_1.standby_node: false
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.common_name.ib0: null
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.name: ib0
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.description: InfiniBand
    ib0 connecting the CLFS router 1 node to the lnet network.
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.aliases: []
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.network: lnet
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.ipv4_address: '10.150.10.65'
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.mac: ''
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.startmode: auto
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.bootproto: static
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.mtu: '65520'
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.extra_attributes:
- IPOIB_MODE='connected'
#cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.module: ''
#cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.params: ''
#cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.unmanaged_interface: false

cray_net.settings.hosts.data.clfs_lnet_1.interfaces.common_name.ib2: null
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.name: ib2
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.description: InfiniBand
    ib2 connecting the CLFS router 1 node to the lnet1 network.
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.aliases: []
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.network: lnet1
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.ipv4_address: '10.151.10.65'
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.mac: ''
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.startmode: auto
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.bootproto: static
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.mtu: '65520'
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.extra_attributes:
- IPOIB_MODE='connected'
#cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.module: ''
#cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.params: ''
#cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.unmanaged_interface: false
```

10. Configure a host as the second boot node for boot node failover.

This node must have "standby_node" set to true.

```
cray_net.settings.hosts.data.common_name.backup_bootnode: null
cray_net.settings.hosts.data.backup_bootnode.description: backup Boot node for the
    system
cray_net.settings.hosts.data.backup_bootnode.aliases:
- cray-boot2
cray_net.settings.hosts.data.backup_bootnode.roles:
- boot
cray_net.settings.hosts.data.backup_bootnode.hostid: c0-0c0s4n1
cray_net.settings.hosts.data.backup_bootnode.host_type: admin
cray_net.settings.hosts.data.backup_bootnode.hostname: boot2
cray_net.settings.hosts.data.backup_bootnode.standby_node: true

cray_net.settings.hosts.data.backup_bootnode.interfaces.common_name.hsn_boot_alias: null
cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.name: ipogif0:1
cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.description: Well
    known address used for boot node services.
cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.aliases: []
cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.network: hsn
cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.ipv4_address: 10.131.255.254
cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.mac: ''
cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.startmode: ''
cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.bootproto: static
cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.mtu: ''
cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.extra_attributes: []
cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.module: ''
cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.params: ''
#cray_net.settings.hosts.data.backup_bootnode.interfaces.hsn_boot_alias.unmanaged_interface: false

cray_net.settings.hosts.data.backup_bootnode.interfaces.common_name.primary_ethernet: null
cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.name: eth0
cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.description: Ethernet
    connecting boot node to the SMW.
```

```
cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.aliases: []
cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.network: admin
cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.ipv4_address: 10.3.1.254
cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.mac: ''
cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.startmode: ''
cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.bootproto: static
cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.mtu: ''
cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.extra_attributes: []
cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.module: ''
cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.params: ''
#cray_net.settings.hosts.data.backup_bootnode.interfaces.primary_ethernet.unmanaged_interface: false
```

11. Configure a host as the second SDB node for SDB node failover.

This node must have "standby_node" set to true.

```
cray_net.settings.hosts.data.common_name.backup_sdbnode: null
cray_net.settings.hosts.data.backup_sdbnode.description: backup SDB node for the system
cray_net.settings.hosts.data.backup_sdbnode.aliases:
- cray-sdb2
cray_net.settings.hosts.data.backup_sdbnode.roles:
- sdb
cray_net.settings.hosts.data.backup_sdbnode.hostid: c0-0c0s3n1
cray_net.settings.hosts.data.backup_sdbnode.host_type: admin
cray_net.settings.hosts.data.backup_sdbnode.hostname: sdb2
cray_net.settings.hosts.data.backup_sdbnode.standby_node: true

cray_net.settings.hosts.data.backup_sdbnode.interfaces.common_name.hsn_boot_alias: null
cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.name: ipogif0:1
cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.description: Well
    known address used for SDB node services.
cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.aliases: []
cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.network: hsn
cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.ipv4_address: 10.131.255.253
cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.mac: ''
cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.startmode: auto
cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.bootproto: static
cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.mtu: ''
cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.extra_attributes: []
cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.module: ''
cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.params: ''
#cray_net.settings.hosts.data.backup_sdbnode.interfaces.hsn_boot_alias.unmanaged_interface: false

cray_net.settings.hosts.data.backup_sdbnode.interfaces.common_name.primary_ethernet: null
cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.name: eth0
cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.description: Ethernet
    connecting SDB node to the SMW.
cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.aliases: []
cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.network: admin
cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.ipv4_address: 10.3.1.253
cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.mac: ''
cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.startmode: auto
cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.bootproto: static
cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.mtu: ''
cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.extra_attributes: []
cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.module: ''
cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.params: ''
#cray_net.settings.hosts.data.backup_sdbnode.interfaces.primary_ethernet.unmanaged_interface: false
```

7.1.25 Update cray_node_groups Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Node Groups service configuration worksheet to add site-specific data. Node Groups are a mechanism for defining logical groupings of Cray system nodes to streamline node specifications for use in other Cray services. The node groups defined are non-exclusive, that is, a node may belong to more than one node group.

For release CLE 6.0.UP01, node groups can be used in custom Ansible plays and with the Cray Simple Sync service. Use this procedure to enable the service and define node groups only if this site will be using them in one or both or those ways.

Procedure

1. Edit `cray_node_groups_worksheet.yaml`.
2. Uncomment `cray_node_groups.enabled` and ensure that it is set to `true`.
3. Define a node group.

   Copy the three commented lines under

   ```
   ** EXAMPLE 'groups' VALUE (with current defaults) **
   ```

   and paste them in this location:

   ```
   # NOTE: Place additional 'group' setting entries here, if desired.
   #
   # ** 'groups' DATA **
   #cray_node_groups.settings.groups.data.group_name.sample_key_a: null <--setting a multival key
   #cray_node_groups.settings.groups.data.sample_key_a.decription: ''
   #cray_node_groups.settings.groups.data.sample_key_a.members: []
   #********** END Service Setting: groups **********
   ```

   Uncomment the lines, replace `sample_key_a` with the identifier chosen for this node group (e.g., special_server_group) in all lines, and remove the `<-- setting a multival key` text at the end of the first line (note that the `null` value is required; do not remove or change it). Finally, add values for the `description` (a string) and `members` (a list) fields. For the `members` field, add each list element on a separate line prefixed by a hyphen and space `-` . Here is an example:

   ```
   # NOTE: Place additional 'group' setting entries here, if desired.
   #
   # ** 'groups' DATA **
   cray_node_groups.settings.groups.data.group_name.example_group_1: null
   cray_node_groups.settings.groups.data.example_group_1.decription: ''
   cray_node_groups.settings.groups.data.example_group_1.members:
   - c0-0c0s8n0
   - c0-0c0s8n1
   #********** END Service Setting: groups **********
   ```

4. Repeat step 3 to define additional node groups.

7.1.25.1 About Node Groups

The Cray Node Groups service (cray_node_groups) enables administrators to define and manage logical groupings of system nodes. Nodes can be grouped arbitrarily, though typically they are grouped by software functionality or hardware characteristics, such as login, compute, service, DVS servers, and RSIP servers.

Sites are encouraged to define their own node groups and specify their members. Administrators can define and manage node groups using any of these methods:

- Edit and upload the node groups configuration worksheet (`cray_node_groups_worksheet.yaml`).
- Use the `cfgset` command to view and modify node groups interactively with the configurator.
- Edit the node groups configuration template (`cray_node_groups_config.yaml`) directly.

After using any of these methods, remember to validate the config set.

Characteristics of Node Groups

- Node group membership is not exclusive, that is, a node may be a member of more than one node group.
- Node group membership is specified as a list of cnames. However, if the SMW is part of a node group, it is specified with the output of the `hostid` command.
- All compute nodes and/or all service nodes can be added as node group members by including the keywords “platform:compute” and/or “platform:service” in a node group.
- The Configuration Management Framework (CMF) exposes node group membership of the current node through the local system "facts" provided by the Ansible runtime environment. This means that each node knows what node groups it belongs to, and that knowledge can be used in Cray and site-local Ansible playbooks.

7.1.26 Update cray_node_health Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Node Health service configuration worksheet. The Node Health service configures the Cray Node Health Checker (NHC).

Procedure

1. Edit `cray_node_health_worksheet.yaml`.

   ```
   smw# vi cray_node_health_worksheet.yaml
   ```

2. Uncomment `cray_node_health.enabled` and set it to `true`.

No other settings need to be changed for a fresh install, unless this site has custom plugins to add. In most cases, sites with custom plugins should install and configure CLE first, and then return later to the Cray Node Health service to configure the `custom_plugins` setting, if desired.

The `custom_plugins` setting can be reconfigured by editing `cray_node_health_worksheet.yaml` again and updating the config set or by invoking the configurator in interactive mode for this service only. For example, for config set p0:

```
smw# cfgset update -s cray_node_health -m interactive -l advanced p0
```

7.1.27 Update cray_persistent_data Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Persistent Data service configuration worksheet to add site-specific data. The Cray Persistent Data service provides persistent storage to nodes, and can be configured on a per-node basis.

Procedure

1. Edit `cray_persistent_data_worksheet.yaml`.

2. Uncomment `cray_persistent_data.enabled` and set it to `true`.

3. Uncomment `cray_persistent_data.settings.directories.data.persistent_space_mount` and set it to match the `fs_mountpoint` for the CLE storage set (cledefault).

   The full setting name is `cray_bootraid.settings.storage_sets.data.cledefault.volume_groups.boot_node_vg.volumes.nvolatile.fs_mount_point`, which is a setting in the `cray_bootraid` service in the global config set. Find this value by using `cfgset search`, and then scan the list of matches for this setting.

   ```
   smw# cfgset search --service cray_bootraid --level advanced \
   --status all --term nvolatile global
   ```

4. If the Cray DRC (dynamic RDMA credentials) service will be used with persistent `/var`, configure space for it.

   In the worksheet, copy the five lines below `# ** EXAMPLE 'mounts' VALUE (with current defaults) **` and paste them below `# NOTE: Place additional 'mounts' setting entries here, if desired.`

   ```
   # ** EXAMPLE 'mounts' VALUE
   # cray_persistent_data.settings.mounts.data.mount_point.sample_key_a: null <-- setting a multival key
   # cray_persistent_data.settings.mounts.data.sample_key_a.alt_storage_path: ''
   # cray_persistent_data.settings.mounts.data.sample_key_a.options: ''
   # cray_persistent_data.settings.mounts.data.sample_key_a.ancestor_def_perms: '0771'
   # cray_persistent_data.settings.mounts.data.sample_key_a.clients: []
   ```

   Uncomment the lines, replace `sample_key_a` with `/var/opt/cray/rdma-credentials` in all lines, and remove the `<-- setting a multival key` text at the end of the first line (note that the `null` value is required; do not remove or change it).

   For the `clients` setting (last line), add the cname of the service node that should be running the DRC service. This is the same value as was set for the `cray_drc.settings.server.data.server_cname` setting in the Cray DRC worksheet (`cray_drc_worksheet.yaml`).

   Leave all other settings at the default values.

```
# NOTE: Place additional 'mounts' setting entries here, if desired.
cray_persistent_data.settings.mounts.data.mount_point./var/opt/cray/rdma-credentials: null
cray_persistent_data.settings.mounts.data./var/opt/cray/rdma-credentials.alt_storage_path: ''
cray_persistent_data.settings.mounts.data./var/opt/cray/rdma-credentials.options: ''
cray_persistent_data.settings.mounts.data./var/opt/cray/rdma-credentials.ancestor_def_perms: '0771'
cray_persistent_data.settings.mounts.data./var/opt/cray/rdma-credentials.clients:
- drcs_cname

#************************ END Service Setting: mounts ************************
```

7.1.28 Update cray_rsip Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray RSIP (realm-specific IP) service configuration worksheet to add site-specific data. RSIP helps to maintain packet integrity.

Procedure

1. Edit `cray_rsip_worksheet.yaml`.

   ```
   smw# vi cray_rsip_worksheet.yaml
   ```

2. Uncomment `cray_rsip.enabled` and do one of the following:

   - Set it to `true` if this system has any service nodes providing the RSIP service. Proceed to the next step.
   - Set it to `false` otherwise. Skip the remaining steps.

3. (Only for systems with RSIP) Uncomment `cray_rsip.settings.service.data.servers` and set it to the list of service nodes (cnames) that will be RSIP servers.

   ```
   cray_rsip.settings.service.data.servers:
   - c0-0c0s0n2
   ```

4. (Only for systems with RSIP) Uncomment `cray_rsip.settings.service.data.nodes_as_client` and set it to the list of service nodes (cnames) that need to be RSIP clients, such as a MOM node.

   ```
   cray_rsip.settings.service.data.nodes_as_client:
   - c0-0c1s1n1
   ```

5. (Only for systems with RSIP) See the guidance for `cray_rsip.settings.service.data.use_xtrsipcfg` if this system has a complex RSIP configuration.

   The guidance describes how to invoke `xtrsipcfg_v2` to make a complex RSIP configuration.

   NOTICE: `xtrsipcfg_v2` can be run only when the CLE system is booted.

7.1.29 Update cray_rur Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray RUR (resource utilization reporting) service configuration worksheet to add site-specific data. RUR is a mechanism for collecting utilization data from nodes within a user application.

Procedure

1. Edit `cray_rur_worksheet.yaml`.

   ```
   smw# vi cray_rur_worksheet.yaml
   ```

2. Uncomment `cray_rur.enabled` and set it to `true`.

   No other settings need to be changed. However, this site may wish to configure the following settings.

3. (Optional) For systems that have GPUs, uncomment `cray_rur.settings.gpustat.data.enable` and set it to `true`.

   NOTICE: The KNC accelerator (GPU) is not supported with the CLE 6.x release.

4. (Optional) To collect memory data, uncomment `cray_rur.settings.memory.data.enable` and set it to `true`.

   CAUTION: This will generate a large amount of data on large systems or systems with a large throughput of applications.

5. (Optional) To use custom data plugins, add this group of settings.

   In the worksheet, copy the five lines below `# ** EXAMPLE 'data_plugins' VALUE (with current defaults) **` and paste them below `# NOTE: Place additional 'data_plugins' setting entries here, if desired`. Uncomment the lines, replace *`sample_key_a`* with a string that describes this particular data plugin, and ensure that the "enable" setting is set to `true`. Do not change the `null` value at the end of the first setting line.

   ```
   # NOTE: Place additional 'data_plugins' setting entries here, if desired.
   #  cray_rur.settings.data_plugins.data.plugin_name.sample_key_a: null <-- setting a multival key
   #  cray_rur.settings.data_plugins.data.sample_key_a.stage: none
   #  cray_rur.settings.data_plugins.data.sample_key_a.post: none
   #  cray_rur.settings.data_plugins.data.sample_key_a.arg: none
   #  cray_rur.settings.data_plugins.data.sample_key_a.enable: true
   #********************** END Service Setting: data_plugins **********************
   ```

6. (Optional) To use custom output plugins, add this group of settings.

   In the worksheet, copy the five lines below `# ** EXAMPLE 'output_plugins' VALUE (with current defaults) **` and paste them below `# NOTE: Place additional 'output_plugins' setting`

`entries here, if desired`. Uncomment the lines, replace *`sample_key_a`* with a string that describes this particular output plugin, and ensure that the "enable" setting is set to `true`. Do not change the `null` value at the end of the first setting line.

```
# NOTE: Place additional 'output_plugins' setting entries here, if desired.
#  cray_rur.settings.output_plugins.data.plugin_name.sample_key_a: null <-- setting a multival key
#  cray_rur.settings.output_plugins.data.sample_key_a.output: none
#  cray_rur.settings.output_plugins.data.sample_key_a.arg: none
#  cray_rur.settings.output_plugins.data.sample_key_a.enable: true
#********************** END Service Setting: output_plugins **********************
```

The configuration worksheet for the Cray ALPS service has two settings that must be configured if RUR is used. See *Update cray_alps Worksheet* on page 105.

7.1.30 Update cray_scalable_services Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Scalable Services configuration worksheet to add site-specific data. Scalable Services defines a tree of servers (nodes), starting with the server of authority (SoA), that are used in the scaling of the system. Configuration of Scalable Services is required for a functioning system.

Procedure

1. Edit `cray_scalable_services_worksheet.yaml`.

2. Uncomment `cray_scalable_services.enabled` and set it to `true`.

3. Uncomment `cray_scalable_services.settings.scalable_service.data.tier1` and set it to a list of all tier1 servers.

   Include in this list the cname of the boot node and any other nodes that have an Ethernet connection to the SMW. The SDB node should have a connection to the SMW, so it can be a tier1 server.

   IMPORTANT: If enabling boot node failover or SDB node failover, ensure that all boot nodes and all SDB nodes are in the list of tier1 nodes and none of them are in the list of tier2 nodes.

```
cray_scalable_services.settings.scalable_service.data.tier1:
- c0-0c0s0n1
- c0-0c1s1n1
```

4. Uncomment `cray_scalable_services.settings.scalable_service.data.tier2` and set it to a list of all tier2 servers.

```
cray_scalable_services.settings.scalable_service.data.tier2:
- c0-0c0s0n2
- c0-0c2s2n1
```

   Notes about tier2 nodes:

- There should be at least two tier2 nodes for moderately sized systems to provide resiliency. For large systems, check with Cray for the latest scalability recommendation.
- For larger systems, do not place tier2 nodes in the same cabinet.
- Compute nodes that were previously repurposed using the `xtcli mark_node service` command can be specified as tier2 servers.
- For optimum performance, do not place other services (e.g., login, RSIP, DVS, MOM) on tier2 nodes.
- Do NOT specify a login node as a tier2 node.
- Boot nodes and SDB nodes are tier1 servers and should not appear in the list of tier2 servers.

Check the guidance for tier2 nodes in this configuration worksheet for additional requirements or limitations.

7.1.31 Update cray_sdb Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Software Database (SDB) service configuration worksheet. The SDB service configures the services and settings for the SDB node.

Procedure

1. Edit `cray_sdb_worksheet.yaml`.
2. Uncomment `cray_sdb.enabled` and set it to `true`.
3. Uncomment `cray_sdb.settings.database.data.db_admin_password` and set the password.

   This password will be stored in clear text in the config set.
4. Leave `cray_sdb.settings.database.data.db_current_root_password` commented out (i.e., unconfigured).

   Leave this root password unconfigured so that the configurator will prompt for its setting when the config set is updated later in the process.

7.1.32 Update cray_service_node Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray service node configuration worksheet. This service configures the services and settings for service nodes.

Procedure

1. Edit `cray_service_node_worksheet.yaml`.

   ```
   smw# vi cray_service_node_worksheet.yaml
   ```

2. Uncomment `cray_service_node.enabled` and set it to `true`.

   No other settings need to be changed for a fresh install.

7.1.33 Update cray_simple_shares Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Simple Filesystem Sharing service configuration worksheet. The Simple Filesystem Sharing service quickly shares files between compute nodes that are connected to the high speed network (HSN).

Procedure

1. Edit `cray_simple_shares_worksheet.yaml`.
2. Uncomment `cray_simple_shares.enabled` and set it to `true`.
3. Uncomment `cray_simple_shares.settings.DVS.data./var/opt/cray/imps.spath: /var/opt/cray/imps`.

7.1.34 Update cray_simple_sync Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure changes the enable/disable setting in the Cray Simple Sync service configuration worksheet. Simple Sync is a mechanism for automatically distributing files to targeted locations on the Cray system.

Procedure

1. Edit `cray_simple_sync_worksheet.yaml`.

   ```
   smw# vi cray_simple_sync_worksheet.yaml
   ```

2. Uncomment `cray_simple_sync.enabled` and set it to `true`.

   No other settings need to be changed.

7.1.35 Update cray_ssh Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray SSH service configuration worksheet. The SSH service allows the system to be accessed through a secure shell.

Procedure

1. Edit `cray_ssh_worksheet.yaml`.

   ```
   smw# vi cray_ssh_worksheet.yaml
   ```

2. Uncomment `cray_ssh.enabled` and set it to `true`.

   No other settings need to be changed.

7.1.36 Update cray_storage Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Storage service configuration worksheet. The Storage service defines which storage set the current partition or system may use for persistent storage. Storage sets are defined in the global config set.

Procedure

1. Edit `cray_storage_worksheet.yaml`.

```
smw# vi cray_storage_worksheet.yaml
```

2. Uncomment `cray_storage.enabled` and set it to `true`.

3. Uncomment `cray_storage.settings.storage.data.active_storage_set` and set it to be the name of the CLE storage set in the `cray_bootraid` service, which is in the global config set.

   Use this command to show all storage sets defined in the global config set.

   ```
   smw# cfgset search -s cray_bootraid global |awk -F'.' '{print $5}' | sort -u
   ```

4. (For reinstall only) Uncomment `cray_storage.settings.storage.data.zero_volumes_on_create` and set it to true if this system is reinstalling to a CLE storage set that had been in use previously.

7.1.37 Update cray_sysconfig Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray System Configuration service configuration worksheet. The System Configuration service controls the configuration (settings and values) of files in `/etc/sysconfig`.

Procedure

1. Edit `cray_sysconfig_worksheet.yaml`.

2. Uncomment `cray_sysconfig.enabled` and set it to `true`.

   No other settings need to be changed.

7.1.38 Update cray_sysenv Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray System Environment service configuration worksheet. The System Environment service specifies values to be defined within the system environment.

Procedure

1. Edit `cray_sysenv_worksheet.yaml`.

2. Uncomment `cray_sysenv.enabled` and set it to `true`.

3. Uncomment `cray_sysenv.settings.system.data.syslog_socket_queue_length.value` and change its value to `'512'` (be sure to use the single quotes in the worksheet).

```
cray_sysenv.settings.system.data.syslog_socket_queue_length.value: '512'
```

7.1.39 Update cray_time Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray Time service configuration worksheet. The Time service configures time settings such as time zone and what NTP servers will be used to sync time with the outside world. This service has a global template as well as a CLE template, and therefore it can be configured to inherit settings from the global config set.

Procedure

1. Edit `cray_time_worksheet.yaml`.

2. Uncomment `cray_time.inherit` and set it to `true`.

   This means that time settings in the global config set will be used instead of time settings in the CLE config set.No other settings need to be changed.

7.1.40 Update cray_user_settings Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray User Settings service configuration worksheet. The User Settings service sets the environment modules that should be loaded automatically when a user logs in to the SMW, login node, or service nodes.

Procedure

1. Edit `cray_user_settings_worksheet.yaml`.

2. Uncomment `cray_user_settings.enabled` and set it to `true`.

No other settings need to be changed for a fresh install.

As other software is installed later, it might be necessary to change the set of module files loaded by default on SMW, login, and service nodes.

7.1.41 Update cray_wlm_detect Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray WLM (workload manager) Detect service configuration worksheet to add site-specific data. The WLM Detect service is a C library and command used to identify the native WLM on the system. If this service is not configured, the system uses ALPS as the workload manager by default.

Procedure

1. Edit `cray_wlm_detect_worksheet.yaml`.
2. Uncomment `cray_wlm_detect.enabled` and set it to `true`.

   No other WLM Detect settings need to be changed for a fresh install.

7.1.42 Update cray_wlm_trans Worksheet

Prerequisites

This procedure assumes that a work area has been set up for editing CLE configuration worksheets and that editing commands are invoked from that work area.

About this task

This procedure configures some basic settings in the Cray WLM (workload manager) Trans service configuration worksheet to add site-specific data. The WLM Trans service is a library that provides WLM-agnostic functions for common tasks such as setting node state and getting a list of jobs being run by a user.

Procedure

1. Edit `cray_wlm_trans_worksheet.yaml`.
2. Uncomment `cray_wlm_trans.enabled` and ensure that it is set to `true`.

   No other WLM Trans settings need to be changed for a fresh install.

7.2 Create New CLE Config Set from Worksheets

Prerequisites

This procedure assumes that worksheets have been obtained, copied to a work area outside of `/var/opt/cray/imps/config/sets/`*`CONFIG_SET_NAME`*`/worksheets`, and modified to include site-specific configuration data.

About this task

This procedure creates a new CLE config set from existing CLE configuration worksheets. Use the following commands, depending on whether this system is partitioned or not. Note that the worksheet path provided must be enclosed in single quotes because of the file glob used. There is no need to specify the config set type because the default is type CLE.

full system Create a config set for a full (unpartitioned) system p0:

```
smw# cfgset create --worksheet-path \
'/var/adm/cray/release/p0_worksheet_workarea/*_worksheet.yaml' p0
```

partitioned Create a config set for each partition. For partition p1:

```
smw# cfgset create --worksheet-path \
'/var/adm/cray/release/p1_worksheet_workarea/*_worksheet.yaml' p1
```

For partition p2:

```
smw# cfgset create --worksheet-path \
'/var/adm/cray/release/p2_worksheet_workarea/*_worksheet.yaml' p2
```

Additional partitions follow the same pattern.

7.3 Update CLE Config Set

Prerequisites

This procedure assumes that one or more config sets have been created.

About this task

This procedure updates the CLE config set(s) to do the following:

- Set encrypted passwords for root, crayadm, and the SDB database root account.
- Check whether any required or basic settings were missed earlier in the process.

In addition, it is common for the configurator, when it gets to the Cray LNet service, to prompt for the value of the flat_routes setting or the fgr_routes setting or both, depending on which is not being used for external Lustre servers or whether direct-attached Lustre (DAL) is used.

Procedure

1. Invoke `cfgset` to update the config set.

   full system Update the config set for a full (unpartitioned) system p0:

   ```
   smw# cfgset update p0
   ```

   partitioned Update the config set for each partition. For partition p1:

   ```
   smw# cfgset update p1
   ```

   For partition p2:

   ```
   smw# cfgset update p2
   ```

   Additional partitions follow the same pattern.

2. Set root and crayadm passwords.

   Enter values for these settings when prompted by the configurator. The first two settings, from the `cray_local_users` service, are for CLE/Linux accounts. They are of type "protected," which means that they must be entered twice (the second time for confirmation), they are not displayed while being entered, and they are stored in the config set in encrypted form. To enter or change the value of a protected setting, enter **+** to modify it, then enter and re-enter the password at the prompts.

   `cray_local_users.settings.users.data.root.crypt` (the root password for the system, a protected setting in the `cray_local_users` service)

   `cray_local_users.settings.users.data.crayadm.crypt` (the password for the crayadm account, a protected setting in the `cray_local_users` service)

   `cray_sdb.settings.database.data.db_current_root_password` (the password for the SDB database root account, a setting in the `cray_sdb` service)

   Not prompted for all of these? If the configurator did not prompt for some or all of these settings, wait until `cfgset` finishes, then run `cfgset` in interactive mode, and select and set these settings from the `cray_local_users` and `cray_sdb` services.

   ```
   cfgset update -m interactive p0
   ```

   For more information about using the configurator, see XC™ Series Configurator User Guide.

3. Enter values for any other settings presented by the configurator.

   If no more settings are presented, it means that all required and basic settings have been set.

When the configurator is done, it displays a message indicating the file name of the changelog file for this configuration session. The changelog is written to a file in the `/var/opt/cray/imps/config/sets/global/changelog` directory.

7.4 Check CLE Hostnames in /etc/hosts File

Prerequisites

This procedure assumes that the CLE config set has been created and updated.

About this task

This procedure checks the `/etc/hosts` file to make sure that hostname entries were added when the CLE config set was updated.

Procedure

1. Confirm that hostname entries exist in the CLE `/etc/hosts` file for boot, sdb, login, lnet, rsip, dvs, and any other names defined on this system.

   full system For a full, unpartitioned system:

   ```
   smw# egrep "boot|sdb|login|lnet|rsip|dvs" /var/opt/\
   cray/imps/config/sets/p0/files/roles/common/etc/hosts
   ```

   partitioned For a partitioned system, with partitions p1 and p2:

   ```
   smw# egrep "boot|sdb|login|lnet|rsip|dvs" /var/opt/cray/\
   imps/config/sets/p1/files/roles/common/etc/hosts
   ```

   ```
   smw# egrep "boot|sdb|login|lnet|rsip|dvs" /var/opt/cray/\
   imps/config/sets/p2/files/roles/common/etc/hosts
   ```

   Trouble? If any expected hostnames are missing, update the config set again (see *Update CLE Config Set* on page 146).

2. Add external hostnames and IP addresses to CLE `/etc/hosts file`.

   If there are any additional site hostnames and IP addresses that need to be in the CLE nodes' `/etc/hosts` file, then add them to this file. These additional entries and any comments will be retained every time the `cfgset update p0` command is run.

   ```
   smw# vi /var/opt/cray/imps/config/sets/p0/files/roles/common/etc/hosts
   ```

7.5 Update /etc/motd for Nodes

About this task

The standard `/etc/motd` on CLE nodes has this information.

```
Identity of node
Compute or service node
Boot image
Size of boot image
CLE release and build
Core and memory info
```

To add a custom message to the standard message of the day for all nodes, use the common role to distribute the `/etc/motd` file to all nodes.

Procedure

Edit the message of the day to add the custom message.

```
smw# vi /var/opt/cray/imps/config/sets/p0/files/roles/common/etc/motd
```

7.6 Copy Files for External Lustre Fine-grained Routing

Prerequisites

This procedure assumes the following:

- The Cray LNet service (cray_lnet) has been configured with fine-grained routing (FGR).
- `ip2nets.conf` and `routes.conf` files have been generated using the `lustre_utils clvft` command.

About this task

This procedure places the `ip2nets.conf` and `routes.conf` files where the LNet routers can use them.

Procedure

1. Create an `lnet` directory under roles in the `p0` config set directory structure for the `ip2nets.conf` and `routes.conf` files.

   ```
   smw# mkdir -p /var/opt/cray/imps/config/sets/p0/files/roles/lnet
   ```

2. Change to the directory where the `ip2nets.conf` and `routes.conf` files reside and copy them to the new LNet directory.

   ```
   smw# cd directory_containing_ip2nets.conf_and_routes.conf
   smw# cp -p ip2nets.conf routes.conf /var/opt/cray/imps/config/sets/p0/files/
   roles/lnet
   ```

7.7 Configure Files for Cray Simple Sync Service

About this task

Cray Simple Sync provides a generic mechanism to automatically distribute files to targeted locations on the system. When enabled, the Simple Sync service is executed on all CLE nodes at boot time and whenever the administrator executes `/etc/init.d/cray-ansible start` on a CLE node. When Simple Sync is executed, files placed in the following directory structure are copied to the root file system (`/`) on the target nodes. Create whatever directory structure is needed to place the target file(s) in the proper location.

The root directory for Simple Sync v2 is as follows:
`smw:/var/opt/cray/imps/config/sets/<config_set>/files/simple_sync/`

./common/files/	# copies to all nodes
./platform/[compute, service]/files/	# copies to all compute or service nodes
./hardwareid/<hardwareid>/files/	# copies to nodes with matching hardware id
./hostname/<hostname>/files/	# copies to nodes with matching hostname
./nodegroups/<node_group_name>/files/	# copies to members of <node_group_name>
./README	

Any directory structure and files below `./files/` in the Simple Sync directory structure on the SMW is replicated on the target node starting at `/`. For example,
`smw:/var/opt/cray/imps/config/sets/<config_set>/files/simple_sync/common/files/etc/myapplicat`
will be placed on all nodes as `/etc/myapplication.conf`.

The ownership and permissions of files in the config set is preserved in the copies made to nodes. For more information and use cases, see *About Simple Sync* on page 150.

7.7.1 About Simple Sync

The Cray Simple Sync service (cray_simple_sync) provides a simple, easy-to-use, generic mechanism for administrators to make configuration changes to their system without resorting to writing a custom Ansible play. When enabled, the service automatically copies files found in source directories in the config set on the SMW to one or more target nodes. Simple Sync is a simple tool and not intended as the sole solution for making configuration changes to the system. Writing custom Ansible plays might provide better maintainability, flexibility and scalability in the long term.

The Simple Sync service is enabled by default and has no additional configuration options. It can be enabled or disabled during the initial installation using worksheets or with the `cfgset` command at any time.

```
smw# cfgset update --service cray_simple_sync --mode interactive <config_set_name>
```

For more information, see man cfgset(8).

How Simple Sync Works

When enabled, Simple Sync is executed on all CLE nodes at boot time and whenever the site administrator executes `/etc/init.d/cray-ansible start` on a CLE node. When Simple Sync is executed, files placed in the following directory structure are copied onto nodes that match these criteria:

`smw:/var/opt/cray/imps/config/sets/<config_set>/files/simple_sync/`

./common/files/	Matches all nodes.
./hardwareid/<hardwareid>/files/	Matches a specific node with that hardware ID, which is the cname of a CLE node or the output of the `hostid` command (e.g., 1eac0b0c) on other nodes. An admin must create both the `<hardwareid>` directory and the `files` directory.
./hostname/<hostname>/files/	Matches a node with the specified hostname. An admin must create both the `<hostname>` directory and the `files` directory.
./nodegroups/<node_group_name>/files/	Matches all nodes in the specified node group. The directories for this `nodegroups` directory are automatically stubbed out when the config set is updated after node groups are defined in configured in the cray_node_groups service.
./platform/[compute, service]/files/	Matches all compute nodes or all service nodes, depending on whether they are placed in `platform/compute/files` or `platform/service/files`. Each time the config set is updated, the HSS data store is queried to update which nodes are service and which are compute.
./README	Provides brief guidance on using Simple Sync and a list of existing node groups in the order in which files will be copied. This ordering enables an administrator to predict behavior in cases where a file may be duplicated within the Simple Sync directory structure.

Simple Sync copies content into place prior to the standard Linux startup (`systemd`) and before `cray-ansible` runs any other services. As a result, Cray services that make small changes to files will operate on the administrator-provided file. Afterwards, the file will contain both non-conflicting administrator-provided content as well as the changes made by the Cray service. Because these changes happen prior to Linux startup, the changes will be in place when the services start up.

Note that there are some config files that are entirely managed by Cray services. Where possible, such files have a comment at the top indicating that the file is completely under the management of the Cray service. Files that have been changed by Cray services can be identified by checking the change logs on the running node in `/var/opt/cray/log/ansible`. Simple Sync does not provide a mechanism to override changes made by Cray services. To override changes made by Cray services, refer to the documentation for the specific service.

The ownership and permissions of copied directories and files are preserved when they are copied to `/` on the matching target nodes. An administrator can run cray_ansible multiple times, as needed, and only the files that have changed will be copied to the target nodes.

Because of the way it works, Simple Sync can be used to configure services that have configuration parameters not currently supported by configuration templates and worksheets. An administrator can create a configuration

file with the necessary settings and values, place it in the Simple Sync directory structure, and it will be distributed and applied to the specified node(s).

Characteristics of Simple Sync

Simple Sync is:	Simple Sync is NOT:
for simple and straightforward use cases	a comprehensive system management solution
for copying a moderate number of moderately sized files	intended to transfer large objects or a large volume of files
	an interface to configure Cray "turnkey" services such as ALPS, Node Health or Lightweight Log Manager (LLM)

Simple Sync was introduced with the CLE 6.0.UP00 / SMW 8.0.UP00 release. For this release, Simple Sync (Simple Sync v2) has been enhanced to:

- run as early in the Ansible execution sequence as possible (it now runs BEFORE other cray-ansible plays, so it can be used to make changes to files that Cray updates, like `sshd_config`)
- run during the Netroot setup sequence so it can be used to change LNet and DVS settings, if needed
- support Node Groups for targeting which system nodes to copy files to (the Node Groups feature is not yet complete, but it can be used with Simple Sync—see *About Node Groups* on page 134)

Simple Sync v2 still does not support:

- removing files
- appending to files
- changing file ownership and permissions (the permissions of the file in the config set are mirrored on-node)
- backing up files
- overriding Cray-set values (it cannot be used to change files that Cray completely overwrites, such as `alps.conf`, or change values in files that Cray modifies such as PermitRootLogin in /etc/ssh/sshd_config)

Note that the original Simple Sync root directory was different: `smw:/var/opt/cray/imps/config/sets/<config_set>/files/roles/simple_sync`.

Using Simple Sync with eLogin

Sites that want to use Simple Sync on eLogin nodes should note the following issues:

- eLogin uses the original Simple Sync in CLE 6.0UP01, not Simple Sync v2. This means content destined for the eLogin nodes must be placed in the old Simple Sync directory structure.
- Simple Sync v1 runs late in the sequence, overwriting content put into place by Cray services.
- Simple Sync v1 does not support node groups.

Cautions about the Use of Simple Sync

- Simple Sync copies files from the config set, which in the case of nodes without a persistent root file-system is cached in a compressed form, locally, in memory. As a result, each file stored in the config set uses some memory on the node. Therefore, using Simple Sync to copy binary files or large numbers of files is inadvisable.

- Be aware of differences in node environments when using Simple Sync. For example, systems configured with direct-attached Lustre (DAL) have nodes running CentOS instead of SLES. Administrators would have to be very careful to avoid putting an inappropriate configuration file into place when using the Simple Sync platform/service target in such a situation.
- Storage and distribution of verbatim config files through Simple Sync creates the potential for unintentional impact to the system when config files evolve due to software changes. Making minimal necessary changes through a site-local Ansible playbook provides more flexibility and minimizes the potential for unintended consequences.

Use Cases

Copy a non-conflicting file to all nodes

1. Place `etc/myfile` under `./common/files/` in the Simple Sync directory structure.
2. Simple Sync copies it to `/etc/myfile` on all nodes.

Copy a non-conflicting file to a service node

1. Place `etc/servicefile` under `./platform/service/files/` in the Simple Sync directory structure.
2. Simple Sync copies it to `/etc/servicefile` on all service nodes.

Copy a non-conflicting file to a compute node

1. Place `etc/computefile` under `./platform/compute/files/` in the Simple Sync directory structure.
2. Simple Sync copies it to `/etc/computefile` on all compute nodes.

Copy a non-conflicting file to a specific node

1. Place `etc/mynode` under `./hostname/c0-0c0s0n0/files/` in the Simple Sync directory structure.
2. Simple Sync copies it to `/etc/mynode` on c0-0c0s0n0.

Copy a non-conflicting file to a user-defined collection of nodes

1. Create a node group called "my_nodes" containing a list of nodes.
2. Update the config set.

```
smw# cfgset update p0
```

3. Place `etc/mynodes` under `./nodegroups/my_nodes/files/` in the Simple Sync directory structure.
4. Simple Sync copies it to `/etc/mynodes` on all nodes listed in node group my_nodes.

Copy to a node a file that has Cray-maintained content

1. Place a version of `sshd_config` with the value “PermitEmptyPasswords yes” under `./nodegroups/login/files/etc/ssh/` in the Simple Sync directory structure.
2. The booted system will contain both:
   - “PermitEmptyPasswords yes” (from the file copied by Simple Sync)
   - “PasswordAuthentication yes” (from modification of file by Cray)

Copy to a node a file that is exclusively maintained by Cray

Files exclusively maintained by Cray such as `alps.conf` cannot be updated using Simple Sync. Please refer to the owning service (such as ALPS) for information on how to update the contents.

Copy to a node a file that resides on a file system that will be mounted during Linux boot

No special operational changes are necessary. However, Simple Sync will put the file in place early in the boot sequence, and then it will be over-mounted by the file system. Because Simple Sync runs again later, it will copy the file into the mounted file system. Due to the ordering of operations, the file will not be present between the time the file system was mounted until the late execution of Ansible.

On Netroot login nodes, modify an LNet modprobe parameter

1. Generate a file `zz_lnet.conf` containing `options lnet router_ping_timeout=100`.
2. Place `zz_lnet.conf` under `./nodegroups/login/files/etc/modprobe.d/` in the Simple Sync directory structure.
3. The `lnet router_ping_timeout` value will be 100.

Note that normally Simple Sync does not allow the user to override Cray values, but this procedure takes advantage of the standard Linux mechanism to override Kernel module options.

Copy a file with an incompatible content to a node file that has Cray-maintained content

While Simple Sync allows an administrator to make changes to the same configuration files as modified by Cray, be very careful to avoid introducing syntax errors or incompatible values that may cause the system to fail to operate correctly.

7.8 Display and Capture all Config Set Information

About this task

This procedure displays all of the configuration settings in a config set and captures them in a typescript file of this software update. It is not required, but it may aid in troubleshooting.

Procedure

Display all configuration settings in the CLE and global config sets, and capture them in a typescript file.

full system Display/capture full information (more verbose) for a full, unpartitioned system:

```
smw# cfgset search -l advanced --format full p0 | tee /var/adm/\
cray/release/p0.${TODAY}.fresh_install.advanced.conf.full

smw# cfgset search -l advanced --format full global | tee /var/adm/\
cray/release/global.${TODAY}.fresh_install.advanced.conf.full
```

Display/capture just the settings and values for a full, unpartitioned system:

```
smw# cfgset search -l advanced p0 | tee /var/adm/cray/release/\
p0.${TODAY}.fresh_install.advanced.conf.full

smw# cfgset search -l advanced global | tee /var/adm/cray/release/\
global.${TODAY}.fresh_install.advanced.conf.full
```

partitioned Display/capture full information (more verbose) for a partitioned system, with partitions p1 and p2:

```
smw# cfgset search -l advanced --format full p1 | tee /var/adm/\
cray/release/p1.${TODAY}.fresh_install.advanced.conf

smw# cfgset search -l advanced --format full global | tee /var/adm/\
cray/release/global.${TODAY}.fresh_install.advanced.conf
```

```
smw# cfgset search -l advanced --format full p2 | tee /var/adm/\
cray/release/p2.${TODAY}.fresh_install.advanced.conf

smw# cfgset search -l advanced --format full global | tee /var/adm/\
cray/release/global.${TODAY}.fresh_install.advanced.conf
```

7.9 Validate Config Sets

About this task

It is important to validate any config set that has been modified, because there is currently no mechanism to prevent the system from trying to use an invalid config set. Validation is useful for determining if the config set is minimally viable for use with the system it is intended to configure.

> **IMPORTANT:** A validated config set does not necessarily equate to a config set with configuration data that will result in a properly configured system. This validation is only a basic check.

When validating a config set, the configurator checks the following:

- Config set has the proper directory structure and permissions.
- All configuration templates have correct YAML syntax.
- All configuration templates adhere to the configurator schema.

- All level `required` fields in enabled services are configured (i.e., their state is `set`).
- Pre-configuration and post-configuration callback scripts ran successfully during the latest config set update.

Procedure

Validate the CLE and global config sets.

full system Validate the p0 and global config sets for a full, unpartitioned system:

```
smw# cfgset validate p0
smw# cfgset validate global
```

partitioned Validate the CLE and global config sets for a partitioned system, with partitions p1 and p2:

```
smw# cfgset validate p1
smw# cfgset validate global
```

```
smw# cfgset validate p2
smw# cfgset validate global
```

7.10 Set Variable for Snapshot Name and Make a Snapshot Using snaputil

Prerequisites

This procedure assumes that the SMW has been booted to a release snapshot.

About this task

This procedure uses `snaputil` to make a snapshot. `snaputil` is a Python utility delivered with the cray-install-support RPM, which is installed by default on the SMW. It enables administrators and installers to create and manage snapshots as an easy way to stage upgrades and roll back or switch between SMW/HSS and CLE releases.

Note that snapshots capture content in these three directories: `/`, `/var/lib/mysql`, and `/var/opt/cray/repos`. Used in conjunction with backups of config sets, they provide enough information to be able to re-create the state of the system at a particular time.

Sites can make as few or as many snapshots as they deem useful. Cray recommends making a snapshot of the system at the software installation/configuration milestones shown in step 3.

Procedure

1. List the available snapshots on the system.

   ```
   smw# snaputil list
   ```

2. Set the `SNAPSHOT` environment variable using the currently booted snapshot name.

Setting a variable for the snapshot name enables better command substitution in later commands dealing with snapshots. The SMW will boot from this snapshot. (SMW HA only) This is especially important for SMW HA systems because it makes it easier to use the exact same snapshot name for both SMWs. Using different snapshots results in HSS database (MySQL) inconsistencies, which causes problems at failover.

```
smw# export SNAPSHOT=$(snaputil list |grep ^cur| awk '{print $2}')
smw# echo $SNAPSHOT
```

3. Create a new snapshot.

```
smw# snaputil create ${SNAPSHOT}.suffix
```

Replace *suffix* with a unique suffix to distinguish among snapshots. Here are suggested suffixes:

Suffix	Associated Milestone
pre-config	after installing a new software release
post-config	after updating the global and CLE config sets
post-boot	after booting the system
post-customize	after customizing a preinstalled system

.

7.11 Back Up Current Global and CLE Config Sets

About this task

Sites can back up the current global and CLE config sets as few or as many times as they deem useful. Cray recommends backing up the config sets at these software installation/configuration milestones.

In the example commands below, replace *suffix* with a unique suffix to distinguish among snapshots. Here are suggested suffixes:

Suffix	Associated Milestone
post-config	after updating the global and CLE config sets
post-boot	after booting the system
post-customize	after customizing a preinstalled system

Procedure

1. Back up the current global config set.

```
smw# cfgset create --clone global global-suffix-${TODAY}
```

2. Back up the current CLE config set.

```
smw# cfgset create --clone p0 p0-suffix-${TODAY}
```

8 Boot the System

Most system configuration is now complete, and this part of the installation and configuration process has procedures to prepare boot images, make sure those images are mapped to node correctly, boot the system, and perform post-boot activities such as running tests and making a snapshot. However, sites must first decide where to place their root file system, because some of these procedures depend on that decision.

1. Decide where to place the root file system using *Where to Place the Root File System* on page 158.
2. Prepare boot images and NIMS maps.
   a. *Create a NIMS Map* on page 159
   b. *About Image Groups and How to Customize Them* on page 159
   c. *Build Boot Images* on page 161
   d. *Check NIMS Information* on page 164
3. *Boot the System using a Boot Automation File* on page 165
4. Perform post-boot activities.
   a. *Run Tests After Boot is Complete* on page 166
   b. *Run the xtgetrev and xtshowrev Tools* on page 167
   c. *Create a Dump with xtdumpsys* on page 169
   d. *Set Variable for Snapshot Name and Make a Snapshot Using snaputil* on page 88 (post-boot)
   e. *Back Up Current Global and CLE Config Sets* on page 157

8.1 Where to Place the Root File System

Use the information provided here to decide whether to use Netroot for some or all compute and login nodes at this site.

About Netroot and Dynamic Shared Objects and Libraries (DSL)

In releases prior to CLE 6.0 / SMW 8.0, the dynamic shared objects and libraries (DSL) feature was optional. It was necessary for many sites because it enabled both dynamic shared libraries and large network-based images, which were needed for systems with NVIDIA GPUs and for most production workloads.

In the current release, DSL is supported by default. Note, however, that the DSL feature no longer includes provision for large network-based images. That capability is now provided by Netroot. Sites that require large network-based images and additional storage should use Netroot.

8.2 Create a NIMS Map

About this task

On a fresh installation a new NIMS (Node Image Mapping Service) map needs to be created. This procedure creates a NIMS map and designates it as the active map.

Procedure

Create a NIMS map and set it as active.

full system For a full, unpartitioned system:

```
smw# cmap create p0
smw# cmap setactive p0
```

partitioned For a partitioned system, with partitions p1 and p2:

```
smw# cmap create p1 --partition p1
smw# cmap setactive p1  -p p1
```

```
smw# cmap create p2 -p p2
smw# cmap setactive p2  -p p2
```

8.3 About Image Groups and How to Customize Them

Image group configuration information is used by the `imgbuilder` command to build boot images. Image groups are defined in the global config set in the cray_image_groups configuration file (`/var/opt/cray/imps/config/sets/global/config/cray_image_groups.yaml`). Here is an example of the contents of that file:

```
cray_image_groups:
  default:
      - recipe: "compute_cle_6.0up01_sles_12_x86-64_ari"
        dest: "compute{note}_cle_{cle_release}-build{cle_build}{patch}_sles_12-
created{date}.cpio"
        nims_group: "compute"
      - recipe: "login_cle_6.0up01_sles_12_x86-64_ari"
        dest: "login{note}_cle_{cle_release}-build{cle_build}{patch}_sles_12-
created{date}.cpio"
        nims_group: "login"
      - recipe: "service_cle_6.0up01_sles_12_x86-64_ari"
        dest: "service{note}_cle_{cle_release}-build{cle_build}{patch}_sles_12-
created{date}.cpio"
        nims_group: "service"
      - recipe: "diag-all_cle_6.0up01_sles_12_x86-64_ari"
        dest: "diag-all_cle_60up01_sles_12_x86-64_ari"
  ...
```

```
  testing:
      - recipe: "compute_cle_6.0up01_sles_12_x86-64_ari"
        dest: "{my_custom_prefix}_compute-TEST-
{my_other_value}_{date}_{time}.cpio"
        nims_group: "compute-test"
```

Sites can modify this information to fit their needs by editing this YAML file directly.

```
smw# vi /var/opt/cray/imps/config/sets/global/config/cray_image_groups.yaml
```

The following sections describe important things to know to successfully customize the `cray_image_groups` configuration file.

What image groups are made of

- The cray_image_groups configuration file can contain multiple *image groups* (this example shows two). When invoked, `imgbuilder` builds one of these. It builds "default" if no image group name is passed as a parameter.
- Each image group contains a list of *image specifications* that will be built: by default, the standard compute, service, and login images.
- Each image specification contains these three fields:

  `recipe` An IMPS (Image Management and Provisioning System) image recipe name. This can be customized to specify which image recipe is used to build a specific boot image.

  `dest` The destination filename used for both the IMPS image root and bootable cpio. This can be customized as described below.

  `nims_group` The NIMS group to map this image to. The nims_group field is specified only for images that are intended as boot images.

How to customize an image name using placeholders

The `dest` values in the above example contain bracketed placeholder strings. At build time, relevant values are substituted for these placeholders. Currently, `imgbuilder` supports the following built-in placeholders for use in the cray_image_groups configuration file:

{date}	the current system date (e.g., 20140314)
{time}	the current system time (e.g., 134514)
{host}	the current system hostname
{user}	the current username
{cle_release}	the currently active CLE release
{cle_build}	the currently active CLE build

User-defined placeholders (optional) are also supported for further customization of image names. If this site wants to define custom placeholders, with substitution at runtime, then when running `imgbuilder`, use two dashes and a space (`--`) followed by any number of placeholder definitions as key/value pairs separated by spaces.

For example, this command would tell `imgbuilder` to build the images in the "testing" image group, map them to the NIMS groups specified in that group, and substitute "foo" everywhere for "my_custom_prefix" and "bar" everywhere "my_other_value" appears.

```
(example only, do not use) smw# imgbuilder --map --image-group testing \
-- my_custom_prefix=foo my_other_value=bar
```

IMPORTANT: When the value of `dest` contains one or more placeholders, the whole expression must be enclosed by double quotes.

8.4 Build Boot Images

Prerequisites

This procedure assumes some knowledge of image groups: how they are defined and how they can be customized for a site. See *About Image Groups and How to Customize Them* on page 159.

About this task

This procedure uses the `imgbuilder` command to build boot images. The `imgbuilder` command uses information in the cray_image_groups configuration file (`/var/opt/cray/imps/config/sets/global/config/cray_image_groups.yaml`) to know which images to build, how to build them, what to call the built images, and which NIMS (Node Image Mapping Service) groups to map those images to. To successfully map an image to a node, `imgbuilder` also needs to know which NIMS group that node belongs to, which means the node must have its NIMS group (i.e., its "group" field) populated. For an initial fresh install, that field may not be populated yet. To ensure that the required node information gets populated prior to building boot images, the `--bootstrap-nims` option must be used. With the bootstrap option, `imgbuilder` looks at the "group" field of each node. If group is empty, `imgbuilder` adds "compute" or "service" depending on the "type" of that node, as reported by the HSS (Hardware Supervisory System).

When invoked, the `imgbuilder` command builds all of the image specifications from one of the image groups defined in the cray_image_groups configuration file. To specify which image group to build, use the `--image-group` option. If that option is not used, `imgbuilder` will build the group called "default." When `imgbuilder` is invoked with the `--map` option, it maps the image in each image specification to the associated NIMS group (the `nims_group` field). For more information, see the `imgbuilder` man page.

Procedure

1. Bootstrap NIMS (Node Image Mapping Service) using `imgbuilder` with the bootstrap option.

   full system For a full, unpartitioned system:

   ```
   smw# imgbuilder --bootstrap-nims
   ```

   partitioned For a partitioned system, with partitions p1 and p2:

   ```
   smw# imgbuilder --bootstrap-nims -p p1
   ```

```
smw# imgbuilder --bootstrap-nims -p p2
```

All nodes have now been assigned to the NIMS service or compute group (i.e., have their "group" field set to either "service" or "compute").

2. Load and install any released patches.

   Check for CLE 6.0.UP01 and SMW 8.0.UP01 patches in `/var/adm/cray/release/patchsets`. Follow the instructions provided for loading and installing the patches, with these differences:

   a. Temporarily suppress building and mapping images.

      It is very important to defer building the images until after the login and DAL nodes have been assigned in steps 3 and 4 and any changes to the default image group have been made in step 5. Setting these two variables prevents the patch scripts from creating images and mapping them to NIMS at this point. All image creation and NIMS mapping will be done during step 6.

      ```
      smw# export PATCHSET_BUILD_IMAGES=false
      smw# export PATCHSET_NIMS_TIMING=deferred
      ```

   b. Follow instructions in patch README files.

      Follow all of the instructions in CLE and SMW patch README files. These instructions will include running the LOAD script and the INSTALL script for each patch, and there may be additional steps for some of these patches, such as running `xtzap` again to update firmware from an SMW patch.

3. Assign login nodes to the login group.

   Assign login nodes to the NIMS login group so that they will be assigned the login boot image for booting. To assign more than one node, use a space-separated list of nodes.

   full system For a full, unpartitioned system, remove from the NIMS service group and add to the NIMS login group:

   ```
   smw# cnode update -G service -g login c0-0c0s1n1
   ```

   partitioned For a partitioned system, with partitions p1 and p2, remove from the NIMS service group and add to the NIMS login group:

   ```
   smw# cnode update -G service -g login c0-0c0s1n1 -p p1
   ```

   ```
   smw# cnode update -G service -g login c0-0c0s3n2 -p p2
   ```

4. For systems using direct-attached Lustre (DAL), assign DAL nodes to the NIMS DAL group.

   Assign DAL service nodes to the NIMS `dal` group so that they are assigned the DAL boot image for booting. To assign more than one node, use a space-separated list of nodes.

   full system For a full, unpartitioned system, remove from the NIMS service group and add to the NIMS DAL group:

   ```
   smw# cnode update -G service -g dal c0-0c0s2n1 c0-0c0s2n2
   ```

   partitioned For a partitioned system, with partitions p1 and p2, remove from the NIMS service group and add to the NIMS DAL group:

```
smw# cnode update -G service -g dal c0-0c0s2n1 c0-0c0s2n2 -p p1

smw# cnode update -G service -g dal c0-0c0s2n1 c0-0c0s2n2 -p p2
```

5. Customize the cray_image_groups configuration file, as needed, by editing `/var/opt/cray/imps/config/sets/global/config/cray_image_groups.yaml`. Move recipe stanzas into the default group for anything to be built by default, and modify or create other image groups as appropriate for this site.

   a. For systems using DAL, ensure that this DAL stanza (image specification) are in the default image group, or customize and use the `tmpfs-w-dal` image group, which already has it.

   ```
       - recipe: "dal_cle_6.0up01_centos_6.5_x86-64_ari"
         dest: "dal{note}_cle_{cle_release}-build{cle_build}{patch}_centos_6.5-
   created{date}.cpio"
         nims_group: "dal"
   ```

   b. For systems using Netroot for either compute or login nodes, those images will be created at a later step in the process. See *Configure Netroot* on page 192.

6. Build images and map them to NIMS groups.

   Create a set of images as defined in `/var/opt/cray/imps/config/sets/global/config/cray_image_groups.yaml` and map them to the specified NIMS groups using the `--map` option.

   NOTICE:

   - Building images for service, compute, and login typically takes at least 15 minutes.
   - Building images for service, compute, login, and DAL typically takes at least 20 minutes.

   full system To build the images in the "default" image group and map them to the NIMS groups specified in "default" for a full, unpartitioned system:

   ```
   smw# imgbuilder --map
   ```

   partitioned To build the images in the "default" image group and map them to the NIMS groups specified in "default" for a partitioned system, with partitions p1 and p2:

   ```
   smw# imgbuilder --map -p p1 -p p2
   ```

7. (Alternative to step 6) Build images and then manually map them to NIMS groups.

   As an alternative to mapping the images using the `--map` option, that mapping can be done manually.

   a. Build the images.

   ```
   smw# imgbuilder
   ```

   b. Set the images for partition p1.

   Replace the cpio filename in these examples with the file names from the `imgbuilder` output in step a.

   ```
   smw# cd /var/opt/cray/imps/boot_images
   smw# ls -ltr
   ```

```
smw# cnode update -i compute_img.cpio --filter group=compute -p p1
smw# cnode update -i service_img.cpio --filter group=service -p p1
smw# cnode update -i login_img.cpio --filter group=login -p p1
smw# cnode update -i dal_img.cpio --filter group=dal -p p1
```

c. Set the images for partition p2.

Replace the cpio filename in these examples with the file names from the `imgbuilder` output in step a. Run these commands from the `/var/opt/cray/imps/boot_images` directory.

```
smw# ls -ltr
smw# cnode update -i compute_img.cpio --filter group=compute -p p2
smw# cnode update -i service_img.cpio --filter group=service -p p2
smw# cnode update -i login_img.cpio --filter group=login -p p2
smw# cnode update -i dal_img.cpio --filter group=dal -p p2
```

8.5 Check NIMS Information

About this task

This procedure lists NIMS (Node Image Mapping Service) information: which maps are active and what NIMS information is stored for each node.

Procedure

1. Check active NIMS maps.

full system For a full, unpartitioned system:

```
smw# cmap list
```

partitioned For a partitioned system, with partitions p1 and p2:

```
smw# cmap list --partition p1
smw# cmap list -p p2
```

2. Check NIMS information for each node.

full system For a full, unpartitioned system:

```
smw# cnode list
```

partitioned For a partitioned system, with partitions p1 and p2:

```
smw# cnode list -p p1
```

```
smw# cnode list -p p2
```

8.6 Boot the System using a Boot Automation File

Prerequisites

This procedure assumes that configuration and image preparation are complete and the system is now ready to boot.

About this task

This procedure describes how to boot with `xtbootsys` using a boot automation file.

Trouble? To troubleshoot a boot, use a different procedure instead: *Boot the System with DEBUG* on page 230.

Procedure

1. Create a site boot automation file.

   ```
   smw# cp -p /opt/cray/hss/default/etc/auto.generic \
   /opt/cray/hss/default/etc/auto.<site_system>.start
   ```

   Copy the Cray generic boot automation file and rename it. Add site customizations, as needed. For sites booting tmpfs images (instead of Netroot) with no SDB node failover, no changes may be necessary. Sites that choose to boot Netroot images will make those changes later in the process after this first boot with tmpfs.

2. Create a site automation file for shutting down the system.

   ```
   smw# cp -p /opt/cray/hss/default/etc/auto.xtshutdown \
   /opt/cray/hss/default/etc/auto.<site_system>.stop
   ```

   Copy the Cray shutdown automation file and rename it. Add site customizations, as needed.

3. Add boot node or SDB node failover to boot automation file.

   If either boot node failover or SDB node failover will be used, then the boot automation file should have a setting to ensure that STONITH has been enabled on the blade that has the primary boot node and the primary SDB node. The STONITH setting does not survive a power cycle. To maintain the STONITH setting, add these lines to the boot automation file.

   Use the blade which contains the primary boot node. If the primary boot node is c0-0c0s0n1, then the blade to use is c0-0c0s0. Add these lines **before** the line for booting the boot node.

   ```
   # Set STONITH for primary boot node
   lappend actions {crms_exec "xtdaemonconfig c0-0c0s0 stonith=true"}
   ```

Use the blade which contains the primary SDB node. If the primary SDB node is c0-0c1s0n1, then the blade to use is c0-0c1s0. Add these lines **before** the line for booting the SDB node.

```
# Set STONITH for primary SDB node
lappend actions {crms_exec "xtdaemonconfig c0-0c1s0 stonith=true"}
```

4. Run `xtbootsys` with `auto.<site_system>.start`.

   full system For a full, unpartitioned system:

   ```
   smw# su - crayadm
   crayadm@smw> xtbootsys -a auto.<site_system>.start
   ```

   partitioned For a partitioned system, with partitions p1 and p2:

   ```
   smw# su - crayadm

   crayadm@smw> xtbootsys -p p1 -a auto.<site_system>.start.p1

   crayadm@smw> xtbootsys -p p2 -a auto.<site_system>.start.p2
   ```

   The boot automation files have different names in this example to indicate that partitions may use different automation files, if needed.

8.7 Run Tests After Boot is Complete

Prerequisites

This procedure assumes that the system has completed booting.

About this task

Log in to the login node as crayadm. This can be done from the SMW to the boot node to the login node or directly from another computer to the login node without passing through the SMW and boot node. Then perform these rudimentary functionality checks.

Procedure

1. Run `apstat` to get the number of nodes to use for the following commands.

   ```
   crayadm@login> NUMNODES=$(($(apstat -v | grep XT | awk "{print \$3}")))
   crayadm@login> echo NUMNODES is $NUMNODES
   ```

2. Verify that all nodes run (from `/tmp`).

   ```
   crayadm@login> cd /tmp; aprun -b -n $NUMNODES -N 1 /bin/cat /proc/sys/kernel/
   hostname
   ```

3. Verify that the home directory is working by running a job.

```
crayadm@login> cd ~; aprun -b -n $NUMNODES -N 1 /bin/cat /proc/sys/kernel/
hostname
```

4. Verify that the Lustre directory is working by running a job.

```
crayadm@login> cd /lustre_file_system
crayadm@login> aprun -b -n $NUMNODES -N 1 /bin/cat /proc/sys/kernel/hostname
```

5. Run `xtcheckssd` to ensure that SMW databases have the current state of compute node SSDs.

```
root@login# pcmd -r -n ALL_COMPUTE "/opt/cray/ssd/bin/xtcheckssd"
```

   This needs to be done after an initial installation, SSD hardware change, system update, and periodically (daily/weekly).

8.8 Run the xtgetrev and xtshowrev Tools

Prerequisites

To run `xtgetrev`, the boot node must be booted and accessible.

About this task

System administrators use the `xtgetrev` and `xtshowrev` commands to gather and display machine, software revision, Field Notice (FN), and patch set information. The `xtgetrev` command collects information from the administrator and from the SMW and boot node. The `xtshowrev` command displays that information, even when CLE is not running. These tools are useful for gathering information to send to Cray (after a software upgrade, FN, or patch set is installed) and for troubleshooting.

This procedure describes how to install these two tools on a Cray XC Series system. These steps (except for running `xtshowrev`) must be executed as root.

> **ATTENTION:** Any information that is submitted to site_install_data@cray.com will only be used internally and will not be made public. The `xtshowrev` command does not submit any information to Cray automatically.

Procedure

1. Load the module to start using the tools.

```
smw# module load xtshowrev
```

2. Run `xtgetrev` to create and populate the initial files.

   Only root can run this command. The first time `xtgetrev` is executed, when there are no files populated, the tool will prompt for site information. If the boot node does not have passwordless ssh, then the tool will prompt for the password.

```
smw# xtgetrev
xtgetrev: No site information has been defined.
```

```
Site name: CRAY/INTERNAL
Serial Number: 9999
System Name [panda1]:
System Type [XC40]:
Install Date [2016-06-01]):
OS Type [CLE]:
/etc/opt/cray/release/pkginfo/site_config written.
No SMW information has been defined.
Install Date for SMW 8.0.UP01 [2016-06-01]):
/etc/opt/cray/release/pkginfo/smw_release written.
Accessing boot

No CLE information has been defined.
Which WLM is in production?
   1) PBS
   2) Moab/TORQUE
   3) SLURM
   4) LSF
   5) WLM
   6) None
   7) Skip
Choose?[7] 7
Install Date for CLE 6.0.UP01 [2016-06-01]:
/etc/opt/cray/release/pkginfo/cle_release written.
/etc/opt/cray/release/pkginfo/sec_release written.

No hardware inventory information has been defined.
Reading /opt/tftpboot/hwinv.p0 ...done
/etc/opt/cray/release/pkginfo/hwinv written.
```

3. Run `xtshowrev` to see the formatted information.

   Any user can run this command.

```
smw# xtshowrev
Site:                CRAY/INTERNAL
S/N:                 9999
System Type:         XC40
Install Date:        2016-06-01
System Name:         panda1
CNL/CLE Release:     6.0.UP01
XT Release:          6.0.96
CLE Kernel:          3.12.51-52.31.1_1.0600.9146
CLE OS:              SLES12
CLE Patch Sets:      01 02 03
CLE FNs:
Lustre Version:      2.5.4
OS Type:             CLE
SMW Release:         8.0.UP01
SMW Build:           8.0.96
HSS Release:         8.0__446__ge75851a-49.1
SMW Kernel:          3.12.51-52.39
SMW OS:              SLES12
SLE Patch Sets:
SMW Patch Sets:
SMW FNs:             5844c
SEC Release:         Cray_SEC 8.0__6__g689802a (sec 2.7.6)
Current Date:        2016-06-01 12:59:21
crayadm@smw>
```

8.9 Create a Dump with xtdumpsys

Prerequisites

This procedure assumes that the system has been booted.

Procedure

1. Start an `xtdumpsys` typescript.

   Start a new window. Start a typescript session for `xtdumpsys` in that new window.

```
smw# su - crayadm

crayadm@smw> export TODAY=`date +%Y%m%d`

crayadm@smw> . /etc/opt/cray/release/cle-release

crayadm@smw> mkdir -p /home/crayadm/dump/${TODAY}_${BUILD}

crayadm@smw> cd /home/crayadm/dump/${TODAY}_${BUILD}/

crayadm@smw> script -af hss.xtdumpsys
```

2. Start `xtdumpsys`.

   Start the dump, but do not press **Ctrl-d** until step 5.

   For a full system:

```
crayadm@smw> xtdumpsys
 INFO: Parsing '/opt/cray/dumpsys/1.0.0-1.0000.31025.41.1/bin/config/
default.conf'
 INFO: Beginning dump
 INFO: Gathering system partition information
 INFO: Gathering system hardware information
 INFO: No session specified, defaulting to current.
 INFO: Moving temporary log files to the dump directory.
 INFO:
################################################################################
 INFO: # Your dump is available in /var/opt/cray/dump/p0-YYYYMMDDtHHMMSS-
NNNNNNNNNN #
 INFO:
################################################################################
 Enter reason for dump:
 (an EOF terminates input, usually CTRL-D)
```

   For a partitioned system:

```
crayadm@smw> xtdumpsys  -p p1
crayadm@smw> xtdumpsys  -p p2
```

   When `xtdumpsys` asks for a dump reason, it will have created the dump directory. Do not continue the `xtdumpsys` at this time (do not press **Ctrl-d**); it will be continued in a later step.

3. Start a `cdump` typescript in a different window.

   Start another window. Start a typescript session for `cdump` in that window.

```
smw# su - crayadm

cdump crayadm@smw> export TODAY=`date +%Y%m%d`

cdump crayadm@smw> . /etc/opt/cray/release/cle-release

cdump crayadm@smw> cd /home/crayadm/dump/${TODAY}_${BUILD}/

cdump crayadm@smw> script -af hss.cdump
```

4. Dump a node with `cdump`.

   Change to the directory created in the `xtdumpsys` window (after `INFO: # Your dump is available in`), then use `cdump` to dump a compute node that successfully booted.

```
cdump crayadm@smw> cd /var/opt/cray/dump/p0-YYYYMMDDtHHMMSS-NNNNNNNNNN
cdump crayadm@smw> mkdir cdumps; cd cdumps
```

   For a full system:

```
cdump crayadm@smw> cdump -AmD -r xt-hsn@boot c0-0c0s3n0
 Wed Apr 24 09:06:32 CDT 2013 start cdump
 ...
makedumpfile Completed.
- done
Wed Apr 24 09:08:08 CDT 2013 cdump: # of nodes 1
 success 1
 failed  0
 skipped 0
cdump crayadm@smw> exit
```

   For a partitioned system, use the host name to specify which boot node. This example uses boot-p1 to `cdump` a node in the p1 partition.

```
cdump crayadm@smw> cdump -AmD -r xt-hsn@boot-p1 c0-0c0s4n0
cdump crayadm@smw> exit
```

5. Continue `xtdumpsys`: enter a reason.

   After `cdump` completes, return to the `xtdumpsys` window and enter a reason that starts with the string "mute:"

```
xtdumpsys window> mute: dumping test MMDD
```

   Then enter an end-of-file (**Ctrl-d**) to end the dump reason.

```
xtdumpsys window> <Ctrl-d>
 mute: dumping test 0424
 INFO: Dump reason:
 ...
 INFO:
################################################################################
 INFO: # Your dump is available in /var/opt/cray/dump/
p0-20130424t081927-1304240904 #
 INFO:
```

```
################################################################################
 INFO: No post-processing plugin found at '/etc/opt/cray/dumpsys/
postprocessing.py'
 INFO: Example plugins can be found at '/opt/cray/dumpsys/
1.2.5-1.0000.35873.20.1/bin/plugins/examples/postprocessing.py.*'
 INFO: Cleaning up
```

```
xtdumpsys crayadm@smw> exit
```

6. Remove dump directory, if desired.

   If there are no errors, it is probably safe to delete the dump directory.

```
xtdumpsys crayadm@smw> rm -rf /var/opt/cray/dump/pX-YYYYMMDDtHHMMSS-NNNNNNNNNN
crayadm@smw> exit
```

8.10 Set Variable for Snapshot Name and Make a Snapshot Using snaputil

Prerequisites

This procedure assumes that the SMW has been booted to a release snapshot.

About this task

This procedure uses `snaputil` to make a snapshot. `snaputil` is a Python utility delivered with the cray-install-support RPM, which is installed by default on the SMW. It enables administrators and installers to create and manage snapshots as an easy way to stage upgrades and roll back or switch between SMW/HSS and CLE releases.

Note that snapshots capture content in these three directories: `/`, `/var/lib/mysql`, and `/var/opt/cray/repos`. Used in conjunction with backups of config sets, they provide enough information to be able to re-create the state of the system at a particular time.

Sites can make as few or as many snapshots as they deem useful. Cray recommends making a snapshot of the system at the software installation/configuration milestones shown in step 3.

Procedure

1. List the available snapshots on the system.

```
smw# snaputil list
```

2. Set the `SNAPSHOT` environment variable using the currently booted snapshot name.

   Setting a variable for the snapshot name enables better command substitution in later commands dealing with snapshots. The SMW will boot from this snapshot. (SMW HA only) This is especially important for SMW HA systems because it makes it easier to use the exact same snapshot name for both SMWs. Using different snapshots results in HSS database (MySQL) inconsistencies, which causes problems at failover.

```
smw# export SNAPSHOT=$(snaputil list |grep ^cur| awk '{print $2}')
smw# echo $SNAPSHOT
```

3. Create a new snapshot.

```
smw# snaputil create ${SNAPSHOT}.suffix
```

Replace *suffix* with a unique suffix to distinguish among snapshots. Here are suggested suffixes:

Suffix	Associated Milestone
pre-config	after installing a new software release
post-config	after updating the global and CLE config sets
post-boot	after booting the system
post-customize	after customizing a preinstalled system

.

8.11 Back Up Current Global and CLE Config Sets

About this task

Sites can back up the current global and CLE config sets as few or as many times as they deem useful. Cray recommends backing up the config sets at these software installation/configuration milestones.

In the example commands below, replace *suffix* with a unique suffix to distinguish among snapshots. Here are suggested suffixes:

Suffix	Associated Milestone
post-config	after updating the global and CLE config sets
post-boot	after booting the system
post-customize	after customizing a preinstalled system

Procedure

1. Back up the current global config set.

```
smw# cfgset create --clone global global-suffix-${TODAY}
```

2. Back up the current CLE config set.

```
smw# cfgset create --clone p0 p0-suffix-${TODAY}
```

9 Configure Other Features and Services

At this stage in the installation/configuration process, the basic SMW and CLE software has been installed, configured, and booted. These procedures complete the configuration of a functional system.

Note that the system cannot be in use during the power management configuration procedure. If this is not a fresh install, CLE must be shut down before performing that procedure. All of the other procedures require CLE to be running.

REQUIRED	*Configure Power Management* on page 173
recommended	*Push Diags Image and Update the Diags Bind Mount Profile* on page 177
recommended	*Reduce Impact to SMW Performance of Btrfs Periodic Maintenance* on page 179
optional (Required if using DAL)	*Configure Direct-attached Lustre (DAL)* on page 179
optional	*LMT Configuration for DAL* on page 186 (Lustre Monitoring Tool for direct-attached Lustre)
optional (Required if using SEC)	*Configure the Simple Event Correlator (SEC)* on page 192
optional (Required if using SEDC)	*Configure System Environmental Data Collections (SEDC)* on page 192
optional (Required if using Netroot)	*Configure Netroot* on page 192
optional	*Prevent Unintentional Re-creation of Mail Configuration Files* on page 197

9.1 Configure Power Management

Prerequisites

This is a required step in bringing up a Cray XC system with CLE 6.0 UP01 and SMW 8.0 UP01. The PostgreSQL database on the SMW is needed even if a site will be using a remote (off-SMW) database node to store Power and SEDC data.

This procedure assumes that a disk drive is available for use as a dedicated drive for the PMDB. The drive should be physically located within the rack-mount SMW at slot 4. The drive should be of the specification 1 TB 7.2K RPM SATA 3Gbps 2.5in HotPlug Hard Drive 342-1998, per the SMW Bill of Materials. On a Dell PowerEdge R815 the device for PMDISK is `/dev/disk/by-path/pci-0000:05:00.0-sas-0x4433221103000000-lun-0`. On a Dell PowerEdge R630 the device for PMDISK is `/dev/disk/by-path/pci-0000:03:00.0-scsi-0:0:4:0`.

The system cannot be in use during this procedure. If this is not a fresh install, CLE must be shut down.

About this task

Power Management allows Cray® XC Series™ systems to operate more efficiently. By monitoring, profiling, and limiting power usage administrators can:

- Increase system stability by reducing heat dissipation
- Reduce system cooling requirements
- Reduce site cooling requirements
- Reduce utility costs by minimizing power usage when rates are the highest
- Respond to external environmental conditions and prevent power outages
- Calculate the actual power cost for individual users and/or jobs

CAUTION: Do not use this procedure in preparation for setting up an SMW HA system. As part of the HA configuration the `SMWHAconfig` copies the contents of the PMDB to a shared power management RAID disk. For more information see *XC™ Series SMW HA Initial Installation Guide (CLE 6.0.UP01)*.

These steps are performed as `root`.

Procedure

1. `su` to root.

```
crayadm@smw> su - root
smw#
```

2. Verify that the PMDISK is inserted into the SMW by entering the correct device name. This example, and the ones that follow, are for a Dell R815.

```
smw# fdisk -l /dev/disk/by-path/pci-0000:05:00.0-sas-0x4433221103000000-lun-0

Disk /dev/disk/by-path/pci-0000:05:00.0-sas-0x4433221103000000-lun-0: 931.5 GiB,
1000204886016 bytes, 1953525168 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x10692081
```

3. Create a new primary partition for the PMDISK, and write it to the partition table. If there are any existing partitions on this disk, manually delete them first.

```
smw# fdisk /dev/disk/by-path/pci-0000:05:00.0-sas-phy3-0x4433221103000000-lun-0
Welcome to fdisk (util-linux 2.25).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.
  Command (m for help): n
Partition type
  p   primary (0 primary, 0 extended, 4 free)
  e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-1953525167, default 2048): [press return]
```

```
Last sector, +sectors or +size{K,M,G,T,P} (2048-1953525167, default 1953525167): [press
return]

Created a new partition 1 of type 'Linux' and of size 931.5 GiB.

Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
```

4. Verify that the partition has been created. On a Dell R815 this should be device `/dev/disk/by-path/pci-0000:05:00.0-sas-phy3-0x4433221103000000-lun-0-part1` . On a Dell R630 this should be `/dev/disk/by-path/pci-0000:03:00.0-scsi-0:0:4:0-part1`.

```
smw# fdisk -l \
/dev/disk/by-path/pci-0000:05:00.0-sas-phy3-0x4433221103000000-lun-0-part1

Disk /dev/disk/by-path/pci-0000:05:00.0-sas-phy3-0x4433221103000000-lun-0-part1: 1000.2
GB, 1000204886016 bytes
81 heads, 63 sectors/track, 382818 cylinders, total 1953525168 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0xffdfd1e1

 Device                          Boot      Start         End      Blocks   Id  System
/dev/disk/by-path/. . .-lun-0-part1       2048  1953525167   976761560   83  Linux
```

5. Create an ext4 file system on the PMDISK partition.

```
smw# mkfs.ext4 \
/dev/disk/by-path/pci-0000:05:00.0-sas-0x4433221103000000-lun-0-part1

Creating filesystem with 244190390 4k blocks and 61054976 inodes
Filesystem UUID: 6d791409-e327-4620-a80c-2933271b3eec
Superblock backups stored on blocks:
       32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
       4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,
       102400000, 214990848

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

6. Stop the RSMS services.

```
smw# systemctl stop rsms
smw# systemctl status rsms
rsms.service - hss daemon control
  Loaded: loaded (/usr/lib/systemd/system/rsms.service; enabled)
  Active: inactive (dead) since Wed 2015-11-04 15:42:04 CST; 19s ago
 Process: 5471 ExecStop=/opt/cray/hss/default/bin/hssctl stop (code=exited, status=0/
SUCCESS)
 Process: 30305 ExecStart=/opt/cray/hss/default/bin/hssctl start (code=exited, status=0/
SUCCESS)

Nov 03 16:01:43 smw hssctl[30305]: Starting daemons: erd erdh state_mana...md
Nov 04 15:42:04 smw hssctl[5471]: Stopping daemons: sec_cmd boot_cmds ca...rd
Hint: Some lines were ellipsized, use -l to show in full.
```

7. Verify that the RSMS services are stopped. While the RSMS services are stopped, the system may continue to run applications, however the high-speed network will be throttled until RSMS is restarted.

```
smw# rsms status
PID              DAEMON               STATE                UPTIME
                erd                  stopped
                erdh                 stopped
                state_manager        stopped
                nid_mgr              stopped
                bootmanager          stopped
                sedc_manager         stopped
                xtpmd                stopped
                erfsd                stopped
                xtremoted            stopped
                xtpowerd             stopped
                nimsd                stopped
                xtsnmpd              stopped
                xtdiagd              stopped
```

8. Run the `xtmvpmdb` script.

```
smw# xtmvpmdb /dev/disk/by-path/pci-0000:05:00.\
0-sas-0x4433221103000000-lun-0-part1 ext4
 - Checking userid
 - Checking destination directory name
 - Checking destination directory existence

Move database to: /dev/disk/by-path/pci-0000:05:00.0-sas-phy3-0x4433221103000000-lun-0-
part1  [y/n] [y]: y
- Checking userid
- Checking destination directory name
- Checking destination directory existence

Move database to: /dev/disk/by-path/pci-0000:05:00.0-sas-0x4433221103000000-lun-0-part1
[y/n] [y]: y
- Checking current PM database directory existence
- Checking for booted system
- Checking for rsms daemons
- Creating directory /media/temp_pgsql_data
Dir: /media/temp_pgsql_data created
- Checking status of PM database process
Checking for PostgreSQL 9.3.8: ..running
postgresql.service - LSB: Start the PostgreSQL master daemon
  Loaded: loaded (/etc/init.d/postgresql)
  Active: active (exited) since Tue 2015-11-03 15:38:45 CST; 24h ago
 Process: 16633 ExecReload=/etc/init.d/postgresql reload (code=exited, status=0/SUCCESS)
 Process: 16255 ExecStart=/etc/init.d/postgresql start (code=exited, status=0/SUCCESS)


- Stopping PM database
- Copy contents of /var/lib/pgsql to /media/temp_pgsql_data
- This may take a few minutes to complete.
- Rename previous DB directory from: /var/lib/pgsql to: /var/lib/pgsql.
11-04-2015t15:43:04
- Unmount device from temporary mount point: /media/temp_pgsql_data
- Unmount btrfs subvolume: /var/lib/pgsql
- Mount device at permanent mount point: /var/lib/pgsql
- Add mount point to /etc/fstab
- Start PM database

- Transfer of PM database complete.
```

9. Restart the RSMS services and verify that the daemons are starting.

```
smw# systemctl start rsms
smw# systemctl status rsms
rsms.service - hss daemon control
  Loaded: loaded (/usr/lib/systemd/system/rsms.service; enabled)
  Active: active (exited) since Wed 2015-11-04 15:44:24 CST; 9s ago
 Process: 5471 ExecStop=/opt/cray/hss/default/bin/hssctl stop (code=exited, status=0/
SUCCESS)
 Process: 9227 ExecStart=/opt/cray/hss/default/bin/hssctl start (code=exited, status=0/
SUCCESS)

Nov 04 15:44:24 smw hssctl[9227]: Starting daemons: erd erdh state_manag...md
Hint: Some lines were ellipsized, use -l to show in full.
```

10. Verify that the RSMS services are running.

```
smw# rsms status
PID               DAEMON              STATE            UPTIME
9306              erd                 running          Wed 2015-11-04 15:43:27 CST
9435              erdh                running          Wed 2015-11-04 15:43:30 CST
9560              state_manager       running          Wed 2015-11-04 15:43:31 CST
9691              nid_mgr             running          Wed 2015-11-04 15:43:32 CST
9827              bootmanager         running          Wed 2015-11-04 15:43:32 CST
9953              sedc_manager        running          Wed 2015-11-04 15:43:33 CST
10703             xtpmd               running          Wed 2015-11-04 15:43:44 CST
11487             erfsd               running          Wed 2015-11-04 15:43:50 CST
12247             xtremoted           running          Wed 2015-11-04 15:43:56 CST
12521             xtpowerd            running          Wed 2015-11-04 15:44:00 CST
12688             nimsd               running          Wed 2015-11-04 15:44:04 CST
12855             xtsnmpd             running          Wed 2015-11-04 15:44:08 CST
13019             xtdiagd             running          Wed 2015-11-04 15:44:12 CST
```

9.2 Push Diags Image and Update the Diags Bind Mount Profile

Prerequisites

This procedure assumes that the system has been booted after a fresh install.

About this task

The online diagnostics image provides some useful tools that are made available on CLE nodes through the Cray Image Binding service using the profile for the diags image root. This procedure describes how to enable that service and configure it to reference the correct diags image and enable the diags profile.

Procedure

1. Get the name of the diags image root.

```
smw# image list | grep diag
```

2. Push the diags image root to the boot node.

```
smw# image push -d boot <image_root_name>
```

3. Update cray_image_binding, which is in the CLE config set (p0 in this example).

```
smw# cfgset update -s cray_image_binding -m interactive p0
```

The configurator displays the **Service Configuration Menu**. The service name and status appear at the top of the menu. That menu also includes a list of settings. The Cray Image Binding service has a single setting: `profiles`. Under it is a list of bind mount profile entries.

4. If this service is not yet enabled, enable it now.

```
Service Configuration Menu (Config Set: p0, type: cle)
  cray_image_binding       [status: disabled] [validation: skipped]
...

IMPS Image Binding Service Menu [default: save & exit - Q] $ E
```

5. Select the `profiles` setting to configure it.

   Enter **1** and press **Enter** to select the `profiles` setting, and then enter **C** and press **Enter** to configure that setting.

   The configurator displays guidance about the `profiles` setting and a numbered list of profile entries that have already been added. A 'PE' profile and a 'diags' profile should be in that list.

6. Change the value of the 'diags' profile image field.

   a. Enter the number for the 'diags' profile followed by 'a' and '*' to change the field for the diags profile image name, then press **Enter**.

      In this example, the number of the 'diags' profile is 2.

      ```
      cray_image_binding.settings.profiles
      [<cr>=set 2 entries, +=add an entry, ?=help, @=less] $ 2a*
      ```

   b. Enter the name of the diags image, then press **Enter**.

      ```
      cray_image_binding.settings.profiles.data.diags.image
      [<cr>=keep 'diags_cle_rhine_sles_12_x86-64', <new value>, ?=help, @=less] $
      diag-all_cle_60up01_sles_12_x86-64
      ```

7. Set bind_directories and callbacks for the diags profile, as needed.

   As with changing the value of the image field, select each of these fields and enter new values for them, as needed. Note that file path for a callback must be a relative path and should not start with a forward slash (/) character.

8. Enable the diags profile.

   Has the diags image root been pushed to the boot node? If not, do steps 1 and 2 first, and then return to this step.

   a. Enter the number for the 'diags' profile followed by 'd' and '*' to select the field for enabling the diags profile, then press **Enter**.

      In this example, the number of the 'diags' profile is 2.

      ```
      cray_image_binding.settings.profiles
      [<cr>=set 2 entries, +=add an entry, ?=help, @=less] $ 2d*
      ```

   b. Enter `true`, then press **Enter**.

```
cray_image_binding.settings.profiles.data.diags.enabled
[<cr>=keep 'false', <new value>, ?=help, @=less] $ true
```

9. Set the profile entries, and then save changes and exit the configurator.

```
cray_image_binding.settings.profiles
[<cr>=set 2 entries, +=add an entry, ?=help, @=less] $ <cr>
...

IMPS Image Binding Service Menu [default: save & exit - Q] $ Q
```

10. Validate the config set.

```
smw# cfgset validate p0
```

11. To use diags on the system, reboot the system.

9.3 Reduce Impact to SMW Performance of Btrfs Periodic Maintenance

About this task

Btrfs (B-tree file system) runs periodic maintenance. The weekly and monthly maintenance scripts, which include balance, trim, and scrub actions, can consume large amounts of compute resource. This can impact a site's ability to use the SMW for normal operations, even using SSH to log into nodes. This procedure explains how to reduce impact to the SMW by controlling when these scripts are run.

Procedure

1. Create a file `/etc/cron.d/cray_btrfs.cron`. Set ownership to root,root with permissions 644.

   The new cron file needs to be in `/etc/cron.d` because the Btrfs RPM installs links to maintenance scripts into the `/etc/cron.{weekly,monthly}` directories.

2. Add these lines to the new file. Adjust as needed for this site.

```
# Control when btrfs maintenance scripts run by deleting the corresponding
# 'lastrun' files at a predetermined time.  Caveat, this affects all of the
# scripts in the corresponding cron directories (/etc/cron.{weekly,monthy})

# Run weekly on Saturday at 2 AM as root
0 2 * * 6 root rm -f /var/spool/cron/lastrun/cron.weekly
# Run monthly on the first Sunday of the month at 2 AM as root
0 2 * * 0 root [ $(date +%d) -le 07 ] && rm -f /var/spool/cron/lastrun/
cron.monthly
```

9.4 Configure Direct-attached Lustre (DAL)

Prerequisites

This procedure assumes the following:

- Service nodes to support direct-attached Lustre® (DAL) have been identified with `xtdiscover` as management server (MGS), metadata server (MDS), or object storage server (OSS) nodes.
- Configuration worksheets for Cray Linux environment (CLE) have been created and updated for DAL:
  - The `cray_lnet` worksheet is updated and the `cray_lnet.enabled` setting is uncommented and set to `true`. See *Update cray_lnet Worksheet* on page 117.
  - The `cray_lustre_client` worksheet is updated and the `cray_lustre_client.enabled` setting is uncommented and set to `true`. See *Update cray_lustre_client Worksheet* on page 121.
  - `cray_lustre_server` worksheet is updated and the `cray_lustre_server.enabled` setting is uncommented and set to `true`. See *Update cray_lustre_server Worksheet* on page 122

    NOTE: There are additional settings which tune the Lustre kernel modules.
  - If using the Lustre Monitoring Tool (LMT), a MySQL database, storage space for that database, Cerebro, and the LMT GUI must be configured on the MGS node. The cray_lmt worksheet inlcudes settings for configuring LMT.
- All DAL service nodes are assigned to the DAL group so that they are assigned the DAL boot image for booting.
- The `imgbuilder` configuration for DAL has the DAL stanza added to the to the default image group.

About this task

This procedure configures direct-attached Lustre (DAL) nodes that provide a Lustre file system.

Procedure

Identify Logical Unit Numbers (LUNs) for DAL

1. Identify the LUNs used for DAL.

   Log in to the DAL service nodes to identify the persistent storage device names to be used for the Lustre file system. Identify all disk device names that will be used for the metadata target (MDT) / management target (MGT) and object storage target (OST) devices.

   ```
   smw# ssh boot
   boot# ssh dal-mds
   ```

2. If the LUN number is known, then use the `lsscsi` command to map the LUN to the short disk name.

   This example shows that LUN 17 is `/dev/sdr`.

   ```
   dal-mds# lsscsi | grep 17
   [0:0:0:17]   disk    LSI      INF-01-00        0786  /dev/sdr
   ```

3. Use the short disk name from the previous step to determine the long persistent disk device name.

This example shows that `sdr` has two different persistent device names that could be used.

CAUTION: Use persistent device names in the Lustre file system definition. Non-persistent device names (for example, `/dev/sdc`) can change when the system reboots. If non-persistent names are specified in the *`fs_name`*`.fs_defs` file, then Lustre may try to mount the wrong devices and fail to start when the system reboots.

For more information about Lustre control utilities, see the `lustre_control(8)` and `lustre.fs_defs(5)` `man` pages.

```
dal-mds# ls -l /dev/disk/by-id | grep sdr
lrwxrwxrwx 1 root root  9 Aug  4 13:23 scsi-360080e500036ae3e000002e6524a8369 -
> ../../sdr
lrwxrwxrwx 1 root root  9 Aug  4 13:23 wwn-0x60080e500036ae3e000002e6524a8369 -
> ../../sdr
```

Create and Install the Lustre fs_defs File

4. Prepare the Lustre `fs_defs` file on the system management workstation (SMW).

   This file is used by `lustre_control` to format, reformat, start, and stop the file system. When creating the Lustre `fs_defs` file in this example, use `/dev/disk/by-id/scsi-0x60080e500036ae3e000002e6524a8369` for LUN 17. Refer to the *XC Series Lustre Administration Guide* for detailed information about how to create an `fs_defs` file for a Lustre file system.

5. Create a variable called `FS_NAME` to be the name of the file system using 8 characters or less ("dal" in this example). The file name of the `fs_defs` file should be similar to the file system it defines.

   ```
   smw# export FS_NAME=dal
   smw# echo $FS_NAME
   dal
   ```

6. Copy the `example.fs_defs` file to the one named after the DAL file system.

   ```
   smw# cp -p /opt/cray-xt-lustre-utils/default/etc/example.fs_defs \
   /home/crayadm/$FS_NAME.fs_defs
   ```

7. Edit the `$FS_NAME.fs_defs` file. This is a simple example for the p0 partition, which calls the file system "dal" and has the MGT on nid00027, MDT on nid00027 and nid00029, first OST on nid00028, and second OST on nid00031. Substitute site-specific values in this site's `fs_defs` file.

   ```
   smw# vi /home/crayadm/$FS_NAME.fs_defs
   ```

8. Locate `fs_name: example` and change `example` to the name defined by `$FS_NAME` ("dal" in this example).

   ```
   fs_name: dal
   ```

9. Set the Lustre server hosts to LNet NIDs mapping.

   ```
   # Lustre server hosts to LNET NIDs mapping.
   # Multiple lines are additive.
   # Use multiple lines with the same nodes if you have more than one nid for each
   # node.
   ```

```
# Nodes and nids can be specified using range expressions. See the
# lustre.fs_defs man page for more information on range expressions.
# Each line should have a one-to-one mapping between the nodes and nids.
nid_map: nodes=nid000[27-29,31] nids=[27-29,31]@gni
```

10. Update the `fs_defs` file with these settings (substituting appropriate site-specific values). Identify which nodes and devices are being used for MGT, MDT, and OSTs. There are other settings in the `fs_defs` file that can be changed, but are probably acceptable for most sites.

```
## MGT
## Management Target
mgt: node=nid00027
     dev=/dev/disk/by-id/scsi-360001ff020021101061ad79111170000

## MDT
## MetaData Target(s)
mdt: node=nid00027
     dev=/dev/disk/by-id/scsi-360001ff020021101061ad79111170100
     index=0
mdt: node=nid00029
     dev=/dev/disk/by-id/scsi-360001ff020021101061ad79111170200
     index=1

## OST
## Object Storage Target(s)
ost: node=nid00028
     dev=/dev/disk/by-id/scsi-360001ff020021101061ad79111170300
     index=0
ost: node=nid00031
     dev=/dev/disk/by-id/scsi-360001ff020021101061ad7a811170400
     index=1
```

11. Install the `fs_defs` file into the appropriate CLE config_set (p0).

```
smw# lustre_control install -c p0 /home/crayadm/$FS_NAME.fs_defs
```

The `lustre_control install` command copies the `fs_defs` file into a directory in the config set, makes a `lustre_control` readable version of it with the suffix `.config.data`, and updates the list of installed file systems.

12. Verify that the `fs_defs` file is installed in the config set by listing the files in the `lustre/.lctrl/` directory of the config set.

```
smw# ls /var/opt/cray/imps/config/sets/p0/lustre/.lctrl/
dal.config.data    dal.filesys.data                 dal.service.data
dal.failover.data  dal.fs_defs.20160421.1461256838  installed_filesystems
```

Modify the Config Set to Load the `lustre-utils` Module

13. Modify `cray_user_settings.settings.default_modules.data.service` to add `lustre-utils`.

a. Update the cray_user_settings service in config set p0.

```
smw# cfgset update -s cray_user_settings -m interactive -l advanced p0
```

b. Select the `service` setting to configure it.

Enter **2** and press **Enter** to select `service`, then enter **C** and press **Enter** to configure it.

```
Cray User Settings Menu [default: save & ext - Q] $ 2
...
Cray User Settings Menu [default: configure - C] $ C
```

c. Enter **+** to add an entry, then enter "lustre-utils" and press **Enter**. Press **Ctrl-d** to finish adding entries, then press **Enter** to set the entries for this setting.

```
cray_user_settings.settings.default_modules.data.service
[<cr>=set 7 entries, +=add an entry, ?=help, @=less] $ +
Add service (Ctrl-d to exit) $ lustre-utils
Add service (Ctrl-d to exit) $ <Ctrl-d>
...
cray_user_settings.settings.default_modules.data.service
[<cr>=set 8 entries, +=add an entry, ?=help, @=less] $ <cr>
```

14. Validate the config set.

- Entire system:

```
smw# cfgset validate p0
```

- Partitioned system:

```
smw# cfgset validate p1
smw# cfgset validate p2
```

Boot the System and Reformat the DAL File System

The DAL file system must be formatted using `lustre_control` from the boot node after initial set up, and before automating the start up and mounting of the DAL file system.

15. Boot the system.

- If CLE is not booted, proceed to step *16* on page 183.
- If CLE is booted, proceed to step *19* on page 183.

16. If CLE is not booted:

```
crayadm@smw> xtbootsys -a auto.<site_system>
```

17. Reformat the DAL file system after a full system boot.

```
smw# ssh boot
boot# export FS_NAME=dal
boot# lustre_control reformat -f $FS_NAME
```

18. Proceed to step *21* on page 184

19. If CLE is booted, run `cray-ansible`, then reboot only the DAL nodes.

Restarting `/etc/init.d/cray-ansible` refreshes the config set cache on the boot node. This example specifies a comma-separated list of `cnames` (for example `c0-0c0s0n0`) for all DAL nodes (MGS, MDS, and OSS) to create a $DALNODES variable.

```
boot# /etc/init.d/cray-ansible start
```

```
crayadm@smw> export DALNODES=mgsnode,mdsnode,ossnode1,ossnode2
crayadm@smw> xtbounce -s $DALNODES
crayadm@smw> xtcli boot DEFAULT $DALNODES
```

20. Reformat the DAL file system after a reboot of only the DAL nodes.

```
smw# ssh boot
boot# module load lustre-utils
boot# export FS_NAME=dal
boot# lustre_control reformat -f $FS_NAME
Continue? (y|n|q) y
```

Start and Mount the DAL File System

21. Start the DAL file system using `lustre_control` on the boot node.

```
boot# lustre_control start -p -f $FS_NAME
```

22. Verify that the Lustre targets are mounted on each DAL node.

```
boot# lustre_control status -f $FS_NAME
```

23. Test mount the DAL file system on a login node.

```
boot# ssh login
login# export FS_NAME=dal
login# mkdir -p /lus/$FS_NAME
```

```
login# mount -t lustre 27@gni:/$FS_NAME /lus/$FS_NAME
```

In the above mount command, substitute the site-specific value for *27@gni*, which is a combination of the nid of the MGT node and the LNet name by which the external Lustre server is accessed (will be something like gni or gni1). The MGT node nid was defined in the `fs_defs` file in step 10, and the LNet name can be found by searching for "gni" the CLE config set (p0 in this example) on the SMW.

```
smw# cfgset search -t gni -l advanced -s cray_lnet p0
# 2 matches for 'gni' from cray_lnet_config.yaml
#-------------------------------------------------------------------------------
-
cray_lnet.settings.local_lnet.data.lnet_name: gni4
cray_lnet.settings.flat_routes.data.o2ib.src_lnet: gni4
```

Add DAL file system to cray_lustre_client Configuration

24. Add the DAL file system to `cray_lustre_client` configuration so that Lustre clients can mount the file system from the Lustre server.

Note that the `cray_lustre_client` service must be enabled in addition to setting information like the settings below (substitute appropriate site-specific values).

```
smw# cfgset update -s cray_lustre_client -l advanced -m interactive p0
```

```
cray_lustre_client.settings.client_mounts.data.fs_name.dal_login: null
cray_lustre_client.settings.client_mounts.data.dal_login.lustre_fs_name: dal
cray_lustre_client.settings.client_mounts.data.dal_login.mount_point: /lus/dal
```

```
cray_lustre_client.settings.client_mounts.data.dal_login.mgs_lnet_nids:
- 27@gni
- 29@gni
cray_lustre_client.settings.client_mounts.data.dal_login.mount_options:
rw,flock,lazystatfs
cray_lustre_client.settings.client_mounts.data.dal_login.mount_at_boot: false
cray_lustre_client.settings.client_mounts.data.dal_login.mount_locations:
- login

cray_lustre_client.settings.client_mounts.data.fs_name.dal_compute: null
cray_lustre_client.settings.client_mounts.data.dal_compute.lustre_fs_name: dal
cray_lustre_client.settings.client_mounts.data.dal_compute.mount_point: /lus/dal
cray_lustre_client.settings.client_mounts.data.dal_compute.mgs_lnet_nids:
- 27@gni
- 29@gni
cray_lustre_client.settings.client_mounts.data.dal_compute.mount_options:
rw,flock,lazystatfs
cray_lustre_client.settings.client_mounts.data.dal_compute.mount_at_boot: true
cray_lustre_client.settings.client_mounts.data.dal_compute.mount_locations:
- compute
```

Add DAL File System to cray_lustre_server Configuration

25. Add the DAL files system nodes to the cray_lustre_server service.

```
smw# cfgset update -s cray_lustre_server -l advanced -m interactive p0
```

```
cray_lustre_server.settings.lustre_servers.data.mgs: c0-0c0s6n1
cray_lustre_server.settings.lustre_servers.data.mds:
- c0-0c0s6n1
- c0-0c0s7n1
cray_lustre_server.settings.lustre_servers.data.oss:
- c0-0c0s7n2
- c0-0c0s6n2
```

Configure LMT to Monitor DAL

26. (Optional) If using LMT to enable monitoring of DAL, see *LMT_Configuration_for_DAL.ditamap#C320691*.

Enable Realm-Specific Internet Protocol (RSIP) on DAL Nodes

27. Enable RSIP on DAL nodes so they can communicate with an external LDAP or NIS server.

DAL nodes do not have external network connections, but require access to LDAP or NIS servers external to the system for uid/gid information associated with the Lustre file system.

a. Add the DAL nodes to the list of `cray_rsip.settings.service.data.nodes_as_client`.

```
smw# cfgset update -s cray_rsip -l advanced -m interactive p0
```

Add the DAL MDS nodes.

```
cray_rsip.settings.service.data.nodes_as_client:
- c0-0c0s6n1
- c0-0c0s7n1
```

28. Validate the config set.

- Entire system:

```
smw# cfgset validate p0
```

- Partitioned system:

```
smw# cfgset validate p1
smw# cfgset validate p2
```

Update the Boot Automation File for DAL

29. Edit the site boot automation file (in `/opt/cray/hss/default/etc/`) so that the DAL file system is started during the CLE boot.

Because the config set modifications made in an earlier step set it up so that login and elogin nodes do not attempt to mount DAL at boot time, but the compute nodes do, add these DAL lines to the site boot automation **file** after the boot of the service nodes but **before** the boot of the compute nodes.

```
#Boot all the service nodes
lappend actions {crms_boot_all_serv}

# start Lustre server on DAL nodes & mount Lustre filesystem on login nodes
 lappend actions { crms_exec_on_bootnode "root" "lustre_control start -f dal" }
 lappend actions { crms_exec_on_bootnode "root" "lustre_control mount_clients -
f dal -w login[1-2]" }

#Boot specific compute nodes
#lappend actions [list crms_boot_loadfile DEFAULT compute "c0-0c0s7n0
c0-0c0s7n1" linux]

#Boot compute nodes
lappend actions {crms_boot_all_comp}
```

This uses a `pdsh` style list of nodes as an argument for the `mount_clients` command. For example, `lustre_control` will interpret `login[1-8]` as nodes `login1` through `login8`. Replace `dal` in the command with the name of the DAL file system for this site.

With `client_mounts.data.dal_compute.mount_at_boot` set to `true` in the cray_lustre_clients service, the compute nodes automatically mount the DAL file system when they boot. This also ensures that they mount the DAL file system even when rebooted individually, outside the control of the auto boot file.

9.5 LMT Configuration for DAL

The Lustre® monitoring tool (LMT) for direct-attached Lustre (DAL) on Cray Linux environment (CLE 6.0) requires some manual configuration during the software installation process.

Configure Storage for the LMT Database	At least 40GB of storage space must be made available to the MGS node. See *LMT Disk Usage* on page 190.
Configure the LMT MySQL Database	The IMPS configuration does not set up this database, so this must be configured manually for CLE 6.0 UP01. See *Configure LMT MySQL Database for DAL* on page 187.
Configure the LMT GUI (Optional)	See *Configure the LMT GUI* on page 189.

Use the configurator to configure the LMT for DAL on CLE 6.0. Guidance is provided for each LMT configuration setting in the `cfgset` utility.

The `cray_lmt` configurator template configures LMT settings for specific nodes when they are booted. The default system configuration value for the LMT service is disabled (`false`). Log in to the SMW as `root` and use the `cfgset` command to modify the `cray_lmt` configuration settings to configure LMT.

```
smw# cfgset update -s cray_lmt -m interactive CONFIG_SET
```

9.5.1 Configure LMT MySQL Database for DAL

Prerequisites

A MySQL server instance must be configured on the management server (MGS) node. All commands described below should be executed on the MGS for the direct-attached Lustre (DAL) file system.

About this task

A MySQL server instance on the management server (MGS) node stores real-time and historical Lustre monitoring tool (LMT) data. The configurator does not handle the initial setup of the LMT MySQL users and database. It must, therefore, be done manually. All commands described below should be executed on the MGS for the DAL file system.

Procedure

1. Log on to the MGS as `root`.

   Where *nidMGS* is the node ID (NID) of the MGS node.

   ```
   boot# ssh nidMGS
   ```

2. Start the MySQL server daemon (if not already running).

   ```
   mgs# /sbin/service mysqld start
   ```

3. Run the `mysql_secure_installation` script to improve MySQL server instance security.

   This sets the password for the `root` MySQL user, disallows remote `root` access to the database, removes anonymous users, removes the test database, and reloads privileges. If this is the first time configuring LMT, create a symlink before running `mysql_secure_installation` to ensure that MySQL uses the correct socket.

   ```
   mgs# ln -s /var/run/mysql/mysql.sock /var/lib/mysql/mysql.sock
   mgs# mysql_secure_installation
   ```

   Prompts and recommended responses generated by the script.

   ```
   Enter current password for root (enter for none): <Enter>

   Set root password? [Y/n] Y
   New password: Enter a secure password
   Re-enter new password: Enter the secure password again

   Remove anonymous users? [Y/n] Y
   ```

```
Disallow root login remotely? [Y/n] Y

Remove test database and access to it? [Y/n] Y

Reload privilege tables now? [Y/n] Y
```

4. Ensure `root` only access to the LMT user configuration file, `/usr/share/lmt/mkusers.sql`.

```
mgs# chmod 600 /usr/share/lmt/mkusers.sql
```

5. Edit the LMT user configuration file `/usr/share/lmt/mkusers.sql`.

   This file will not be used at run time by any LMT or MySQL processes. It is simply a script that will be run to create the MySQL users on the persistent storage set up for use by the MySQL databases. Once it is run through MySQL, it is no longer needed.

```
mgs# vi /usr/share/lmt/mkusers.sql
```

   This file contains MySQL statements that create users named `lwatchclient` and `lwatchadmin`. It gives them privileges only on databases that start with `filesystem_`. Cray recommends making the following changes to `mkusers.sql`.

Edit the GRANT Statement	Edit the GRANT statements to grant privileges on only `filesystem_`*`fsname`*`.*` where *`fsname`* is the name of the file system. This will only grant permissions on the database for the file system being monitored.
Edit the Password	Edit the password for `lwatchadmin` by changing `mypass` to the desired password. Also add a password for the `lwatchclient` user.

```
CREATE USER 'lwatchclient'@'localhost' IDENTIFIED BY 'foo';
GRANT SELECT ON filesystem_scratch.* TO 'lwatchclient'@'localhost';

CREATE USER 'lwatchadmin'@'localhost' IDENTIFIED BY 'bar';
GRANT SELECT,INSERT,DELETE  ON filesystem_scratch.* TO
'lwatchadmin'@'localhost';
GRANT CREATE,DROP           ON filesystem_scratch.* TO
'lwatchadmin'@'localhost';

FLUSH PRIVILEGES;
```

6. Save the changes and execute the following command. This prompts for the MySQL `root` user password, which was set when `mysql_secure_installation` was executed.

```
mgs# mysql -u root -p < /usr/share/lmt/mkusers.sql
```

7. Create the database for the file system to be monitored.

```
mgs# lmtinit -a fsname
```

   Where *`fsname`* is the name of the DAL file system (LMT data will be inserted into the LMT MySQL database the next time the Cerebro service is restarted on the MGS).

8. Restart Cerebro.

```
mgs# service cerebrod restart
```

9. Verify that LMT is adding data to the MySQL database.

   a. Initiate the LMT shell.

   ```
   mgs# lmtsh -f fsname
   ```

   b. List tables.

   ```
   fsname> t
   ```

   c. List tables again after several seconds to verify that `Row Count` is increasing.

9.5.2 Configure the LMT GUI

About this task

The Lustre monitoring tool (LMT) graphical user interface (GUI) package is installed on login nodes. It contains a GUI called `lwatch` and a command-line tool for viewing live data called `lstat`. The configuration file `~/.lmtrc` must be set up prior to using either tool.

Procedure

1. Login to the MGS node as `root`.

2. Edit the sample configuration file `/usr/share/doc/packages/lmt-gui/sample.lmtrc` to reflect the site specific LMT configuration—where *`db_name`* is set to the name of the MySQL database used by LMT, that is, filesystem_*`fsname`*.

   ```
   # LMT Configuration File - place in $HOME/.lmtrc

   filesys.1.name=<insert_fsname_here>
   filesys.1.mountname=<insert_/path/to/mountpoint_here>
   filesys.1.dbhost=<insert_db_host_ip_here>
   filesys.1.dbport=<insert_db_port_here>
   filesys.1.dbuser=<insert_db_client_username_here>
   # Leave dbauth blank if the given client has no password
   filesys.1.dbauth=<insert_db_client_password_here>
   filesys.1.dbname=<insert_db_name_here>
   ```

3. Save the updated `.lmtrc` as `~/.lmtrc`.

   Both `lwatch` and `lstat` are now usable.

   Here is an example for configuring access to the LMT database for the file system named *`scratch_1`*, which was set up so that the user *`lwatchclient`* has no password. In this example, access is being configured on the LMT server node, so the database is local. Thus, the `db_host` is *`localhost`*.

   ```
   filesys.1.name=scratch_1
   filesys.1.mountname=/lus/scratch_1
   filesys.1.dbhost=localhost
   filesys.1.dbport=3306
   filesys.1.dbuser=lwatchclient
   filesys.1.dbauth=
   filesys.1.dbname=filesystem_scratch_1
   ```

After setting up `~/.lmtrc`, `lwatch` and `lstat` can be run on this node.

To run the GUI from a remote node, the MySQL database must be configured to allow remote access for the read-only user, `lwatchclient`. See *Configure LMT MySQL for Remote Access* on page 190.

9.5.3 Configure LMT MySQL for Remote Access

In order to run the Lustre monitoring tool (LMT) graphical user interface (GUI) on a separate node from the LMT server, the MySQL server instance (running on the LMT server) must be configured to enable remote access for the LMT read-only user, `lwatchclient`. These MySQL statements can be added to `/usr/share/lmt/mkusers.sql` prior to executing the statements in that file. They can also be executed directly. In these examples, *`FSNAME`* is the name of the file system being monitored.

```
CREATE USER 'lwatchclient'@'%' IDENTIFIED BY 'foo';
GRANT SELECT ON filesystem_FSNAME.* TO 'lwatchclient'@'%';
```

To execute these statements directly, log on to the DAL MGS node, open a mysql shell as the root MySQL user, and run the statements as follows.

```
mgs# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
...
mysql> CREATE USER 'lwatchclient'@'%';
Query OK, 0 rows affected (0.00 sec)
...
mysql> GRANT SELECT ON filesystem_FSNAME.* TO 'lwatchclient'@'%';
Query OK, 0 rows affected (0.00 sec)
```

This enables the user named `lwatchclient` to connect from any hostname.

To allow connections from a certain IP address, replace the '%' with an IP address in single quotes.

```
CREATE USER 'lwatchclient'@'10.11.255.252' IDENTIFIED BY 'foo';
GRANT SELECT ON filesystem_FSNAME.* TO 'lwatchclient'@'10.11.255.252';
```

9.5.4 LMT Disk Usage

LMT requires at least 40GB persistent storage attached to the LMT server (i.e., the management server (MGS)) to store historical data. If the storage becomes full, data can be deleted from the database using MySQL delete statements.

MySQL Tables

Five tables store general file system statistics. These tables are populated by `lmt_agg.cron` script.

Table 8. General File System Tables

Table Name	On-Disk Growth Rate
FILESYSTEM_AGGREGATE_HOUR	0.8 KB/hour
FILESYSTEM_AGGREGATE_DAY	0.8 KB/day

Table Name	On-Disk Growth Rate
FILESYSTEM_AGGREGATE_WEEK	0.8 KB/week
FILESYSTEM_AGGREGATE_MONTH	0.8 KB/month
FILESYSTEM_AGGREGATE_YEAR	0.8 KB/year

Table 9. MDS Aggregate Tables and Growth Rates

Table Name	Approximage On-Disk Growth Rate
MDS_AGGREGATE_HOUR	0.5 KB/hour/MDS
MDS_AGGREGATE_DAY	0.5 KB/day/MDS
MDS_AGGREGATE_WEEK	0.5 KB/week/MDS
MDS_AGGREGATE_MONTH	0.5 KB/month/MDS
MDS_AGGREGATE_YEAR	0.5 KB/year/MDS

Table 10. OST Aggregate Tables and Growth Rates

Table Name	On-Disk Growth Rate
OST_AGGREGATE_HOUR	0.7 KB/hour/OST
OST_AGGREGATE_DAY	0.7 KB/day/OST
OST_AGGREGATE_WEEK	0.7 KB/week/OST
OST_AGGREGATE_MONTH	0.7 KB/month/OST
OST_AGGREGATE_YEAR	0.7 KB/year/OST

Calculate Expected Disk Usage for a File System

Use this formula to calculate the approximate rate of disk space usage for a file system. Disregard the AGGREGATE tables as they grow so much more slowly than the raw data tables.

```
(56 KB/hour/filesystem) * (# of filesystems) + (1000 KB/hour/MDS) * (# of MDSs)
    + (44 KB/hour/OSS) * (# of OSSs) + (70 KB/hour/OST) * (# of OSTs) = Total KB/
hour
```

Calculate the Disk Usage for a File System for 1 Year

In this example, LMT is monitoring one file system with one MDS, four object storage servers (OSS), and eight object storage targets (OST). The amount of disk space used by the LMT database to is expected to grow at this hourly rate.

```
56 KB/hour/filesystem * 1 filesystem + 1000 KB/hour/MDS * 1 MDS
    + 44 KB/hour/OSS * 4 OSSs + 70 KB/hour/OST * 8 OSTs = 1792 KB/hour
```

Which translates to this yearly rate.

```
1792 KB/hour * 24 hours/day * 365 days/year * 1 MB/1024KB
       * 1 GB/1024MB = 15 GB / year
```

9.6 Configure the Simple Event Correlator (SEC)

The Simple Event Correlator (SEC) is an SMW utility that parses every line being appended to system log files, watching for specific strings that represent the occurrence of significant system events. When a specified string is detected, SEC sends notification that this has happened, either by email, IRC, writing to a file, or some user-configurable combination of all three.

SEC is enabled by default, and by default is configured to generate email notifications to `crayadm`. The types of notifications generated and the recipients to whom notifications are sent are defined in the SEC configuration file, `/etc/opt/cray/cray_sec_actions_config`.

This release includes the Open Source simple event correlator (SEC) package, `sec-2.7.6`, and an SEC support package, `cray-sec-8.0`. The SEC support package contains control scripts to manage the starting and stopping of SEC around a Cray mainframe boot session, in addition to other utilities.

For configuration procedures, see *Configure Cray SEC Software* (S-2542) for release CLE 6.0.UP01 / SMW 8.0.UP01.

9.7 Configure System Environmental Data Collections (SEDC)

SEDC is a tool that collects and reports in real time the environmental data on all Cray systems. Data includes information from sensors located on significant hardware components at the cabinet and blade level, such as power supplies, processors, memory and fans. SEDC refers to these sensors as *scan IDs*. Examples of collected data include cabinet and blade/node temperatures, voltage, current, power, cooling system air pressure, humidity, and statuses. At the node level, data is collected only from the nodes that are powered on.

For configuration procedures, see *System Environmental Data Collections Guide* (S-2491) for release CLE 6.0.UP01 / SMW 8.0.UP01.

9.8 Configure Netroot

This part of the installation and configuration process is optional unless this site has decided to use Netroot. See *Where to Place the Root File System* on page 158 for more information.

Netroot needs a matched pair of images for compute nodes and login nodes.

compute nodes	• initrd-compute-large: the NIMS boot image is set to this image • compute-large: the NIMS kernel parameter "netroot" is set to this image, and this image is pushed to the boot node
login nodes	• initrd-login-large: the NIMS boot image is set to this

- login-large: the NIMS kernel parameter "netroot" is set to this image, and this image is pushed to the boot node

The following procedures describe how to configure the Netroot images, push Netroot images to the boot node, and reboot nodes with Netroot.

9.8.1 Configure Netroot Images

Prerequisites

This procedure assumes the following:

- Basic configuration is complete and the system has been booted.
- No Netroot images have been built yet. If that is not the case, and the Netroot specifications shown below are already in the "default" image group of `cray_image_groups.yaml`, then skip this procedure.

About this task

This procedure adds entries to the "default" image group (if they are not already there) and then builds the Netroot images. Going forward, with the Netroot image specifications added to the "default" image group, both compute and login Netroot images will be built every time `imgbulder` is run.

Procedure

1. Add Netroot image specifications to `cray_image_groups.yaml` to the "default" image group.

   ```
   smw# vi /var/opt/cray/imps/config/sets/global/config/cray_image_groups.yaml
   ```

   Add these two image specifications to the "default" image group, if they are not already there. (Safest to find these two image specifications elsewhere in the file, then copy and paste them from there to the default image group.)

   ```
   - recipe: "initrd-compute-large_cle_rhine_sles_12_x86-64_ari"
     dest: "initrd-compute-large{note}_cle_{cle_release}-build{cle_build}
   {patch}_sles_12-created{date}.cpio"
     nims_group: "compute"
   - recipe: "initrd-login-large_cle_rhine_sles_12_x86-64_ari"
     dest: "initrd-login-large{note}_cle_{cle_release}-build{cle_build}
   {patch}_sles_12-created{date}.cpio"
     nims_group: "login"
   ```

2. Create new images.

   full system For a full, unpartitioned system:

   ```
   smw# imgbuilder --map
   ```

   partitioned For a partitioned system, with partitions p1 and p2:

   ```
   smw# imgbuilder --map -p p1
   ```

   ```
   smw# imgbuilder --map -p p2
   ```

At the end of the output from `imgbuilder`, there will be a command hint for how to push the resulting Netroot images to the boot node. Note the `image push` command with the specific image name that needs to be pushed to the boot node.

```
IMPORTANT: The netroot image for initrd-compute-large_cle_6.0.DV00-
build201508120201_sles_12-created20150813.cpio must be pushed to the boot node:
smw:# image push -d boot compute-large_cle_6.0.DV00-build201508120201_sles_12-
created20150813

IMPORTANT: The netroot image for initrd-login-large_cle_6.0.DV00-
build201508120201_sles_12-created20150813.cpio must be pushed to the boot node:
smw:# image push -d boot login-large_cle_6.0.DV00-build201508120201_sles_12-
created20150813
```

9.8.2 Push Netroot Images to Boot Node

Prerequisites

This procedure assumes the following:

- The boot node is booted.
- Netroot images have been built using `imgbuilder`, and the output of that command provided the specific image name that needs to be pushed to the boot node.

Procedure

Push the Netroot images to the boot node.

full system For a full, unpartitioned system:

```
smw# image push -d boot compute-large_cle_6.0.DV00-build\
${SOFTWARE}_sles_12-createdYYYYMMDD

smw# image push -d boot login-large_cle_6.0.DV00-build\
${SOFTWARE}_sles_12-createdYYYYMMDD
```

partitioned For a partitioned system, with partitions p1 and p2:

```
smw# image push -d boot-p1 compute-large_cle_6.0.DV00-build\
${SOFTWARE}_sles_12-createdYYYYMMDD

smw# image push -d boot-p1 login-large_cle_6.0.DV00-build\
${SOFTWARE}_sles_12-createdYYYYMMDD
```

```
smw# image push -d boot-p2 compute-large_cle_6.0.DV00-build\
${SOFTWARE}_sles_12-createdYYYYMMDD

smw# image push -d boot-p2 login-large_cle_6.0.DV00-build\
${SOFTWARE}_sles_12-createdYYYYMMDD
```

The nodes that will use the Netroot images can be warmbooted, or the entire system can be rebooted.

Trouble? If passwordless `ssh` has not been prepared between root@smw and root@boot, then the system will prompt for the password for root@boot twice.

9.8.3 Reboot Nodes with Netroot

Prerequisites

This procedure assumes that Netroot images have been pushed out to the boot node.

About this task

This procedure reboots nodes with the new Netroot images, either by shutting down the entire system and rebooting it or by warm booting only the nodes that need the new Netroot images.

Procedure

1. Reboot entire system with new Netroot images.

   In these examples, replace `auto.<site_system>` with the site boot automation file.

   full system For a full, unpartitioned system:

   ```
   smw# su - crayadm
   crayadm@smw> xtbootsys -s last -a auto.xtshutdown
   crayadm@smw> xtbootsys -a auto.<site_system>
   ```

   partitioned For a partitioned system, with partitions p1 and p2:

   ```
   smw# su - crayadm

   crayadm@smw> xtbootsys -s last -p p1 -a auto.xtshutdown
   crayadm@smw> xtbootsys -p p1 -a auto.<site_system>

   crayadm@smw> xtbootsys -s last -p p2 -a auto.xtshutdown
   crayadm@smw> xtbootsys -p p2 -a auto.<site_system>
   ```

2. (Alternative) Warm boot only nodes needing new Netroot images.

   a. Warm boot all login nodes.

      Provide the same list of cnames for the login nodes to both `xtnmi` and `xtbootsys --reboot`.

      full system For a full, unpartitioned system:

      ```
      smw# su - crayadm

      crayadm@smw> export LOGINNODES=c0-0c0s7n3,c0-0c0s8n0
      crayadm@smw> xtnmi $LOGINNODES
      crayadm@smw> sleep 60

      crayadm@smw> xtbootsys --reboot -r "warmboot for login netroot"
      $LOGINNODES
      ```

      partitioned For a partitioned system, with partitions p1 and p2 (example for p1):

      ```
      smw# su - crayadm

      crayadm@smw> export
      ```

```
LOGINNODES=c0-0c0s9n0,c0-0c0s10n0,c0-0c0s11n0,c0-0c0s13n0
crayadm@smw> xtnmi -p p1 $LOGINNODES
crayadm@smw> sleep 60

crayadm@smw> xtbootsys --reboot -r "warmboot for login netroot" -p
p1 $LOGINNODES
```

b. Warm boot all compute nodes.

full system For a full, unpartitioned system (example for p0 with cabinet c0-0):

```
smw# su - crayadm

crayadm@smw> export COMPUTENODES=$(xtcli status p0 | \
egrep -v "empty|service|disabled" | grep c0-0 | \
awk '{ FS=":"; print $1 }' | tr ':' ' ' | \
awk '{ printf "%s,", $1 }' | sed s'/.$//') \
echo -e "Compute nodes are:\n$COMPUTENODES"

crayadm@smw> xtcli status  $COMPUTENODES
crayadm@smw> xtcli shutdown $COMPUTENODES
crayadm@smw> xtcli status $COMPUTENODES
crayadm@smw> xtnmi $COMPUTENODES
crayadm@smw> sleep 60
crayadm@smw> xtcli status $COMPUTENODES

crayadm@smw> xtbootsys --reboot -r \
"warmboot for compute netroot" $COMPUTENODES

crayadm@smw> xtcli status $COMPUTENODES
```

partitioned For a partitioned system, with partitions p1 and p2 (example for p1 with cabinet c0-0):

```
smw# su - crayadm

crayadm@smw> export COMPUTENODES=$(xtcli status p1 | \
egrep -v "empty|service|disabled" | grep c0-0 | \
awk '{ FS=":"; print $1 }' | tr ':' ' ' | \
awk '{ printf "%s,", $1 }' | sed s'/.$//') \
echo -e "Compute nodes are:\n$COMPUTENODES"

crayadm@smw> xtcli status  $COMPUTENODES
crayadm@smw> xtcli shutdown $COMPUTENODES
crayadm@smw> xtcli status $COMPUTENODES
crayadm@smw> xtnmi -p p1 $COMPUTENODES
crayadm@smw> sleep 60
crayadm@smw> xtcli status $COMPUTENODES

crayadm@smw> xtbootsys --reboot -p p1 -r \
"warmboot for compute netroot" $COMPUTENODES

crayadm@smw> xtcli status $COMPUTENODES
```

9.8.4 Disable Netroot Images

Prerequisites

This procedure assumes Netroot images are in use.

About this task

This procedure disables Netroot images and switches back to the non-Netroot images, which includes changing the NIMS group from "netroot_compute" back to "compute" and from "netroot_login" back to "login."

Procedure

1. Disable Netroot images.

   full system For a full, unpartitioned system:

   ```
   smw# cnode update -g compute --filter group=netroot_compute
   smw# cnode update -g login --filter group=netroot_login
   ```

   partitioned For a partitioned system, with partitions p1 and p2:

   ```
   smw# cnode update -p p1 -g compute --filter group=netroot_compute

   smw# cnode update -p p1 -g login --filter group=netroot_login

   smw# cnode update -p p2 -g compute --filter group=netroot_compute

   smw# cnode update -p p2 -g login --filter group=netroot_login
   ```

2. Switch back to non-Netroot images.

   Run `imgbuilder` to set the images back to the normal image name and clear the `netroot=` kernel parameter.

   full system For a full, unpartitioned system:

   ```
   smw# imgbuilder --map -p p0
   ```

   partitioned For a partitioned system, with partitions p1 and p2:

   ```
   smw# imgbuilder --map -p p1

   smw# imgbuilder --map -p p2
   ```

9.9 Prevent Unintentional Re-creation of Mail Configuration Files

This procedure is optional. It applies to systems where postfix or sendmail are configured on the SMW.

To prevent the `master.cf` and `main.cf` postfix configuration files from being re-created during software updates or fixes, edit the `/etc/sysconfig/mail` file on the SMW and ensure that the `MAIL_CREATE_CONFIG` setting is set to "no."

```
smw# vi /etc/sysconfig/mail

MAIL_CREATE_CONFIG="no"
```

10 Install Additional Software

This is the final stage in the installation/configuration process. These procedures provide additional functionality for a Cray XC Series system.

optional	*Install the Dell Systems Management Tools and Documentation DVD* on page 199
optional	*Install and Configure DataWarp* on page 200
optional	*Install Cray Programming Environment (PE) Software* on page 200 (PE)
optional	*Install and Configure a Workload Manager (WLM)* on page 205
optional	*Install and Configure CMC/eLogin* on page 206

10.1 Install the Dell Systems Management Tools and Documentation DVD

About this task

This procedure installs the OpenManage Server Administrator (OMSA) software from the Dell Systems Management Tools and Documentation DVD, which is shipped with the SMW. This software enables advanced control over the Integrated Dell Remote Access Controller (iDRAC) and provides features such as Automatic Recovery (automatic system boot after a power event).

Visit the Dell OpenManage Linux Repository to view the Dell OpenManage Server Administrator documentation: *http://linux.dell.com/wiki/index.php/Repository/OMSA*

Procedure

1. Obtain the Dell System Management Tools and Documentation DVD.
2. Log on to the SMW as `root`.
3. Mount the DVD.

   ```
   smw# mount /dev/cdrom /media/cdrom
   ```

4. Go to the location of the installation scripts.

   ```
   smw# cd /media/cdrom/SYSMGMT/srvadmin/linux/supportscripts
   ```

5. Execute the script to install the software.

```
smw# sh srvadmin-install.sh --express
```

6. Start the Server Administrator services.

```
smw# sh srvadmin-services.sh start
```

7. Double-click the icon named **Launch Server Administrator** on the SMW screen.
8. Enter the SMW user name `root`.
9. Enter the SMW `root` account password.

The system can now be managed for Properties, Shutdown, Logs, Alert Management, and Session Management.

10.2 Install and Configure DataWarp

Cray DataWarp provides an intermediate layer of high bandwidth, file-based storage to applications running on compute nodes. It is comprised of commercial SSD hardware and software, Linux community software, and Cray system hardware and software. DataWarp storage is located on server nodes connected to the Cray system's high speed network (HSN). I/O operations to this storage completes faster than I/O to the attached parallel file system (PFS), allowing the application to resume computation more quickly and resulting in improved application performance. DataWarp storage is transparently available to applications via standard POSIX I/O operations and can be configured in multiple ways for different purposes. DataWarp capacity and bandwidth are dynamically allocated to jobs on request and can be scaled up by adding DataWarp server nodes to the system.

For installation and configuration procedures, see *DataWarp Installation, Configuration, and Administration Guide* for this release .

10.3 Install Cray Programming Environment (PE) Software

About this task

The Cray Developers Toolkit (CDT) for Cray XC Series systems and the Cray Application Developer's Environment (CADE) for Cray XE and Cray XK systems are available from Cray Inc. as separate software packages. The two packages are comparable but not interchangeable.

The CDT package consists of the basic libraries and components needed to develop and compile code on Cray systems, including the GNU Fortran, C, and C++ compilers. The CDT also includes the Cray Compiling Environment (CCE), but a valid license key is required before CCE can be installed. All other compilers are sold, installed, and licensed separately.

This procedure installs and configures the Cray Programming Environment (PE) software to make its content available on Cray XC Series compute nodes. A typical PE installation takes about 10 minutes.

Procedure

1. Create the PE image root.

a. Use a PE image for several of the monthly releases of PE software and use a fresh image with each new CLE release.

The `pe_compute_cle_6.0up01_sles_12` name is preconfigured in the Image Binding Service. If using a different name for the PE image (`$PEIMAGE`), also update the name in the PE profile of the `cray_image_binding` service for the CLE configuration set (a later step in this procedure).

```
smw# export PEIMAGE=pe_compute_cle_6.0up01_sles_12
```

Check for an existing `pe_compute` image.

```
smw# image list | egrep "^[ ]*$PEIMAGE"
```

b. Create PE image on the SMW.

1. Get the name of the PE image recipe on the system.

```
smw# recipe list | grep ^pe
pe_image_cle_6.0up01_sles_12
```

2. Create `$PEIMAGE` image using the recipe name discovered by the command in the previous step.

```
smw# image create -r pe_image_cle_6.0up01_sles_12 $PEIMAGE
```

2. Install the compiler license RPMs.

The Cray Compiling Environment (CCE), Intel, and PGI compilers all require licenses. These licenses must be installed at this point before installing any of the PE software. For instructions, see

CCE	*Cray Compiling Environment Release Overview and Installation Guide*, available at *http://pubs.cray.com*
Intel compilers	*http://software.intel.com/en-us/articles/intel-software-technical-documentation*
PGI compilers	*http://www.pgroup.com*

3. Copy the most recent PE ISO to the SMW and mount the ISO.

```
smw# mkdir -p /var/adm/cray/release/pe
smw# cd /var/adm/cray/release/pe
smw# mkdir -p /var/adm/cray/release/pe/mount_iso
smw# mount -o loop,ro <downloaded PE ISO> /var/adm/cray/release/pe/mount_iso
```

4. Install the `craype-installer` rpm from the PE ISO on the SMW.

```
smw# rpm -ivh /var/adm/cray/release/pe/mount_iso/installer/\
craype-installer-*.x86_64.rpm
```

5. Configure the installer configuration file.

a. Copy the install configuration file from the `craype-installer` installation directory.

```
smw# cp -p /opt/cray/craype-installer/default/conf/install-cdt.yaml .
```

b. Create logs directory that will be used by the installer.

```
smw# mkdir ./logs
```

c. Update the configuration file. When `install-cdt.yaml` is opened, there are comment blocks before every keyword listed below describing the valid values for each.

1. For `IMAGE_DIRECTORIES` specify the directory (or directories) for the installer to install into. This example uses an `image_root` of `pe_compute_cle_rhine_sles_12`. This parameter must have data on the next line. The data must have four space characters and then a dash character and then a space character and the path to the directory.
2. Specify **`YES`** in each of the `INSTALL_*_LIBRARIES` for the compiler specific PE libraries to be installed. The Pathscale compiler is no longer supported by PE.
3. If the system includes an `ACCELERATOR`, change **`NONE`** to a comma separated list of one or more of the supported accelerators - **`FERMI or KEPLER`**. See the comments in `install-cdt.yaml` for examples and more information.
4. If the system has more than one type of processor installed, then specify the lowest common denominator for the processor for `CRAY_CPU_TARGET`.

```
smw# vi install-cdt.yaml
```

```
---
HAS_MAMU_NODES : NO
ACCELERATORS                          : NONE
NETWORK_TYPE                          : NONE
CRAY_CPU_TARGET                       : sandybridge
BOOTNODE_HOSTNAME                     : NONE
BOOTNODE_ROOT_DIRS                    :
    - /rr/current
ESMS_HOSTNAME                         : NONE
ESMS_IMAGE_DIRS                       :
    - /cm/images/<your image name>
UNMANAGED_ESLOGINS  : NONE
IMAGE_DIRECTORIES :
    - /var/opt/cray/imps/image_roots/pe_compute_cle_6.0up01_sles_12
LOGS_DIR                              : ./logs
ISO_MOUNT_DIR                         : ./mount_iso
INSTALL_CCE_LIBRARIES                 : YES
INSTALL_GNU_LIBRARIES                 : YES
INSTALL_INTEL_LIBRARIES               : YES
INSTALL_PATHSCALE_LIBRARIES           : NO
INSTALL_PGI_LIBRARIES                 : YES
```

6. Install PE software from the most recent PE installation media and installer.

a. Link `/opt/cray/pe/bin` to `/opt/cray`.

```
smw# chroot /var/opt/cray/imps/image_roots/$PEIMAGE ln \
-s /opt/cray/pe/bin /opt/cray/bin
```

b. Run the PE installer.

```
smw# module load craype-installer
smw# craype-installer.pl --install --install-yaml-path ./install-cdt.yaml
```

When the installation completes, the following will be displayed, summarizing the installed packages.

```
1) atp-1.7.5-0_3605.x86_64              (/var/opt/cray/imps/image_roots/
pe_compute_cle_rhine_sles_12_x86-64_ari)
 2) cray-ccdb-1.0.3-0_3575.x86_64        (/var/opt/cray/imps/image_roots/
pe_compute_cle_rhine_sles_12_x86-64_ari)
```

```
 3) cray-dwarf-14.2.0-0.x86_64          (/var/opt/cray/imps/image_roots/
pe_compute_cle_rhine_sles_12_x86-64_ari)
 <snip>
71) perftools-clients-6.2.2-1.x86_64    (/var/opt/cray/imps/image_roots/
pe_compute_cle_rhine_sles_12_x86-64_ari)
```

c. Set the default versions for PE (if the install succeeds) by running `set_default` scripts.

```
smw# craype-installer.pl --set-default --install-yaml-path ./install-cdt.yaml
```

d. Unmount the ISO.

```
smw# umount ./mount_iso
```

e. Clean up the PE ISO and PE rpms.

These can be removed since they are large and use up disk space.

```
smw# rm *.iso *.rpm *.tar.gz
```

7. Push the PE image root to the boot node.

For p0:

```
smw# image push -d boot $PEIMAGE
INFO - Remotely cloning Image '<name of image>' to 'boot'...
INFO - Checking remote destination...
INFO - Transferring Image '<name of image>' to 'root@boot:/var/opt/cray/imps/
image_roots/<name of image>'...
INFO - Cloned Image '<name of image>' to remote host 'root@boot:/var/opt/cray/
imps/image_roots/<name of image>'.
```

For partitioned systems, push to the boot node for that partition, `boot-p1`:

```
smw# image push -d boot-p1 $PEIMAGE
```

8. Enable PE.

For a fresh install, configure and enable the PE bind mount profile in the Cray Image Binding service, and then validate the config set.

a. Update cray_image_binding, which is in the CLE config set.

Full system:

```
smw# cfgset update -s cray_image_binding -m interactive p0
```

Partitioned system:

```
smw# cfgset update -s cray_image_binding -m interactive pN
```

b. Select the `profiles` setting to configure it.

Enter **1** and press **Enter** to select the `profiles` setting, then enter **c** and press **Enter** to configure it.

c. Change the value of the 'PE' profile image field to match the name of the image used in earlier steps (`$PEIMAGE`).

In this example, the number of the 'PE' profile is 1, so enter **1a*** to edit the PE `image` setting.

```
cray_image_binding.settings.profiles
[<cr>=set 2 entries, +=add an entry, ?=help, @=less] $ 1a*
```

Enter the name of the PE image.

```
cray_image_binding.settings.profiles.data.PE.image
[<cr>=keep 'pe_compute_cle_rhine_sles_12', <new value>, ?=help, @=less] $
PE_image_name
```

d. Ensure that the 'PE' profile callbacks field is set.

In this example, the number of the 'PE' profile is 1, so enter **1c*** to edit the PE `callbacks` setting.

```
cray_image_binding.settings.profiles
[<cr>=set 2 entries, +=add an entry, ?=help, @=less] $ 1c*
```

Press **Enter** (`<cr>`) to set the existing entry, enter **1*** to edit that entry, or **+** to add a new entry. Note that the default callbacks entry (`opt/cray/pe/bin/pe_postmount_callback.sh`) is a relative path. If a new path is entered, it also must be a relative path and should not start with a forward slash (`/`) character.

```
cray_image_binding.settings.profiles.data.PE.callbacks
[<cr>=set 1 entries, +=add an entry, ?=help, @=less] $ <cr>
```

e. Enable the PE profile.

Has the PE image root been pushed to the boot node? If not, do step 7 first, and then return to this step.

In this example, the number of the 'PE' profile is 1, so enter **1d*** to edit the PE profile `enabled` setting.

```
cray_image_binding.settings.profiles
[<cr>=set 2 entries, +=add an entry, ?=help, @=less] $ 1d*
```

```
cray_image_binding.settings.profiles.data.PE.enabled
[<cr>=keep 'false', <new value>, ?=help, @=less] $ true
```

f. Set the profile entries, and then save changes and exit the configurator.

```
cray_image_binding.settings.profiles
[<cr>=set 2 entries, +=add an entry, ?=help, @=less] $ <cr>
...

IMPS Image Binding Service Menu [default: save & exit - Q] $ Q
```

g. Validate the config set.

Full system:

```
smw# cfgset validate p0
```

Partitioned system:

```
smw# cfgset validate pN
```

9. Reboot the system with PE.

```
smw# su - crayadm
crayadm@smw> xtbootsys -s last -a auto.xtshutdown
crayadm@smw> xtbootsys -a auto.<site_system>
```

10. Build a sample MPI job that leverages the PE product by compiling and executing the application.

a. Test basic `aprun` functionality.

1. Log in to the login node.

```
crayadm@smw> ssh boot
crayadm@boot> ssh login
```

2. Run `apstat` to get the number of nodes to use for the following commands:

```
crayadm@login> NUMNODES=$(($(apstat -v | grep XT | awk "{print \$3}"))); \
echo NUMNODES is $NUMNODES
```

```
crayadm@login> aprun -n $NUMNODES -N2 python -c "print 'hello world.'"
```

b. Compile a sample MPI program.

1. Unload the default `PrgEnv` module.

```
crayadm@login> module unload PrgEnv-cray
```

2. Load modules.

```
crayadm@login> module load PrgEnv-gnu cray-mpich
crayadm@login> cd /tmp
crayadm@login> export CRAY_CPU_TARGET=x86-64
```

3. Obtain sample MPI code for compile.
4. Compile sample MPI code.
5. Execute sample MPI code.

c. Log out of the login node and boot node and su session to return to being root on the SMW.

```
crayadm@login> exit
crayadm@boot> exit
crayadm@smw> exit
smw#
```

11. Make a snapshot post PE installation.

Cray recommends saving a snapshot of the system immediately after the PE software installation is complete. If any root users make bad changes after the software install is complete, revert to this snapshot to avoid a redo of the entire software install.

```
smw# snaputil list
smw# export SNAPSHOT=$(snaputil list |grep ^cur| awk '{print $2}')
smw# echo $SNAPSHOT
smw# snaputil create ${SNAPSHOT}.save4.postpe --from ${SNAPSHOT}
```

12. Back up the config set post PE installation.

```
smw# cfgset create --clone p0 p0.postpe
```

10.4 Install and Configure a Workload Manager (WLM)

Cray XC Series systems support the use of workload manager (WLM) software products. The CLE 6.0.UP01 / SMW 8.0.UP01 release supports these three WLM products: PBS, Moab/TORQUE, and SLURM. Each product requires installation and configuration prior to use.

PBS Professional™	PBS Professional is a commercial product licensed by Altair Engineering, Inc. ● For more product information: *http://www.altair.com* ● Other notes: PBS Professional uses a license manager, which requires a network connection between the license server and the SDB node on a Cray system.
Moab™ and TORQUE	Moab and TORQUE are commercial products licensed by Adaptive Computing. ● For more product information: *http://www.adaptivecomputing.com*
Slurm	Slurm (Simple Linux Utility for Resource Management) is an open source application that is commercially supported by SchedMD, among others. ● For more product information: *http://www.schedmd.com/* ● For Cray-specific installation/configuration instructions: *Slurm Software Installation Guide for Cray CX Series Systems*

For the most up-to-date information regarding workload manager software compatibility with CLE releases, look on the CrayPort website at *http://crayport.cray.com*.

10.5 Install and Configure CMC/eLogin

External services formerly provided by ESlogin are now supported by eLogin. eLogin uses Cray System Management Software (CSMS) installed on a separate management server to deploy eLogin images to external Cray Development and Login (CDL) nodes. The management server and the CDL nodes are each deployed to Dell 720s.

The management server connects to the SMW, which provides shared image and configuration services. A recipe for eLogin nodes exists on the SMW so that an image root for eLogin can be created and then packaged into the proper boot image format, enabling the CSMS node to deliver it to the eLogin node.

> **NOTE:** In previous software versions, the CMC was called the CIMS or esMS node, and the eLogin nodes were called the CDL or esLogin nodes.

For installation and upgrade instructions, see *eLogin Installation Guide* for release CLE 6.0.UP01 / SMW 8.0.UP01.

11 Back Up the Newly Installed and Configured SMW/CLE Software

After installing and configuring the new SMW/CLE software, create a backup of it, if needed.

Dell R815 SMW with software RAID	If an R815 SMW was configured using the recommended software RAID1 configuration for the boot disk, there is no need to make a backup of the SMW boot disk because the boot disks are mirrored automatically by the RAID software.
Dell R630 SMW with software RAID	If an R630 SMW was configured using the recommended hardware RAID5 virtual disk, there is no need to make a backup of the SMW boot disk. Any one of the four disks in the RAID5 configuration can fail without losing any data.

12 Customize a Preinstalled SMW/CLE System

Cray ships System Management Workstation (SMW) systems that are completely installed and configured with Cray-specific hostnames and IP addresses, among other things. To complete the configuration on site, reconfigure the system using these procedures.

Note that many commands require root privilege.

1. *Update Site Information and Install Needed Patches* on page 208.
2. *Change the Default SMW Passwords* (includes instructions for logging in as root).
3. *Change the Time Zone* on page 211.
4. (Optional) *Configure the SMW Firewall* on page 214.
5. *Configure LAN on the SMW* on page 215.
6. *Change Networks and IP Addresses in Global Config Set* on page 215.
7. *Change Networks and IP Addresses in CLE Config Set* on page 217.
8. Configure iDRAC network information.
   - For a Dell R630 SMW: *Set Up iDRAC for a Dell R630 SMW* on page 219.
   - For a Dell R815 SMW: *Set Up iDRAC for a Dell R815 SMW* on page 222.
9. *Change the Default iDRAC Password* on page 225.
10. (Optional) *Configure the Simple Event Correlator (SEC)* on page 192.
11. (Optional) *Configure Site Lightweight Log Manager (LLM)* on page 226.
12. (Optional) *Prevent Unintentional Re-creation of Mail Configuration Files* on page 197.
13. *Set Variable for Snapshot Name and Make a Snapshot Using snaputil* on page 88 (post-site-customization).
14. *Back Up the Newly Installed and Configured SMW/CLE Software* on page 207.

12.1 Update Site Information and Install Needed Patches

Prerequisites

This procedure uses the `xtshowrev` tool. If that module not yet loaded, see *Run the xtgetrev and xtshowrev Tools* on page 167.

About this task

The first task in customizing a preinstalled system is to ensure that the site name and serial number are set correctly, and all needed patches are installed.

Procedure

1. Determine which patches were installed in the factory.

   Use the `xtshowrev` command. The example output shows where to look for CLE, SLE, and SMW patch information.

```
smw# xtshowrev
Site:                CRAY/INTERNAL
S/N:                 9999
System Type:         XC40
Install Date:        2016-06-01
System Name:         panda1
CNL/CLE Release:     6.0.UP01
XT Release:          6.0.96
CLE Kernel:          3.12.51-52.31.1_1.0600.9146
CLE OS:              SLES12
CLE Patch Sets:      01 02 03      <------------------ CLE patches applied
CLE FNs:
Lustre Version:      2.5.4
OS Type:             CLE
SMW Release:         8.0.UP01
SMW Build:           8.0.96
HSS Release:         8.0__446__ge75851a-49.1
SMW Kernel:          3.12.51-52.39
SMW OS:              SLES12
SLE Patch Sets:                    <------------------ SLE patches applied
SMW Patch Sets:                    <------------------ SMW patches applied
SMW FNs:             5844c
SEC Release:         Cray_SEC 8.0__6__g689802a (sec 2.7.6)
Current Date:        2016-06-01 12:59:21
crayadm@smw>
```

2. Update site information in the `site_config` file.

   For an initial install, the `xtgetrev` command is used to enter site information. For a preinstalled system, enter this information by manually editing the `site_config` file instead.

```
smw# vi /etc/opt/cray/release/pkginfo/site_config
---
site name: CRAY/INTERNAL      <------------------ change this
serial number: 9999           <------------------ change this
system name: panda1           <------------------ change this, if needed
system type: XC40             <------------------ change this, if needed
install date: 2016-06-01
os type: CLE
```

   It is especially important to change/enter the serial number because that is the key into the Site Configurations Database, and it is used to determine whether a site has access to future patches.

3. Check for patches released by Cray.

Day-1 patches are noted in the Errata docs that are included with the release. For other patches, check CrayPort, which is updated with available patches for the entitled site serial numbers when a patch is released. If patches need to be applied, continue with the remaining steps.

CONTINUE ONLY IF PATCHES NEED TO BE APPLIED

4. Download the patches.

5. Install the patches.

   - To install a single patch, simply follow the instructions provided in the patch README file.
   - To load and install multiple patches, complete the following substeps. When installing more than one patch, Cray recommends postponing the building and mapping of images until the last patch is installed.

   a. Temporarily suppress building and mapping images.

   ```
   smw# export PATCHSET_BUILD_IMAGES=false
   smw# export PATCHSET_NIMS_TIMING=deferred
   ```

   b. Follow all of the instructions in the patch README files.

   These instructions will include running the LOAD script and the INSTALL script for each patch, and there may be additional steps for some patches, such as running `xtzap` again to update firmware from an SMW patch.

   c. Build and map new images.

   ```
   smw# imgbuilder --map
   ```

12.2 Change the Default System Management Workstation (SMW) Passwords

About this task

The SMW contains its own `/etc/passwd` file that is separate from the password file for the rest of the CLE system. After logging on to the SMW for the first time, Cray recommends changing the default passwords, as described in the following instructions.

Procedure

1. Log in to SMW as root.

   When the login screen is displayed with the crayadm account as the account which will be logged in:

   a. Select **Not listed?**, then enter `root` for the username.

   b. Either press **Enter** or select **Sign In**.

   c. Enter the password for root.

2. Change default passwords on the SMW by executing the following commands.

```
smw# passwd root
```

```
smw# passwd crayadm
```

```
smw# passwd mysql
```

It is also necessary to change the iDRAC password, which uses a different procedure. See *Change the Default iDRAC Password* on page 225.

12.3 Change the Time Zone

Prerequisites

This procedure assumes the XC system is booted.

About this task

This procedure changes the time zone of an XC system by changing some configuration and then rebooting components. Most of these commands must be performed as root.

Procedure

1. Check the current time zone.
   a. Check time zone on SMW.

      ```
      smw# date
      ```

   b. Check time zone on cabinet and blade controllers.

      ```
      smw# xtrsh -l root -s date
      ```

   c. Check time zone on boot node.

      ```
      smw# ssh boot date
      ```

   d. Check time zone on SDB node.

      This command works from the SMW if the SDB node is a tier1 node with an Ethernet connection to the SMW.

      ```
      smw# ssh sdb date
      ```

   e. Check time zone on all service nodes.

      ```
      smw# ssh sdb pcmd -r -n ALL_SERVICE_NOT_ME "date"
      ```

   f. Check time zone on all compute nodes.

      ```
      smw# ssh sdb pcmd -r -n ALL_COMPUTE "date"
      ```

   Continue to the next step only if the time zone needs to be changed.

2. Change the SMW local time zone, if needed.

   The default timezone on the SMW is **America/Chicago**. To change it:

   a. Execute this command:

      ```
      smw# yast2 timezone
      ```

      `yast2` opens a new window for changing the timezone, then a pop-up window appears with this message: "file `/etc/ntp.conf` has been changed manually. YaST might lose some of the changes."

   b. Select the **Do not show this message anymore** checkbox, then select **Continue**.

   c. Choose the timezone either by selecting a region on the map or by using the drop-down menus for **Region** and **Time Zone**.

   d. Select **Other Settings** if the time is incorrect, then select the **Manually** radio button and enter **Current Time** and **Current Date**. Select **Accept** when done.

   e. Select **OK** when done with timezone settings.

   The change on the SMW is immediate, but any users on the system need to log out and then log in again to get the new environment. This does not change the time zone for the CLE nodes or the cabinet and blade controllers. Continue to step 3 to make those changes.

3. Change the time zone in the global config set.

   a. Set `cray_time.settings.service.data.timezone` to the desired time zone.

      A list of possible time zones is available on the SMW in `/usr/share/zoneinfo/zone1970.tab`.

      ```
      smw# cfgset update -s cray_time -m interactive global
      ```

   b. Validate the config set.

      ```
      smw# cfgset validate global
      ```

4. Change the time zone in the CLE config set.

   If the CLE config set has `cray_time.inherit` set to true, then the time zone and other time settings from the global config set will be inherited by the CLE config set. If the CLE config set has `cray_time.inherit` set to false, then use these commands to change the setting and validate the config set.

   a. Set `cray_time.settings.service.data.timezone` to the desired time zone.

      A list of possible time zones is available on the SMW in `/usr/share/zoneinfo/zone1970.tab`.

      ```
      smw# cfgset update -s cray_time -m interactive p0
      ```

   b. Validate the config set.

      ```
      smw# cfgset validate p0
      ```

5. Put the SMW time zone setting where the cabinet and blade controllers can access it.

   ```
   smw# cp /etc/localtime /opt/tftpboot/localtime
   ```

6. Reboot to set the new time zone for all components.

Follow these steps to set the new time zone for all components in the SMW and CLE system after the global and CLE config sets and SMW `yast2` have been updated with the new setting.

a. Shutdown CLE and reboot the SMW.

```
crayadm@smw> xtbootsys -s last -a auto.xtshutdown
crayadm@adm> su - root
smw# reboot
```

After the SMW reboots, check that the SMW has the desired time zone setting.

```
smw# date
```

b. Power down the system.

```
smw# xtcli power down s0
```

c. Reboot the cabinet controllers, then verify that all cabinet controllers are up.

```
smw# xtccreboot -c all

smw# sleep 120

smw# xtalive -a l1sysd -l l1 s0
```

Repeat the `xtalive` command until all cabinet controllers are alive.

d. Reboot the blade controllers, then verify that all blade controllers are up.

```
smw# xtccreboot -b all

smw# sleep 120

smw# xtalive s0
```

Repeat the `xtalive` command until all blade controllers are alive.

e. Power up the system.

```
smw# xtcli power up s0
```

f. Boot CLE nodes for the new time zone using the site boot automation file.

```
crayadm@smw> xtbootsys -a auto.<site_system>
```

7. Check the current time zone again.

a. Check time zone on SMW.

```
smw# date
```

b. Check time zone on cabinet and blade controllers.

```
smw# xtrsh -l root -s date
```

c. Check time zone on boot node.

```
smw# ssh boot date
```

d. Check time zone on SDB node.

   This command works from the SMW if the SDB node is a tier1 node with an Ethernet connection to the SMW.

```
smw# ssh sdb date
```

e. Check time zone on all service nodes.

```
smw# ssh sdb pcmd -r -n ALL_SERVICE_NOT_ME "date"
```

f. Check time zone on all compute nodes.

```
smw# ssh sdb pcmd -r -n ALL_COMPUTE "date"
```

   If these checks show the correct time zone, then the time zone has been successfully changed.

12.4 Configure the SMW Firewall

Prerequisites

This procedure assumes that SLES 12 has been installed as the base operating system on the SMW.

About this task

The SuSE firewall settings may need to be adjusted to match site firewall policy and to customize for site IP addresses. This procedure enables and configures the firewall.

TIP: It is not necessary to shut down the system before performing this task.

Procedure

1. Save the SuSE firewall configuration.

   Before modifying the SuSE firewall settings, make a copy of the configuration file.

```
smw# cp -p /etc/sysconfig/SuSEfirewall2 /etc/sysconfig/SuSEfirewall2.orig
```

2. Check current firewall settings.

   Check current firewall settings and change to support any site requirements. During the process of configuring Cray SMW and CLE software, some of the firewall settings may be adjusted. SSH access is one of the protocols permitted through the firewall from the external network to the SMW.

```
smw# iptables -L
smw# vi /etc/sysconfig/SuSEfirewall2
```

3. Start the firewall immediately.

   Invoke the modified configuration.

```
smw# systemctl start SuSEfirewall2_init.service
smw# systemctl start SuSEfirewall2.service
```

4. Ensure that the firewall will start at next boot.

Execute the following commands to start the firewall at boot time.

```
smw# systemctl enable SuSEfirewall2_init.service
smw# systemctl enable SuSEfirewall2.service
```

5. Verify firewall changes.

   Verify the changes to the `iptables`.

   ```
   smw# iptables -nvL
   ```

12.5 Configure LAN on the SMW

About this task

This procedure sets the network configuration for eth0 and the hostname for the SMW.

Procedure

1. Execute this command:

   ```
   smw# yast2 lan
   ```

   The **Network Settings** screen appears with the **Overview** tab highlighted.

2. Select the **eth0** line on the **Overview** tab, then select **Edit**.

   The **Network Card Setup** screen appears with the **Address** tab highlighted.

3. Select **Statically Assigned IP address** on the **Address** tab and enter values for IP address, subnet mask, and hostname (including the domain name). Then select **Next**.

4. Select the **Hostname/DNS** tab on the **Network Settings** screen.
   1. For the **Hostname and Domain Name** area, enter Hostname and Domain Name.
   2. For the **Name Servers and Domain Search List**, enter Name Server 1, Name Server 2, Name Server 3, and Domain Search.

5. Select the **Routing** tab on the **Network Settings** screen, then enter the Default IPv4 Gateway (for the network connected to eth0) and set Device to eth0 using the dropdown menu.

6. Click **OK** after all of the **Network Settings** have been prepared.

12.6 Change Networks and IP Addresses in Global Config Set

Prerequisites

This procedure assumes that the SMW and CLE software has been installed so that the global config set is present.

About this task

This procedure suggests some settings to change in the Cray Global Networking service (in the global config set) to add site-specific data. It also includes steps to validate the config set and run Ansible plays on the SMW to effect the changes.

Procedure

1. Check the global config set for information that may need to be changed.

   ```
   smw# cfgset search -t "ipv4_address|management" global
   ```

2. Update the cray_global_net service to change any settings that need site-specific information.

   ```
   smw# cfgset update -s cray_global_net -m interactive -s advanced global
   ```

   At a minimum, these settings will need to be changed:

   - Information for the management network, which is the customer network connected to the SMW.

     ```
     cray_global_net.settings.networks.data.management.ipv4_network
     cray_global_net.settings.networks.data.management.ipv4_netmask
     cray_global_net.settings.networks.data.management.ipv4_gateway
     cray_global_net.settings.networks.data.management.dns_servers
     cray_global_net.settings.networks.data.management.ntp_servers
     ```

   - IP address of the SMW on the management network.

     ```
     cray_global_net.settings.hosts.data.primary_smw.interfaces.customer_ethernet.
     ipv4_address
     ```

3. Validate the global config set.

   ```
   smw# cfgset validate global
   ```

4. Run Ansible plays on the SMW.

   After the global config set has been updated, reapply any Ansible plays that consume global config set data.

   > **NOTE:** (SMW HA only) Both SMWs require this command. The procedure to install and configure the second SMW includes this command.

   ```
   smw# /etc/init.d/cray-ansible start
   ```

   Logs from running Ansible plays, such as cray-ansible, are stored on the SMW in `/var/opt/cray/log/ansible`.

12.7 Change Networks and IP Addresses in CLE Config Set

Prerequisites

This procedure assumes that the SMW and CLE software has been installed so that a CLE config set is present.

About this task

The Cray Networking service defines all network information for CLE nodes. This procedure suggests some settings to change in the Cray Networking service configuration worksheet to add site-specific data. It also includes steps to validate the config set and run Ansible plays on the SMW to effect the changes.

> **REMEMBER:** For partitioned systems, each partition generally has its own config set and associated configuration worksheets. Follow this procedure for each partition.

Procedure

1. Save a copy of original CLE worksheets.

   Copy the original configuration worksheets into a new directory to preserve them in case they are needed later for comparison.

   ```
   smw# ls -l /var/opt/cray/imps/config/sets/p0/worksheets

   smw# cp -a /var/opt/cray/imps/config/sets/p0/worksheets \
   /var/opt/cray/imps/config/sets/p0/worksheets.orig
   ```

2. Make a work area for CLE worksheets.

   Copy the CLE configuration worksheets to a new work area for editing. The worksheets should not be edited in their original location for two reasons: (1) the configurator will not permit updating a config set from worksheets within that config set, and (2) edits would be overwritten when the config set is updated.

   ```
   smw# cp -a /var/opt/cray/imps/config/sets/p0/worksheets \
   /var/adm/cray/release/p0_worksheet_workarea
   ```

   Change to the work area directory to simplify the editing commands in the following steps.

   ```
   smw# cd /var/adm/cray/release/p0_worksheet_workarea
   ```

3. Check the CLE config set for information that may need to be changed.

   ```
   smw# cfgset search -s cray_net -t ipv4 p0
   ```

4. Edit `cray_net_worksheet.yaml` to change any settings that need site-specific information.

   At a minimum, these settings will need to be changed:

   a. Change these values to site-specific values for the "Customer network" to which the login nodes connect.

      ```
      cray_net.settings.networks.data.login.ipv4_network
      cray_net.settings.networks.data.login.ipv4_netmask
      ```

b. (Only for systems with an external Lustre server) Change these values to site-specific values for each external Lustre server.

```
cray_net.settings.networks.data.lnet.ipv4_network
cray_net.settings.networks.data.lnet.ipv4_netmask
```

c. Change this value to the IP address of the login node's eth0 interface on the "login" network.

```
cray_net.settings.hosts.data.login_node.interfaces.login_ethernet.ipv4_addres
s
```

When making changes, keep this mind:

- Add values for the `dns_servers` and `dns_search` settings to the login network only, not to any other network.
- DO NOT add a value for the `ntp_servers` setting for any network used for CLE nodes, because CLE nodes must source their time/NTP settings from the SMW rather than try to contact NTP servers on the login network.

5. Configure additional hosts, as needed.

If this system has additional service nodes that need to have hostname or hostname alias or network interface settings, then add a section like this for each of the hosts. The first example shows the host configuration of a DVS node with the hostname set to "dvs1," a hostname alias of "dvs," and one Ethernet interface connected to the "login" network.

```
cray_net.settings.hosts.data.common_name.dvs_node: null
cray_net.settings.hosts.data.dvs_node.description: DVS node
cray_net.settings.hosts.data.dvs_node.aliases:
- dvs
cray_net.settings.hosts.data.dvs_node.roles: []
cray_net.settings.hosts.data.dvs_node.hostid: 'c0-0c0s0n2'
cray_net.settings.hosts.data.dvs_node.host_type: ''
cray_net.settings.hosts.data.dvs_node.hostname: 'dvs1'
cray_net.settings.hosts.data.dvs_node.standby_node: false

cray_net.settings.hosts.data.dvs_node.interfaces.common_name.eth0: null
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.name: eth0
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.description: Ethernet
    connecting the DVS node to the customer network.
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.aliases: []
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.network: login
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.ipv4_address: '172.30.50.128'
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.mac: ''
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.startmode: auto
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.bootproto: static
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.mtu: ''
cray_net.settings.hosts.data.dvs_node.interfaces.eth0.extra_attributes: []
#cray_net.settings.hosts.data.dvs_node.interfaces.eth0.module: ''
#cray_net.settings.hosts.data.dvs_node.interfaces.eth0.params: ''
#cray_net.settings.hosts.data.dvs_node.interfaces.eth0.unmanaged_interface: false
```

The second example shows the host configuration for an LNet router node that has two different InfiniBand interfaces (ib0 and ib2) to connect to two different networks.

```
cray_net.settings.hosts.data.common_name.clfs_lnet_1: null
cray_net.settings.hosts.data.clfs_lnet_1.description: CLFS router 1 node
cray_net.settings.hosts.data.clfs_lnet_1.aliases: []
cray_net.settings.hosts.data.clfs_lnet_1.roles: []
cray_net.settings.hosts.data.clfs_lnet_1.hostid: 'c0-0c1s0n1'
cray_net.settings.hosts.data.clfs_lnet_1.host_type: ''
cray_net.settings.hosts.data.clfs_lnet_1.hostname: lnet1
cray_net.settings.hosts.data.clfs_lnet_1.standby_node: false
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.common_name.ib0: null
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.name: ib0
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.description: InfiniBand
    ib0 connecting the CLFS router 1 node to the lnet network.
```

```
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.aliases: []
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.network: lnet
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.ipv4_address: '10.150.10.65'
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.mac: ''
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.startmode: auto
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.bootproto: static
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.mtu: '65520'
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.extra_attributes:
- IPOIB_MODE='connected'
#cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.module: ''
#cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.params: ''
#cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib0.unmanaged_interface: false

cray_net.settings.hosts.data.clfs_lnet_1.interfaces.common_name.ib2: null
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.name: ib2
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.description: InfiniBand
    ib2 connecting the CLFS router 1 node to the lnet1 network.
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.aliases: []
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.network: lnet1
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.ipv4_address: '10.151.10.65'
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.mac: ''
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.startmode: auto
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.bootproto: static
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.mtu: '65520'
cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.extra_attributes:
- IPOIB_MODE='connected'
#cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.module: ''
#cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.params: ''
#cray_net.settings.hosts.data.clfs_lnet_1.interfaces.ib2.unmanaged_interface: false
```

6. Upload the modified worksheet into the CLE config set.

   Note that the full filepath must be specified in this `cfgset` command.

   ```
   smw# cfgset update -w \
   /var/adm/cray/release/p0_worksheet_workarea/cray_net_worksheet.yaml p0
   ```

7. Validate the CLE config set.

   ```
   smw# cfgset validate p0
   ```

8. Run Ansible plays on the SMW.

   After the CLE config set has been updated, reapply any Ansible plays that consume CLE config set data.

   > **NOTE:** (SMW HA only) Both SMWs require this command. The procedure to install and configure the second SMW includes this command.

   ```
   smw# /etc/init.d/cray-ansible start
   ```

   Logs from running Ansible plays, such as cray-ansible, are stored on the SMW in `/var/opt/cray/log/ansible`.

12.8 Set Up iDRAC for a Dell R630 SMW

Prerequisites

This procedure requires the following:

- Physical access to the SMW console
- iDRAC6 IP address, subnet mask, and default gateway

- SMW `root` account password

About this task

An integrated Dell Remote Access Controller (iDRAC) enables remote management of the System Management Workstation (SMW). This procedure sets up and enables an iDRAC for an R630 SMW. For an R815 model, see *Set Up iDRAC for a Dell R815 SMW* on page 222.

Procedure

1. If the SMW is up, `su` to `root` and shut it down.

```
crayadm@smw> su - root
smw# shutdown -h now;exit
```

2. Connect an Ethernet cable to the iDRAC port. The cable is located on back of the R815 SMW in the lower left corner.

3. Power up the SMW.

4. Change the iDRAC settings.

   Select **iDRAC Settings** on the **System Setup Main Menu**, then press **Enter**.

   The **iDRAC Settings** screen appears.

   Figure 24. Dell R630 iDRAC Settings Screen

5. Change the iDRAC network.

   a. Select **Network** to display a long list of network settings.

   b. Change the DNS DRAC name.

Use the arrow key to scroll down to **DNS DRAC Name**, then enter an iDRAC hostname that is similar to the SMW node hostname (e.g., cray-drac).

c. Change the static DNS domain name.

Use the arrow key to scroll down to **Static DNS Domain Name**, then enter the DNS domain name and press **Enter**.

d. Change the IPv4 settings.

Use the arrow key to scroll down to the **IPV4 SETTINGS** list.

1. Ensure that IPv4 is enabled.
   a. If necessary, select **Enable IPV4**, then press **Enter**.
   b. Select **<Enabled>** in the pop-up window, then press **Enter** to return to the previous screen.
2. Ensure that DHCP is disabled.
   a. If necessary, select **Enable DHCP**, then press **Enter**.
   b. Select **<Disabled>** in the pop-up window, then press **Enter** to return to the previous screen.
3. Change the IP address.
   a. Select **Static IP Address**.
   b. Enter the IP address of the iDRAC interface (`ipmi0`) for the SMW, then press **Enter**.
4. Change the gateway.
   a. Select **Static Gateway**.
   b. Enter the appropriate value for the gateway of the network to which the iDRAC is connected, then press **Enter**.
5. Change the subnet mask.
   a. Select **Subnet Mask**.
   b. Enter the subnet mask for the network to which the iDRAC is connected (such as `255.255.255.0`), then press **Enter**.
6. Change the DNS server settings.
   a. Select **Static Preferred DNS Server**, enter the IP address of the primary DNS server, then press **Enter**.
   b. Select **Alternate DNS Server**, enter the IP address of the alternate DNS server, then press **Enter**.

e. Change the IPMI settings.

Change the IPMI settings to enable the Serial Over LAN (SOL) console.

1. Use the arrow key to scroll down to the **IPMI SETTINGS** list.
2. Ensure that **Enable IPMI over LAN** is selected.

   TIP: Use the left-arrow or right-arrow to switch between two settings.
3. Ensure that **Channel Privilege Level Limit** is set to **Administrator**.

f. Exit Network screen.

Press the **Esc** key to exit the **Network** screen and return to the **iDRAC Settings** screen.

6. Change hostname in iDRAC LCD display.

Change front panel security to show the hostname in LCD display.

a. Use the arrow key to scroll down and highlight **Front Panel Security** on the **iDRAC Settings** screen, then press **Enter**.

b. Select **Set LCD message**, then press **Enter**.

c. Select **User-Defined String**, then press **Enter**.

d. Select **User-Defined String**, then enter the SMW hostname and press **Enter**.

e. Press the **Esc** key to exit the **Front Panel Security** screen.

7. (Optional) Change the iDRAC **System Location** fields.

Change the **System Location** configuration on the **iDRAC Settings** screen to set any of these fields: **Data Center Name**, **Aisle Name**, **Rack Name**, and **Rack Slot**.

8. Set the password for the iDRAC root account.

a. Use the arrow key to highlight **User Configuration** on the **iDRAC Settings** screen, then press **Enter**.

b. Confirm that User Name is root. Select **User Name**, then enter the "root" user name.

c. Select **Change Password**, then enter a new password.

d. Reenter the new password in the next pop-up window to confirm it (the default password is "calvin").

e. Press the **Esc** key to exit the **User Configuration** screen.

9. Exit iDRAC settings.

a. Press the **Esc** key to exit the **iDRAC Settings** screen.

A "Settings have changed" message appears.

b. Select **Yes**, then press **Enter** to save the changes.

A "Success" message appears.

c. Select **Ok**, then press **Enter**.

The main screen (**System Setup Main Menu**) appears.

12.9 Set Up iDRAC for a Dell R815 SMW

Prerequisites

This procedure requires the following:

- Physical access to the SMW console
- iDRAC6 IP address, subnet mask, and default gateway
- SMW `root` account password

About this task

An integrated Dell Remote Access Controller (iDRAC) enables remote management of the System Management Workstation (SMW). This procedure sets up and enables an iDRAC for an R815 SMW. For an R630 model, see *Set Up iDRAC for a Dell R630 SMW* on page 219.

Procedure

1. If the SMW is up, `su` to `root` and shut it down.

```
crayadm@smw> su - root
smw# shutdown -h now;exit
```

2. Connect an Ethernet cable to the iDRAC port. The cable is located on back of the R815 SMW in the lower left corner.

3. Power up the SMW.

4. Change the iDRAC settings.

   Watch the screen carefully as text scrolls until the **iDRAC6 Configuration Utility 1.57** line is visible. When the line **Press <Ctrl-E> for Remote Access Setup within 5 sec...** displays, press **Ctrl-E** within 5 seconds.

```
0 5 0 ATA WDC WD5000BPVT-0 1A01 465 GB
LSI Corporation MPT2 boot ROM successfully installed!
iDRAC6 Configuration Utility 1.57
Copyright 2010 Dell Inc. All Rights Reserved
iDRAC6 Firmware Revision version: 1.54.15
Primary Backplane Firmware Revision 1.07
-------------------------------
IPv6 Settings
-------------------------------
IPv6 Stack : Disabled
Address 1 : ::
Default Gateway : ::
-------------------------------
IPv4 Settings
-------------------------------
IPv4 Stack : Enabled
IP Address : 172. 31. 73.142
Subnet mask : 255.255.255. 0
Default Gateway : 172. 31. 73. 1
Press <Ctrl-E> for Remote Access Setup within 5 sec...
```

   The **iDRAC6 Configuration Utility** menu appears.

5. Set **iDRAC LAN** to **ON**.

6. Configure the iDRAC LAN.

   Select **LAN Parameters**, then press **Enter**.

   a. Configure iDRAC6 name.

      Use the arrow key to scroll down and select **iDRAC6 Name**, then press **Enter**. Enter a value for **Current DNS iDRAC6 Name** (e.g., smw-drac), then press **Enter**.

   b. Configure domain name.

Use the arrow key to scroll down and select **Domain Name**, then press **Enter**. Enter a value for **Current Domain Name** (e.g., us.cray.com), then press **Enter**.

c. Configure hostname string.

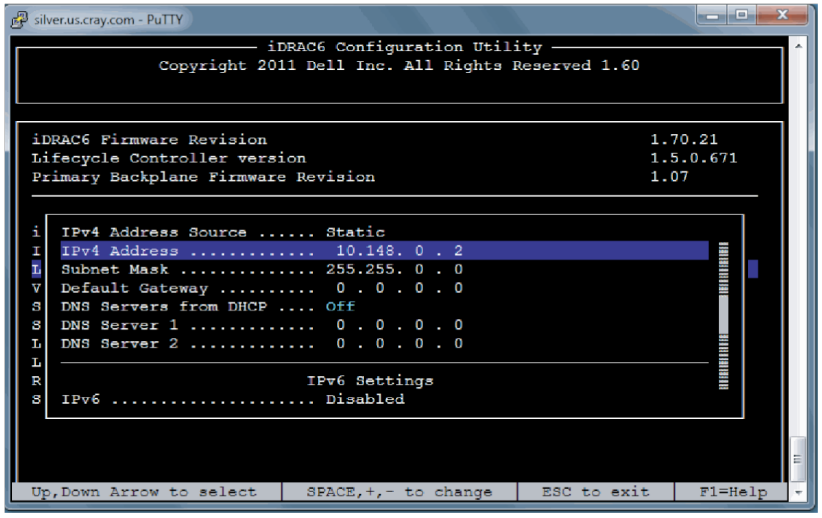
Use the arrow key to scroll down and select **Host Name String**, then press **Enter**. Enter a value for **Current Host Name String** (e.g., smw-drac), then press **Enter**.

d. Configure IPv4 settings.

Use the arrow key to scroll down into the **IPv4 Settings** group and confirm that the **IPv4 Address Source** is set to **static**. Then enter values for the following:

IPv4 Address (the SMW DRAC IP address)
Subnet Mask (the SMW iDRAC subnet mask)
Default Gateway (the SMW iDRAC default gateway)
DNS Server 1 (the first site DNS server)
DNS Server 2 (the second site DNS server)

Figure 25. Dell R815 SMW DRAC IPv4 Parameter Settings



e. Configure IPv6 settings.

Use the arrow key to scroll down into the **IPv6 Settings** group and ensure that **IPv6** is disabled.

f. Press **Esc** to exit **LAN Parameters** and return to the **iDRAC6 Configuration Utility** menu.

7. Configure iDRAC virtual media.

a. Select **Domain Name**, then press **Enter**.

b. Select **Virtual Media Configuration**, then press **Enter**.

c. Select the **Virtual Media** line and press the space key until it indicates **Detached**.

d. Press **Esc** to exit the **Virtual Media Configuration** menu.

8. Set the password for the iDRAC LAN root account.

Using the arrow keys, select **LAN User Configuration**, then press **Enter**. The following configuration is for both SSH and web browser access to the iDRAC.

a. Select **Account User Name** and enter the account name "root."

b. Select **Enter Password** and enter the intended password.

c. Select **Confirm Password** and enter the intended password again.

d. Press **Esc** to return to the **iDRAC6 Configuration Utility** menu.

9. Exit the iDRAC configuration utility.

   a. Press **Esc** to exit the **iDRAC6 Configuration Utility** menu.

   b. Select **Save Changes and Exit**.

   The **BIOS Boot Manager** menu appears.

12.10 Change the Default iDRAC Password

About this task

This procedure describes how to log in to the iDRAC web interface and change a user password.

Procedure

1. Bring up a web browser.
2. Go to: `https://cray-drac`, where *`cray-drac`* is the name of the iDRAC.

   A login screen appears.
3. Log in to the web interface as `root`.
4. Select **iDRAC settings** on the left navigation bar.
5. Select **Network/Security** on the main top navigation bar.
6. Select **Users** on the secondary top bar.
7. Select the user whose password is changing. For example, userid `2` and username `root`.
8. Select **Configure User**, then **Next**.
9. Enter the new password into the **New Password** and **Confirm New Password** fields.
10. Select **Apply** to complete the password change.

12.11 Configure the Simple Event Correlator (SEC)

The Simple Event Correlator (SEC) is an SMW utility that parses every line being appended to system log files, watching for specific strings that represent the occurrence of significant system events. When a specified string is detected, SEC sends notification that this has happened, either by email, IRC, writing to a file, or some user-configurable combination of all three.

SEC is enabled by default, and by default is configured to generate email notifications to `crayadm`. The types of notifications generated and the recipients to whom notifications are sent are defined in the SEC configuration file, `/etc/opt/cray/cray_sec_actions_config`.

This release includes the Open Source simple event correlator (SEC) package, `sec-2.7.6`, and an SEC support package, `cray-sec-8.0`. The SEC support package contains control scripts to manage the starting and stopping of SEC around a Cray mainframe boot session, in addition to other utilities.

For configuration procedures, see *Configure Cray SEC Software* (S-2542) for release CLE 6.0.UP01 / SMW 8.0.UP01.

12.12 Configure Site Lightweight Log Manager (LLM)

About this task

If this site uses the Lightweight Log Manager (LLM) to send logs from the SMW to a site loghost, use this procedure to update settings in the cray_logging service, which is in the global config set.

Procedure

1. Update the cray_logging service.

   ```
   smw# cfgset update -s cray_logging -m interactive -l advanced global
   ```

2. Update the following settings, as needed.

   ```
   cray_logging.settings.site_loghost.data.name:
   cray_logging.settings.site_loghost.data.ip_protocol: tcp
   cray_logging.settings.site_loghost.data.ip_port: 514
   cray_logging.settings.site_loghost.data.syslog_format: rfc5424
   ```

3. Validate the global config set

   ```
   smw# cfgset validate global
   ```

4. Apply configuration changes, if any.

   Run `cray-ansible` so Ansible plays that consume config set data will apply that data to the SMW.

   ```
   smw# /etc/init.d/cray-ansible start
   ```

12.13 Prevent Unintentional Re-creation of Mail Configuration Files

This procedure is optional. It applies to systems where postfix or sendmail are configured on the SMW.

To prevent the `master.cf` and `main.cf` postfix configuration files from being re-created during software updates or fixes, edit the `/etc/sysconfig/mail` file on the SMW and ensure that the `MAIL_CREATE_CONFIG` setting is set to "no."

```
smw# vi /etc/sysconfig/mail

MAIL_CREATE_CONFIG="no"
```

12.14 Set Variable for Snapshot Name and Make a Snapshot Using snaputil

Prerequisites

This procedure assumes that the SMW has been booted to a release snapshot.

About this task

This procedure uses `snaputil` to make a snapshot. `snaputil` is a Python utility delivered with the cray-install-support RPM, which is installed by default on the SMW. It enables administrators and installers to create and manage snapshots as an easy way to stage upgrades and roll back or switch between SMW/HSS and CLE releases.

Note that snapshots capture content in these three directories: `/`, `/var/lib/mysql`, and `/var/opt/cray/repos`. Used in conjunction with backups of config sets, they provide enough information to be able to re-create the state of the system at a particular time.

Sites can make as few or as many snapshots as they deem useful. Cray recommends making a snapshot of the system at the software installation/configuration milestones shown in step 3.

Procedure

1. List the available snapshots on the system.

   ```
   smw# snaputil list
   ```

2. Set the `SNAPSHOT` environment variable using the currently booted snapshot name.

   Setting a variable for the snapshot name enables better command substitution in later commands dealing with snapshots. The SMW will boot from this snapshot. (SMW HA only) This is especially important for SMW HA systems because it makes it easier to use the exact same snapshot name for both SMWs. Using different snapshots results in HSS database (MySQL) inconsistencies, which causes problems at failover.

   ```
   smw# export SNAPSHOT=$(snaputil list |grep ^cur| awk '{print $2}')
   smw# echo $SNAPSHOT
   ```

3. Create a new snapshot.

```
smw# snaputil create ${SNAPSHOT}.suffix
```

Replace *suffix* with a unique suffix to distinguish among snapshots. Here are suggested suffixes:

Suffix	Associated Milestone
pre-config	after installing a new software release
post-config	after updating the global and CLE config sets
post-boot	after booting the system
post-customize	after customizing a preinstalled system

.

12.15 Back Up Current Global and CLE Config Sets

About this task

Sites can back up the current global and CLE config sets as few or as many times as they deem useful. Cray recommends backing up the config sets at these software installation/configuration milestones.

In the example commands below, replace *suffix* with a unique suffix to distinguish among snapshots. Here are suggested suffixes:

Suffix	Associated Milestone
post-config	after updating the global and CLE config sets
post-boot	after booting the system
post-customize	after customizing a preinstalled system

Procedure

1. Back up the current global config set.

   ```
   smw# cfgset create --clone global global-suffix-${TODAY}
   ```

2. Back up the current CLE config set.

   ```
   smw# cfgset create --clone p0 p0-suffix-${TODAY}
   ```

12.16 Back Up the Newly Installed and Configured SMW/CLE Software

After installing and configuring the new SMW/CLE software, create a backup of it, if needed.

Dell R815 SMW with software RAID	If an R815 SMW was configured using the recommended software RAID1 configuration for the boot disk, there is no need to make a backup of the SMW boot disk because the boot disks are mirrored automatically by the RAID software.

Dell R630 SMW with software RAID	If an R630 SMW was configured using the recommended hardware RAID5 virtual disk, there is no need to make a backup of the SMW boot disk. Any one of the four disks in the RAID5 configuration can fail without losing any data.

13 Troubleshoot SMW/CLE Software Installation

The following procedures address issues that may occur while installing, configuring, or booting Cray System Management Workstation (SMW) and Cray Linux Environment (CLE) software.

13.1 Boot the System with DEBUG

Prerequisites

This procedure assumes that the SMW and CLE have been configured and images have been built and mapped to nodes.

About this task

Because booting with DEBUG is not commonly used, these instructions describe it for a full system (p0) and not for a partitioned system.

Procedure

1. Run rtr to set up the routes.

```
crayadm@smw> rtr -R
```

2. Enable DEBUG boot.

   As root, set debug for the boot parameters of all nodes. This ensures they will stop the boot process allowing a console login.

```
smw# cnode update --add-parameter DEBUG=true "*"
```

3. Boot the boot node.

   a. Boot the boot node.

```
crayadm@smw> xtcli boot DEFAULT -o bootnode c0-0c0s0n1
```

   b. Log in to console.

      If the boot fails, log in to the console to get a shell and debug the problems.

```
smw# xtcon c0-0c0s0n1
```

c. Restart Ansible configuration.

After changing YAML files or other config set files on the SMW, rerun this command on the console.

```
console# /etc/init.d/cray-ansible start
```

4. Boot additional service nodes.

```
crayadm@smw> xtcli boot DEFAULT <node_list>
```

5. Boot compute nodes.

```
crayadm@smw> xtcli boot DEFAULT -o compute p0
```

6. Disable DEBUG boot.

As root, disable debug for the boot parameters of all nodes. This ensures they will not stop the boot process allowing a console login for debugging.

```
smw# cnode update --remove-parameter DEBUG=true "*"
```

14 Miscellaneous Procedures

14.1 Back Up Site Data

When a fresh install is performed on a system, disks are wiped clean. Before beginning any installation procedures, sites should back up configuration files, log files, or other files that need to be preserved. Site may also want to archive important SMW and CLE information even if there are no immediate plans to install or reinstall a software release. Saving such information elsewhere will make a later reinstall easier whether it is planned or part of disaster recovery. This procedure helps sites identify and back up important data from the SMW, boot, and SDB nodes.

What data should be saved at a particular site depends on several things, such as what is currently installed and where data is stored. A site might have CLE 5.x / SMW 7.x or CLE 6.0.UP00 / SMW 8.0.UP00 installed, or it might already have CLE 6.0.UP01 / SMW 8.0.UP01 installed and is now planning to do a fresh install and wants to reuse configuration data files. The information to save would be different in each case. And there could be site data in home directories or other parts of the file system unknown to Cray and therefore not listed here. These suggestions about what data to preserve assume a reinstallation of CLE 6.0.UP01 / SMW 8.0.UP01 that wipes out an earlier installation of UP01.

SMW Data to Save before Reinstalling UP01

SMW Configuration Data

/var/opt/cray/imps	Save the entire directory, which has global config sets (`/var/otp/cray/imps/global`) and CLE config sets (`/var/opt/cray/config/sets/p0`). Saving only the worksheet YAML would miss any site files added for distribution by simple sync or any site Ansible plays. Of particular importance in the global config set is cray_bootraid_config.yaml (or cray_bootraid_worksheel.yaml) which describes how the storage on the Boot RAID is being used.
/etc/	Save the entire directory. Information related to image recipes is stored in `/etc/opt/cray/imps/image_recipes.d` (especially any site changes to `image_recipes.local.json`) and `/etc/opt/cray/imps/package_collections.d`.
/opt/cray/hss/default/etc	Save the boot automation files (`/opt/cray/hss/default/etc/auto.*`) and any other files with custom settings.
/var/opt/cray/repos	Save any site repos which have been created in this directory.
/home/crayadm/*fs_defs	Save this file if direct-attached Lustre (DAL) was configured.

/var/adm/cray/release/pe/install-cdt.yaml	Save the PE installer YAML configuration file.
Command output	Save output from these commands: • Are any nodes disabled? `smw# xtcli status s0` • What are the boot and SDB nodes and are any CLE partitions present? `smw# xtcli part_cfg show`

SMW Operational Data

/home	Save any user data in this directory, especially in `/home/crayadm`.
/var/opt/cray/disk/1	Save all files in this directory, which has logs, dumps, and debugging information.
/var/opt/cray/imps/image_roots and /var/opt/cray/imps/boot_images	No need to save data in these two directories as long as the image recipes are saved, because these files can be rebuilt from the image recipes. And when they are rebuilt, they can be pushed to the boot node or CMC (for eLogin).
/var/lib/mysql	Perform a `mysql dump` of `/var/lib/mysql`. This data will be regenerated by rerunning `xtdiscover`.

CLE Data to Save before Reinstalling UP01

CLE Boot Node Data

/var/opt/cray/imps	No need to save the files in this directory. They are all copies of files on the SMW.
/non-volatile and /cray_home	Save the data in these two directories for possible restoration after the fresh install.

CLE SDB Node Data

/alps_shared and /var/lib/mysql	No need to save the data in these two file systems. It will be regenenerated at the first boot with the newly installed software. The only side effect is that all ALPS apids will start over at apid 100.

14.2 Back Up Current Global and CLE Config Sets

About this task

Sites can back up the current global and CLE config sets as few or as many times as they deem useful. Cray recommends backing up the config sets at these software installation/configuration milestones.

In the example commands below, replace *suffix* with a unique suffix to distinguish among snapshots. Here are suggested suffixes:

Suffix	Associated Milestone
post-config	after updating the global and CLE config sets
post-boot	after booting the system
post-customize	after customizing a preinstalled system

Procedure

1. Back up the current global config set.

```
smw# cfgset create --clone global global-suffix-${TODAY}
```

2. Back up the current CLE config set.

```
smw# cfgset create --clone p0 p0-suffix-${TODAY}
```

14.3 Shut Down the System

If it is necessary to shut down the system, use one of these commands.

full system For a full, unpartitioned system:

```
crayadm@smw> xtbootsys -s last -a auto.xtshutdown
```

partitioned For a partitioned system, with partitions p1 and p2:

```
crayadm@smw> xtbootsys -s last -a auto.xtshutdown -p p1
```

```
crayadm@smw> xtbootsys -s last -a auto.xtshutdown -p p2
```

14.4 Run the xtgetrev and xtshowrev Tools

Prerequisites

To run `xtgetrev`, the boot node must be booted and accessible.

About this task

System administrators use the `xtgetrev` and `xtshowrev` commands to gather and display machine, software revision, Field Notice (FN), and patch set information. The `xtgetrev` command collects information from the administrator and from the SMW and boot node. The `xtshowrev` command displays that information, even when CLE is not running. These tools are useful for gathering information to send to Cray (after a software upgrade, FN, or patch set is installed) and for troubleshooting.

This procedure describes how to install these two tools on a Cray XC Series system. These steps (except for running `xtshowrev`) must be executed as root.

> **ATTENTION:** Any information that is submitted to site_install_data@cray.com will only be used internally and will not be made public. The `xtshowrev` command does not submit any information to Cray automatically.

Procedure

1. Load the module to start using the tools.

```
smw# module load xtshowrev
```

2. Run `xtgetrev` to create and populate the initial files.

   Only root can run this command. The first time `xtgetrev` is executed, when there are no files populated, the tool will prompt for site information. If the boot node does not have passwordless ssh, then the tool will prompt for the password.

```
smw# xtgetrev
xtgetrev: No site information has been defined.

Site name: CRAY/INTERNAL
Serial Number: 9999
System Name [panda1]:
System Type [XC40]:
Install Date [2016-06-01]):
OS Type [CLE]:
/etc/opt/cray/release/pkginfo/site_config written.
No SMW information has been defined.
Install Date for SMW 8.0.UP01 [2016-06-01]):
/etc/opt/cray/release/pkginfo/smw_release written.
Accessing boot

No CLE information has been defined.
Which WLM is in production?
   1) PBS
   2) Moab/TORQUE
   3) SLURM
   4) LSF
   5) WLM
   6) None
   7) Skip
Choose?[7] 7
Install Date for CLE 6.0.UP01 [2016-06-01]:
/etc/opt/cray/release/pkginfo/cle_release written.
/etc/opt/cray/release/pkginfo/sec_release written.

No hardware inventory information has been defined.
Reading /opt/tftpboot/hwinv.p0 ...done
/etc/opt/cray/release/pkginfo/hwinv written.
```

3. Run `xtshowrev` to see the formatted information.

   Any user can run this command.

```
smw# xtshowrev
Site:                CRAY/INTERNAL
```

```
S/N:                  9999
System Type:          XC40
Install Date:         2016-06-01
System Name:          panda1
CNL/CLE Release:      6.0.UP01
XT Release:           6.0.96
CLE Kernel:           3.12.51-52.31.1_1.0600.9146
CLE OS:               SLES12
CLE Patch Sets:       01 02 03
CLE FNs:
Lustre Version:       2.5.4
OS Type:              CLE
SMW Release:          8.0.UP01
SMW Build:            8.0.96
HSS Release:          8.0__446__ge75851a-49.1
SMW Kernel:           3.12.51-52.39
SMW OS:               SLES12
SLE Patch Sets:
SMW Patch Sets:
SMW FNs:              5844c
SEC Release:          Cray_SEC 8.0__6__g689802a (sec 2.7.6)
Current Date:         2016-06-01 12:59:21
crayadm@smw>
```